HAL
open science

Environnement de réalité augmentée 3D coopératif : approche colocalisée sur table

Raphaël Grasset

To cite this version:

Raphaël Grasset. Environnement de réalité augmentée 3D coopératif : approche colocalisée sur table. Interface homme-machine [cs.HC]. Université Joseph-Fourier - Grenoble I, 2004. Français. NNT : . tel-00528881

HAL Id: tel-00528881

https://theses.hal.science/tel-00528881

Submitted on 22 Oct 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Université Grenoble I - Joseph Fourier

Environnement de réalité augmentée 3D coopératif : approche colocalisée sur table

Raphaël Grasset

Thèse présentée pour l'obtention du titre de :

Docteur de l'Université Joseph Fourier

Spécialité Informatique

Arrêté ministériel du 5 juillet 1984 et du 30 mars 1992

Préparée au sein du laboratoire :

ARTIS-GRAVIR/IMAG-INRIA (UMR CNRS C5527)

Soutenue le 19 Avril 2004

Composition du jury :

Bruno	Arnaldi	Président
Christophe	Chaillou	Rapporteur
Pascal	Guitton	Rapporteur
Sabine	Coquillart	Examinateur
Dieter	Schmalstieg	Examinateur
Claude	Puech	Directeur de thèse
Jean-Dominique	Gascuel	Co-directeur de thèse

Dédié à..

Remerciements

...

Résumé

La Réalité Augmentée (RA) peut être définie comme le mélange du monde réel et du monde virtuel. Basée sur ce concept, cette thèse s'intéresse à l'application de la RA dans un cadre de collaboration pour des tâches 3D. Nous nous sommes plus particulièrement concentrés sur le concept de réunion autour d'une table dans lequel plusieurs personnes peuvent manipuler simultanément des éléments virtuels 3D dans un espace partagé.

L'étude et l'analyse des besoins a permis de mettre en place un environnement nommé *MARE* (*Multiuser Augmented Reality Environment*) qui fournit un cadre d'étude, d'expérimentations et de développement d'applications de RA collaboratives 3D. Ce travail a dégagé un certain nombre de contributions :

1. **la définition d'une architecture logicielle et une nouvelle plate-forme matérielle hétérogène** offrant un intérêt particulier de modularité, d'extensibilité, d'ergonomie et de temps réel ;
2. **de nouvelles techniques de calibrage statique simples et intuitives pour l'alignement réel/virtuel** basées sur une approche guidée par l'utilisateur ;
3. **différentes métaphores d'interactions pour la manipulation d'objets virtuels** avec une décomposition dimensionnelles des tâches applicatives, mais aussi avec un nouveau concept d'interface hybride haut niveau (documents virtuels) ;
4. **un nouveau type d'interaction hybride entre éléments réels et virtuels**, avec un prototype de peinture virtuelle interactive sur des objets réels ;
5. **l'utilisation des travaux liés au collaboratif dans un cadre de RA 3D**, en proposant ainsi différentes méthodes pour la protection des données mais aussi l'assistance utilisateur ou la coordination des actions.

En complément, des expérimentations avec notre système et de nombreuses évaluations nous ont servi à montrer l'intérêt de cette approche. Finalement, ce travail permet de définir une première base de critères nécessaires pour une mise en œuvre efficace d'environnement de réalité augmentée collaboratif 3D sur table.

Table des matières

1. Introduction **15**
1. Contexte : collaboratif et tâches 3D, quelles solutions ? . . . 15
2. Motivations . . . 17
3. Objectifs et approche . . . 17
4. Organisation du document . . . 18

2. Présentation de la problématique : définitions et approche **19**
1. Réalité augmentée : une introduction . . . 19
   1.1. Définitions . . . 19
   1.2. Taxonomies . . . 20
   1.3. Bref historique . . . 22
   1.4. Mise en œuvre . . . 23
      1.4.1. Approche classique . . . 24
      1.4.2. Technologies émergentes . . . 25
2. Collaboratif : introduction et portée . . . 26
   2.1. Définition et contexte . . . 26
   2.2. Travail collaboratif et informatique : définitions et intérêts . . . 28
   2.3. Quelques exemples de collecticiels . . . 29
3. Collaboratif et tâches 3D . . . 31
   3.1. Travail collaboratif et informatique : contenu 3D et environnement virtuel . . . 31
   3.2. Quelques exemples d'environnements virtuels collaboratifs . . . 33
4. Cas restreint : travail collaboratif colocalisé synchrone . . . 34
   4.1. Généralités . . . 34
   4.2. Type d'activité spécifique : la réunion, intérêt et classification . . . 35
   4.3. Tâches 3D colocalisées : spécificités . . . 36
5. Implications de la réalité augmentée dans le cadre collaboratif 3D colocalisé synchrone . . . 36
6. Approche choisie : environnement générique . . . 37
   6.1. Environnement : intérêt et définition . . . 37
   6.2. Méthode : approche globale . . . 38
7. Conclusion . . . 39

3. Environnements collaboratifs colocalisés : état de l'art **41**
1. Introduction . . . 41
2. Étude des systèmes existants . . . 42
   2.1. Systèmes basés réalité virtuelle . . . 43
      2.1.1. Environnements à grande audience : vers des surfaces verticales . . . 43
      2.1.2. Le projet *Continuum* . . . 44
      2.1.3. *CAVE* . . . 45
      2.1.4. Système hybride : le *PIT* . . . 46
      2.1.5. Approche table 3D . . . 46
      2.1.6. Approche individuelle : casques ou écrans côte à côte . . . 47
      2.1.7. Synthèse des approches RV . . . 48

2.2. Systèmes basé IHM : informatique ubiquitaire et évanescente 48
2.2.1. *Colab* 49
2.2.2. *I-Land* et *Ambiente* 49
2.2.3. *Interactive Workspace* 50
2.2.4. *Interactive Space* et *WorkSPACE* 51
2.2.5. *LiveBoard* 51
2.2.6. SDG (Single Display Groupware) 52
2.2.7. Tables digitales 52
2.2.8. *PDH* (*Personal Digital Historian*) 53
2.2.9. *Peebles* 53
2.2.10. Synthèse des approches IHM 54
2.3. Systèmes basé Réalité Augmentée 2D 54
2.3.1. *Augmented Space* 54
2.3.2. *Build-It* 55
2.3.3. *OFuturO* 56
2.3.4. Synthèse des approches basés RA 2D 56
2.4. Systèmes basés réalité augmentée 3D 57
2.4.1. *Seamless Design* 57
2.4.2. *EMMIE* 58
2.4.3. Studierstube 58
2.4.4. *Shared Space* et *Tiles* 59
2.4.5. *ARTHUR* 60
2.4.6. *MixDesign* 61
2.4.7. *MagicMeeting* 62
2.4.8. *DWARF* et *Sheep* 62
2.4.9. *SCAPE* 63
2.4.10. Synthèse des approches RA 3D 64
3. Synthèse globale 64
4. Conclusion 67

4. Architecture **69**
1. Introduction 69
2. Architecture matérielle 69
2.1. Introduction 69
2.2. Besoins et approche 70
2.3. Support communicatif : la table 71
2.3.1. Introduction et besoins 71
2.3.2. Approche : caractéristiques et ergonomie 71
2.4. Base visuelle 74
2.4.1. Besoins 74
2.4.2. Approche 75
2.5. Connaître le contexte de l'utilisateur 77
2.6. Base interactive : orienté interfaces de réunion 79
2.7. Retour et enregistrement sonore 83
2.8. Support de calcul : architecture centralisée 84
2.9. Mise en œuvre et résultats 85
2.10. Synthèse 88
3. Architecture logicielle 89
3.1. Introduction 89
3.2. Définitions et besoins 90
3.3. Etat de l'art 91

3.3.1. Bibliothèques et boîte à outils . . . 91
3.3.2. Canevas logiciels objets . . . 92
3.3.3. Canevas logiciel à composants . . . 92
3.3.4. Outils auteurs . . . 93
3.3.5. Librairies spécialisés . . . 93
3.3.6. Gestion de contenu 3D . . . 94
3.4. Notre approche : modularité et hétérogénéité . . . 94
3.4.1. Vue générale . . . 95
3.4.2. Principe de conception . . . 95
3.4.3. Présentation . . . 96
3.4.4. Description des fonctionnalités . . . 97
3.4.5. Outils auteur et développement . . . 100
3.5. Discussion . . . 101
4. Conclusion . . . 102

5. Mixage : virtuel et réel 103
1. Introduction . . . 103
2. Integration spatiale : concepts et définitions . . . 103
3. Erreur statique : calibrage . . . 105
3.1. Etat de l'art : analyse . . . 105
3.2. Notre Approche : vers le *plug'n play* . . . 105
3.2.1. Modélisation de notre configuration . . . 106
3.2.2. Calibrage des systèmes de suivi . . . 107
3.2.3. Calibrage écrans : groupe et public . . . 109
3.2.4. Calibrage casques : optique et vidéo . . . 110
3.2.5. Calibrage des périphériques d'interactions : pointeur et tablette . . . 115
3.2.6. Calibrage du repère locale ($T_{W \leftarrow L}$) . . . 116
3.2.7. Interface . . . 116
3.3. Résultats et Discussion . . . 116
4. Erreur dynamique : réduction de la latence . . . 117
4.1. Approche : pipeline de haute qualité et algorithme prédictif . . . 118
4.2. Résultats et discussion . . . 119
5. Approche globale : changer l'environnement . . . 119
5.1. Problème . . . 119
5.2. Approche . . . 120
5.3. Mise en œuvre . . . 122
5.4. Résultats, évaluation et discussion . . . 122
6. Premier pas vers une digitalisation interactive du réel par esquisse . . . 123
6.1. Introduction . . . 124
6.2. Approche : vers une esquisse du réel en RA . . . 125
6.2.1. Modèle polygonal . . . 125
6.2.2. Modèle courbe . . . 126
6.2.3. Modèle quelconque . . . 127
6.3. Discussion . . . 128
7. Conclusion . . . 129

6. Interaction avec le virtuel : tâches de base 131
1. Introduction . . . 131
2. Interaction : terminologie et taxonomies . . . 131
2.1. Définitions . . . 132
2.2. Taxonomies . . . 132

2.3. Méthodes de conception . . . 133
2.4. Spécificités . . . 134
2.4.1. Spécificités de l'interaction 3D . . . 134
2.4.2. Spécificités de l'interaction en réalité augmentée . . . 135
2.4.3. Spécificités de l'interaction colocalisé sur table . . . 136
2.5. Notre approche . . . 137
3. Interaction : vers «l'hybridité» . . . 137
3.1. Manipulation . . . 137
3.1.1. État de l'art . . . 137
3.1.2. Approche choisie . . . 138
3.1.3. Techniques d'interaction : tâche 2D sur table . . . 140
3.1.4. Technique d'interaction : tâche 2D 1/2 sur table . . . 147
3.1.5. Technique d'interaction : tâche 3D sur table . . . 150
3.1.6. Conclusion, limitations et perspectives . . . 156
3.2. Navigation . . . 157
3.2.1. Etat de l'art . . . 157
3.2.2. Notre approche . . . 158
3.2.3. Technique d'interaction : navigation locale . . . 158
3.2.4. Technique d'interaction : navigation globale . . . 160
3.2.5. Technique d'interaction : vues multiples et transition RA/RV 161
3.3. Contrôle d'application . . . 161
3.3.1. Etat de l'art : analyse . . . 161
3.3.2. Notre approche . . . 162
3.4. Tâches spécifiques : ajout d'éléments et saisie de texte virtuel . . . 164
3.4.1. Ajout d'élément virtuel . . . 164
3.4.2. Saisie de texte alphanumériques pour annotations . . . 164
4. Conclusion . . . 165

7. Interaction virtuel : multiples espaces et multiples surfaces 167
1. Introduction . . . 167
2. Analyse et approche . . . 168
3. Etat de l'Art . . . 168
3.1. Extension du Bureau numérique . . . 168
3.2. Approche informatique ubiquitaire . . . 169
3.3. Approche RV . . . 169
3.4. Approche RA . . . 169
4. Nouvelle approche : un autre pas vers l'hybridité . . . 170
4.1. Ontologie : multiples surfaces et multiples espaces . . . 170
4.2. Description plus spécifique des fenêtres . . . 173
4.3. Couplage : approche multiniveaux et générique . . . 177
5. Mise en œuvre . . . 179
6. Résultats et applications . . . 180
6.1. Premier prototype . . . 180
6.2. Re-interpretation de prototypes de recherche . . . 182
6.3. Applications . . . 183
7. Limitations et perspectives . . . 185
8. Conclusion . . . 185

8. Interaction avec le réel **187**
1. Introduction et concept . . . 187
2. Etat de l'art . . . 189
3. Le pipeline de l'*interactive mediated reality* . . . 190
4. Modification local d'un objet : peindre, coller étaler . . . 191
4.1. Approche : métaphore du studio virtuel . . . 192
4.2. Architecture . . . 192
4.3. Prototype . . . 194
4.3.1. Modification de l'objet : au niveau du pixel . . . 194
4.3.2. Modification de l'objet : au niveau du patch . . . 197
4.3.3. Modification de l'objet : au niveau de la matière . . . 197
4.4. Résultats et applications . . . 199
4.5. Applications à des situations réeles . . . 200
4.6. Etude préliminaire . . . 202
4.6.1. Tâche . . . 203
4.6.2. Résultats . . . 203
4.7. Etude principale . . . 204
4.7.1. Tâche . . . 204
4.7.2. Protocole . . . 205
4.7.3. Résultats et Discussion . . . 205
5. Conclusion . . . 207

9. Aspects collaboratif **209**
1. Introduction . . . 209
2. Contexte collaboratif . . . 209
3. Approche et mise en œuvre . . . 212
3.1. Respect des mécanismes de collaboration . . . 212
3.1.1. Communication explicite . . . 212
3.1.2. Communication conséquente . . . 213
3.1.3. Coordination des activités . . . 214
3.1.4. Protection . . . 216
3.1.5. Assistance . . . 218
3.1.6. Surveillance d'activités . . . 220
3.1.7. Planification . . . 220
3.2. Respect des besoins pour un travail collaboratif co-localisé . . . 221
4. Conclusion . . . 221

10. Applications et retours d'éxpériences **223**
1. Retour d'éxpériences . . . 223
2. Application : ludique . . . 224
2.1. Contexte . . . 224
2.2. Etat de l'art . . . 224
2.3. Approche . . . 225
2.4. Prototypes et résultats . . . 226
3. Application scientifique : chimie . . . 227
3.1. Contexte . . . 227
3.2. État de l'art . . . 227
3.3. Approche . . . 228
3.4. Prototypes et résultats . . . 229
4. Application : architecture . . . 230
4.1. Contexte . . . 230
4.2. État de l'art . . . 230

4.3. Approche . . . 231
4.4. Prototypes et résultats . . . 232
5. Conclusion . . . 234

11. Conclusion **237**
1. Contributions . . . 237
2. Perspectives . . . 238
3. Bilan et réflexion . . . 239

A. MARE : détails techniques de la réalisation **241**
1. Historique . . . 241
2. Architecture matérielle . . . 242
2.1. Mobilier : la table d'interaction . . . 242
2.2. Visualisation : casques, écrans et projection vidéo . . . 244
2.3. Systèmes de suivi . . . 246
2.4. Interfaces . . . 251
2.5. Micros et haut parleurs . . . 254
2.6. Support de calcul : architecture centralisé . . . 255
3. Architecture Logicielle . . . 256

B. Questionnaire pour l'évaluation de l'interactive mediated reality et l'application à l'architecture **259**
1. Questionnaire sur l'évaluation avec le système de peinture . . . 259
2. Questionnaire pour l'évaluation en architecture . . . 263

Introduction

> Do we think of ourselves as working primarily in the computer but with access to physical world functionnality, or do we think of ourselves as working primarily in the physical world with access to computer functionnality ?
>
> — *The Digital Desk* (1993). Pierre Wellner.

L'interaction avec des objets tridimensionnels est devenue un point prépondérant durant ces dernières années. En effet, les progrès technologiques permettent aujourd'hui d'afficher sur de simples machines des objets à géométrie complexe avec un rendu de très haute qualité. Ce développement a alors induit un large besoin de manipulation de ces objets.

Depuis les années 80, un grand nombre de techniques et de concepts a été proposé pour interagir avec les objets virtuels 3D. Mais ce problème reste largement ouvert, la définition de techniques simples et intuitives reste une question fondamentale (figure 1.1). En effet, on remarque que malgré la diversité des solutions proposées, peu de nouvelles techniques sont aujourd'hui utilisées dans des logiciels commerciaux. Elles restent en général basées sur des métaphores très limitées par rapport aux capacités humaines. Elles reposent principalement sur la station de travail, avec souris et clavier comme unique périphérique d'interaction.

(a) (b)

Fig. 1.1: Interfaces du futur ? (a) *Haptic Workstation* [Worb] et (b) le film *Minority Report*© [FD02].

Ceci est d'autant plus problématique que l'on souhaite maintenant une *interaction collaborative*, de multiples utilisateurs interagissant sur le même contenu, telle qu'on le rencontre dans des activités professionnelles ou de loisirs. Permettre à l'utilisateur d'interagir et de collaborer dans un cadre de tâche 3D de façon très intuitive complexifie notablement la mise en œuvre de solutions pour l'interaction 3D.

1. Contexte : collaboratif et tâches 3D, quelles solutions ?

Nous remarquons que dans les activités professionnelles, l'échange d'idées et la coordination du travail nécessitent un fort degré d'interaction collaborative (figure 1.2). Elle se présente sous les formes les plus diverses : réunion, présentation, compte-rendu, *brainstorming*. Dans nos activités de loisirs, la collaboration prend de même une place importante,

autant dans les activités de jeux (jeux de plateau, jeux de rôle, jeux vidéo), que dans les activités sportives (jeux de groupe).

(a) réunion de travail (b) discussion informelle (c) jeu de rôle en petit groupe

Fig. 1.2: Activités collaboratives : professionnelles et ludiques.

La visualisation 3D est quant à elle devenue un élément de travail dans un grand nombre d'activités professionnelles. Une visualisation spatiale, une représentation réaliste, la navigation autour d'éléments conceptualisés constituent quelques uns des nombreux avantages d'une représentation tridimensionnelle. Dans le domaine des loisirs, il est incontestable que la place prise par l'utilisation de scène 3D dans les activités du jeu vidéo augmente rapidement.

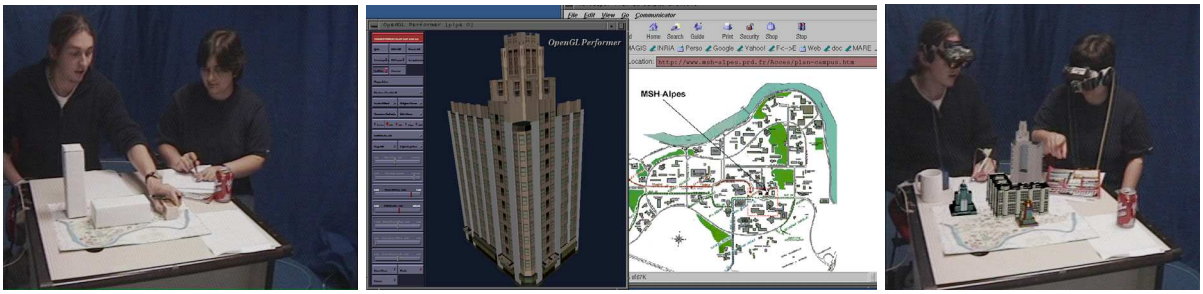
Malheureusement peu de solutions commerciales fournissent un bon support d'interaction collaborative avec des éléments 3D (figure 1.3). Les approches de type système d'affichage par projection supportent en général peu d'utilisateurs en forte interaction et nécessitent de travailler dans des conditions trop limitantes. Ces systèmes offrent peu de luminosité, en général un seul utilisateur dont le mouvement est suivi et des outils d'interaction difficilement manipulables. Les approches de type station de travail semblent aussi très contraintes : interaction alternative limitée à la souris, pas de visualisation stéréoscopique multi-utilisateurs, pas de vue personnelle et des difficultés à fournir des outils et des informations personnalisées.

(a) deux utilisateurs en collaboration dans un espace immersif (b) application d'architecture sur table virtuelle (c) travail devant un écran de bureau

Fig. 1.3: Systèmes commerciaux pour travail collaboratif 3D : quelle approche ? (a) espace immersif [CNSA+92], (b) table avec affichage par rétro-projection [ABFH97] (c) station de travail.

La Réalité Augmentée (RA) constitue peut être une première solution à ces problèmes. Elle permet d'intégrer des objets virtuels dans notre environnement réel et de pouvoir les manipuler naturellement (figure 1.4). Basée sur ce concept, un utilisateur peut conserver son environnement collaboratif habituel et intégrer les possibilités d'interaction 3D.

Cette thèse trouve son contexte d'étude dans l'utilisation de la réalité augmentée dans un cadre collaboratif. Plus particulièrement, nous nous intéresserons au support du cadre de



(a) (b) (c)

Fig. 1.4: Une nouvelle approche : réalité augmentée et environnement collaboratif sur table (a) une situation réelle (b) une application 3D de bureau (c) son intégration dans notre environnement.

réunion dans un même lieu, correspondant au schéma colocalisé synchrone (termes définis ultérieurement dans le chapitre 2, section 2.2).

2. Motivations

L'utilisation de l'espace réel pour communiquer assure une conservation des métaphores naturelles de communication et un support proprioperceptif : en Réalité Virtuelle (RV) il est généralement nécessaire de les reproduire. Or, les nombreux travaux menés à ce jour en RV ont montré une grande limitation dans cette «virtualisation» des métaphores naturelles de communications. Les solutions sont très limitées et sont encore loin des métaphores réelles (voir [GV99]). La réalité augmentée conserve les métaphores naturelles, mais en complément les «augmente» par diverses propriétés.

On note de plus que graçe à la préservation du cadre visuel réel, la RA pourra tirer parti des possibilités d'utilisation d'éléments réels tangibles. Et elle pourra aussi se baser sur des métaphores gestuelles pour l'interaction 3D.

Une solution de type RA s'applique à tout domaine collaboratif nécessitant une visualisation tridimensionnelle. Nous pouvons citer : réunion scientifique, architecture, géologie, éducation, loisir, etc.

Dans ce contexte, incorporer le virtuel au monde réel peut induire un certains nombres de questions : «Comment manipuler le virtuel ?», «Comment gérer l'interaction simultanée avec le virtuel ?», «Comment gérer les relations réel/virtuel dans un cadre collaboratif ?» etc.

Une autre motivation est l'absence aujourd'hui de système complet qui permette de proposer de nombreuses applications tridimensionnelles en réalité augmentée dans un cadre de réunion. La mise en place d'un environnement nous a donc paru un aspect intéressant à approfondir. De plus, il nous semble que la table peut fournir un support de tâche intéressant (référentiel spatial, support physique) qui sera un point d'étude privilégié de nos travaux. L'usage de cet élément a été peu étudié à ce jour en interaction 3D.

3. Objectifs et approche

Cette thèse aura plusieurs objectifs principaux :

- **identifier les besoins** d'une réunion collaborative dans le cadre d'applications tridimensionnelles. Identifier l'intérêt et l'implication de la réalité augmentée dans ce cadre. Identifier les propriétés et cadre applicatif d'une approche par réalité augmentée par rapport aux autres systèmes collaboratifs ;
- **fournir une architecture** matérielle et logicielle générique répondant à cette analyse. Elle devra nous procure un système d'expérimentation de réalité augmentée collaboratif

sur table ;
- étudier et proposer des **métaphores d'interaction** avec les éléments virtuels et réels construit sur cette architecture. Les aspects ergonomiques et collaboratifs sont bien entendu des éléments clefs à respecter ;
- **évaluer** l'environnement proposé par la réalisation de plusieurs prototypes applicatifs. Il pourra être réaliser sous la forme d'une collaboration avec des spécialistes d'un domaine ;
- **définir des critères** servant de guide pour la création d'environnement collaboratif de RA.

Comme le lecteur l'observera au cours de cette étude, la réalité augmentée provient de deux domaines bien distincts : le domaine de l'image de synthèse et le domaine de l'interaction homme-machine (IHM). L'approche retenue pour ce travail a été de tirer parti de ces «deux mondes». Nous avons cherché à en extraire une nouvelle approche qui garde à l'esprit les besoins et méthodes des deux environnements (biaisée dans notre cas par une forte dominante de l'image de synthèse).

Au delà de l'apport de collaborations externes, nous avons choisi de mener une étude bibliographique d'une grande ampleur sans rester confiné à notre domaine de prédilection. Cette étude permet d'avoir une base solide pour assurer la sûreté de ce travail.

4. Organisation du document

Ce document est divisé en 10 chapitres. Le chapitre suivant (chapitre 2) décrit la problématique et l'approche d'étude choisies. Il sera suivi d'un chapitre (chapitre 3) centré sur l'état de l'art des systèmes collaboratifs manipulant du contenu virtuel et leurs spécificités.

Le premier chapitre de contribution (chapitre 4) nous amènera à décrire l'architecture matérielle et logicielle mise en place, répondant aux besoins définis dans le chapitre précédent. Le chapitre 5 décrira les techniques mises en œuvre pour satisfaire les conditions d'un système de réalité augmentée : un mixage «suffisant» entre le réel et le virtuel.

Puis les chapitres suivants auront trait à la description de nouvelles méthodes collaboratives d'interaction. Nous présenterons des nouvelles techniques et approches pour interagir avec le virtuel (chapitre 6), mais aussi avec le réel (chapitre 8), basées sur la définition d'un cadre générique et sa mise en pratique.

Le dernier chapitre de contributions (chapitre 9) présentera la mise en œuvre de l'aspect collaboratif de notre environnement basé sur un canevas établi par les spécialistes du domaine collaboratif. Enfin, l'avant dernier chapitre (chapitre 10) présentera des applications de notre système, servant de premier cadre de validation de l'environnement proposé.

Nous conclurons par un résumé des contributions et par quelques pistes de recherche futures. Enfin, différentes annexes proposent un état de l'art de la RA et détaillent des parties techniques de notre système.

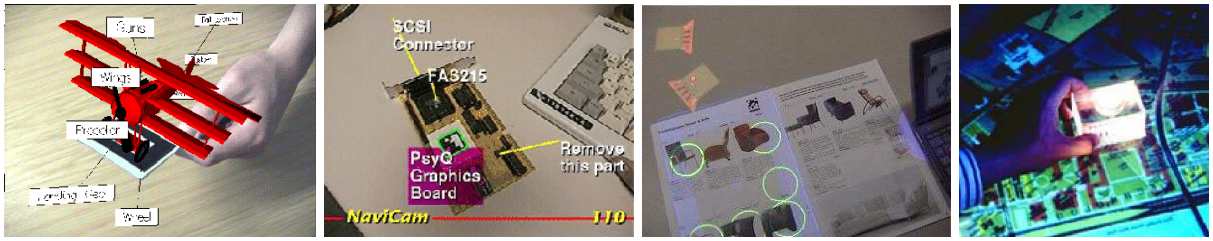
Chapitre 2

Présentation de la problématique : définitions et approche

Nous nous intéresserons dans ce chapitre à introduire et à définir précisement les notions clefs sur lesquelles nous allons nous appuyer pour notre étude sur la réalité augmentée et son utilisation dans un cadre collaboratif colocalisé sur table.

Ce chapitre serra décomposé en six sections sous forme d'une gradation vers le contexte d'utilisation de la RA dans un cadre très précis. On commencera par une présentation de la réalité augmentée (section 1), une présentation du «collaboratif» (section 2), une présentation du collaboratif lié à des tâches 3D (section 3), une présentation du cas colocalisé en section 4 (avec le cas de tâche 3D), puis des hypothèses de l'usage de la RA sur ce dernier point (section 5) et enfin l'approche méthodologique choisie (section 6).

1. Réalité augmentée : une introduction



(a) manipulation dans l'espace réel d'un objet virtuel

(b) superposition d'annotations virtuelles sur un objet réel

(c) projection d'objets virtuels sur un catalogue réel

(d) interaction grâce a un objet réel sur du contenu virtuel

Fig. 2.1: La réalité augmentée en images : quelques exemples.

1.1. Définitions

Intéressons-nous tout d'abord à l'étymologie du terme réalité augmentée (figure 2.1). Le Petit Robert (2002) nous donne comme définition :

> **réalité** n. f. bas lat. *relitas*->rien. **1.** caractère de ce qui est réel, de ce qui ne constitue pas seulement un concept, mais une chose, un fait. ... **3.** ce qui est réel, actuel, donnée comme tel à l'esprit.
>
> **augmentée** vr tr. augmenter **1.** rendre plus grand, plus considérable, par addition d'une chose de même nature. ...

A partir de ces deux définitions, différents éléments peuvent être directement interprétés :
- l'augmentation de la réalité consiste à ajouter des objets de même nature, dans le sens d'objets concrets ;
- on ajoute des choses à l'environnement qui prend plus d'importance, des choses qui agrandissent la réalité, dans le sens où cette augmentation nous conduit à surpasser ses limitations et donc étendre avec des choses difficilement concevables avec des moyens purement réels ;
- ou plus simplement, l'incrémentation d'un contenu perçu de façon identique par l'esprit.

Dans la bibliographie, on note que le terme *réalité augmentée* a été introduit par Caudel [CM92] pour la première fois. A partir de la définition des environnements virtuels, Azuma - dans son état de l'art de la réalité augmentée [Azu97] - énonce «*AR allows the user to see the real world, with virtual objects superimposed upon or composited with the real world. Therefore, AR supplements reality rather than completely replacing it*»[1]. Il propose de définir la réalité augmentée en se reposant sur trois critères : *mélange réel/virtuel, temps réel, alignement réel/virtuel* (on note que l'alignement n'est plus uniquement 3D comme il le souligne dans l'extension de son état de l'art [ABB+01], mais complètement multimodale). Feiner dans un article [FMS93] déclare «*There are many situations, however, in which we would like to interact with the surrounding real world. An augmented reality can make possible by presenting a virtual world that enriches, rather than replaces, the real world*»[2].

De nombreux auteurs ont donné des définitions similaires à la réalité augmentée où l'on va chercher principalement à augmenter la vision de l'utilisateur. Ces définitions sont prises dans un contexte de représentation et proviennent du domaine des images de synthèse où la notion de visualisation est la composante principale.

Dans le domaine de recherche de l'interaction homme-machine (IHM ou *CHI* en anglais), différentes notions ont été introduites suite aux travaux de références de Wellner [Wel91] (on pourra se reporter à la thèse de Dubois [Dub01] qui souligne très bien les deux approches dans son manuscrit). Wellner, sans introduire directement le terme de réalité augmentée, considère le fait d'augmenter les possibilités des documents papiers réels avec les moyens numériques (Mackay le définit comme RA dans [MVC+93]). Par la suite, de nombreux travaux s'inspireront de ce nouveau type d'augmentation, le concept prédominant restant d'offrir une meilleure accessibilité au contenu numérique (virtuel) en utilisant comme interface le réel - et ses possibilités d'appréhension.

Enfin, récemment utilisée comme support applicatif, la RA se définit dans le domaine de la vision par le mélange de l'image réelle avec du contenu virtuel, les limitations temps réel n'étant pas étudiées (on parle aussi de RA pour des séquences vidéo en post-production).

Dans notre cas, nous retiendrons une définition au sens large : *«le fait de mélanger éléments virtuels et éléments réels dans un même contexte»*. Pour situer plus précisément cette définition, nous nous nous reportons aux taxonomies proposées dans la littérature permettant d'identifier la réalité augmentée par rapport à d'autres éléments.

1.2. Taxonomies

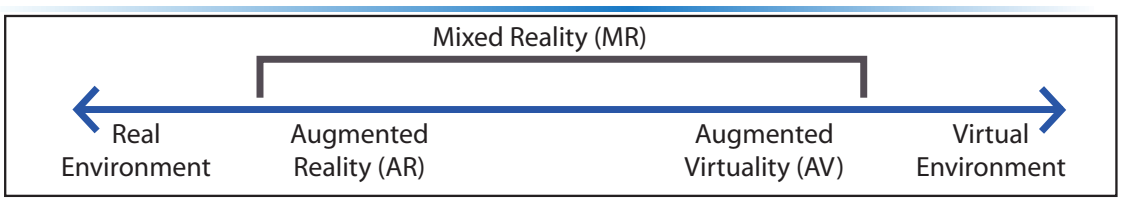


Fig. 2.2: Continuum de Milgram.

Milgram (figure 2.2) interprète la réalité augmentée incluse dans un continuum linéaire qui va du réel au virtuel [MK94]. Il définit le terme de *réalité mixte (RM, mixed reality (MR) en anglais)* comme l'intervalle entre le réel et le virtuel. Cette réalité mixte contient la réalité augmentée qui consiste intégrer du virtuel dans le monde réel, mais aussi la

[1] «La réalité augmentée autorise l'utilisateur à voir le monde réel, avec des objets virtuels superposés au dessus ou composé avec le monde virtuel. La RA complète la réalité, plutôt que de la remplacer complètement»

[2] «il y'a beaucoup de situations pourtant, dans lesquelles on souhaiterait pouvoir interagir avec notre environnement réel. La *réalité augmentée* permet de rendre cela possible en présentant un monde virtuel qui enrichit au lieu de remplacer le monde réel»

virtualité augmentée qui consiste à intégrer du réel dans le monde virtuel. Il étend cette taxonomie des systèmes d'affichages de RA en se basant selon trois principes fédérateurs : portée de la connaissance du monde, fidélité de reproduction, portée de la métaphore de présence.

MacIntyre définit plus précisément le terme de réalité mixte par rapport au terme de réalité augmentée [Mac] : « *The key ingredient for an AR or MR experience is to have a virtual experience that is tightly integrated with the physical world ; in AR, that is typically 3D graphic registered with the world, and MR extends the scope to include such things as computer-enhanced devices*»[3].

Du point de vue de l'IHM, Mackay [Mac96] propose une classification basée sur le contenu augmenté : l'utilisateur, l'objet ou l'environnement. Berard [Bér94] propose quant à lui un espace sous deux axes (action, objet) et définit les termes d'interface augmentée en entrée et en sortie. Dubois analyse très précisément les différentes approches dans sa thèse [Dub01] et propose sa propre taxonomie [DNT+99] basée sur l'objet de la tâche et le type d'augmentation (figure 2.3). L'*objet de la tâche* peut alors être une entité réelle (réalité augmentée, «quantité» de virtuel ajoutée au réel) ou une entité virtuelle (virtualité augmentée, «quantité» de réel ajoutée au virtuel). Le *type d'augmentation* peut être une exécution augmentée (action nouvelle, amélioration des possibilités) ou une évaluation augmentée (ajout d'informations non perceptibles). Il propose d'utiliser le terme système mixte, moins ambigu et redéfinit un équivalent de la représentation I^3 (Interaction, Immersion et Imagination, voir figure 2.14) utilisée comme symbolique de la réalité virtuelle avec les termes interaction, enrichissement et monde réel pour les systèmes mixtes.

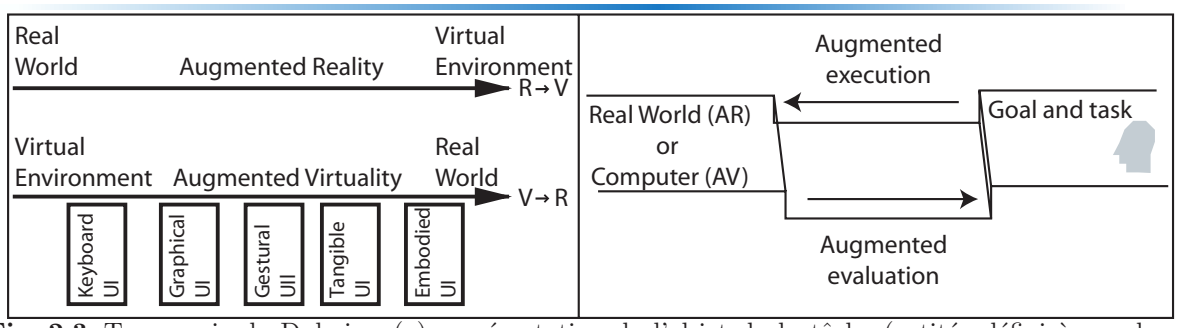


Fig. 2.3: Taxonomie de Dubois : (a) représentation de l'objet de la tâche (entitée défini à gauche et augmentée vers la droite à partir de différentes métaphores d'interaction possibles) (b) représentation du type d'augmentation (échange bidirectionnel entre un monde et un utilisateur).

De même, inscrit dans une optique de spécification de système d'intervention médicale assisté par ordinateur, il propose la notation *ASUR* (raffinement de sa notation *OPAS* [DNT+99]), première modélisation formelle d'un système de Réalité Augmentée [DGN02]. *ASUR* repose principalement sur quatre éléments : le système informatique (S), l'utilisateur (U), les objets réels de la tâche (R) et les adaptateurs (A), éléments liens entre les entités informatiques et le monde réel. Ces éléments mis en relation et respectant un certain nombre de règles permettent de décrire un système de réalité augmentée et d'en avoir une vision globale.

Le terme de réalité augmentée est généralement associé à une multitude de concepts dérivés ou similaires (dans le sens d'une relation réel/virtuel). Nous donnons ici quelques termes usuels en identifiant leur relation par rapport à la réalité augmentée, ce classement tentant de fixer les idées sur la variété des interprétations de la bibliographie. Nous pouvons

[3] L'ingrédient clef pour une éxpérience de RA ou de RM est d'avoir un expérience virtuelle qui est fortement intégré dans le monde physique ; en RA, c'est typiquement par alignement graphique 3D avec le monde et en RM c'est inclure aussi des éléments tels que des périphériques améliorés par l'informatique

distinguer :

- **Mediated Reality**[Man94] : consiste à modifier notre vision de la réalité à partir de dispositifs physiques ou logiciels, la RA devenant un type de filtre d'additif dans ce concept (définition propre à Mann qui peut être inversée, la *mediated reality* devenant une catégorie de la RA) ;
- **Diminished Reality**[MF01] : consiste à filtrer la réalité, la modifier et supprimer ou remplacer (visuellement) des éléments réels par des éléments virtuels ;
- **Artificial Reality**[KGH85] : introduit par Krueger, il le définit comme un monde virtuel, dans lequel un utilisateur peut interagir si naturellement qu'il lui procure une sensation d'immersion : la différence avec la RV s'inscrivant - d'après l'auteur - dans l'usage des techniques vidéos remplaçant l'encombrement de dispositifs physiques (capteurs et casques) par une interface gestuelle et corporelle ;
- **Amplifying reality**[FRB99] : définit le fait d'augmenter les propriétés publiques (visible par tous) des objets réels, par des ressources informatiques inscrites dans l'objet (complémentaire de la réalité augmentée). L'auteur décrit par exemple un prototype de badge numérique qui présente des informations publiques ;
- **Ubiquitous Computing** (informatique ubiquitaire) [Wei91] : présentée par Weiser, c'est une vision dans laquelle l'informatique est transparente, inscrite dans notre environnement accessible partout et en tout lieu à l'aide de multiples périphériques chacun avec un positionnement et une conception adaptée à une ou plusieurs tâches. Notion assimilable : ordinateur disparaissant (*disappearing computer*) ;
- **Pervasive Computing** (informatique évanescente) [Sat01] : assimilable au terme précédent, on peut voir la nuance dans une accessibilité à tout élément numérique sous n'importe quelle forme, pouvant être déplacé avec l'utilisateur : informatique nomade (*nomad/mobile computing*) ou portée par l'utilisateur : informatique vestimentaire (*wearable computing*), convergeant aussi vers sa disparition pour l'utilisateur. Notions assimilables : informatique diffuse, disséminée ;
- **Tangible User Interface (TUI)**[IU97] : utilisation d'élément réel pour l'interaction avec des éléments virtuels/numériques. Notions assimilables : interface saisissable (*graspable user interface*[FIB95]) ou *natural user interface*[RS96].

Pour compléter cette terminologie, on introduit dans la suite un bref historique permettant d'illustrer plus précisément la RA, son intérêt et ses applications.

1.3. Bref historique

La réalité augmentée a vu le jour avec les travaux de Sutherland [Sut65], [Sut68] (figure 2.4) qui a réalisé le premier système dit de réalité augmentée, basé sur un casque suivi par un capteur de mouvement. Avec ce dispositif, l'utilisateur pouvait alors visualiser et naviguer autour d'éléments virtuels positionnés dans l'espace réel. Durant les années 80, le concept de réalité augmentée a été surtout utilisé dans un cadre militaire pour l'affichage d'informations virtuelles sur les pare-brise des cockpits d'avion (*Head-Up Display, HUD*).

Le véritable essor de ce domaine a débuté durant les années 90 par les travaux de Bajura [BFO92], puis State [SLH+96], pour donner au médecin la possibilité de visualiser directement des données d'imagerie à ultrasons sur le corps du patient, augmentant alors sa perception (interaction et visualisation dans un même espace). Feiner [FMS93] propose quant à lui un système interactif pour l'apprentissage et la maintenance d'une imprimante basée sur la superposition d'informations virtuelles (de démontage) sur un casque et grâce à une imprimante équipée de capteurs. Par la suite, la réalité augmentée s'est développée en introduisant un grand nombre d'applications cibles : médecine ([GLPI+96]) ingénierie/production ([CM92], [ABC+94]), architecture (visualisation augmentée [WFM+96], design [AKB+95]), robotique ([MZDD93]) et loisirs [OSYT98].

(a) premier casque de RA (b) vue d'un médécin lors d'une échographie (c) vue d'un utilisateur avec un affichage virtuel du casier à papier d'une imprimante

(d) projection de données médicales sur un patient (e) prototypage virtuel de voiture

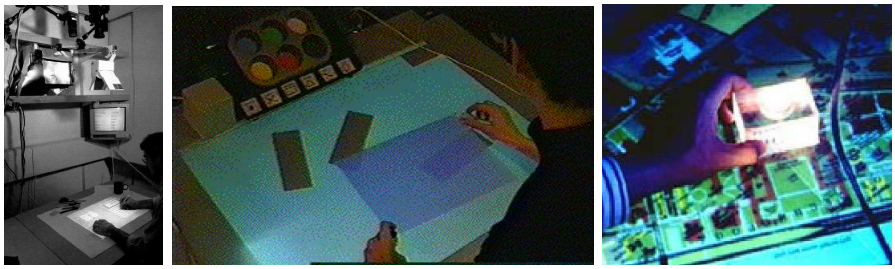
Fig. 2.4: Réalité augmentée de Sutherland à aujourd'hui : point de vue image de synthèse. (a) [Sut65] (b) [BFO92], (c) [FMS93], (d) [GLPI+96] (e) [KDB+02].

En parallèle, la réalité augmentée a aussi trouvé sa place dans le domaine de l'interaction homme-machine (IHM) (figure 2.5). Wellner a proposé la possibilité d'interagir naturellement avec des documents numériques, par l'intermédiaire d'une projection de ces éléments dans notre monde réel (*Digital Desk* [Wel91]). Fitzmaurice [FIB95] puis Ishii [IU97] ont proposé, dans ce contexte, le principe d'interagir à l'aide d'éléments réels pour modifier des éléments numériques, remplaçant les interfaces classiques de bureau (*WIMP*[4]) par des interfaces beaucoup plus souples et adaptées à l'utilisateur (interaction à deux mains, multiplexage spatial, symboliser les éléments virtuels par des manipulateurs tangibles). Rekimoto [RN95] propose l'idée d'interagir sur le réel à travers un dispositif physique spécifique, le *NaviCam*. Par la suite, de nombreux chercheurs d'IHM se sont alors intéressés à cette relation virtuel-réel avec diverses approches : on peut citer par exemple Underkoffler et al.[UI98], Back et al.[BCG+01] ou Renevier et al.[RP01].

Une convergence entre les différentes points de vue conduit maintenant la RA vers une approche multidisciplinaire regroupant des domaines tels que l'informatique graphique, l'interaction homme-machine, la vision ou les travaux autour du travail collaboratif. Dans ce contexte, la forte réduction des coûts des dispositifs matériels, l'évolution des cartes graphiques et la disponibilité d'outils simples et évolués ([KB99]) font maintenant de la réalité augmentée un domaine fortement accessible.

Durant les dernières années, les applications se sont étendues vers un plus grand nombre de domaines : la visualisation scientifique (la chimie [FV02], les mathématiques [FLS97]), l'éducation (mathématique [HK03], l'astronomie [SH02]), la géographie [HBP+01] ou la culture [HPF99]. Mais nous notons aussi de nouvelles applications pour la médecine [BBB03], l'architecture (extérieure [TPG99], design [KBP+00]), la planification [JG02], la maintenance [Dal02], la production [Fri02], l'automobile [KDB+02]) l'apprentissage [JS99] ou encore l'expression artistique [CWY+02] et le jeu (en salle [Mur01] ou en exté-

[4]WIMP = Windows, Icons, Menus and Pointers = désigne l'interface standard de bureau présente sur tous les ordinateurs d'aujourd'hui, mis au point par Douglas Englebart à la fin des années 60 et rendue célèbre par le Macintosh en 1984[WIM]



(a) bureau digitale (b) édition tangible avec deux briques réelles (c) interaction tangible sur carte virtuelle

(d) annotation virtuelle d'illustration (e) conception de paysage en RA avec projection et modèle déformable

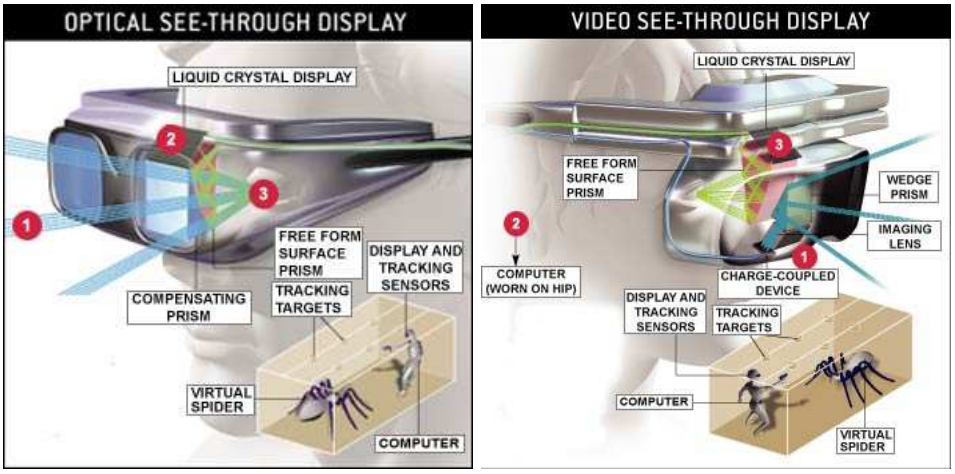
Fig. 2.5: Réalité augmentée de Wellner à aujourd'hui : point de vue interaction homme-machine. (a) [Wel91], (b) [FIB95], (c)[RN95], (d) [IU97], (e) [PRI02].

rieur [TCD+00]).

1.4. Mise en œuvre

Nous présentons ici les solutions matérielles pour la mise en œuvre des concepts de RA. Nous introduirons en premier lieu les solutions matérielles actuellement mise en œuvre, puis les nouvelles technologies émergentes.

1.4.1. Approche classique



(a) (b)

Fig. 2.6: Principe de fonctionnement des casques (a) semi-transparents optiques (b) semi-transparents vidéos (images réalisés par Bryan Christie. *Scientific American*. Tous droits réservés).

Pour la visualisation du virtuel, on utilise des dispositifs d'affichages dédiés qui permettent

alors de mixer réel et virtuel (figure 2.7) . Nous pouvons distinguer différentes classes de système :

- *les affichages de types casques (HMD)* (figure 2.6) : le système est couplé à la tête de l'utilisateur et ils se distinguent en deux catégories. Les casques dit semi-transparent optique (*optical see-through HMD*), constitue d'un écran LCD couplé à un miroir semi-transparent dont le mixage réel et virtuel est fait par l'œil de l'utilisateur [Sut65]. Puis, nous avons les casques dit semi-transparent vidéo (*video see-through HMD*) dont le mixage est fait entre un rendu graphique et l'image provenant d'une camera, ce mélange étant alors présenté à l'utilisateur [EKEK93] ;
- *les affichages de type écran* : la visualisation est faite à travers un écran (optionnellement suivi), à l'aide d'une caméra couplée à cet élément qui peut être : écran de bureau [ABC+94], d'ordinateur de portables [SMW+02], un écran portable à la main [RN95][MKBP02] ou une solution mixte [BKP01a] ;
- *les affichages par projection* : la visualisation est faite sur une surface du monde réel, la projection pouvant être sur une table [RFK+97], un mur [RS99] ou sur une surface quelconque [RWC99],[PRI02], [Pin01].

(a) (b) (c) (d)

Fig. 2.7: Configurations et différents dispositifs de RA : (a) classique [SFH00], (b) mobile [Jul00], (c) de bureau [MP02], (d) publique [AMS+01a].

Pour l'interaction, on utilise la variété des capteurs provenant du monde de la Réalité Virtuelle (magnétique, ultrasons, optique, inertiel), mais aussi des dispositifs plus légers (RFID, capacitif, pression). Divers travaux ont introduits des nouveaux capteurs supportant des zones d'échanges plus grandes [FHP98], [WBV+01]. Une autre solution peut être un suivi par vision qui offre alors une approche robuste et précise pour la réalité augmentée, basée sur des algorithmes dédiés au temps réel [Kli97], [KB99]. Comme nous le verrons au chapitre 5, ce type d'approche peut être couplé avec une technologie plus rapide, une complémentarité qui offre alors un système de haute qualité ([SLH+96]).

La solution idéale reste aujourd'hui largement à définir. Rolland a étudié la différence entre casque semi-transparent optique et vidéo [RHF94] et montre qu'aucune technologie n'est actuellement meilleure, sur des critères technologiques et ergonomiquess, chacun ayant leurs propres avantages. L'optique offre une véritable vue du réel (résolution, vue périphérique, alignement œil-image, instantanée), meilleure acceptation de l'utilisateur. L'approche vidéo offre un alignement de meilleure qualité (mais un conflit sensoriel entre vision et proprio perception), et une meilleure gestion des occultations, contrôle total de la vision de l'utilisateur. Tandis que Azuma dans [Azu93] définit trois besoins pour les systèmes de suivi pour la RA : haute précision, peu de latence et large plage de suivi.

1.4.2. Technologies émergentes

Nous présentons aussi ici un certains nombres de nouveaux dispositifs reposant sur des progrés techniques ou des adaptations dédiées à la RA (figure 2.8).

Hua introduit un prototype de casque à affichage projectif dans un cadre de RA [HGB+02], [HGBR02]. L'approche consiste à utiliser des surfaces rétro-réfléchissantes, offrant une

solution à fort angle de vue, gérant automatiquement les occultations, et peu de distorsion optique. Kiyokawa propose quant à lui un nouveau type de casque optique semi-transparent gérant aussi les occultations base sur l'adjonction d'un panneau de LCD positionné dans le prolongement de l'écran optique associé à un ensemble de caméras permettant de reconstruire une carte de profondeur [KK01b], [KK01a], [KBCW03].

La taille et l'inconfort du casque constituent un point très limitant de l'usage de ce système. Dans ce cadre, différentes solutions alternatives ont vu le jour. D'un point de vue plus ergonomique, Micro-optical a introduit les premiers prototypes de lunettes RA [SZC+01], pouvant, suivant les modèles, se fixer sur de véritables lunettes de vue (aux performances encore fortement limitées). Une nouvelle approche est l'utilisation de la projection rétinienne permettant à l'aide d'une technologie laser modulé, de projeter les informations directement au fond de l'œil [VPNF98].

En complément, de nouvelles solutions d'écrans portables ont été proposées : on peut citer les assistant digital personnel [PW03], [Wag03] et les tablets PC. Dans un cadre collaboratif, le système Virtual Showcase [BFSE01] reposant sur l'utilisation de la surface de projection d'un Workbench pour restituer un espace de visualisation offre une très bonne solution pour des taches peu interactives telles que des présentations multimédia.

(a) (b) (c) (d)

Fig. 2.8: Nouveaux types d'affichages : (a) *HMPD* [HGB+02], (b) *ELMO* [KBCW03], (c) *EyeWear* [SZC+01], et (d) *Virtual Showcase* [BFSE01]

Pour le suivi 3D, [RPF01] [vLM03] proposent quant à eux des solutions par vision basées sur de caméras stéréo-scopiques offrant une solution alternative. Nous noterons pour finir une forte émergence d'architectures matérielles mobiles : *MARS* [HFT+99], *Tinmith* [PGT99], *Archeoguide* [DK02], *OCAR* [GR03].

Les premières notions introduites ici vont nous servir de base pour définir la relation entre RA et environnement collaboratif, dont nous allons alors présenter les concepts de base.

2. Collaboratif : introduction et portée

Nous introduirons tous d'abord une description du collaboratif (section 2.1) puis une étude plus précise de sa relation avec l'informatique (section 2.2).

2.1. Définition et contexte

Nous présenterons en premier lieu la notion de collaboratif puis sa spécialisation par l'informatique et son application à des tâches 3D.

Qu'est ce que le collaboratif ?

Le terme «collaboratif», employé abusivement depuis le début de ce document, n'existe pas dans la langue française : il est généralement utilisé comme adjectif de collaboration (anglicisme de *collaborative*). En effet, ce terme se trouve aujourd'hui de plus en plus

utilisé sous les formes «travail collaboratif» ou «espace collaboratif»[5]. Cette notion de coopération, peut être ramenée au terme «collaboration» qui désigne d'après le Petit Robert (2002) :

> **collaboration** n. f. **1.** un travail en commun ; entre plusieurs personnes qui génère la création d'une œuvre commune.

Mais on peut noter, comme le souligne Cord [Cor03] que «Contrairement à une collaboration, une coopération n'engendre pas nécessairement la création d'une œuvre commune. Nous parlerons de travail collaboratif lorsque la cible commune du travail d'une équipe consiste, outre le travail en groupe, en la réalisation d'un produit final. Par travail collaboratif, nous désignons donc, d'une part, la coopération entre les membres d'une équipe et, d'autre part, la réalisation d'un produit final».

Nous notons aussi la différence entre collaboratif et multi-utilisateurs : multi-utilisateurs désigne une relations multiple hommes vers machine, alors que collaboratif désigne une relation homme-homme par l'intermédiaire de la machine.

Dans notre étude, on s'intéressera au sens large donné à collaboratif allant de collaborer à coopérer, en se concentrant principalement sur l'aspect *travail* (on abusera alors du terme collaboratif (comme adjectif) et travail collaboratif dans la suite du document). Nous considérerons une collaboration sous une forme incluse dans notre propre définition :

> session dans un intervalle de temps contraint[6], faisant intervenir plusieurs individus, utilisant diverses voix de communication pour s'échanger et coordonner des informations (avec ou sans but commun).

Comme le définit Dix [Dix94] on retiendra deux éléments fondamentaux pour communiquer basés sur les capacités humaines (figure 2.9) : communication directe entre participants ou indirecte par l'utilisation d'objets partagés. Nous présenterons brièvement deux domaines particuliers d'étude que l'on a distingués : le travail collaboratif déjà introduit et les activités de loisirs.

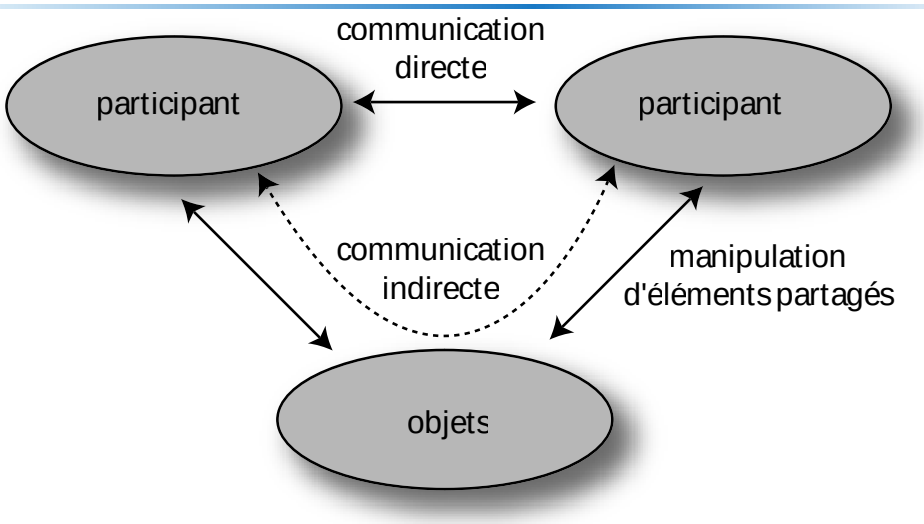


Fig. 2.9: Représentation schématique d'une collaboration (par Dix [Dix94]).

[5]Nous pouvons noter 141600 citations sous *Google* du terme français collaboratif, 51600 citations de «travail collaboratif» (1966 sous CiteSeer) et 4600 citations d'«espace collaboratif» au moment de la rédaction de ce document.
[6]L'échelle pouvant être horaire, journalière, mensuelle ou annuelle mais facilement représentable à un niveau humain.

Travail collaboratif

Dans le cadre professionnel, comme le présente Beaudouin-Lafond dans [BLEE+99], les activités de groupe occupent généralement une grande part du temps : réunion, appels téléphoniques, envoi de messages, rencontre informelle ou discussion, sont des éléments omniprésents dans notre gestion du temps professionnel. Ces méthodes de communication sont des éléments essentiels pour la coordination des activités de travailleurs isolés, permettant de coordonner l'activité.

Comme le défini Schlichter [SKB97], la coopération des personnes engagées dans un but commun nécessite la *coordination des tâches* relatives à l'activité et la *gestion des ressources* utilisées. La communication est utilisée pour mener à bien cette coopération et sert principalement deux buts : l'échange d'information au niveau de données et la coordination des activités au niveau relationnel [SKB97].

Nous pouvons alors remarquer que l'outil informatique est devenu un élément essentiel de nos activités autonomes, mais n'est pas encore aujourd'hui un outil utilisé de façon similaire dans un cadre collaboratif. L'approche simpliste consistant à essayer de reproduire le même environnement s'est révélé beaucoup trop difficile, l'aspect multi-utilisateurs introduisant un grand nombre de difficultés.

Activité ludique collaborative

Dans le domaine culturel et social, la collaboration se retrouve fortement dans les activités sportives ou de jeu. Le fonctionnement est en général basé sur une série de règles pour la réalisation d'un but, constituant le coeur de l'activité. Le jeu vidéo a pris un essor sans précédent ces dernières années se tournant de plus en plus vers des scénarii multi-utilisateurs, brisant les barrières géographiques généralement restreintes pour ce type d'activité. Toutefois l'approche retenu par les professionnels du jeu vidéo est généralement en retard sur les travaux de recherche généraux ou utilise uniquement des solutions dédiées au contexte du jeu[7] (voir une analyse dans Dyck [DPBG03]). De plus, ils ne font pas souvent une analyse profonde des besoins de coopération pour ce type d'activité (communication souvent par mode texte dans les *First Person Shooter, FPS*[8] [Man02], [TM03]).

On retrouve aussi dans le ludique différents type d'activités collaboratives, pouvant être classés suivant le degré d'implication (en entrée et sortie) de l'utilisateur envers l'activité et les autres participants :

- **observation** : se retrouvent dans les activités telles que le cinéma, le théâtre, le sport, où un certain nombre de personnes plus au moins en communication suivent l'activité d'autres personnes, de façon visuelle (spectateur), auditive (auditeur) ou les deux (spectateurs). On a alors une relation entre un groupe actif et un groupe passif (par exemple, cette relation est plutôt de type multi-utilisateurs dans le cadre des spectateurs de cinéma et parfois collaborative dans le cadre du théâtre) ;
- **compétition** : les utilisateurs (ou groupes d'utilisateurs), généralement à un même niveau de compétences, confrontent simultanément leur capacités comme dans les activités de sports ou de jeux (les utilisateurs deviennent participants) ;
- **collaboration** : les utilisateurs ont ici le même rôle et doivent réaliser une tâche de façon commune (telle que des acteurs dans le cadre d'activité de spectacle) ;
- **un meneur** : se retrouve fortement dans le sport mais aussi dans une certaine catégorie de jeu, où une personne a un rôle particulier, guidant les autres utilisateurs à collaborer ou à se confronter (tel que le jeu de rôle).

Nous noterons que l'activité de jeu est de plus en plus étudiée en recherche, principalement sur une population enfantine, pour la compréhension des relations collaboratives (voir les travaux de Sanneblad [SH03] ou Garth [GS00]).

[7] qui offre des aspects intéressants tels qu'une forte configurabilité des interfaces, l'apprentissage par observation du comportement du joueur ou l'aspect communautaire.

[8] type de jeu vidéo dont l'activité principale est l'élimination de ses conccurents.

Nous nous intéressons plus précisément dans la suite à l'utilisation de l'informatique dans le cadre collaboratif, en se focalisant sur l'aspect travail collaboratif.

2.2. Travail collaboratif et informatique : définitions et intérêts

Le *TCAO (Travail Coopératif Assisté par Ordinateur)* ou le terme plus couramment utilisé en anglais *CSCW (Computer Supported Cooperative Work),* décrit le domaine d'étude de l'utilisation de l'outil informatique dans un cadre de travail collaboratif. Les premiers travaux datent de 1984 où le terme semble avoir été introduit par Iren Grief et Paul Cashman. Ce domaine généralement fortement multidisciplinaire (induisant alors une forte complexité d'étude, de mise en œuvre et de réalisation) est défini par Wilson [Wil91] comme «*a generic term which combines the understanding of the way people work in groups with the enabling technologies of computer networking and associated hardware, software and techniques*»[9]. Comme le soulignent différents auteurs, on peut voir le TCAO comme l'étude des relations homme-homme à travers l'outil par opposition à l'IHM qui s'intéresse à une relation homme-machine. Cependant elle est aussi défini par d'autres auteurs comme une sous-branche de l'IHM.

On définit alors généralement le *collecticiel* (*groupware* en anglais), comme une application, un outil, un système ou un environnement informatique supportant une activité collaborative sous diverses formes (terme introduit par Peter and Trudy Johnson-Lenz en 1978 et par Levy dans [Lev90] pour le terme francais). Dans la bibliographie, Bob Johansen [Joh91] définit le groupware comme «*a generic term for specialized computer aids that are designed for the user of collaborative work groups... Groupware can involve software, hardware, services and/or group process support.*»[10].

Cameron [CDOS95] définit le collecticiel comme «*technology that communicates and organize unpredictable information allowing dynamic groups to interact across time and space*»[11]. Comme il semble le signifier, les possibilités induites par ce type de système consiste à surpasser les limitations des outils usuels, favorisant fortement la productivité dans le cadre de travail : échange d'information plus rapide, actions à distance et simultanées sur du contenu partagé, dynamicité, persistance, contrôle aisé du contenu, etc.

Ellis [EGR91] a classifié les différents types de collecticiels en fonction de critères spatiaux et temporels (figure 2.10). Le paramètre temps distingue l'action de collaboration dans un même espace de temps (**synchrone**) ou non (**asynchrone**). La variation spatiale/géographique définit une collaboration en un même lieu (**colocalisée**) ou à des places différentes (**distante**). On peut alors facilement décrire un type de collecticiel dans une certaine catégorie, tout en considérant l'existence d'outils «entre deux mondes» (comme Rodden l'introduit dans [Rod91]).

D'autres classements proposent une vision par types de tâches (voir [Lau02]) ou basée sur la taxonomie de Dix [Dix94] du travail collaboratif.

On utilisera la taxonomie de Ellis pour situer nos travaux et on s'appuiera sur la définition du collecticiel de Johansen pour définir les mises en œuvre de solutions collaboratives.

2.3. Quelques exemples de collecticiels

Nous présentons ici quelques exemples de collecticiels en se basant sur la taxonomie de Ellis. Le lecteur pourra se reporter au livre de Beaudouin-Lafon [BLEE+99] pour trouver de plus amples informations sur la TCAO.

[9] «Un terme générique qui combine la compréhension de la manière dont les personnes travaillent en groupe avec les nouvelles technologies des réseaux d'ordinateurs et les matériels, logiciels et techniques associées.»

[10] un terme générique pour une assistance informatique qui est destiné à un participant d'un groupe de travail collaboratif... Le collectif peut recouvrir du logiciel, du matériel, des services et/ou la gestion de processus de groupe.

[11] «une technologie qui communique et organise l'information imprévisible, autorisant des groupes variables à interagir à travers le temps et l'espace»

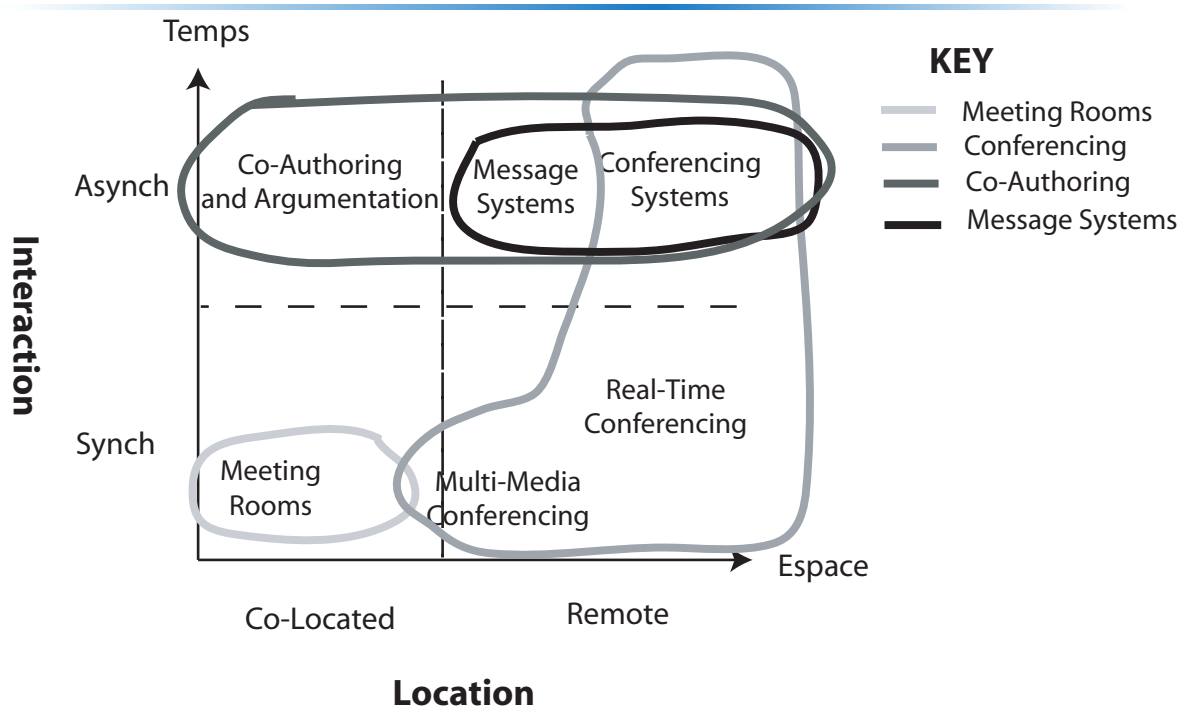
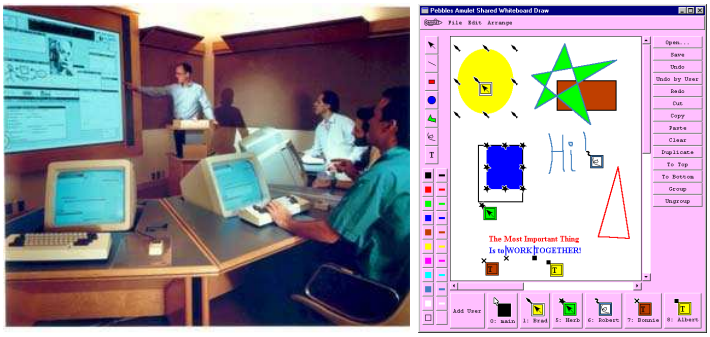


Fig. 2.10: Taxonomie de Ellis modifiée par Rodden [Rod91]. Le lecteur distinguera les différentes types d'applications regroupés en famille.

Collecticiels colocalisés synchrones

Outil de support de réunion ou de présentation, *CoLab* [SBF+87] (figure 2.11a) est un des premiers systèmes développé en 1986 à Xerox. Les chercheurs souhaitaient un outil permettant de sauvegarder le contenu écrit sur le tableau et sur les cahiers de notes durant une réunion. Ils conçoivent alors un système électronique répondant à cette demande sous la forme d'un tableau électronique partagé et l'accès à des ordinateurs personnels pour chaque utilisateur (restreint à un petit groupe).



(a) un exemple de réunion avec le système numérique

(b) l'interface du collecticiel de dessin

Fig. 2.11: Exemples de collecticiels colocalisés synchrones : (a) *CoLab* [SBF+87] et (b) *PeeblesDraw* [MSG98b].

En 1997 le projet *Peebles* [MSG98b] (figure 2.11b) propose un système collaboratif pour dessiner sur un espace partagé, l'utilisateur manipulant un pointeur à l'aide d'un assistant digital personnel (*PDA*).

On trouve aussi d'autres approches (table digitale, multiples surfaces) qu'on présentera plus tard dans ce document (chapitre 3).

Collecticiels colocalisés asynchrones

Ils sont les descendants des tableaux d'affichages généralement placés dans des espaces publics (au travail, dans des centres d'activités, etc.). Nous pouvons citer les travaux de Greenberg [GR01] avec le système *Notification Collage* qui s'intéressent aux avantages de la restitution numérique sur ce support mural, permettant alors de placer vidéo et images et de donner accès à distance à son contenu.

Collecticiels distants synchrones

Pour cette catégorie, les utilisateurs ne sont plus directement face à face et les canaux de communication usuels sont supprimés. Il est alors nécessaire de restituer la présence de l'utilisateur à travers l'interface du collecticiel. Cet élément est généralement nommé *téléprésence* [Bux92]. Ce type d'interface associe en général un espace de travail partagé et un espace interpersonnel de téléprésence.

Sujet de recherche depuis de nombreuses années, on trouve un grand nombre d'outils dans cette catégorie : téléconférence (*NetMeeting* de *Microsoft* [Neta]), dessin (*Groupsketch* [GB91]), édition coopérative (*Dolphin* [SGHH94]), etc.

Pour le travail coopératif, on trouve le prototype *Clearboard* de Ishii[IKJG92] (figure 2.12) qui offre un nouveau type d'interface. Il propose la réduction de la distance de ces deux éléments grâce à un écran dans lequel on voit l'image de son interlocuteur superposée au contenu de la tâche.

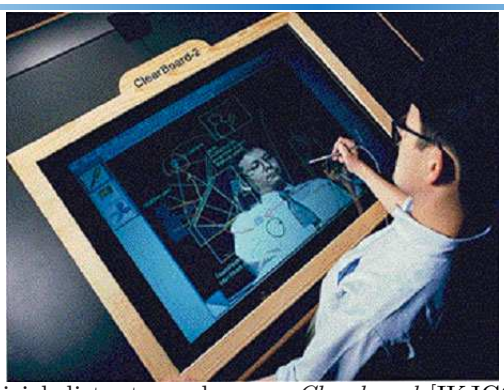


Fig. 2.12: Exemple de collecticiel distant synchrone : *Clearboard* [IKJG92]. Deux utilisateurs peuvent esquisser un modèle virtuel.

Les *mediaspaces* introduits par Stults [Stu86] constituent une autre branche de cette même classe et cherchent à recréer un espace de communication usuel permettant une communication permanente (favorisant les échanges informelles). Un problème inhérent reste le support de l'aspect privé, ces technologies reposant en général fortement sur des systèmes vidéos. Le Mediaspace *CoMedi* [CBC+99] propose une gestion de l'espace privé en se basant principalement sur des techniques de vision.

Collecticiels distants asynchrones

Ils sont les descendants de la lettre et du fax. Cette classe couvre les collecticiels tels que la messagerie, les *newsgroups*, les *mailings list* et les dernières approches comme les messages SMS (*Short Message System*).

Conclusion

Ce tour d'horizon nous a permis d'identifier les catégories de collecticiels existants : on note que cette taxonomie est généralement utilisée en TCAO et IHM et se base fortement sur des applications 2D. Nous nous appuyons sur ce classement pour introduire les collecticiels liés à notre domaine d'étude, et nous nous intéresserons plus précisément par la suite au contenu tridimensionnel et à sa relation avec le TCAO.

3. Collaboratif et tâches 3D

Nous présentons la relation entre le domaine du collecticiel, des images de synthèse et plus particulièrement de la réalité virtuelle.

3.1. Travail collaboratif et informatique : contenu 3D et environnement virtuel

L'utilisation d'éléments virtuels tridimensionnels constitue un élément clef pour des domaines tels que l'architecture, la recherche scientifique, l'archéologie (figure 2.13). En effet, la possibilité de visualiser et d'interagir avec une représentation tridimensionnelle de molécules 3D ou de maquettes virtuelles de bâtiments architecturaux ont un fort intérêt dans le cadre de leur spécification et conception (voir en particulier des articles comme [CLPR98], [SATdV00]).

(a) représentation 3D d'une molécule d'actin

(b) simulation 3D de l'apparence finale d'une maison

(c) rendu 3D d'un modèle de voiture par une carte graphique récente

Fig. 2.13: Exemples d'applications manipulant du contenu tridimensionnel : (a) chimie [Gro], (b) architecture [Joh] et (c) prototypage industriel [Dem].

Ces dernières années, la réduction des coûts et les progrès technologiques offrent un support efficace pour un large développement de collecticiels dédiés à des applications 3D. Le travail collaboratif autour d'une tâche 3D trouve alors tout son intérêt dans de la visualisation scientifique partagée, dans la conception d'éléments manufacturés ou dans l'apprentissage d'un contenu 3D (e.g. maintenance d'une voiture). Revenons brièvement sur cette évolution pour mieux comprendre comment les collecticiels de RV s'inscrivent dans les collecticiels standards et quelles sont les propriétés de la RV par rapport à celle de la RA (présentées précédemment).

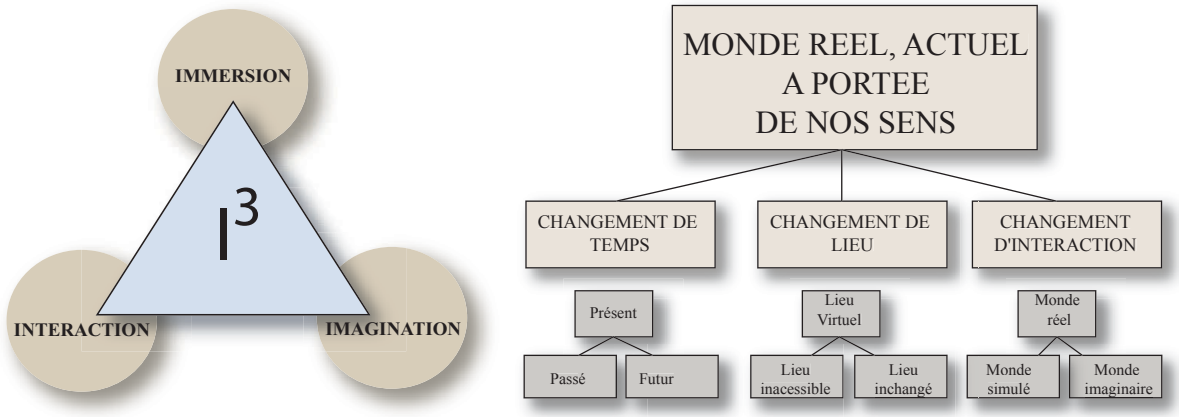
Les environnements virtuels ont pris leurs essor simultanément dans le domaine du jeu et des applications militaires au début des années 80 [SKH02]. Dans le domaine du jeu, on a cherché à reproduire un monde virtuel (environnement virtuel) dans lequel plusieurs utilisateurs géographiquement isolés peuvent interagir (entre eux, avec des personnages ou agents autonomes), principalement à l'aide d'une interface textuelle (nommé *MUD*, *Multi User Dungeons*). Pour l'armée, l'intérêt s'est trouvé dans le cadre de simulation supportant l'entraînement de plusieurs soldats mis en situation, les environnements étant distribués sur réseau (*NVE*, *Networked Virtual Environment*[12].).

La réalité virtuelle (RV) a pris naissance en cherchant à fournir une reproduction du monde réel et son interaction avec l'utilisateur. La Réalité Virtuelle (terme introduit par Jaron Lanier au début des années 1980) a d'abord été définie par le fameux triangle I^3 déjà évoqué précédemment (figure 2.14). Devant la multitude de définitions, les erreurs et les difficultés de spécification des limites de ce terme, on retiendra pour notre part deux définitions introduites par Fuchs dans son ouvrage [FM01] :

[12] On trouvera aussi le terme *Distributed Virtual Environment (DVE)* lié à la notion de distribution des ressources ou *Distributed Interactive Simulation (DIS)* plutôt utilisé historiquement par l'armée

– **définition fonctionnelle** : la réalité virtuelle va permettre de s'extraire de la réalité physique pour changer virtuellement de temps, de lieu et/ou de type d'interaction : interaction avec un environnement simulant la réalité ou interaction avec un monde imaginaire ou symbolique ;
– **définition technique** : les techniques de réalité virtuelle sont fondées sur l'interaction en temps réel avec un monde virtuel à l'aide d'interface comportementale permettant l'immersion «pseudo-naturelle» de l'utilisateur dans cet environnement.

Pour une vision plus complète et détaillée de la RV, on se reportera à l'ouvrage de Fuchs.



(a) (b)

Fig. 2.14: (a) Triangle de la RV de Burdea [BC93] et (b) Taxonomie de Fuchs [FM01] des fonctions intrinsèques de la RV.

L'avancée technologique a alors permis une restitution d'une représentation tridimensionnelle dans les systèmes collaboratifs, donnant naissance à la notion d'*Environnement Virtuel Collaboratif* 3D (EVC ou en anglais *Collaborative Virtual Environment, CVE*)[13]. Ils définissent des environnements où les utilisateurs et les éléments sont représentés sous formes géométriques dans un référentiel spatial, où ils peuvent communiquer entre eux et interagir naturellement avec les éléments du monde (le 3D désignant suivant la bibliographie le type de visualisation et/ou de tâche et/ou de contenu).

Le large essor d'internet et le développement d'interfaces simplifiées (telle que les *plugins* VRML dans les navigateurs hypertextes) a conduit au développement d'environnement complexe et à l'identification de véritable communauté virtuelle. Différents termes ont alors vue le jour : monde virtuel (*virtual world*), monde persistant ou environnement virtuel partagé (*SVE : Shared Virtual Environment*).

Nous pouvons définir plus précisément un EVC 3D de TCAO comme un collecticiel dédié à la manipulation de contenu 2D (avec une représentation 3D) et 3D[14]. En pratique, les collecticiels 3D se différencient fortement des collecticiels classiques, dans la mesure où ils ont une forte tendance à reproduire les caractéristiques spatiales du monde réel (d'où la notion forte d'*environnement virtuel*) et de son contenu, au lieu de fournir des représentations alternatives ou simplifiées.

3.2. Quelques exemples d'environnements virtuels collaboratifs

A partir de cette présentation, il nous semble judicieux de présenter quelques EVC (illustrés figure 2.15) pour comprendre le type d'application collaborative 3D existante et pour

[13]Notons bien qu'un EVC pouvant être aussi 2D, l'interaction forte et l'espace partagé sont les points prioritaires de cette notion.

[14]Généralement le terme EVC 3D est employé pour une vision géographiquement distante des utilisateurs, au lieu de définir l'ensemble des collecticiels supportant un espace 3D. La raison est inscrit dans l'historique de son développement.

mieux situer par la suite la différence par rapport aux solutions de RA.

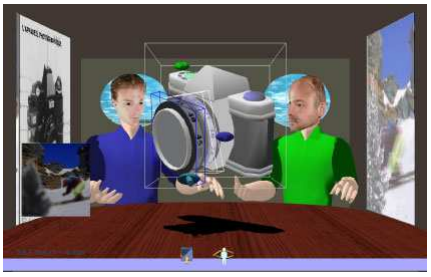
Un des précurseurs est sans doute *DIVE* (*Distributed Interactive Virtual Environment*) [CH93] qui propose un environnement distribué sur internet. Il permet la navigation et l'interaction avec des éléments ou des applications de l'environnement virtuel qui peuvent être générés à partir d'une interface basée sur une programmation par script.

NPSNET [CMBZ00] propose un système modulaire, dynamique et extensible pour des environnements virtuels sur réseau à grande échelle. Les applications sont aussi bien la défense militaire que les applications ludiques.

Spin-3D [DDC+99] introduit une plate-forme pour du TCAO distribué synchrone pour de petits groupes. Il propose une visualisation et une interaction 3D (avec reproduction de la métaphore de réunion sur table) et une architecture pour le développement rapide d'applications coopératives (voir les thèses de Dumas [Dum99] et Louis dit Picard [dP03] pour de plus amples informations).

Au niveau ludique, on peut citer le monde virtuel *Cryopolis* [Cry01], qui définit une cité virtuelle dans laquelle un utilisateur peut mener différentes activités et rencontres. Le principal développement revient sans doute au jeu vidéo, avec la naissance des jeux en ligne massivement multijoueurs (*massivelly multiplayer online (game), MMO*), fortement poussé par les jeux de tirs et les jeux de rôles. Nous pouvons citer dans ce contexte *Everquest 2* (2004) [Eve04], fournissant un support de communication par interface textuelle, avec une orientation jeux de rôles.

(a) vue d'un utilisateur lors d'une simulation militaire



(b) deux utilisateurs représentés par des clones en interaction avec une application d'apprentissage de fonctionnement d'un appareil photo

(c) un utilisateur en action, l'interface textuelle procurant des informations sur l'état du jeu et des messages des autres joueurs

Fig. 2.15: Exemples d'EVC 3D : (a) *NPSNET V*[CMBZ00], (b) *Spin-3D* [DDC+99] et (c) *Everquest 2* [Eve04].

4. Cas restreint : travail collaboratif colocalisé synchrone

Nous remarquons qu'un grand nombre de travaux présenté ci-dessus considère l'utilisateur isolé devant son système : la communication et le partage se réalisent uniquement par l'intermédiaire des interfaces d'entrées et de sorties de la machine. Nous pouvons alors soulever la question concernant la possibilité d'interagir dans un *même lieu physique réel* pour des tâches virtuelles (système dit colocalisé). Nous présenterons dans cette section plus spécifiquement les applications colocalisés de ce type.

4.1. Généralités

Le cas *colocalisé synchrone* nous semble le plus intéressant car il possède les meilleures capacités d'échanges (en terme de qualité) : la *collaboration face-à-face* s'inscrivant dans toute notre culture. Il se définit par une coopération dans un même lieu, au même instant et

entre plusieurs personnes. Nous pouvons donner en exemple une réunion, une présentation, un cours, etc.

Ce type de collaboration permet un *échange direct* d'informations, une perception et une connaissance aisée des sentiments et impressions des autres utilisateurs ainsi qu'une très forte dynamique de mise en place ou de changement d'aspect d'une coopération.

Le retour perceptif de la présence des autres utilisateurs (*awareness*) constitue aujourd'hui un des sujets d'études des collecticiels distants, le face-à-face permettant des moyens intuitifs de communication qu'il faut reproduire dans les solutions distantes [GGR96]. Greenberg a étudié précisément l'apport de l'informatique dans ce type de collaboration (voir [GC89]). Il distingue deux approches : l'utilisation du média traditionnel et le média informatique (outrepassant les limites physiques du premier). Il montre les limites des solutions actuelles (éloignement de l'utilisateur de la tâche, peu de fonctionnalités améliorées, solutions trop complexes).

Nous présenterons dans la prochaine section un cas spécifique d'étude, la réunion, qui semble être pour nous une des applications majeurs du TCAO colocalisé synchrone.

4.2. Type d'activité spécifique : la réunion, intérêt et classification

Le terme *réunion* étant très génerique, on ne considérera dans notre cas que les utilisateurs très actifs, c.-à-d. en forte interaction bidirectionnelle et très impliqués dans l'activité de travail. La concept de réunion se trouve aussi présent dans un cadre collaboratif distant, mais on s'intéresse uniquement ici au cas colocalisé.

Tang [Tan89] a fortement étudié au cours de sa thèse l'activité menée durant ce type de collaboration, principalement au sein d'une réunion sur table. Dans ses expériences, différents utilisateurs partagent un espace de dessin commun comme support pour une tâche de conception. Il montre (dans [Tan91]) que :

- les gestes sont beaucoup plus utilisés pour communiquer de l'information que pour dessiner ;
- le processus de création du dessin contient plus d'information que le résultat lui même ;
- il y a un fort nombres d'activités simultanées et entrelacées dans l'espace de dessin ;
- l'orientation des utilisateurs fixe l'activité ;
- l'activité doit supporter l'accès concurrent ;
- il distingue trois types d'action : énumérer, dessiner et faire des gestes (*list*, *draw*, *gesture*) ;
- il distingue trois types de fonction : sauver l'information, exprimer des idées, coordonner l'interaction ;

Ces éléments nous serviront comme fondement dans la suite de notre étude (en souhaitant les conserver ou les «augmenter»).

Nous présentons une classification de quelques types d'activités collaboratives centrées sur le concept de réunion.

- **Conception collaborative** : différents acteurs de même niveau travaillent sur la réalisation d'un concept. On trouve par exemple dans ce cadre le design d'un même modèle ou l'*eXtreme Programming (XP)*[15].
- **Réunion** (*meeting*) : assemblée entre différentes personnes pour la discussion, coopération, collaboration autour d'une ou plusieurs tâches.
- **Revue** (*review*) : réunion généralement dédiée à l'analyse d'un élément, de façon très précise.
- ***Brainstorming*** : réunion entre différentes personnes basée sur une méthode consistant à favoriser la créativité et l'expression spontanées d'idées (traduisible par remue-

[15] Nouvelle technique de programmation consistant à un travail en binôme devant la conception de module de programme, voir l'ouvrage dédié à ce sujet [Bec99]

méninges d'après le dictionnaire Hachette (1997)).
- ***Briefing*** : ce terme est un anglicisme et désigne un type de réunion définissant les objectifs (donnant «informations» et «consignes») d'une activité prochaine, très utilisés dans le secteur militaire.
- ***Debriefing*** : ce terme est aussi un anglicisme (compte rendu ou réunion de clôture semblent les termes équivalents les plus proches). Il désigne un type de réunion analysant le déroulement d'une activité passée.

Nous nous intéressons à partir de la présentation de ces termes aux spécificités du travail collaboratif colocalisé dans le cadre de tâches 3D.

4.3. Tâches 3D colocalisées : spécificités

Nous présentons quelques spécificités et besoins nécessaires dans un cadre de tâches 3D colocalisées. Ces spécificités tiennent compte des différences par rapport aux collecticiels 2D, 3D distribués et des propriétés des réunions réelles sur des taches 3D. Nous introduisons successivement : des besoins utilisateurs, des besoins coopératifs et des besoins systèmes.

Une assimilation tridimensionnelle de la tâche par l'utilisateur (de façon cognitive) : il faut fournir à l'utilisateur des mécanismes explicites pour l'assimilation d'une connaissance, compréhension et représentation cognitive de la tâche en 3D. Nous pouvons citer des approches par multiples vues, rendu 3D, représentation stéréoscopique, etc.

Uune interaction avec le contenu tridimensionnel : la non homogénéisation des méthodes d'interaction et des interfaces 3D et l'absence de méthodes ou de référentiels pour l'interaction 3D (telle que le WIMP en 2D) constitue un défaut important dans la facilité d'utilisation des éléments de la tâche 3D. Ceci est d'autant plus difficile à gérer dans un cadre coopératif.

Un retour sensoriel rapide des actions, contrainte temps réel : contrairement aux outils à distance, la perception d'un retard même de faible durée entre action et retour sensoriel conduira à une forte baisse des performances du système. Le délai doit être de quelques dizaines de millisecondes et ce pour tous les utilisateurs. La mise en œuvre de technique temps réel et la réalisation d'architecture dédiée est un élément clef.

Une représentation tridimensionnelle partagée : la possibilité d'indications gestuelles, d'assimilation du contexte à partir d'un point de vue globalement identique nécessite de mettre en œuvre des mécanismes permettant de visualiser dans un même (ou proche) espace réel le contenu tridimensionnel (sa superposition avec les outils d'interactions n'est pas un besoin nécessaire).

Une modélisation tridimensionnelle : il faut fournir des mécanismes de création et de gestion d'éléments tridimensionnels. Des architectures logicielles dédiées et des formats explicites doivent permettre cette mise en œuvre.

A partir de la description de la RA, du collaboratif et de son support de tâche 3D (par la RV), on s'intéresse maintenant à étudier quel peut être l'apport de la RA dans un cadre collaboratif colocalisé pour tâche 3D.

5. Implications de la réalité augmentée dans le cadre collaboratif 3D colocalisé synchrone

Nous présenterons ici brièvement des pistes qui nous montrent l'intérêt de la RA dans un cadre collaboratif synchrone pour des tâches 3D. Les comparaisons sont faites par rapport aux concepts existants, solutions de la RV ou de l'IHM.

Vision du monde réel comme support de proprioperception : les troubles (anxiété, perte d'équilibre) et la difficulté de répérage induits par la RV sont largement réduits en

RA, car la vision du monde réel permet une localisation dans un environnement connu, avec des repères naturellement identifiables.

Vision du monde réel comme support de communication : la difficulté de reproduction des autres utilisateurs (en distribué), la difficulté de communication dans un espace contraint (peu d'éclairage dans le cadre de *CAVE* ou de *Workbench*), ou l'absence d'éléments de perception des autres utilisateurs (contact visuel) sont supprimés en RA, la communication reste naturelle ne nécessitant pas de gestion spécifique.

Vision du monde réel comme support de l'environnement de la tâche : la conservation de support et d'espace réel (table, mur, pièce) permet alors d'éviter la définition d'interfaces spécifiques. La compréhension de l'introduction d'élément virtuels dans le monde réel réduit la difficulté d'apprentissage et les limitations des environnements des systèmes existants.

Utilisation du monde réel comme support d'interaction pour la tâche : la possibilité de voir mais aussi de manipuler facilement des objets réelles conduit à une meilleure utilisation d'interface de type TUI, de métaphores multimodales et d'interfaces simples. Leur compréhension, leur utilisation plus directe réduit la difficulté de manipulations que l'on a avec les outils "classiques" de la RV.

Utilisation du monde réel comme contexte de la tâche : l'introduction de la possibilité de visualiser des éléments virtuels en tous lieux, grâce à des solutions dites mobiles, permet à l'utilisateur un nouveau type d'interaction. La visite collaborative d'une ville virtuelle peut être remplacée par le déplacement physique à travers cette ville en augmentant le champ de vue de l'utilisateur avec des informations sur les différents lieux (description historique, informations sur le contenu d'un restaurant, aide à la navigation). Le premier prototype *MARS* [HFT+99] a démontré un résultat de ce type, actuellement fortement étudié dans le projet *ARCHEOGUIDE* [GD01].

Ces différentes pistes montrent l'intérêt de la RA par rapport aux solutions actuelles qui souffrent de nombreuses limitations. Elles constituent pour nous une motivation suffisante pour la réalisation d'une étude plus poussée, d'une analyse par rapport aux éléments existants et d'une démonstration des pistes développées.

Mais il semble que ces nombreux avantages induisent un certain nombre de difficultés de plusieurs ordres :

- **technologie** : un besoin de cohérence entre les deux mondes doit être respecté pour une sensation d'intégration entre ces éléments (voir chapitre 5) ;
- **collaboratif** : le référentiel commun donné et naturel induit alors des contraintes à respecter (e.g même position pour le pointage d'un élément virtuel par différents utilisateurs) ;
- **applicatif** : la présence du réel incite alors à connaître des informations sur ces éléments et donc «percevoir» l'environnement et son contexte (e.g annotations d'éléments réels nécessitant l'identification de ces objets).

A cause de tous ces éléments, nous nous intéressons dans la suite à l'expérimentation, l'évaluation, la mise en œuvre de la RA dans un cadre collaboratif colocalisé. Nous introduisons donc dans la suite de ce chapitre la méthodologie qui a servi comme cadre durant le déroulement de la thèse.

6. Approche choisie : environnement générique

L'étude et l'expérimentation de la réalité augmentée dans un contexte collaboratif colocalisé, nécessitent de définir une stratégie de recherche qui guidera alors le type de contributions apporté par ce travail. Dans cette thèse notre choix est basé sur la *proposition d'un environnement générique* : explicitons plus précisément les motivations de ce

choix.

6.1. Environnement : intérêt et définition

La validation de pistes de recherche dans les domaines qui nous intéressent nécessite généralement la mise en œuvre de différentes expérimentations et prototypes. Mais il nous semble certain qu'une démonstration de ces concepts dans différents cadres applicatifs est indispensable, conduisant ainsi à la réalisation d'un grand nombre de prototypes. En complément, une telle démonstration nécessite la mise en œuvre de différentes techniques adaptées à la conceptualisation d'un prototype de RA collaboratif tel que des techniques d'interactions ou de gestion de caractéristiques induites par ce domaine particulier.

En conséquence, il nous a semblé intéressant de se concentrer sur la proposition d'un *environnement*, répondant à ces concepts. Par **environnement** nous entendons : un ensemble des moyens et concepts disponibles pour offrir une réponse adaptée, complète et facile à mettre en œuvre pour répondre à une tâche donnée.

Dans le cadre de cette thèse, un environnement de RA collaboratif colocalisé nous fournira un milieu dans lequel plusieurs utilisateurs pourront effectuer des tâches usuelles de leur domaine et interagir avec des tâches 3D/2D. L'ensemble des moyens mis en œuvre devra être le plus complet et plus accessible possible permettant de se concentrer sur les problématiques non fonctionnelles.

Nous avons choisi de plus de proposer un environnement dit *générique* donc non dédié à un domaine spécifique d'applications, ni à un cadre de réalisation (i.d. conception, expérimentation, évaluation).

Nous nous posons alors la question de la méthode de réalisation de cet environnement.

6.2. Méthode : approche globale

Nous avons retenu une stratégie dite globale, cherchant à étudier une solution intégrale au lieu de cibler un sujet précis d'étude pour la constitution de cet environnement. L'avantage est la génération d'un squelette d'éléments générique, fournissant une vision englobante des problématiques reliées. L'inconvénient principale par rapport à une stratégie locale est le risque d'un approfondissement non complet ne garantissant pas une totale robustesse de la validation.

Cette stratégie globale repose sur une approche dite centrée utilisateur [IBM], reposant sur le facteur humain comme élément moteur d'évaluation dans un contexte «itératif» (prototypages, evaluations et raffinement). Cette approche nécessite la mise en œuvre d'une définition des besoins et une décomposition en tâches. Elle nous a conduits dans notre cas à une décomposition en différents modules, définissant des classes de problématiques d'études.

Description de l'approche (cf. figure 2.16) : dans un premier temps on réalisera une étude et une analyse de la réalité augmentée et des environnement collaboratifs, des systèmes développés et des travaux liés. Ce premier point va alors nous permettre de montrer la validité de l'intérêt de la RA puis de définir des éléments de conception initiaux pour la proposition d'un environnement (qui constitue des cas d'étude précis détaillés dans les prochains chapitres). A partir de ces premières solutions les éléments sont ensuite unifiés pour former notre prototype d'environnement générique, nommé *MARE* (*Multiuser Augmented Reality Environment*).

En se reposant sur cet environnement, nous obtenons une base minimale pour la définition d'applications et une première validation complète de notre approche. D'un point de vue applicatif, il est bien sûr nécessaire de réaliser une étude des propriétés d'un domaine, des besoins ergonomiques puis une adaptation des éléments identifiés, qui nous serviront

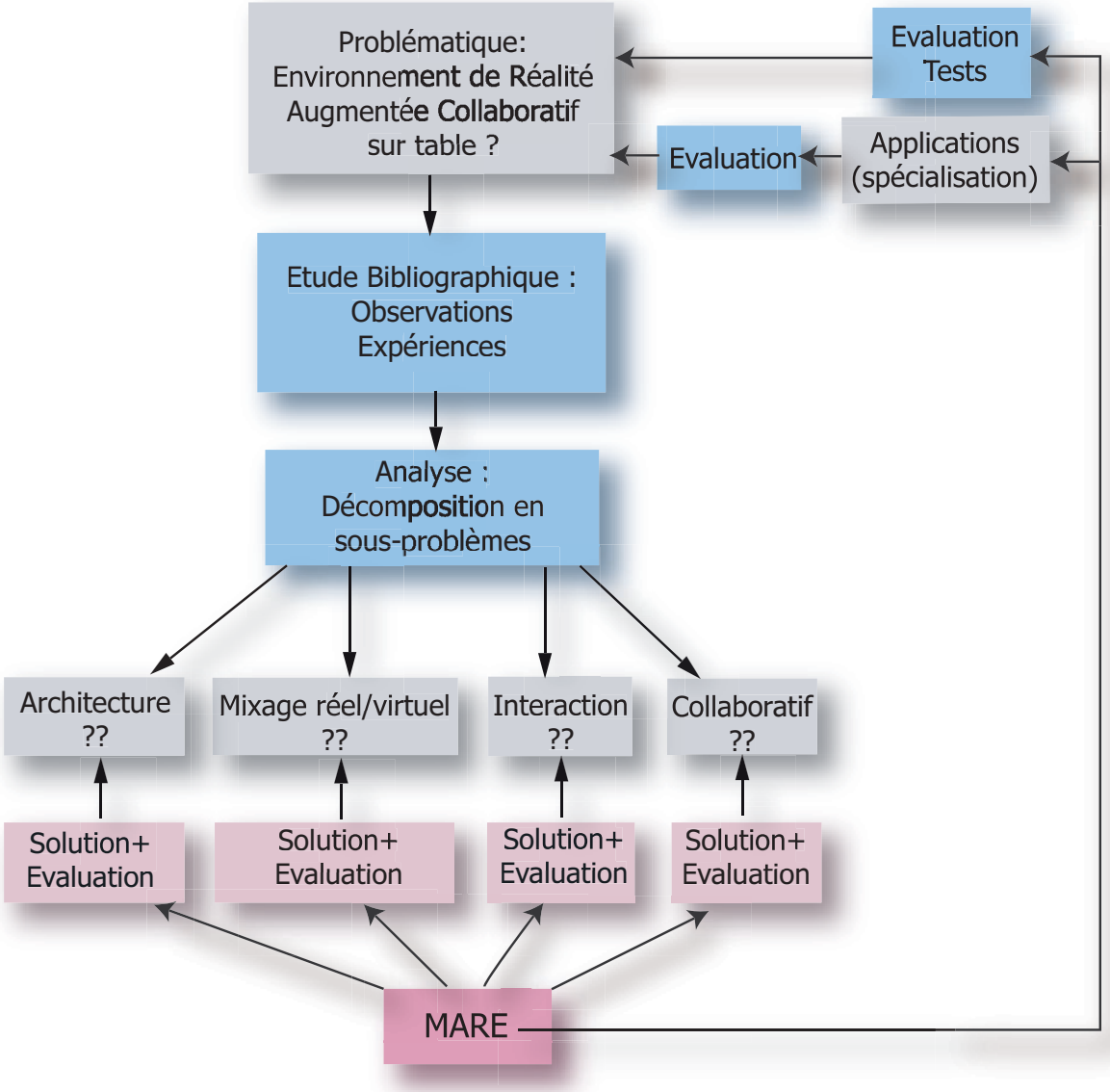


Fig. 2.16: Approche globale : décomposition en sous problèmes (synoptique).

pour la réalisation d'un prototype applicatif (telle que par l'inscription dans un cycle en V [Cou00]). Ce prototype sera évalué dans un cycle itératif (dans l'approche centrée utilisateur) permettant alors de mettre à jour les éléments de conception et de redéfinir à partir de plusieurs domaines la validité de l'environnement générique. En complément une pré-étude sur la globalité du système permet de collecter de nombreuses informations.

Un cycle complet permet alors d'identifier les problèmes et de raffiner certains éléments de conception ce qui nous aidera à délivrer un guide, des méthodes et principes pour la conception d'environnement de RA. La définition complète de méthode de conception nécessite en plus de cet thèse un large travail additionnel : nous nous limiterons à la proposition de quelques pistes de conceptions. Nous nous reposerons aussi dans ce cadre sur les travaux de Gabbard [Gab01] et de Dubois [Dub01] fournissant de premiers éléments de conception d'environnement de RA (non collaboratif, ni colocalisé).

7. Conclusion

Ce chapitre a permis de se familiariser avec les éléments de la réalité augmentée (RA), du

travail collaboratif assisté par ordinateur (TCAO) et de son implication dans un cadre de tâches 3D. L'étude du cas restreint colocalisé montre un intérêt certain de l'utilisation de la réalité augmentée qui peut dépasser certaines limitations de la réalité virtuelle (RV) et compléter les approches proposées par le domaine de l'interaction homme-machine (IHM). Ceci nous a donc conduis à nous intéresser dans cette thèse à l'étude de la RA dans un contexte collaboratif colocalisé dédié à des tâches principalement 3D.

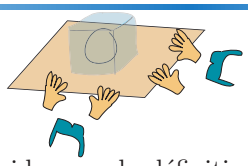


Fig. 2.17: Schéma qui nous servira de guide pour la définition des sous-problématiques. Nous pouvons distinguer les éléments symbolisant la visualisation (casques), l'interaction (main), l'espace d'interaction et de communcication (la table) et la tâche (cube et sphère).

De ce fait, nous avons choisi de proposer un environnement nous permettant un large choix d'expérimentations, d'applications et de démonstrations de techniques adaptées. Nous avons donc défini une méthodologie d'étude qui va nous servir comme composantes pour nous guider dans la suite des travaux. Elle nous servira comme fondement pour l'analyse des résultats et pour en déduire la validité du découpage proposé.

Nous noterons ici que nous utiliserons dans la suite de cette thèse une représentation schématique des problématiques soulevées pour la réalisation de notre environnement. Cet élément représente les composantes principales des propriétés de notre environnement : interaction et visualisation collaboratives sur table d'éléments virtuels 3D (figure 2.17).

Chapitre 3

Environnements collaboratifs colocalisés : état de l'art

Nous avons montré pourquoi la réalité augmentée nous semble être un cadre propice au support d'environnement collaboratif colocalisé pour des tâches 3D. Nous présentons ici un état de l'art sur les environnements collaboratifs colocalisés, supportant l'interaction avec des tâches 3D. La question que l'on se posera au cours de ce chapitre est la suivante : «Quelles sont les solutions actuelles pour collaborer sur des tâches 3D dans un même lieu physique ?» (figure 3.1).

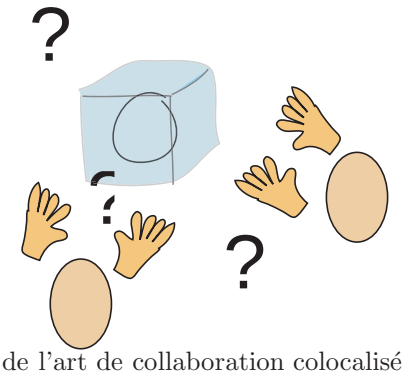


Fig. 3.1: Problématique de l'état de l'art de collaboration colocalisé sur tâches 3D : comment interagir avec le contenu 3D partagé ? dans quel espace ? quelle relation établir avec les autres utilisateurs ?

Nous introduirons tout d'abord le cadre d'analyse utilisé pour réaliser cette étude (section 1). Après la présentation des travaux existants (section 2), une synthèse nous conduira à valider l'apport hypothétique de la RA (section 3) et nous motivera pour une proposition d'un environnement de RA collaboratif colocalisé (section 4).

1. Introduction

Nous identifions tout d'abord la portée de l'étude (le contexte de restriction de l'analyse), puis le cadre d'analyse (les critères discriminants des travaux) et enfin la méthode d'analyse qui nous servira pour la synthèse globale.

Portée de l'etude

Domaines : suite au survol du chapitre précédent, il nous a semblé judicieux de ne pas se restreindre au domaine de la réalité augmentée dans cet état de l'art pour répondre à la problématique définie précédemment. En effet, les travaux relatifs menés en RV et IHM/TCAO offrent de nombreux éléments qui peuvent servir comme base d'inspiration dans la définition d'un environnement collaboratif colocalisé. Notre choix a donc été une étude étendue, qui reste toutefois fortement orientée par notre domaine : la RA 3D (3D signifiant «à visualisation 3D»).

Type : l'étude portera principalement sur des environnements complets (commerciaux ou projets libres) et non des éléments d'un environnement. La distinction est généralement floue dans la littérature ce qui rend le sens de ce terme difficilement perceptible. Nous ne sélectionnons donc pas les contributions d'architecture qui n'ont pas été démontrées

à l'aide d'une structure complète des applications ou un support d'utilisation à court terme (c.-à-d. ni des prototypes matériels, ni de librairies/plate-forme logicielle). Nous ne sélectionnons pas non plus les méthodes d'interactions (éventuellement collaboratives) qui n'ont pas été proposées dans un cadre dédié (et en accord) avec une architecture (c.-à-d. ni méthode d'interaction spécifique, ni technique avec prototype de casque quelconque). Nous regrouperons les travaux ayant une certaine généricité (classe d'environnement) sans toutefois être précisément définis (un environnement particulier avec tous les besoins), dans une même section.

Restrictions : Concernant le lieu, nous nous limiterons à des solutions en environnement intérieur, correspondant généralement plus à des solutions de réunions explicites. Nous ne présenterons donc pas des systèmes tel que *MARS* [HFT+99], *OCAR* [RS04] ou *Augmented Stroll* [RP01]. Nous considérerons des utilisateurs se trouvant dans un même lieu, à distance pour un support communicatif. Nous nous limiterons de plus à des systèmes supportant et démontrant au moins deux utilisateurs : nous ne présenterons donc pas non plus *Digital Desk* [Wel91], *VIP* [AMS+01b], ni *Live Paper* [RR01].

Critères et besoins

Il est nécessaire de se baser sur un certain nombre de critères identifiant les caractéristiques intrinsèques de chacun.

Nous proposons donc un canevas d'analyse s'appuyant sur notre étude et sur différents classements proposés dans Krueger [KC02]. En effet, une large étude bibliographique, l'identification des travaux liés à la réalité augmentée et leurs analyses nous a permis de raffiner et d'identifier des éléments importants pour la conception de systèmes collaboratifs colocalisés.

Pour définir notre canevas d'analyse, nous introduisons trois classes d'éléments de conception des environnements :

- **Ressources** : le type de ressources matérielles et logicielles utilisé par le système. La partie matérielle définit les dispositifs physiques mis en œuvre pour visualiser et interagir avec du contenu, le nombre d'utilisateurs supportés et le type de systèmes proposés (nombres de surfaces, dimensions). La partie logicielle définit le type de plate-forme (librairie, canevas, outils graphiques), la gestion de données (telle que pour le contenu 3D, des primitives bas niveau, un graphe de scène ou une description textuelle) et les outils utilisés pour la réalisation (outils de déboguage, d'évaluations).
- **Interaction** : le type d'interfaces et d'outils d'interactions proposés par le système. Nous présenterons aussi la présence de métaphores et le types de données manipulables.
- **Collaboratif** : les ressources et les limitations des éléments permettant la collaboration : nous distinguons le support de communication naturel (échanges verbaux/gestuels, regard, *awareness*), les ressources pour la tâche définissable par la communication médiatisée (coordination, protection des données, etc.). La présence d'espace de communication mais aussi d'espace personnel sera aussi introduite.

Méthode d'analyse

Basée sur ces critères, la méthode d'analyse retenue consiste en une description fonctionnelle et technique de l'article identifié, puis en une brève analyse à partir des critères définis précédemment. Une synthèse sera présenté en conclusion de tous les travaux (section 3).

2. Étude des systèmes existants

La description est ici classée en fonction du domaine d'étude et de mise en œuvre de la solution. Nous avons retenu quatre classes majeures :

- système basé réalité virtuelle (section 2.1) ;
- système basé interaction homme-machine (section 2.2) ;

– système basé réalité augmentée 2D (section 2.3) ;
– système basé réalité augmentée 3D (section 2.4).

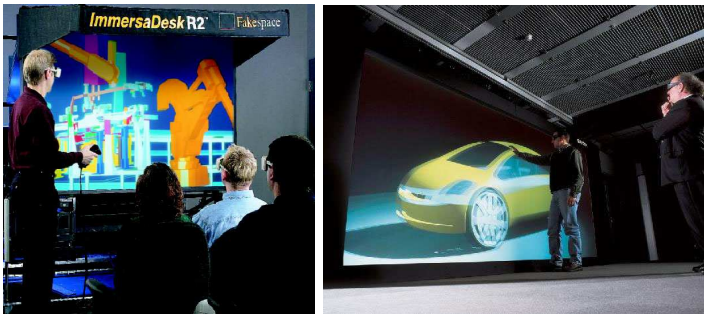
2.1. Systèmes basés réalité virtuelle

Nous présentons ici les travaux liés à la RV que nous classons en fonction du nombre de personnes (décroissante) selon les périphériques de visualisation utilisés. Nous aurons donc d'après notre étude : les environnements à grande audience (sous-section 2.1.1), le projet *Continuum* (sous-section 2.1.2), le *CAVE* (sous-section 2.1.3), puis le *PIT* (sous-section 2.1.4), l'approche sur table 3D (sous-section 2.1.5) et les approches individuelles (sous-section 2.1.6).

(a) salle de type géode offrant une large capacité

(b) session interactive de navigation pour un petit groupe

(c) revue pour un large groupe sur un large écran plat



(d) revue en petit groupe sur grand écran

(e) conception à deux utilisateurs sur large écran

Fig. 3.2: Approches large public : (a) *Evans and Sutherland Digistar 3* [E], (b) *SGI Reality Center* [Cen] (c) *Barco PowerWall* [Pow] (d) *Fakespace ImmersaDesk 2* [Immb] et (e) *Barco Cadwall* [Cad].

2.1.1. Environnements à grande audience : vers des surfaces verticales

Description

Les solutions par dôme reposent sur un écran sphérique de grande taille sur lequel un projecteur omnidirectionnel affiche une information stéréoscopique. Elles reposent sur des architectures matérielles telle que *Evans and Sutherland Digistar 3* [E], *Seos V-DOME* [VD] (figure 3.2a). Chaque utilisateur est équipé de lunettes stéréoscopiques passives : le système supporte un très grand nombre d'utilisateurs.

Son utilisation implique l'absence d'éclairage secondaire autre que le système de diffusion d'images, la collaboration se limite donc à des échanges verbaux entre utilisateurs. Il est généralement destiné à des présentations de documentaires scientifiques (astronomie, biologie, chimie) ou culturelles. Le développement et le support d'applications est largement contraint par les fournisseurs de l'architecture matérielle, les solutions restant propriétaires et complexes à mettre en œuvre.

Dans la même catégorie, nous trouvons les systèmes cylindriques (ou semi-cylindriques) constitues d'un écran de forte courbure (typiquement de 4 m à 6 m de rayon), supportant l'immersion d'un groupe d'utilisateurs placés dans une position centrale (figure 3.2b). Généralement installés dans des salles de grande taille, ils reposent sur l'utilisation d'au moins trois projecteurs qui fournissent la restitution d'une image tridimensionnelle continue (*SGI Reality Center* [Cen]). Une autre solution consiste en l'agrégation de plusieurs écrans actifs (comme *Fakespace Curv* [Cur]). Les utilisateurs équipés de lunettes stéréoscopiques peuvent communiquer avec leurs voisins ou intervenir avec le présentateur durant l'application. Celui-ci contrôle le déroulement de l'application selon un scénario prédéfini et peu réagir à la demande explicite d'un spectateur. Il utilise dans ce cadre des périphériques de bureau (clavier, souris) ou des périphériques de RV (souris 6D, capteur de mouvement, gants, etc.). Ce type de système sert alors à des présentations en petit groupe, à des revues de simulations ou à la visualisation de modèle assurée par une restitution de grande taille et de haute résolution.

Enfin, nous pouvons regrouper les systèmes utilisant des écrans plans, basés sur une projection (rétro ou frontale), servant alors à des présentations pour des réunions en groupe de taille moyenne (figure 3.2c) ou de petite taille (figure 3.2d). L'absence de courbure réduit la présence immersive, favorisant une visualisation dans un cadre de tâche de revue, par rapport à des applications de navigation (cette planarité était jusqu'à très récemment indispensable pour pouvoir utiliser des vidéos projecteurs à très forte luminosité, donc compatible avec la lumière d'une salle de travail mais sur lesquelles les corrections sphériques n'étaient guère possibles). Ce type de solution de typel DLP aussi alors être utilisé plus facilement pour de la conception collaborative (figure 3.2e).

Analyse

La principale restriction de ce type de solution est le fort coût de mise en œuvre qui reste quasi inabordable pour des sociétés de petite taille. La configuration et l'exploitation de ce type de système restent à ce jour complexe (réglages fréquents et délicats). De plus, l'interaction est généralement dirigé par un médiateur unique ou par l'utilisation d'une interface collective (comme le *CAT (Control Interaction Table)* [HGRT03]).

La luminosité de la salle étant faible (afin de préserver la capacité limitée des systèmes de projection), la communication se fait dans des conditions difficiles : en l'absence de métaphores non verbales (regard, expression faciale) et dans l'impossibilité d'un accès facile à des éléments personnels (cahier, notes, etc.).

Nous notons toutefois que ce type de système convient parfaitement à un cadre multi-utilisateurs de grande taille et pour immerger les spectateurs dans une présentation prédéfini. La croissance du nombre d'utilisateurs induits toujours une décroissance du degré de collaboration (par des facteurs ergonomiques comme nous le verrons au chapitre suivant). Mais dans une tâche de conception, la vue personnelle constitue un élément indispensable, mais impossible avec ce support à cause des différences de paralaxe, d'où une impossibilité de désignation simultanée d'un point du modèle et une difficulté de gestion en groupe.

2.1.2. Le projet Continuum

Description

Les environnement collaboratifs amplifiés (*ACE, Amplified Collaboration Environment*) sont des extensions informatiques des environnements de «salle de guerre» ou de projets pour des interactions fortes ciblés autour d'une tâche.

l'EVL[1] propose le *Continnum* [LJP+02] (figure 3.3a) comme démonstration d'un *ACE* reposant sur : une large salle comprenant un écran pour la visualisation de contenu 3D à base de stéréo passive (*AGAVE* [LDT+01]), des écrans plasmas supportant l'affichage de

[1] Electronic Visualization Laboratory, Chicago, USA

(a) collaboration autour du système avec les différents écrans

(b) navigation de deux utilisateurs immergés

(c) interaction collaborative 3D sur molécule

Fig. 3.3: (a) Projet *Continuum* [LJP+02], (b) *CAVE* [CNSD+92] et (c) *PIT* [API+98].

flux vidéos distants (*Acces Grid*[2]), un système optionnel de mur d'écran (pour le partage visuel de données distantes par *TerraVision*) et un écran plasma tactile.

Le *Continnum* permet l'interaction de type annotations à partir d'un stylo passif. L'utilisateur a aussi accès pour l'interaction à un contrôle par portable, *tabletPC* ou *PDA*. Il peur alors interagir sur l'information présentée sur l'ensemble des écrans à l'aide de *SpaceGlider*, architecture logicielle reposant sur *VNC* [VNC]. Les démonstrations ont été faites principalement dans un but scientifique (large quantité de données) pour des professionnels, mais aussi dans un contexte éducatif.

Analyse

Ce projet combine une large variété de supports permettant l'accès simultané à un grand nombres d'éléments hétérogènes (3D, 2D, vidéo, textuelle), l'organisation spatiale conservant les avantages d'une réunion réelle. A ce jour, la définition de méthode d'interaction et l'accès aux éléments est très restreint, en particulier pour fournir un contrôle global dédié à ce type d'environnement. Le traitement collaboratif des ressources est réalisé de façon analogue à l'équivalent réel des salles de guerre, ne fournissant pas de description explicite de l'accès simultané aux données ni de gestion des conflits.

2.1.3. CAVE

Description

Un *CAVE* (ou *Collaborative Virtual Environment*) est constitué d'un espace cubique. Sur 2 à 6 de ces faces on retro projette une image stéréo. Les utilisateur à l'interieur sont équipés de lunettes stéréoscopiques et peuvent interagir principalement avec des périphériques de types pointeur, gant de données ou souris 3D (figure 3.3b). Ils ont une sensation d'immersion omnidirectionnelle (voir [CNSD+92]). Basé sur la librairie *CAVELIB* [CAV] largement répandue, un grand nombre d'applications a été démontré : visualisation scientifique, architecture, artistique, etc.

Le projet *Blue-C* [GWN+03] définit un nouveau système collaboratif d'interaction reprenant les concepts du *CAVE*, en introduisant des écrans semi-transparents[3] permettant l'utilisation de techniques de vision à l'aide de caméras placées derrière l'écran.

Analyse

Cette technologie est fortement limitée par le besoin d'avoir pour chaque utilisateur un point de vue en fonction de sa position dans le CAVE (sinon les utilisateurs doivent se placer près de celui dont la tête est suivie par un capteur). La posture debout et l'immersion forte limitent son utilisation pour de longues sessions (fatigue, pertes d'équilibres et

[2]défini une intergiciel (*middleware*) et un ensemble de ressources pour la communication distante dans des réunions de grande taille ou des présentations couvrant un support audio/vidéo à large bande passante.

[3]le semi désigne ici une non persistance temporelle, la transparence pouvant être affichée fréquentiellement permettant d'alterner vision à travers l'écran et projection sur l'écran

nausées pour les novices). Le risque d'occultations[4] limite les déplacements des utilisateurs et l'absence de support physique réduit l'utilisation d'interfaces naturelles et d'outils réels (cahier de note, portable, etc.).

2.1.4. Système hybride : le PIT

Description

Le *PIT* [API+98] est un système sur table pour deux utilisateurs permettant de restituer un espace virtuel d'interaction au-dessus de la table (figure 3.3c). Le système repose sur l'utilisation de deux écrans perpendiculaires (face à chacun des utilisateurs) et délivrent une vue stéréoscopique à chaque utilisateur. Il introduit plusieurs outils d'interactions tel que des boutons, sliders physiques ou bras à retour d'effort, ainsi que la possibilité d'utiliser un portable ou un stylo d'interaction 3D. La démonstration est faite dans un cadre applicatif de chimie (à partir d'une plate-forme logicielle interne).

Analyse

La première limitation a trait au passage à l'échelle difficile, le système reste contraint technologiquement à deux utilisateurs, voir 3 ou 5. Son très fort intérêt repose sur l'hétérogénéité des outils utilisables. Nous noterons d'un point de vue collaboratif la limitation spatiale du système, une interaction face-à-face nous semblant plus naturelle (voir le contenu alignées dans l'axe des utilisateurs), mais techniquement impossible avec cette solution.

(a) manipulation à deux sur une table virtuelle

(b) manipulation à deux sur une table avec suivi et orientable

(c) manipulation à deux sur une table de plus grand espace

Fig. 3.4: Approche sur table : (a) *Responsive Workbench* [KBF+95], (b) *Fakespace ImmersaDesk* [Imma] et (c) *Tan Holobench* [Hol].

2.1.5. Approche table 3D

Description

Le *Responsive Workbench* [KBF+95] reprend le concept de table numérique pour la manipulation d'objets virtuels 3D (figure 3.4a). L'idée est d'utiliser une rétroprojection sous la table équipée d'un écran horizontal, procurant une visualisation dans un espace au dessus (ou à travers) la table (ou à 45 degrés pour l'ImmersaDesk). Le *Tan Holobench* [Hol] étend cette solution en introduisant un deuxième écran disposé verticalement, donnant à l'utilisateur un espace 3D d'interaction beaucoup plus grand.

Avec une table virtuelle l'utilisation de périphériques 3D de type baguette magique (*wand*), joystick ou gant de données est privilégié. Récemment, différents prototypes ont introduit un couplage avec un dispositif haptique (adjonction d'un bras *Sensable Phantom* [BIJ+00] ou d'une technologie de type *SPIDAR* [TCH+03]). Starner a proposé le *Perceptive Workbench* [LSR+00] en introduisant de nouvelles techniques d'interaction moins invasives re-

[4] un utilisateur entre la vue d'un autre utilisateur et l'écran

posant sur la détection de gestes ou l'utilisation d'éléments tangibles positionnés sur la surface de la table.

Analyse

Les solutions de type table limitent fortement l'espace de visualisation, contraignant l'interaction sur la table et non devant l'utilisateur (dans un contexte de position debout). Les extensions de type *Holobench* comble cette lacune, mais ne permettent pas une interaction face à face mais plutôt côte à côte. Malgré les intéressants résultats obtenus par Agrawala [ABFH97] la visualisation de points de vue multiples reste aussi limitée à deux utilisateurs[5].

L'interaction 3D est aussi difficile, contrainte à une interaction sous faible luminosité à l'utilisation de métaphore complexe et elle est fatiguante. L'utilisation d'images rétro-projetées introduit alors des conflits lors de la manipulation : les bras de l'utilisateur occultent la vision lors de la manipulation. Malgré le support physique procuré par la table et démontré par Coors [CJJ99], Starner [LSR+00] la surface reste trop fragile et tolère mal la présence physique d'outils comme c'est le cas avec des tables réels. Nous retrouvons dans ce cadre un bon support communicatif (même espace d'interaction, vision raisonnable[6] des autres utilisateurs) mais une absence d'un véritable espace personnel comme dans le monde réel (positionner ses outils, ses notes, sa tasse, etc.).

(a) deux utilisateurs plongés dans l'environnement, équipés de casques et gants

(b) partie de jeu en réseau entre amis

Fig. 3.5: Approches (a) multicasques (*iHABS* [WC03]) et (b) multipostes.

2.1.6. Approche individuelle : casques ou écrans côte à côte

Description

Cette classe d'approches est à la limite des relations collaboratives colocalisées. La première catégorie regroupe des approches casques (figure 3.5a). Chacun des utilisateurs est équipé d'un casque de visualisation et de périphériques d'interactions usuels de RV (pointeur, gants, tablette), réunis dans le même espace physique. Ils peuvent conserver les canaux de communication auditifs et verbaux et donc facilement discuter. Nous trouvons par exemple le projet *iHABS* : une simulation basée sur un monde virtuel, dans lequel l'utilisateur navigue avec une montgolfière [WC03].

La deuxième catégorie propose l'utilisation d'ordinateurs personnels (écrans classiques et périphériques d'interaction de bureau ou de RV) où chaque station se trouve dans la même pièce, sous différentes configurations spatiales (alignées, circulaires comme visibles figure 3.5b). Ce type de configuration se retrouve fortement en entreprise (réunion de travail sur

[5]Le concept consiste à alterner à très grande fréquence l'affichage des images gauche et droite à chaque utilisateur, en jouant sur la persistance rétinienne des utilisateurs. Le système fonctionnant à 120hz, il procure alors 30hz pour chaque utilisateur en stéréo (on note que 30Hz est une valeur très faible pour des applications à très forte dynamique). Malheureusement aucune solution n'a été proposée pour dépasser cette limitation.

[6]malgré les faibles conditions d'éclairage

un projet) ou dans le cadre ludique («jeu en réseau entre amis»). Les utilisateurs (malgré l'interface de communication textuelle et l'immersion) utilisent alors très fortement les voies orales de communication.

Analyse

Le premier type d'outils fournit un support personnel de visualisation et un fort degré d'interaction avec le contenu 3D, mais malheureusement la collaboration se limite très fortement à des échanges oraux, les autres canaux de communication se trouvant représenté sous forme virtuelle. Une difficulté supplémentaire est posée par la coopération avec casque (absence de lieu visuel) et le risque d'accident de déplacement entre utilisateurs, car ils n'ont pas de repère visuel réel.

Dans une collaboration devant écran, l'absence de vue commune ou de contexte réel partagé rend alors difficile la communication naturelle : on se retrouve dans une situation similaire à des collecticiels fonctionnant à distance (télépointeur). Dans un jeu, l'absence de référentiel commun peut ne pas poser de problème si les utilisateurs sont à deux positions virtuelles différentes, mais posera plus de difficulté dans le cadre d'une revue en entreprise (discussion autour d'un prototype virtuel). On pourrait imaginer d'autres types de métaphores comme une version 3D multi-utilisateurs de *ClearBoard* (un « *Virtual Showcase* [BES03] amélioré») ou l'utilisation d'ordinateurs portable (vue privée) associée à un large écran (vue publique).

Pour finir, nous noterons que ce cas limite nous semble être utilisé dans un contexte non approprié : en effet un utilisateur peut se trouver en état incohérent entre le contenu visuel perçu et la spatialisation.

2.1.7. Synthèse des approches RV

Nous remarquons en premier lieu que ces approches favorisent généralement la tâche plutôt que l'utilisateur. Nous avons alors généralement des solutions très intrusives, l'utilisateur étant équipé de périphériques (entrées ou sorties).

Les deux classes d'approches (écrans publics ou écrans personnels) semblent posséder chacune des avantages, la solution mixte n'étant pas privilégiée. Les approches privées fournissent un point de vue personnel qui n'est pas possible techniquement avec les systèmes projectifs publics. Les approches écrans publics permettent quant à elles une visualisation naturelle des autres utilisateurs, un large champ de vue. Mais elles sont contraintes par le positionnement de l'utilisateur (côte à côte et non face à face), par le faible éclairage de l'espace de travail (faible luminosité des vidéos projecteurs) et par des méthodes d'interaction peu naturelles (faible conservation d'objets et d'outils réels) contraignant à du «tout virtuel» (métaphores et outils).

Dans un application de visualisation collaborative peu interactive, les approches de types projection murale semblent suffisantes avec un point de vue globalement similaire et l'absence d'outils d'interactions personnels (similaire au cadre collaboratif devant un tableau). Mais por de fortes interactions, la liberté d'avoir un point de vue et la présence d'outils personnels semblent indispensables.

Il ne semble pas y avoir d'outil idéal pour le collaboratif 3D, en particulier pour de fortes interactions. De notre point de vue, la solution de type *PIT* fournit un des plus forts niveaux d'utilisation : visualisation et interaction collaborative simultanée, soutien de son propre point de vue (correct et cohérent grâce à la stéréovision), interaction sous de multiples formes et conservation des outils traditionnels.

2.2. Systèmes basé IHM : informatique ubiquitaire et évanescente

Après les outils de RV, nous allons maintenant analyse la bibliographie de l'IHM, et plus

les travaux colocalisés d'informatique ubiquitaire.

La bibliographie peut être décomposée avec trois catégories majeures. La première catégorie définit les espaces de travails numériques de type pièce (nommé *Roomware*) avec des solutions comme *Colab* (sous section 2.2.1), *I-Land* (sous section 2.2.2), *Interactive Workspace* (sous section 2.2.3) ou *WorkSPACE* (sous section 2.2.4). Pour les systèmes à surface verticale nous avons identifié *LiveBoard* (sous section 2.2.5). Enfin, les surfaces de type horizontale sont présentées avec des systèmes tels que les *SDG* (sous section 2.2.6), les tables digitales (sous section 2.2.7) ou *PDH* (sous section 2.2.8). Un dernier cas à part reste *Peebles* (sous section 2.2.9) qui est une solution hybride.

2.2.1. Colab

Description

Colab [SBF+87] précédemment introduit (section 2.3) est un des premiers environnements de conférence basé sur un support numérique dédié. Il comprend un large écran public sensible au toucher et chaque utilisateur est équipé d'une station personnelle (écran et clavier-souris). Reposant sur une architecture logicielle objet (Loops, Lisp objet) deux applications ont été démontrées : *Cognoter* (outil pour organiser des idées) et *Argnoter* (outil pour considérer et évaluer des propositions). Le système se concentre principalement sur les éléments textuels et intègre une gestion de fenêtres publiques et privées.

Analyse

Ce système offre une approche hybride entre un espace public partagé et un espace personnel. L'architecture logicielle est très performante, basée sur des mécanismes objets, une communication par réplications et offre des outils de développement. L'interaction reste toutefois limitée (pas de support 3D) et il n'y a pas de gestion de données complexes (image, vidéo, etc.). Une large part de l'étude de Stefik a été consacrée à l'organisation du collaboratif d'un point de vue de l'espace privé, de la gestion des conflits, de la perception des actions des autres utilisateurs ou de l'interaction simultanée.

(a) plusieurs utilisateurs coopérant avec les différents dispositifs du système

(b) réunion dans un cadre de conception : utilisation de multiples surfaces simultanément

(c) interaction à deux sur du contenu 3D

Fig. 3.6: (a) *Ambiente* [SPMT+02], (b) *Interactive Workspace* [JFW02], (c) *WorkSPACE* [GMØ01].

2.2.2. I-Land et Ambiente

Description

Le projet *Ambiente* [SPMT+02](précédemment *I-Land*) propose un *roomware* basé sur un ensemble de solutions pour le travail collaboratif dédié à la conception (représenté figure 3.6a). L'environnement repose sur quatre éléments basés sur des technologies à écrans actifs : un écran mural de grandes surfaces, une table tactile, des sièges numériques de travail personnel et des tables connectables (*connecTables* [TPMT+01]). Chacun des éléments définit des métaphores spécifiques et supporte l'echange de données avec les autres éléments. Les composants peuvent être utilisés conjointement. Le projet propose une architecture

nommée *BEACH* [Tan03] : c'est une architecture à couches offrant des services bas niveau jusqu'à une interface haut niveau de gestion de documents (en complément, Streitz inscrit de façon orthogonale à cette décomposition les différents modèles des concepts de bases).

L'interaction se fait sous forme gestuelle par l'utilisation de stylo et par le support des périphériques de bureau (clavier, souris). Il propose la métaphore du *passage* consistant à utiliser un élément physique transitoire portant l'information d'une surface d'interaction à une autre. Les auteurs introduisent en complément le *couplage physique* entre deux éléments comme l'unification d'un espace d'interaction.

La table numérique permet à plusieurs utilisateurs de déplacer des fenêtres 2D positionnées sur l'écran intégré, d'utiliser des gestes comme élément d'interaction ou d'avoir une version clone d'une fenêtre et donc un environnement multi-utilisateurs tenant compte de l'orientation des personnes.

Analyse

L'environnement répond à différentes situations collaboratives (à deux, en groupe, sur table, présentation murale) grâce aux différents dispositifs physiques développés. Nous notons que l'ensemble des technologies utilisées ne permet pas un support stéréoscopique de visualisation 3D. La plate-forme de développement (*BEACH*) semble difficile à utiliser et aucune application n'a vraiment été démontrée avec ce contexte (encore moins pour la manipulation de contenu 3D).

L'interaction est limitée à la 2D et hormis l'interface de reconnaissance de geste peu de détails sont donnés concernant d'autres types de métaphores. D'un point de vue collaboratif, le système n'aide pas l'interaction simultanée, mais introduit fortement la notion de surfaces privées ou publiques. la gestion collaboratif est aussi présent par l'adaptation et l'orientation de contenu affiché en fonction de la position relative de l'utilisateur.

2.2.3. Interactive Workspace

Description

Le projet *Interactive Workspace* [JFW02] (*iRoom*, maintenant *iSpaces*) développé à Stanford s'intéresse aussi à définir un *Roomware* pour la manipulation de contenu numérique sur de multiples surfaces (plusieurs surfaces murales de type *SmartBoard* [Sma], une table numérique et des ordinateurs portables, figure 3.6c). Les applications sont alors réalisées à partir de l'architecture logicielle *IROS* proposée par Johanson, avec principalement le sous-système *EventHeap* [JF02] basé sur une approche d'échanges de messages (chaque message peut être une liste de taille quelconque de divers éléments).

Ils proposent des concepts intéressants d'interface utilisateur tel que *PointRight* [JHWS02] (utilisation d'un périphérique d'entrée unique pour de multiples surfaces), *Multibrowsing* [JPSF01] (l'environnement permettant l'affichage de pages web sur de multiples écrans), *iStuff* [BRSB03] (infrastructure permettant l'utilisation de multiples périphériques comme support d'interaction), *FlowMenu* [GW00] (extension des *Marking Menus* [KB94]) ou *PostBrainstorm* [GSW01] (interaction sur surface murale à partir de stylos et de descriptions par contexte de «feuille»). Ils ont démontré l'utilisabilité de leur système dans des applications de *brainstorming*, de gestion de projet, de visualisation d'imagerie médicale ou de mathématique (*Geometer's workbench* [GWW00]).

Analyse

Ce système offrent une large modularité et dynamicité dans un cadre de réunion sous plusieurs formes (tant du point de vue matériel que logiciel). La large variété des outils proposés permet une interaction avec de nombreux types de données. L'interaction avec des éléments 3D est toutefois limitée à des métaphores de type «trackball» ou visualisation de coupe sur modèle 3D et à la gestion d'objets isolés. Dans ce dernier cas, aucune démonstration n'a été faite pour de l'interaction 3D simultanée (peu de résultats sur le

projets de visualisation 3D). Nous pouvons aussi constater que l'espace de communication réel est conservé, mais peu de détails sont donnés concernant la gestion du collaboratif à travers les éléments numériques.

2.2.4. Interactive Space et WorkSPACE

Description

Le département d'informatique de l'université d'Aarhus s'est consacré depuis quelques années au développement d'un espace de travail permettant le support du design interactif dans un environnement physique. Un grand nombre de leurs travaux a visé à unifier la diversité des documents manipulés au sein d'un même espace virtuel, sorte d'environnement de bureau 3D partagé (*Manufaktur* [BMS+99]).

Ils décrivent dans Büscher et al.[BCG+00] un de leur premier prototype basé sur un *Holobench*. Ils introduisent l'utilisation de *Manufaktur* dans cet environnement, associé à des techniques d'interaction telle que *Go-Go* [PBWI96] et le *WIM* [SCP95] pour la navigation, couplé avec des menus popups 2D.

Ils présentent dans leur derniers travaux, la salle interactive et le *Collaboratorium* [BKP01b] (basé sur trois scénarii inscrits dans un «workflow» de design). La salle interactive (*WorkSPACE*) [GMØ01] utilise le *3D Whiteboard* et *Designer's Workbench* (figure 3.6c). Le *3D Whiteboard* permet à différents utilisateurs de manipuler les documents de *Manufaktur* sur un système mural rétroprojeté. L'interaction repose sur une interface commerciale de type *Mimio* [Mim], une reconnaissance par pointeur laser et l'introduction de deux nouveaux périphériques : le *FingerPen* et le *Command Device* (respectivement interface distante à deux doigts et pointeur complémentaire pour la seconde main).

Le *Designer's workbench* est une table 3D (rétroprojection) où l'utilisateur peut manipuler des documents virtuels 3D (visible par stéréo passive) mais aussi des documents 2D. Ils proposent une nouvelle interface, le *Space Pincher*, sorte de pince permettant de saisir et de déplacer des objets virtuels.

Analyse

Ce projet introduit la nécessité d'une interaction 3D tout en minimisant l'aspect intrusif (sans fils et peu d'équipements sur l'utilisateur, en favorisant la notion de *graspable user interface*). Basée sur une interface de haut niveau (fournis par *Manufaktur*), la mise en œuvre d'applications se fait alors très facilement. Ce projet introduit aussi des outils et une transition 2D/3D permettant de répondre à du contenu sous de multiples formes.

L'ensemble des solutions proposées permet alors de fournir une interaction 3D peu intrusive. Malheureusement, les auteurs soulignent les limitations très fortes de leur système : interaction difficile sur mur avec les métaphores employées ; pas de vue personnelle pour deux utilisateurs sur la table ; pas de métaphores d'interaction 3D développées sur les nouvelles interfaces.

2.2.5. LiveBoard

Description

Introduit en 1992 par Elrod, *LiveBoard* [EBG+92] est un système basé tableau proposant un support dans le cadre de réunion de groupe, de présentation ou de collaboration distante. L'interaction se fait à l'aide d'un stylo sans fil équipé de boutons permettant une interaction par contact ou à distance du support (suivie par technologie optique derrière l'écran). Il propose une architecture logicielle reposant sur une extension du gestionnaire de fenêtre *TWM*. Deux applications servent de démonstrateur : un tableau blanc (pour esquisser des idées) et un outil de présentation.

Analyse

C'est un des premiers systèmes numériques muraux pour réunion collaborative. Le dispositif physique permet une restitution de grande taille, mais ne supporte pas la visualisation 3D, ni l'interaction avec du contenu 3D. De nombreux autres systèmes proposent une interface d'interaction similaire (telle que les systèmes commerciaux *Mimio* [Mim] ou *SmartBoard* [Sma]), mais ils n'introduisent pas de logiciel de base telle que les systèmes commerciaux. L'interaction simultanée n'est pas proposée, ni de réel gestion du collaboratif (mis à part la conservation de l'espace réel de communication).

(a) interaction simultanée devant un écran

(b) interaction simultanée sur table digitale

(c) manipulation simultanée d'objets virtuels à deux mains

Fig. 3.7: (a) SDG [SBD99], (b) *DiamondTouch* [DL01] et (c) *SmartSkin* [fFMoIS03].

2.2.6. SDG (Single Display Groupware)

Description

Les *SDG (Single Display Groupware)* peuvent être défini comme la possibilité d'interagir simultanément sur le même dispositif d'affichage avec plusieurs dispositifs d'entrées [SBD99] (figure 3.7a). L'affichage pouvant être horizontal ou vertical (des environnements spécifiques horizontaux sont présentés dans la suite). Les systèmes proposés cherchent à combler et à fournir un support multi-utilisateurs par rapport aux environnements de bureau, principalement à partir de périphériques 2D classiques (souris) et d'écrans standard de bureau. Au delà de la présentation d'architectures logicielles pour du contenu 2D [TG04], Shoemaker et Inkpen [SI01] ont présenté un système supportant l'interaction 3D basé sur la même approche qu'Agrawala [ABFH97], mais avec des écrans de bureau.

Nous notons que le domaine du jeu basé sur des architectures de type console (muni de plusieurs manettes) définit aussi une catégorie de SDG, chaque joueur pouvant interagir simultanément avec l'application 3D. Les applications actuelles des SDG sont principalement le jeu et les applications artistiques (dessin).

Analyse

Ce type d'architecture se place dans le contexte de collaboration de bureau, la principale difficulté étant la gestion des multiples périphériques d'entrées et le support d'espaces privés. La diversité des boîtes à outils utilisées incluant ces notions offrent un très bon point de départ pour le développpement d'applications 3D.

Un grand nombre de métaphores 3D ont été démontrées dans le domaine ludique sur console, mais peu sont efficaces dans un cadre de travail (et il n'y a pas d'architectures simples dans ce cas). Nous observons que l'extension en visualisation stéréoscopique ne fournit pas en soit de manipulations 3D.

Le collaboratif est largement médiatisé par l'identification des pointeurs des autres utilisateurs et par la gestion d'un espace privé (on notera aussi une conservation naturelle de la sensation de présence des autres utilisateurs). Une limitation reste l'absence de visualisation et de support d'interaction 3D collaboratif.

2.2.7. Tables digitales

Description

Sous catégorie des SDG, ils proposent une approche par écran sur table (*tabletop display*). Par exemple, *Diamond Touch* [DL01] est un nouveau type de table numérique tacile (projection par dessus) développé par le MERL[7] (figure 3.7b). Basé sur une mesure capacitive, le système permet d'identifier plusieurs points de contact simultanés, mais aussi plusieurs zones, ce qui permet alors à plusieurs utilisateurs d'interagir à un doigt, plusieurs doigts ou à deux mains (notion de *table digitale*). Grâce à ces éléments, un boîte à outils a alors été développée reprenant les travaux SDG (nommé *DiamondSpin* [SVFR04]).

Ravin [WB03] a proposé quant a lui de nouvelles métaphores d'interactions dédiées à ce type d'environnement, permettant d'interagir avec du contenu 2D ou 3D. Une démonstration est proposée pour la planification d'architecture d'intérieur.

Récemment une extension *UbiTable* [SKER03] a été proposée complétant la zones publique de la table digitale avec des zones personnelles et des zones privées affichées grâce à des ordinateurs portables. Un système permet alors d'identifier des documents privés ou publics, ainsi que le passage de sa zone privée (sur le portable) à personnelle (sur la table) puis à la zone publique (au centre de la table).

Rekimoto introduit également une solution de table digitale, *SmartSkin* [fFMoIS03], détectant un champ de pression (figure 3.7c). Le système permet à plusieurs utilisateurs d'interagir sur le contenu virtuel projeté sur la table. Il démontre quelques possibilités de ce type de dispositifs en introduisant l'interaction à plusieurs doigts, le déplacement d'éléments par une métaphore de saisie, le menu contextuel autour de la paume de la main et la reconnaissance d'objets. On citera aussi le projet *e-table* [KC02] qui se limite à une interaction à la souris.

Analyse

Ce type de système permet une restitution naturelle de manipulations sur table et donc du cadre collaboratif avec ce type d'environnement. Les principales limitations sont l'impossibilité de restitution visuelle 3D des éléments et la difficulté d'utilisation et de positionnement d'éléments réels (livre, portable) sur le support contraignent actuellement à du «tout numérique». Les derniers travaux introduisent une gestion de l'espace privé/personnel/public et des documents, médiatisant ainsi des éléments importants dans un cadre collaboratif.

2.2.8. PDH (Personal Digital Historian)

Description

le MERL a aussi introduit un prototype de table circulaire (*Personal Digital Historian* [SLM+01]) interactive avec une projection par dessus permettant une interaction avec du contenu multimédia. Le projet est ciblé sur la gestion de contenu digital, dédié à la gestion d'informations personnelles. L'interaction est gérée à partir d'une interface *mimio* et chaque utilisateur est équipé d'un stylo. Les auteurs [VLS02] introduisent des algorithmes de gestion de visualisation dans le cadre de table circulaire, en tenant compte de l'orientation de l'utilisateur. Le prototype repose principalement sur la librairie *DiamondSpin* dédiée à la gestion d'interaction sur table.

Analyse

Malgrès l'utilisation de *DiamondSpin*, ce système offre une interface fortement dédiée et ne gère pas non plus la manipulation ou la visualisation de contenu 3D. Les besoins collaboratifs sont fournis par *DiamondSpin* et par l'interaction de plusieurs utilisateurs sur le même contenu (mais pas de façon simultanée, l'interface *mimio* ne le permettant pas).

[7] Mitsubishi Electric Research Laboratories, Cambridge, USA

2.2.9. Peebles

Description
Myers [MSG98a] introduit dans le cadre du projet *Peebles*, l'utilisation d'assistant digital personnel associé aux différents participants d'une réunion. Cet élément lui sert d'interface d'entrée et de saisie, mais aussi d'espace personnel. Les assistants sont alors reliés à un ordinateur personnel fournissant une surface partagée publique. Il démontre son utilisation dans plusieurs applications : dessin (*PeeblesDraw*), discussion (*PeeblesChat*), contrôle de présentation (*SlideShowCommander*) Contrôle à distance de PC tel que pour des annotations de présentations (*Remote Commander*).
Analyse
Ce type de système offre un contrôle et une interaction personnelle de haut niveau pour chaque utilisateur. Il le contraint faiblement dans le positionnement, lui permettant une interaction à distance. L'interaction et la visualisation 3D n'ont toutefois pas été étudiées. L'aspect collaboratif est largement présent principalement dans le cadre de *PeeblesDraw* par l'intermédiaire de la bibliothèque *Amulet* [Amu] et la gestion d'interaction simultanée.

2.2.10. Synthèse des approches IHM

Les approches proposées sont largement centrées utilisateurs, limitent l'intrusivité des techniques de RV et sont dédiées à de petits groupes. L'approche dominante et l'utilisation un support multiples surfaces/espaces (pièce), sur table ou mural. Toutes reposent sur des systèmes projectifs et des outils 2D de types souris, stylo ou interaction avec les doigts. Les architectures logicielles étendent les interfaces du bureau numérique (gestion de contexte) et fournissent généralement de nouveaux type de boîtes à outils (tel que *DiamondKit*).

Différents travaux se sont intéressés à des interfaces de haut niveau (encore peu étudiées en RV) tenant compte de la multiplicité des documents et de différents types d'espaces. Mais les techniques d'interactions restent fortement limitéess pour l'interaction 3D, reproduisant les métaphores de bureau et se limitant à une «visualisation 3D démonstrative». Peu de techniques spécifiquement 3D ont été développées, les quelques travaux existants cherchant d'abord une continuité entre du contenu 2D et 3D. De plus, la problématique de vue personnelle 3D reste une forte limitation de ce genre de système.

Nous voyons que l'approche moins intrusive de ce type de système ne donne généralement pas l'interaction 3D de haut niveau voulu pour notre cadre d'étude.

Nous choisisons donc d'étudier à présent les travaux de RA et d'envisager leur utilisation après analyse de leurs propriétés.

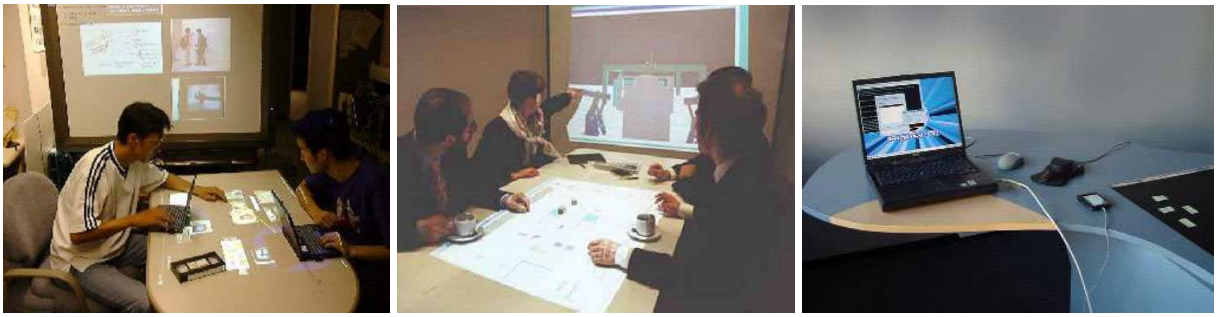
2.3. Systèmes basé Réalité Augmentée 2D

Nous présentons dans cette section les systèmes utilisant la réalité augmentée pour des tâches 3D mais avec une approche 2D (provenant plutôt du domaine de l'IHM). On regroupe sous cette approche les solutions considérant une visualisation principalement 2D (par projection), avec une absence de repère 3D pour le positionnement des objets. La possibilité d'avoir de plus un point de vue 3D personnel suivi (RA 3D) serra présentée à la section suivante (section 2.4).

2.3.1. Augmented Space

Description
Introduit par Rekimoto [RS99], ce travail a démontré une nouvelle approche dans l'utilisation de multiples surfaces de RA[8]. Son concept consiste à étendre l'espace personnel fourni par un équipement personnel (portable), à une zone d'interaction partagée défini

[8] Rekimoto la définit comme *augmented environment*, qui est inclus dans la notion de réalité augmentée



(a) collaboration sur une application de gestion de documents multimédias

(b) application multi-utilisateurs de planification

(c) espace pour un utilisateur : surface privée, périphériques et surface publique

Fig. 3.8: (a) *Augmented Space* [RS99] , (b) *Build-It* [RFK+97] et (c) *OFuturO* [GEG+03].

dans l'environnement réel, c'est-à-dire sur la table (support d'interaction de groupe) et sur le mur (publique), les documents pouvant facilement transiter entre ces surfaces (figure 3.8a). La visualisation dans les espaces partagés est réalisés par une projection murale et sur la table.

Il définit des paradigmes d'interaction tel que *l'hyperdragging* (qui permet à l'utilisateur de déplacer des éléments de son écran vers la table à l'aide d'une souris) ou la manipulation par pointeur laser (suivi par vision) des éléments 3D projetés. Il montre la faisabilité du concept pour la gestion de documents personnels (mélange de documents réels et virtuels) ou de la planification architecturale 3D (conteneur réel ou virtuel d'objets 3D avec manipulation tangible de la caméra). La visualisation 3D est alors réalisée sous la forme d'une vue de dessus par projection sur la table et d'une vue de coupe (par la projection murale).

Analyse

Cette approche permet de combiner l'interaction sur des éléments virtuels avec les possibilités du réel (utilisation de documents réels, d'objets de manipulation). En fournissant une extension des outils de l'utilisateur au lieu de les remplacer, l'adaptation semble plus aisée. La possibilité d'importer et d'exporter des documents numériques ou réels permet alors d'éviter l'emprisonnement dans un espace purement virtuel, permettant aux utilisateurs une utilisation aisée des possibilités des deux mondes.

Les mécanismes d'interaction avec du contenu 3D permettent d'importer des modèles et de les manipuler mais restent toutefois limités à des transformations planaires, soit 3 DDL (Dégrés De Liberté) au total (2 DDL en translation et 1 DDL en rotation). Nous observons d'un point de vue logiciel, de nombreux exemples démonstratifs d'applications mais l'absence de plate-forme de développement. Le collaboratif est médiatisé par la conservation du lien entre l'utilisateur et son pointeur, l'interaction simultanée par *l'hyperdragging* ainsi que la conservation des métaphores de communication naturelle.

2.3.2. Build-It

Description

Morten Fjeld [RFK+97] présente un des premiers systèmes de RA tangibles pour l'interaction simultanée de plusieurs utilisateurs autour d'une table, nommé *Build-It* (figure 3.8b). Son système couple une vue projetée sur la table avec une projection murale dédié à des tâches 3D. Reposant sur des techniques de vision, un utilisateur peut coupler un élément physique (brique rectangulaire) à un élément virtuel.

Il propose différentes métaphores de navigation basées sur l'utilisation des briques physiques pour définir un volume de visibilité du contenu 3D (après avoir proposé dans

[FVB+99] une approche par briques pour fixer les différents paramètres d'une caméra, semblant très fastidieux). Les outils applicatifs sont positionnées autour de la zone de projection centrale et séléctionnable en plaçant une brique dessus. Le système est démontré dans un cadre de planification architecturale. Des récents travaux [Netb] proposent l'utilisation de son système dans la conception architecturale 3D, le prototypage virtuel 3D et la manipulation moléculaire (les résultats prouvent que le système encourage le travail collaboratif et la créativité).

Analyse

Ce système s'inscrit dans une approche de la théorie de l'action, le système définissant une «externalisation» de la représentation de la tâche, la RA réunissant pour l'auteur de nombreux avantages (lire [FV02]). Sa vision permet par exemple le mélange entre cartes 2D réelles (annotées) et une projection de bâtiments virtuels. La grande faiblesse du système repose sur le couplage des deux vues, mais qui nécessite de la part de l'utilisateur un effort cognitif pour construire une représentation mentale de la tâche 3D. De ce point de vue, le système n'est pas vraiment 3D qui conserve la facilité d'interaction sur table.

Nous noterons également que les applications démontrées se limitent généralement à 3 DDL (transformation planaire). Une métaphore supplémentaire permet de manipuler en hauteur des éléments, mais elle n'a pas été montré avec une utilisation simultanée des deux autres degrés en translation. Certains travaux [Netb] ont montré les limitations de ce type de système pour des manipulations 3D plus complexes (telles qu'en chimie où on a vraiment besoin de 6 DDL). Le système permet plusieurs interactions simultanées mais choisit délibérément l'absence d'outils ou de zones privées, conduisant à l'absence de mécanisme de protection de données, de gestion d'objets ou d'outils personnels.

2.3.3. OFuturO

Description

Goebbels [GEG+03] introduit un nouvel environnement de réalité mixte reposant sur une table digitale 3D, une zone personnelle d'interaction et un écran supplémentaire vertical pour la collaboration à distance (figure 3.8c). La table digitale utilise un système à deux projecteurs au plafond, permettant à l'aide de stéréo passive la restitution de contenu 3D. L'interaction est réalisée par les outils classiques 2D, la *Spacemouse* ou une interface digitale (*Cirque Glidepoint touchpad*). Il démontre son système dans un cadre de brainstorming, avec des cartes électroniques et des outils tangibles de modification (basé sur *ARToolkit*).

Analyse

Le système fournit un cadre expérimental pour du mixage 2D/3D et virtuel/réel. Une application 2D est démontrée montrant la conservation des outils réels tels que le portable ou des objets réels de manipulation. Malheureusement la restitution de vue personnelle 3D n'est pas abordée dans l'article, celle-ci étant techniquement impossible avec la configuration choisie (stéréo passive avec un système défini pour quatre utilisateurs). Aucune technique d'interaction 3D n'est introduite, des interfaces sont justes présentées, c'est pourquoi nous l'avons mis dans cette section 2D. D'un point de vue collaboratif nous retiendrons principalement la division de l'espace, mais qui nous semble très rigide (la conception de la forme de la table fixant le nombre d'utilisateur, quatre dans le cadre du prototype proposé).

2.3.4. Synthèse des approches basés RA 2D

L'utilisation de la réalité augmentée semble fournir un cadre intéressant permettant une manipulation de contenu numérique tout en conservant la connaissance et les possibilités du réel. La projection de contenu numérique dans un espace connu (réel), couplée avec des outils connus (numérique ou réel), favorise l'interaction et l'apprentissage par l'utilisateur.

Contrairement aux environnements de RV, mais de façon similaire aux approches IHM, la communication et la coordination se font dans le monde réel. Nous remarquons cependant que les architectures logicielles ne sont pas présentées, ne fournissant pas d'aide à la création d'application dans ces environnements.

Ces travaux de RA permettent de manipuler du contenu 3D avec des méthodes plus évoluées que les approches d'informatique évanescente (section 2.2) en offrant soit deux vues sur le contenu 3D soit une projection stéréo passive. Bien que ces approches favorisent une interaction naturelle et intuitives, elles ne fournissent pas les possibilités des systèmes de RV telle qu'une véritable visualisation 3D pour des utilisateurs suivis, une manipulation 3D simultanée de haut niveau (isométrie 3D ou modélisation 3D), ni une vue personnelle comme avec les casques de RV. Un système de plus haut niveau consisterait donc à conserver les avantages de la RA tout en fournissant une meilleur logistique pour l'interaction 3D : cette approche existe et est définie sous le terme de réalité augmentée 3D, présentée à la prochaine section.

2.4. Systèmes basés réalité augmentée 3D

Nous introduisons ci-après les systèmes de RA permettant une visualisation 3D. Nous définissons la RA 3D par la présence d'une navigation par un suivi de mouvement par capteur, la possibilité de changement de point de vue personnel et le positionnement 3D des éléments dans un espace commun (et de façon optionnelle la vision stéréoscopique).

Nous présenterons en premier lieu des des environnements plutôt génériques : *Seamless Design* (section 2.4.1), *EMMIE* (section 2.4.2), *Studierstube* (section 2.4.3), *Shared Space et Tiles* (section 2.4.4). Puis nous présenterons les environnements utilisant plus fortement la présence de la table : *ARTHUR* (section 2.4.5), *MixDesign* (section 2.4.6), *MagicMeeting* (section 2.4.7), *DWARF avec Sheep* (section 2.4.8) et *SCAPE* (section 2.4.9).

(a) collaboration avec deux utilisateurs à deux mains avec capteurs magnétiques

(b) les différentes interfaces de visualisation hybride

(c) collaboration dans le cadre d'un storyboard 3D, avec de multiples fenêtres 3D

Fig. 3.9: (a) *Seamless Design* [KTY99b] , (b) *EMMIE* [BHF+99] et (c) *Studierstube* [SFH+02].

2.4.1. Seamless Design

Description

Kiyoshi Kiyokawa [KTY99b] présente le système d'interaction sur table, *Seamless Design*, basé sur l'utilisation de deux casques optiques semi-transparents (suivi par capteur), les utilisateurs étant équipés de manettes d'interaction 3D (figure 3.9a). L'environnement proposé est dédié au prototypage rapide d'objets 3D basé sur des contraintes. Il introduit différents concepts d'interaction 3D : curseurs 3D, conteneurs de primitives 3D, widgets 3D pour le choix de couleur et un curseur de navigation.

Basé sur une étude de l'interaction en RA par rapport à la RV, le système permet de s'immerger dans l'environnement virtuel, par l'utilisation d'un «curseur 3D» de transition

qui change le mode de fonctionnement du casque en mode RV et place le point de vue de l'utilisateur dans le monde virtuel à la position de son avatar (peut être interprété comme un dérivé du *World In Miniature, WIM*). Leur prototype, utilisant plus le concept de table que le concept de réunion face-à-face, démontre l'assemblage d'objets virtuels en RA et repose sur des techniques classiques de contraintes (contraintes sur grilles, détection de collision, contraintes hiérarchiques par graphe de scène).

Analyse

Malgré la faible ergonomie du dispositif, ce projet reprend un certain nombre de techniques d'interaction 3D généralement utilisées en RV couplées avec la communication et coordination naturelle supportées par le contexte de la RA. Les utilisateurs peuvent ainsi naturellement échanger des idées, coopérer sur des actions 3D tout en conservant un cadre naturel.

Nous notons que la table n'a pas ici un grand rôle, servant principalement à fixer un référentiel visuel de support de la tâche. Le projet ne propose malheureusement pas d'infrastructure logicielle pour la réalisation d'autres applications. Les possibilités du réel sont aussi sous-estimées par rapport aux approches 2D, ne favorisant ici que la communication. Nous constatons qu'une étude approfondie a été menée concernant certains mécanismes collaboratifs, telle que la direction de regard des autres utilisateurs ou l'intérêt de la vision du monde réel [KTY99a].

2.4.2. EMMIE

Description

Feiner [BHF+99] propose l'environnement *EMMIE*[9] basé sur la réalité augmentée, qui fournit un espace virtuel (nomme «ether») qui englobe les utilisateurs, les périphériques et les systèmes (figure 3.9b). Les utilisateurs positionnés autour d'une table peuvent alors visualiser à l'aide de casques ou d'écrans suivis des objets virtuels 3D dans cet espace. Grâce à des périphériques spécifiques, les utilisateurs peuvent manipuler le contenu multimédia, en les positionnant sur d'autres objets virtuels (représentant des applications) ce qui permet de les exécuter (e.g lire un son avec un lecteur).

Il introduit de plus un nouveau type d'*interaction hybride* : un utilisateur peut déplacer des objets virtuels depuis les écrans (représentation 2D) vers un espace (représentation 3D) défini au dessus de la table, mais aussi lier données 2D et 3D entre surface et espace (visible à travers le casque par superposition d'un espace et une surface). Il propose des techniques de gestion de données privées basées sur des lampes virtuelles et des miroirs ne réfléchissant que les données publiques des utilisateurs. Basé sur leur architecture *Coterie* [MF96], le système est entièrement distribué et implanté dans le langage compilé *Modula*. L'application démontrée est la gestion de documents multimédias, l'architecture logicielle ayant été validée dans d'autres applications (telle que dans le projet *MARS (Mobile Augmented Reality Systems)* [HFT+99]).

Analyse

Feiner propose ici à partir de la RA un environnement introduisant - dans un contexte 3D - une interface de haut niveau pour la gestion de multiples ressources hétérogènes. La conservation d'un repère réel, le support de la table et la facilité de manipulation 3D en RA semble prometteuse.

De plus, dans le cadre de tâches de base (manipulation, navigation) peu de techniques nouvelles sont introduites, celles utilisées ne sont en général ni évaluées ni justifiées. Des mécanismes de gestion du collaboratif sont explicitement traités (protection des données), mais d'autres fortement nécessaires manquent (support interaction simultanée).

2.4.3. Studierstube

[9] Environment Management for Multi-user Information Environments

Description

Studierstube [SFSG96] est un large projet mené par Dieter Schmalstieg depuis de nombreuses années à l'université Technique de Vienne, comprenant un grand nombre de travaux relatifs à la RA, à l'approche collaborative et ainsi que des applications dédiées (figure 3.9c). *Studierstube* est un environnement de réalité augmentée multi-utilisateurs basé sur des casques personnels et de multiples périphériques d'interaction (pointeur, gant, *TouchPad*, interface à reconnaissance vocale, etc.). L'interaction repose initialement sur le concept du *PIP* (*Personal Interaction Panel*) [SG97b], tablette personnelle suivie associée à un stylo d'interaction suivi.

L'approche choisie repose sur une volonté de fusion des concepts des environnements de bureau numérique (standard), du travail collaboratif et de l'informatique ubiquitaire. Dans un article, les auteurs [SFH00] proposent une décomposition basée sur trois éléments clefs : les notions de *locale*, *window* et *context*. Le *context* décrit les composants d'une application (donnée, représentation, code). Les *window* définissent une representation 3D d'une application. Le *locale* représente des référentiels réels («conteneur» de données) qui sert à positionner des contextes.

La principale caractéristique de ces travaux est une architecture logicielle couplée à une interface de haut niveau qui reproduit les environnements de bureau standard : les fenêtres 2D sont étendues à l'espace, les événements 2D étendus à la 3D, et l'interface de programmation est présentée sous de multiples formes (appel en retour *callback*, boucle principale ou *signal/slots*). Ce travail a ensuite été étendu avec une version distribuée [SRH02], ou une architecture spécifique pour la gestion des périphériques (*OpenTracker* [SR01]) et avec une gestion haut niveau du contenu à partir d'un langage descriptif de scénario *APRIL* [FL03].

Les démonstrateurs sont très nombreux : mathématiques [HK03], visualisation scientifique [FLSG98], médecine [Sys], navigation [GR03], jeu [Ulb03], etc. L'environnement est suffisamment générique pour permettre une extension à d'autres domaines.

Dans un cadre collaboratif, deux prototypes sur table ont été crées : un jeu de mah-jong [SEG98] et le projet *AR conferencing table* [Taba]. Le premier introduit un concept de jeu, la table définissant l'espace partagé par les joueurs où ils peuvent positionner leur pions (la zone personnelle est représentée par le PIP). Le deuxième se base sur un concept d'écran de RA, équipés de caméras vidéos avec une visualisation du contenu virtuel positionné au centre de la table. L'interaction se limitant à la sélection d'élément par un capteur 3D sur un cahier (conteneur d'objets virtuels) ou un changement de l'élément sur la table par une interface souris dans un menu.

Analyse

Cet environnement est sans doute un des plus évolués en terme d'architecture, d'infrastructure de développement, d'interfaces et de démonstrateurs. Le succés du projet (nombre et utilisabilité du démonstrateurs, collaboratif 3D) justifie les choix matériels (casque, PIP) et logicielle (architecture, gestion des droits). Nous noterons que les prototypes sur table n'utilisent pas forcément tous les avantages apportés par ce support (placement d'éléments personnels, retour pseudo haptique, etc.).

2.4.4. Shared Space et Tiles

Description

Le projet *Shared Space*[BWF98] propose un environnement collaboratif où deux utilisateurs équipés de casques peuvent interagir grâce à une interface vocale dans une application hypertexte collaborative, le contenu virtuel 2D étant positionné dans l'espace (figure 3.10a). Les auteurs [BKWF99] introduisent aussi pour un système de RA distant (télécon-

(a) navigation par interface vocale

(b) collaboration avec l'interface tangible entre un utilisateur equipée d'un casque et l'autre par visualisation sur un écran

Fig. 3.10: (a) *Shared Space* [BWF98] et (b) *Tiles* [PTB+02].

férence) . Dans ce projet, Ils présentent alors l'utilisation de RA tangible (*tangible AR*) : marqueurs réels associés à des objets virtuels pour la manipulation et le positionnement d'éléments virtuels dans un référentiel réel (dans son contexte des polygones texturés avec l'image vidéo des personnes en communication distante).

Récemment Poupyrev a proposé *Tiles* [PTB+02], un environnement reposant sur le concept d'éléments virtuels associés à des cartes réelles (figure 3.10b). Ces cartes réelles sont typées (contenu, outils, container) et peuvent être associées (par relation de proximité induisant une action prédéfinie) introduisant un contexte d'interaction de haut niveau. Le démontrateur propose une interface pour la conception d'interface de cabine de pilotage d'avion. Les utilisateurs conservent dans ce contexte les surfaces réelles pour positionner les éléments (muraux) ou faire des annotations réelles. L'ensemble de ces travaux repose sur une bibliothèque dédiée au suivi des marqueurs, *ARToolkit* [KB99].

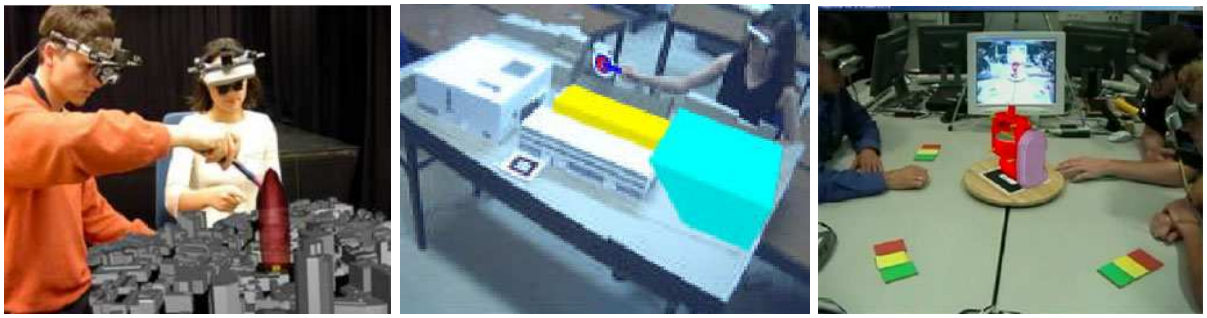
Analyse

Au-delà de la définition de plusieurs configurations matérielles, Bilinghurst introduit principalement le concept d'interfaces avec association objet réel/virtuel en privilégiant le casque comme périphérique de visualisation. *Tiles* définit véritablement un environnement complet mais l'interaction se limite à des objets virtuels 2D, mais manipulables dans un référentiel 3D.

L'interface de couplage physique/virtuel peut être problématique si on a un grand nombre d'objets ou des objets de petite taille (la taille minimale de suivi des marqueurs physiques étant assez grande pour une utilisation à distance «normale»). L'architecture offre une plate-forme très facilement accessible, simple mais reste de bas niveau (peu de modules de gestion implémentés). L'aspect collaboratif a aussi été très étudié par Bilinghurst, telle que les espaces de communication et d'interaction simultanée [BBGK02], mais peu d'éléments ont été introduits dans le même environnement et ré-utilisés de prototype en prototype.

Il est à noter que la bibliothèque *ARToolkit* est diffusée gratuitement et est très largement utilisée par d'autres auteurs (aussi intégrée dans notre environnement comme nous le verrons dans le chapitre suivant). Malgrès les avantages d'un suivi temps réel 6D et d'un suivi simultané de plusieurs marqueurs, cette librairie est encore peu robuste (problèmes d'occultation et de *jittering*).

2.4.5. ARTHUR



(a) un utilisateur changeant l'échelle d'un bâtiment

(b) maquette hybride : réel et virtuel

(c) revue en RA avec le Cake Platter et les marqueurs d'annotations

Fig. 3.11: Solutions sur table : (a) *ARTHUR* [GMS+03], (b) *MixDesign* [DMS+02] et (c) *Magic Meeting* [RWB02].

Description

Wolfgang Broll propose le prototype *CAMELOT* (précédemment projet *Virtual Round Table* [BMS00]), suivi du prototype *ARTHUR (Augmented Round Table for Architecture and Urban Planning)* [GMS+03] basé sur un même concept et dédié à la planification urbaine et à l'architecture (visible figure 3.11a). Le prototype *CAMELOT* introduit un environnement de visualisation (avec casques) et d'interaction d'objets virtuels sur table, basé sur une architecture matérielle légère et se voulant à faible coût (PC de bureau). Dans le cadre du projet *ARTHUR* chaque utilisateur, équipé d'un casque optique de haute résolution suivi et de caméras, peut interagir avec des objets virtuels à l'aide de *«placeholder»*, objet réel associé à des objets virtuels (approche tangible avec un suivi par vision). Plus récemment, il propose en complément des techniques de reconnaissances de geste par vision pour la manipulation et l'accès au contrôle de l'application [MSL+04], ainsi que l'utilisation de la métaphore baguette magique. L'ensemble repose sur une architecture logicielle distribuée développée dans un cadre de RV : *DWTP* (*Distributed World Transfert and communication Protocol*) [Bro98].

Analyse

Ce système proposant une interaction et une visualisation de haut niveau (suivi par vision, casque de haute qualité) mais qui n'a pas à ce jour montré de résultats probants. Le suivi par vision reste très limitant : initialement réduit à un suivi à 3 DDL, le système permet aujourd'hui un suivi à 5 DDL pour la baguette magique (avec deux caméra), mais aucun résultat n'a été publié dans un cadre multi-utilisateurs avec des performances acceptables (le suivi à 3 DDL fonctionnait à 10 Hz, celui à 5 DDL à 7 Hz). Peu de résultats sont donnés sur le fonctionnement de l'association objet virtuel/réel, sur la manipulation ou sur la gestion du contrôle de l'application.

Nous noterons aussi qu'à ce jour aucune évaluation globale ni outils dédiés au domaine proposés n'ont aussi été présentés (tel que le mixage avec des documents ou maquettes réels). Nous remarquons que les aspects collaboratifs, telle que la manipulation simultanée ou la gestion de zone personnelle autour de la table, ne semblent pas été étudiés.

2.4.6. MixDesign

Description

Dias présente un nouveau prototype, MixDesign (figure 3.11b), dédié à l'architecture [DMS+02]. Il permet l'interaction simultanée de plusieurs utilisateurs sur une maquette mixte positionnée sur une table. Les utilisateurs sont équipés de casques vidéo semi-transparents. Au delà de la description des intérêts identifiés dans le domaine de l'architecture, il introduit différentes techniques d'interactions basées sur des interfaces tangibles.

Il présente l'utilisation de la palette virtuelle de Kato [KBP+00] pour déplacer des objets virtuels ou pour faire de la reconnaissance de geste et l'utilisation d'objets tangibles pour les transformations usuelles.

Plus récemment [DJC+03], il a introduit l'utilisation de techniques digitales basées sur l'espace image pour manipuler les propriétés des objets ou accéder à des menus 2D (deux marqueurs associés aux pouces des utilisateurs). Il propose une architecture bas niveau [DSB+03] sous la forme d'une extension d' *ARToolkit*.

Analyse

Reposant fortement sur *ARToolkit*, la robustesse du suivi est limitée. Les techniques d'interactions digitales récentes semblent plus souples mais souffrent de l'approche basée image et limitent donc l'interaction naturelle. L'aspect applicatif a été peu développé à part l'ajout d'éléments virtuels ou le mode de représentation.

2.4.7. MagicMeeting

Description

Le projet *MagicMeeting* [RWB02], développé au centre de recherche de *Daimler Chrysler*, s'intéresse à définir un environnement de RA sur table pour des réunions dans le domaine de l'automobile. Basé sur l'utilisation d'un suivi par la librairie *ARToolkit*, chaque utilisateur est équipé d'un PDA, d'un casque vidéo semi-transparent et peut manipuler un objet virtuel positionné sur un élément réel tangible central, le *Cake Platter* (plateau tournant). Le système fournit en complément deux écrans positionnés de chaque côté de l'espace libre (figure 3.11c).

Ils introduisent différents outils d'interaction (éclairage, plan de coupe, annotations), ainsi que des interfaces (souris 2D, souris 3D). On peut aussi utiliser les fenêtres des applications de bureau directement dans l'espace virtuel (projetées sur une surface réelle). Il définit en complément le passage entre différents espaces et surfaces :

– entre un PDA et *Cake Platter* ;
– entre PDAs par communication infrarouge ;
– depuis le *Cake Platter* vers une application 2D de bureau et inversement.

Tout ceci repose sur une architecture à composant propriétaire, largement inspiré de *CORBA CCM* [CCM].

Analyse

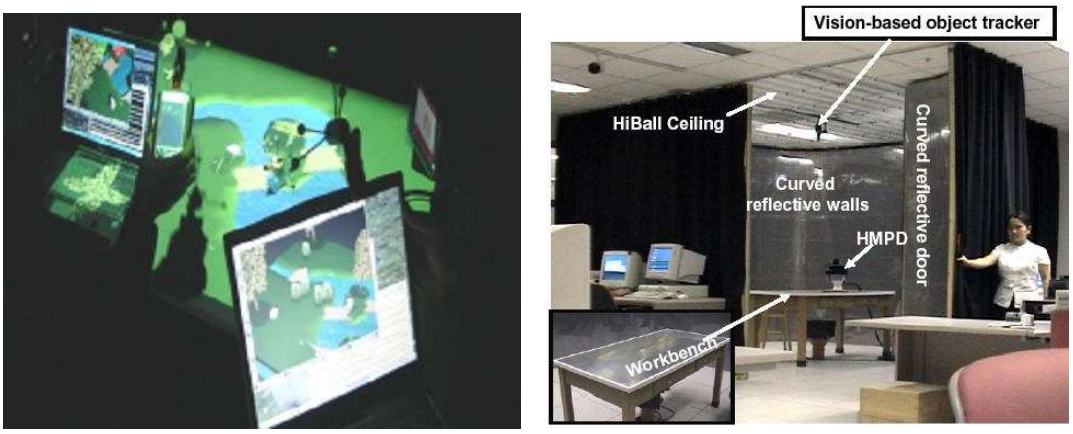
Le système proposé prend fortement en compte les possibilités de la table (espace et outils personnels, support, orientation). Malheureusement, comme le souligne l'évaluation, la robustesse du suivi est très faible, ce qui rend l'interaction difficile. Les annotations sont très limitées ne procurant pas une interface naturelle et de haut niveau. Le choix délibéré du placement du contenu sur le *Cake Platter* limite l'interaction, la gestion de grosses scènes et la manipulation des éléments (pas de changement d'échelle, ni de forte translation).

Les possibilités d'associations d'espaces/surfaces fournissent des mécanismes et des solutions d'interface de haut niveau plus étendus que ceux introduits dans le domaine par *EMMIE* (section 2.4.2) mais elles restent démonstratives, non formalisées et très limitées (interaction PDA limitée au transfert). L'aspect collaboratif est partiellement étudié : actions simultanées (pas démontrées véritablement), identification des utilisateurs.

2.4.8. DWARF et Sheep

Description

Ce projet est un cas limite tenant plus du prototype et de l'architecture logicielle que de l'environnement, mais ses propriétés nous semblaient assez intéressantes pour le présenter. *DWARF (Wearable Augmented Reality Framework*[BBK+01] est un environnement logiciel



(a) vue extérieure avec les multiples surfaces

(b) vue du système avec la table et le mur cylindrique

Fig. 3.12: Autres approches : (a) Sheep [SMW+02] et (b) SCAPE [HBGA03].

permettant un développement rapide d'applications. Dans ce contexte, une démonstration a été développée nommée *Sheep* [SMW+02]. LA démonstration est une application de jeu (de plateau, sur un concept de poursuite «loup-mouton», voir figure 3.12a). Il propose un grand nombre d'interfaces d'entrées : pion tangible suivi, reconnaissance vocale, interaction avec PDA, interaction manuelle (saisie d'objet et modification) ou interface graphique standard sur portable. Il propose aussi beaucoup de dispositifs de sorties : PDA, écran de portable, casque semi-transparent optique et projection sur table.

Divers outils ont été proposés autant pour la gestion de l'application que pour son développement basé sur l'architecture distribuée à composant de DWARF : outil de surveillance d'activités des composants, outils de deboguage ou outils de tests. D'un point de vue collaboratif, différents utilisateurs peuvent alors interagir simultanément avec les différents dispositifs, basé sur des métaphores d'interactions simples, le système contrôlant automatiquement l'interaction multiple.

Analyse

L'hétérogénéité des solutions proposées par ce système constitue un fort intérêt pour la RA, permettant une adaptation à la tâche et aux préférences de l'utilisateur. L'architecture logicielle de haut niveau permet la mise en œuvre rapide d'applications. Malgré la diversité des interfaces, les auteurs soulignent les limitations des métaphores proposées qui nécessitent de fortes améliorations. De plus l'aspect collaboratif n'a pas été étudié à l'exception de la gestion de l'interaction simultanée de bases fourni par l'architecture.

2.4.9. SCAPE

Description

Hong Hua [HBGA03] introduit un nouveau type de système collaboratif : *SCAPE (Stereoscopic Calibration in Augmented and Projective Environment)*, illustré figure 3.12b. Basé sur des casques projectifs (HMPD), Hua combine l'utilisation d'une table rétro-réfléchissante et un dispositif semi-cylindrique rétro-réfléchissant. L'utilisateur peut alors accéder à une visualisation de contenu 3D sur la table (vue exocentrique de la scène) et une vue immersive sur l'écran cylindrique (vue égocentrique).

L'interaction se fait sous la forme de pions tangibles positionnés sur la table et associés à un avatar (représentant chaque utilisateur), une loupe tangible ou un *CoCube* [BH03] (permettant de sélectionner des éléments, leur visualisation et l'accès à un complément d'information sur ces éléments). Une architecture logicielle est proposée reposant sur une

interface objet distribuée et sur le concept d'acteur. Le démonstrateur présenté est spécialisé dans la navigation architecturale (archéologie).

Analyse

L'approche proposée permet de dépasser les limitations techniques imposées par le faible champ de vue des casques classiques et les problèmes d'occultations : en effet, le principe de fonctionnement permet de toujours avoir une occultation visuelle si un objet réel se trouve entre la surface rétro-réflechisante et l'écran du casque de l'utilisateur. Malheureusement, cette technique reste difficile à mettre en œuvre : elle nécessite un environnement adapté telle que la surface spécifique de la table. De plus, l'occultation automatique fait défaut pour une interaction aisée avec les autres objets de l'environnement (on ne peut avoir d'objet virtuel sur un objet réel et donc devant lui dans l'axe de visualisation de l'utilisateur).

La possibilité de changement de mode de navigation et de type (AR/VR) offre à l'utilisateur une grande souplesse. L'aspect collaboratif n'est pas étudié. Nous avons juste la possibilité d'activité en sous groupe mais la résolution des cas complexes n'est pas décrite[10].

2.4.10. Synthèse des approches RA 3D

Ces approches combinent les possibilités de la RV (manipulation 3D, angle de vue propre) avec les avantages des approches IHM (conservation des outils réels, manipulation tangible et digitale/manuelle plus aisée). L'utilisation de dispositifs plus intrusifs par rapport à la RA 2D réduit l'utilisabilité de ces approches, mais semble actuellement la seule solution pour des systèmes multi-utilisateurs extensibles avec une visualisation 3D.

On retrouve aussi un grand nombre d'environnement de programmation de haut niveau permettant une création rapide d'applications. Les méthodes d'interactions offrent une facilité de manipulation avec du contenu 3D mais aussi 2D, virtuel aussi bien que réel. De plus, des interfaces de plus haut niveau ont émergé fournissant un cadre générique de représentations, de gestion et de développement d'applications.

Une évolution récente tend maintenant vers la définition d'environnement augmentés dédiés, orientés par exemple vers les réunions sur table. Mais malheureusement, au delà des considérations ergonomiques, ce type de système introduit des difficultés qui seront abordées dans la suite de la thèse (chapitre 4 et chapitre 6).

3. Synthèse globale

Nous présentons ici une synthèse globale des différentes approches des sections 2.1 à 2.4. Nous nous intéressons d'une part à la validitation de l'intérêt de la RA par rapport à notre problématique et ensuite à la place de nos travaux par rapport à la bibliographie existante.

Résumé

La problématique soulevée initialement est le besoin d'un environnement collaboratif colocalisé dans le cadre de tâches 3D. Nous avons représenté quatre grandes classes d'approches provenant de la littérature représentés figure 3.13.

- La première (RV) fournit un environnement optimal pour la manipulation et la visualisation de contenu 3D, mais l'aspect collaboratif reste très limité, la définition propre de métaphores 3D est difficile et souffre de limitations techniques pour du travail collaboratif («vue 3D» exacte, faible éclairage). Le débat reste largement ouvert dans la communauté sur le choix du support d'interaction 3D dans ces environnements. Nous

[10] En effet, les deux utilisateurs peuvent avoir un point de vue spatiallement séparés sur le mur cylindrique. Mais peu d'explications sont fournies pour avoir un point de vue commun ou la possibilité de réunir le point de vue lorsque les deux utilisateurs sont immergés dans un même lieu.

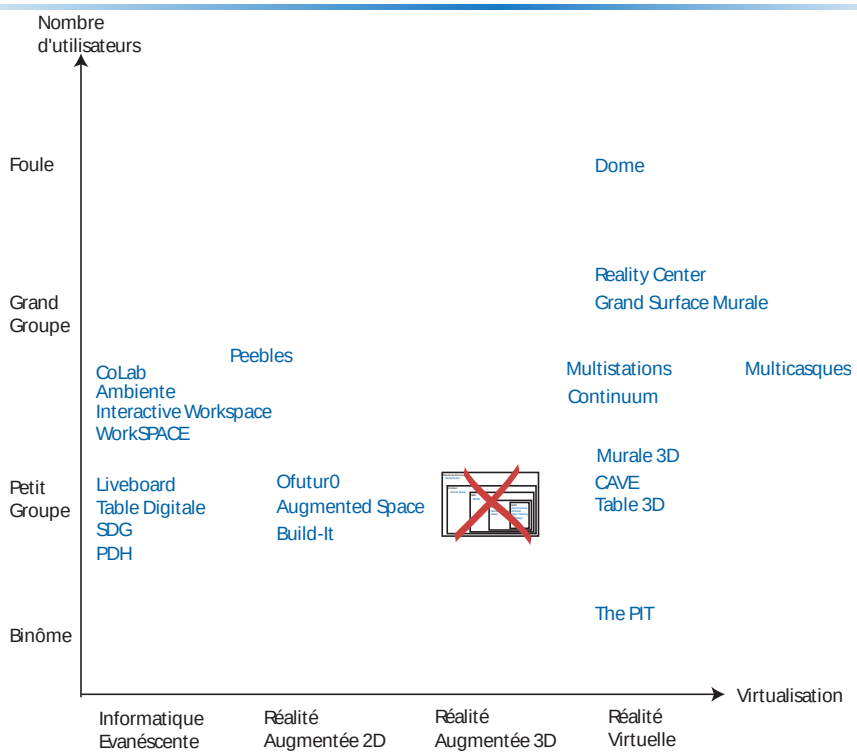


Fig. 3.13: Contexte de recherche : RA 3D pour petit groupe (on cible notre recherche dans la catégorie symbolisée par la croix).

pourrions le résumer ainsi : «faut il fournir une nouvelle manière d'interagir, de penser et de voir avec des éléments virtuels 3D ou tenter d'inclure les méthodes d'interactions et les outils du réel (stylo, papier, table, objets tangibles, portable, PDA) ?».

- Les approches interaction homme-machine offrent en revanche des interfaces peu intrusives, avec des métaphores d'interactions naturelles et intuitives, une bonne gestion du collaboratif mais peu de techniques dédiées à des tâches 3D.
- La réalité augmentée 2D semble fournir un compromis entre ces deux approches, favorisant l'utilisation mixte d'éléments réels 2D/3D (physiques ou numériques), la conservation de l'aspect collaboratif et interactif avec les besoins de tâches 3D. Mais les techniques proposées restent toutefois loin des méthodes introduites en RV.
- La réalité augmentée 3D fournit alors un support idéal regroupant :
  - le cadre d'interaction et de visualisation 3D de la RV ;
  - le support collaboratif et d'interfaces naturelles et intuitives de l'IHM ;
  - la manipulation de documents 2D/3D des systèmes de RA 2D.

  Les démonstrations d'environnement de RA ont prouvé leurs intérêts pour des situations en petit groupe, qui constituent aussi un point de notre motivation.

Au vu de cette analyse, le cadre de RA 3D nous semble un environnement propice comme possible solution répondant à notre problématique et valider l'intérêt de notre approche. Nous étudierons donc par la suite plus précisément un possible cadre de réalisation d'un environnement correspondant à notre contexte de recherche.

Contexte de recherche

Nous nous intéressons donc à la possible mise en œuvre d'un environnement de RA collaboratif colocalisé dans le cadre de tâche 3D (figure 3.14). Dans cette classe de travaux, nous pouvons établir un classement à partir de critères spatiaux.

De notre point de vue, une approche sur table semble reproduire les contextes de situations réelles permettant une utilisation plus souple que des approches «debout», où dans un environnement spécifique : nous privilégions donc cette piste. Nous observons que certains

travaux utilisent la table uniquement comme référentiel ou support (*EMMIE*, *Seamless Design*, *Sheep*) et d'autres s'intéressent à utiliser ses propriétés dans un cadre applicatif (*ARTHUR*, *Magic Meeting*, *MixDesign*, *Tiles* et certains travaux de *Studierstube*).

Sur ce dernier point nous pouvons alors noter que :

- des techniques d'interactions ont été présentées mais restent très limitées et non dédiées au support de la table : utilisation de la surface lors des déplacements, retour haptique faible, place de l'interaction manuelle et sa relation avec la table, etc ;
- des interfaces de haut niveau (gestion des fenêtres, utilisateurs, applications) ont été proposées mais non dédiées à un cadre sur table (tels que les travaux en 2D de *DiamonSpin*) ;
- de nombreux éléments étudiés en TCAO et IHM concernant la gestion des besoins collaboratifs ont été faiblement repris dans ce contexte ;
- une large part des systèmes limitent fortement la conservation des outils réels (physiques ou numériques), aucun démonstrateur ne montre véritablement une augmentation de l'environnement sur table, mais plutôt un remplacement (de façon plus réduite pour *MagicMeeting* et *Tiles*) ;
- enfin, la fusion des deux mondes reste limitée à la présence de la table réel ou à la simple démonstration d'une faisabilité d'une incrustation réel/virtuel.

Il nous semble donc intéressant d'orienter notre thèse sur la mise en œuvre d'un environnement de RA sur table en se basant sur une *intégration naturelle et utile du support de manipulation 3D*.

Nous noterons qu'au début de ces travaux (Septembre 2000), excepté le prototype de *Studierstube* aucun travail ne proposait une approche sur table. Le choix d'une nouvelle approche dédié sur table était encore plus justifié. Nous avons tout de même voulu présenter ici les travaux apparus entre temps, les notres s'adaptant et réagissant aux différentes approches proposéess au cours de ces dernières années, guidant et raffinant aussi la définition des besoins et des critères importants.

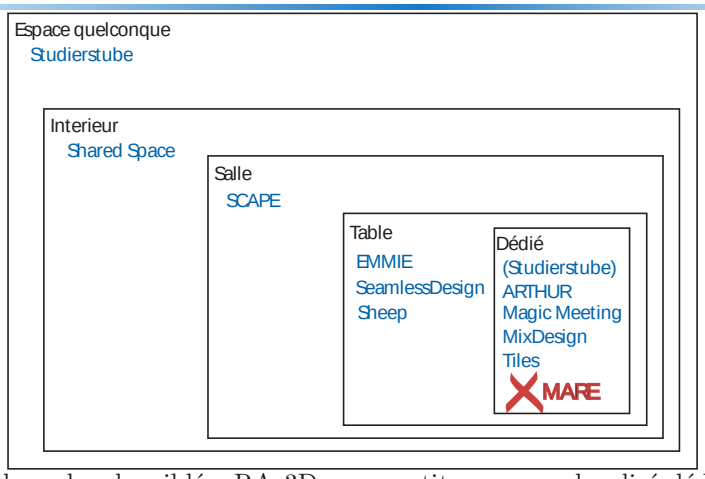


Fig. 3.14: Contexte de recherche ciblé : RA 3D pour petit groupe colocalisé dédié sur table (on cible notre recherche dans la catégorie symbolisée par la croix).

Décomposition en sous problématiques : analyse et présentation

Comme présenté dans le chapitre 2, l'approche retenue favorisera le développement d'un environnement complet par rapport à une étude précise d'un élément particulier. La décomposition en sous problématiques nécessite une approche fortement itérative, raffinant les éléments important et cherchant à les identifier. L'ensemble des travaux que l'on a réalisés, pendant ces trois ans, nous ont permis de réaliser cette décomposition, complétée par les points identifiés par les travaux précédents. Dans ce contexte, nous avons retenu

quatre grandes classes d'éléments qui seront présentés au cours de cette thèse. Chaque classe définit des besoins, des approches et les solutions apportées par nos travaux.

Brièvement les quatre classes sont :
- **architecture** : définit l'infrastructure matérielle et logicielle pour la réalisation d'un environnement de RA collaboratif ;
- **mixage** : définit la gestion de la fusion entre les deux mondes d'un point de vue spatial (alignement) et visuel ;
- **interaction** : définit les méthodes et techniques pour manipuler les objets virtuels ou réels, 2D ou 3D et la fusion du monde réel et virtuel ;
- **collaboratif** : définit la gestion et le respect du contexte collaboratif de notre environnement telle que la gestion des interactions simultanées ou l'observation d'une action par tous les autres utilisateurs.

4. Conclusion

Ce chapitre a présenté une synthèse des principaux travaux relatifs liés de près ou de loin aux environnements collaboratifs colocalisés pour des tâches 3D. Une large étude et analyse bibliographique nous a montré l'intérêt de la RA 3D dans ce cadre. De plus, un certain nombre de problématiques non résolues par les solutions existantes nous conduisent à la proposition d'un nouvel environnement. L'approche choisie est une solution sur table, avec le but de maximiser les mélanges entre les deux mondes et de minimiser les modifications d'un contexte de réunion réelle.

La décomposition choisie pour l'étude de l'environnement a été définie à partir de l'analyse retenant quatre éléments principaux présentés dans la suite : architecture (chapitre 4), mixage (chapitre 5), interaction (chapitre 6 et 7) et collaboratif (chapitre 8).

Chapitre 4

Architecture

> Il est de règle que l'architecture d'un édifice soit adaptée à sa destination de telle façon que cette destination se dénonce d'elle-même au seul aspect de l'édifice.
>
> —Notre Dame de Paris. Victor Hugo.

1. Introduction

L'architecture définit l'infrastructure d'un système ou d'une application. Elle peut être décomposée en deux sous branches, l'architecture matérielle et logicielle. La problématique soulevée dans ce chapitre peut se résumer ainsi : «Quelle architecture matérielle et logicielle générique est appropriée pour des applications collaboratives colocalisées sur des tâches 3D ?» (figure 4.1).

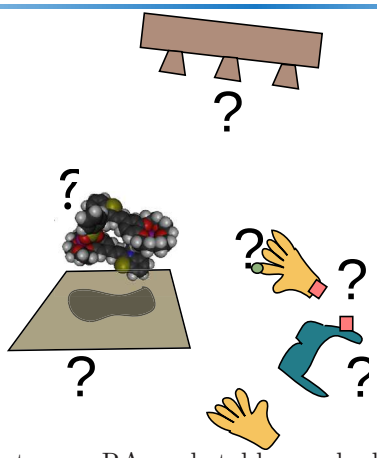


Fig. 4.1: Problématique de l'architecture en RA sur la table : quels choix des dispositifs ? quelles relations avec les utilisateurs ? quel positionnement des éléments ?

Nous nous intéressons au développement d'une architecture pour un système collaboratif 3D en réalité augmentée. Nous souhaitons que cette architecture soit ouverte pour supporter un grand nombre d'expérimentations, mais aussi de prototypes applicatifs. Nous décomposons l'étude de cette architecture en deux parties. Nous présenterons dans ce chapitre l'architecture matérielle mise en place (section 2) et l'architecture logicielle développée et dédiée à la réunion 3D sur table (section 3).

2. Architecture matérielle

2.1. Introduction

L'architecture matérielle définit les constituants physiques nécessaires à la réalisation des tâches d'une application. Nous y trouvons divers éléments tels que des systèmes de calcul pour l'exécution des tâches ou des dispositifs d'affichage pour visualiser une représentation 3D d'une application.

La définition et la mise en œuvre de ce type d'environnement reste aujourd'hui un travail de recherche tant par l'absence de méthodologie ou de techniques, que par les difficultés de paramétrages et d'évaluations. Nous présentons dans la suite les besoins identifiés et la

méthodologie retenue (section 2.2), l'approche poursuivie pour chacun des besoins (section 2.3 à 2.8), la mise en œuvre et les résultats (section 2.9) pour finir par une synthèse (section 2.10).

2.2. Besoins et approche

Nous introduisons en premier lieu l'approche choisie et les besoins identifiés.

Vue générale de l'approche

La réalisation d'un système ou d'une application peut amener à distinguer deux types de besoins, les requis fonctionnels et les requis non-fonctionnels. Les *requis fonctionnels* vont définir les fonctions liées à la tâche (e.g. manipuler des objets, simuler, modéliser, etc.). Les *requis non fonctionnels* [MCN92] définissent quant à eux les propriétés non nécessaires à la tâche mais garantissant la qualité du système. On trouve par exemple la robustesse, l'efficacité, la sécurité, etc.

Nous proposons ici de reprendre ces éléments dans le cadre de la définition de notre architecture matérielle. Nous allons nous appuyer sur les besoins fonctionnels pour contraindre le choix matériel de notre environnement. Par exemple, le besoin fonctionnel d'une visualisation 3D fortement immersive conduit à éviter l'utilisation d'écran d'ordinateur portable. Orienter vers une architecture générique, nous ciblerons donc les fonctionnalités élémentaires de toute application de RA 3D collaborative colocalisée.

Dans la suite nous présenterons en détails chacun des éléments fonctionnels identifiés en précisant les besoins non fonctionnels spécifiques. Notre méthode se décompose alors en différentes étapes très standard : la définition des besoins, l'approche choisie, la mise en œuvre et l'évaluation. La mise en œuvre repose elle aussi sur une décomposition très simple : le positionnement, la définition des connectiques, les réglages internes des dispositifs (étalonner et paramétrer) et les tests.

Nous nous baserons principalement sur une *modélisation visuelle géométrique* pour la phase conceptuelle (voir [AMD04], [RPBM03] ou dans le cadre de la RA [DSAP03]). L'idée est de représenter différentes contraintes identifiées dans les besoins ou fixées dans l'approche, sous une forme visuelle et par une représentation géométrique 3D. La conception est alors guidée et étudiée partiellement avec cette représentation. Par exemple, représenter visuellement l'espace de suivi d'un capteur en fonction de l'espace nécessaire à la tâche guidera le choix du capteur et son positionnement.

Cette modélisation est réalisée à partir des mesures effectuées par Laurence Boissieux de notre espace de travail. Nous nous reposons de plus sur un logiciel standard *Discreet 3DS Max V 5*© [Max] ayant des bonnes propriétés d'édition et de modification de scènes 3D. Quelques exemples sont présentés en annexe A.

Enfin, nous proposons un certain nombre d'outils pour les tests et évaluations des différents éléments offrant un ensemble d'utilitaires pour la gestion et l'évolution de notre architecture (voir section 3).

Vue général des besoins

Nous avons retenu principalement trois classes de besoins fonctionnels :

- **base de communication** : fournir les moyens de promouvoir la communication, la coordination et l'échange collaboratif entre les individus. Les besoins non-fonctionnels dans ce cadre sont principalement les performances systèmes (coût mémoire, vitesse d'exécution), la taile de l'espace de communication, etc. ;
- **base d'interaction** : fournir les moyens à l'utilisateur de visualiser et d'interagir avec les éléments virtuels (3D ou 2D), de connaître le contexte de l'utilisateur (son état vis à vis de l'application) et de lui fournir un retour multimodal. Les besoins non-fonctionnels dans ce cadre sont principalement des critères ergonomiques (confort, utilisabilité, effi-

cacité), les critères visuels et les critères d'«interface fortement couplée» (voir la thèse de Berard [Ber99]) ;
– **base d'application** : fournir les moyens d'accéder et de gèrer une application (outils et données). On peut citer comme critères non-fonctionnels dans ce cadre : le temps réel, la représentation des données, l'interface de programmation (langage).

La suite présente plus en détails chacun des éléments fonctionnels identifiés : support communicatif (section 2.3), puis la base d'interaction décomposé en base visuelle (section 2.4), base contextuelle (2.5), base interactive (2.6) et base audio (section 2.7). Enfin la base d'application est définie d'un point de vue matériel par le support de calcul (section 2.8).

2.3. Support communicatif : la table

2.3.1. Introduction et besoins

Les activités collaboratives colocalisées nécessitent le positionnement de plusieurs interlocuteurs dans un même *espace de communication* (verbale ou gestuelle). L'activité se déroule dans un espace, nommé *espace des tâches*, qui peut prendre différentes positions relatives à l'espace de communication (couplée, séparée, à 10 cm de distance, etc.). Nous pouvons distinguer différents cas régulièrement utilisés dans les activités courantes : discussion informelle debout, discussion autour d'une table, discussion autour d'un tableau, etc.

Notre cadre d'étude est principalement dédié à des tâches collaboratives liées à des situations de type réunion. Dans ce contexte, nous allons voir que l'utilisation d'un support physique est très utile. De plus, nous montrerons qu'un *support de type horizontal, telle qu'une table*, offre alors une solution intéressante dans un cadre collaboratif en petit groupe avec une forte interaction entre utilisateurs.

Une première justification est liée aux capacités physiques humaines limitées, principalement dans le fait d'être en position verticale sans support physique pour des tâches de longue durée. La table fournit alors un support pour le positionnement des bras. Elle fournit aussi un support pour le reste du corps, associé à la position assise, améliorant la possibilité de travail pendant de longues périodes.

Une autre justification est la base de communication et de tâche qu'établit une table (voir Greenberg [GC89]). La table offre les avantages suivants :
– elle permet de gérer un espace personnel, autorisant l'utilisateur à déposer ses éléments propres et à effectuer des tâches privées (telle que prendre des notes) sur l'activité en cours ;
– elle permet d'établir un espace public directement sur le support physique. Cet espace peut servir comme base pour la tâche, mais aussi comme espace d'échange d'éléments hors tâches (système de projection, ensemble de documents, etc.) ;
– elle permet de définir une orientation d'un utilisateur par rapport aux autres individus et à la tâche. L'orientation fournit un contexte de visualisation et de présentation propre ;
– la proximité favorise l'interaction. Les espaces publics de communication définis pour chaque personne sont en intersection ;
– elle permet un accès simultané de tous les utilisateurs à la tâche, contrairement à une configuration de type tableau.

Il nous semble donc que le table fournit une bonne base pour du travail collaboratif. Nous considérons que le choix de cet élément nécessite une étude plus précise. Dans ce but, nous présentons notre approche dans la partie suivante.

2.3.2. Approche : caractéristiques et ergonomie

(a) table ronde pour petit groupe

(b) table «bateau» pour groupe de taille moyenne

(c) composition de table pour réunion en large groupe

(d) travail collaboratif en petit groupe

(e) présentation avec et sans support physique

(f) réunion en large groupe

Fig. 4.2: Exemples de tables et implications collaboratives.

Pour étudier plus précisément les caractéristiques de la table, nous nous sommes basés sur une brève revue bibliographique d'ergonomie et une étude des travaux liés à la TCAO avec des solutions sur table. Nous nous sommes aussi appuyés sur une observation informelle des table commerciales (figure 4.2) à partir plusieurs sites web[1]. Nous en présentons maintenant les résultats.

Nos observations informelles nous ont amenés à identifier différentes propriétés d'une table :

- caractéristiques géométriques : taille, forme (carrée, rectangulaire, ronde, forme bateau, etc.) ;
- caractéristiques de composition : autonome, extensible, composable ;
- caractéristiques physiques : apparence, matière, rugosité, etc (voir Coutaz [CLBB02] pour les propriétés de surfaces en IHM).

La recherche en TCAO s'est beaucoup intéressée aux caractéristiques des environnements sur table. Magerkuth et Tandler définissent dans un de leurs articles [TM02] trois *types de tables* possibles :

- **table de réunion** : ils la caractérisent par un grand nombre de personnes autour d'une table, où l'activité principale est la communication orale. L'importance de l'espace partagé est minimum et l'espace privé est essentiel (prise de notes, lecture de documents). L'augmentation de la taille de la table réduit la quantité et la facilité des échanges communicatifs ;
- **table de collaboration** : de plus petite taille avec un faible nombre de personnes, la tâche se trouve sur la table, facilement accessible et visible par les différents intervenants. L'espace privé est minime et laisse place à l'espace public ;
- **table d'ambiance** : table de rencontres informelles (comme une table de café). En général sert de support temporaire avec peu d'accès numériques. Elle peut alors servir dans le cadre de réunion que nous désignons par *réunion spontanée*.

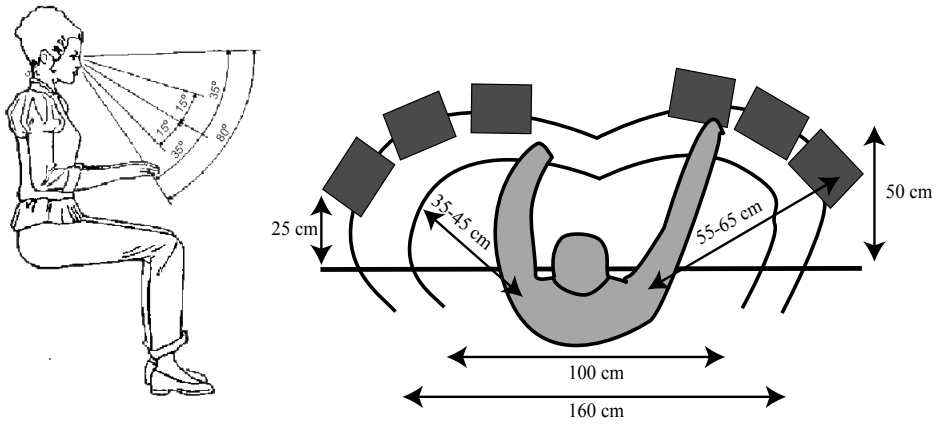
La tâche guide alors le choix des propriétés. Dans notre thèse, nous nous orientons principalement vers le type de configuration de catégorie table de collaboration, mais aussi table de réunion.

[1] *Smith Office*[Off], *Herman Miller*[Mil], *Dupontavice*[Dup], *Bedimo*[Bed], *Yahoo Shop Store*[Sto]

Les différents propriétés géométriques sont adaptées à l'utilisateur suivant des propriétés ergonomiques reposant sur des mesures anthropomorphiques [tAMHYCWK02]. Le schéma suivant figure 4.3 montre le bon *positionnement d'un utilisateur* dans le contexte d'un travail sur un bureau [oEUSoA] : bras à 90 degrés accoudés à la table, dos contre la chaise, regard à 45 degrés pour visualiser les éléments sur la table. D'un point de vue d'activités manuelles sur la table, il faut alors prendre en compte deux critères qui définissent l'*espace de travail* d'un utilisateur :

– espace de travail normal : espace accessible avec les mains sans déplacer les bras des côtés du corps ;
– espace de travail maximum : espace accessible en étirant les bras au maximum.

Dans ce contexte, l'utilisateur doit pouvoir travailler dans l'espace normal, qui peut être de façon occasionnelle remplacé par du travail dans l'espace maximum (actions peu fréquentes et de courte durée).



(a) (b)

Fig. 4.3: Tables et ergonomie : (a) position d'un utilisateur [oEUSoA] et (b) espace de travail défini par les caractéristiques anthropométriques d'un individu [BLEE+99].

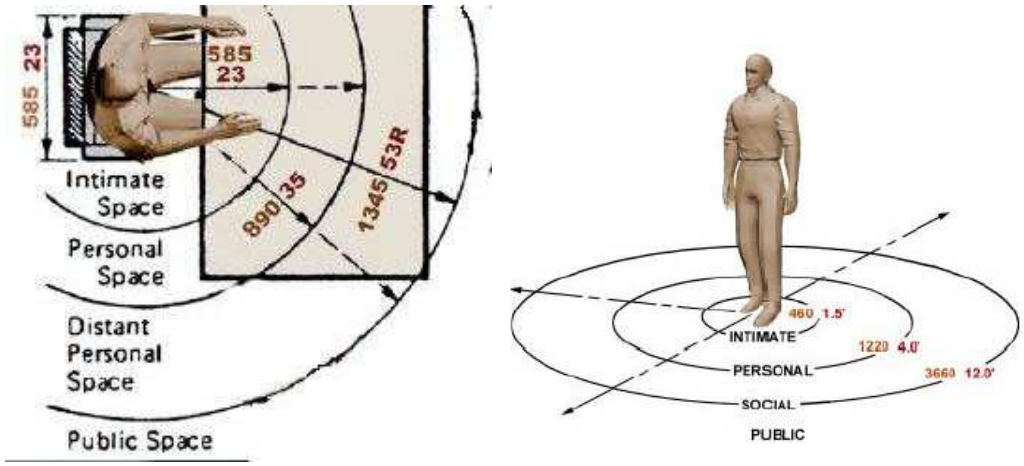
Dans un cadre collaboratif, les concepts d'ergonomie [Div] définissent des *espaces de communications* usuels centrés sur l'utilisateur. Ils sont décomposés suivant la distance à l'utilisateur, nous distinguons : **espace intime**, **espace personnel**, **espace personnel distant**, **espace public** (figure 4.4). L'intersection entre ces espaces définit le type d'échange communicatif dépendant fortement de la place et du rôle des personnes. Un cadre intime est plutôt réservé et limité en accès avec un échange fort. Un cadre public ne met plus les personnes en relation (pas de contact) et la trop forte distance entre intervenants réduit l'utilisation des métaphores naturelles de communications. La distance et l'agencement des utilisateurs autour de la table définira alors le type de collaboration.

La table devra donc tenir compte de la tâche, des critères ergonomiques mais aussi des besoins nécessaires pour l'efficacité des périphériques utilisés. Par exemple, une table avec de la visserie métallique dégradera les mesures de capteurs magnétiques.

A partir de ces éléments nous décrivons maintenant les choix réalisés :

– forme : une table de forme rectangulaire. Ce choix se justifie par les propriétés de composabilité de ce type de table[2], de sa prépondérance dans la vie active, de ses nombreuses possibilités de configuration ;
– hauteur : nous nous appuierons sur les valeurs moyennes des études anthropomorphiques permettant un respect des critères ergonomiques de travail sur table ;

[2]Ceci permet d'envisager une extension des travaux menés ici à des configurations plus larges, comme des configurations en U en V, ou des tables de grande taille.



(a) (b)

Fig. 4.4: Espace de communication : (a) dans un cadre de travail sur bureau (données fourniis en pouces et en cm) [Div] et (b) en position debout dans un cadre de discussion [Div].

– surface de travail : une table de petit format, fournissant un meilleur cadre de fortes interactions et communications entre les utilisateurs. La taille est définie pour respecter différentes solutions de configurations entre plusieurs utilisateurs. Elle est dédiée à un support de collaboration, avec conservation d'espace privé de réunion. Enfin la taille a été contrainte par la couverture spatiale du capteur (présenté dans la suite des travaux) ;
– rugosité : une table lisse qui favorisera l'utilisabilité d'interface tactile ou digitale directement en contact avec la table ;
– apparence : une table de couleur neutre et peu réfléchissante,, qui n'interfère pas avec les algorithmes d'extractions de couleur des systèmes de vision ;
– matière : table entièrement en bois, pour éviter des interférences magnétiques avec des systèmes de suivi de mouvement magnétique.

Nous nous intéressons à partir de la définition de ce support de communication (et d'interaction), au support de visualisation du contenu 3D dans la prochaine section.

2.4. Base visuelle

2.4.1. Besoins

Nous nous intéressons aux moyens nécessaire à mettre en œuvre pour visualiser des éléments virtuels 3D ou 2D à partir de l'espace communicatif de la table.

Facteurs ergonomiques : souhaitant une visualisation d'éléments virtuels 3D dans un espace réel, le dispositif de visualisation doit s'approcher des caractéristiques de visualisation d'objets réels. D'une part, fournir la restitution des signaux perceptuels 3D classiques : parallaxe de mouvement, vision stéréoscopique. Il faut alors supporter les caractéristiques physiologiques de notre système visuel (figure 4.5) : vision binoculaire (accommodation, vergence), résolution (1 arc/minute), champ de vue (150x135 degrés par œil, 120x130 degrés de champ binoculaire) et plages de couleurs visibles (380 à 730 nm). D'autre part, fournir des dispositifs légers et confortables, voir non intrusifs.

Facteurs collaboratifs : la visualisation devra être faite de façon simultanée dans le temps et dans l'espace pour tous les utilisateurs. Le système doit permettre la restitution d'un espace de visibilité commun entre utilisateurs : il doit fournir la possibilité de visualiser les gestes indicatifs des autres utilisateurs (geste déictique). Le système devra assurer une vue personnelle à chaque utilisateur possédant leur propre angle de vue. Le système

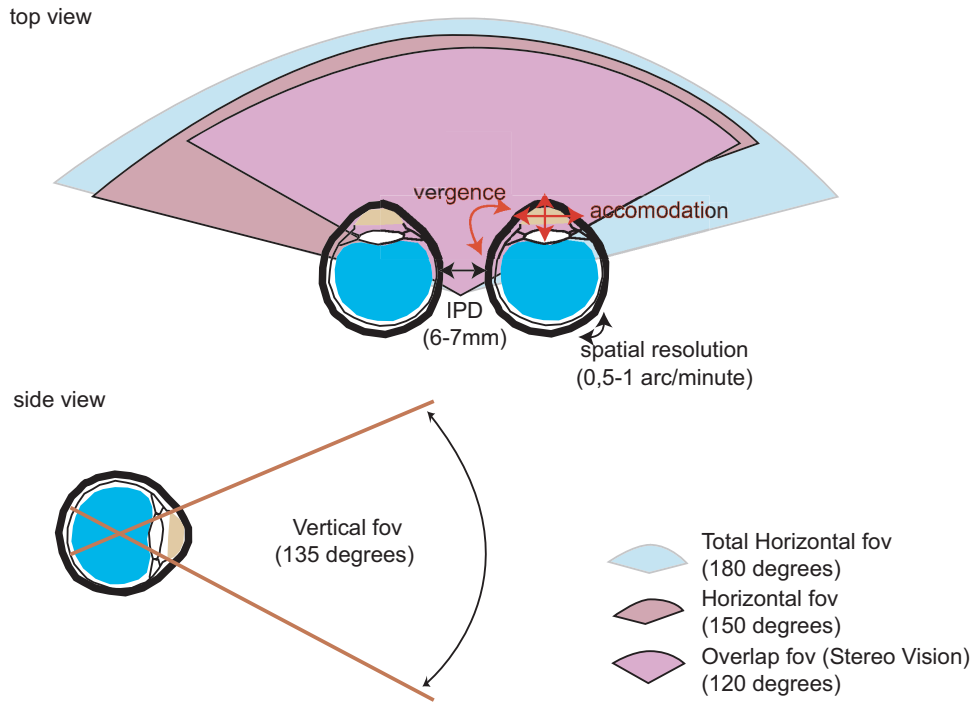


Fig. 4.5: Caractéristiques du système visuel humain et valeurs quantitatives (échelle non respectée ici).

devra permettre l'affichage d'information privée pour chaque utilisateur (pouvant n'être affiché que pour une seule personne).

Facteurs interaction : le dispositif ne devra pas être en conflit avec un quelconque dispositif d'interaction. Le système devra conserver la totale liberté de mouvements de l'utilisateur (dans la configuration assise, pour la tête et les mains).

2.4.2. Approche

Nous avons choisi de pouvoir fournir *une visualisation de contenu 3D dans l'espace parallélépipédique défini sur la table*, le plan inférieur reposant sur le dessus de la table (figure 4.6). Cette solution semble la plus évidente pour conserver l'espace de communication lors d'une réunion et peut coupler dans un même référentiel spatial contenu réel et virtuel. De ce fait, une interaction directe, similaire au contexte de réunion, est alors envisageable avec les objets virtuels 3D (et 2D positionné sur la base de l'espace c.à.d le dessus de la table).

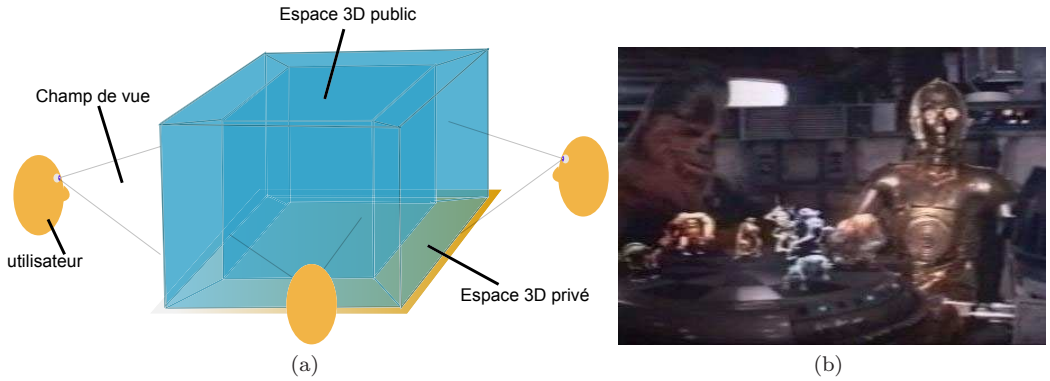


Fig. 4.6: (a) Espace de visualisation souhaité et (b) système de visualisation 3D «idéal» : table de jeu virtuelle dans le film *Star Wars* [Cop77].

Comme le rappelle Halle [Hal97], il reste techniquement impossible de projeter du contenu directement dans l'espace sans médium. De ce fait, après avoir étudié diverses solutions (voir figure 4.7) nous avons choisi une approche par casques équipés d'un capteur de suivi ayant de nombreux avantages :

- affichage non contraint à un espace physique : le contexte de visualisation est défini dans le repère de l'utilisateur, des informations peuvent être alors positionnées et visualisées dans tout l'environnement (hormis les considérations de suivi) ;
- totale liberté de visualisation de l'utilisateur : le point de vue d'un utilisateur n'est pas contraint, visualisation suivant son propre angle de vue ;
- persistance temporelle de la visualisation : le casque reposant sur la tête de l'utilisateur l'information 3D est persistante, limitant la fatigue induite par les solutions que l'on tient à main ;
- possibilité d'afficher des informations personnelles : l'association d'un périphérique privé à un unique utilisateur donne la possibilité à celui-ci de visualiser des informations privées ;
- pas de modification de l'environnement réel : l'espace de communication et d'interaction reste identique à une situation réelle, l'utilisateur étant augmenté. Les approches de type *Reachin*[Rea] ou bras articulé nous semblent trop contraignantes car elles modifient l'espace de travail et de communication ;
- support efficace de mixage des deux mondes : nous n'avons pas de conflit réel vers virtuel, l'information virtuelle étant toujours affichée avant le réel (mais il reste un conflit virtuel vers réel comme étudié au chapitre suivant).

D'un point de vue ergonomique nous choisirons un dispositif léger, adapté à la morphologie des utilisateurs. Chaque utilisateur étant équipé de casque, nous pouvons alors leur restituer une impression de contenu virtuel 3D dans l'espace de la table.

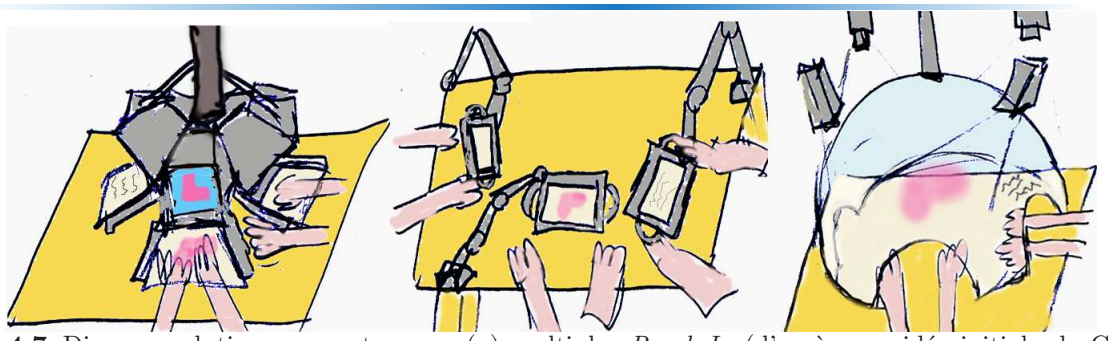


Fig. 4.7: Diverses solutions non retenues : (a) multiples *Reach-In* (d'après une idée initiale de Gilles Debunne et démontrée récemment dans [MB04]) (b) multiples RA *Boom Chameleon*[TFK+02] ou (c) *Virtual Showcase*[BES03] hémisphérique.

Dans le même esprit que la démonstration *Sheep* (chapitre 3, figure 3.12) nous proposons une solution hétérogène de visualisation en utilisant des dispositifs de visualisation complémentaires au casque semi-transparent (voir annexe A). Nous reprenons la décomposition proposée par Rekimoto [RS99] pour les différents espaces de travail dans un cadre de réunion : espace de visualisation personnel, de groupe et public en introduisant dans notre cas une hétérogénéité 2D/3D.

Vue personnelle : les casques nous offrent une restitution tridimensionnelle au dessus de la table. Un utilisateur peut souhaiter accéder à des documents 2D ou avoir la possibilité de visualiser un contenu 3D avec une vue supplémentaire. Pour cela il nous a semblé utile d'adjoindre un dispositif physique supplémentaire avec une bonne résolution (au lieu d'afficher ces informations directement sur le casque). Nous proposons dans ce cas l'utilisation d'une approche de RA «basé écran semi-transparent» reposant sur un ordinateur portable : il convient parfaitement à une approche d'utilisation à faible fréquence, il peut

fournir une visualisation de contenu 2D en haute résolution et est faiblement intrusif.

Vue de groupe : elle doit permettre un accès à courte distance de contenu 2D ou 3D ou sert à définir une vue publique pour un petit nombre de personnes autour d'une table. Nous avons retenu deux solutions : écran orientable et écran statique positionnés sur la table. L'écran orientable peut fournir une vue partagée entre utilisateurs (en mode RA) mais aussi servir comme surface collaborative pour des documents 2D (visualisation page web, résultats, simulation etc.). L'écran statique ne sera utilisable que comme surface d'affichage de contenu 2D en haute résolution, ou de visualisation en 2D d'éléments virtuels 3D (avantages d'une meilleur résolution par rapport au casque). Par exemple, dans un cadre architectural, un projet de construction pourra être affiché dans l'espace de la table, l'affichage de documents techniques ou d'un modèle 3D sélectionné sur l'écran de groupe.

Vue publique : elle se caractérise par l'utilisation d'une vue commune pour tous les utilisateurs. Un suivi n'est alors pas nécessaire. Ses propriétés sont le support d'un grand champ de vue et un affichage en haute résolution pour l'affichage de contenu spécifique. On pourra l'utiliser pour afficher des scène de grande taille, du contenu 2D (cartes ou grand ensemble de documents) ou une autre vue 3D commune de la scène. Nous avons choisi une approche par affichage murale avec une visualisation 2D (pas de complémentarité d'utilisation des techniques stéréo avec les contraintes du casque).

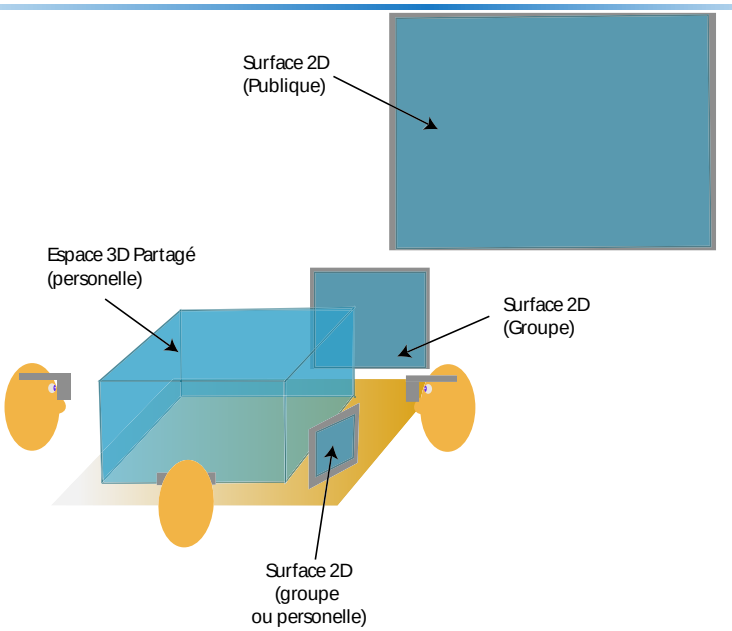


Fig. 4.8: Schéma de notre approche avec les différents espaces et surfaces retenus.

L'approche proposée s'inscrit dans les espaces naturels de travail collaboratif (personnel, groupe, publique) tout en fournissant une hétérogénéité sur le type de contenu adapté aux possibilités des dispositifs.

2.5. Connaître le contexte de l'utilisateur

Nous nous intéressons à connaître le «contexte» de l'utilisateur dans notre cadre collaboratif. Le contexte est défini ici au sens de l'IHM, voir par exemple Crowley [CCRR02] pour plus de détails.

Besoins

Un des critères nous semblant majeur est la position et posture du corps de l'utilisateur, telle que la position de la tête (pour la visualisation stéréoscopique) ou des mains (pour l'interaction). On note aussi le besoin de connaissance de l'état des objets utilisés. D'autres

facteurs expressifs peuvent être aussi pris en compte : la direction du regard, l'expression faciale.

Critères ergonomiques : Le système doit permettre un suivi des éléments caractéristiques des parties du corps de l'utilisateur en mouvement. De plus, le système idéal doit être transparent pour l'utilisateur : sans fils, léger, conservation de la liberté de mouvements, discrétion visuelle.

Critères collaboratifs : le système doit permettre un suivi de plusieurs utilisateurs. Il devra fournir un service adapté au rôle de chaque utilisateur dans un cadre collaboratif.

Critères applicatifs : le système doit permettre un suivi dans tout l'espace de travail. Il doit minimiser la présence d'éléments non nécessaire à la tache dans l'espace de la table. Le système doit fournir un suivi en haute précision, stable, peu bruité et robuste (occultations, conditions environnementales) avec un nombre de DDL adaptés à la tâche. Enfin, le système doit permettre le passage à l'échelle du nombre d'éléments ainsi que leur identification.

Approche

Nous avons choisi de limiter notre approche au suivi de mouvement de l'utilisateur et des objets, en se reposant sur une solution hétérogène et adaptée aux types d'éléments suivis (figure 4.9). Nous considèrons le suivi de la tête, de la position des mains, des doigts et d'objets positionnés sur la table. Nous souhaitons un suivi dans l'espace personnel mais aussi dans l'espace public (espace central). Nous considérons une approche adaptée à l'espace de suivi : nous souhaitons n'avoir pas d'objets avec des capteurs très intrusifs dans l'espace public, peu dans l'espace personnel et un nombre acceptable dans l'espace intime. Comme aucun système de suivi actuel peut répondre à tous ces critères, nous fusionnons trois techniques adaptées à ces différents contextes.

1. Considérant que les techniques de vision et de suivi de marqueurs naturels ne sont pas utilisables dans notre cadre[3], nous nous sommes reportés vers des *approches par vision outside-in* (voir définition chapitre 2) offrant un large champ de vue et une forte précision. Nous utiliserons cette technologie pour le suivi de la tête et des mains. L'utilisation d'une technologie avec fils réduit l'utilisabilité. Toutefois l'utilisateur étant en position statique et assis, les fils peuvent s'inscrire dans la continuité de son cou ou de ses bras réduisants alors la gêne engendrée (les fils plongeant sous la table ce qui empêchera leur présence sur la table ou sur des objets).
2. Nous noterons toutefois que cette technologie souffre des problèmes d'occultation : pas d'obstacle entre l'objet suivi et le capteur et besoin d'un angle de vue suffisant entre les deux éléments. Nous avons donc choisi de compléter le suivi par une *technique magnétique* : elle n'est pas sensible à l'occultation, délivre un suivi à 6 DDL mais est moins précise que l'optique. Nous l'utiliserons donc pour des objets souvent occultés ou nécessitant une large couverture de l'espace 6D (beaucoup de rotations et translations «complexes»). Nous privilégierons alors son utilisation pour des éléments à suivre dans l'espace personnel et pour les interfaces de type pointeurs. En particulier, elle ne sera pas utilisée par exemple pour la tête, pour les mains et pour les mouvements contraints par les caractéristiques physiologiques. Dans le cadre d'une technologie avec fils, les éléments suivis se trouvant alors près du bord de la table, ceci réduira alors l'encombrement et la gêne procurée (avec la souris 2D standard).
3. Les deux techniques proposées permettent alors de suivre les utilisateurs, les objets occultés et objets d'expérimentations mais elles se limitent généralement à un faible nombre d'objets identifiables et ne fournissent pas de solutions pour les éléments

[3] Aujourd'hui il semble qu'aucune solution purement vision ne permettrait de suivre en temps réel les positions de la tête, des mains, des doigts et des objets manipulés pour plusieurs utilisateurs en gérant les problèmes de variation lumineuse, de précision, de bruit ou des occultations.

suivis dans l'espace partagé. Pour ces derniers nous ajoutons à notre système *un suivi optique par vision* pour l'identification et le suivi sans fils d'éléments sur la table (à faible précision).

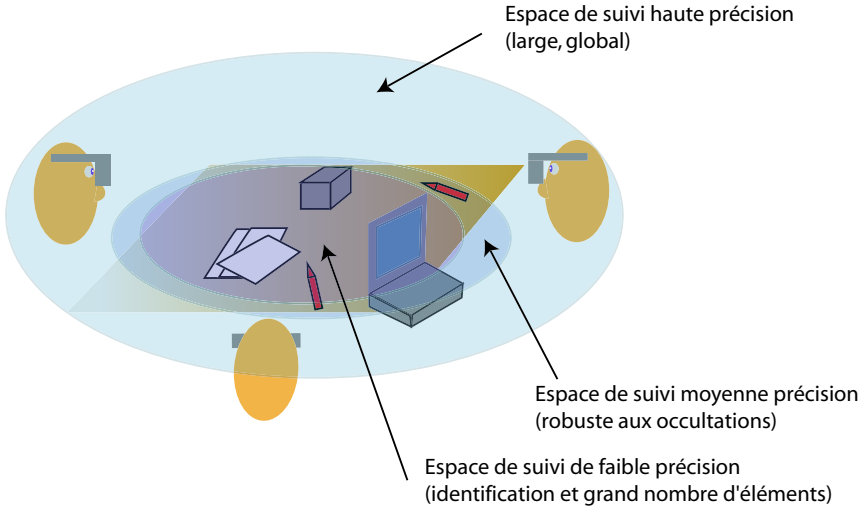


Fig. 4.9: Schéma résumé de notre approche pour le suivi.

2.6. Base interactive : orienté interfaces de réunion

Nous nous intéressons aux dispositifs physiques pour l'interaction avec les éléments présentés dans les espaces de visualisation des utilisateurs. Dans ce cadre, nous nous appuyons sur les dispositifs de suivi introduits section 2.5.

Besoins

Critères ergonomiques : D'après Berard [Ber99] nous pouvons retenir la latence, la résolution et la stabilité statique comme critères discriminants pour une interaction en RA (qui s'apparente à une interaction fortement couplée). Nous complétons ces critères par l'usage de dispositifs légers, maniables, simples et peu intrusifs avec des propriétés intuitivement perceptibles (*affordance* [Nor98]).

Critères collaboratifs : Les dispositifs devront s'adapter aux rôles des utilisateurs et supporter une interaction simultanée. Nous tiendrons compte aussi des besoins définis dans Tandler [Tan00] : des dispositifs utilisable simultanément, des dispositifs partagés, de multiples dispositifs par utilisateur et des dispositifs privés.

Critères applicatifs : les dispositifs devront supporter les besoins usuels d'applications 3D mais aussi 2D (dans un cadre collaboratif). les fonctionnalités devant être supportées sont plus précisément décrites au chapitre 6 (principalement la navigation, la manipulation et le contrôle d'applications).

Approche

Il n'exist pas aujourd'hui une interface idéale pour la manipulation de contenu (2D et 3D) dans un environnement de RA (ou de RV). L'approche que nous avons retenue est une approche hétérogène avec :
- un usage de plusieurs dispositifs 2D et 3D ;
- un usage de périphériques privés et publics ;
- une utilisation dans l'espace personnel ou public ;
- une complémentarité entre de nouveaux dispositifs et des périphériques standards ;

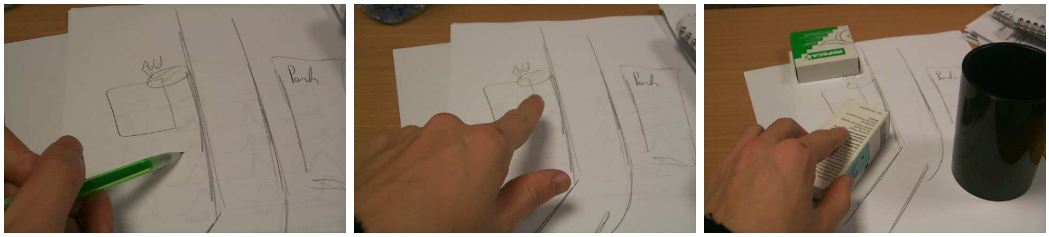
L'aspect temporel d'utilisation d'un périphérique est un critère essentiel pour le choix de l'interface. La gêne résultante de s'équiper de périphérique corporel (*wearable*) peut être négligeable si la main est dédiée à une tâche précise sur une longue durée (sans prendre

en compte les facteurs ergonomiques de ce type d'équipement). Par contre, dans le cadre de multiples tâches avec de multiples contextes, ce type d'interface ne sera pas forcement optimal à cause des hautes fréquences de changements d'outils et d'éléments manipulés. Dans notre cadre, l'intervalle typique d'une réunion est approximativement entre 15 minutes et 2 heures, pour un faible nombre de tâches différentes (dans un environnement de travail collaboratif). Nous ne négligerons donc pas la possibilité d'utilisation d'interface corporelle.

Nouveaux types d'interfaces

Nous nous sommes inspirés des métaphores et outils employés dans les réunions du monde réel pour définir trois types d'interfaces complémentaires. Nous distinguons dans une réunion réel (figure 4.10) :

- **l'interface corporelle**. Nous pouvons la décrire par l'utilisation du bras, de la main ou des doigts. Elle sert la communication par des gestes mimétique ou déictique (geste qui modélise un principe ou indique un objet). Elle a aussi un rôle ergotique (application d'une énergie qui modifie un état) pour la manipulation des éléments présents sur la table ;
- **le pointeur physique**. La limitation physiologique de la précision induite par la main et les doigts est alors comblée par l'utilisation d'un pointeur. Principalement utilisé dans des actions déictiques, pour indiquer des éléments à forte précision. Sous la forme d'un stylo, il peut alors aussi servir à la prise de note ou le tracé 2D ;
- **les artefacts physiques**. Nous considérons l'usage d'éléments réels intermédiaires pour la manipulation. Principalement basée sur l'utilisation des éléments contrôlés par la main, c'est l'action induite par ces éléments qui définit l'interaction, le résultat de la tâche. Dans les réunions réelles nous remarquons l'utilisation principal de maquette avec parfois des représentations simplifiées (par exemple une forme cubique pour représenter un bâtiment). Ces éléments peuvent être prédéfinis ou utilisés dynamiquement en se reposant sur la capacité créative du cerveau. Par exemple, on peut représenter une maison à l'aide d'une tasse à café ou la position d'un élément à l'aide d'une bouteille d'eau ;



(a) (b) (c)

Fig. 4.10: Trois types d'interfaces réelles utilisées dans une réunion : (a) pointeur (b) main et (c) artefact tangible.

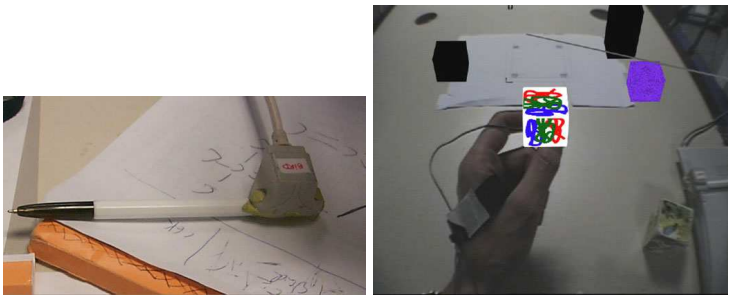
A partir de ces caractéristiques, nous proposons trois nouveaux types d'interfaces (on remarque que le type pointeur peut se ramener au concept tangible) en décrivant leurs principes et les besoins spécifiques pour la mise en œuvre. Ces interfaces reposent sur l'utilisation des capteurs de mouvements présentés précédemment.

Outil de pointage : le *Mixed Pen*

Principe : nous proposons un nouveau type de périphérique de pointeur mixte : un stylo muni d'un capteur magnétique à 6 DDL, nommé le *Mixed Pen*, avec un rôle d'interaction réel et d'interaction virtuel (figure 4.11a). Il constitue d'un élément physique pour la prise de note et les gestes déictiques. Pour le virtuel, il représente un outil virtuel pour le pointage ou l'interaction virtuelle à petit grain. Les intérêts de ce dispositif sont le multiplexage temporel pour le réel/virtuel, l'utlisation naturelle (*affordance*) de ce type d'éléments dans

les deux mondes et la conservation des métaphores de manipulation similaires aux réels pour les tâches nécessitants de la précision.

Besoins spécifiques : ils sont principalement d'ordre ergonomique. Nous privilégierons un suivi robuste du stylo, une prise en main aisée du pointeur (en saisie, fonctionnement, relâchement), un maximum de maniabilité de l'outil (suivi et contraintes physiques), de la souplesse et la légèreté du système. D'un point de vue collaboratif nous souhaitons la possibilité d'un suivi de multiples éléments avec identification de l'utilisateur de l'outil. D'un point de vue fonctionnel, l'outil doit supporter un pointage précis.



(a) (b)

Fig. 4.11: Différentes interfaces utilisées dans notre système : (a) le prototype de *Mixed Pen* avec le capteur magnétique et l'adjonction d'un bouton (b) le *Lightweight AR glove* durant la manipulation d'objets virtuels.

Outil d'interface gestuelle : le *Lightweight AR Glove*

Principe : nous souhaitons fournir à l'utilisateur une interaction plus souple principalement basée sur l'interaction digitale (avec les doigts). Nous voulons retrouver la facilité de manipulation d'objets réels, tout en évitant d'équiper l'utilisateur de système trop intrusif tels que des gants de RV. La solution proposée est d'équiper plusieurs doigts de capteur optique de position (figure 4.11b) un système peu intrusif, nommée le *Lightweight AR Glove*, qui permet à l'utilisateur de garder une grande liberté de mouvement, tout en pouvant effectuer des tâches réelles ou virtuelles.

L'approche a aussi consistée à une étude de type de mouvements et postures de la main fait par un utilisateur lors d'une réunion. Nous nous sommes basés sur une étude bibliographique des domaines spécialisés : physiologique ([WMSB98]), robotique([BK00]) et utilisation en images de synthèse.

Besoins spécifiques : les besoins pour ce type d'interface sont une transparence pour l'utilisateur dans sa gestuelle avec : un faible poids, une conservation naturelle des libertés de mouvement, une conservation de la facilité de manipulation d'éléments réels (maquette, papier, stylo), un support de suivi simultané de multiples éléments. D'un point de vue fonctionnel, c'est la possibilité de manipulation avec des éléments virtuels (sélection, déplacement) mais aussi de métaphore de contrôle d'application (choix d'outils, appels de menus).

Props* et artefact physique : *Tangible Augmented Reality

Principe : Nous souhaitons conserver la possibilité d'utiliser des éléments positionnés sur la table pour mimer des actions ou représenter des éléments virtuels 3D. Nous considérons de plus que la surface apportée par la table fournit un bon cadre de support d'interaction spatiale. Dans cet esprit, nous reprenons le concept de *Tangible Augmented Reality* (voir chapitre 6), que l'on couple au concept de *Real Reality* [BB96].

Choix des éléments : Fjeld [FV02] a étudié le bon choix d'interface tangible pour une interaction sur table et s'est limité au cadre d'une utilisation de brique. Son choix était fortement conditionné par son architecture matérielle, suivi 2D par vision. Nous nous

restreignons dans notre cas à tout objet de faible taille et poids, pouvant facilement être manipulé par un utilisateur (à un doigt, à une main, à deux mains).

Nous proposons l'utilisation de deux types d'éléments : naturels et spécifiques.

Le premier type d'éléments est formé d'objets réels naturels présents à portée de main et utilisés dans le cadre de réunion : tasse, cahier, livre, gomme, boîte, etc. Nous considérons alors une assignation dynamique de fonctionnalité à ce type d'élément de façon interactif (contenu, outils, propriétés). Dans ce cadre, l'utilisateur peut augmenter les mimiques qu'il utilise parfois en illustrant une idée par l'utilisation d'objets virtuels simples de son environnement (la forme ou le déplacement restant les critères importants). Par exemple, l'utilisateur peut coupler en cours de session sa tasse réelle avec un bâtiment virtuel pour montrer son placement final sur une maquette virtuelle (exemple proposé par Broll [BMS00]).

Le deuxième type d'éléments considéré est la classe des éléments prédéfinis pour le système et dédiés à ce type d'interaction (offrant alors une connaissance et une gestion aisée par le système). Ils sont alors identifiables grâce à une spécificité visuelle. La fonctionnalité de ces éléments peut alors être prédéfinie (coder en dur), ou facilement assignée en cours de session. Dans notre thèse, nous considérons leur usage pour la manipulation de contenu virtuel ou la définition d'outils spécifiques (rotation, translation).

Comme les interfaces précédentes, ces éléments doivent être couplés à des capteurs pour que le système suive leur mouvement et qu'ils puissent être utilisable. Leurs espace d'utilisation est alors défini par l'espace de la table. On les emplois comme interface absolu isotonique (voir définition chapitre 6).

Utilisation : nous nous basons sur le concept de *Real Reality* qui consiste à suivre le mouvement d'objets réels à partir du mouvement d'une interface digitale couplée avec cet élément (une transformation rigide constante dans le temps durant le déplacement). La contrainte est la connaissance par le système de la position initiale de chaque objet.

Introduit par Bruns [BB96], sa démonstration reposait sur l'utilisation d'un gant de donnée de type *Cyberglove* équipé d'un capteur magnétique sur la main. il détectait la saisie (*grasping*) d'un objet par détection de collision entre l'index et l'objet, associée avec la reconnaissance d'un geste de saisie (le relâchement est aussi détecté par la fin du geste de saisie). Pour des raisons d'efficacité[4] il a ensuite proposé [EKB99] de remplacer le système initial par des capteurs au bout des doigts et un capteur de pression permettant de détecter le contact avec l'objet réel (démontré sur une visualisation sur écran mais aussi en RA avec un système projectif 2D d'informations sur des briques réelles).

(a) poussée avec un doigt (b) saisie par dessus (pinch grip) (c) saisie sur le côté (lateral grip)

Fig. 4.12: Les trois types de mouvements retenus pour des déplacements sur table.

Nous avons adapté ce concept à la RA 3D et à l'interface manuelle présentée précédemment. Notre système permet une véritable utilisation en 3D avec un système plus léger

[4] gant imprécis, calibrage initial nécessaire

et orienté pour des déplacements des objets sur la table que la solution de Bruns. Nous distinguons quatre phases de contrôle : détection de saisie, initialisation d'une action, action (e.g. déplacement) et détection de relâchement. Nous définissons trois classes de saisie basées sur les travaux de Mas [MT94] associées à la détection de collisions des points d'interactions de l'interface manuelle (figure 4.12). La sélection de l'élément (après la détection de saisie) et le relâchement sont effectués sous forme de geste explicite (tels que lever et descendre le doigt) ou de temps de pause (plus robustesse mais moins efficace).

Interface standard

Pour les interfaces standards nous avons privilégié des interfaces 2D privées de pointage et de support alphanumériques. Nous utilisons dans ce cadre des souris, claviers et tablettes numériques. Elles pourront être utilisés dans le même cadre que les travaux en *SDG*. La nouveauté tenant ici au fait de travailler dans un espace de visualisation 3D et non plus une surface 2D, ce choix offrant de nouvelles caractéristiques à intégrer (gérer de multiples pointeurs dans 3D dans une espace 3D).

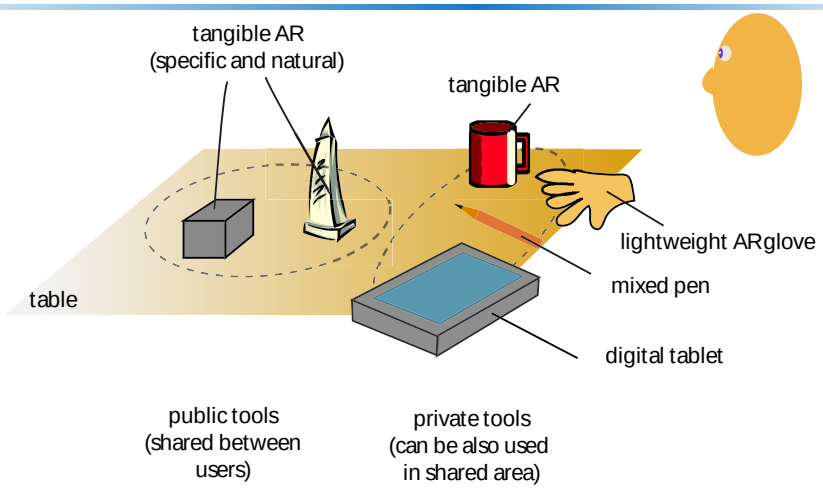


Fig. 4.13: Schéma résumé de notre approche pour l'interaction.

2.7. Retour et enregistrement sonore

Nous nous sommes également intéressés au support de deux autres canaux sensoriels permettant de fournir une interaction et un retour supplémentaire avec l'environnement : les canaux auditifs et oraux.

Besoins

Facteurs applicatifs : nous souhaitons pouvoir fournir un retour sonore à l'utilisateur lors de l'interaction dans notre environnement. Nous considérons que le retour sonore peut :

- fournir une restitution d'ambiance, diffuse autour de l'environnement ;
- fournir un retour auditif sur l'état du système, ou sur le déroulement d'une action (saisie, déplacement d'objet, etc.) ;
- autoriser une représentation phonique d'un objet virtuel (bruit d'une voiture, bruit d'un instrument, etc.) ;

La prise de son peut servir à des systèmes de reconnaissance vocale (pour interagir avec le système tel que dans [Bol80]) mais aussi à l'enregistrement d'annotations sonores.

Facteurs collaboratifs : le système doit pouvoir permettre la restitution sonore pour les différents utilisateurs, mais aussi pour un utilisateur particulier (robuste aux bruits). Le système devra pouvoir aussi supporter l'enregistrement des allocutions ou les ordres vocaux individuels (supportant une interaction simultanée), mais aussi publics lors de phase coopérative.

Facteurs ergonomiques : le système devra garantir une diffusion personnelle non intrusive lors de la restitution des données sonores. Le système devra aussi fournir une restitution spatiale (directionnelle et position) pour la diffusion sonore d'éléments définis dans un repère réel.

Approche

Nous avons choisi d'introduire trois espaces sonores de diffusions (figure 4.14). Le premier est un espace sonore public constitué d'une *restitution sonore spatialisée* dans l'espace de la table. La restitution spatialisée définit la restitution d'un son virtuel en fonction de sa position et la position de l'utilisateur, telle que définit dans l'environnement réel (respectant alors les principes physiques tels que l'effet *Doppler*, la réverbation, les occultations, etc. Voir [Tsi98] pour plus d'informations).

Le deuxième est la restitution d'un *espace d'ambiance* ou de messages publics (musique, voix, «bips systèmes») à distance lointaine (positionnement indéfini, direction définie).

La restitution d'un *espace sonore privé* autour de l'utilisateur constitue le dernier champ fournissant des informations personnelles à l'utilisateur : informations systèmes, messages collaborateurs distants, etc.

Pour l'enregistrement vocal, nous notons que le cadre de réunion limite l'efficacité de ce type de système contraignant les autres utilisateurs à diminuer l'intensité de leurs discours voir de se taire (coordination envisageable avec un protocole social). Nous retiendrons donc une approche par micros directionnels principalement dédié à des enregistrements réalisés non simultanément par les utilisateurs. Un microphone d'ambiance pourra servir en cas de tâches fortement collaboratives à un enregistrement public.

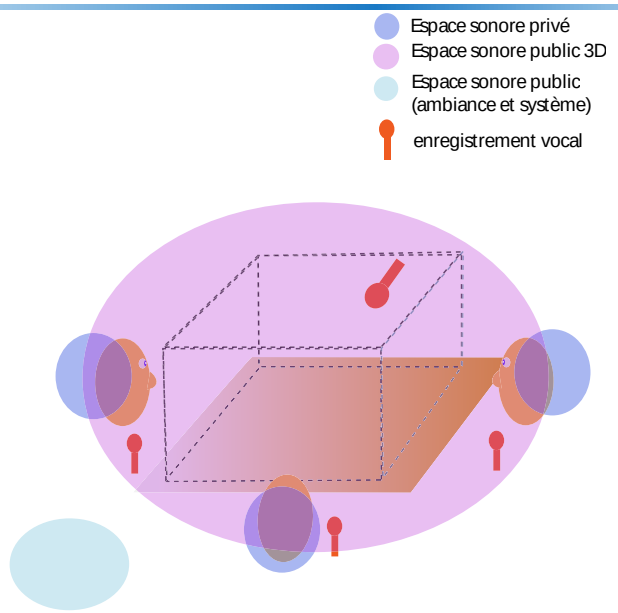


Fig. 4.14: Schéma résumé de notre approche pour le son.

2.8. Support de calcul : architecture centralisée

Besoins

Chacun des composants proposés précédemment doit être relié à un serveur de calcul servant aussi au déroulement de l'application. Dans notre cadre orienté vers des tâches 3D, nous définissons un besoin de haute performance : carte graphique de haute qualité, multiprocesseurs (nombreuses tâches à supporter et à paralléliser), grand espace mémoire (contenu 3D de grande taille de l'ordre de 100 Mo). La communication entre éléments devra être robuste pour réduire des temps de latence perceptibles (inférieur à 50 ms).

Le facteur principale restant une fréquence d'affichage et d'interaction élevée telle que nécessaire en RV (25 Hz minimum).

Approche : centralisé ou faiblement distribué

Nous nous sommes principalement orientés vers une *configuration de type centralisé* reposant sur une machine très puissante (Onyx 3, 8 processeurs, 2 cartes graphiques haut de gamme). Ceci limite les problèmes telle que la synchronisation ou le parallélisme de tâches, et permet de se concentrer principalement vers les tâches applicatives et les besoins des utilisateurs. Toutefois cette approche a trouvé ses limites durant notre étude (limitation du nombres d'entrées matérielles, fonctionnalités logicielles limitées) et a conduit à étendre notre approche vers une solution distribuée hétérogène. Cette distribution consistant alors plus une adjonction de modules qui communiquent avec l'élément central, qu'un système de réplication de l'application (approche client/serveur ou maître/esclave).

2.9. Mise en œuvre et résultats

Nous présentons ici les résultats de la réalisation des six éléments introduits précédemment. La figure 4.15 présente une vue générale de l'architecture mise en œuvre. Nous décrivons brièvement les différents résultats de chaque partie : une description beaucoup plus détaillée, comprenant simulation et mesures, se trouve annexe A.

Espace de communication (figure 4.16) : la table fournit un bon support pour une interaction collaborative côte-à-côte ou face-à-face pour deux utilisateurs et limitée pour trois utilisateurs (espace intime en intersection). La table répond parfaitement aux critères de «non nuisance» des systèmes de suivi (vision et magnétique).

Visualisation (figure 4.17, figure 4.18) : le mixage des solutions casque vidéo et casque optique fournit un bon élément de visualisation, l'évaluation montre une bonne gestion de différents types de données. Nous regrettons la faible qualité des casques dont nous avons disposé : par leur résolution et leur champs de vue limitant leur utilisation dans un cadre professionnel (avec en plus des problèmes de luminance pour l'optique et de distorsion d'image pour la vidéo). La vue publique projetée fournit un bon composant complémentaire (grande taille), mais les documents textuels restent difficilement lisibles. Enfin, la vue RA avec le portable offre une approche intéressante (bonne visibilité, bonne qualité) mais trop limitée par le recul nécessaire pour la visualisation de contenu au centre de la table et le champ de vue réduit de la caméra utilisée.

Suivi (figure 4.19) : le capteur *Optotrak 3020* de *Northern Digital* nous offre une très bonne précisionm, une excellente stabilité, et permet un suivi de quatre utilisateurs en temps réel (l'évaluation du suivi de la tête montre de très bons résultats dans le positionnement du capteur et des leds posés sur le casque). Le capteur magnétique ne souffre pas de distorsion magnétique et le positionnement choisi garantit une utilisation dans de bonnes conditions. Enfin le suivi par vision (*ARToolkit*) fonctionnant à 10 Hz offre un temps de latence acceptable (client distribué), avec des marqueurs d'une taille de 8 cm et supportant un grand nombre de marqueurs reconnaissables.

Interaction (figure 4.20) : l'hétérogénéité de la configuration montre des résultats satisfaisant, et nous fournit un panel intéressant pour la mise en œuvre de métaphores d'interaction. La tablette digitale offre une bonne résolution mais la simulation et l'expérience montre une trop large place prise par ce support dans l'espace personnel d'un utilisateur.

Nous décrivons rapidement nos conclusions sur les nouvelles interfaces proposés et réalisés :
- Le *Mixed pen* utilisé avec un capteur magnétique permet la mise en œuvre aisée des approches *virtual hand*[BLMP00] de RV, mais le poids du capteur et de son fil est un défaut dans un cadre de RA sur table ;
- Le *Lightweight AR glove* offre une interface de grande précision et très stable ;

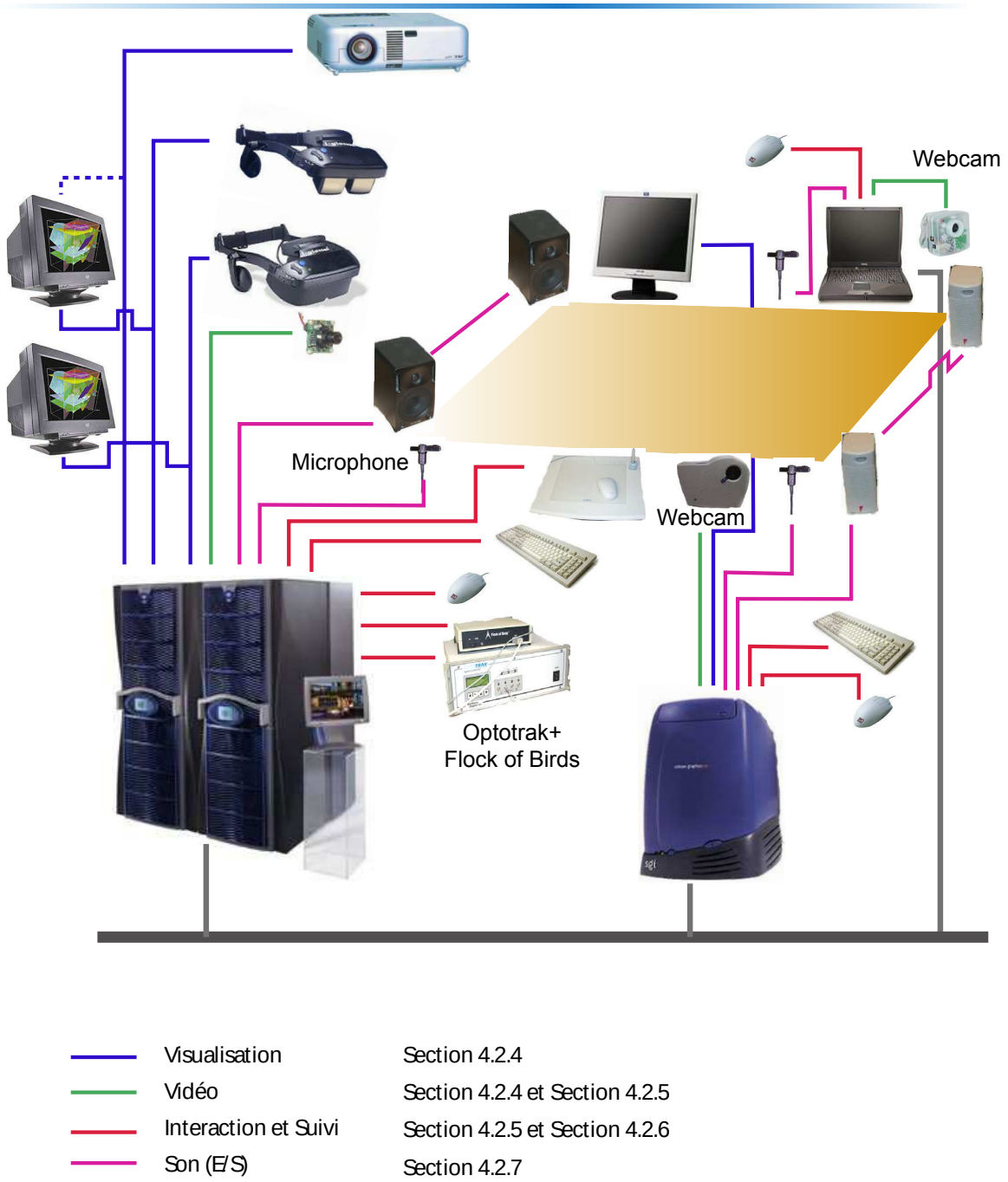
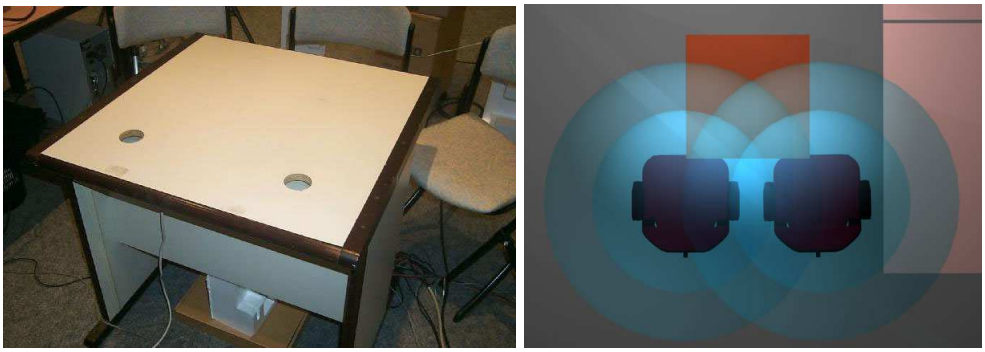


Fig. 4.15: Schéma global de l'architecture matérielle avec les différents périphériques et les connections.

– L'utilisation du système de suivi d'objets est une interface très naturelle, répondant à nos critères mais toutefois limité par la librairie de suivi. Le concept de Real Reality offrent de bons résultats. mais nécessitent une évaluation plus précise.

La complémentarité entre les trois éléments répond parfaitement aux besoins définis, une étude plus précise de l'utilisabilité et une réalisation de meilleure qualité est nécessaire pour un véritable usage de ces interfaces (chapitre 6).

Son : le système présenté est quadraphonique reposant sur la synchronisation entre deux paires d'enceintes sur deux machines différentes (réalisation ad-hoc). La qualité sonore est satisfaisante pour le retour d'ambiance ou privée. Le retour spatial est de faible qualité d'un point de vue de la synchronisation entre les deux machines (deux moteurs sonores spatiaux *OpenAL*[Ope]). Les enregistrement vocaux sont aussi satisfaisants pour la prise



(a) (b)

Fig. 4.16: Table utilisée pour nos expériences : (a) vue de la table (b) résultat de la simulation d'espaces de communication pour deux utilisateurs.

(a) (b) (c)

Fig. 4.17: Les solutions de visualisation privée : (a) le casque optique semi-transparent *I-Glasses* (b) le casque vidéo semi-transparent *I-Glasses* (c) le portable semi-transparent.

(a) (b)

Fig. 4.18: Les solutions de visualisation publique : (a) une vue publique statique avec écran plat (b) une vue publique par projection pour une réunion.

d'annotation sonore. Par contre, les essais de reconnaissance vocale restent non satisfaisant avec un environnement bruité ou du matériel de faible qualité.

Gestion du calcul : nous nous reposons principalement sur une *SGI Onyx 3400* avec 8 processeurs, 8 Go de mémoire vive et 2 cartes graphiques. En complément, nous utilisons une *SGI O2* pour le suivi par vision et la gestion du son (E/S), et un portable *DELL Inspirion 8000*. pour la gestion de la reconnaissance vocale. L'Onyx nous offre sans difficulté toutes les ressources nécessaires, même si l'affichage de scène complexe reste très coûteux.

(a) (b) (c)

Fig. 4.19: Le suivi hétérogène : (a) capteur *Optotrak* positionné pour un suivi grand champ (b) le capteur magnétique *Flock of Birds* sous la table (c) la caméra placée pour le suivi par vision.

(a) (b)

(c)

Fig. 4.20: Différentes interfaces utilisées dans notre système : (a) interfaces tangibles : approche naturelle et (b) avec des marqueurs *ARToolkit* (c) la tablette *Wacom Intuos* dans l'espace personnel d'un utilisateur.

2.10. Synthèse

Nous avons proposé un concept d'architecture matérielle hétérogène générique en introduisant une large complémentarité de dispositifs adaptés à différentes tâches. Notre concept d'architecture couvre un large panel de besoins dans le cadre de tâches 3D/2D collaboratives sur table, et la réalisation nous fournit une plate-forme expérimental et applicative. L'originalité de ce travail se trouve aussi dans une réalisation complète basé sur une analyse détaillé des éléments d'un environnement de RA 3D sur table. L'approche retenu (modélisation spatiale, étude des besoins fonctionnels/non fonctionnels, éléments de conception)

offre une démarche reproductible simple dans la conception de ce type d'environnement (très peu réalisé à ce jour en RA 3D).

Enfin, nous avons proposé deux nouveaux dispositifs originaux (le *mixed pen* et le *lightweight AR glove*) qui offrent des possibilités d'interaction intéressantes que nous verrons plus en détail au chapitre 6.

Nous comparons brièvement notre système aux approches colocalisées sur table similaires. Notre système offre un support hétérogène dans le même esprit que *MagicMeeting* et *EMMIE* mais avec un plus grand nombre d'éléments complémentaires (son, visualisation, interaction) et avec la définition et une analyse des besoins (support ergonomique, collaboratif et applicatif). Nous fournissons un suivi plus robuste et plus utilisable que *MagicMeeting* qui repose uniquement sur une approche *inside-out*. Nous introduisons un support d'interface très similaire aux récents travaux d'*ARTHUR* mais avec une approche différente et de meilleures performances malgré notre solution intrusive : l'approche par vision proposée dans leur cadre contraint l'environnement (fond uni), se limite au suivi d'un doigt et d'une baguette magique, avec des interfaces tangibles fonctionnant à faible fréquence (comme on peut le voir sur leurs vidéos).

L'évaluation des différentes composantes a montré de nombreuses lacunes à notre mise en œuvre. De notre point de vue, les éléments à améliorer seraient :
- support communicatif : une table légèrement plus grande avec un pied central, offrant un espace privé plus grand et de meilleurs conditions ergonomiques ;
- support de visualisation : des casques de meilleure qualité optique, et un support de projection sur table (offrant par exemple un mixage 2D/3D sur le contenu présent sur la table ou une vision périphérique hybride telle que dans [BG02]). Nous pourrions compléter le support de visualisation de documents 2D avec des *tabletPC* offrant une interface plus légère qu'un portable (mais posant des difficultés pour le mode semi-transparent) ;
- suivi : un système sans fils avec les mêmes caractéristiques que l'Optotrak semble intéressant mais non disponible actuellement ;
- interaction : une utilisation d'autres types de périphériques de bureau (*paddle*, *spacemouse*, *tabletPC*) offrirait de nouvelles voies d'approches. Pour l'interaction sur les éléments de la surface murale une approche par pointeur laser peut être une solution ;
- son : un système sonore de type *Dolby Digital 5.1*, *ADAT* ou un chaînage de hauts-parleurs USB, fournirait une meilleure restitution spatiale. Une étude complémentaire sur la reconnaissance vocale avec des micros portables mais de haute qualités doit aussi être envisagés ;
- haptique : nous nous sommes limités à une approche passive, une étude des systèmes actifs reste à effectuer. Mais aucun à ce jour ne semble compatible avec notre type d'environnement collaboratif.

3. Architecture logicielle

Basée sur cette plate-forme matérielle, nous nous concentrons maintenant sur la composante logicielle pour notre environnement collaboratif en RA 3D. Nous nous intéressons aux solutions logicielles utilisables pour la réalisation d'applications reposant sur notre architecture matérielle générique. Nous présenterons l'approche retenue après une introduction des concepts étudiés et l'analyse des solutions actuelles.

3.1. Introduction

La réalisation d'une application peut être définie à plusieurs niveaux suivant les ressources utilisées et dépend du contexte de réalisation. Nous distinguons plusieurs modes de programmation d'une application [Gol98] :

– développement total depuis zéro : le developpeur implémente toutes les fonctionnalités de son application en s'appuyant sur les couches les plus basses possibles. Il utilise l'interface de programmation (*API, Application Programming Interface*) du système ou des appels directs aux composants matériels ;
– utilisation de **bibliothèques** (*libraries*) : le développeur s'appuie sur un certain nombre de bibliothèques spécifiques qui fournissent des services dédiés : graphisme, son, réseau, etc. Le programme consiste alors à l'assemblage, l'appel de ces fonctionnalités et à compléter les services non fournis ;
– utilisation de **boîtes à outils** (*toolkit*) : l'utilisation d'une boîte à outils fournit généralement un modèle de conception et un ensemble des fonctionnalités complémentaires. Le développeur peut alors s'appuyer sur ses fonctionnalités pour se concentrer sur les parties applicatives du programme ;
– utilisation d'un **canevas logiciel** (*framework*) : l'utilisateur dispose d'un squelette d'applications et complète alors un certain nombre d'éléments. Le canevas logiciel est une architecture de plus haut niveau que la boîte à outils car il contrôle le déroulement du programme. L'utilisateur n'a alors qu'à «remplir les trous» de l'interface fournie ou à configurer simplement l'application par description ;
– utilisation d'une application existante : l'utilisateur dispose des sources d'une application similaire à la tâche à réaliser et la modifie.

Notre cadre de travail très versatile nécessite une large souplesse dans la mise en œuvre de solutions variées (grand nombre de programmes d'expérimentation), complexes (prototype applicatif) et expérimentales (mesures de performances, test utilisateurs). Il nous semble qu'un support par canevas logiciel ou par boîte à outils doit être utilisé offrant une plate-forme logicielle générique et souple. Ces solutions permettent la simplification du développement, la ré-utilisation de code et fournissent un environnement rapide de développement d'application.

Nous étudierons donc les solutions envisageables d'architecture pour la réalisation de collecticiels 3D en RA. Nous présentons au préalable les besoins pour ce cadre.

3.2. Définitions et besoins

Nous décrivons maintenant les besoins d'une architecture logicielle générique. Dans notre thèse les besoins couvriront les fonctionnalités nécessaires dans un cadre de RA sur table, mais seront aussi étendu à la RV et à l'informatique évanescente. Ces besoins ont été principalement extrait d'un grand nombre de plate-formes logicielles de RA/RV. Nous distinguerons les besoins non-fonctionnels (standard à tout type d'architecture) et les besoins fonctionnels (pour notre domaine). Les besoins peuvent être défini à l'exécution ou pour le développement.

Besoins non-fonctionnels standards :
– **accessibilité** : facilité d'utilisation ou de développement avec le système ;
– **flexibilité** : possibilité d'une grande variation des caractéristiques d'application (ressources, contenu, etc.) ;
– **performance** : offrir les meilleurs performances en terme de coût des ressources et des besoins applicatifs ;
– **extensibilité** : possibilité d'étendre les fonctionnalités de la plate forme (d'un objet, d'une classe, de l'application ou du comportement de l'application) ;
– **configurabilité** : possibilité de configurer aisément les fonctionnalités de l'application sans avoir à modifier le programme ;
– **dynamicité** : possibilité d'effectuer des opérations à l'exécution, telles que le chargement de données, de modules, de services, de contenu, de mise en relation de ces éléments ou de reconfiguration ;

- **passage à l'échelle** : supporter le changement quantitatif de plate-forme matérielles utilisées ;
- **adaptatif** : fournir des mécanismes et des choix d'adaptations de la solution au contexte ;
- **persistance** : possibilité de sauvegarder l'état de l'application et de ses données ;
- **sécurité et protection** : supporter l'identification et le contrôle des ressources utilisées ;
- **tolérance aux fautes** : supporter les erreurs systèmes ou applicatives ;
- **maintenance** : facilité de mise à jour, d'optimisation et de corrections sur le programme ;
- **support multi plate-forme** : avoir une architecture executable et fonctionnelle sur plusieurs plate-formes (processeur et système d'exploitation), transparente et portable ;
- **support du modèle d'exécution** : supporter de multiples modèle de fonctionnement en mémoire de l'application (*single-threaded, multi-thread, multi-process*), ou de distribution (non distribué, répliqué, réparti) ;
- **modularité** : offrir une décomposition en unités logiques, promulgant souplesse de programmation et de combinaison des fonctionnalités (au contraire des solutions monolithiques) ;
- **introspection** : offrir la possibilité de connaître les services de façon dynamique ;
- **interface de programmation** : supporter de multiples interfaces de programmation, une adaptation à l'utilisateur et la possibilité de prototypages rapides ;
- **outils de programmation** : disposer d'outils de mise en œuvre, d'expérimentations, d'évaluations, de configurations (avant/durant et après le développement). A l'extrême nous pouvons avoir un environnement intégré de réalisation (*IDE, Integrated Development Environment*).

Besoins (fonctionnels et non-fonctionnels) spécifiques de notre domaine :

- **périphériques RV/RA** : gestion des multiples périphériques aussi bien en entrée (suivi, interaction 3D, vidéo) qu'en sortie (affichage, son) ;
- **contenu multimédia** : gestion de représentation, manipulation du contenu classique : modèle 3D, images, son, vidéo, texte. Il faut aussi fournir le moyen de définir (représentation et comportement), «instancier», charger, représenter, modifier, faire évoluer ces objets ;
- **représentation graphique avancée** : gestion du réalisme visuelle : effets visuels, animations, apparence des objets, éclairage, etc. ;
- gestion des **tâches de base** : nous distinguons les ainsi les besoins en algorithmique (structures de contrôle de haut niveau, graphe d'état), en mathématique (structures de géométrie algébrique) ou en physique (détection de collisions) ;
- **de mécanismes d'interactions 3D et 2D** : gestion de mécanisme et méthodes d'interactions avec le contenu et les applications et connections avec les retours sensitifs à l'aide des périphériques disponibles. Nous pouvons citer la navigation et la manipulation 3D ;
- **collaboratif** : gestion des besoins d'identifications et de relations avec les autres utilisateurs (retours perceptuels, coordination, communications).

3.3. Etat de l'art

On étudie maintenant les solutions possibles en présentant les bibliothèques (section 3.3.1) et canevas logiciels (section 3.3.2 et 3.3.3) disponibles ainsi que les outils auteurs (section 3.3.4), les librairies spécialisées (section 3.3.5) et les outils de description de contenu pour la RA (section 3.3.6).

3.3.1. Bibliothèques et boîte à outils

MR-Platform[UTS+02] est une plate-forme logicielle en C++ pour la création d'applications de RM. Ce système commercial, fourni avec des casques vidéos semi-transparents contient différents modules bas niveau : capture vidéo, gestion des capteurs 6 DDL, traitement d'images, affichages et procédures de calibrage (complété par un outil graphique pour ce dernier). La bibliothèque ne fournit pas de mécanismes explicites de gestion de l'aspect collaboratif.

ImageTclAR[OTX03] est une bibliothèque qui fournit un environnement de développement à partir d'une interface essentiellement sous forme de langage script. Dédié à un but éducatif, elle fournit une plate-forme modulaire pour le support d'interface graphique, de traitement d'images, de gestion de contenu RA (capteurs, module d'importation 3D, calibrage, affichage). Elle offre une forte flexibilité, modularité et plusieurs interfaces de programmation (C++, Tcl, Outil graphique). L'aspect multi-utilisateurs n'est toutefois pas accessible.

MXToolkit [DSB+03] du projet MixDesign peut être défini comme une extension d'*ARToolkit*, introduisant la gestion du contenu et de l'interaction avec des objets virtuels associés à des marqueurs réels. L'architecture est modulaire, centralisé et propose comme services : gestion de base de données, capture vidéo, suivi de marqueurs, gestion vidéo et rendu 3D. Cette boîte à outils propose en complément des outils auteurs pour l'association d'objets virtuels et d'objets réels et la création de scènes.

3.3.2. Canevas logiciels objets

GRASP [ABC+94] a été une des premières architectures logicielles offrant un support de gestion de RA. Ces fonctionnalités reposant sur des mécanismes C++ qui se classent en gestion : de la géométrie, des objet de la scène, du rendu, des technique de calibrage, des E/S eet d'une interface utilisateur avec une gestion d'événements.

Le système *Tinmith-evo5* [PT03] propose une architecture dédiée aux applications mobiles. A base d'objets, il offre un mécanisme de flots de données depuis la mesure des capteurs jusqu'à l'affichage graphique 3D. Il repose sur une architecture en couche : une interface système, une couche de transformation, une couche de rendu (graphe de scène propriétaire), une couche d'interface utilisateur et une couche de définition d'application. Spécifiquement conçu pour un cadre mobile, son utilisation semble peu efficace en intérieur (faible performance, pas de gestion multi-utilisateurs).

Schmalstieg propose un environnement de RA nommé *Studierstube*. [SFH+02] reposant sur des mécanismes objets (C++) et défini comme une sur-couche d'*Open Inventor*. Le système intègre un noyau d'exécution et propose une approche modulaire dynamique, à partir de chargement de segments partagés à l'exécution. Leur approche est multi plate-formes, flexible, fortement configurable, offre de multiples interfaces de programmations (C++, python et déclarative avec le format *Open Inventor*). Au niveau fonctionnelle, elle offre une large gestion de périphériques d'interactions, de visualisation ou de modules haut niveau (multi-applications, multi-utilisateurs). Les solutions proposées ont été validées dans un grand nombres d'applications.

MacIntyre propose un canevas nommé *COTERIE (Columbia Object-oriented Testbed for Exploratory Research in Interacive Environments)* [MF96], pour la réalisation d'environnements virtuels distribués gérant en complément les applications RA. Cette architecture repose sur un concept de mémoire partagée et d'appel en retour par objets, fournissant à l'utilisateur une transparence lors de la réalisation de nouveaux prototypes. L'architecture a été démontré dans plusieurs projets. Le canevas repose sur des mécanismes objets en Modula-3 et utilise une librairie graphique propriétaire *Repo-3D*.

3.3.3. Canevas logiciel à composants

Le projet *DWARF (Design Wearable Augmented Reality Framework)* [BBK+01] introduit un canevas logiciel très performant pour la conception d'applications de RA (reposant sur une analyse descendante des besoins logiciels). S'appuyant sur un mécanismes d'objets distribués CORBA, il permet la génération, la configuration et l'exécution dynamique de composants logiciels spécifiques à notre domaine. Ils introduisent différents composants élémentaires : système de suivi, modèle de la scène, moteur de flot de tâches, service contextuel et interface multimodale. La validité de leur approche a été démontrée dans le projet *Sheep*. Ils ont récemment introduit un support d'outils de développement dans ce cadre sous forme d'outils graphiques de simulation et de monitoring à l'exécution d'une application.

Le projet *AMIRE* [DGHP02] [PHP+02]propose un canevas logiciel décomposé en trois couches : collections de *gems* de RA (équivalent aux *graphics gems*), collection de composants et couche unificatrice. La communication est assuré par un principes de *slot/signal* fortement typés et configurable à partir d'une structure hiérarchique d'attributs (*properties*). Réalisé dans le cadre d'un projet européen, différentes applications ont vu le jour dans un cadre mono-utilisateur.

VHD++ [PPM+03] introduit une architecture à composants dédiée à des applications de RV distribués supportant la présence d'agents virtuels. Ce système repose sur des moteurs d'éxecutions présent sur chaque machine et configurable par des propriétés au niveau applicatif. Un grand nombre de modules est proposé comprenant aussi des éléments utiles à la de RA : vidéo, suivi par vision, mixage visuel réel/virtuel. Elle semble toutefois limitée dans le développement d'application de RA complexe.

le projet *Magic Meeting* [RWB02] propose une solution par composants inspirés de *CCM* [Moda]. Basé sur des outils par description XML, le système fournit la définition, l'assemblage et le déploiement de solutions par composants (assemblage dynamique et exécution réalisé par des contrôleurs). Il introduit alors différentes classes de composants : périphérique, interaction, adaptateur, décorateur, aire et gestionnaire d'aire. La communication est essentiellement démontrés par communication asynchrone par événements.

Ces approches par composants semblent les plus flexibles et performantes. On note toutefois que *DWARF* bien qu'offrant des mécanismes haut niveau applicatif, démontre encore de faible résultats d'un point de vue des performances graphiques 3D (visualisation et interaction) et peu de services dédiés à des tâches 3D. Une intégration récente avec *Studierstube* tend vers l'ajustement ces défauts [BHM+03]. On note aussi que Magic Meeting n'est pas disponible gratuitement et que *AMIRE* se limite à la plate-forme Windows.

3.3.4. Outils auteurs

DART (The Designer' s Augmented Reality Toolkit) [MGB+03] est dédié à la création rapide d'expérience de RA. Basé sur *Macromedia Director* et ses spécificités (partition, script, comportement) il permet la création par interface visuelle d'une application de RA. Les auteurs introduisent en complément une gestion de modules de bases en RA : capture vidéo, suivi, alignement sur marqueurs, objets 3D. On notera que cette solution est peu flexible pour le support d'application complexe ou temps réel.

3.3.5. Librairies spécialisés

ARToolkit [BKWF99] offre un suivi à partir de marqueurs réels et l'intégration visuelle d'objets 3D sur ces marqueurs. Reposant sur GLUT, ARToolkit offre une accessibilité bas niveau sous forme de boîte à outils étendu. On peut rajouter qu'une extension permet l'importation de modèle VRML. C'est sans contestation possible la boîte à outils la plus utilisée pour la RA

OpenTracker [SR01] est une librairie développée dans le cadre de *Studierstube*, offrant une couche d'abstraction indépendante pour la gestion des périphériques (distribués ou non). Utilisable aussi dans un cadre de RV, ce canevas logiciel fournit un support pour la description sous forme de flots de données de différents types de données d'entrées : position 3D, position 2D, vocale. On retiendra particulièrement le mécanisme descriptif à base de graphes (XML) offre des fonctionnalités avancées de traitement de ces données : transformation, filtrages, opérations de fusions, transmission ou importation de données distantes.

Luk propose un environnement pour la reconfiguration d'architecture pour la RA [LLR+99]. Les fonctionnalités principales permettent un mixage temps réel de diverses source, et une résolution par masque des occultations.L'originalité de cette approche est la réalisation d'un système hybride matériel et logiciel.

3.3.6. Gestion de contenu 3D

D'un point de vue applicatif le contenu (modèles 3D, images, sons ou vidéos) peut être décrit sous différents formes : soupe de polygones, graphe de scène par des modules de haut niveau tels que dans [NM98], [FPTP01], [HPF99], [FL03].

Différentes solutions ont étés introduites pour la gestion de contenu applicatif de RA. Initialement Ulrich Neumann [NM98] a proposé le concept d'*ARML (Augmented Reality Modelling Language)* unifiant le contenu multimédia en RA avec une adaptation du format en fonction du support. Il définit l'usage d'outils auteurs dans ce cadre (création de contenu, informations géométriques de l'environnement), la génération automatique de support pour l'exécution (annotations virtuelles, marqueurs, relations) et des outils d'éxecution.

APRIL (Augmented Presentation and Interaction Authoring Language) [FL03] est un langage développé à l'IMS Tu-Wien dans le cadre du projet *Virtual Showcases* [BES03], dédié à la présentation de contenu en RA, tel que dans le domaine du *storytelling*. Au delà de la description du contenu de RA (adapté et transformé à la plate-forme logicielle), il propose une description sous forme de graphe d'état d'une présentation (depuis UML), de comportements d'objets inspirés de *SMIL* [W3Cb] ou de support d'interactions par événements réactifs pour l'évolution de l'application.

Récemment dans le cadre du projet *MARS* [HFT+99] et de la définition d'un outil auteur dédié, Güven [GF03] introduit le langage de description *CMIL++ (Contextual Media Integration Language)* pour la représentation de contenu hypermédia de RA (vidéo, texte, modèle 3D). Il fournit des mécanismes supportant le positionnement spatial (dans un repère écran ou monde) des éléments (réunis dans des *snippet*), des effets de transition (apparition, disparition, lien). L'outil auteur proposé s'inspire d'outils tels que *Macromedia Director* restituant une construction visuelle et interactive du scénario applicatif.

En résumé, ces différents approches de gestion de contenu favorisent toutes une définition déclarative basée sur des langages tel que XML, offrant le contenu, son évolution spatiale et temporelle, et des outils auteurs dédiés.

Après analyse des propriétés de chacune des solutions présentées, aucune ne semble répondre à nos attentes : haute performance, gestion multi-utilisateurs et surtout adapté à notre architecture matérielle. Nous introduisons donc un nouveau canevas logiciel de RA 3D présenté dans la suite de ce chapitre.

3.4. Notre approche : modularité et hétérogénéité

Dans cette section, nous proposons une nouvelle plate-forme logicielle dédiée à un cadre collaboratif en RA 3D sur table. Après une présentation générale (section 3.3.1) nous

exposerons les principes de conception (section 3.3.2), une présentation détaillée de notre architecture (section 3.3.3), puis nous décrirons les modules proposés (section 3.3.4) et les outils auteurs développés (section 3.3.5). Nous finirons par une discussion présentant les résultats et les limitations de notre approche (section 3.5).

3.4.1. Vue générale

Nous avons choisi de mettre en œuvre un canevas logiciel orienté objet et basé sur une architectures en couches avec une décomposition modulaire (visible figure 4.21). Il est dédié aux développement de collecticiels 3D/2D en RA. Nous avons aussi souhaité une ouverture assez large des possibilités pour *un usage complémentaire en RV ou informatique ubiquitaire*, tendant vers une architecture unifiée. Il offre une des premiere solution dédiée la mise en œuvre de collecticiel 3D de RA sur table.

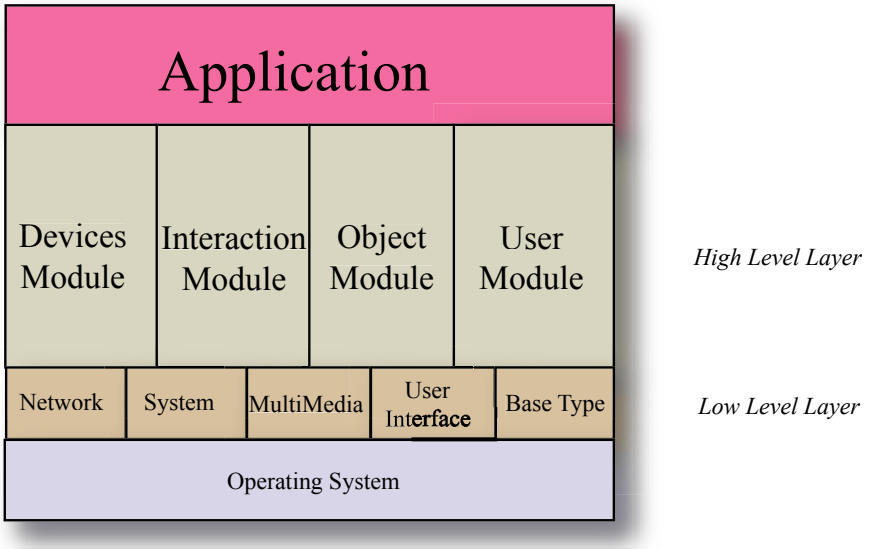
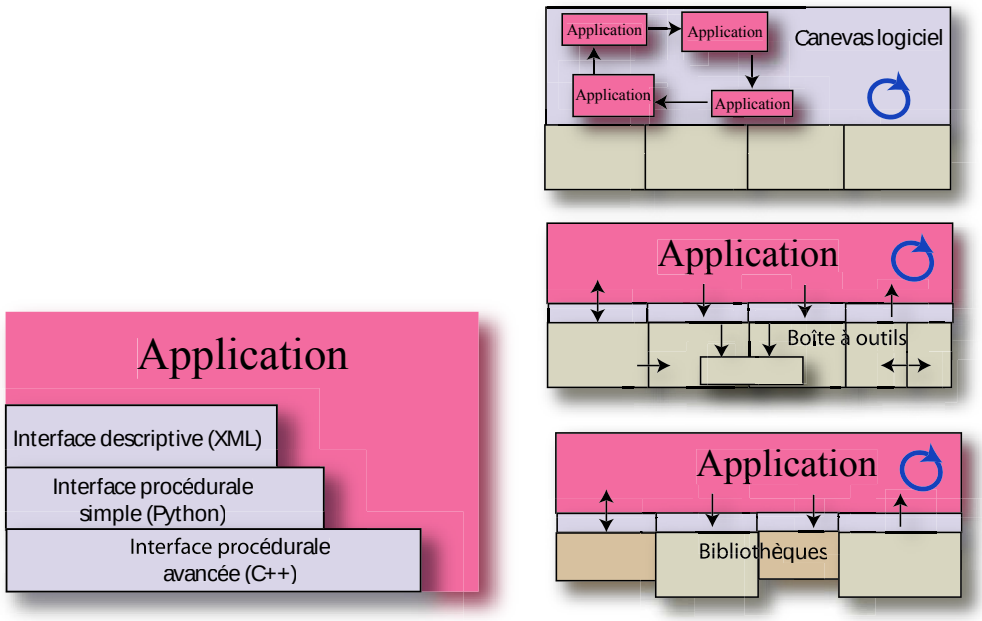


Fig. 4.21: Schéma des couches de notre canevas logiciel.

Nous proposons en premier lieu de *multiples interfaces de programmation* (figure 4.22(a)) :
- une interface procédurale compilé : basé sur le langage C++, elle offre une accés aux propriétés du développement par objets (encapsulation, héritage, abstraction, etc.) ;
- une interface procédurale interprétée : basé sur le langage Python, elle offre un support pour le prototypage et le développement rapide d'application pour le développeur novice (compilation à l'éxecution, faiblement typé) ;
- interface descriptive : basé sur XML, elle offre un support pour la spécification, la modification et l'évaluation rapide d'application.

Avec l'interface procédurale nous introduisons alors *plusieurs modèles d'utilisation* de notre architecture (figure 4.22(b)) :
- utilisation sous forme de bibliothèques : chaque module indépendant offre un service pour l'accès à une fonctionnalité précise ;
- utilisation en boîte à outils : l'utilisateur combine les multiples fonctionnalités des modules avec un contrôle explicite sur leurs possibilités (type d'objets manipulés, métaphore d'interaction choisi, gestion du collaboratif) ;
- utilisation en canevas logiciel : le développeur peut rapidement créer une application en se concentrant sur l'aspect applicatif (contenu et éxecution).

3.4.2. Principe de conception



(a) (b)

Fig. 4.22: Interfaces : (a) différentes interfaces de programmation (b) différents modèles d'utilisation de notre architecture (de haut en bas utilisation par canevas logiciel, par boîte à outils ou par bibliothèques).

La réalisation a été faite avec un langage objet C++. Cette approche nous fournit une forte propriété d'extensibilité pour le développement de notre architecture. Nous avons ainsi largement utilisé les mécanismes des langages objets tels que l'héritage, le polymorphisme, l'abstraction ou l'identification dynamique de type (RTTI).

L'approche retenue a été un respect des principes des *modèle à composants* pour nos classes. Nous avons choisi des interfaces typées et formatées, une description précise de services fournis et le support d'unité d'exécution indépendante ou connectable.

Nous nous sommes reposés sur un certains nombre de patrons de conceptions (*design patterns*) tels que *singleton*, *factory*, *methode template* ou *observer*.

Nous nous sommes principalement concentrés sur les performances haut niveau et l'aspect temps réel (besoins non-fonctionnels) en se posant des problématiques telles que celles soulevées par Ponder [PPM+03] : critique vs. non-critique, services légers vs. services lourds en calcul, contrôle vs. pilotage par événements, synchrone vs. asynchrone, compilé vs. résolution dynamique.

Enfin, nous respectons certains principes directeurs :
- «ne pas ré-inventer la roue» : utiliser au maximum les bibliothèques standards pour les sous domaines spécifiques retenues ;
- «pas une seule solution» : offrir une liberté maximale au développeur lui permettant d'avoir un choix sur l'approche et la méthode suivant le contexte de réalisation ;
- «extreme programming» : offrir rapidement un programme minimal pouvant être étendu facilement.

3.4.3. Présentation

Notre canevas se découpe en deux couches principales : la couche bas niveau et la couche haut niveau modulaire. La couche bas niveau gère les fonctionnalités systèmes et modules standards partagées par les couches fonctionnels (telle que les matrices,vecteurs, etc.). Elle

fournit une abstraction de la couche matérielle et fait le lien avec les librairies ou boîtes à outils utilisées par notre architecture.

La couche modulaire haut niveau nous donne accès aux fonctionnalités courantes pour la gestion des utilisateurs, de l'interaction et du contenu applicatif. Cette couche intègre alors les services pour la RA : mixage visuel, alignement des éléments, interaction avec les éléments réels. Les différents modules sont indépendants, mais peuvent être combinés entre eux (e.g. valeur d'un périphérique en entrée d'une classe de manipulation).

```
<?/xml version="1.0" encoding="iso-8859-1" ?>
<!-- 3d document-->
<document name="testDocument">
 <head>
  <author name="Raphael Grasset" />
 </head>
 <body>
  <spacewindow name="globalSpace">
   <position pos="0. 0. 0.1" rot="0. 0. 0. 1" scale="1. 1. 1." />
   <content>
    <Model3DRepresentation name="cube1">
     <geometry file="tree3Model.pfb" type="pfb" />
     <position pos="-15. -5. 5." rot="0. 0. 0. 1." scale="1. 1. 1." />
     <bounding center="-4.94 0.10 92.79" size="87.04 87.36 185.59" newsize="5. 4. 10." />
    </Model3DRepresentation>
    <Model3DRepresentation name="table3D">
     <geometry file="tableModel.pfb" type="pfb" />
     <position pos="0. 0. 0." rot="0. 0. 0. 1." scale="1. 1. 1." />
     <bounding center="0. 0. 0." size="1. 1. 1." newsize="50. 50. 0.2" />
    </Model3DRepresentation>
   </content>
  </spacewindow>
 </body>
</document>
```

Fig. 4.23: Exemple de fichier XML pour configurer le contenu d'une application.

L'architecture se décompose en classes organisées pas paquetages et sous paquetages, chacun de ces éléments pouvant être aisement étendu. L'utilisation de l'interface C++ permet d'accédér et manipuler directement les classes de l'architecture, à partir des arbre d'héritage de classes déjà fortement développés. L'interface Python permet d'accèder à un ensemble des fonctionnalités, dédiée au développement rapide d'applications simples. En complément la configurabilité est fortement assurée par l'utilisation d'interface de méthodes normalisées pour l'accès au propriétés des objets et par l'usage de fichier XML (figure 4.23).

L'interface canevas logiciel (illustrée figure 4.24) est utilisée à partir d'un squelette prédéfini d'application que l'utilisateur peut surcharger et complèter (similaire à *VRJuggler* [Jug]). On propose plusieurs exemples d'interface «classiques» (comme deux utilisateurs avec une visualisation 3D et un pointeur d'interaction). Dans notre thèse nous avons surtout favorisé l'interface par boîte à outils, les expérimentations étant très versatiles.

Au niveau de l'éxecution, notre environnement - reposant principalement sur *OpenGL Performer* - supporte le multi-processeur/multi-thread, la communication synchrone et asynchrone permettant la mise en œuvre de solutions temps réel (sous forme d'appel direct de procédure, de protocole par message ou par le concept de signal/slot).

3.4.4. Description des fonctionnalités

Nous présentons maintenant rapidement les services fournis par notre environnement. Tous les détails se trouvent dans l' annexe A.

Couche bas niveau

Nous nous appuyons principalement sur des librairies standards pour mettre en place une couche homogène. Les services matériels gérés sont : la gestion du système, du réseau, des

RotationSlider.C

```
...
void RotationSlider::Update(PosType pos, User user)
{
   switch (state):
   {
     case SLIDER_NONE:
        if (user.name=="raphael")
         {
           intersect->Update(pos);
           if (intersect->isIntersect())
          {
              state=SLIDER_SELECT;
         }
....
```

MyApplication.C

```
....
void MyApplication::InitDevices()
{
  trackball=new TrackballInput(3);
  trackball->Init();
}

void MyApplication::UpdateContent()
{
  PosType* pos=trackball->GetData();
  ObjectManager::GetObject("car")->SetPosition(pos);
}
...
```

main.C

```
#include "MyApplication.h"


int
main(int argc,char *argv[],char *envp[])
{
  Application *app=new MyApplication(argc,argv,envp);
  app->Init();
  app->Run();
  app->Finish();
  delete app;
  return 0;
}
```

Fig. 4.24: Exemple d'application à partir du canevas logiciel. L'utilisateur peut définir et surcharger l'interface graphique 3D (`RotationSlider.C`), la manipulation des objets (`MyApplication.C`). Le code principale d'appel de l'application est très classique (`main.C`).

péripheriques, de l'audio, de la video et du graphisme. Au niveau applicatif nous avons retenu des services algorithmique (structure de contrôle avancées) et des services d'accès aux données (XML). Enfin au niveau de l'interaction nous avons surtout retenu l'accès aux interfaces graphiques de bureau. Les bibliothèques complémentaire que nous avons écrites représentent environ 10000 lignes de C++.

Module `Périphérique`

Il décrit la gestion des péripheriques pouvant se connecter à un système et accessibles par une application. Basé sur une approche hybride inspirée d'*OpenTracker* [SR01] et de *Microsoft DirectShow* [Sho], chaque élément du module peut être interconnecté avec une gestion par flots de données (composabilité par des profiles d'interfaces). La figure 4.25 montre un exemple d'utilisation pour une application. On trouve principalement la gestion des dispositifs physiques, des filtres, des modificateurs, des éléments de combinaisons (aggrégation de données). Il représente 30 000 lignes de C++.

Module `Interaction`

Ce module intègre les mécanismes de traitement des métaphores d'interaction fournissant une interface entre périphériques et objets. Il fournit un support indépendant des métaphores avec les périphériques se reposant uniquement sur les types de données fournis en entrée. Il fournit aussi des mécanismes de retour perceptuels aux utilisateurs (visuel ou sonore). Dans un cadre 3D nous nous reposons fortement sur le mécanisme de détection de collision (type segment-polytopes) disponible sous OpenGL Performer (inspiré de

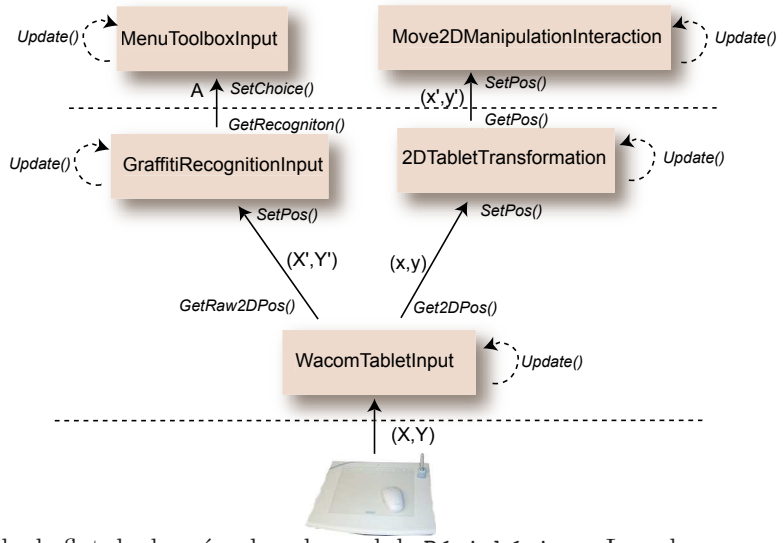


Fig. 4.25: Exemple de flot de données dans le module `Périphérique`. La valeur mesurée avec la tablette sert pour la reconnaissance de signe (lettre A) ou le déplacement 2D (position).

H-Collide [GLGT99]). On trouve des composants tels que la gestion de la navigation en RA, la gestion des métaphores de sélection d'élément, des interfaces graphiques, etc. Ce module représente environ 20 000 lignes de C++.

Module `Objet`

Ce module est chargé du support du contenu applicatif (figure 4.27). Dans ce cadre nous nous reposons sur le modèle MVC (*Model View Controller*), et nous introduisons les éléments `modèle` (élément applicatif), `vue` (représentation) et `moteur de rendu` (rend la vue). Nous introduisons dans ce cadre des modèles et des vues par défaut, que l'utilisateur peut alors étendre. Le contenu d'une scène peut être décrit à plusieurs niveaux : sous formes d'une description déclarative, sous forme d'un graphe de scène (par l'intermédiaire d'OpenGL Performer), ou sous forme d'objets instanciés. On gère alors de multiples formats de données : objets 3D (réel ou virtuel), son 3D, texte 2D et 3D, etc. Ce module représente quelques 20 000 lignes de C++.

Fig. 4.26: Contenu multimédia virtuel et réel : objets 3D, images, vidéos, objet son (boutons blancs), etc.

Ce module intégre aussi des mécanismes de gestion de données avancées par l'intégration

de librairies haut niveau. De ce fait, notre architecture supporte le rendu volumique ou la simulation physique d'objets rigides (figure 4.27).

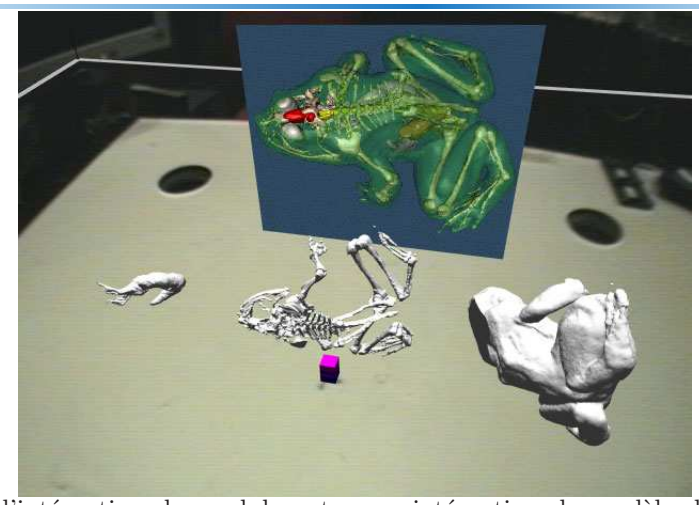


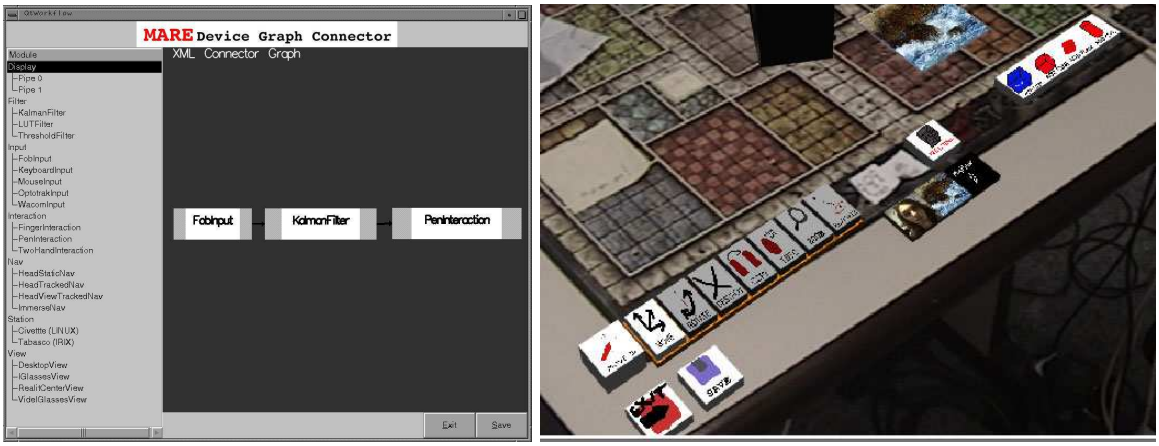
Fig. 4.27: Exemple d'intégration de module externe : intégration de modèle de grenouille depuis la librairie VTK.

Module `Utilisateur`

Ce module permet d'introduire la notion d'utilisateur, lui offrant un contexte applicatif personnel, un support de ressources dédiées et des droits de contrôle sur l'application. On trouve par exemple la gestion d'un pointeur 3D, les mécanismes de protection ou de perception des autres utilisateurs.

3.4.5. Outils auteur et développement

Bauer [BBK+01] introduit quatre vues pour un développement logiciel reposant sur une approche par composants : vue du directeur de projet (composants ré-utilisables), vue de l'utilisateur (module connectable) et vue du développeur d'application (services en couches à remplir), vue du développeur de modules (besoins et services). Dés l'or, elle structure son architecture pour suivre ces différentes vues. Nous avons choisi ici de reprendre ce principe dans le cadre de propositions d'outils auteurs (figure 4.28). Nous avons identifés plusieurs catégories présentées dans la suite.



(a) (b)

Fig. 4.28: Exemple d'outils auteurs : (a) MAREDevicesConnector (b) MAREARSceneEditor.

Outils de développement : nous proposons un certain nombre d'outils pour l'extension de notre environnement ou la construction d'applications (inspiré de *DWARF* et *Studierstube*). Ces outils permettent de réaliser des simulations d'éléments en entrée (remplacés par un capteur) ou en sorties (affichage dégradé) sous la forme de composants graphiques paramétrables. Divers outils permettent la conception d'interfaces ou la génération de squelettes applicatifs.

Outils administratifs : Nous introduisons plusieurs outils pour configurer différents éléments de la plate-forme logicielle et matérielle. Le premier est un éditeur graphique pour définir le flot de données des périphériques utilisés, `MAREDevicesConnector` (similaire à *ICON* [DF01]). Il intègre entre autres un mécanisme de découverte des ressources disponibles similaires à *UPnP* [Pla] dédié à des machines en réseau local (`MAREDiscoverRessources`). Nous proposons une liste d'utilitaires graphiques pour le calibrage des différents éléments (`MARECalibEditor`), chacun définissant leurs propres interfaces.

Outils applicatifs : orientés principalement dans le cadre de réunion, nous les avons décomposés en trois catégories : avant, pendant et après la réunion (définis ci-après comme session). Nous introduisons différents outils graphique pour définir le contenu : configurer des objets (`MAREObjectEditor`), assembler des objets (`MARESceneEditor`). Nous introduisons aussi une nouvelle approche dans le même esprit que Grimm [PHP+02] qui consiste à utiliser des outils auteurs dans le réel. Dans ce cadre, nous introduisons par exemple `MAREARSceneEditor` qui permet de créer et éditer une scène avec un système de RA mono-utilisateur. Enfin, nous proposons un éditeur pour rejouer et analyser la session après la session (`MARESessionPlayer`).

3.5. Discussion

Utilisation : Cette architecture a été largement utilisée durant les trois années de notre thèse pour développer un grand nombre de prototypes de tests. Elle répondu à nos exigences offrant une grande flexibilité. Elle a aussi permis la mise en œuvre de différents prototypes applicatifs dans le cadre de la chimie, de l'architecture ou du jeu.

Réalisation : Notre architecture comprend environ 300 classes réalisées et une dizaine d'outils de développement. Nous en sommes actuellement à la troisième version avec plus de 100000 lignes de C++ (comprenant les différents modules, les programmes d'expérimentations et les outils auteurs).

Comparaison aux solutions actuelles : notre librairie offre un grand nombre de fonctionnalités similaires à *Studierstube* mais proposées sur une plate-forme bas niveau (OpenGL Performer) plus performante, offrant un support multiprocesseurs, multi cartes graphiques et un contrôle fin du niveau d'exécution. Dédiée à un cadre de réunion sur table, notre solution semble plus développée que l'approche de *MagicMeeting* avec un intérêt particulier pour l'interface avec le collaboratif et de très nombreux périphériques. Notre plate-forme offre un aspect dédié au temps réel semblant offrir une solution plus efficace que *DWARF* mais beaucoup moins souple et extensible. Enfin, nous noterons que notre approche tente de fournir un environnement unificateur pour la RA/RV/informatique ubiquitaire, approche faiblement démontrée à ce jour.

Limitations : les principales limitations de cette architecture sont :
- un noyau centralisé n'autorisant pas une distribution de l'application (un utilisateur par machine) ;
- un mécanisme de communications essentiellement synchrone, nécessitant une extension à des mécanismes plus souples ;
- pas d'abstraction du modèle de graphe de scènes (utilisation d'OpenGL Performer) ;
- développement d'applications uniquement à partir de l'architecture, pas de mécanismes externes de gestion d'applications existantes (*EAI, External Authoring Interface* comme

proposé en RA par Schmalstieg [SR01] ou en RV dans *Spin-3D*[dP03]) ;
– peu de gestion des fonctionnalités de l'informatique pervasive (dispositifs mobiles sans fils, éléments physiques).

Le choix d'une architecture en couche ne semble pas adapté à une forte distribution, où une approche par composants serait plus intéressante ([MAC+02]).

Il nous semble de plus que l'étude des outils auteurs en RA reste un élément à étudier plus précisément (comme le propose Kulas [KSK04]). Par ailleur l'apport d'outils de conception visuels et souples devrait être aussi proposé aux gens de notre domaine leur offrant des moyens de développement rapide (un *AR IDE*).

4. Conclusion

Ce chapitre propose une nouvelle architecture pour la collaboration sur table colocalisée. Cette contribution originale est double :

– nous proposons une **architecture matérielle** offrant une grande souplesse et un support hétérogène (et adapté) pour la visualisation et l'interaction avec des tâches virtuelles 3D (complétées par un retour sonore). Nous proposons une liste de besoins fonctionnels et non-fonctionnels pour les architectures matériels de RA collaboratif sur table. Ces éléments peuvent fournir des briques pour la définitions de futures architectures. Nous proposons en outre deux prototypes originaux de dispositfs d'interaction peu intrusifs. Nous introduisons enfin une représentation par modélisation visuelle qui offre une nouvelle voie pour la définition d'environnements de RA ;
– nous introduisons une nouvelle **architecture logicielle** dédiée à la TCAO pour des tâches 3D. Nous avons identifés une liste exhaustive de besoins dans ce cadre. Nous offrons un environnement générique pour le développement de collecticiels en RA 3D, supportant aussi le développpement d'applications de RV ou d'informatique ubiquitaire. Les propriétés de notre architecture sont une forte modularité, un support multi-tâches, une dynamicité, une extensibilité, de multiples interfaces de programmations, des outils auteurs et un large ensemble initial de composants proposés.

Chapitre 5

Mixage : virtuel et réel

1. Introduction

L'alignement stable et précis entre objets réels et objets virtuels dans notre environnement est un élément indispensable pour garantir une visualisation cohérente. Pour ce faire, il est nécessaire de mettre en jeu des méthodes pour garantir cet alignement d'un point de vue spatial et temporel. La problématique de ce chapitre peut alors se résumer ainsi : «Comment fournir et garantir un mixage de bonne qualité entre monde réel et monde virtuel dans un cadre collaboratif en RA sur table ?» (figure 5.1).

Un point non abordé dans ce chapitre concerne le mixage visuel entre les deux mondes. La gestion des occultations (relation de visibilité entre les éléments) mais aussi de l'éclairage cohérent entre les deux mondes sont deux des critères les plus pertinents. Une intégration réaliste des objets conduira à une meilleur sensation de «fusion» entre ces deux mondes. L'usage d'ombre permettrait de guider les métaphores d'interactions avec de meilleurs indices visuels (comme le démontre Dumas [DDC+99] dans le projet *SPIN-3D* et plus récemment Sugano [SKT03] en RA). Nous présenterons au chapitre 7, quelques techniques pour la gestion des occultations.

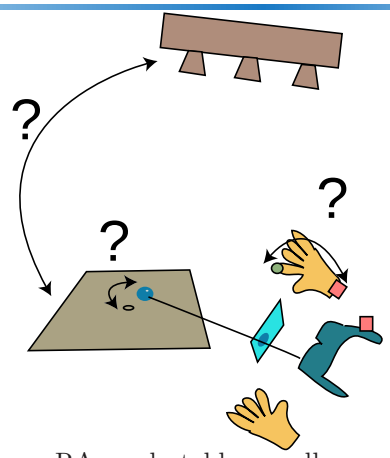


Fig. 5.1: Problématique du mixage en RA sur la table : quelles sont les relations entre les éléments de l'environnement ? Comment bien intégrer les éléments virtuels avec le contexte réel ?

Nous proposons dans ce chapitre différentes techniques répondant à ce problème et dédiées à une gestion multi-utilisateurs pour notre environnement sur table. Après une description du problème (section 2), nous présenterons les problématiques liées à l'alignement des deux mondes en position fixe (section 3) puis dans un cadre dynamique (section 4). Nous présenterons ensuite des nouvelles techniques pour introduire de facon dynamique des objets réels dans des scènes virtuelles (section 5 et 6).

2. Integration spatiale : concepts et définitions

L'alignement (*registration*) entre réel et virtuel correspond au positionnement exact d'objets virtuels projetés dans l'image du monde réel perçu par l'utilisateur. Cet alignement peut être fait directement dans l'image. Il peut aussi être le résultat d'une suite de

transformations appliquées aux objets virtuels et dépendant du système physique utilisé : casque optique suivi, casque vidéo avec système binoculaire, caméra statique, etc.

L'obtention (et la conservation) de cet alignement reste à l'heure actuelle non triviale dans sa mise en œuvre (erreur d'alignement), alors qu'il est un critère discriminant pour la perception visuelle d'intégration des deux mondes (le système visuel est très sensible aux différences dynamique). Nous présentons maintenant les causes de cette difficulté (figure 5.2).

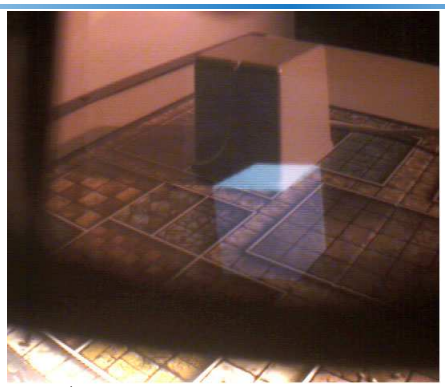


Fig. 5.2: Problème de l'alignement R/V : le cube virtuel devrait être aligné avec le cube réel, cet objet virtuel étant sa représentation numérique. Quelles sont les causes de cette erreur ?

Azuma lors de ses travaux [AB94] s'est beaucoup intéressé au prérequis pour l'alignement entre réel et virtuel (alignement R/V). Il a défini deux catégories d'erreurs :

- *erreur statique* : définit comme le mauvais alignement perçu par l'utilisateur en position immobile ;
- *erreur dynamique* : définit comme l'erreur d'alignement lorsque l'utilisateur est en mouvement.

L'**erreur statique** est due à de nombreuses causes d'après Holloway [Hol95b] : distorsion numérique, erreur du système de suivi, mauvais alignement mécanique, mauvaise estimation des paramètres (champ de vue, distance œil capteur, distance interpupillaire, etc.). Dans la bibliographie, la mauvaise estimation des paramètres a été l'erreur la plus étudié. Une estimation fiable des paramètres du système nécessite de mettre en œuvre des méthodes robustes basées sur une définition de contraintes, nommées procédures de **calibrage** (*calibration*). A partir d'une modélisation de l'environnement et de ses transformations, il faut alors concevoir des techniques permettant d'estimer ces transformations ou d'ajuster les valeurs obtenues.

L'**erreur dynamique** est quant à elle principalement due au temps de retard entre le mouvement d'objets ou de l'utilisateur et le retour visuel de cette information : ce délai est nommé la **latence** du système. Elle est d'après Mine [Min93] la résultante d'une succession de délais de traitement par les différents éléments mis en jeu pour la restitution visuelle (temps de mesures des capteurs, temps de transmission, temps de réception et traitement, temps de mise à jour de l'application, etc.). La correction de cette erreur passe alors par une approche de réduction de cette latence, sa disparition étant techniquement impossible (l'application et le système ont toujours un coût non réductible en temps de traitement).

Sous un autre angle, Holloway [Hol95a] range les erreurs en quatre grandes classes : acquisition des données (acquisition et alignement de l'environnement), suivi (estimation de la position de la tête de l'utilisateur), vue (estimation du modèle du point de vue de l'utilisateur) et affichage (erreurs du périphérique tel que mauvais alignement de l'image, aliasage, non linéarité du dispositif d'affichage, distorsion optique, aberration des couleurs).

Dans cette thèse il nous semble donc nécessaire de réduire erreur statique et erreur dynamique, en s'appuyant sur les différents critères et classes définis par Holloway et Mine.

Nous noterons qu'au delà de vouloir réduire ces erreurs, leur estimation constitue une difficulté en amont. A ce propos, Holloway [Hol95b] introduit un modèle mathématique théorique de l'erreur d'alignement. McGarrity, Tuceryan et al.[mT+01] ont défini quant à eux une méthode d'estimation de l'erreur à partir d'une projection planaire (de façon similaire aux approches automatiques utilisées en vision). Nous nous baserons sur cette dernière approche pour nos évaluations, le modèle d'Holloway restant complexe à utiliser.

Il nous a semblé judicieux d'identifier les paramètres de base permettant la mise en œuvre d'un système d'un niveau suffisant (non optimal), dédié à notre configuration. Nous souhaitons obtenir une solution offrant une grande flexibilité et avec une erreur minimale pour garantir une utilisation de l'environnement dans de bonnes conditions. Dans ce but, nous introduisons dans la suite notre approche pour gérer l'erreur statique (section 3) et pour l'erreur dynamique (section 4).

3. Erreur statique : calibrage

Après une brève synthèse de l'état de l'art nous introduirons notre méthode, sa mise en œuvre et les résultats obtenus.

3.1. Etat de l'art : analyse

Le lecteur pourra se reporter à notre article [GG03] pour un état de l'art sur les techniques de mixage en RA. Nous présentons ici juste une analyse de cet étude.

L'ensemble des méthodes proposées reste difficile à mettre en œuvre. Leur *évaluation robuste* constitue aussi un obstacle à leur validation (pas uniquement basées sur des critères perceptuels[1]). Nous retiendrons les critères suivants pour la mise en œuvre de techniques de calibrage : robustesse, flexibilité (multiples configurations), précision, répétabilité, facilité d'utilisation, indépendance (utilisateur, matériel, étapes précédentes), coût.

Au vu des solutions proposées, nous avons identifié trois classes de canevas complet de calibrage :
- technique partielle/globale complexe avec lourd appareillage externe telle que Oishi [OT96] ;
- technique partielle/globale «difficile» telle que Garrity [MT99] ;
- technique globale simple et intuitive telle que Ikits [IHJ03], Summers [SBC+99], Fuhrmann [FPTP01] (ou Genc [GTN02] pour une approche partielle).

Dans cette thèse nous favoriserons la dernière approche. Malheureusement les techniques de cette approche nous semblent peu adaptées au contexte de table, à notre configuration matérielle et au cadre collaboratif. De plus, elles nous semblent non optimales d'un point de vue de l'interface et de l'efficacité des tâches proposées. Nous introduisons donc dans la section suivante une nouvelle approche.

3.2. Notre Approche : vers le plug'n play

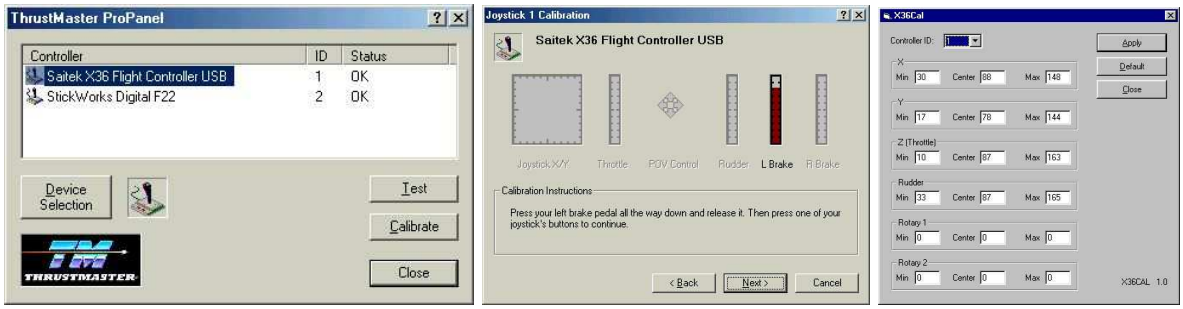
Nous proposons un *canevas de calibrage* pour notre système reposant sur différents critères. Premièrement, nous avons choisi une approche itérative guidée par l'utilisateur, et basée sur une succession d'opérations simples, rapides, et contraintes.

Deuxièmement, nous nous sommes orientés vers des méthodes *intuitives guidées par interface graphique* en s'inspirant des outils utilisés pour configurer les joysticks sur PC (voir figure 5.3). L'utilisateur a un retour visuel à différentes étapes lui décrivant la tâche à ce niveau. En fin de calibrage, il peut valider le résultat complet à l'aide d'un outil de test.

[1] En effet il semble que la meilleur façon de juger le bon alignement c'est une évaluation visuelle par l'utilisateur, qui posent des difficultés de généricité de cette approche (calibrage adapté à un utilisateur, compétences de jugement de l'utilisateur, etc.)

Nous avons repris ce concept dans le cadre de notre environnement, en nommant cette approche *plug'n calib and play*, la phase de calibrage étant nécessaire mais minimale.

Le troisième point à noter est la distinction entre périphériques privés et publics, et donc l'introduction de *techniques globales ou personnelles*. Les éléments publics sont configurés par une seule personne, l'organisateur d'une session. Nous considérons le cas de la possession de périphériques privés par un utilisateur qui peut les apporter lors d'une réunion. Il est alors nécessaire d'avoir une méthode de calibrage rapide et efficace et réitérée pour chaque nouvel utilisateur (e.g. applicable au jeu).



(a) choix du périphérique à calibrer

(b) configuration par interface graphique

(c) configuration manuelle

Fig. 5.3: Interface utilisée pour calibrer un joystick sur ordinateur individuel[Cal].

Pour notre thèse, nous faisons un certain nombre de remarques :

- Les méthodes de calibrage proposées doivent être facilement utilisable dans notre environnement (table) et s'orienter vers la minimisation des déplacements des utilisateurs (principalement assise) ;
- Les méthodes proposées doivent être réalisable par un utilisateur naif ;
- Les méthodes doivent permettre l'aspect collaboratif, avec des actions simultanées ou séquentielles pour les différents utilisateurs ;
- Les méthodes doivent supporter un calibrage en cours de session, minimisant les modifications architecturales. Elles doivent donc être rapide.

L'erreur résiduelle après calibrage doit être minimale. Nous orientant vers des réunions sur table, notre intérêt va vers des applications non-critiques. Au vu des possibilités des périphériques, Nous choisissons une réduction à un erreur de quelques millimètres.

Nous présentons à partir de ces différents critères, la modélisation choisie puis les méthodes mises en œuvre.

3.2.1. Modélisation de notre configuration

A partir de la description des composantes matérielles, nous avons identifié les principaux repères de notre système (figure 5.4) et les transformations nécessaires pour l'utilisation de notre système (tableau 5.2). L'ensemble des transformations est choisi à partir des approches de modélisation faites par Robinet [RH94] pour un système de visualisation en RV et Tuceryan [TGW+95a] pour son système de RA basé moniteur.

Concernant les notations, une transformation d'un système de coordonnées B vers A sera noté $T_{A \leftarrow B}$. Ces transformations de l'espace sont inversibles, et $T^{-1}_{A \leftarrow B} = T_{B \leftarrow A}$. Nous noterons les projections ($2D \leftarrow 3D$) $P_{A \leftarrow B}$. Les repères sont définis par des termes anglais par convention, et listés dans le tableau 5.1.

Comme Tuceryan [TGW+95a], nous avons déterminé pour chacune des transformations son statut temporel (fixe ou dynamique), sa nature (spécifique à l'utilisateur, spécifique à l'application, indépendant). Chaque transformation T peut générer une erreur (note ΔT) qui va conditionner l'erreur finale dans le cas d'une succession de transformations. Une

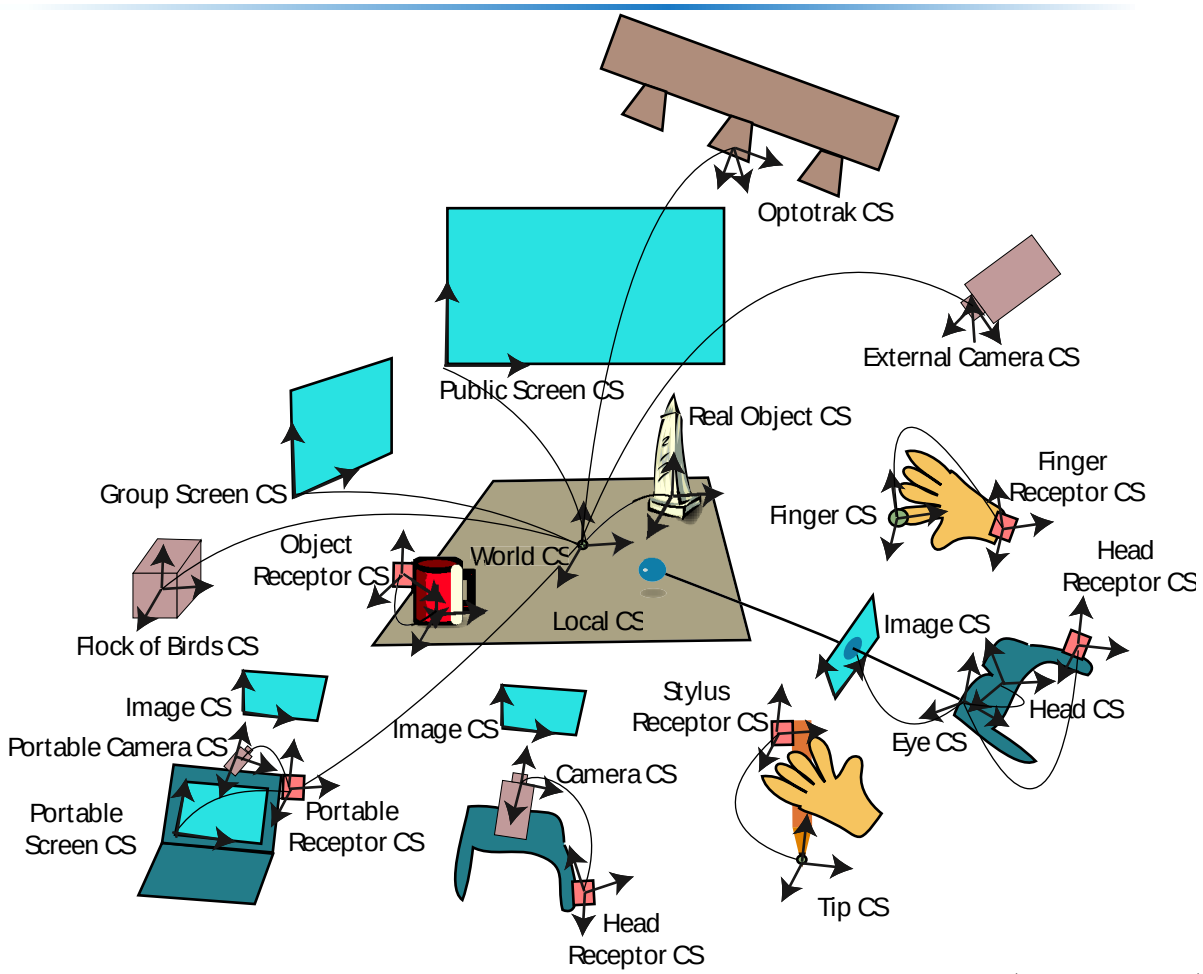


Fig. 5.4: Schéma de modélisation de notre configuration. Nous noterons que le portable (en bas à gauche) est défini soit comme écran de groupe (seul Portable Screen CS est défini), écran de groupe suivi (on rajoute Portable Receptor CS), soit comme dispositifs individuelle de RA (repères actifs alors similaire au casque vidéo semi-transparent).

transformation fixe ne nécessitera donc qu'une estimation initiale qui minimisera l'erreur alors qu'une transformation dynamique nécessitera une correction à chaque pas de temps. De la même façon que Ikits [IHJ03] nous définissons l'erreur nominale comme l'erreur que l'on peut obtenir sans technique de calibrage, avec une méthode ad-hoc (par itération avec des «essais-erreurs»).

Un certain nombre d'éléments sont calibrés de façon interne (paramètres intrinsèques), telle que le mode de fonctionnement du capteur magnétique. Dans notre cadre nous considérons le calibrage comme l'estimation des paramètres manquants pour l'utilisation conjointe de l'ensemble des éléments.

Nous avons classé les transformations à obtenir en trois catégories distinctes :
- calibrage des systèmes de suivi ($T_{W\leftarrow O}$, $T_{W\leftarrow F}$, $T_{W\leftarrow EC}$) ;
- calibrage des systèmes de visualisation : public/groupe ($T_{W\leftarrow PS}$, $T_{W\leftarrow GS}$, $T_{W\leftarrow POS}$, $T_{POS\leftarrow POR}$), personnel ($T_{H\leftarrow HR}$, $T_{E\leftarrow H}$, $T_{I\leftarrow E}$, $T_{C\leftarrow HR}$, $T_{I\leftarrow C}$, $T_{POC\leftarrow POR}$, $T_{I\leftarrow POC}$) ;
- calibrage des systèmes d'interaction ($T_{W\leftarrow TA}$, $T_{TI\leftarrow SR}$, $T_{FI\leftarrow FR}$, $T_{W\leftarrow RO}$, $T_{RO\leftarrow OR}$).

3.2.2. Calibrage des systèmes de suivi

Méthode et mise en œuvre

Les différents systèmes de suivi doivent être définis dans le repère globale positionné au centre de la table ($T_{W\leftarrow O}$, $T_{W\leftarrow F}$, $T_{W\leftarrow EC}$). L'utilisateur détermine la transformation

Tab. 5.1: Listes des différents repères de notre configuration.

Repère	Description
World CS (W)	repère globale choisi pour l'ensemble des éléments, définit au centre de la table
Local CS (L)	repère utilisé pour une application spécifique
Optotrak CS (O)	repère du capteur optique fixé au mur
Flock of Birds CS (F)	repère de l'émetteur du capteur magnétique
External Camera CS (EC)	repère de la caméra externe pour le suivi
Public Screen CS (PS)	repère de l'écran publique
Group Screen CS (GS)	repère de l'écran de groupe
Portable Screen CS (POS)	repère de l'écran de groupe de type portable
Head CS (H)	repère de la tête d'un utilisateur pour le casque semi-transparent optique
Eye CS (E)	repère d'un œil d'un utilisateur pour le casque semi-transparent optique
Camera CS (C)	repère de la caméra pour le casque semi-transparent vidéo
Portable Camera CS (POC)	repère de la caméra pour le portable semi-transparent vidéo
Head Receptor CS (HR)	repère d'un capteur associé à un système de visualisation sur tête.
Portable Receptor CS (POR)	repère d'un capteur associé au système de visualisation portable (pour le suivi en écran de groupe ou en mode semi-transparent).
Image CS (I)	repère de l'image vue par l'utilisateur
Tablet CS (TA)	repère de la tablette digitale
Real Objet CS (RO)	repère d'un objet réel : centre ou point d'interaction
Object Receptor CS (OR)	repère d'un capteur associé à un objet réel
Tip CS (T)	repère du point d'interaction d'un objet de type pointeur
Stylus Receptor CS (SR)	repère d'un capteur associé à une interface de type pointeur
Finger CS (FH)	repère d'un point d'interaction d'une interface gestuelle
Finger Receptor CS (FR)	repère d'un capteur associé à une interface digitale

entre le système de suivi et la table. Nous utilisons un objet de référence du système de suivi pour calibrer ce périphérique. Nous nous baserons sur des points discriminants des éléments de la table, servant de points de références pour les étapes de calibrage. Nous décrivons brièvement les choix pour les différents systèmes (illustrés figure 5.5).

Nous calibrons l'Optotrak à l'aide d'un élément rigide à 6 DDL fourni avec le système (objet *PLATE*), que l'on place au centre de la table, sur le repère d'origine (point de référence sur la première led du *PLATE*). Nous effectuons plusieurs mesures et nous moyennons le résultat pour obtenir la transformation $T_{W \leftarrow O}$.

Une autre approche consiste à positionner l'élément aux quatre coins de la table et au centre. En pratique nous observons une différence très faible entre les deux méthodes (quelques mm en position, - de 1 degré en orientation), la première étant plus facile à mettre en œuvre.

Nous calibrons le système magnétique suivant le même principe. La non connaissance à priori du centre du capteur nous conduit à utiliser un élément ponctuel (3 DDL) représenté ici par un stylo (point de référence à la pointe du stylo). Le stylo est lui même calibré suivant une procédure indépendante de tout autre calibrage (section 3.2.5). Nous appliquons la méthode consistant à cliquer sur les différents coins de la table et le centre. Ce qui nous permet de reconstruire les 6 DDL d'un repère et nous permet de calculer la

Tab. 5.2: Listes des transformations utilisées de notre configuration. La notation est la suivante : F=Fixe, D=Dynamique, I=Independant, U=dépendant de l'utilisateur, A=dépendant de l'application.

Identifiant	Description	F	D	I	U	A	Précision Nominale
$T_{W\leftarrow L}$	Local-to-World	X				X	10-20 mm
$T_{W\leftarrow O}$	Optotrak-to-World	X		X			10-20 mm
$T_{O\leftarrow R}$	Receptor-to-Optotrak		X	X			–
$T_{W\leftarrow F}$	Flock-of-birds-to-World	X		X			10-30 mm
$T_{F\leftarrow R}$	Receptor-to-Flock-of-birds		X	X			–
$T_{W\leftarrow EC}$	External-Camera-to-World	X		X			10-30 mm
$T_{EC\leftarrow R}$	Receptor-to-External-Camera		X	X			–
$T_{W\leftarrow PS}$	Public-Screen-to-World	X		X			10-20 mm
$T_{W\leftarrow GS}$	Group-Screen-to-World	X		X			10-20 mm
$T_{W\leftarrow POS}$	POrtable-Screen-to-World	X		X			10-20 mm
$T_{POS\leftarrow POR}$	POrtable-Receptor-to-POrtable-Screen	X		X			10-20 mm
$T_{H\leftarrow HR}$	Head-Receptor-to-Head	X			X		10-30 mm
$T_{E\leftarrow H}$	Head-to-Eye	X			X		2-10 mm
$T_{I\leftarrow E}$	Eye-to-Image	X			X		2-10 mm
$T_{C\leftarrow HR}$	Head-Receptor-to-Camera	X			X		10-30 mm
$T_{I\leftarrow C}$	Camera-to-Image	X		X			2-10 mm
$T_{POC\leftarrow POR}$	POrtable-Receptor-to-POrtable-Camera	X		X			10-30 mm
$T_{I\leftarrow POC}$	POrtable-Camera-to-Image	X		X			2-10 mm
$T_{W\leftarrow TA}$	TAblet-to-World	X				X	2-5 mm
$T_{W\leftarrow RO}$	Real-Object-to-World	X				X	2-5 mm
$T_{TI\leftarrow SR}$	Stylus-Receptor-to-TIp	X		X			2-5 mm
$T_{FI\leftarrow FR}$	Finger-Receptor-to-FInger	X			X		2-5 mm
$T_{RO\leftarrow OR}$	Object-Receptor-to-Real-Object	X				X	2-5 mm

transformation $T_{W\leftarrow F}$.

Pour le calibrage de la caméra statique, nous utilisons une mire de calibrage en se reposant sur la méthode de Heikkila [Hei00] (point de réference à une distance relative du premier point de la grille). Cet algorithme nous fournit les paramètres intrinsèques et extrinsèques. Le système étant couplé avec la librairie de suivi *ArToolkit*, il est nécessaire de calibrer aussi les paramètres permettant un suivi des marqueurs à 6 DDL dans le repère de la table (les valeurs des paramètres extrinsèques entre la librairie et les valeurs fournis par l'algorithme d'Heikkila diffère fortement). Nous réutilisons les paramètres intrinsèques obtenus par Heikkila pour initialiser le suivi. Nous positionnons ensuite un marqueur de façon séquentielle aux coins de la table et au centre, nous fournissant alors la transformation entre le repère de la caméra et la table ($T_{W\leftarrow EC}$).

Résultats

Nous validons les différents algorithmes, en déplaçant différents éléments de calibrage sur le plan de la table à différentes positions (régulières ou aléatoires) et en calculant la différence entre la position correcte et la position mesurée. Nous avons réalisé dans ce cadre 20 mesures pour les différents capteurs. Nous constatons alors une erreur moyenne de 0.01 cm pour l'Optotrak, de 0.15 cm pour le Flock of Birds et une erreur moyenne de 0.4 cm pour le suivi ARToolkit.

3.2.3. Calibrage écrans : groupe et public

L'obtention de la transformation $T_{W\leftarrow GS}$ ou $T_{W\leftarrow POS}$ s'obtient facilement en cliquant sur 3 ou 4 bords de l'écran (la position exacte de l'écran physique par rapport au bord est connue d'avance). Si le portable est suivi, à partir de la connaissance de $T_{W\leftarrow POS}$, $T_{W\leftarrow O}$ et de la valeur mesurée de $T_{O\leftarrow POR}$ nous en déduisons facilement et rapidement la transformation (position du capteur dans le repère local de l'écran).

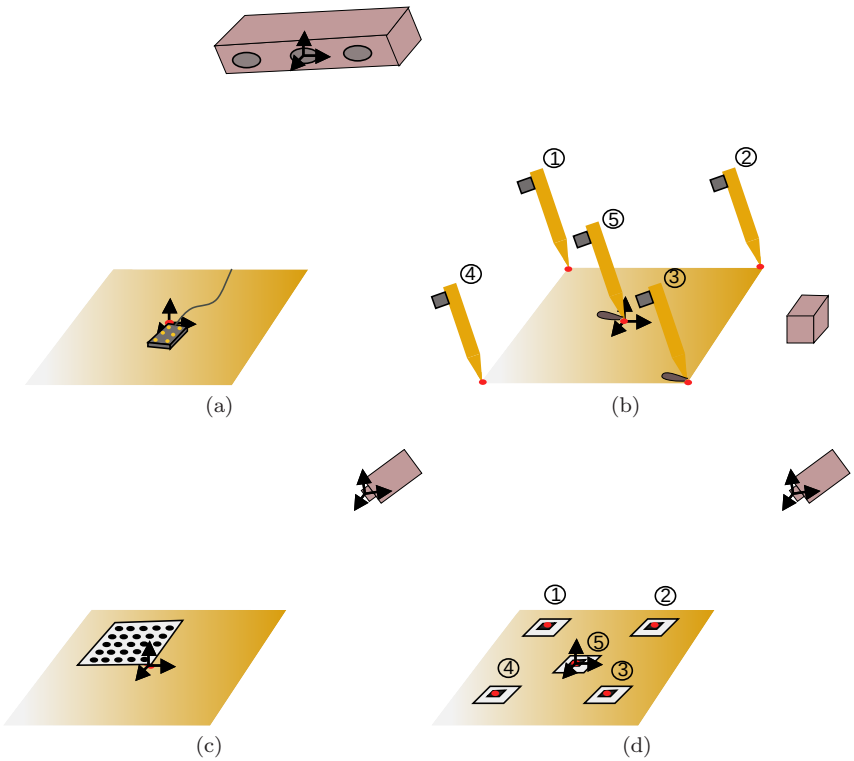


Fig. 5.5: Schéma résumé des méthodes de calibrage des systèmes de suivi : optique, magnétique, caméra et algorithme de suivi ARtoolkit. Le point de réference est indiqué par un disque rouge.

Pour l'écran public nous ne considérons pas ici un décalage entre le centre de projection et la position du projecteur, ce cas pouvant se résoudre pour une estimation d'homographie (voir des travaux comme Raskar [RvBC02] pour des méthodes de correction). La position de l'écran public n'est pas mesurable à l'aide d'un capteur (hors champs) et pas accessible physiquement (utilisateur assis et à distance de l'écran par hypothèses). Une solution pourrait être la suivante : l'utilisateur mesure la position de l'écran par triangulation (spécification de 3 lignes et un point d'intersection) et l'orientation par spécification des axes par l'utilisateur. Il utilise pour ce faire un stylo laser équipé d'un capteur qui sert comme interface pour cette opération (le stylo doit être calibré a priori ou suivant une procédure analogue au calibrage du stylo, section 3.2.5). Cette dernière solution n'a pas été réalisée ni évaluée.

3.2.4. Calibrage casques : optique et vidéo

Méthode

Nous proposons une approche original dans la lignée des travaux d'Azuma [AB94], de Tuceryan [TN00] et de Fuhrmann [FSP99], avec une utilisation de contraintes d'alignements entre des éléments réels et virtuels.

Nous nous intéressons ici à l'estimation des transformations pour la visualisation, et donc principalement aux transformations entre la mesure du capteur (dans le repère Optotrak CS) et l'image perçue par l'utilisateur (dans le repère Image CS). Une question ouverte dans la bibliographie [IHJ03] concerne la position du repère pour l'œil (*eyepoint*) par rapport au dispositif de types casques/lunettes : nous choisissons ici l'entrée de la pupille.

Le principe choisi est une approche interactive dans laquelle *l'utilisateur compense l'erreur d'alignement* entre un modèle virtuel et le modèle réel correspondant, celui ci étant positionné à l'origine. Pour cela l'utilisateur **fige la vue** qu'il a du modèle virtuel fourni par le système à un instant donné. Il déplace l'image par des mouvements de rotation

et translation de la tête l'image, jusqu'à un positionnement perceptuel correct de l'objet virtuel aligné sur l'objet réel. Il peut alors valider son résultat et observer l'alignement sous un autre angle. L'objet positionné sur la table est visible par tous les utilisateur, ce qui permet un calibrage simultané. Formalisons maintenant la méthode.

La transformation finale de restitution d'une image pour l'œil de l'utilisateur (voir figure 5.4), et donc la projection 2D x_i d'un point 3D X_i dans le repère du monde est défini par :

$$x_i = T_{I \leftarrow E} * T_{E \leftarrow H} * T_{H \leftarrow HR} * T_{HR \leftarrow O} * T_{O \leftarrow W} * X_i$$

Dans ce cadre la transformation $T_{O \leftarrow W}$ a été estimée précédemment et la transformation $T_{HR \leftarrow O}$ est connue à chaque pas de temps. La transformation $T_{E \leftarrow H}$ comprend uniquement une translation avec la distance interpupillaire $\pm IPD/2$. Nous considérons cette valeur connue, disponible et spécifiée par l'utilisateur en début de chaque session.

La transformation $T_{I \leftarrow E}$ représente les paramètres intrinsèques du casque qui une transformation projective pouvant se noter $P_{I \leftarrow E}$. Nous considérons que les paramètres de la transformation sont fournis par le fabricant ou sont donnés par l'utilisateur. La seule transformation à estimer reste donc $T_{H \leftarrow HR}$.

Lorsque l'utilisateur fige son point de vue, les points X_i et les points l_i, r_i restent fixes (pour une visualisation stéréoscopique). Ceci induit donc que l'équation (1) peut se ramener à :

$$l_i = P_{I_l \leftarrow E_l} * T_{E_l \leftarrow W} * X_i$$

$$r_i = P_{I_r \leftarrow E_r} * T_{E_r \leftarrow W} * X_i$$

avec

$$T_{E \leftarrow W} = T_{E \leftarrow H} * T_{H \leftarrow HR} * T_{HR \leftarrow O} * T_{O \leftarrow W}$$

et que $T_{E \leftarrow W}$ est égale à la derniere valeur mesurée lorsque l'utilisateur a figé la vue. Sachant que $T_{R \leftarrow O}$ est mesuré à chaque pas de temps, que $T_{E \leftarrow H}$ et $T_{O \leftarrow W}$ sont connus et constants nous en déduisons que :

$$T_{H \leftarrow HR} = T^{-1}_{E \leftarrow H} * T_{E \leftarrow W} * T^{-1}_{O \leftarrow W} * T^{-1}_{HR \leftarrow O}$$

Lors de la phase d'alignement par l'utilisateur, on recalcule à chaque pas de temps $T_{H \leftarrow HR}$. Lorsque l'utilisateur valide son choix (alignement jugée satisfaisant), nous conservons la dernière mesure, ce qui nous donne une nouvelle mesure de calibrage de $T_{H \leftarrow HR}$.

Il existe ensuite différentes stratégies possibles pour obtenir une valeur finale de $T_{H \leftarrow HR}$ avec plusieurs séquences d'alignements. Nous utilisons soit une moyenne pondérée des différentes mesures obtenues, soit de conserver uniquement la dernière mesure effectuée par l'utilisateur.

L'avantage de notre méthode est une approche itérative qui procure un énorme gain d'étapes pour un utilisateur expérimenté, pouvant réaliser le calibrage en 1 ou 2 essais avec une précision suffisante. Nous signalons, que la méthode est efficace si l'utilisateur voit deux ou trois faces simultanéments lorsque il fige la vue : cette solution réduit l'erreur car l'alignement est plus contraint (plus de points, de lignes et de faces alignées). Pour le cas du casque stéréo, la vision stéréoscopique peut aussi favoriser la procédure.

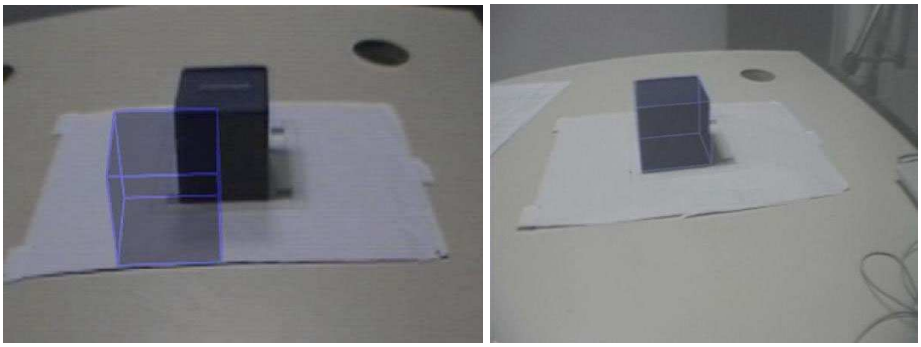
Mise en œuvre

Nous avons choisi d'utiliser la même méthode pour les deux casque optique et vidéo à notre disposition. Nous ne nous reposerons donc pas sur des techniques par vision de type Bajura [BN95] mais nous proposons une méthode originale et unifiée. Pour réduire l'influence des petits tremblements du corps, nous réalisons une moyenne sur 10 mesures au moment de la confirmation d'un bon alignement.

(a) (b)

Fig. 5.6: calibrage du casque et exemple de vue lors de l'expérimentation avec le casque optique : (a) optique avant alignement (b) optique après une phase d'alignement.

La première mise en œuvre a été réalisée avec un capteur magnétique où l'alignement reposait sur l'alignement de l'émetteur magnétique avec un modèle virtuel approximatif identique. Nous considérons dans ce cadre $T_{F \leftarrow W} = Id$. Un exemple est visible figure 5.6.



(a) (b)

Fig. 5.7: calibrage du casque et exemple de vue lors de l'expérimentation avec le casque vidéo : (a) vidéo avant alignement (b) vidéo après alignement.

Pour une seconde réalisation, et le suivi par capteur optique *Optotrak*, nous avons choisi d'utiliser un cube réel, ses paramètres géométriques étant connu par le système. Lors de la mise en œuvre, nous avons constaté que la superposition entre le modèle virtuel opaque et le modèle réel sombre est assez difficile. Nous avons utilisé un modèle virtuel semi-transparent avec des arrêtes sur-imposées («fil-de-fer»). Un exemple de cette procédure avec le casque vidéo est visible figure 5.7.

Évaluation

L'évaluation avec le premier prototype (capteur magnétique) a été réalisée de façon expérimental en mesurant sous différents angles de vue le déplacement d'un pion virtuel sur un damier. L'erreur obtenue est de l'ordre de 0,2 cm à 30 cm, 0,5 cm à 60 cm (distance usuelle de travail), 1,5 cm à 70 cm (limite d'utilisation fiable du capteur magnétique).

Pour le deuxième prototype (capteur optique), nous avons réalisé en premier lieu une évaluation expérimentale à partir du déplacement d'un cube virtuel par rapport à l'origine (figure 5.8) et d'un modèle virtuel par rapport à un cadre réel tracé sur une feuille centrée sur l'origine. L'erreur obtenue est inférieure à 0.5 cm (mesuré entre 20 cm et 80 cm, soit dans un espace utilisateur normal).

Nous avons de plus effectué un test grâce à la méthode d'évaluation proposée par Garrity, Tuceryan et al. [mT+01] qui nous donne une valeur d'erreur planaire. Différents points

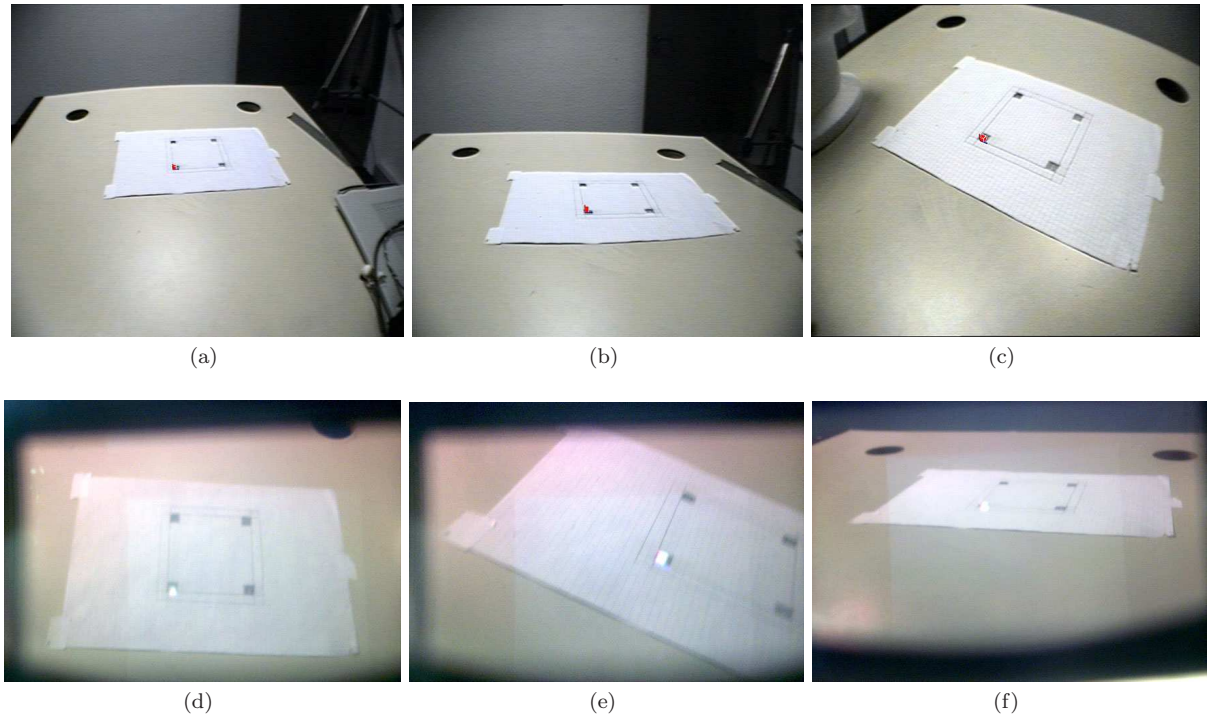


Fig. 5.8: Résultats avec une visualisation sous différents angles d'un cube unitaire de 0.5 mm de longueur positionné à l'origine. (a) (b) (c) avec le casque vidéo (cube rouge) et (d) (e) (f) avec le casque optique (cube blanc).

2D sont choisis aléatoirement sur une grille 2D régulière placée sur la table. La grille n'est pas affichée, mais sa position est connu par le système (dans notre cas centrée à l'origine). Un point de la grille est choisi et est affiché à l'utilisateur à partir des paramètres calibrés (sous forme d'un cube virtuelle). Pour chaque point l'utilisateur change de points de vue (en orientation et position) et marque sur la surface le point observé. Pour cela, il utilise une led du système de suivi qu'il superpose au point visible et fait une validation par un clic souris. Comme on connait la véritable position du point nous pouvons alors mesurer l'erreur. Il est important ici d'utiliser le même système de suivi (pour le calibrage et l'evaluation) pour minimiser les erreurs de transformations des différents repères. La mesure est effectué pour 40 points, avec 3 points de vue par mesure. Les résultats sont visible figure 5.9. Nous obtenons une erreur moyenne de 0,41 cm, bornée par une erreur maximum de 1,12 cm.

Discussion

Dans la plage d'utilisation normale (utilisateur assis), les résultats sont satisfaisants. Si nous réalisons un calibrage debout ou très près du centre les résultats sont alors de qualité moindre. Nous notons que la stratégie de conservation de la dernière mesure est aussi efficace pour un utilisateur expérimenté, le calibrage s'opérant facilement avec une ou deux mesures. Nous noterons au niveau de la mise en œuvre que l'ombre du cube réel fausse la perception de la base du cube : l'éclairage doit être de forte intensité et très diffus pour le calibrage du casque optique.

De même que Fuhrmann [FPTP01], nous observons une large différence entre les systèmes de suivi utilisés. Les premiers résultats avec le système magnétique nous donnait une erreur beaucoup plus importante qu'avec le système optique. Un facteur non négligeable est bien sur l'efficacité de l'utilisateur, reposant sur son aptitude à corréler le bon point de vue et la stabilité du mouvement de la tête.

Cette méthode s'appuie sur le fait de connaître les paramètres intrinsèques ($T_{I \leftarrow E}$) et

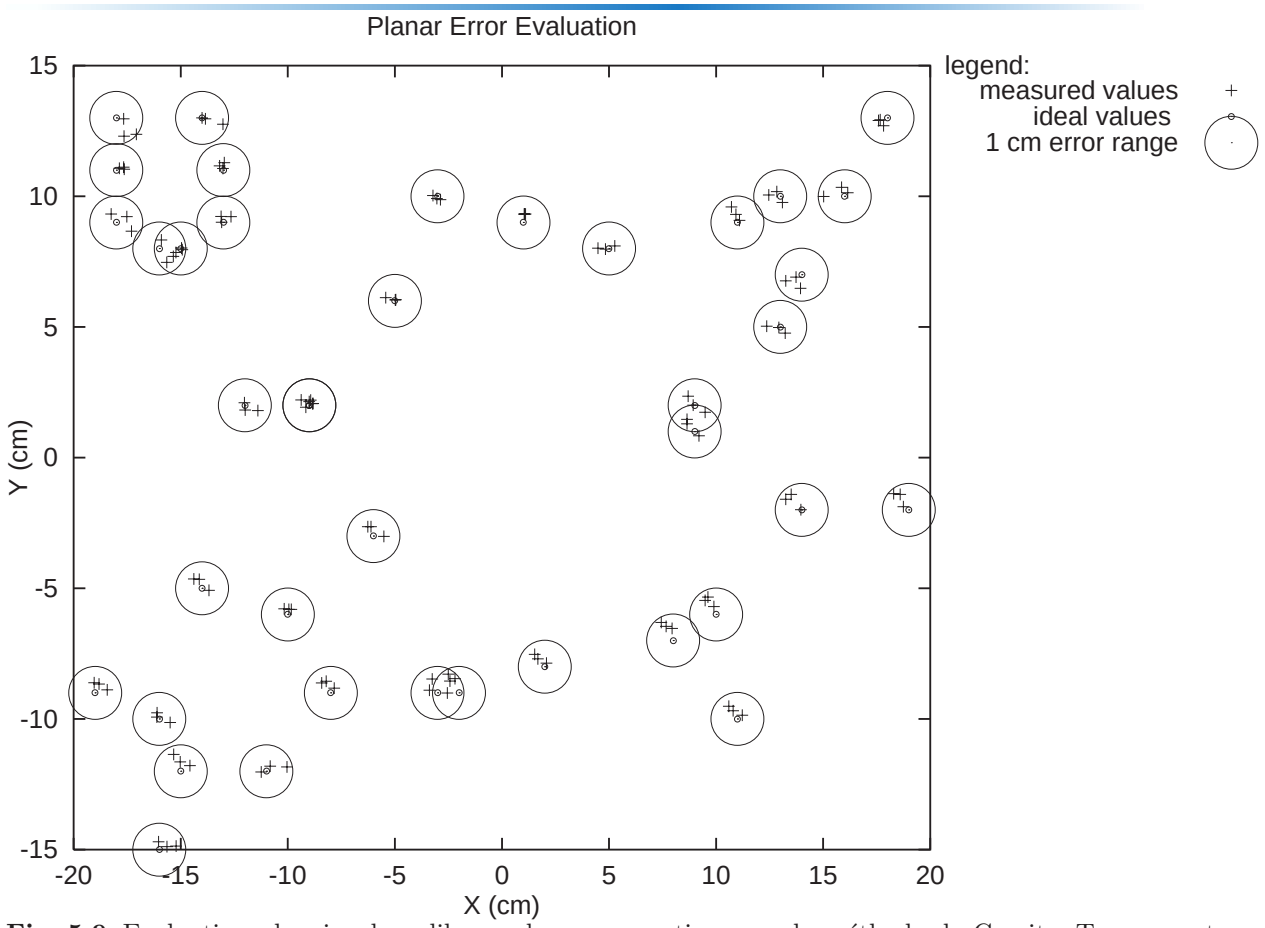


Fig. 5.9: Evaluation planaire du calibrage du casque optique par la méthode de Garrity, Tuceryan et al.[mT$^+$01] : les cercles délimitent une zone d'erreur de 1 cm (le point à l'erreur maximum se trouve dans le premier cercle en haut à gauche).

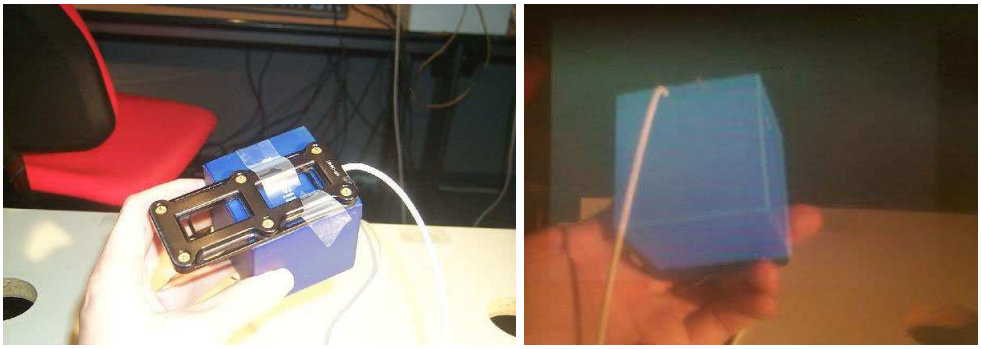
d'avoir un bon positionnement du casque ($T_{E \leftarrow H}$), ce qui n'est pas encore le cas aujourd'hui (calculés dans notre cas par des méthodes ad-hoc). Nous soulignerons aussi que le bon positionnement du cube à l'origine (transformation considérée nulle) influe sur la précision de la méthode.

Extension de la méthode

Récemment Tang [TZO03] a évalué les facteurs ergonomiques liés à différentes techniques de calibrage. Il énonce que l'implication seule du système moteur de mouvement de la tête est beaucoup trop limitant pour la réalisation de calibrage interactif de bonne qualité (bande passante réduite, fatigue). Il considère qu'une préférence doit être allouée à des approches reposant sur la relation main/œil. Dans ce cadre notre méthode ne répond donc pas à ces contraintes (contrairement à l'approche initiale de Fuhrmann [FSP99]).

Nous proposons donc la modification de la méthode précédente par l'*adjonction d'un suivi du cube* (équipé de capteur), permettant alors à l'utilisateur de positionner le cube dans l'espace au dessus de la table et non plus obligatoirement sur la table, réduisant la prise de postures fatigantes et complexes (figure 5.10). Pendant cette phase le cube reste suivi, permettant une fois la position de la tête fixée, de vérifier sous différentes orientations l'alignement correct de l'objet.

Nous notons qu'on obtient des résultats du même ordre avec une utilisation plus pratique tout en conservant une approche sur table. La méthode étant rapide, peu de mouvements de la tête sont engendrés, réduisant une possible fatigue. Mais le problème de bande passante de la tête reste non résolu. Une approche intéressante serait aussi d'inverser le processus : au lieu de compenser par des mouvements de tête, l'utilisateur pourrait



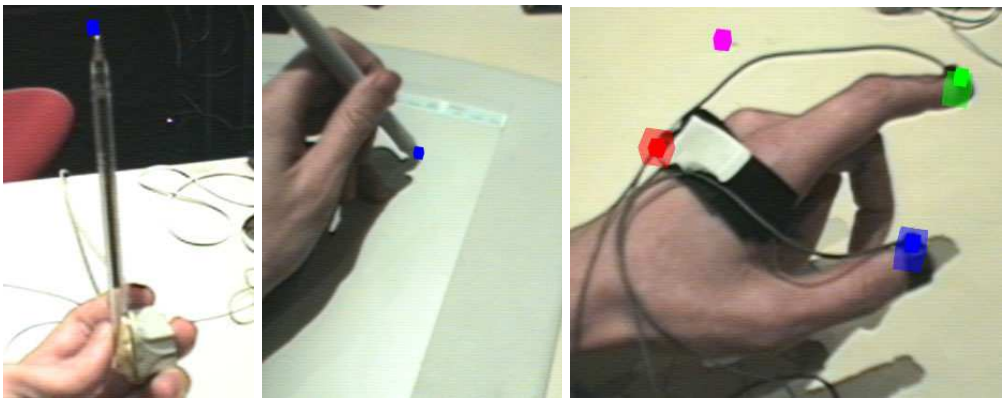
(a) (b)

Fig. 5.10: Extension de la méthode : (a) outil utilisé (b) et résultat.

compenser par des mouvements de la main, la tête restant libre pour observer le résultat : malheureusement les tests effectués avec cette technique sont peu concluant. La tâche revenant à une tâche de «docking 3D» reste fastidieuse et difficile dans ce cadre.

3.2.5. Calibrage des périphériques d'interactions : pointeur et tablette

Nous présentons successivement le calibrage du stylo, pointeur doigt, tablette et le repère locale. Les résultats sont visibles figure 5.11.



(a) (b) (c)

Fig. 5.11: Résultats du calibrage (a) du stylo, (b) de la tablette et (c) de la main.

Calibrage du stylo ($T_{TI \leftarrow SR}$)
L'approche proposée par Tuceryan [TGW+95b] et reprise par Fuhrmann [FPTP01] nous a semblé être intéressante mais pas forcement aisée à réaliser. Il s'agit en effet de calculer la position de la pointe du stylo (donc $T_{TI \leftarrow SR}$) en cherchant le point fixe pour diverses orientation du stylo, la pointe restant immobile. De notre point de vue l'obtention d'un échantillonage regulier semble difficile par l'erreur induite par les capteurs. Nous avons repris l'approche proposée dans notre DEA [Gra99a] : nous utilisons seulement quatre directions discriminantes dans lesquelles l'utilisateur positionne le pointeur, favorisant une approche statique et suffisamment robuste. Les quatre directions privilégiés sont définies par les points cardinaux. Pour plus de détails voir les résultats obtenu lors du DEA [Gra99a]. L'erreur étant inférieure à 0,5 mm.

Calibrage du pointeur doigt ($T_{FI \leftarrow FR}$)
La main est équipé de trois LEDs, mesurant trois positions : l'extrémité de l'index, du pouce, et le dessus de la paume. Contrairement à Veigl et al. [VKL+02] qui fixe de façon empirique $T_{FR \leftarrow FI}$, ou Ikits [IHJ03] qui l'évalue à partir d'une mesure d'un autre capteur,

nous choisissons une mesure par calibrage à une position initiale. Nous avons retenus deux postures que l'utilisateur doit réaliser avec la main.

La première consiste à positionner la main, doigts joints, à l'origine du repère du monde (index à l'origine et main dans une direction spécifique). Nous en déduisons $T_{FI \leftarrow FR}$ pour l'index et le pouce, c'est à dire principalement la différence de hauteur entre le capteur et le centre des dernières phalanges des doigts ou le centre de la main. Cette posture nous sert aussi pour une utilisation dans un cadre de reconnaissance de geste (nous mesurons ici la distance relative entre chaque doigts et l'orientation initiale de la main). La deuxième posture est utilisé uniquement pour la reconnaissance de geste. Cette posture elle la même que précédemment avec le pouce ecarté. Ces deux postures pourront être utilisable avec un système de reconnaissance de geste telle que proposé par Joseph [JZ99].

Calibrage de la palette digitale ($T_{W \leftarrow TA}$)
Nous considèrons la tablette en position statique durant les sessions et posée à plat sur la table. Nous la calibrons simplement en positionnant l'élément rigide du capteur de suivi sur l'origine de la tablette. Les paramètres intrinsèques de la palette (résolution, taille, positionnement du pointeur) sont obtenus à partir des données constructeur. La composition de transformation des paramètres externes avec celle des paramètres intrinsèques nous donne alors $T_{W \leftarrow TA}$.

Calibrage des objets réels ($T_{W \leftarrow RO}$)
Nous introduisons plusieurs méthodes complémentaires pour calibrer des objets réels ($T_{W \leftarrow RO}$, $T_{OR \leftarrow OR}$), utilisables comme éléments ou supports d'interaction (*props*). La présentation est réalisée plus en détails dans la section 4.

3.2.6. Calibrage du repère locale ($T_{W \leftarrow L}$)

L'utilisateur peut avoir besoin de spécifier un sous-espace de l'espace réel pour une application spécifique. Il utilise pour cela un widget 3D (cube virtuel avec deux points interactifs) qu'il positionne pour définir l'espace de travail de l'application ($T_{W \leftarrow L}$). Le widget est contraint à rester posé sur la table (3 DDL). L'utilisateur peut en complément modifier sa taille (3 DDL). L'utilisateur effectue cette séquence à l'aide du casque suivi et du pointeur déjà calibrés.

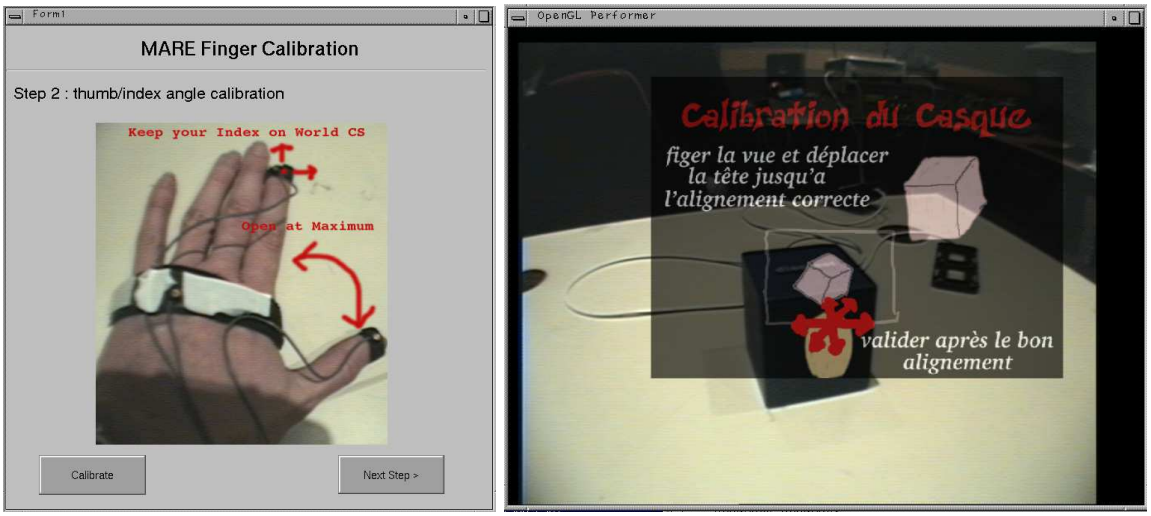
3.2.7. Interface

L'utilisateur peut à l'aide d'un dispositif par défaut (souris) valider et avancer dans les différentes actions, l'interface se présente sous une succession d'écrans de contrôle très similaires aux interfaces de type *Wizzard* (figure 5.12). Chacun des outils peut être lancé séparément, mais aussi à travers une interface qui permet de calibrer entiérement le système. Les résultats obtenus sont sauvegardés au format XML et pouvent ainsi être chargés par une application dans la suite.

3.3. Résultats et Discussion

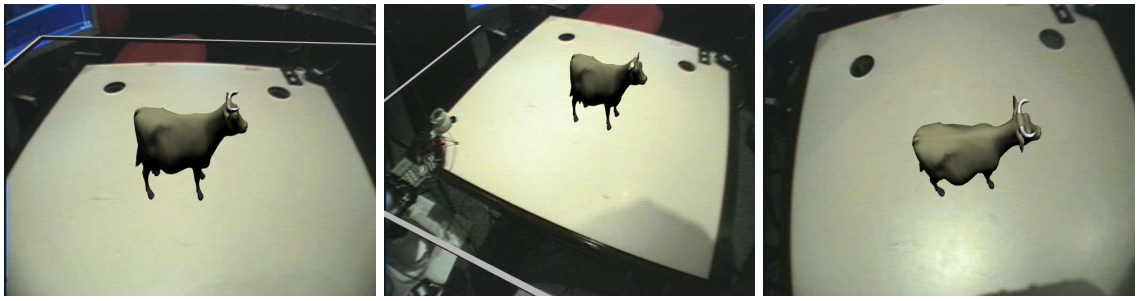
La faible qualité des casques limitent fortement la mise en œuvre de ces méthodes. L'erreur finale obtenue est de l'ordre de 0.5-1 cm (figure 5.13). La plage d'erreur se situe dans le même ordre que les techniques de la même catégorie (canevas complet et rapide). L'ensemble de la procédure dure moins de 40 minutes pour utilisateur novice et 30 minutes pour un utilisateur expert, ce qui nous semble très raisonnable.

Il nous semble qu'une piste intéressante serait une étude plus précise de la méthode de calibrage du casque, en identifiant par exemple des positions spécifiques générant un ensemble de contraintes qui peuvent être facilement fusionnées pour obtenir un résultat de bonne qualité. De plus, pour les casques vidéos ou les casques optiques complémentés de



(a) (b)

Fig. 5.12: Photos d'écrans de l'interface de calibrage pour utilisateur novice : (a) interface graphique écran pour calibrage main (b) et interface RA pour calibrage casque.



(a) (b) (c)

Fig. 5.13: Résultat final du calibrage avec un modèle vu sous trois points de vues (a) (b) et (c).

caméras, des techniques basées sur la vision, apporteraient un calibrage plus automatique, plus précis et minimiseraient les tâches utilisateur.

Nous résumons brièvement dans le tableau 5.3 les propriétés de notre approche (critères similaires à Summers [SBC+99]). Notre méthode offre les même critères ergonomiques que Fuhrmann mais dédiée à une configuration sur table et dans un cadre multi-utilisateurs. L'interface que l'on propose offre des fonctionnalités plus complètes (mode RA/GUI, outils de suivi). Par rapport au résultat de Summers (moyens matériels analogues), notre système offre des résultats d'aussi bonne qualité.

Nous noterons que la limitation observée lors des tests est l'absence de continuité temporelle de l'alignement avec des sauts et décalages lors des déplacement de la tête : l'étude et la réduction de cette erreur constitue la suite de ce chapitre.

4. Erreur dynamique : réduction de la latence

Holloway [Hol95a] démontre que la latence est la cause principale d'erreur dynamique. Une optimisation des traitements du système n'est pas suffisante pour l'éliminer et il préconise l'utilisation de techniques efficaces pour la minimiser, telle que les algorithmes de prédiction. Holloway définit la latence totale (*end-to-end latency*) comme une addition des temps de retard de chacun des éléments suivants : capteur, ordinateur, génération d'image, synchronisation vidéo, tracé vidéo, dispositif d'affichage.

Tab. 5.3: Listes des caractéristiques de nos méthodes

Critères	Valeur
Flexibilité	chaque méthode peut facilement être substitué par une nouvelle approche.
Précision	de l'ordre de 0.5 mm
Répétabilité	supporté sequentiellement et parallèlement pour certaines phases
Robustesse	dépendant de l'utilisateur (précision, tremblement). sensible aux occultations du capteur.
Facilité d'utilisation	plusieurs types d'interfaces guident l'utilisateur.
Indépendance	indépendances entre chaque catégorie (dispositifs, interface, visualisation) mais dépendance avec les dispositifs de suivi.
coût	trés faible hors considérations des dispositifs physiques.
avantages	facile à mettre en œuvre, intuitif, rapide.
inconvénients	trop forte dépendance utilisateur (besoin approche semi-automatique). nécessite paramètres intrinseques. problème de la bande passante de la tête pour le calibrage du casque.

Le lecteur trouvera de plus amples informations sur l'état de l'art dans Grasset [GG03]. Nous reprendrons les approches classiques qu'on adaptera à notre configuration, la contribution étant ici plus une étude dans le cadre de la configuration (sur table) que nous avons choisi.

4.1. Approche : pipeline de haute qualité et algorithme prédictif

Nous avons choisi ici une minimisation reposant sur deux approches :
– réduction de la latence du canevas logiciel ;
– prédiction par Filtre de Kalman [Azu97] ;

Estimation de la latence : nous réalisons une étude sur les différents composants de notre système à partir des valeurs fournies par le système d'évaluation de performances sous *OpenGL Performer*. Nous nous baserons aussi sur l'étude de Swindells et al. [SDB00] qui fournit des résultats quantitatifs pouvant être utilisés dans notre contexte. Nous considérons ici la latence depuis le déplacement de la tête jusqu'au retour visuel (figure 5.14).

Nous définissons alors empiriquement la latence de notre système :
– temps de mesure du capteur (T_{track}) : comprend le temps de mesure et de transfert sur l'Onyx. D'après l'étude de Swindells nous pouvons l'estimer ici à 10 ms ;
– temps de transmission et d'accés au valeur ($T_{offhost}$) : le système fonctionnant en SCSI, une synchronisation bas niveau permet d'accéder rapidement aux valeurs, cette valeur est incorporé dans le temps suivant ;
– temps de calcul (T_{comp}) : temps entre l'obtention de la valeur, calcul de la nouvelle vue et demande d'affichage au système : estimé aussi par *OpenGL Performer* à 7 ms pour une application «à vide» ;
– temps de rendu (T_{render}) : fourni par *OpenGL Performer* il comprend le temps de filtrage par le des données (*culling*) et de rendu. Sur une scène très simple nous obtenons 2-3 ms par œil ;
– temps d'affichage ($T_{display}$) : le signal est alors converti au format PAL, ce temps de traitement est considéré comme négligeable. Comme nous utilisons de la vidéo entrelacé

à 50 Hz (25 Hz pour un affichage complet), le temps d'affichage est de 40 ms (affichage du dernier pixel sur le deuxième écran) ;
– temps de synchronisation ($T_{synchro}$) : Il définit le temps d'attente pour le changement de buffer synchronisé avec le système d'affichage (*swap buffer*) et les différents autres temps intermediaires de synchronisation entre éléments. Il est d'en moyenne 20 ms (moitié du tracé de l'image) et au maximum 40 ms.

Nous obtenons une latence moyenne d'environ 80 ms avec un maximum de 100 ms.

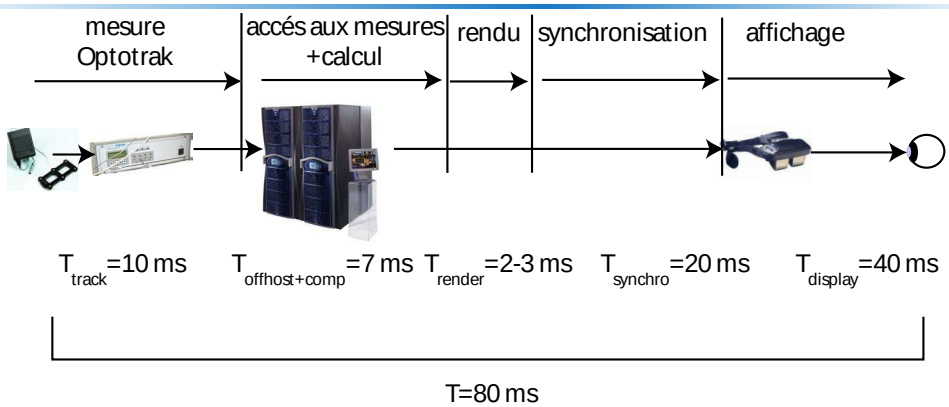


Fig. 5.14: Représentation des différentes étapes de l'affichage et évaluation empiriques des temps de traitement.

Ce temps de latence pourrait être réduit à 80 ms par une optimisation de la boucle système en réduisant le temps de synchronisation grâce à un délai ajustable inséré avant la mesure.

Prédiction : nous avons expérimenté un filtre de Kalman discret avec vitesse constante pour le suivi du casque et des différents périphériques. Suite aux résultats obtenus avec un capteur magnétique suivi dans nos précédents travaux [Gra99a] nous avons repris la technique pour le système de suivi optique.

4.2. Résultats et discussion

D'un point de vue perceptuel la mise en œuvre de ces éléments réduit la sensation de retard, tous en ne supprimant pas les problèmes d'instabilité durant un mouvement de la tête.

Le gain du parallélisme se fait principalement ressentir sur des grosses scènes sinon il reste négligeable, principalement conditionné par la puissance de la station de travail utilisée.

Nos résultats confirment les résultats obtenus par Azuma [AB94] : difficulté de paramétrer le modèle du filtre, oscillations sur le bruit. Nous observons un fort gain sur la latence au prix d'un bruit croissant. Si nous minimisons ce bruit, nous observons un dépassement en fin de mouvement. La figure 5.15 montre les résultats obtenus avec le filtre sur une prédiction à 70 ms.

Il aurait été aussi intéressant de réaliser une estimation plus précise de la latence de notre système avec une technique similaire à Swindells [SDB00].

5. Approche globale : changer l'environnement

Au dela de l'alignement entre réel et virtuel, le mixage des deux mondes peut aussi nécessiter l'ajout d'objets réels (e.g. maquette, tasse, document papier). En effet, la possibilité d'ajouter de façon simple et efficace des objets dont on connaît un modèle numérique peut être une fonction intéressante pour notre environnement.

5.1. Problème

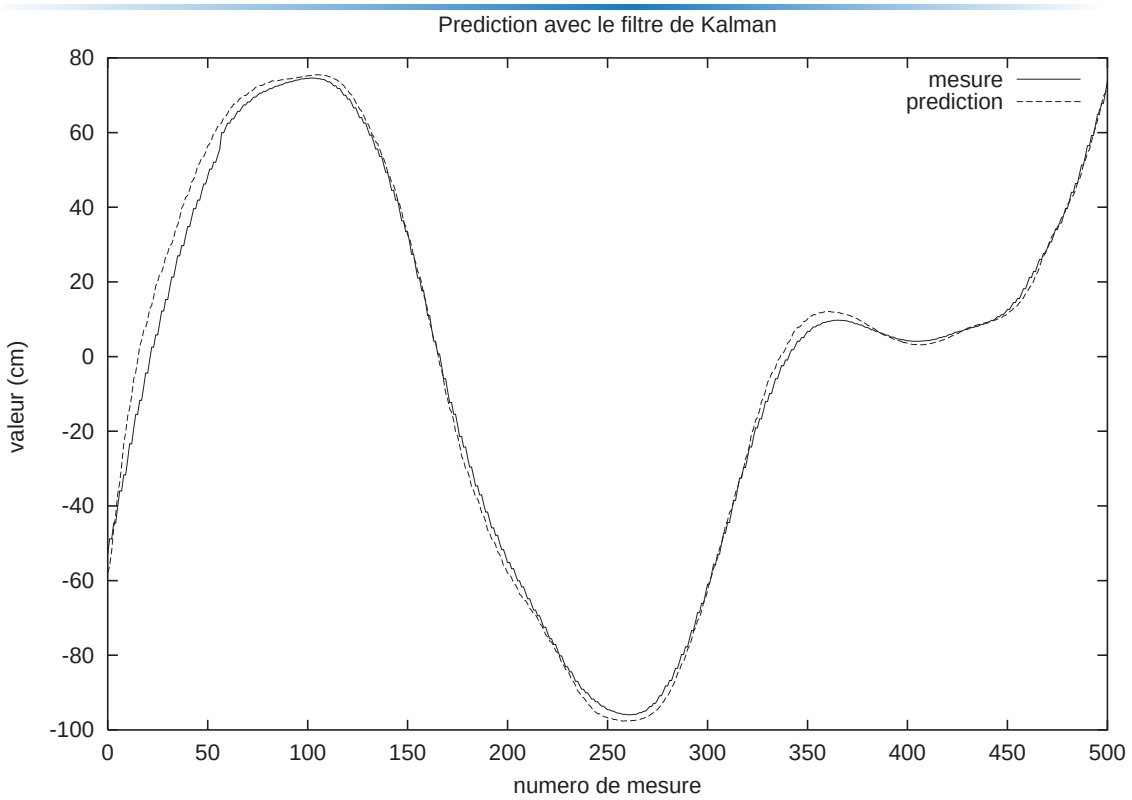


Fig. 5.15: Evolution d'un mouvement de la tête (composante Y) et sa prédiction à 70 ms avec un filtre de Kalman. On note le retard qu'a le filtre sur les changements de mouvements.

Pour pouvoir utiliser un objet réel avec un système de RA, il faut fournir les caractéristiques de cet élément au système. Nous avons retenus *l' identification/la classe de l'objet, sa géométrie et sa position spatiale* (figure 5.16).

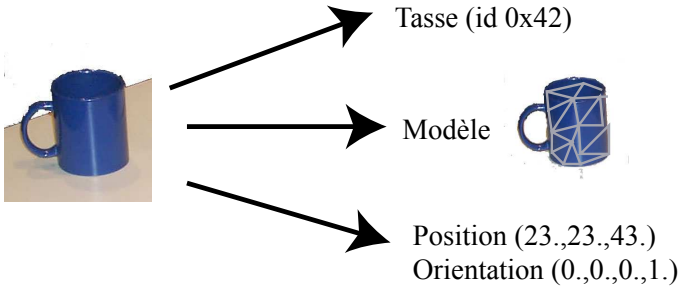


Fig. 5.16: Critères minimum pour l'ajout d'objets à notre environnement : forme (et apparence) et position.

Cette problématique se ramène alors à la mise en œuvre de technique pour l'alignement d'objets dans le monde réel dont le mouvement peut être suivi. Peu des travaux en RA ne proposent la possibilité d'ajouter un objet 3D durant la session.

5.2. Approche

Nous nous sommes concentrés sur l'ajout dynamique, c.-à-d. l'ajout et la spécification des caractéristiques au cours d'une session. Cette approche est motivée par deux types d'applications :

- l'utilisation de *tangibles user interfaces* qui sont associés à du contenu ou des outils virtuels ;
- l'ajout d'éléments de décors réels à une scène virtuelle. Cette application semble no-

tamment intéressante pour l'architecture dans la mesure où elle permet l'intégration à une maquette virtuelle de maquettes réelles, arbres, pierre, etc.

Nous avons choisi d'aborder cet objectif selon une approche interactive, celle-ci semblant permettre le plus de flexibilité. De notre point de vue la reconnaissance automatique de modèle 3D reste encore un problème non résolu. Nous présentons nos choix pour l'obtention des différentes caractéristiques d'un objet réel.

Obtention de la géométrie : Suivant le type d'application nous considérons le modèle de l'objet réel connu ou non. Nous disposerons parfois directement du modèle 3D de l'objet réel (application de CAO, par exemple). Pour les autres cas, nous introduirons une méthode ad-hoc permettant d'obtenir une représentation simpliste du modèle (voir section 6).

Obtention de la position : l'approche choisie repose sur *une mise en correspondance manuelle sous contraintes*, entre le modèle réel et sa représentation virtuelle. Cette méthode est souvent utilisée en vision comme procédure d'initialisation dans l'espace image. Elle ici réalisée dans l'espace 3D. L'intérêt est de pouvoir vérifier l'alignement sous différents angles de manière à réduire l'erreur par rapport à un alignement fait en 2D. On propose alors deux techniques à partir de ce concept de mise en correspondance.

Technique 1 : une représentation virtuelle de l'objet est affiché à une position prédéfinie dans le monde réel. L'utilisateur doit alors superposer l'objet réel à cette représentation. L'utilisateur valide la manipulation dès qu'il est satisfait.

Cette approche, de type spatial, nécessite un espace libre devant l'utilisateur pour positionner la représentation virtuelle : on utilisera principalement l'aire partagée pour un ajout avant la session et la zone personnelle de l'utilisateur pour un ajout pendant la session. Cette approche peut aussi être utilisée dans le cadre d'ajout d'un objet dont le mouvement est suivi avec un capteur attaché rigidement à cet objet. En effet, une fois l'alignement effectué on connaît la position du centre de l'objet et la transformation entre le capteur et le centre de l'objet peut donc être déduite.

Technique 2 : contrairement à la première approche, on va ici aligner le modèle virtuel sur l'objet réel. Cette approche est constitué des étapes suivantes :

1 Placer un modèle réel dans l'espace de reconnaissance.

2 Obtenir un modèle virtuel de l'objet.

3 Aligner le modèle virtuel sur le modèle réel ce qui nous permet d'en déduire sa position.

La phase d'alignement de l'objet nécessite l'obtention des 6 DDL de l'objet réel (position et orientation, le modèle réel étant naturellement «à l'échelle»). Dans notre contexte nous pouvons nous ramener à un cas plus simple à partir de quelques remarques :

- d'après les lois de la gravité, les objets réels reposent nécessairement sur un support physique. Chaque élément étant positionné sur la table (ou sur un support en relation avec la table de façon transitive), on pourra toujours obtenir la position de ce support ;
- on considère que les objets se placent debout (axe vertical prédéfini).

La métaphore se réduit alors à la spécification de 3 DDL : 2 DDL position, 1 DDL de rotation. Il nous semble que ce choix est assez général, les cas dégénérés (e.g surface sur un pic) n'étant que peu présents dans des applications typique (jeu, architecture, planification urbaine etc.). La technique 2 se ramène alors à deux étapes : une étape d'alignement 2D en position (la troisième dimension obtenue automatiquement) et un alignement 1D en rotation.

Quel type d'interface peut alors être utiliser pour réaliser cet alignement ? Dans Grasset [Gra99a] nous considérions l'alignement par arête ou par boîte englobante. Nous proposons ici une nouvelle approche qui nécessite uniquement de pouvoir sélectionner un point

quelconque du modèle virtuel ; par contact avec celui ci, la transformation rigide déduite contraint le placement.

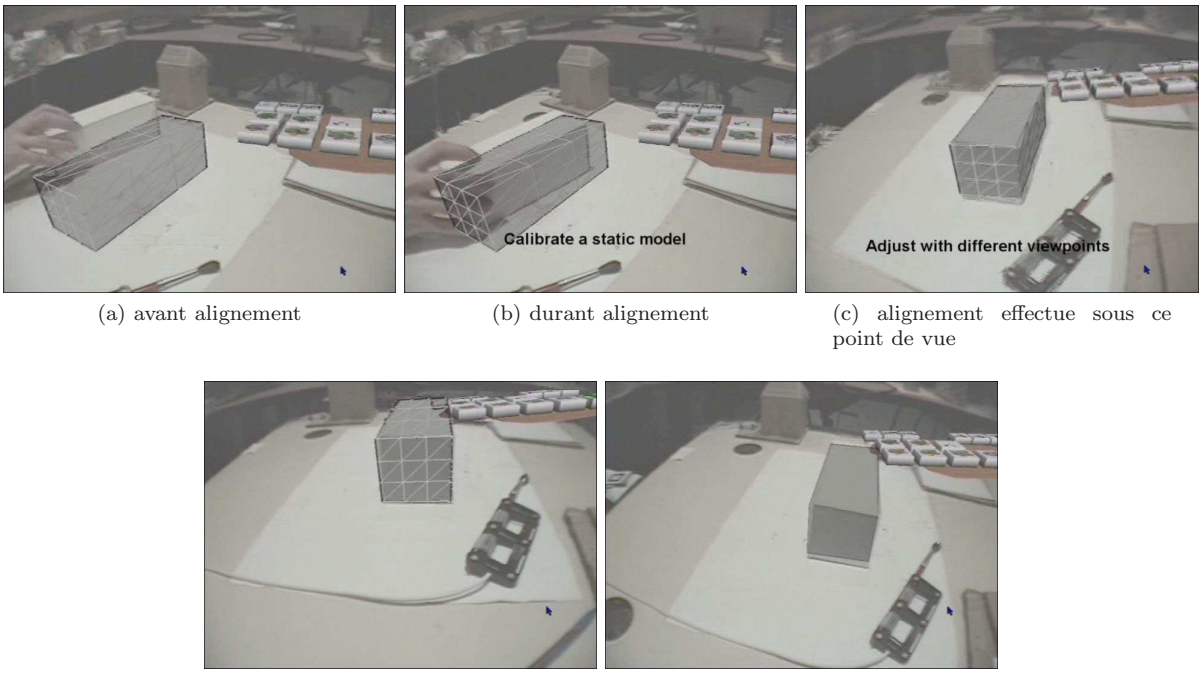
Pour finir on retiendra que les deux techniques peuvent être étendus par une méthode ad-hoc pour le suivi de ces objets. Une fois la position d'un objet réel définie, l'utilisateur peut y fixer dynamiquement un capteur. On utilise une détection de collision qui permet alors de connaître la position du contact entre les deux objets (capteur et objet réel). Nous obtenons alors automatiquement la transformation correspondante.

5.3. Mise en œuvre

Nous avons réalisés les deux techniques. La première technique a été réalisé dans notre architecture «standard» à l'aide du système de suivi optique. Cette technique a été reprise dans le cadre d'initialisation statique pour le système de peinture présenté chapitre 8. La deuxième a été réalisée à partir de notre première version de notre architecture. L'interface reposait sur l'usage du système magnétique et du clavier.

Pour l'alignement dans les deux techniques, on préconise un mode d'affichage fil de fer facilitant la mise en relation visuel sur les arêtes de l'objet quand il y'en a. Dans le cas contraire il semble évident que la texture joue alors un rôle essentiel : un affichage par transparence aidera alors à l'alignement.

5.4. Résultats, évaluation et discussion



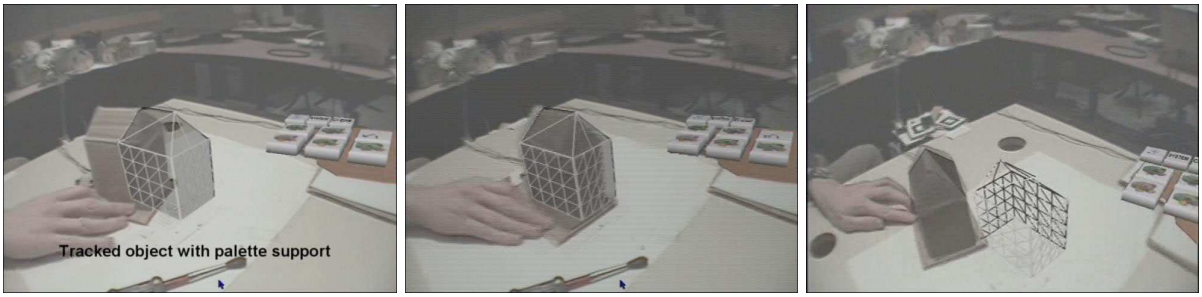
(a) avant alignement (b) durant alignement (c) alignement effectue sous ce point de vue

(d) alignement sous un autre point de vue (e) résultat sous un troisième point de vue avec affichage du fantôme

Fig. 5.17: Ajout d'objet avec modèle prédéfini avec la première approche. Le modèle réel est une boite en carton, le modèle virtuel sa représentation en fil de fer positionnée à l'origine.

La figure figure 5.17 montre le résultat avec le premier prototype de la technique 1. La figure 5.18 montre le résultat avec un objet équipé d'un capteur. La figure figure 5.19 montre le résultat avec le premier prototype de la technique 2. Le système est fiable, facile à utiliser avec une erreur visuel inférieur à 5 mm (évaluation visuel et empirique).

On évalue le résultat d'un ajout d'objets réels à l'aide de test d'occultations pour les deux techniques. On associe un objet «fantôme» correspondant à l'objet réel. On utilise des



(a) avant l'alignement (b) durant l'alignement (c) après l'alignement : le fantôme est superposé à la maquette

Fig. 5.18: Ajout d'objet avec modèle prédéfini avec la première approche, pour un objet suivi. Le modèle réel est une maquette de maison, le modèle virtuel sa représentation en fil de fer positionnée à l'origine. le suivi est par capteur optique.

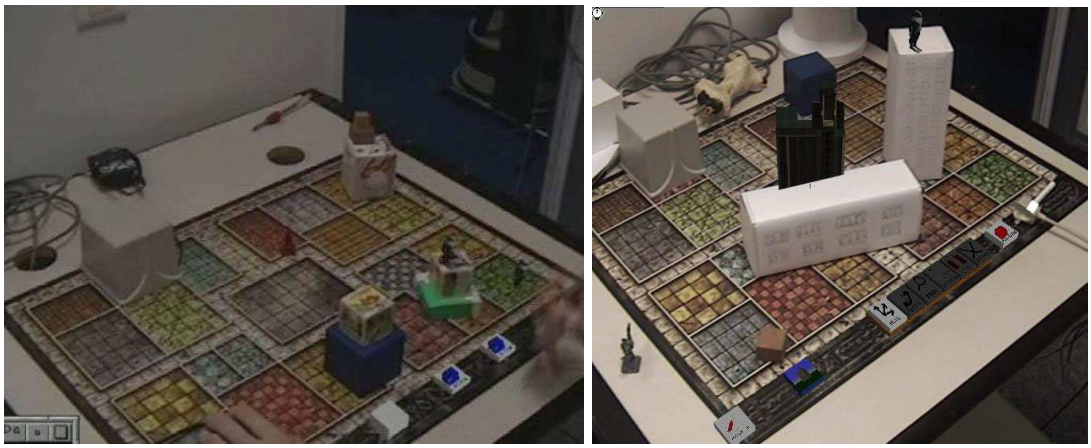
(a) alignement en position (b) confirmation de fin d'alignement en position.

(c) alignement en rotation. (d) confirmation de fin d'alignement en rotation.

Fig. 5.19: Ajout d'objet avec modèle prédéfini dans le premier prototype. Le modèle réel sont deux boîtes superpositionnés. Le modèle virtuel utilisé est affiché en bleu.

objets virtuels que l'on intègre avec le modèle réel pour étudier la cohérence d'occultation entre les éléments (figure 5.20). On note un décalage faible mais suffisant pour gêner une intégration parfaite.

6. Premier pas vers une digitalisation interactive du réel par esquisse



(a) l'utilisateur interagir avec l'élément en se servant comme décor, placement d'un personnage virtuel

(b) occultation pour une scène architecturale

Fig. 5.20: Résultat avec des intégrations d'objets réels et virtuels.

6.1. Introduction

Le cas précédent considérai la connaissance de la géometrie de l'objet. Nous proposons dans cette dernière section nos premiers travaux pour pouvoir ajouter dynamiquement un objet de *géométrie inconnue* en cours de session (boîte, canette, livre, sculpture, etc.).

Nous restreignons le cadre d'étude à notre contexte de table : acquisition d'objets de petites tailles et pouvant être ajoutés durant une réunion. Nous nous intéressons donc à une méthode rapide, intuitive et dynamique pour acquérir le modèle d'un objet réel.

(a) (b) (c)

Fig. 5.21: Trois approches pour la digitalisation d'objets réels : (a) manuelle (b) automatique (c) et semi-automatique (logiciel *Canoma* [Can].

L'acquisition d'un modèle nécessite l'obtention de la géométrie de l'objet (ici sous forme polygonale), et l'obtention de la réflectance de l'objet (ici sous forme de texture et matériau de type Phong). On peut différencier trois types d'approches (figure 5.21) pour l'acquisition de modèle réel (d'après Lee [LHS01]) :

- *acquisition manuelle* : principalement réalisée à l'aide d'un outil de type bras articulé, l'utilisateur fait l'acquisition point par point de l'objet réel ;
- *Acquisition automatique* : elle repose sur l'utilisation de périphériques physiques (lasers) et d'algorithmes de reconstruction qui fournissent automatiquement le modèle ;
- *Acquisition semi-automatique* : L'utilisateur définit un certain nombre de contraintes à l'aide d'un dispositif et un processus fournit une solution qu'il peut alors raffiner suivant le niveau de souplesse de l'application.

Souhaitant se limiter à des solutions peu coûteuses, flexibles et avec un faible temps de

traitement, l'approche semi-automatique nous semble la plus avantageuse. Dans ce but, nous avons expérimenté les différents outils professionnels par modélisation basée image : *Photo Modeler*© [Modb], *RealViz ImageModeler*© [Ima], *MetaCreations Canoma*© [Can]. Malgré la simplicité d'utilisation, ces outils sont peu efficaces lorsque l'on souhaite acquérir simplement et rapidement un modèle peu complexe (sans tenir compte du temps de prise d'images du modèle). Dans le cas d'objets courbes ils sont généralement peu souples.

Nous proposons ici une approche semi-interactive 3D qui fonctionne dans l'espace objet. Cette solution reposera sur notre compétence naturelle de perception 3D et l'utilisation d'un système de RA. Celui-ci nous donne accès à la géometrie mais aussi aux propriétés de réflectance. Nous présentons au préalable les travaux relatifs aux approches semi-interactives de digitalisation.

6.2. Approche : vers une esquisse du réel en RA

L'approche que l'on a retenu est dans la même esprit que Han [HM97], *esquisse de réel*, mais ici à partir d'un système interactif de RA comme Lee [LHS01]. Cette solution offre un contrôle fort et une forte adaptation au modèle. Un système de RA autorise l'usage d'algorithmes de vision (par l'image d'un casque semi-transparent vidéo), un espace 3D d'action (comme Agrawala [ABL95]), un espace de superposition de la reconstruction (tel celui proposé par Raskar [RLW01]) et une approche tangible. En effet, l'objet à acquérir est alors facilement manipulable (tel celui décrit par Lee [LHS01]).

Les approches par digitalisation manuelle posent généralement des difficultés dans le cas de tracé de ces arêtes : Han l'énonce bien dans son article [HM97] : «l'être humain a plus de facilité à tracer sur des surfaces planes que définir arrête et coin 3D». Contrairement à Han, nous avons choisi un découpage en trois classes de géométries d'objets (figure 5.22) : forme polygonale (e.g. boîte, plan), forme courbe (e.g. bouteilles, toiture) ou maillage quelconque (e.g. sculpture, jouet).

(a) (b) (c)

Fig. 5.22: Décomposition en trois ensembles : (a) polygonale (b) courbe et (c) maillage quelconque (c).

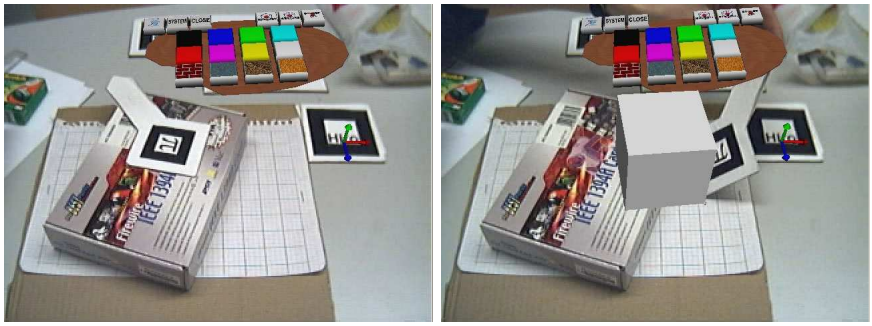
Nous présentons dans la suite les premiers résultats obtenus pour chaque catégorie et une discussion des limitations des solutions.

6.2.1. Modèle polygonal

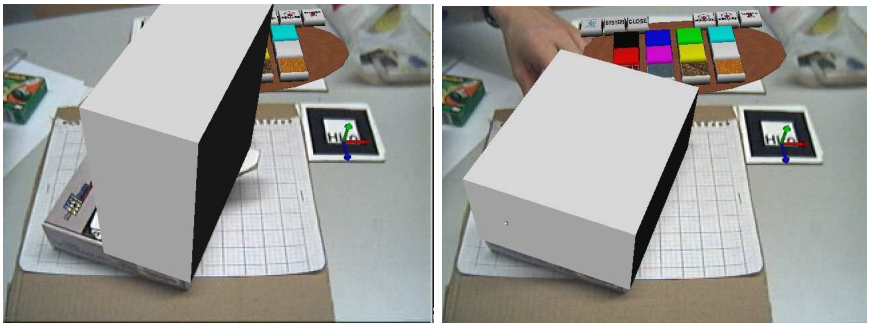
Ce cas s'inscrit dans une généralisation des premiers travaux de Grasset [Gra99a]. On considère dans cette catégorie tout objet se rapportant à des primitives simples (cube, cylindre, tétraèdre) mais aussi des objets définissables par un ensemble de plan.

Nous nous basons sur les caractéristiques géométriques définissant les règles de construction de l'objet. L'approche consiste alors à les reproduire sur l'objet réel par superposition interactive de la construction virtuelle de l'objet (figure 5.23, figure 5.24). Notre prototype

fonctionne actuellement sur les primitives suivantes : plan, cube, cylindre, sphère, cône, pyramide, tétraèdre. Pour l'apparence, on utilise une méthode ad-hoc similaire à celle de Lee, par extraction automatique de la radiance à partir de la caméra.



(a) (b)



(c) (d)

Fig. 5.23: Esquisse d'un modèle polygonal par primitive parallélépipédique.

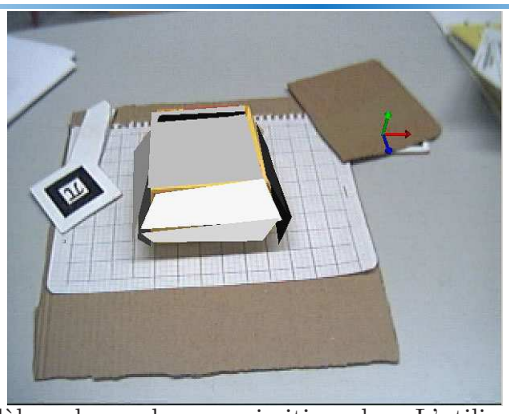


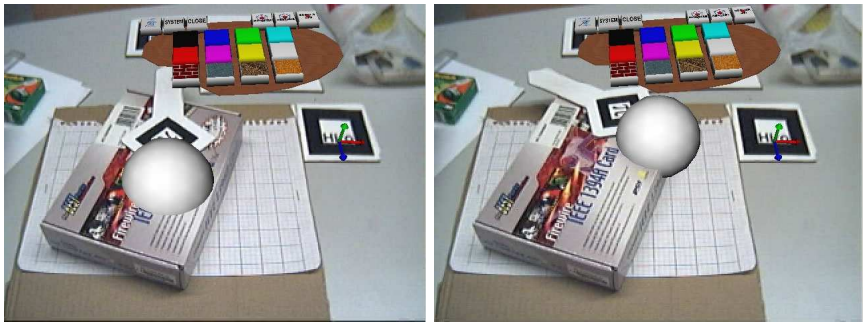
Fig. 5.24: Esquisse d'un modèle polygonale par primitive plan. L'utilisateur construit des plans indépendant qu'il peut aisément connecter.

Le modèle généré nous fournit alors un modèle initial de résolution satisfaisante pour différents contextes : gestion des occultations avec des objets virtuels figure 5.25), gestion des ombres, étiquette et détection de collision.

6.2.2. Modèle courbe

On a repris le concept d'outils haut niveau présenté dans Fiorentino[FdAMS02] pour l'esquisse d'objets virtuels courbes. On réutilise alors des méthodes telles que *skinning*, *extrusion* ou surfaces de révolution pour définir la surface d'un objet réel.

La méthode de *skinning* utilise celle de la librairie ACIS [ACI]. Les autres techniques sont implémentées par les algorithmes standards [FPH+94]. Un résultat est visible figure 5.26.



(a) (b)

Fig. 5.25: Résultat pour l'esquisse sur modèle polygonal avec la gestion des occultations. Un modèle fantôme est associé à la boîte sur les figures (a) et (b) et une intersection est faîte avec une sphère virtuelle.

Fig. 5.26: Esquisser un modèle courbe avec un outil de surface de révolution.

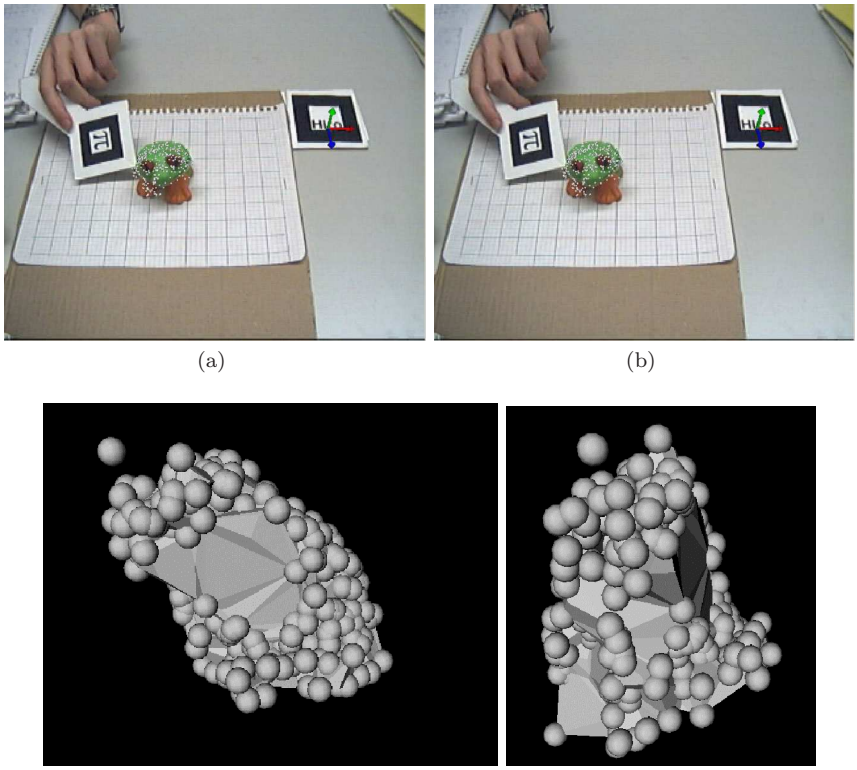
6.2.3. Modèle quelconque

Ce cadre définit les modèles non assimilés aux deux premières catégories. On introduit deux approches.

La première technique utilisée est un échantillonage interactif associé à un algorithme de reconstruction de surface. Contrairement aux outils d'acquisition manuelle, l'utilisateur a un retour visuel de la digitalisation directement sur la surface. Pour ce faire, l'utilisateur déplace un pointeur sur la surface de l'objet, ses différentes positions sont alors enregistrées. Pour la reconstruction on a retenu l'algorithme *Power Crust* [ACK01] offrant les meilleures performances en terme de gestion d'un échantillonnage irrégulier, de rapidité et de disponibilité. Nous notons que l'utilisation de l'algorithme de Hoppes [HDD+92] nécessite un maillage trop régulier, une forte résolution (divergence sur nos modèles de test).

L'expérimentation a été réalisée sur un modèle de grenouille en plastique offrant de bonnes caractéristiques de test : objet de petite taille, des détails à différents niveaux, une surface courbe et lisse, différents types de courbures et de concavité. Pour la mise en œuvre, nous avons choisi un affichage des points échantillonnées sous forme de sphère. Ce choix permet d'avoir un bon retour visuel à l'utilisateur sur la position et la densité de points positionnés. Les résultats sont visibles sur figure 5.27.

En analyse, la reconstruction respecte la forme globale mais est de très faible qualité (erreur et imprécision). Au vu des résultats, l'usage de notre solution se limite à un usage où l'utilisateur ne désire pas un modèle parfait (décor de maquette mixte, TUI).

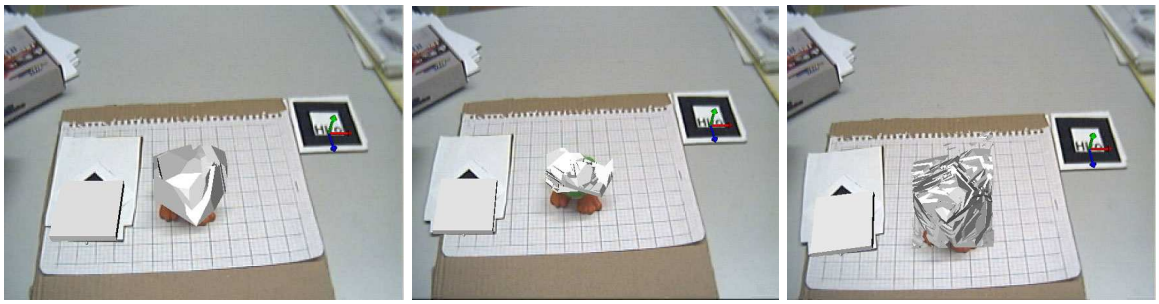


(a) (b) (c) (d)

Fig. 5.27: (a) (b) esquisse d'un modèle quelconque (c) résultat de la reconstruction sous deux points de vue.

La deuxième technique repose sur le concept de *sculpture* : nous choisissons de supprimer la matière superflue d'un bloc de matière virtuel dans lequel est plongé un objet. Pour ce, l'utilisateur peut creuser cette matière à l'aide d'outils tangibles spécialisés. De cette façon, il peut mimer le travail d'un artiste avec de la matière brute (sculpture sur pierre, sur bois).

Ce concept a été mis en œuvre à l'aide de la librairie ACIS, à partir d'opérations CSG entre l'outil et la matière. On a réalisé des tests à partir du même modèle que précédemment avec des outils d'extraction sous la forme d'une palette. On observe des résultats plutôt de mauvaise qualité (figure 5.28). On explique principalement cette erreur par la faible précision du système de suivi utilisé (*ARToolkit*) et des problèmes de discrétisation de maillage avec *ACIS*.



(a) (b) (c)

Fig. 5.28: Esquisse d'un modèle quelconque avec l'approche par sculpture interactive.

6.3. Discussion

Les techniques proposées démontrent l'approche générale mais l'interface et les résultats finaux restent fortement limités. On considère que dans le cadre d'usage d'objets réels pour des TUI ou des décors, elles procurent déjà des solutions intuitives et simples à mettre en œuvre.

Le système de suivi choisi réduit considérablement la démonstration de l'approche (précision trop limitée pour ce projet). Une première suite à ce travail serait donc de porter l'application sur notre environnement standard. Ensuite, il faudrait améliorer les algorithmes pour se rapprocher d'un mode plus semi-automatique et respectant plus les principes d'esquisse (figure 5.29).

Fig. 5.29: Travaux futurs : nous souhaitons esquisser la tasse (a) de façon très naturelle en définissant les surfaces courbes ((b) ou (c)).

7. Conclusion

Nous nous sommes intéressés dans ce chapitre à un mixage entre réel et virtuel. Nous avons proposé une environnement original de calibrage basé sur des méthodes simples et intuitives (publié dans [GDG01]). Les résultats démontrent l'efficacité et l'utilisabilité de notre méthode. Nous avons ensuite expérimenté une réduction de la latence par une minimisation du traitement système et l'usage d'algorithme de prédiction. Les résultats montrent une réduction apparente de la latence mais une difficulté dans l'utilisation du filtre de Kalman qui nécessiterait une étude plus pousée.

L'approche la plus prometteuse nous semble ce situer dans la continuité des premiers travaux de Shaw [SL92], en cherchant à définir un modèle de mouvement de la tête pour le filtre de Kalman. En sus, l'usage d'algorithmes adaptatifs [MEJ02] permettrait une meilleure gestion de la latence suivant l'utilisateur ou le contexte applicatif.

Nous avons ensuite introduit des méthodes pour le mixage entre des objets réels et une scènne virtuel, avec des nouvelles techniques permettant un ajout dynamique par une approche interactive. Nos premiers études sur une acquisition de la géometrie de l'objets par esquisse montrent des premiers résultats intéressants mais qui nécessitent d'être experimentés plus en détails.

A partir de cette intégration et à l'aide de notre architecture, il semble maintenant intéressant d'étudier la possibilité d'interagir et de naviguer dans du contenu mixte grâce à notre environnement. Nous aborderons donc ce thème au chapitre suivant.

Chapitre 6

Interaction avec le virtuel : tâches de base

> WIMP GUI based on the keyboard and the mouse are the perfect interface only for creature with a single eye, one or more single jointed fingers, and no other sensory organs.
>
> —Bill Buxton (Alias/Wavefront)

1. Introduction

Dans ce chapitre, nous nous intéressons à l'interaction entre des utilisateurs et un contenu virtuel positionné dans l'espace de la table. Les éléments proposés dans les précédents chapitres nous fournissent un bon cadre expérimental pour la définition de telles méthodes d'interaction 3D.

La réalité virtuelle peut nous procurer une base intéressante pour l'applicabilité des différentes taxonomies, concepts et métaphores de l'interaction dîtes 3D. Toutefois, il nous semble que la RA présente de nombreux nouveaux aspects par rapport à la RV pouvant alors induire à la proposition de concepts spécifiques à ce domaine.

Dans cette optique, nous présentons dans cette partie différentes métaphores d'interaction entre des utilisateurs et du contenu virtuel (3D ou 2D), dans un contexte collaboratif de RA sur table. L'interaction avec du contenu réel serra présenté au chapitre suivant. Un certains nombre de problématiques peuvent cependant être déjà intuitivement posées, nous servant de fil conducteur au cours du développement de ce chapitre (figure 6.1) :

- Quelles implications la vision du monde réel at-elle sur l'interaction ?
- Comment interagir avec des éléments virtuels dans le monde réel ? Faut il reprendre des métaphores provenant de la réalité virtuelle ? D'autres domaines ?
- Est ce que l'utilisateur doit manipuler les éléments 3D directement dans l'espace au dessus de la table ? Quel rôle peut jouer la table ?

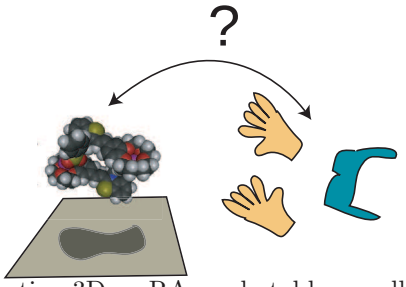


Fig. 6.1: Problématique de l'interaction 3D en RA sur la table : quelle relation établir entre l'utilisateur et le contenu 3D ?

Après une présentation du concept d'interaction 3D (section 2), nous introduirons notre décomposition suivant le type de tâche et un certains nombres de métaphores appropriées (section 3).

2. Interaction : terminologie et taxonomies

Nous commençons tout d'abord par le concept d'interaction 3D, puis par l'interaction en réalité augmentée et enfin par les spécificités liées à notre environnement.

Pour introduire le concept d'interaction 3D, nous revenons tout d'abord à la terminologie de ce domaine.

2.1. Définitions

L'*interaction*, aussi nommée *interface utilisateur*, peut être définie comme le lien entre l'individu et la tâche (ainsi que son contenu). Une *technique d'interaction* peut se définir comme une méthode pour accomplir cette relation (on la désigne aussi de façon moins conceptuelle par une *métaphore* ou *un paradigme d'interaction*). La technique peut se décomposer en différents éléments ([BLMP00]) :

- des valeurs d'entrées fournient à un système par un ou plusieurs périphériques manipulés par l'utilisateur ;
- La transformation de ces valeurs par une *fonction de transfert* vers des valeurs compréhensibles par l'application ;
- La présentation du résultat à l'utilisateur (sous forme visuelle, tactile, etc.).

Nous notons que la notion d'*interface* définit aussi généralement la représentation matérielle et logicielle de cette métaphore.

Les principales classes usuelles d'interface utilisateur sont : interface textuelle (ligne de commande), interface graphique (sélection directe et interface de bureau), orale (reconnaissance vocale), gestuelle (reconnaissance de gestes), tangible (à l'aide d'un élément physique) voir multimodale.

Les **degrés de libertés (DDL)** définissent le nombre de variables en évolution dans le système associé. On parle de DDL de périphérique d'entrée, DDL d'une tâche ou DDL d'une technique d'interaction.

L'**interaction 2D** défini la classe dominante d'études en IHM, historiquement associé sur des interfaces de type souris qui recouvre donc les applications de bureau.

L'**interaction 3D** est reliée au type d'interaction manipulant du contenu à représentation tridimensionnelle. Un grand nombre de classes usuelles d'interfaces 2D ont été reprises en interaction 3D, mais l'utilisation d'un rendu dit réaliste (voir stéreoscopique) a conduit à favoriser les approches par interface graphique et principalement des techniques 3D directes. L'interface physique standard recouvre généralement l'usage de capteurs 6D permettant de mesurer à la fois 3 positions et 3 angles (3 translations et 3 rotations).

A partir de ces définitions, nous étudions plus en détails les caractéristiques de l'interaction 3D par l'étude des taxonomies et des méthodes de conception.

2.2. Taxonomies

Bowman [Bow99] et Mine [Min95] se sont intéressés aux tâches élémentaires usuelles que l'on retrouve dans toutes les applications utilisant des métaphores d'interaction 3D et qui peuvent être regroupées en quatre catégories :

- la **sélection** : identifier pour le système l'élément sur lequel une action va être effectuée (comprenant la désignation et la validation de ce choix). Les techniques de validations sont trés variés : clic mécanique (bouton souris), temps de pause ou geste comme la détéction d'un mouvement de tape sur un support physique (commme un doigt sur une table) ;
- la **manipulation** : liée à la tâche de déplacement rigide d'objets 3D. On distingue principalement les paramètres de rotation et translation (voir l'échelle de l'objet) ;
- la **navigation** : correspond au déplacement du point de vue de l'utilisateur dans un monde virtuel ;
- le **contrôle d'application** : correspond la modification des paramètres de l'application. On retrouve par exemple dans cette catégorie le type d'outil ou le type de représentation

de l'application.

Chacune de ces tâches a été décomposé en différentes étapes par Bowman qui fournit un guide pour la réalisation de nouveaux paradigme d'interaction. Nous nous baserons sur cette taxonomie au cours de notre étude.

D'un autre point de vue, Uchiyama [UYKT97] classifie quand à lui les techniques d'interaction en trois sous-catégories basées sur une augmentation de la perception du réalisme :
- interaction au **niveau 0** : rendu temps réel et déplacement facile du point de vue (dédié à des applications de type *walk-through*) ;
- interaction au **niveau 1** : translation/rotation des objets et interférence entre objets, détection de collision ;
- interaction au **niveau 2** : déformation des objets et introduction de champ de gravitation, de contrainte d'élasticité, etc.

Dans notre thèse nous nous appuierons principalement aux niveaux 0 et 1 ; le cas 2 reste encore difficile à effectuer en temps réel. En effet le coût de ce type de simulation reste élevé [Deb00] et difficile à mettre en œuvre en sus de la réalisation d'un système interactif performant.

Comme le souligne Kato [KBP+00], les métaphores peuvent être classées en deux catégories : **naturelles** ou **magiques**. Les métaphores naturelles se basent sur la reproduction d'une action du monde réel (elles sont isomorphiques). Les métaphores magiques n'ont pas de correspondance directe avec des actions réelles, et sont donc définies avec des métaphores artificielles (non-isomorphiques).

Dans un cadre d'interaction 3D, la réalisation de tâches annexe reste nécessaire telle que la définition de contenu alpha-numérique (ou de tracé 2D). Nous nous pencherons sur ce besoin dans la suite de notre étude.

2.3. Méthodes de conception

La mise en œuvre de méthodes d'interaction 3D reste intrinsèquement difficile et ce pour plusieurs raisons :
- aucun standard ou méthode n'est disponible pour fournir une base commune pour la réalisation d'interfaces 3D (contrairement aux environnements de bureau tel quel le concept du *WIMP*) ;
- utilisation forte de périphériques 3D offrant une mauvaise précision, une mauvaise stabilité, une faible facilité naturelle d'utilisation (*affordance*) par rapport au périphérique 2D (e.g souris) ;
- manque de contraintes induites par un certains nombre de périphériques (mauvaise adéquation des DDLs, trop souvent superieur a ceux necessite par la tache telle que ne nombre de DDL possibles supérieur à ceux de la tâche) ;
- difficultés ergonomiques telle que la fatigue visuelle, musculaire, etc ;
- gestion non triviale de la perception 3D qui introduit un facteur de difficulté très important pour le positionnement et l'arrangement des éléments.

Nous retiendrons les critères suivants (défini par Bowman [Bow99]) pour la réalisation d'une métaphore d'interaction :
- la **performance** : efficacité, précision, productivité ;
- l'**utilisabilité** : facilité d'utilisation, d'apprentissage, confort de l'utilisateur ;
- l'**la transparence** (*usefullness*) : permet a l'utilisateur de se concentrer sur la tâche au lieu du fonctionnement de la métaphore.

Comme le montre Poupyrev [BLMP00] le développement de nouvelles métaphores d'interaction peut être basé sur une inclusion entre différents éléments (figure 6.2). Dans notre cadre, le rôle joué par les techniques d'interactions de RV et d'interactions 2D constitue des éléments essentiels pour la réalisation de nouvelles techniques d'interaction en RA.

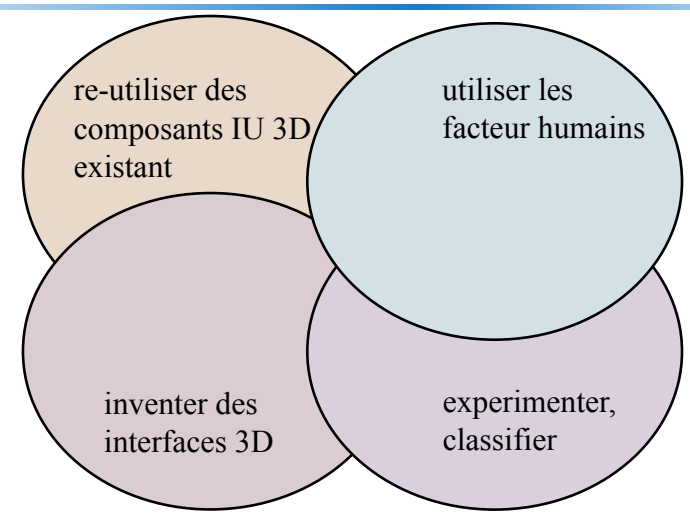


Fig. 6.2: Composants pour la réalisation d'une interface graphique d'après Poupyrev [BLMP00].

Étudions plus précisément les spécificités et le contexte de l'interaction en réalité augmentée.

2.4. Spécificités

Nous avons choisi un cadre de travail particulier. Nous revenons dans cette section sur plusieurs concepts et spécificités proposées.

2.4.1. Spécificités de l'interaction 3D

Zhai [ZM94] a largement étudié les propriétés des périphériques de bureau et a établi différentes taxonomies et relations avec le contrôle des méthodes d'interaction (voir annexe A pour une introduction à la terminologie). Il montre que les périphériques isotoniques favorisent le contrôle de position, tandis que les périphériques isométriques favorisent un contrôle de vitesse. Il montre aussi dans [ZM98] que le temps de réalisation d'une tâche (de *docking*) est plus court pour un périphérique à mouvement libre que pour un périphérique élastique, mais la trajectoire de mouvement est moins coordonnée. Il introduit alors un nouveau critère qui représente la coordination entre les multiples degrés de libertés (avec la métrique associée). Il discute alors de l'isomorphisme et compare l'usage d'outils lui conduisant à proposer une taxonomie (visible figure 6.3).

Il énonce alors que l'approche isomorphique est plus intuitive et nécessite moins de temps d'apprentissage (ce qui les rend applicables aux jeux vidéos), mais entraîne de forts risques de fatigue, un contrôle de mauvaise qualité et des restrictions du mouvement contraint anatomiquement. A contrario, les périphériques de types outils nécessitent un plus fort temps d'apprentissage, mais réduisent la fatigue, offrent de meilleurs mouvements de coordination et soufrent moins de limitations anatomiques (périphériques applicables à la télé-opération, à la visualisation).

Jacob [JSMM94] a étudié la relation entre la structure de contrôle d'un périphérique et la structure perceptuelle de la tâche à réaliser. Il se base sur la théorie de la structure perceptuelle de Garner qui considère qu'un objet peut posséder des attributs intégraux ou séparables : les attributs intégraux se combinent perceptuellement, formant un ensemble, tandis que ceux qui peuvent se distinguer sont dits séparables. Il ramène alors ces éléments à son cadre d'étude et montre alors que les performances sont meilleures si l'espace de contrôle du périphérique correspond à l'espace perceptuel de la tâche d'interaction

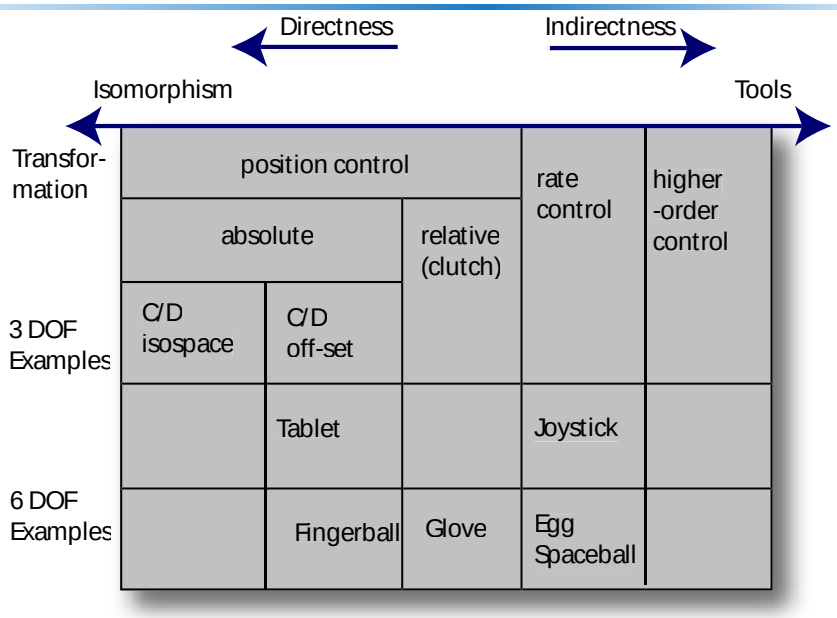


Fig. 6.3: Continuum d'isomorphisme de Zhai [ZM98] : taxonomie de différents périphériques d'entrées. L'axe des abscisse représente la relation d'utilisation du périphérique pour l'utilisateur (contrôle en position, vitesse ou autre). L'axes des ordonnées représente des exemples suivants le nombre de DDL du peripherique. La notation C/D défini les espaces de control (C) et d'affichage (D), et dénote ici leurs relations de superposition. La notion d'espace absolu et relatif défini la relation entre le mouvement physique de l'interface est le mouvement obtenu par rapport à un repere global (absolu) ou local (relatif).

graphique[1].

Nous retiendrons dans la suite les définition suivantes :

- **périphérique intégrale** : on peut se déplacer diagonalement entre les dimensions, le mouvement se situe dans un espace euclidien et on peut passer à travers toutes les dimensions du controlleur.
- **périphérique séparable** : on ne peut pas se déplacer diagonalement entre les dimensions, le mouvement est contraint à une marche en escalier (*city-block pattern*), le mouvement ne peut être que dans une dimension à la fois (voire 2D pour une interaction 3D).

Hinckley [HTP+97] s'est intéressé à l'analyse des techniques de rotation 3D et montre que l'efficacité est conservée avec un périphérique intégrale tel que *Virtual Sphere* ou *Arcball*. Il démontre que des critères tels que la facilité naturelle d'utilisation (*affordance*), l'information tactile, les propriétés de saisie et le temps d'acquisition du périphérique sont des critères essentiels.

Basé sur un système de réalité augmentée avec interaction sur table, Wang [WM00] démontre que le retour haptique passif incrémente fortement la vitesse de manipulation, mais réduit légèrement la précision par rapport à un périphérique libre. Avec une tâche à 3 DDL et un périphérique à 6 DDL, il confirme la théorie de Jacob. Il démontre de plus le peu de différence de résultats entre une méthode par glissement ou par soulèvement. Ces travaux font suite aux résultats de Lindeman [LSH99] sur l'utilisation d'une palette physique pour superposer des objets graphiques 3D, montrant de meilleurs résultats en temps de réalisation et précision par rapport à une solution libre. Mason [MWLM01] a étudié la structure de déplacement d'objet, et montre que le transport et la rotation sont des tâches parallèles et interdépendantes.

[1]en étudiant la relation entre une périphérique intégrale (capteur de mouvement Polhemus) et séparable (souris avec changement de mode pour la troisième dimension) et une tache intégrale (x,y,taille) et séparable (x,y et couleur)

2.4.2. Spécificités de l'interaction en réalité augmentée

La réalité augmentée présente des caractéristiques intéressantes pour l'interaction comme nous le résumons ici :

- proprioperception naturelle : on perçoit facilement la présence de notre corps, et nos membres en action ;
- conservation de l'environnement réel : cette caractéristique fournit une connaissance à l'utilisateur de l'environnement, facilitant alors la navigation. Nous notons, comme le montre Kiyokawa [KTY99a] que la RA ne peut pas tre dédiée à des tâches fortement immersives tel que *walkthrough* dans un bâtiment ;
- possibilité d'interagir avec des éléments réels : la visualisation de tous objet réels permet de l'utiliser sans difficultés comme objet d'interaction : ces objets peuvent être augmentés en représentation et sont définis dans le même espace que l'espace d'interaction ;
- meilleur intégration de contenu 2D et 3D : il est grandement facilité par la possibilité d'utilisation conjointe aisée de systèmes de visualisation 3D et 2D, telle que la projection murale ou les *tabletPC* ;
- support efficace d'interaction collaborative : on conserve les canaux de communication naturels (verbales, gestuelles) et la sensation de présence des autres utilisateurs ;
- interaction mobile : les système mobiles introduisent un nouveau type d'interaction qui tend à l'augmentation du monde réel avec la gestion de la mobilité de l'utilisateur.

Les méthodes d'interactions en réalité augmentée peuvent être classées avec la taxonomie introduite par Nigay [NDRT03]. Elle distingue deux paramètres essentiels qui sont le type d'élément d'interaction (objet réel ou virtuel) et le contexte de manipulation (tâche réel ou virtuel).

2.4.3. Spécificités de l'interaction colocalisé sur table

Après avoir vu les caractéristiques de l'interaction en RA, nous présentons maintenant les spécificités liées au cadre que nous avons choisi, l'interaction 3D sur table. Nous avons distingués trois éléments majeurs : l'espace de travail, le positionnement des utilisateurs et l'usage de la table.

Espace de Travail : la première spécificité introduite par notre environnement est un espace de travail réduit (table) qui limite alors le type de tâche et d'interaction réalisable avec ce support (pas de visualisation immerive de batiment, ni de voiture a l'echelle 1 contrairement des solutions de type *Reality Center* ou *Responsive Worbench*). Cet espace a comme avantage de pouvoir unifier espace de communication, de visualisation et d'interaction dans un même référentiel (contrairement à des approches de type projection murale). De plus, cet espace est partagé d'un point de vue de son accessibilité physique. Les concepteurs peuvent en effet facilement mettre en œuvre des techniques d'interaction 3D coopératives (mais engendrant aussi des conflits). Cet espace peut alors sembler plus approprié pour des techniques de manipulation directe.

Positionnement des utilisateurs : le fait de pouvoir autoriser un positionnement naturel respectant le cadre de réunion favorise l'interaction. La position statique assise réduit les possibilités de métaphores de navigations basées sur la mobilité physique du sujet. Elle réduit aussi les possibilités de navigation immersive d'un point de vue égocentrique (des tâches basées sur l'usage d'un point de vue exocentrique semblent plus appropriés).

Support de la table : la table permet tout d'abord de favoriser l'ergonomie des utilisateurs entre les phases d'interactions lui fournissant un bon support de repos. De plus, elle garantie une surface utile pour le développement de techniques d'interactions (e.g. pour des déplacements d'objets réels par interaction tangible, manuelle ou digitale). Cette table peut alors servir comme surface pour la reproduction de méthodes d'interaction 2D et leur

partage avec un cadre 3D.

Au vue de ces possibilités, nous nous sommes reposés sur ces caractéristiques pour la mise en œuvre de techniques d'interaction pour notre environnement.

2.5. Notre approche

Nous avons choisi de séparer en *deux niveaux d'interaction* les métaphores proposées. Le premier aspect concerne principalement les éléments de la taxonomie présentés par Bowman (interaction bas niveau). L'autre point d'étude est la prise en compte de l'aspect multiples contextes, multiples documents présent dans les environnements de bureau numérique ou dans les réunions réels (interaction haut niveau). Ce deuxieme point sera aborde dans le chapitre suivant.

Nous nous appuierons très fortement sur le double aspect réel-virtuel, en essayant de toujours s'intéresser à l'*attrait du réel* au delà de son concept de conteneur spatial.

3. Interaction : vers «l'hybridité»

Nous présenterons dans cette section des techniques d'interaction originales en suivant la taxonomie de Bowman : nous introduirons en premier lieu la manipulation d'objets virtuels 3D dans notre environnement (section 3.1), puis la navigation (section 3.2), le contrôle d'application (section 3.3) et des méthodes pour tâches 2D (section 3.4).

3.1. Manipulation

La *manipulation* est le déplacement d'un objet virtuel 3D dans l'espace d'interaction (contrôle possible jusqu'à 6 DDL). Nous présentons dans la suite les travaux relatifs à l'interaction en RA 3D pour la manipulation puis notre approche.

3.1.1. État de l'art

Les diverses techniques de manipulation 3D en RA reprennent en général un grand nombre de métaphores de RV. De nombreux travaux se reposent sur le suivi par marqueurs *ARToolkit* contraignant technologiquement trop fortement les possibilités des approches présentés. Nous distinguons principalements trois classes : technique classique, gestuelle ou tangible.

Classique : Schmalstieg [SFSG96], Tuceryan [TGW+95b] ou Butz [BHF+99] reprennent la techniques classique de RV nommée *virtual-hand*, à l'aide d'un pointeur suivi équipé de boutons. complète la faible précision des capteurs de mouvement, par des techniques de *snapping* face à face pour le positionnement d'éléments. Slay [SPVT01] reprend le concept du rayon de RV alors que Rekimoto [Rek96] reprend le concept du plan d'interaction. Les approches d'IHM favorisent aussi l'interaction par pointeur laser ([RS99]).

Gestuelle : Dans le cadre d'interaction gestuelle différents prototypes ont été introduit. Thomas [TP02] reprend les concepts des *Pinch Glove* et propose son propre modèle équipé d'un marqueur 3D sur l'index et le pouce pour le positionnement spatiale. Dorfmuller [DUS01] propose un système de suivi stéréoscopique de boules réfléchissantes, limité à un déplacement 3D et au mouvement de flexion d'un doigt. Veigl [VKL+02] propose aussi une solution pour de la RA mobile à l'aide de marqueurs positionnés sur un support un rigide autour d'un gant avec à un bouton fixé au bout du pouce.

Des approches plus souples par vision ont aussi été proposés dans ce contexte, généralement 2D ([CC95], [NSK01]) ou 2D dans un espace 3D ([Mal02], [DSB+03]). Le prototype *Sheep* propose de l'interaction gestuelle en positionnant un marqueur sur le dos de la main pour sélectionner et déplacer des éléments.

Comme déjà introduit des techniques multi-utilisateurs ont vue le jour par les tablette digitales 2D ([WB03]).

Tangible : Bilinghurst [BKWF99] introduit quant à lui le concept d'interfaces tangibles (*TUI : Tangible User Interface*) dans un cadre de réalité augmentée 3D. On détaille plus spécifiquement ce point car c'est un concept incoutournable pour la RA.

La manipulation d'éléments virtuels se fait sous la forme d'une interaction avec des éléments réels, l'association étant prédéfini. Ullmer décompose dans sa thèse [Ull02] les TUI en quatre types (objets, instruments, surface et espace tangibles) et propose récemment trois grandes classes de TUI : spatial, relationnel et constructif.

Dans le cadre de système numérique, Undwekoffler et Ishii définissent un continuum sur l'utilisation d'élément réel appliqué à des interfaces tangibles [UI99] :
- l'objet en tant que nom : ils sont representation d'eux même et ont leur propre modèle dans l'espace numérique ;
- l'objet en tant que verbe : l'élément est fonctionnel, agit sur d'autres éléments ou sur son environnement ;
- l'objet en tant qu'outil reconfigurable : similaire à la métaphore souris-pointeur, l'objet offre un multiplexage temporelle de fonctionnalité (on favorisera des outils contextuels, sans l'utilisation de menus ou d'interface non naturelle) ;
- l'objet en tant qu' attribut : une propriété de l'objet sert d'interface avec le système, telle que sa forme, sa couleur, ou sa taille.

L'utilisation d' interfaces tangibles a déjà été étudié en IHM et RA 2D. Différentes approches basées sur la technologie ont été proposé : par capteur magnétique ([Fit96], [PIHP01]), par vision ([IU97], [RFK+97]). Plus récemment Berard [Ber03] propose l'utilisation de pions de couleur pour la sélection et manipulation d'annotations réels digitalisées en cours de session.

Un des travaux les plus aboutis est sans doute BUILD-IT [FSSK00] qui introduit diverses métaphores d'interaction RA basé sur un système de visualisation 2D. A l'aide d'une brique réel on peut accrocher des objets virtuels et ensuite déplacer l'élément dans 3 DDL (2DDL translation + 1 DDL rotation). Pour le support de la troisième dimension, Fjeld propose le positionnement vertical d'un groupe d'objets virtuels à l'aide d'un ascenseur dans un menu particulier.

Divers travaux ont vu le jour en RA 3D reprenant le concept de Billinghurst ([PTB+02], [DMS+02]). Ulbricht [Ulb03] propose une amélioration du système par un suivi sous une table transparente (vue de dessous) résolvant une partie des occultations liés au système de suivi. Le projet *Spacecube* [Gos01] étend la robustesse du suivi par vision l'utilisation de cube tangible (avec six marqueurs démonstré sur la construction d'une molécule). Dans *SCAPE*, Brown [BH03] propose l'utilisation du Co-Cube, un cube réel qui permet la sélection d'un élément par alignement de cette objet avec le monde virtuel, puis l'observation de l'objet par rotation. Deux techniques ont été proposées pour la manipulation sur table : le prototype *VOMAR* et *MagicCup* [KBP+00] et [KTT+03]. Le premier propose l'utilisation d'une palette pour déplacer les objets par dessous. Le deuxième propose une selection et un déplacement en positionnant un objet réel par dessus l'objet virtuel.

Multimodale : Le prototype SenseShape [OBF03] introduit une approche multimodale (gestuelle et orale) basé un reconaissance probabiliste. Le projet RASA [MCW00] propose un environnement de RA 2D multimodale par utilisant un système de reconnaissance textuelle et un système vocale.

Les prototypes *VOMAR* [KBP+00] ou *MagicCup* [KTT+03] offrent les solutions les plus intéressantes pour l'interaction sur table. Mais ils nous semblent trop restreints. *MagicCup* semble difficilement utilisable pour des objets de grande hauteur et *VOMAR* peu efficace dans le cadre de tâches de longue durée (métaphore simple mais longue et peu robuste).

3.1.2. Approche choisie

Nous présentons ici les principes et propriétés retenus du modèle d'interaction proposé.

Décomposition par tâches

Contrairement à de nombreux travaux, nous avons choisit de fournir plusieurs nouvelles techniques d'interaction en fonctions du type de tâche, des degrés de liberté et de l'espace d'interaction utilisé. Basés sur le contexte introduit par Aliakseyeu [AaSR02], nous considérons trois sous-types de classes de tâches représentés figure 6.6 :

- des *tâches que l'on définit comme 2D*, supposant une scène planaire reposant sur un plan 2D ou avec de très faibles variation d'altitude (exemple : planification urbaine, archéologie) ;
- des *tâches que l'on définit comme 2D1/2 (ou 2D5)*, supposant du contenu virtuel dont on controle la variation d'altitude (exemple : géographie, architecture). L'introduction du terme 1/2 spécifie que la hauteur est toujours définissable par une fonction $z = f(x, y)$;
- Des *tâches que l'on définit comme 3D*, n'ayant aucune base physique associée à la manipulation d'éléments virtuels et n'étant pas caractérisées par un respect gravitationnel (exemple : visualisation scientifique, ingénierie).

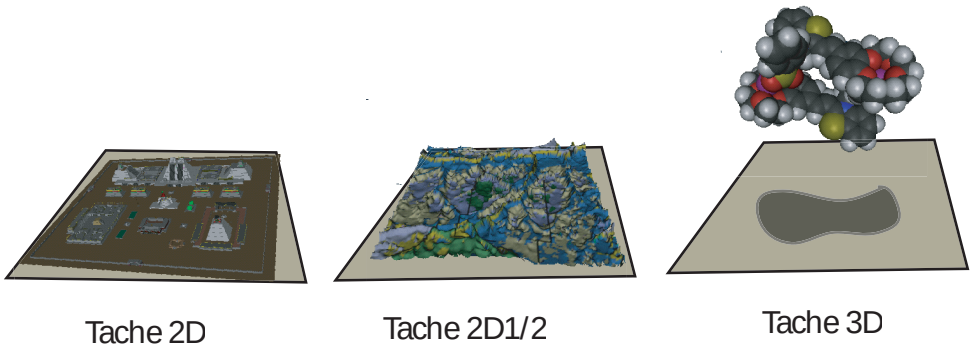


Fig. 6.4: Trois classes de tâches sur table en fonction des DDL.

Nous utiliserons fortement la notion de contrainte afin de réduire les DDL libres. Comme le définit Mine [Min95], il existe deux types de contraintes sur lesquelles nous nous appuierons lors de notre étude : les *contraintes physiques* qui restreignent le déplacement de (ou des) périphérique(s) dans un sous-espace et les *contraintes virtuelles* qui restreignent le résultat de la fonction de transfert à un sous-espace plausible.

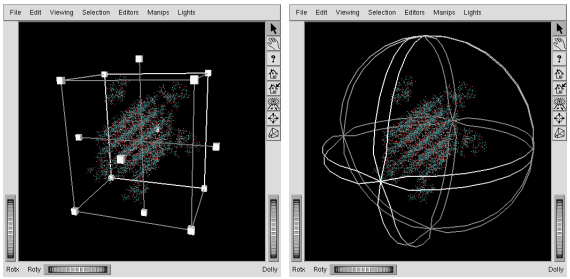
Manipulateurs

Nous nous inspirerons largement des métaphores d'interaction introduite par *Open Inventor* [SGI] reposant sur la notion de manipulateur (*manipulator*), interface visuelle locale de manipulation d'objet (figure 6.5). Chaque manipulateur [MBE03] comprend une ou plusieurs métaphores (*draggers*) accessibles par des objets virtuels dédiés nommés poignées (*handles*). Le fort succès de ce type d'interface repose sur une souris mono-utlisateur et une simplicité d'utilisation et de représentation. Dans cette these nous nous intéressons au passage de la surface d'interaction 2D écran à la surface d'interaction table réel (représentation du *picking* alors modifiée).

Espaces d'interaction

Nous choisissons de représenter nos espaces d'interaction et de visualisation en trois grandes classes (figure 6.6) :

- espace de visualisation 3D couplé avec l'espace d'interaction 3D, représenté par l'espace au dessus de la table qui respecte alors une interaction directe «forte» ;



(a) (b)

Fig. 6.5: Deux exemples de manipulateurs d'*Open Inventor* : (a) *HandleBox* (b) *Trackball*.

– espace de visualisation 3D aligné avec l'espace d'interaction 2D, représenté par la surface du dessus de la table supportant une contrainte d'alignement pouvant favoriser l'interaction et la représentation induite de l'utilisateur ;
– espace de visualisation 3D en relation quelconque avec une surface ou espace d'interaction, assimilable à la métaphore de bureau (écran-souris), avec ici l'utilisation d'un conteneur à position quelconque.

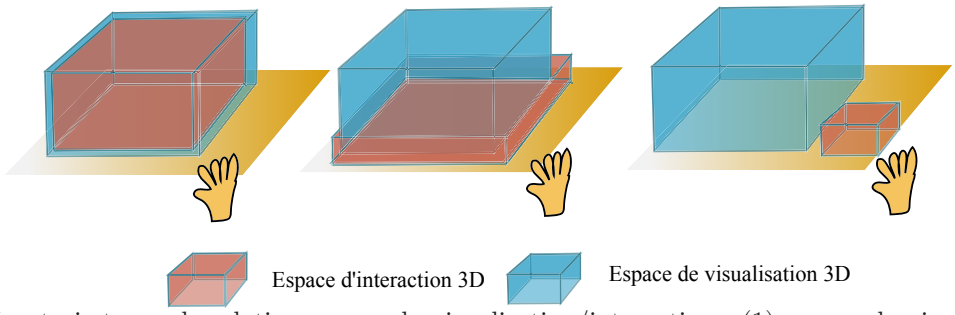


Fig. 6.6: Les trois types de relation espace de visualisation/interaction : (1) espace de visualisation et d'interaction superposés (2) alignement espace de visualisation et espace d'interaction et (3) relation quelconque entre les espaces.

Nous nous intéressons particulièrement au deuxième aspect, en choisissant comme approche de maximiser les contacts avec le support physique, besoin important démontré dans un cadre d'interaction 3D (voir les résultats d'étude de Wang [WM00], Lindeman [LSH99]).

Interfaces dédiées

Nous nous appuyons sur des interfaces spécifiques dédiées au contexte de réunion sur table et présentées chapitre 4. Nous utiliserons les interfaces et les méthodes d'interaction «classiques» de RV (ou de bureau) pour des comparaisons de performances.

Nous présentons maintenant les techniques proposées à l'aide de ces interfaces pour les tâches 2D (section 3.1.3), tâches 2D1/2 (section 3.1.4) et tâches 3D (section 3.1.5).

3.1.3. Techniques d'interaction : tâche 2D sur table

Nous nous intéressons à la première catégorie de tâche où les éléments sont contraints à rester positionnés sur la table. Dans ce cadre l'utilisateur peut : déplacer sur le plan l'élément (2DDL), spécifier son orientation (1DDL) ou spécifier sa taille (3DDL ou 1 DDL avec un changement uniforme).

Comme décrit par Bowman [Bow99], nous considérons trois étapes de réalisation d'une tache de manipulation : sélection, action, désélection. La phase de sélection comprend la métaphore d'identification et de désignation de l'élément sur lequel est appliquée la tâche

et la validation de ce choix. La phase d'action correspond au déplacement, à la rotation ou au changement d'échelle. La dernière phase, désélection, marque la fin d'action et optionnellement la fin de l'identification de l'élément.

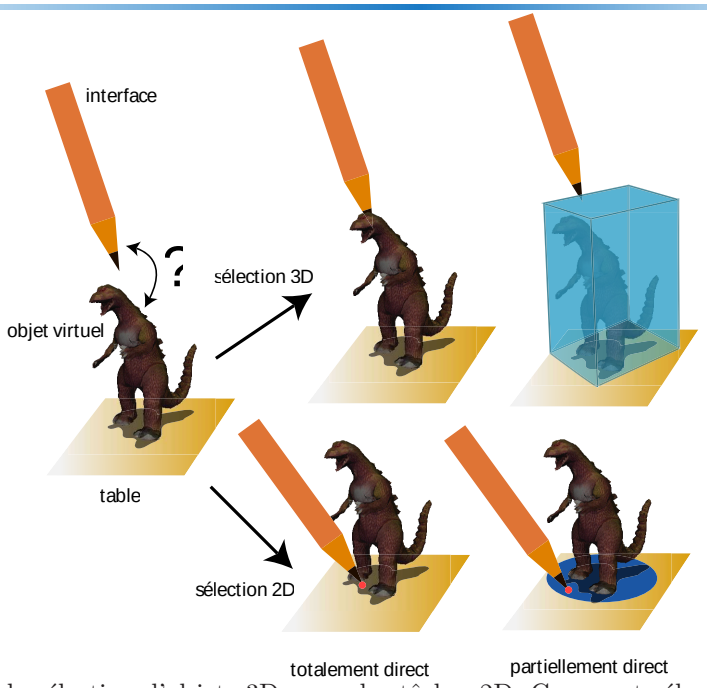


Fig. 6.7: Métaphore de sélection d'objets 3D pour des tâches 2D. Comment sélectionner l'objet virtuel 3D (dinosaure) ? On différencie une approche 3D (en haut) d'une approche 2D (en bas).

Nous nous intéressons à la sélection de l'élément 3D (figure 6.7) selon deux approches : 3D ou 2D, et sous plusieurs types. Le premier type est la sélection «totalement directe» de l'élément qui correspond à un contact entre un point (ou plusieurs) d'interaction de l'outiil et l'enveloppe visuelle de l'élément.

L'interaction peut également se faire non plus directement avec l'élément, mais par l'intermédiaire d'un objet annexe, on parle alors de sélection «partiellement directe». Cet objet est un mécanisme inséré dans la scène pour la manipulation de l'élément. Il pourra correspondre soit à un objet virtuel quelconque en relation avec l'élément, soit à une extension de l'enveloppe visuelle, un élément simplificateur englobant (boite englobante) ou encore à une partie de la scène virtuelle (e.g. ombre de l'objet).

Enfin la «sélection indirecte» peut se faire sous la forme d'une interface graphique (menu) ou par une approche multimodale (telle que dans *Put That There* [Bol80]). Travaillant dans un espace restreint nous nous orienterons vers une méthode directe/partiellement directe principalement sous la forme de point de contact.

Houde [Hou92] a réalisé une étude des méthodes de manipulation directe d'objets réels 3D. Ses observation montrent qu'il n'y a pas de méthodes idéales de manipulation, chaque personne utilise ses propres métaphores de déplacement. Les utilisateurs ont des espérances sur la façon de saisir et manipuler les éléments. Les approches choisies par les utilisateurs sont principalement le glissement sur plan, le soulèvement et l'orientation des éléments. Elle montre que l'utilisation de métaphores génériques, telles que des poignées sur la boîte englobante, est très efficace si la représentation de ces poignées aide l'utilisateur à leur compréhension, permet une réduction du nombre de DDL de manipulations par contrainte et autorise une transition aisée entre translation et rotation.

Méthodes

A partir de ces observations nous présentons dans notre notre thèse quatre nouvelles méthodes originales pour l'interaction dans le cadre de tâche 2D sur table. Chacune de

ces méthodes serra décrite et argumentée, leur mise en œuvre explicitée plus tard dans la suite du document.

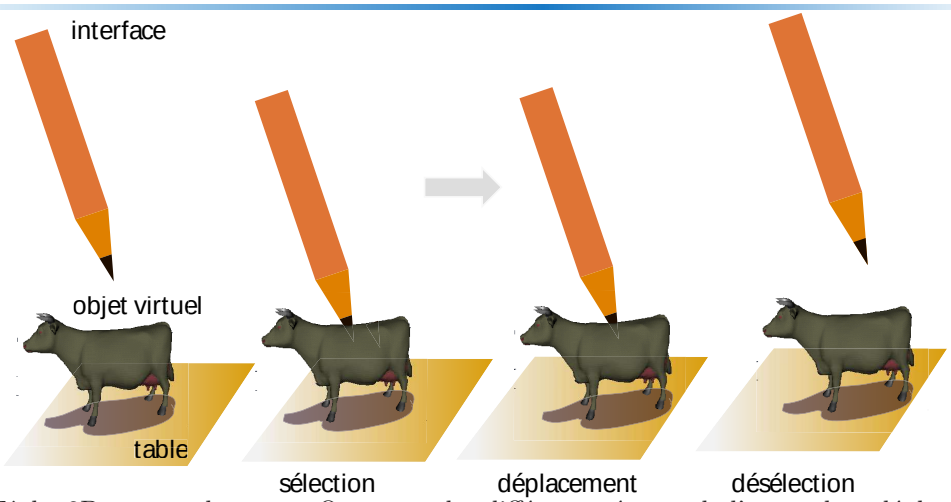


Fig. 6.8: Tâche 2D : approche naïve. On notera les différentes étapes de l'approche : déplacement libre du pointeur (stylo), la sélection de l'objet virtuel, le déplacement de l'objet et le relâchement.

Approche naïve : la première approche, dite naïve, consiste à repliqué les approches qui existent déjà en RV (ou IHM) et les transposer dans notre cas (un espace 3D au dessus de la table). Cette approche se décompose en une sélection 3D suivie d'un déplacement 3D non contraint (figure 6.8). La sélection 3D se fait par placement du point d'interaction sur l'objet virtuel, puis son déplacement dans l'espace déplace directement l'élément. La désélection peut se faire sous plusieurs formes (dans l'ordre de préférencee d'utilisabiilité nous choisissons le déplacement vertical, clic bouton ou fin de contact entre deux doigts avec l'interface digitale).

Le mouvement induit par le stylo est transformé pour ne conserver que les 2 DDL, la tâche étant intrinsèquement de type 2D. Comme le montrent de nombreux auteurs en RV (ou IHM) cette approche est limitée, le périphérique n'étant pas adapté à la tâche car fournissant trop de degrés de liberté pour l'utilisateur. Une amélioration consiste alors à mettre en œuvre des techniques de *snapping* lorsque deux objets sont à proximité, limitant alors les degrés de libertés de cet objet.

Approche «*lift-and-move*» : cette solution semblant trop libre, nous introduisons une contrainte dans la méthode précédente consistant à forcer le contact avec un élément physique (la table) lors des étapes de sélection ou de desélection. Cette deuxième approche (figure 6.9) est aussi une sélection 3D et un déplacement 3D non contraint, mais nous introduisons ici l'analogie de déplacement d'un objet par dessus un obstacle (comme par exemple lors de la prise au jeu d'échecs).

L'utilisateur sélectionne l'élément en 3D, puis il est contraint à valider la sélection par contact avec la table suivi d'un déplacement vertical (assimilable à la saisi digitale réelle). Il peut ensuite faire un mouvement libre dans l'espace et l'objet est finalement repositionné grâce à des méthodes de détection de collision entre l'objet et la surface (pouvant être complétée avec à méthode de *snapping*). La fin de sélection s'effectue après le contact physique entre le point d'interaction et la surface d'interaction (techniquement fini lorsque le point d'interaction sort de l'objet ou de son volume englobant).

Le déplacement sous une forme connue contraint alors l'utilisateur dans la réplication d'un geste maîtrisé, favorisant sa compréhension et son apprentissage. Cette introduction du contact physique nous a semblé intéressante pour procurer une sensation de retour haptique «faible» avec l'élément (retour 2D pour objet 3D) et aider au bon positionnement

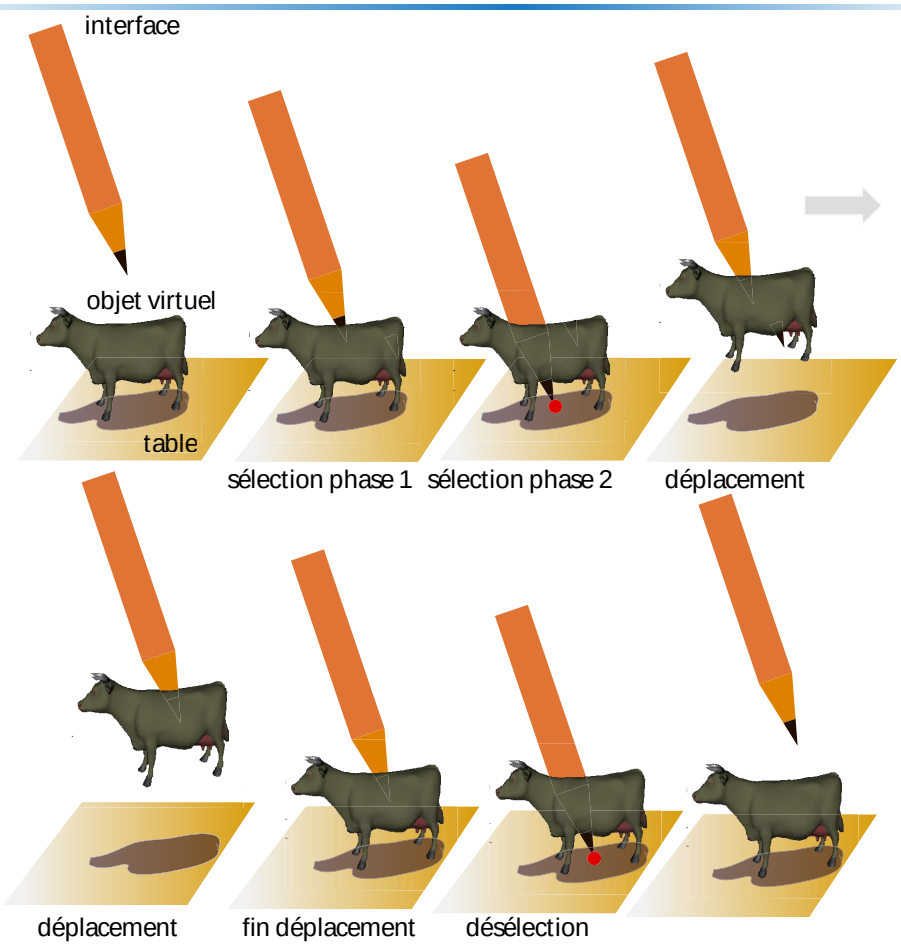


Fig. 6.9: Tâche 2D : approche «*lift and move*».

en phase terminale de l'objet par la caractéristique de decéleration favorisée par cette approche (voir l'étude de Mason [MWLM01]).

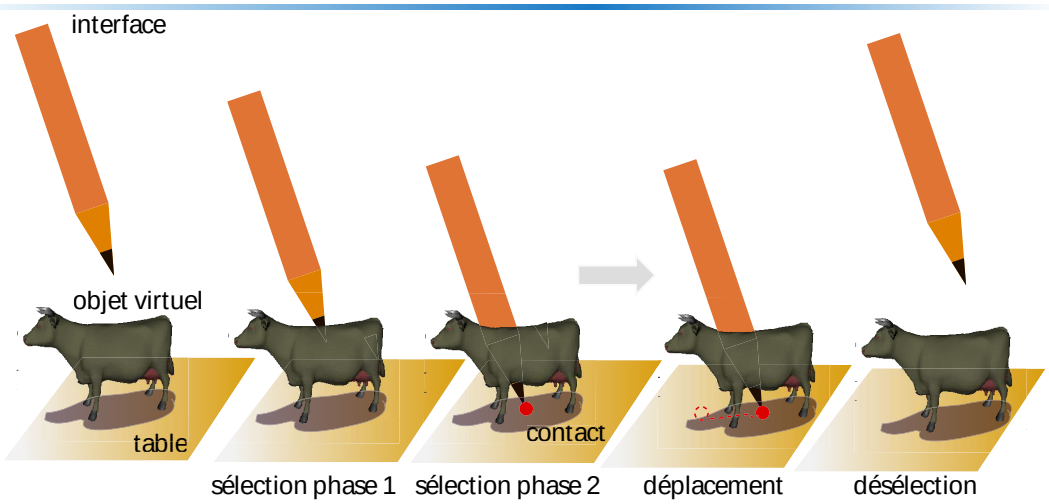


Fig. 6.10: Tâche 2D : approche *slide and move* avec sélection 3D.

Approche «*slide-and-move*» : la troisième approche proposée (figure 6.10) est une sélection 3D, mais avec un déplacement 2D. L'intérêt de la sélection 3D est la conservation de la métaphore usuelle de manipulation réel. Pour les utilisateurs expérimentés des outils de RV, elle offre une similitude avec leurs outils usuels. L'intérêt du déplacement 2D

repose quand à lui sur une contrainte physique (déplacement sur support plan), et une correspondance directe entre DDL de la tâche et DDL de la métaphore (l'analogie peut se comparer ici au glissement d'un pion sur un plateau).

Les différennts étapes sont : une sélection identique à la précédente méthode, puis validation de la sélection par contact entre le stylo et la table, déplacement par glissement du point d'interaction sur la table et enfin soulèvement vertical pour marquer la désélection (on nomme cette métaphore *slide and move*).

Nous pouvons discuter de facon empirique des avantages de cette méthode par rapport à la précédente. Une première limitation du *slide-and-move* apparaît lors de son usage pour un mouvement entre deux surfaces planes disjointes, qui ne permettent donc pas un glissement entre ces deux surfaces. Ensuite, travaillant dans un espace mixte le *lift-and-move* permet de contourner des obstacle réels (décor). Nous n'avons pas réaliser de tests comparatifs entre les deux méthodes mais les evaluations informelles ont justifiés les hypothèses précédentes. Il serait aussi utile de juger de l'efficacité des deux méthodes pour des obstacles virtuels comme l'impact de passer à travers (tels que des murs pour une maquette virtuelle).

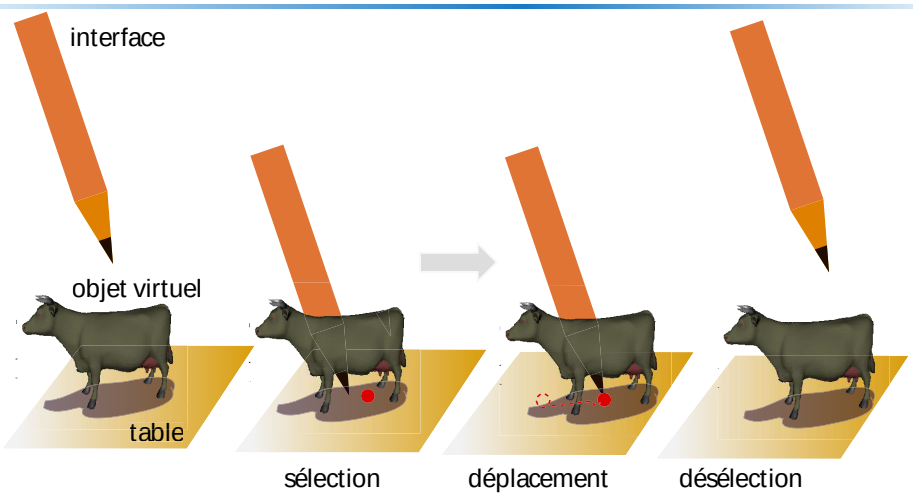
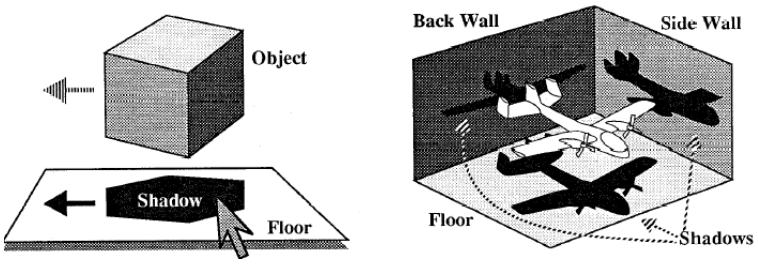


Fig. 6.11: Tâche 2D : approche *slide and move* avec sélection 2D.

Approche «*slide-and-move 2D*» : La dernière approche proposée (figure 6.11) est une sélection 2D suivie d'un déplacement 2D. Cette solution nous ramène à des métaphores d'interaction 2D en RA, la troisième dimension servant uniquement à la visualisation. Mais une difficulté réside alors sur le choix de la zone de sélection.

Le prototype *BUIlD-IT* propose une sélection 2D par superposition d'une brique réel 2D sur la projection d'objets virtuels sur la table. Cette projection, méthode de visualisation 2D, restreint alors le choix de la méthode de sélection. Dans notre cas, visualisation 3D, l'approche naïve consiste à considérer le bas de l'objet comme unique zone de sélection. Le problème est alors la difficulté d'accès à cette surface pour certains éléments : pour un sapin par exemple, les branches contraignent la visibilité et l'accessibilité à cette zone (le bas du tronc). Nous choisissons donc d'introduire un élément virtuel complémentaire en forte corrélation avec l'élément à saisir.

Le choix de cet élément peut être une deuxième représentation virtuelle de l'objet positionné sous cet élément, de taille suffisamment grande pour couvrir une surface d'interaction toujours accessible à l'utilisateur. Nous considèrons ici une approche qui consiste à utiliser **l'ombre de l'objet sur la table** L'intérêt de l'ombre est sa compréhension aisée par l'utilisateur dans les situation réelles, et sa forte présence dans notre environnement, et un élément visuel en forte relation avec l'objet lui même.



(a) (b)

Fig. 6.12: *Interactive shadow widget* de Herndon [HZR+92] avec plan virtuel.

Dans un contexte d'interaction 3D de bureau, Herndon [HZR+92] a présenté une nouvelle métaphore nommé *interactive shadow widget* (figure 6.12), reposant sur l'utilisation de ces ombres pour l'interaction (dans un cadre de bureau virtuel manipulé à l'aide d'une souris). Nous nous intéressons dans notre cas à reproduire ce concept mais avec de nombreuses différences : utlisation en RA collaborative, alignement entre élément et ombre favorisant la résolution des conflits dans le cadre de *picking 2D*, interaction plus directe (même espace de visualisation/interaction) et contrainte physique correspondant au plan d'interaction.

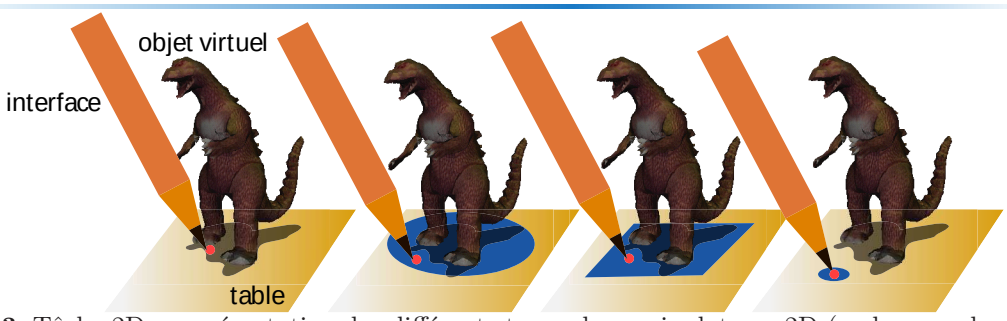


Fig. 6.13: Tâche 2D : représentation des différents types de manipulateurs 2D (ombres, ombres symboliques ou autre formes).

Cette ombre (figure 6.13) peut aussi être modélisée de façon plus symbolique, comme sous une forme iconographique simplifiée (e.g. avec un élément englobant 2D, un rectangle ou une ellipse). Cette approche conserve la relation visuelle 2D ou 3D, tous en fournissant un généricité de l'élément et une simplicité de mise en œuvre (voir les problèmes de coûts dans l'article de Herndon).

Cette ombre pourrait prendre différentes formes suivant le mode d'action d'un utilisateur en interaction. Nous définissons ces éléments sous le terme «widget de réalité augmentée sur table» (*tabletop AR widget*).

Approche pour la rotation : Ces approches originales sont limitées à des mouvement de translation. Nous proposons une méthode complémentaire pour la rotation (restreinte dans cette tâche 2D a la rotation suivant l'axe perpendiculaire à la table). Nous nous basons ici aussi sur une analogie, principalement le geste mimétique pour expliciter un mouvement de rotation. Le principe (figure 6.14) est ici de reprendre le concept précédent de widget de RA sur table, la différence se situant après la phase de sélection : l'utilisateur peut tourner l'objet en effectuant un mouvement de rotation autour de cet objet en gardant le point d'interaction en contact sur la table, et en effectuant une rotation de l'intterface par rapport au point de sélection (rotation avec conservation du point fixe pour l'interface *mixed-pen*, rotation du doigt glissant sur la table pour l'interface *light weight glove*).

Approche pour la rotation et translation : pour la rotation et la translation simultanées, nous combinons les méthodes précédentes de translation et rotation. Cette nouvelle

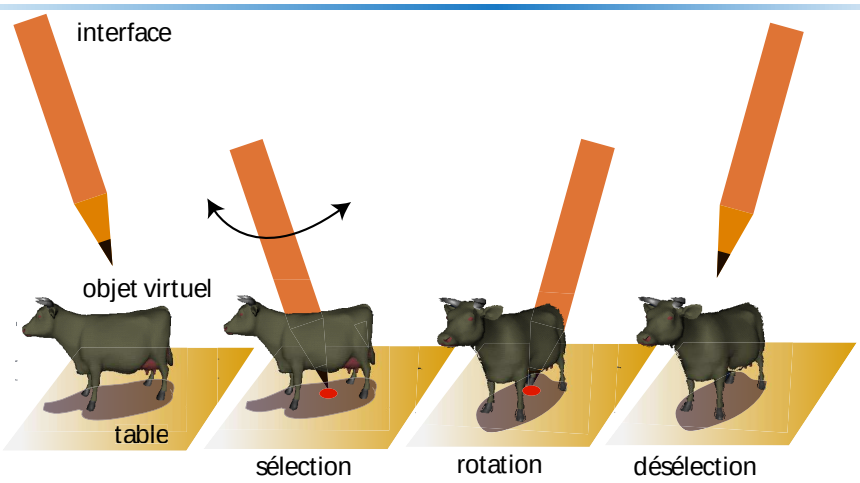


Fig. 6.14: Tâche 2D : rotation avec sélection 2D.

méthode définie une approche nouvelle en RA 3D inspiré des techniques utilisés en IHM (comme la métaphore «trackball» de Ahlers [CFH97] ou les travaux pour les *SDG* de Balakrishnan [WB03]).

Mise en œuvre

Nous privilégierons les interfaces *mixed-pen* et *lightweight AR glove* pour les méthodes avec sélection 3D. Nous privilégierons l'utilisation d'éléments tangibles pour les approches par sélection 2D et glissement (la désélection pouvant alors reprendre le principe du masquage, du temps de pause ou du clic mécanique). Pour le retour visuel nous avons choisi un retour sonore public et un affichage de boîte englobante (avec modification de sa forme et de son apparence au cours des étapes de manipulation).

Résultats

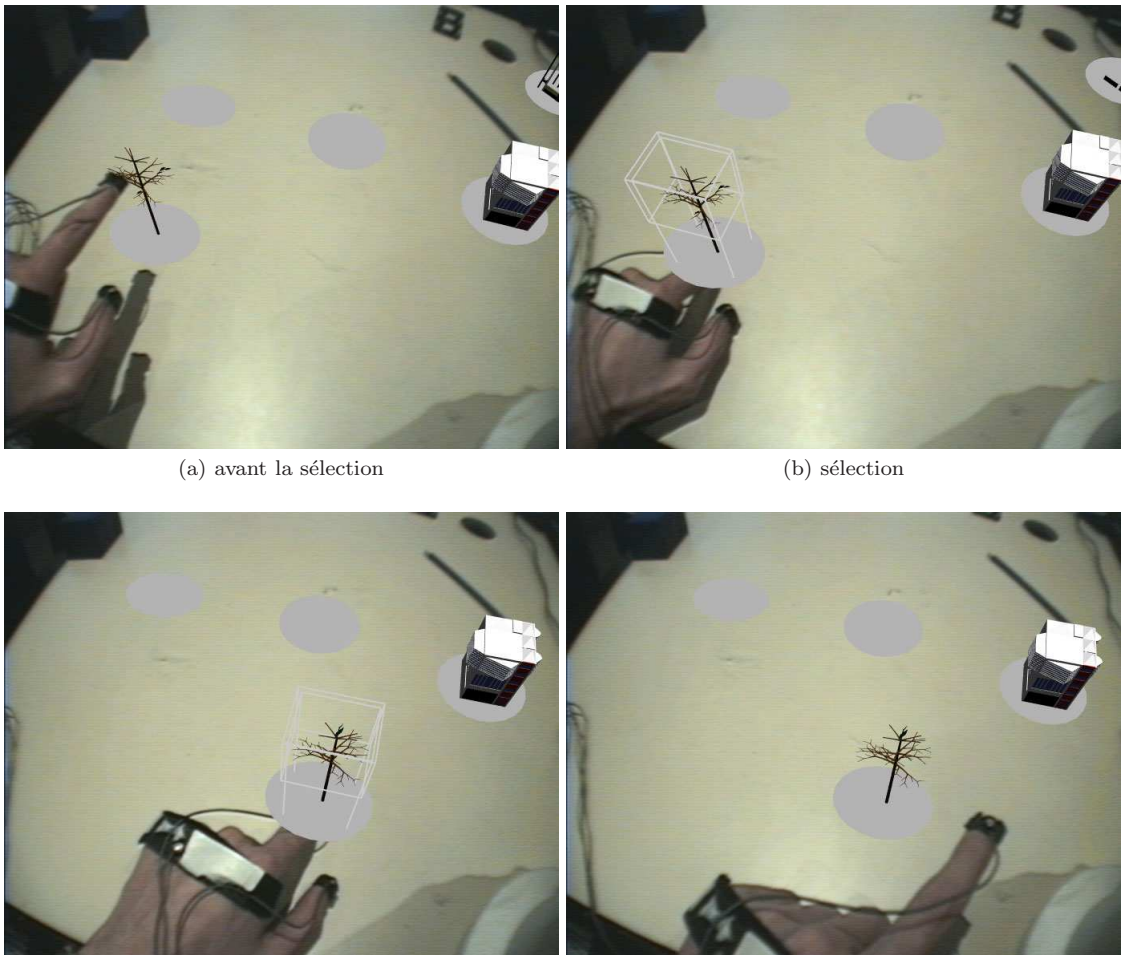
La figure 6.15 représente le résultat obtenu avec la méthode *slide-and-move 2D*. Nous notons qu'un problème soulevé est l'occultation du doigt réel par l'objet virtuel (à cause du casque vidéo), mais il est peu gênant en pratique (le couplage doigt-objet semble suffisament fort).

Nous avons **évalué de façon informelle les méthodes** *slide-and-move* et *lift-and-move* sur une trentaine de personnes au cours de ces dernières années. Le protocole général était alors : description de l'interface, essai de perception RA par l'utilisateur, apprentissage assisté de la méthode, manipulation libre par les utilisateurs. Nous avons observé une forte facilité de compréhension, d'apprentissage et de dextérité de manipulation des deux méthodes par les utilisateurs.

Ces tests informmels nous ont permis d'en déduire et de réaliser trois améliorations pour nos métaphores *slide-and-move* et *lift-and-move* :

1. remplacer le le retour visuel sous la forme de flèche et de surimpression de la boîte englobante par l'affichage d'une zone de sélection 2D sur la table pour le début de la méthode.
2. privilégier l'interface digitale offrant une meilleur souplesse et précision aux utilisateurs.
3. les tests avec le capteur magnétique ont montré des difficultés pour repositionner les objets à cause de l'erreur d'estimation de la position exacte de la table.

Un utilisateur a proposé de remplacer le déplacement vertical par un temps de pause. Un autre a proposé un mélange des méthodes *slide-and-move* et *lift-and-move* nécessitant un concept de règle implicite restant difficile à définir (le mouvement initial étant le même et les préférences utilisateurs non connues).



(a) avant la sélection (b) sélection

(c) déplacement par glissement (d) après désélection

Fig. 6.15: Déplacement d'un arbre virtuel avec la méthode *slide-and-move 2D* sur une ombre symbolique elliptique.

Les méthodes *slide-and-move 2D* et *rotation et translation* récemment expérimentées par 6 personnes ont montré aussi une facilité d'utilisation, d'assimilation et d'efficacité. L'expérimentation a démontré un problème de visibilité des marqueurs dans le cas de la méthode *rotation et translation*, conduisant à définir un facteur multiplicatif pour la rotation

Les méthodes présentées reposent fortement sur des analogies (saut d'obstacle,glissement, geste mimétique pour la rotation) qu'il est nécessaire d'évaluer plus en détails. De plus le choix d'une solution appropriée reste difficile, chaque technique ayant ses avantages propres. Des évaluations comparatives de performances non réalisées sont nécessaires pour aller plus avant dans la validité de nos nouvvelles méthodes.

3.1.4. Technique d'interaction : tâche 2D 1/2 sur table

Le placement, par exemple, d'un bâtiment virtuel sur un terrain virtuel est défini par un point de positionnement spécifique, valeur obtenue à partir du modèle du terrain. Plus généralement, nous supposons q'une fonction $z = f(x, y)$ permet de connaître la troisième composante de placement. Cette fonction peut donner plusieurs solutions, si on considère par exemple le cas d'une table, un objet pouvant être placé dessus ou dessous. Nous traiterons alors séparément les deux cas : fonction à une solution (cas 1) ou fonction à plusieurs solutions (cas 2).

Cas 1 : solution unique pour le placement

Comme dans les cas des tâches précédentes on cherchera à maximiser le contact avec la table (illustré sur le cas de placement d'un bâtiment sur un terrain virtuel). L'approche basique (méthode naïve) consiste ici en un déplacement libre de l'élément dans l'espace au dessus de la table et en des techniques de *snapping* lorsque le modèle se trouve près du terrain.

Une autre approche serait de réutiliser les technique *slide-and-move* et *lift-and-move*. Mais ceci semble ici difficile à cause d'un problème de visibilité. L'utilisation par exemple d'un terrain virtuel supprime la visibilité de la surface de la table. Les techniques précédentes utilisant fortement le contact de la table et la visibilité des points d'interaction est alors réduit (manipulation en «aveugle»).

Nous remarquons de plus que si le modèle est de hauteur conséquente, le choix des méthodes précédentes semble moins interessant que l'usage de la méthode basique. Le gain en utilisabilité devient négligeable par rapport à la perte en efficacité. En effet, avec nos méthodes l'utilisateur effectuera alors deux mouvements de forte hauteur (sélection sur table puis déplacement inverse jusqu'à la position de manipulation) par rapport à une solution directe avec la méthode naïve.

Méthode

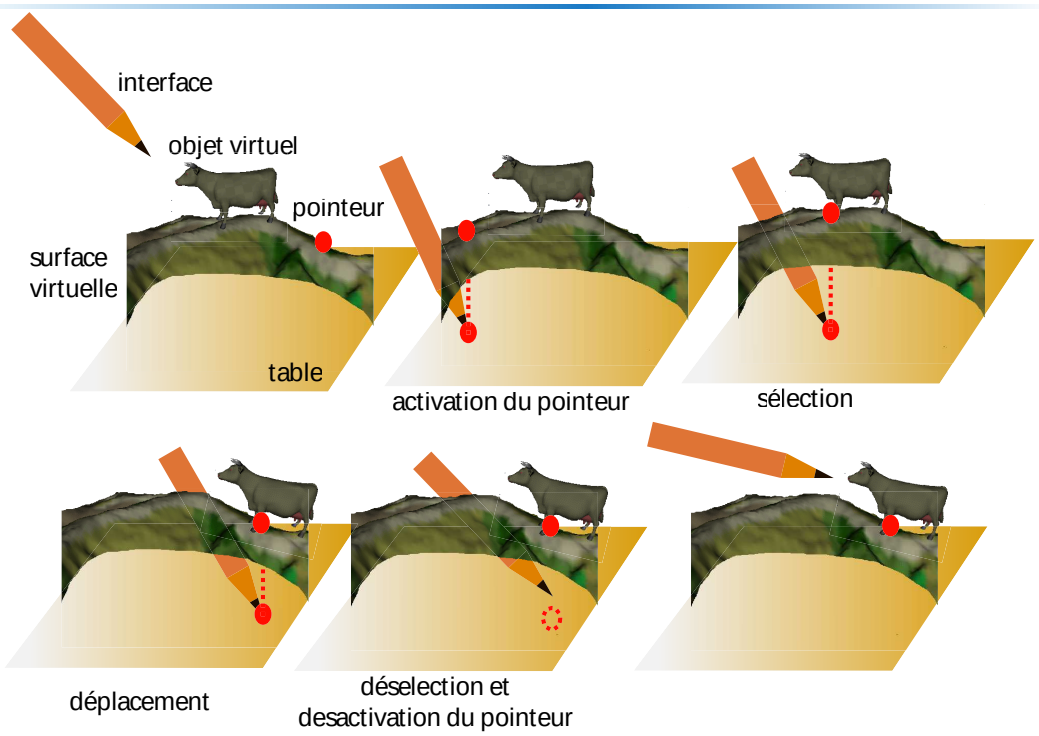


Fig. 6.16: Tâche 2D 1/2 : métaphore *glue and slide*.

La solution que l'on a retenu est la séparation entre le point d'interaction (curseur virtuel) et le point d'action de l'interface lors de la manipulation (e.g. doigt) : le curseur virtuel est associé au déplacement de l'interface, et si l'interface passe sous le modèle (en hauteur), le curseur virtuel est alors contraint au dernier point d'intersection avec le modèle (se colle au modèle), plaçant le système dans un état intermédiaire. L'utilisateur doit alors réaliser un contact avec la table pour rediriger le curseur et peut alors faire glisser le point d'action de l'interface sur la table, le curseur virtuel suivant le profil en fonction de la valeur rendue par f.

Avec ce principe, le déplacement d'un objet se fait par détection de collision entre curseur et modèle, validation et déplacement (métaphore *glue and slide*, figure 6.16). La sélection

peut se faire automatiquement, par temps de pause ou détection de tape sur la table. La désélection est simplement réalisée par un principe analogue ou avec une détection de fin de contact avec la table.

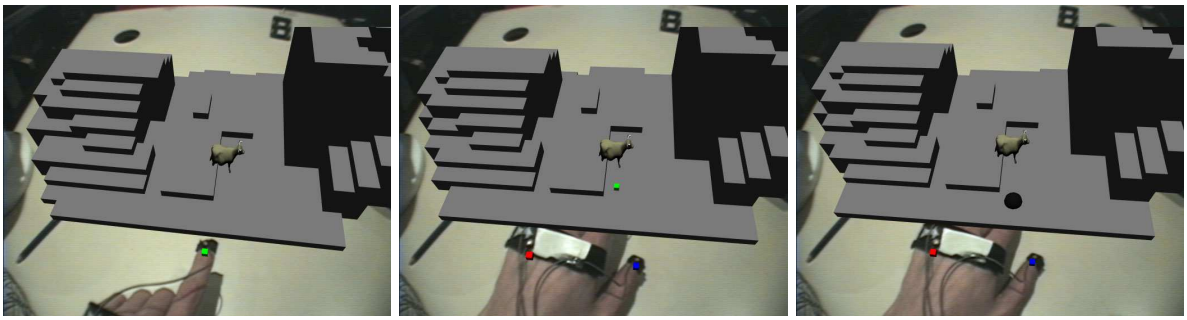
Mise en œuvre

Une question se pose sur la de transition entre le moment de contact avec la table et le passage à travers le modèle conduisant à une perte de visibilité avec l'interface. Nous nous reposons pour cet métaphore sur le fait que l'utilisateur se base principalement sur sa proprioperception de l'environnement physique et la simplicité du mouvement pour effectuer le contact (mouvement vertical).

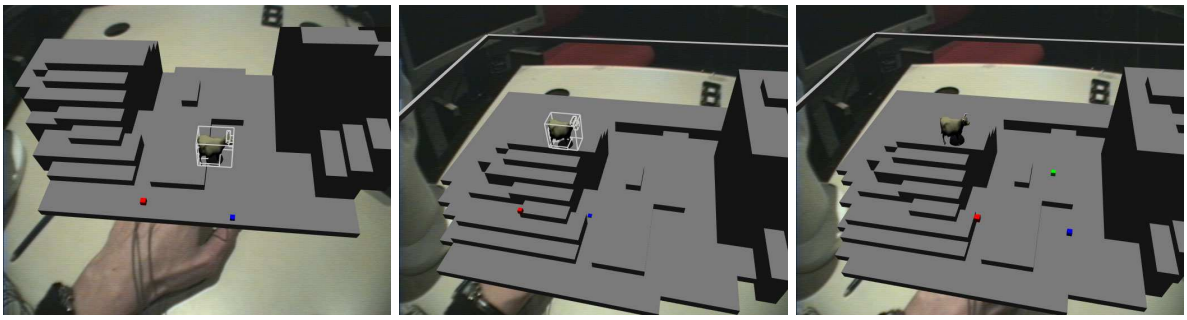
l'éxpérience nous montre en effet que la manipulation de la souris dissociée physiquement de son curseur pose peu de difficultés à l'utilisateur, seule la forte connaissance de la relation de contrôle entre curseur et interface est nécessaire. Une solution envisagée est de rendre transparente la zone de passage du doigt lorsqu'il rentre dans le modèle jusqu'au contact avec la table, mais la modification de l'apparence du modèle peut avoir une influence négative sur la cohérence de la representation pour l'utilisateur.

Résultats

La figure 6.17 présente un exemple d'utilisation. Les différents tests réalisés montrent l'efficacité et la simplicité de la méthode. Nous notons que l'utilisation de la position absolue offre un déplacement plus facile mais une acquisition initiale plus difficile avec le pointeur qu'une approche relative.



(a) doigt hors de la scène (b) doigt au dessus de la scène (c) doigt sous la scène, apparition du curseur (boule noir)



(d) sélection de l'objet (e) déplacement (f) désélection (curseur reste à la dernière position)

Fig. 6.17: Déplacement d'un objet virtuel avec la méthode proposée (avec l'interface digitale, le point d'interaction est représenté par le cube vert).

Cas 2 : Solution multiples pour le placement

Parfois, nous avons plusieurs solutions discrètes pour le positionnement en hauteur d'un élément virtuel. La solution naïve consiste un à laisser mouvement libre à l'utilisateur pour le placement, des techniques de snapping peuvent alors aider à un positionnement

précis. Mais cette technique naturelle peut poser de grande difficultés du au manque de stabilité des interfaces 3D actuelles.

Méthode

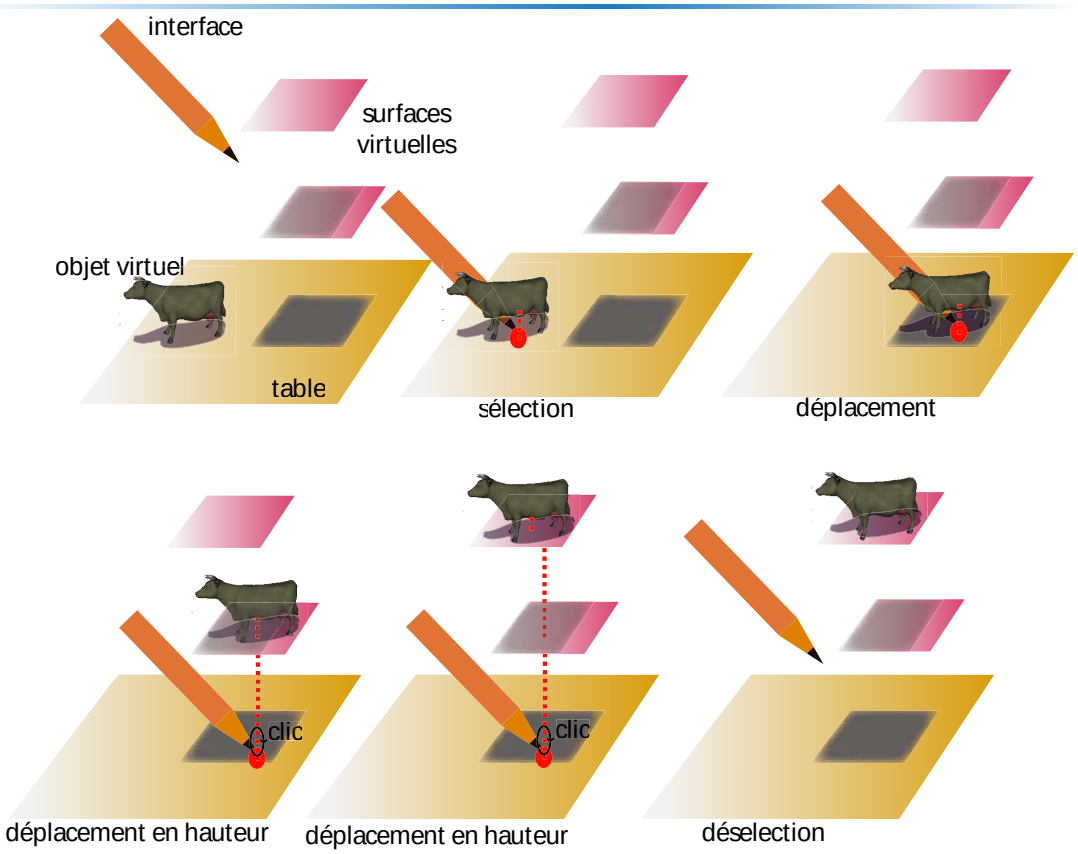


Fig. 6.18: Tâche 2D 1/2 : métaphore pile cyclique.

Nous proposons une nouvelle approche différente (figure 6.18), qui consiste à un déplacement 2D contraint sur la table et le choix d'une des solutions possibles pour le placement vertical par un parcours cyclique des solutions (métaphore de la «pile cyclique»). Cette solution trouve son interet par la plus forte facilité du placement précis d'un élément, laissé à la machine (alors que la position 2D est contrôlé par l'utilisateur). L'autre intérêt se trouve dans la rapidité d'éxécution du cycle par rapport au positionnement vertical : on réduit le déplacement par des opérations discrètes (un clic sur la table en pratique, définissable comme une tape avec l'interface digitale).

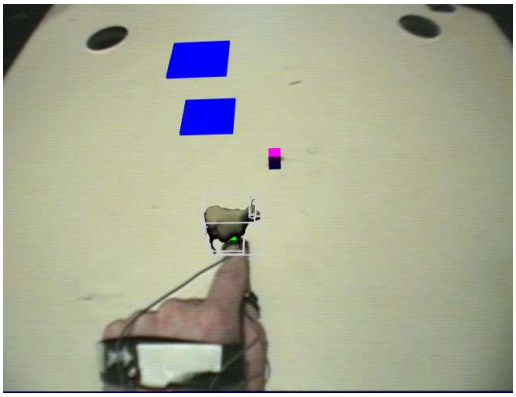
En se basant uniquement sur la loi de Fitz, le gain est conséquent si la hauteur de la cible est largement supérieur à la somme des mouvements de doigt effectués pour le changement de solution. De plus si le nombre de solutions est trop important, il semble alors que la fatigue induite par un nombre répété de clics sur la table réduise fortement la dextérité de l'utilisateur (fatigue musculaire).

Mise en œuvre

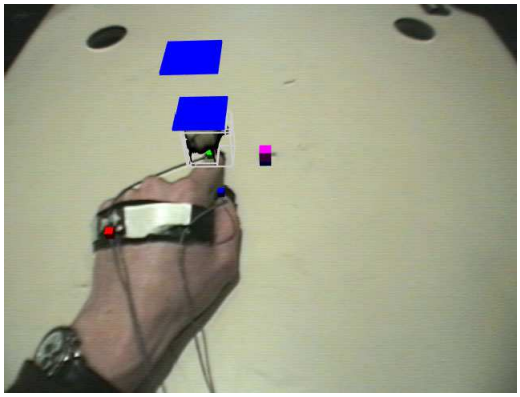
Nous avons réalisé un simple prototype avec l'interface digitale (*lightweight AR glove*) et de multiples surfaces. La réalisation de la fonction de détection de surfaces ce fait par une détection de collision avec une demi-ligne ayant son origine a la position du doigt.

Résultats

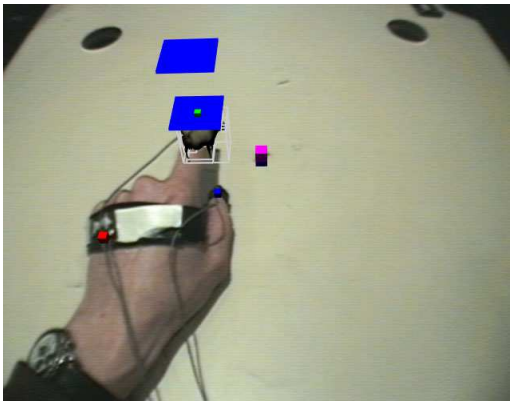
La figure 6.19 illustre l'utilisation de la technique. Il nous semble que la précision de placement planaire et de détection d'une tape est largement facilité par notre système. Une approche étendu consisterait en l'usage d'une détection de pression ou d'un élément mécanique comme un clic à la place des tapes (fatigue possible ?).



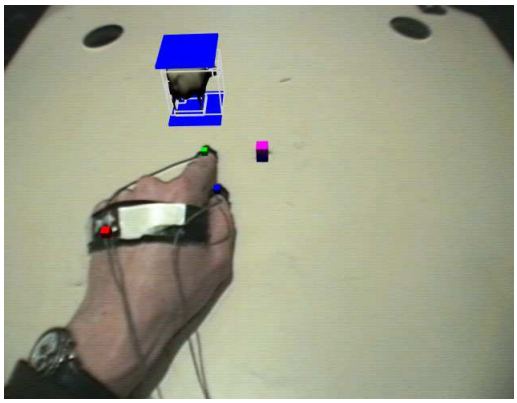
(a) un objet est séléctionné.



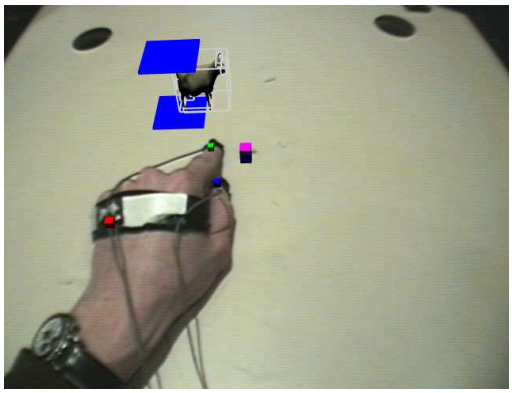
(b) il est déplacé sous le premier niveau



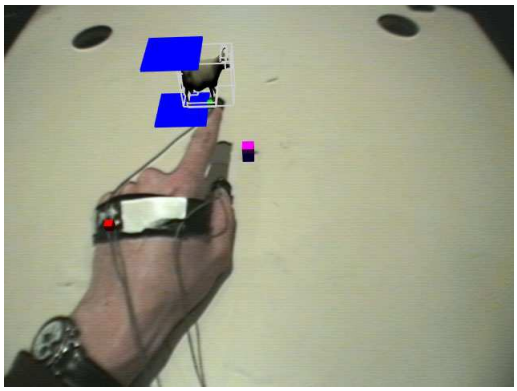
(c) l'utilisateur léve le doigt (point vert en hauter)



(d) et le repose pour valider le déplacement en hauteur



(e) l'utilisateur déplace l'objet au niveau



(f) et peut recommencer la procédure pour un nouveau niveau (il lève le doigt).

Fig. 6.19: Déplacement d'un objet virtuel avec la méthode proposée (avec l'interface digitale, le point d'interaction est représenté par le cube vert).

3.1.5. Technique d'interaction : tâche 3D sur table

Le dernier cas se rapporte à des objets 3D non contraints, pouvant être placés librement dans l'espace de la table tel que pour des applications de chimie (manipulation de molécules), l'astronomie (visualisation) ou d'observation d'un objet («trackball»). Comme précédemment la sélection peut être 3D ou 2D, et l'approche classique consiste dans ce cas à une sélection 3D puis à un déplacement 3D. Nous proposons deux nouvelles approches en se restreignant à l'étude des 3 DDLs de translation.

Méthode : technique *push'n roll*

Notre intuition est qu'une méthode indirecte utilisant la contrainte physique de la table peut être aussi efficace que l'approche naïve. Dans ce but, il faut donc spécifier une méthode contraignant le déplacement sur la table (2 DDL) et qui à partir d'une autre information réalise le déplacement 3D (3 DDL) du curseur dans l'espace (approche nommée 2D+1D). Comme nous le justifierons dans la suite du document, nous nous limitons à des solutions cherchant un déplacement simultané des 3 DDL. De ce fait, nous proposons différentes solutions :

- utiliser les deux mains : une main contrôle 2 DDL et l'autre main peut alors servir à spécifier la dernière composante. La fonction étant alors indépendante ou relative à la position de l'autre main. L'efficacité de cette solution réside alors sur l'avantage des interfaces bimanuelle, comme démontré par Hinckley [HPP+97], Zeleznik [ZFS97], Balakrishnan [BK99] ;
- utiliser plusieurs doigts : un doigt spécifie la position, un autre pouvant alors servir à spécifier la dernière composante, basé sur une fonction relative à l'autre main ;
- utiliser la rotation de la main : la possibilité de rotation de la main peut alors servir à spécifier la dernière composante.

Cette dernière solution, nous a semblé la plus simple à apprendre et à mettre en œuvre. Elle nous semble moins «couteuse» et l'usage d'une interaction à deux mains permet d'effectuer des déplacement simultanés d'objets. Nous avons choisi une approche par interface tangible, la rotation et le transport d'objets réels étant des tâches assez naturelles et très efficaces. Le déplacement sur la table de l'objet spécifiant deux valeurs, la rotation spécifiant la troisième (métaphore **push and roll** illustrée figure 6.20). Le curseur virtuel 3D est alors dissocié de la position de l'interface et permet la sélection et le déplacement 3D (étape de sélection/désélection par geste sur le périphérique, confirmation par clic ou pause).

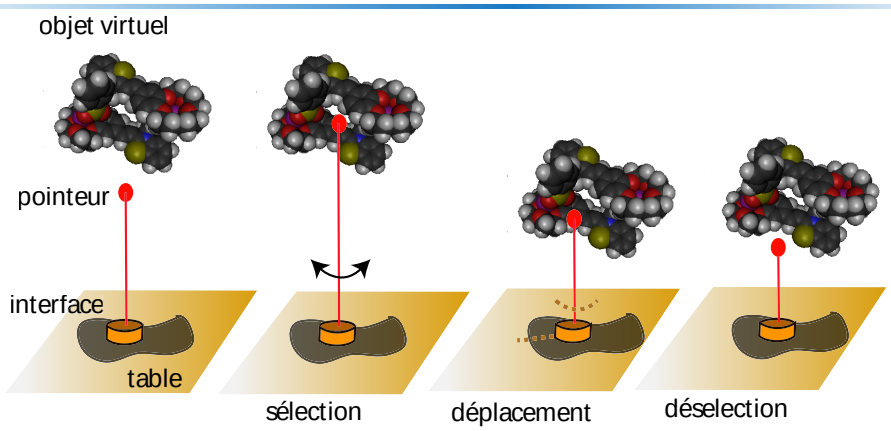


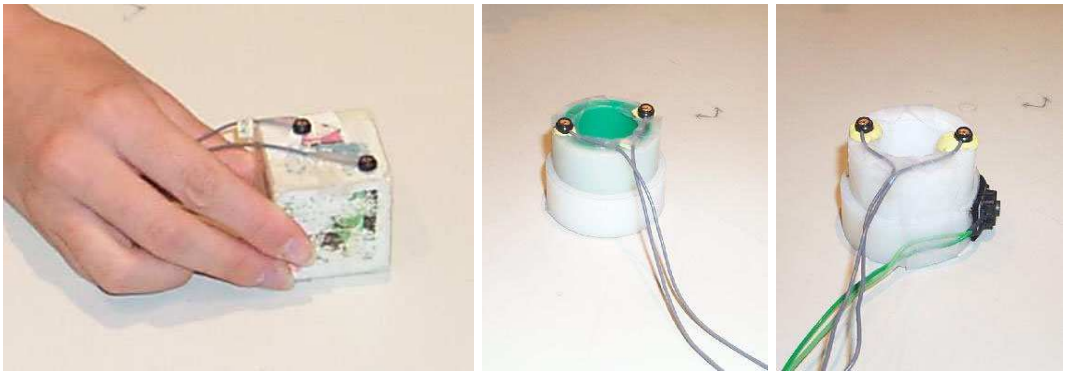
Fig. 6.20: tâche 3D : métaphore *push and roll*.

Ce système s'inscrit dans le concept de souris 3D largement traité dans la bibliographie. Nous pouvons citer *Sensable* [PIHP01], *Rockin'Mouse* [BBKF97], *VideoMouse* [HSH+99], *Two Ball Mouse* [Pal97], *ToolStone* [RS00] ou les prototypes de puck de Wacom [FIB95]. Notre solution s'intégre alors dans une approche complétement nouvelle : appliqué à la RA 3D, dédié à de l'interaction 3D collaborative sur table, dédié à du collaboratif et avec une visualisation 3D avec le même espace d'interaction et de visualisation.

Mise en œuvre

Nous avons réalisé plusieurs prototypes de cette métaphore. Ils sont présentés figure 6.21. Ces premières versions sout toutefois trés limitée d'un point de vue technique et ergonomique par rapport aux précédents projets de souris 3D. La figure 6.22 illustre un exemple d'utilisation dans un cadre de chimie.

Résultat et évaluation



(a) (b) (c)

Fig. 6.21: Les trois prototypes de l'interface pour tâche 3D. (a) version initiale (b) version de base (c) version avec bouton.

Nous proposons d'évaluer cette technique par rapport à une technique dite classique sur les critères d'efficacité et d'utilisabilité de cette méthode.

Nous nous intéressons ici à une tache intégrale, le positionnement 3D en translation (voir section 2.4.1). La méthode proposée est un contrôle en position avec un périphérique isotonique pour des besoins de précision. S'inscrivant dans la logique des souris 3D, elle offre un support physique (retour haptique passif), une facilité naturelle d'utilisation, de bonnes propriétés de saisie et une facilité d'acquisition. Basée sur les résultats de Mason [MWLM01], la relation entre transport et rotation en font un bon choix pour un support intégrale. Contrairement à la *Rockin'Mouse* [BBKF97], cette interface nous semble mettre en relation la même catégorie de système musculaire pouvant permettre une rotation et une translation simultanées. Nous pouvons alors nous interroger sur la validité de plusieurs points de cette approche :

- Est-ce que la disjonction de l'espace de visualisation 3D et la surface d'interaction 2D rend perceptuellement peu compréhensible le système pour l'utilisateur ?
- est-ce que cette interface est intégrale pour l'utilisateur ? Est ce-qu'elle peut le devenir par apprentissage ? Est-ce que l'utilisateur peut coordonner les 3 DDL ?
- Est-ce que la stabilité induite par le contrôle et le support physique peuvent garantir des performances identiques (précision, rapidité, erreur) à un périphérique libre ?
- Est-ce que l'apprentissage peut permettre des résultats similaires à un système libre ?

Nous avons donc réalisé une étude comparant notre méthode avec l'approche de périphérique libre (approche naïve). Cette étude formelle est motivée par l'évaluation des critères de rapidité, précision, stabilité, efficacité, phase d'apprentissage et utilisabilité entre les deux méthodes.

Nous avons réalisés deux expériences : une expérience préliminaire et une expérience. Elles correspondent à une étude de formation[2] (*formative*).

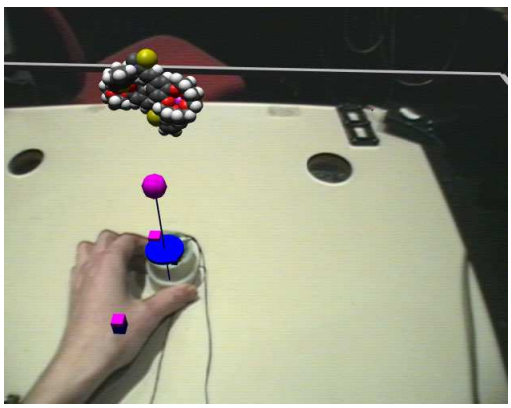
Expérience préliminaire

On décrit maintenant le dispositif, la tâche et le protocole de l'éxpérience.

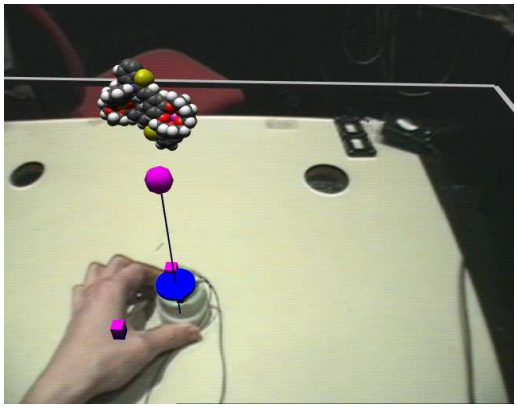
Dispositif (figure 6.23) : l'utilisateur est positionné assis devant la table, équipé du casque semi-transparent optique. Il tient dans la main non-dominante une souris pour la validation par clic des différents tests. Pour l'interface 3D nous avons choisis un cube en bois equipé d'une led. L'interface 2D+1D correspond au premier prototype en bois carré (figure 6.21a).

Tâche : il s'agit d'une superposition de 2 cubes virtuels (*docking*). Le cube cible apparait

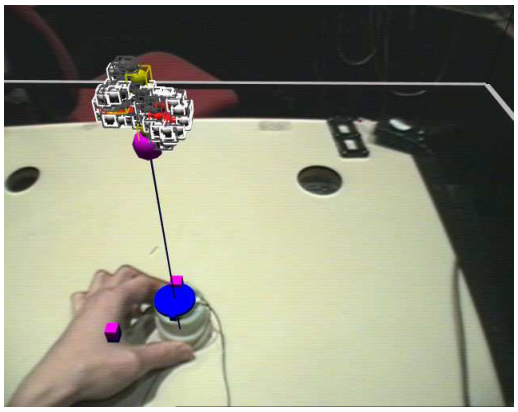
[2]ce différencie d'une étude finale (*summative*) dans le sens ou elle aide à améliorer la solution durant sa réalisation et non à la valider en fin de réalisation



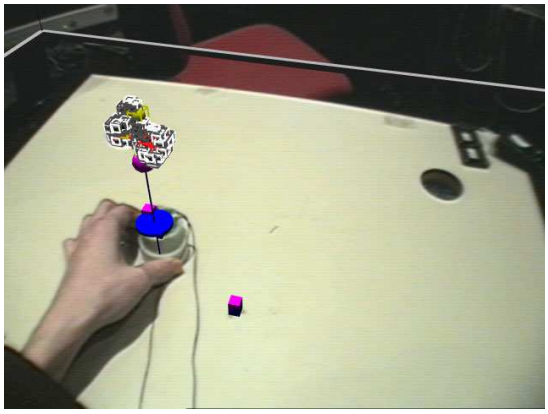
(a) déplacement de l'interface vers la cible



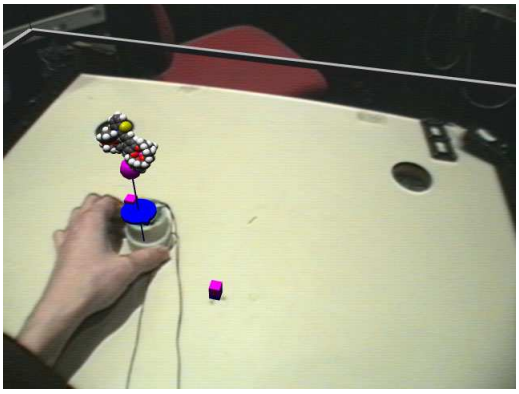
(b) montée du pointeur (rotation de l'interface)



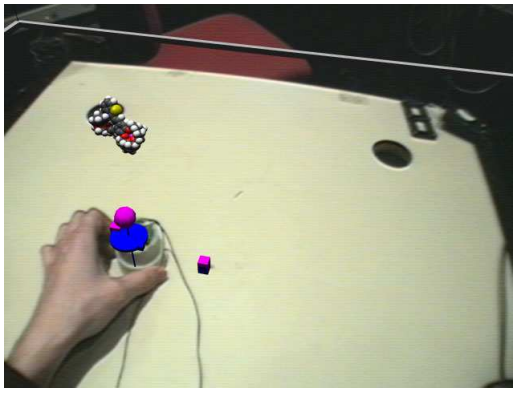
(c) sélection (par temps de pause)



(d) déplacement à une nouvelle position



(e) fin de la séléction (par temps de pause)



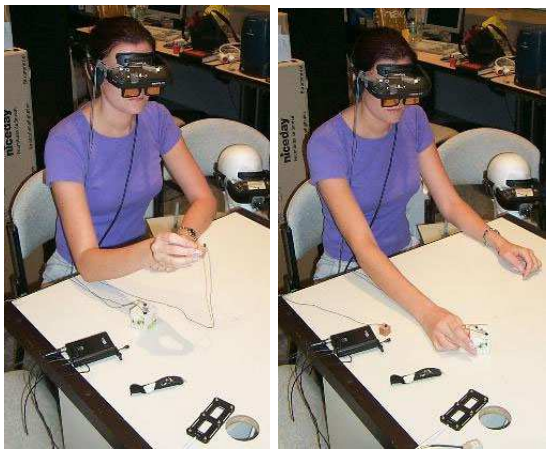
(f) déplacement de l'interface à une autre position

Fig. 6.22: Déplacement d'une molécule 3D avec la métaphore *push and roll* avec le second prototype.

aléatoirement au dessus de la table et l'utilisateur doit alors réaliser l'alignement avec le cube virtuel contrôlé par l'interface testée. Il valide son choix par un clic souris. Un retour sonore confirme la validation.

Procédure : les différents sujets effectuent séquentiellement une session de 50 essais pour chaque interface. Nous mesurons le temps d'exécution et la précision obtenu. Nous observons les sujets pendant l'éxpérience avec une attention particulière à la coordination avec les interfaces. A la fin de l'expérience, nous interrogeons l'utilisateur sur ses préférences et les difficultés éprouvées durant l'expérience.

Résultats de l'expérience préliminaire : la première expérience a été réalisé sur quatre personnes, dont deux qui ont répété l'expérience sur 3 jours. Durant cette expérience



(a) (b)

Fig. 6.23: Configuration d'évaluation : (a) configuration libre (b) configuration 2D+1D (ici avec un ancien prototype).

préliminaire, nous avons choisi de ne pas évaluer la précision obtenu : la validation de l'alignement était effectuée automatiquement à partir d'un seuil de proximité de la cible (évalué avec 0.5 mm et 0.2 mm pour un cube de 1 cm de côté). Les résultats montrent des temps meilleurs pour l'interaction 3D que pour l'interaction 2D+1D. Nous observons une convergence des deux courbes vers un seuil constant (figure 6.24a).

L'usage du casque pose des problèmes de visualisation de la cible. L'analyse des essais à montré deux phases. Nous notons une phase de recherche visuelle de la cible (déplacement de tête) puis une phase d'alignement. Nous avons réduis la première phase (dépendance de visibilité) pour les évaluations suivantes en s'assurant d'afficher la prochaine cible toujours dans l'angle de vue du casque de l'utilisateur.

Dans le cas 3D, nos observations montrent une fatigue induite par le fait d'avoir le bras en l'air, l'utilisateur posant alors intuitivement son coude sur la table comme point d'appui.

Pour l'interface 2D+1D, nos observations montrent aussi qu'en fin de session les utilisateurs semblent avoir plus de facilité à déplacer et tourner simultanément l'interface (confirmant une convergence vers une utilisation intégrale). Nous notons enfin une facilité de compréhension de la technique, un temps d'apprentissage très court.

Les utilisateurs n'ont pas de préférences sur le choix de l'interface. Cette expérience a aussi permis de remplacer le prototype initiale (cube), pour une interface offrant un contrôle plus fin avec une forme circulaire.

Résultats de l'expérience pilote : une expérience a été réalisé sur 3 sujets pour une cible de 2 cm de taille. Le protocole est identique à l'expérience précédente mais avec la validation par l'utilisateur de l'alignement.

Les trois utilisateurs sont des utilisateurs expérimentés des interfaces 3D. La précision obtenue pour la cible 2 cm est meilleur en 2D+1D (0.189±0.14 cm[3]) qu'en 3D (0.30±0.30 cm) pour le premier sujet, identique pour le deuxième sujet (3D : 0.224±0.18 cm, 2D+1D : 0.219±0.156 cm) et inférieur pour le troisième (3D :0.321±0.2 cm, 2D+1D : 0.377±0.125 cm). Dans tous les cas les temps d'éxecutions sont meilleurs en 3D.

Nous avons refait la même expérience avec une cible de 1 cm de taille. Pour un utilisateur expérimenté les résultats sont meilleur en précision en 2D (3D : 0.196±0.15 cm, 2D+1D : 0.22±0.21 cm) et avec des temps d'exécution de faible différence (3D : 0.35±0.64 s, 2D+1D : 2.08±0.44 s). Pour un utilisateur non expérimenté les résultats sont aussi

[3]cette notation nous donne la valeur moyenne et la variance mésurée.

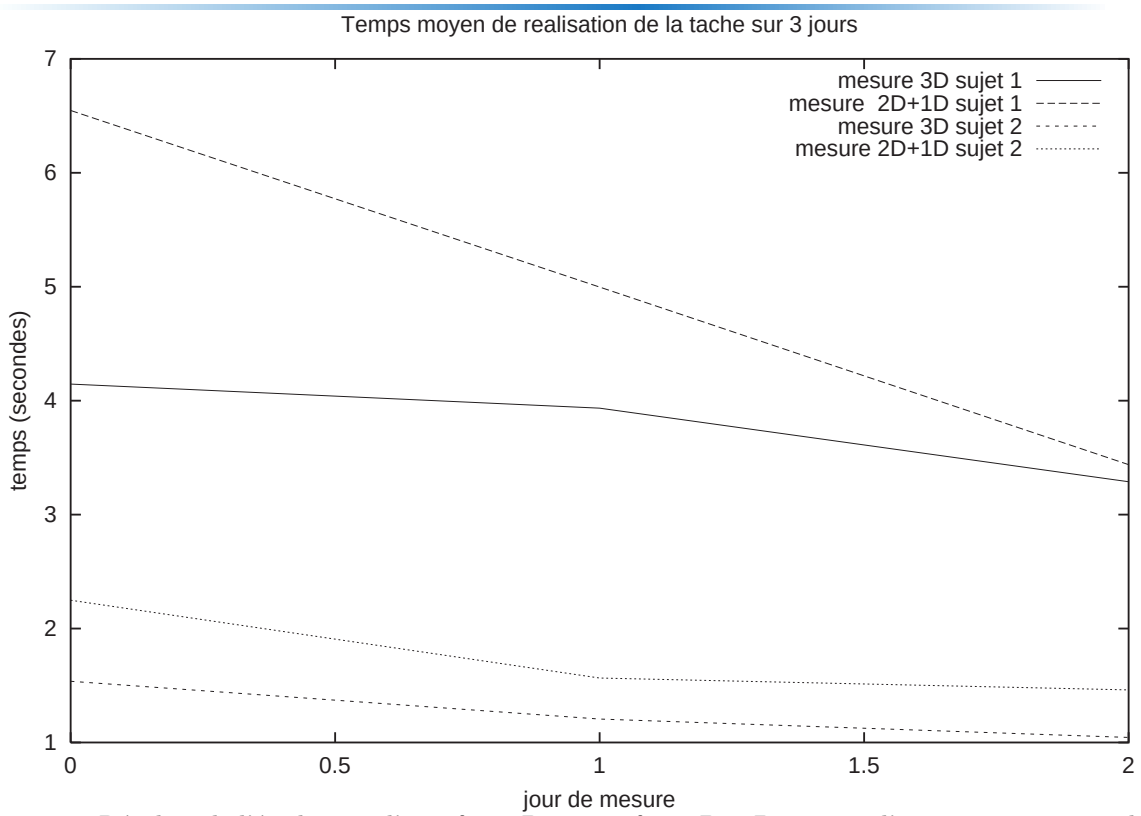


Fig. 6.24: Résultat de l'étude pour l'interface 3D et interface 2D+1D : temps d'apprentissage pour deux sujets avec une précision différentes dans l'etude préliminaire (sujet 1 à 0.2 mm, sujet 2 à 0.5 mm).

meilleur en précision 2D (3D : 0.43±0.50 cm, 2D+1D : 0.62±0.76 cm) mais une plus forte différence en temps de reussite (3D : 6.24±4.84 s, 2D+1D : 4.15±2.13 s).

Nous avons observés le même effet de fatigue avec l'interface 3D que précédemment. Au niveau de l'utilisabilité les utilisateurs n'ont pas de difficultés à appréhender l'interface 2D+1D et à l'utiliser. Une observation des courbes de déplacement montre une meilleur coordination avec l'interface 3D. Nous remarquons cependant que l'interface 2D semblent suivre un mouvement partiellement intégrale : l'utilisateur a tendance à effectuer un ajustement de la dernière composante lors de l'étape de positionnement précis de la cible. Nous observons aussi plus d'oscillation en fin d'alignement en 3D pouvant être du à l'absence de retour haptique (correction essai-erreur par l'individu ?).

Bien sur nous ne pouvons tirer de conclusion à partir du faible échantillon de sujets, mais cette étude nous offre en contrepartie des pistes sur l'efficacité et l'intérêt de la méthode. Les premiers résultats tendent à confirmer notre intuition : facilité de comprehension de la méthode, bon temps d'apprentissage, résultat aussi efficace en 2D+1D. Il nous semble qu'un seuil de performance entre les deux interfaces peut être estimé. Il dépendra principalement de la précision de la tâche, du temps d'utilisation en continu de l'interface et de la répétabilité de l'usage. Une étude plus précise sur le choix de l'interface reste à réaliser (digitale vs tangible).

3.1.6. Conclusion, limitations et perspectives

En nous basant sur nos différentes interfaces, nous avons proposé de nouvelles méthodes d'interaction pour trois types de tâches (tâche purement 2D, 2D1/2 et 3D limité en translation). Les premiers tests et analyses montrent la faisabilité et l'utilisabilité de ces métaphores dans leurs cadres respectives. Une nouvelle conception des interfaces semble né-

cessaire d'un point de vue technique (robustesse, légérété) et ergonomique (sans fils). Les techniques de manipulation doivent bien sur être étudié plus précisément et expérimentés dans un cadre applicactif (évaluation comparative, utilisabilité, performance, collaboratif, etc.).

Un première limitation pour une application de RA sur table est le conflit entre interaction avec le réel et le virtuel. Ce cas non traité, il nous semble que des solutions avec des changements de mode (réel ou virtuel) constiturait une première approche de solutions (avant la définition de méthode automatique).

Notre approche était orientée vers un espace d'interaction fusuinné avec un espace de visualisation. Le point suivant de l'étude pourrait alors être une dissociation complète de ces deux espaces, laissant à l'utilisateur l'interaction dans un espace privé et la visualisation dans l'espace partagé (analogie souris-écran).

D'autre part, des techniques complémentaires doivent être proposées pour les cas de changement d'échelle, ou d'orientation des objets virtuels pour les différentes classes de tâches (jusqu'à 7 DDL pour des tâches 3D).

3.2. Navigation

Nous nous intéressons maintenant à la problématique de la *navigation* en RA. Stoakley[SCP95] définit la navigation comme le contrôle du point de vue dans une scène virtuelle (modification de la position de la caméra virtuelle). Selon Bowman [Bow99] trois types de tâches peuvent êtres distinguées pour la RV :

- **l'exploration** : déplacement sans but précis, principalement pour identifier et découvrir des informations ;
- **la recherche** : déplacement vers une position ou un élément précis ;
- **le manoeuvrage** : déplacement à une position précise afin de réaliser une tâche («on se place dans une bonne position»).

La réalisation de ces tâches s'effectuée principalement avec deux référentiels :

- référentiel **égocentrique** : l'utilisateur est au centre de la scène virtuelle et la position du point de vue est dépendant du mouvement réel des parties actives de son corps (mouvement tête, corps, yeux) ;
- référentiel **éxocentrique** : le point de vue est défini hors de l'espace de la tâche, procurant un regard «extérieur» sur la scène.

Plusieurs travaux ont classés les connaissances spatiales mises en jeu lors des phases de recherche dans le monde virtuel en trois catégories :

- la connaissance du repère *landmark knowledge*) : identifier sa position par des indices visuels ;
- la connaissance de la route *route knowledge*) : identifier les relations entre ces indices ;
- la connaissance de l'horizon (*survey knowledge*) : avoir une connaissance globale de l'environnement.

Très souvent une vue exocentrique complémentaire est fourni à l'utilisateur comme aide à la navigation (carte orientable).

La navigation en RA, dépendant de la perception du monde réel, nécessite des métaphores différentes de la RV : par exemple un pas en RV peut être interprété comme un déplacement à grande échelle mais semble difficile en RA (conflits d'alignements réel et virtuel).

3.2.1. Etat de l'art

De nombreux travaux propose un déplacement à l'échelle lorsque les éléments virtuels sont intégrés à l'espace réel : les mouvements des utilisateurs sont suivis et un alignement garanti alors le bon point de vue.

Studierstube introduit un des premiers systèmes de RA avec déplacement a l'échelle de façon collaborative. Svazalari reprend avec le PIP [SG97a] la possibilité d'afficher un deuxieme point de vue contrôlable par un pointeur 3D. Il propose aussi l'utilisation du PIP comme un *WIM (World In Miniature)* [SCP95] pour changer le contenu d'une fenêtre 3D. Il reprend le concept du *magic-lenses*, et propose la posssibilité de visualiser mais aussi zoomer sur certains éléments (repris recemement par Brown [BH03] dans *SCAPE*).

SCAPE [HBGA03] introduit aussi une navigation multi-vues : le deplacement de pion réel sur une vue exocentrique (par un avatar tangible), permet alors le déplacement d'un point de vue egocentrique projete sur un écran cylindrique.

Fjeld introduit différentes méthodes de navigations dans le cadre de déplacement du point de vue (rendu sur les deux surfaces de projection de son système). Il propose dans ses premiers travaux [FIV+99] le déplacement du point de vue murale par manipulation d'une (ou deux) briques associe à la caméra virtuelle. Il introduit ensuite différente techniques à deux mains [FVB+99] basé sur l'utilisation de 2 briques pour la manipulation du point de vue ego et exo-centrique.

Le prototype *VIP* [AaSR02] propose la navigation par coupe virtuelle d'un modèle basé sur l'utilisation d'un plan de coupe réel.

Kiyokawa a montré que l'immersion était favorisée par la RV, et que la RA était plus approprié pour son côté collaboratif. Il [KTY99b] introduit alors la métaphore de «l'ascenseur virtuel» permettant de passer entre ces deux mondes. Billinghurst étend ce concept et décrit une interface de transition, *MagicBook*, permettant de passer du réel, à la RA, à la RV à l'aide d'un livre avec un marqueur *ARToolkit* sur une page.

Les approches par casque utilisent généralement un référentiel egocentrique pour visualiser des scènes virtuelles, la scène étant positionnée sur une table, dans une piece ou en extérieur.

3.2.2. Notre approche

Nous considérons une scène virtuelle visible dans l'espace de la table. Dans le cadre où les utilisateurs sont assis, les déplacements de la tête seront fortement limités à des rotations et des faibles translations de la tête. De plus, le contenu 3D visible se limite à l'espace de la table.

Deux types de besoins se dégagent. D'une part les utilisateurs doivent pouvoir changer personnellement leur point de vue sur la scène (**navigation locale**). D'autre part ils souhaitent aussi pouvoir sélectionner la scène dans sa globalité et zoomer sur une partie de celle-ci (**navigation globale**). Nous étudierons ici ces deux concepts ainsi que la transition RV/RA proposée par Kiyokawa [KTY99b].

Modélisation

Nous rappelons ici brièvement la modélisation de notre environnement de la même façon que dans le chapitre 5 (figure 6.25). Nous traitons le cas de deux utilisateurs, leur positions étant suivies par des capteurs externes. La positions d'un point de vue est défini par la transformation $p1, p2$. La scène virtuelle s sous ensemble de la scène complète S est quand à elle placée à l'origine du repère de l'environnement, en O.

A partir de cette modélisation, nous introduisons nos différentes nouvelles solutions.

3.2.3. Technique d'interaction : navigation locale

Ce premier type de navigation (figure 6.26(a)) définit une navigation libre de la tête similaire à un déplacement réel (mode de navigation standard en RA). Nous restituons la vision du monde mixte par la suite des transformations conduisant du positionnement de la caméra virtuelle à la position de l'œil.

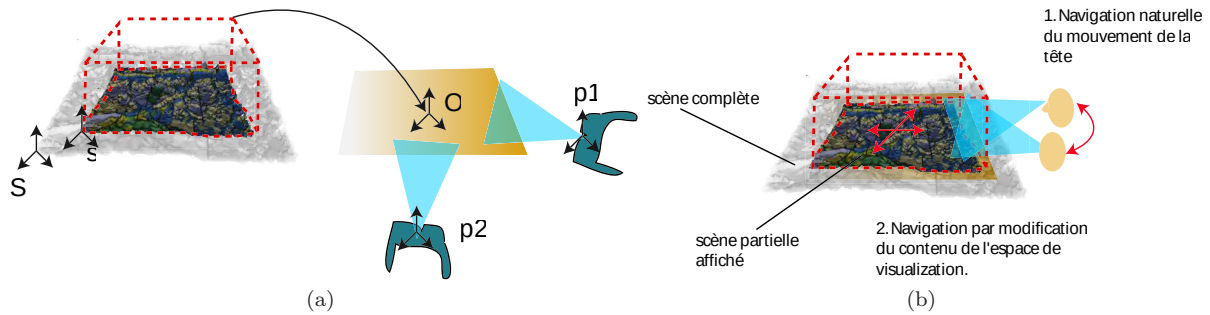


Fig. 6.25: Navigation 3D : (a) modélisation de l'environnement et (b) deux types de navigations : globale (2) et locale (1).

L'espace fourni par cette méthode est donc limité au point de vue de l'utilisateur contraint par les spécificités du système. Les mesures réalisées annexe A montrent un espace de visualisation assez restreint. Ceci s'explique par les limitations physiques du mouvement de la tête et du corps et de la position du sujet par rapport à la table.

De fait, nous définissons en complément deux nouveaux modes de navigation locaux qui permettent à l'utilisateur de visualiser des éléments inaccessibles naturellement (face cachées des objets). L'idée est de conserver les transformations du point de vue (comme le cas précédent) mais d'agir sur les paramètres de la scène vue par un utilisateur (échelle, rotation). Ces deux méthodes cassent alors l'alignement parfait (comme des objets posés sur la table) et servent juste a un support d'observation à courte durée pour un utilisateur et non pour du travail continu.

La seconde méthode (figure 6.26(b)) de navigation est dite inverse, car elle transforme les mouvements de position relative de la tête comme une transformation inverse du contenu de la scène. L'utilisateur peut alors accéder à des angles opposés à son angle de vue,tournant ainsi à volonté autour de l'objet cible. Il peut alors visualiser totalement l'apparence de l'objet. Cette métaphore s'inscrit dans des concepts tels que *scene in hand* [WO90], la scène étant liée ici à la tête de l'utilisateur.

La troisième métaphore (figure 6.26(c)) est le zoom. Il permet à l'utilisateur de zoomer d'un facteur determiné avec un simple déplacement réel. L'utilisateur peut alors spécifier le facteur et visualiser plus précisément l'objet en s'en rapprochant (et optionnellement fixer un point d'intérêt).

3.2.4. Technique d'interaction : navigation globale

La navigation globale consiste à modifier la position, l'orientation ou l'échelle du contenu de la scène (de façon similaire à *BUILD-IT*). Ce déplacement correspond au mouvement de s par rapport à O. On se trouve alors dans un référentiel exocentrique des utilisateurs par rapport au contenu positionné sur la table. La modification est alors publique, répercutés sur la vision de tous les utilisateurs. Nous proposons également une approche *scene in hand* [WO90] où l'utilisateur va modifier la position de la scène virtuelle en fonction de l'espace de la table.

Nous reprenons pour la navigation globale la même décomposition en tâches que celle effectuée pour l'étude de la manipulation 3D et proposons différentes méthodes.

Tâche 2D, 2D5 (figure 6.27) : nous définissons quatre paramètres qui sont x,y, un facteur d'échelle et l'angle de rotation vertical. En 2D5 l'altitude du point de vue s'adapte automatiquement grâce à la conservation d'une hauteur moyenne par rapport à la table. Nous choisissons ici de reprendre un élément d'interface 2D+1D développée précédemment, permettant alors de spécifier le *panning* (x,y), et le zoom (échelle). Comme le montre Jacob

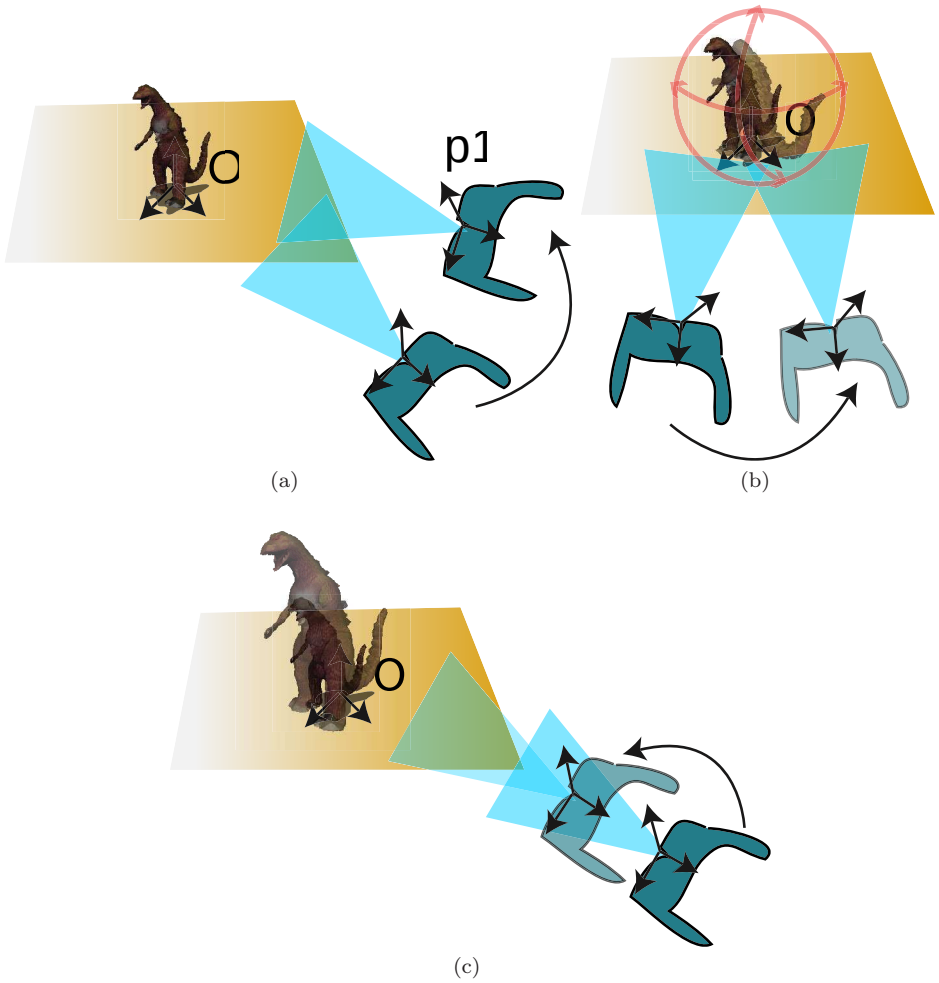


Fig. 6.26: Navigation locales : (a) méthode libre, (b) inverse, (c) zoom facteur.

[JSMM94], il nous semble en effet que ces deux opérations ne sont pas perceptuellement séparables. Notre interfaces permet un contrôle intégrale sur ces critères. La rotation peut quant à elle être fixée à posteriori, ou à l'aide d'un deuxième élément 2D+1D (par exemple en fonction de la traectoire suivi ou de la position de l'utilisateur autour de la table).

Tâche 3D : dans ce cadre il est nécessaire d'obtenir 9 paramètres, position, échelle et rotation. Nous n'avons pas actuellement de solution pour ce cas. Une piste intéressante serait l'utilisation d'une nouvelle interface 2D+1D offrant plus de degrés de libertés (couplant par exemple le système cylindrique à une boule comme utilise dans les

3.2.5. Technique d'interaction : vues multiples et transition RA/RV

Au delà du déplacement du contenu de la scène, il est parfois utile de garder le même contexte, tout en ayant accès à un autre point de vue. Comme l'a utilisée Schmalstieg [SFH+02] cette deuxième vue permet en contrepartie l'utilisation de métaphore telle que *World In Miniature* [SCP95]. Nous considérons dans notre cas l'utilisation d'une seconde vue virtuelle globale servant de carte pour spécifier la position de la vue principale.

En complément, nous proposons de fournir à l'utilisateur une vue publique complémentaire projetée sur l'écran murale fournissant une représentation haute résolution(outrepassant la résolution limité de nos casque) de certains éléments. La position de ce point de vue est alors contrôlé par le déplacement d'un élément 2D+1D sur la table, de façon identique à *SCAPE* qui utilisait des jetons suivi par vision.

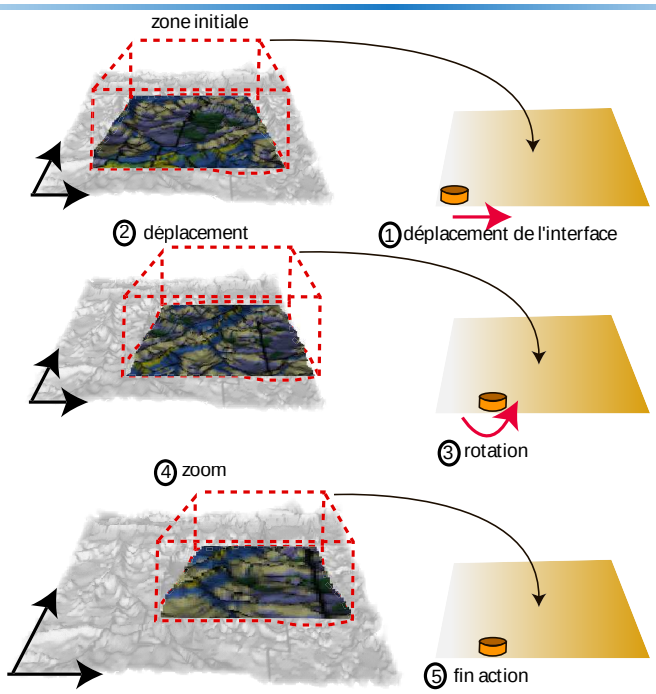


Fig. 6.27: Navigation globale : interaction sur tâche 2D.

Le dernier point concerne le passage d'un mode RA à un mode RV. Nous utilisons une méthode nommée «*dive in the scene*» développé précédemment dans [Gra99b], permettant une transition en douceur entre RA et RV contrairement aux approches de Bilinghurst [BKP01a] ou Kiyokawa [KTY99b]. L'utilisateur indique un point spécifique de la scène et se retrouve immergé dans cette dernière (point de vue égocentrique).

3.3. Contrôle d'application

Ce dernier point concerne l'interaction avec les tâches définies comme *contrôle d'application*. Bowman [Bow99] spécifie celle-ci comme étant les tâches de changement d'état d'un système ou du mode d'interaction (modification des options de visualisation, changement d'outils, etc.). Il propose quatre types de métaphores utilisables (figure 6.28) :
– interfaces **graphiques** : utilisation de widget, de menus ([GC01]) ;
– commande **vocale** : usage de la reconnaissance vocale ;
– interaction **gestuelle** : spécification des choix par reconnaissance de geste ;
– **outils** : utilisation de différents outils (par multiplexage spatial ou temporel).

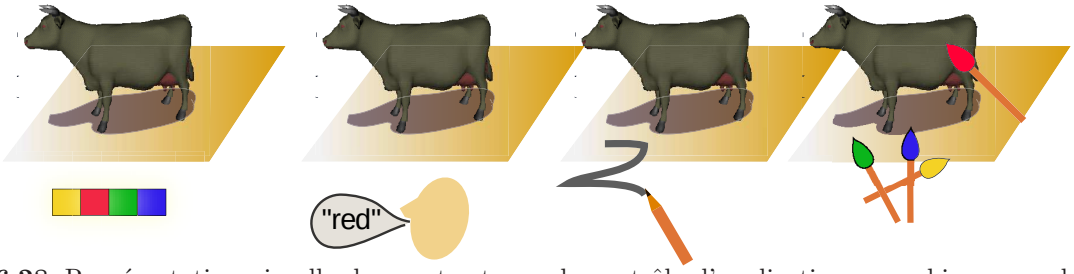


Fig. 6.28: Représentation visuelle des quatre types de contrôle d'application : graphique, vocale, gestuelle, outil. L'exemple de tâche étant ici la modification de la couleur d'un modèle virtuel.

3.3.1. Etat de l'art : analyse

La RA a repris un grand nombre des métaphores développés en RV, cependant le fait de n'être pas limité par une vision virtuelle 3D permet de profiter pleinement des interfaces numériques 2D (PDA, laptop, utilisé de plus en plus dans les environnements semi-immersifs).

Sujet encore peu étudié, l'approche consiste généralement à reprendre les techniques de RV ou d'IHM : menus hiérarchiques 2D ([RFK+97]), boîte à outils, reconnaissance de gestes ([EBSC99]).

Le travail sans conteste le plus abouti reste l'introduction du PIP [SG97a], palette physique associée à des interface graphiques virtuelles d'une application. Elle peut alors fournir à l'utilisateur de multiples fonctionnalités : zone privée pour la manipulation d'éléments, zone de sélection d'outils, de changement d'applications, zone de reconnaissances de gestes, etc.

L'utilisation d'interface tangible constitue une autre approche par association d'un outil à un marqueur (comme dans Tiles [PTB+02]) ou des paramètres à un marqueurs ([SPVT01]) ou de menu 2D sur les faces d'un cube réel ([KvL04]).

3.3.2. Notre approche

Nous avons retenus deux classes de métaphores - menus et boîte à outils - que nous présentons dans la suite de cette partie.

Ces métaphores sont utilisées dans notre environnement pour :
- Changer les outils et les différentes techniques d'interaction ;
- Changer les options d'un outil ;
- Changer les options de l'application : paramètre de visualisation,d' état,de mode ;
- Ajouter/supprimer/modifier des données dans l'espace de la table (modèle 3D, images,texte).

Menus : vers les concepts 2D

Nous choisissons d'utiliser la surface de la table comme support d'interaction. Dans ce cadre nous avons porté une attention particulière sur les méthodes 2D existantes que nous adaptons à notre contexte. Nous avons différencié menu privé/public et menu contextuel se rapportant à un objet ou à l'application.

Pour les menus privés/publics, nous utilisons la surface de la table comme support : un retour tactile, une possibilité d'orientation/positionnement libre. Il nous semble en effet qu'une approche par tablette suivie en 3D (tel que le *PIP*) trouve son équivalent dans l'utilisation de la table dans notre environnement (accès aisé à sa zone privée, même propriétés haptique, réduction de la fatigue de tenir la tablette).Les menus peuvent alors être placés soit dans la zone privée d'un utilisateur ou soit dans la zone publique.

Nous avons mis en œuvre une boîte à outils 3D mais avec un fonctionnement 2D de boîte à outils graphiques de bureau (e.g. *Qt* [Tro]). Elle reprend les concepts classiques de GUI : ascenseur, bouton poussoir, cadran, menus hiérarchiques 2D, case à cocher, case à options. Des exemples sont présentés figure 6.29. Nous affichons en complément des boîte à outils classiques sur les surfaces physiques (e.g. écran de groupe) de notre environnement.

Nous proposons également l'utilisation de menus contextuels co-localisés sur l'objet et donc spécifiques pour chaque objet. Il réduit les déplacements pour l'accès aux outils dans les menus de la zone privée d'un utilisateur. L'état de visibilité temporelle du menu peut être défini à trois niveau (figure 6.30) : toujours visible (**statique**), visible par activation **explicite** (clic) ou activation **implicite**.

Nous proposons différentes représentations de ces outils (figure 6.31). La première solution reprend les concepts d'*Open Inventor* en disposant les éléments autour de la boite englobante. La deuxième solution est l'utilisation des métaphores d'ombres symbolique proposé précédemment dans la partie manipulation.

(a) (b) (c)

Fig. 6.29: Widget 3D placés dans l'espace privé d'un utilisateur (vue à travers le casque optique) : (a) et (b) ascenseur (c) boutons poussoirs.

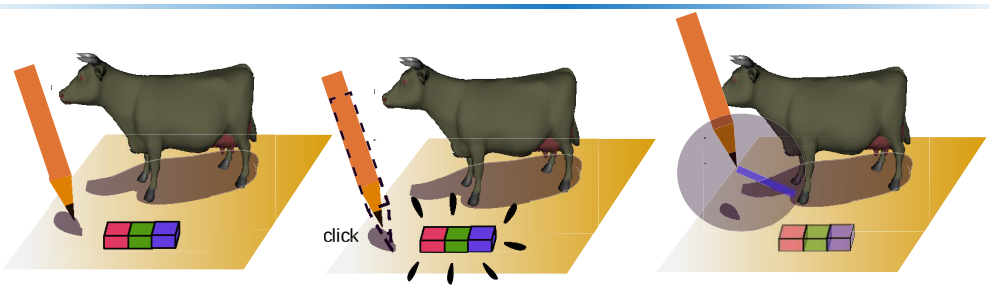


Fig. 6.30: Menu contextuel co-localisé à un objet : statique, explicite, implicite.

(a) (b) (c)

Fig. 6.31: Différentes représentations de menus contextuels : (a) outils sur la boite englobante 3D (b) représentation simplifié de l'ombre (menu circulaire) ou (c) point local.

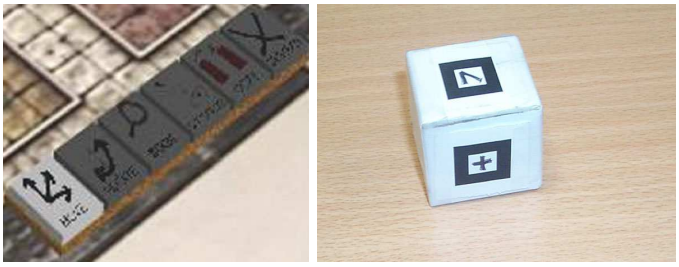
Enfin nous introduisons des menus contextuel (*popup menu*) globaux (spécifiques à une application) reposant aussi sur la surface de la table. Nous nous somme limités aux *pie menu* placés sur la surface de la table (efficace et offrant retour tactile).

Multiplexage de la boîte à outils

Nous choisissons une limitation du multiplexage spatial extrême, en adoptant une stratégie intermédiaire, similaire à celle proposé par Fitzmaurice dans sa thèse [Fit96].

Dans un cadre de multiplexage temporelle nous avons repris le concept du *toolchest* (palette d'outils) de Cutler [CFH97]. Nous utilisons une barre d'icones d'outils virtuels associables à nos differentes interfaces (figure 6.32(a)).

Nous proposons aussi l'utilisation du multiplexage spatial en se basant sur les interfaces présentées chapitre 4. Nous introduisons ainsi l'utilisation d'interface multiples faces, chacune des faces étant associées à un outil (similaire au prototype *flipbrick* de Fitzmaurice [FIB95] et étendu dans Rekimoto [RS00]). Chacun des outils est activable par positionnement de l'outil virtuel actif sur la face opposée à celle en contact sur la table. L'utilisateur a accès à 6 outils par face, lui procurant une large palette d'outils avec 2-3 cubes (au delà, la latence et la fiabilité du système sont trop fortement dégradées).



(a) (b)

Fig. 6.32: Outils : (a) barre d'outils virtuels (multiplexage temporelle) et (b) différents outils sur les face d'un cube (multiplexage spatial).

3.4. Tâches spécifiques : ajout d'éléments et saisie de texte virtuel

Nous étudions maintenant en détails deux tâches particulières identifiées dans un cadre de réunion : ajout de données en cours de session et saisie de données alphanumériques.

3.4.1. Ajout d'élément virtuel

La tâche consiste en la sélection d'un objet virtuel (après éventuellement navigation dans une liste), puis son placement avec les techniques de manipulations déjà présentées. Pour ce faire, nous introduisons trois méthodes complémentaires (figure 6.33) : l'utilisation d'outil **tangible**, de **menus hiérarchiques** et de **reconnaissance gestuelle**, représentant une gradation dans la difficulté d'usage, la dernière étant dédiée aux experts.

La première méthode repose sur la métaphore du cahier introduit par Kato [KBP+00]. Il consiste en l'association d'éléments virtuels à des positions d'un cahier réel. L'utilisateur peut ainsi parcourir les pages du cahier réel et facilement sélectionner des éléments. Ils sont ensuite selectionés et placés grace à une de nos interfaces. Les menus hiérarchiques permettent quant à eux une sélection iconographique des éléments.

La dernière méthode basée sur une reconnaissance gestuelle 2D, par tracé de symboles sur une surface étant ici la tablette digitale. Une association symbolique d'un geste à un élément virtuel permet son ajout. L'utilisateur a un retour visuel du geste effectué et de sa réussite, le contour de l'objet devenant rouge en cas d'échec.

(a) (b) (c)

Fig. 6.33: Méthode d'ajout d'objets virtuels sur la table : (a) interaction tangible (b) interaction classique (menu hierarchique) (c) interaction par geste.

Cette derniere approche servira principalement à l'ajout d'objets virtuels majeurs d'un usage courant. La raison principale est la difficulté d'apprentissage d'un grand nombre de geste différents pour l'utilisateur ainsi que le nombre d'éléments identifiables par la librairie (une trentaine de caractères avec *unistroke*).

3.4.2. Saisie de texte alphanumériques pour annotations

Dans le contexte d'une réunion la tâche d'annotations et de prise de notes nous semble important. Nous souhaitons donc fournir à l'utilisateur un moyen d'annoter des éléments virtuels 3D/2D de façon naturelle et intuitive.

Fig. 6.34: Saisie de texte sur la tablette.

Notre choix s'est porté sur l'utilisation de la tablette digitale, car elle permet de reproduire la métaphore du stylo et papier (figure 6.34). La résolution de ce type d'outil ne nous offre pas actuellement les performances d'un outil réel (résolution stylo/tablette, latence) mais reste utilisable pour des annotations (texte ou croquis).

Aprés validation, l'utilisateur peut placer un élément tracé à une position précise sur un objet virtuel. Le contenu est mémorisé sous forme vectorielle.

4. Conclusion

Au cours de ce chapitre nous avons présentés les différentes méthodes d'interaction que nous avons introduit dans notre système MARE formant une base complete (par rapport à ce qui existe dans le domaine exploré). De plus, nous espérons également avoir montré que cette base est orthogonale, au sens où les outils élémentaires sont conçus pour coopérer efficacement dans la création d'interfaces plus complexe et d'applications spécifique. Ces travaux ont été présentés dans [GG02].

Sans aucun doute l'ensemble définit une proposition qu'il est nécessaire d'évaluer plus en détails au cours d'évaluations formelles et comparatives. Comme on a pu le voir le réel nous offre de nombreuses possibilités mais induit en contrepartie de nouvelles problématiques à résoudre.

Chapitre 7

Interaction virtuel : multiples espaces et multiples surfaces

1. Introduction

Nous avons proposé dans la chapitre précédent des techniques d'interaction dédiées à la manipulation et à la navigation avec une scène virtuelle, définissant un contexte unique de donnée (2D et 3D). Nous avons défini ces techniques comme «bas niveau», modifiant les paramètres intrinsèques des objets.

Il nous semble intéressant d'aller beaucoup plus loin ici, en proposant une interface qui pourrait être définie comme «haut niveau». Ce type d'interface serait équivalent aux interfaces de «bureau numérique» (gestion des fenêtres) présentes dans la plupart des systèmes d'exploitations actuels. Ce type d'environnement généralement restreint à un utilisateur, un écran (un espace vertical 2D), une souris, pourrait être étendu ici dans le cadre d'un environnement sur table, avec de multiples périphériques, multiples espaces, multiples surfaces d'interactions virtuelles ou réelles, 2D ou 3D.

Feiner [FS91] a introduit ce concept sous le terme d'*Hybrid User Interface*, l'hétérogénéité étant à de multiples niveaux (contenu, système d'affichage, d'interaction, utilisateur, etc.). A ce jour, peu de travaux se sont intéressés à fournir une interface pour ce type d'environnement et encore moins dans un cadre collaboratif en RA sur table. Nous nous intéresons ici à l'étude de ce concept pour *MARE* et à la proposition d'une telle interface.

Nous nous attacherons aux problématiques suivantes (figure 7.1) :
- Quelle(s) type(s) d'interface(s) fournir pour la manipulation de multiples contextes dans le cadre d'une réunion en RA sur table ?
- Comment gérer de multiples visualisation 3D ou 2D, de données 2D ou 3D, réelles ou virtuelles ?
- Comment gérer de multiples périphériques d'entrées ? l'interaction simultanée ?
- Comment représenter une application, un contexte ? réel ou virtuel ?
- Comment gérer les préférences des utilisateurs ?

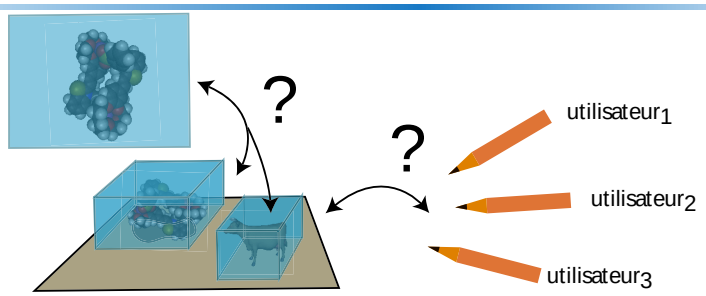


Fig. 7.1: Problématique de l'interaction hybride : comment gérer de multiples contenus ? comment gérer de multiples points d'interaction ?

Dans ce but, nous avons réalisé une observation des environnements de bureau numérique et les environnements réels pour voir si une schéma général et un certain nombre d'éléments pouvaient être utilisés dans notre cadre. Toutefois, comme l'énonce Dumas dans sa thèse [DDC+99], la définition de nouveau type d'interface de haut niveau nécessite de repartir

à zéro tant le contexte est spécifique et nouveau.

2. Analyse et approche

L'observation montre que l'élément clef est la présence inhérente du *document* sous forme papier dans l'environnement réel, sous forme de fenêtre dans les bureaux numériques. Dans l'environnement réel, l'outil informatique n'a pas encore remplacé l'usage de document papier, ses propriétés interactives restant très fortes autant dans un usage personnel que dans un cadre collaboratif.

Nous observons que les environnements de bureau numérique ont reproduit initialement le concept de bureau réel, les métaphores étant similaires. Malheureusement, sans évolution depuis de nombreuses années, elles ne semblent pas appropriées aujourd'hui à un support des futures générations d'environnements (collaboratif, ubiquitaire, RV, RA).

Il nous semble alors judicieux de pouvoir mixer les avantages de manipulation de documents réels (modifiable, transportable, orientable) avec les avantages apportés par le virtuel (représentation 3D, dynamique, multimédia). Notre choix consiste à reproduire les métaphores des réunions réelles en le complétant par les possibilités du virtuel.

Nous notons bien évidemment qu'on ne saurait définir un formalisme conséquent sur ces observations purement informelles. Elles nous guident juste ici sur notre propre identification de certains éléments clefs. Nous présentons dans la suite une partie des travaux liés à ce contexte.

3. Etat de l'Art

Notre étude des nouveaux types d'interface haut niveau se décompose en quatre parties :
– étude des extensions de bureau numérique ;
– étude des environnements en informatique ubiquitaire ;
– étude des environnements en RV ;
– étude des environnements en RA.
Nous observerons principalement les interfaces proposées, les architectures et les relations entre les multiples éléments des interfaces utilisateurs hybrides.

3.1. Extension du Bureau numérique

Myers [MSG98a] a introduit une taxonomie identifiant les critères d'un gestionnaire de fenêtres dans le cadre du bureau virtuel. L'arrangement de fenêtre fut étudié par Bly [BR86] comparant le mode *tiled et overlapped*. Dans sa suite Kandogan [KS97] proposa le concept de fenêtres élastiques s'ajustant automatiquement à l'espace libéré. Badros [BNB00] propose l'utilisation de contraintes paramétrables par un langage fonctionnel, fournissant une extension de l'arrangement des fenêtres. Quand à Bell [BF00] il propose une nouvelle approche permettant un gestion automatique des fenêtres avec des règles de positionnement automatique.

Mander [MSW92] introduit un nouveau concept, la métaphore en pile, basé sur l'analogie avec des documents réels. Beaudouin-Lafon [BL01] introduit de nouvelles techniques d'interaction pour fenêtre : des fenêtres en onglet, des fenêtres orientables, et des fenêtres «zippantes» offrant plus de souplesse à l'utilisateur pour la manipulation de fenêtres.

Différents travaux se sont intéressés à remplacer l'espace 2D, par un espace 3D. *Information Visualizer* [kCRM91] montre qu'un espace 3D favorise l'accès lors de la manipulation de larges quantités de données. Dans ce contexte Robertson [RCL+98] montre qu'une représentation 3D améliore largement la facilité de manipulation de documents par rapport à un explorateur classique de type *Microsoft Explorer*©. Nous pouvons citer le prototype

Task Gallery [RvDR+00], utilisant un espace cubique et un système de conteneur 3D, associé avec une palette virtuelle. Le projet *3DWM* [Elm] s'intéresse à reproduire un bureau virtuel 3D, offrant la possibilité de manipuler des documents 2D placés sur des surfaces de la scène.

Le projet *VideoWorkspace* [Rou02] présente quant à lui une redéfinition des environnements classiques de gestion des fenêtres à partir d'une accélération graphique 3D. Il réalise alors un prototype sur la librairie openGL permettant de définir des fenêtres avec des formes quelconques, déformables, ou transparentes.

L'ensemble de ces concepts ont été étudiés uniquement dans le cadre d'environnement de bureau avec un affichage vertical contraint à un écran. Les éléments prépondérants sont l'apport d'une gestion d'agencement automatique, de dynamique de positionnement des fenêtres et des possibilités d'affichage 3D. Dans notre cadre nous reprendrons ces concepts les étendant à un aspect multi-utilisateurs avec une gestion par graphe de scène des relations entre éléments (en se basant aussi sur un affichage par accélération graphique openGL).

3.2. Approche informatique ubiquitaire

Cette partie reprend en partie les commentaires introduits dans le chapitre 3. Rekimoto [RAH98] propose l'utilisation de multiple surfaces avec une transition d'éléments entres ces surfaces par des interfaces pointeur lasers ou «pointeurs extensible» (*hyperdragging*).

Le projet *ROOMWARE* [STMTK01] propose de multiples surfaces avec une architecture logicielle reprenant les concepts d'aire d'affichage, d'espace de travail et de données (l'espace de travail définissant l'équivalent d'une fenêtre). Le couplage physique de table est une métaphore pour l'assemblage d'espaces de travail. Quant à la transition dans *interactive workspace* [BRTF02], elle repose sur un pointeur multi-surfaces (*PointRight*) ou l'usage de liens hypertextes entre les surfaces (*Multibrowsing*).

Récemment, le projet *Gloss* [Glo] propose une gestion multi-surfaces physiques permettant l'existence d'une surface logique continue entre différents éléments. Basée sur une description ontologique [CLDC03] de la notion de multiples surfaces, Lachenal [LC03] propose une architecture logicielle à partir de clusters de ressources. Leur système permet un assemblage dynamique de surfaces.

Aucun de ces travaux ne propose le concept d'espace 3D, ni de gestion de contenu 3D dédié. Mais ils proposent des architectures logicielles intéressantes que nous reprendrons et étendrons à la notion d'espace.

Shen [SVFR04] propose *DiamonSpin* une architecture logicielle pour la gestion de documents numériques pour une interaction sur table numérique. Les principes de modularité et de flexibilité de la librairie (découpage en conteneur, vue et fenêtre) nous servirons pour la mise en œuvre d'une approche similaire.

3.3. Approche RV

Peu de travaux ont été menés en RV pour la gestion de documents ou de fenêtre. Larimer [LB03] propose *VEWL*, une architecture logicielle pour la gestion de fenêtres dans un CAVE à partir d'un espace sphérique logique pour le positionnement des fenêtres.

Le prototype CRYSTAL [TL97] représente le travail le plus évolué avec une architecture logicielle pour la gestion de multiples contextes. Il propose en complément un gestionnaire dynamique de fenêtres et une interface de programmation très simple.

3.4. Approche RA

Dans le cadre de l'interaction homme machine, le projet *LivePaper* [RR01] propose d'augmenter des documents papiers réels avec des informations digitales. Il introduit les *transparencies*, élément lien pour l'association réel-virtuel. Ils peuvent contenir des menus d'outils (attachés au document papier à l'extérieur de sa surface), l'affichage projeté d'un modèle 3D et la réplication du contenu à distance.

Rekimoto introduit les *DataTiles* [RUO01], qui définissent une association entre un élément réel 2D rigide et du contenu virtuel 2D projeté sur cet élément. Il introduit le couplage par contact physique des éléments, contraint à un positionnement sur une grille. Nos travaux s'inscrivent fortement dans ce principe.

Dans le domaine de la RA 3D, Feiner [FMS93] introduit la gestion de multiples fenêtres définissables dans quatre référentiels : dans le monde, sur un objet suivi, sur l'écran ou dans un espace virtuel entourant l'ordinateur. *EMMIE* [BHF+99] introduit un espace virtuel au-dessus de la table («ether») et le transfert entre de multiples peripheriques, de l'espace à l'écran, de l'écran au mur. Ces systèmes nous semblent trop restrictif, ne fournissant pas une véritable modélisation des échanges et limités à un seul espace 3D.

Le projet Tiles [PTB+02] introduit l'utilisation de surfaces tangibles et le couplage entre elles pour agir sur des objets 3D (copie, aide ou effacement). Le couplage est limitée aux objets sans l'introduction d'un haut niveau d'interaction ou de gestion d'applications plus complexe.

Le projet *Magic Meeting* [RWB02] propose l'association entre espace et surface. Principalement définis sous forme de démonstration, ils proposent le transfert entre PDA et un espace 3D, le déclenchement d'affichage d'éléments 3D dans cet espace lors d'une présentation 2D ou l'association de fenêtre 2D à des surfaces tangibles.

Dans la lignée de *Crystal*, l'environnement *Studierstube* [SFH00] propose une architecture logicielle haut niveau supportant de multiples fenêtres 3D (déjà décrit dans le chapitre 3). Schmalstieg propose l'utilisation de fenêtres tangibles pour la migration d'application. Il associe un marqueur réel avec du contenu virtuel, personnification physique pour la manipulation du contenu à une position donnée. Ils limitent son usage pour la migration entre différentes machines ou le déplacement dans un environnement hétérogène. Notre approche est similaire, mais orientée pour des réunions sur table avec une extension de leur première étude du concept de fenêtre 3D.

Dernièrement le projet *ARWin* [DNH03] propose une extension de l'environnement de bureau par la RA, proposant l'association d'application 3D/2D à des marqueurs réels pouvant être positionnés dans l'espace du bureau. Il propose une architecture reprenant fortement les concepts de *X-Window*. Il réutilise le concept de proximité pour faire communiquer des éléments ici entre des applications (pouvant être réliées par des flots de données virtuelles).

Au vue de cette étude, on remarque qu'un grand nombre de nouveaux concepts valides ont été proposés avec des optiques très différentes. Nous remarquons de plus que peu d'architectures logicielles ou de modèles n'ont été proposés en RA 3D. Nous pensons que l'émergence de supports de visualisation d'espace 3D telle que le *Virtual Showcases* doivent alors être dès maintenant gérée dans une même continuité que les surfaces 2D.

En conséquence, nous choisissons de proposer un environnement avec l'idée d'unifier les différentes approches (figure 7.2) en se focalisant sur le concept d'interaction sur table multi-utilisateurs. Au vue de la complexité du problème nous nous limiterons à la proposition d'un premier modèle et à la description d'un prototype simplifié.

4. Nouvelle approche : un autre pas vers l'hybridité

Notre approche se base sur une extension d'une description ontologique de surfaces (sec-

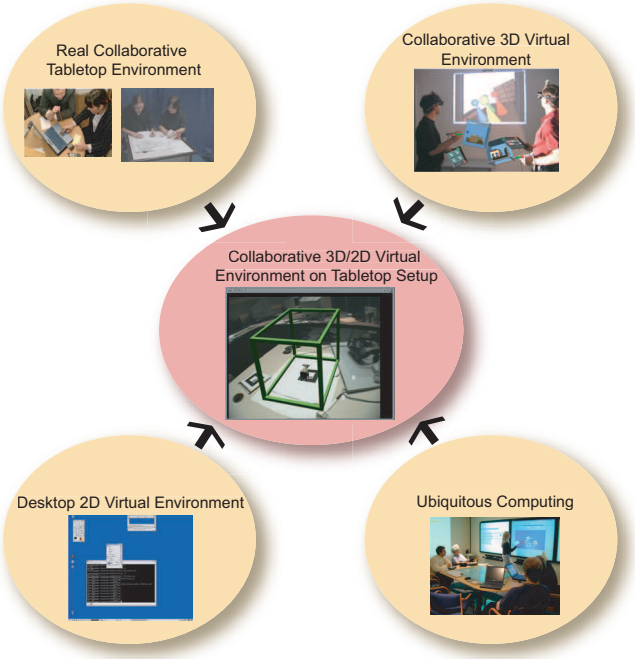


Fig. 7.2: Approche choisie : unifier les travaux dans un cadre de RA 3D sur table.

tion 4.1), sur une approche tangible (section 4.2) et enfin sur la généralisation de la notion de couplage (section 4.3).

4.1. Ontologie : multiples surfaces et multiples espaces

Description et extension de l'ontologie de Lachenal

Nous nous basons principalement sur l'ontologie définie par Lachenal [CLDC03],[LC03] pour de multiples surfaces. Nous présentons ici une extension en intégrant la notion d'espace (figure 7.3).

Son approche repose sur la notion de *ressources d'interactions*. Elles définissent des médiateurs entre des acteurs naturels (les utilisateurs) et des acteurs artificiels (le système). Une ressource d'interaction peut être de deux types : des *instruments* ou bien des *surfaces* (notées S dans la suite du document). Les instruments sont les canaux d'interactions sur une surface. Les surfaces sont la partie externe du contenu de l'information. Elles peuvent se classer en deux catégories : les S d'interaction/d'action, et les S de visualisation (voir les deux simultanément). Les acteurs naturels ou artificiels utilisent des *capteurs* pour recevoir des informations et des *actuateurs* pour modifier les ressources d'interactions.

Les utilisateurs peuvent alors interagir sur le contenu de S par l'intermédiaire des outils. L'utilisation de multiples S nécessitent de les mettre en relation pour avoir une vue globale d'un environnement. La *topologie* et la *géométrie* peuvent servir à définir les relations spatiales entres les différents éléments (acteurs, S ou instruments). Le nombre de relations peut être vite restreint à une certaine sous catégorie d'ensembles naturels : par exemple une surface perpendiculaire à une autre horizontale peut servir à représenter une vue 2D d'éléments 3D, la vue perpendiculaire une vue 3D.

Lachenal introduit en complément la notion de *couplage* et de *compatibilité* entre les surfaces. La notion de couplage est définie quand un élément A agit sur un autre élément B. La compatibilité définit le type de relations entre les S :

- *complémentarité* : coopération de tâche telle que définit par exemple dans la métaphore du peintre où les éléments sont complémentaires avec une relation outils-contenu ;
- *redondance* : dans ce cadre on utilise de multiple S pour le même contenu. Typiquement

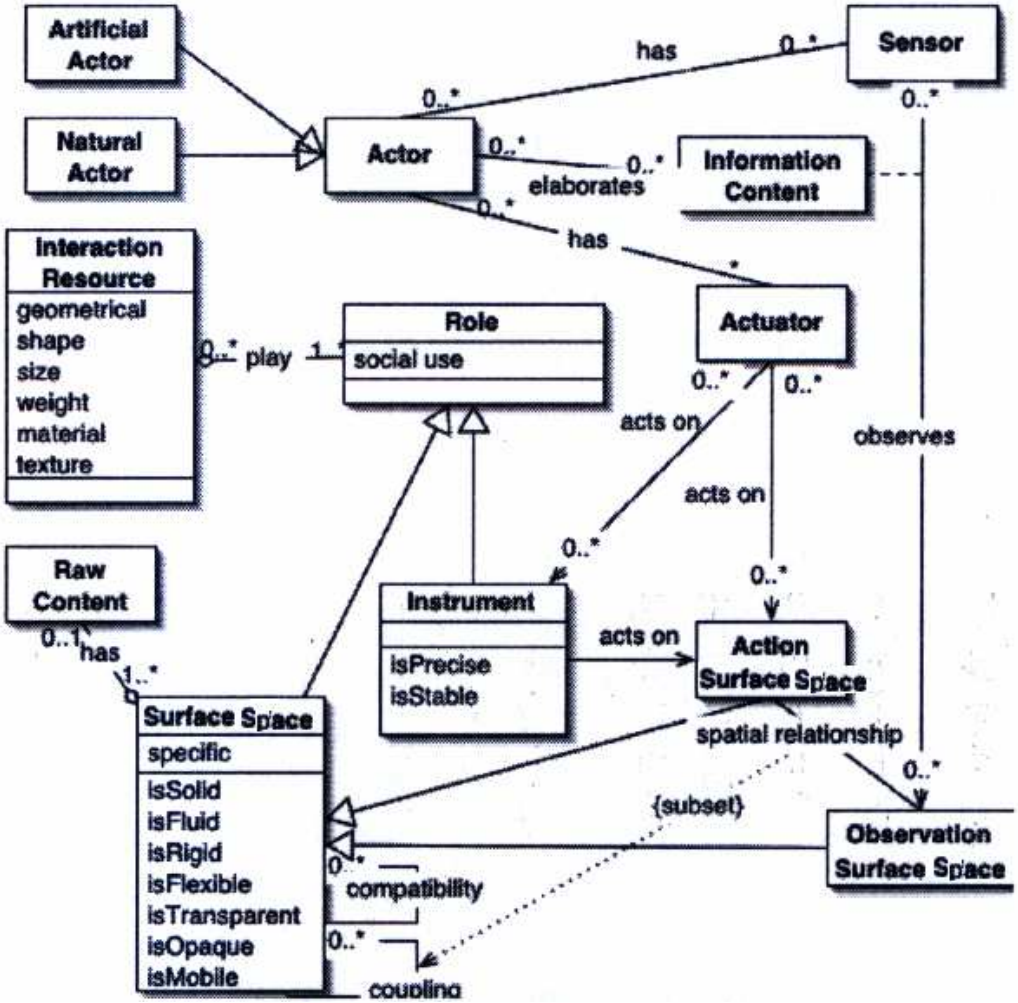


Fig. 7.3: Ontologie pour de multiples surfaces et espaces (extension de Lachenal [CLDC03]).

utilisé lors des présentations, l'orateur a une vue privée du document et les auditeurs une vue publique du même contenu ;
- *équivalence* : on utilise ici les éléments par alternance (temporelle), avec différents éléments pour la même tâche ;
- *assignation* : on affecte un rôle immuable à la surface.

De nouvelles notions

Nous nous basons sur ces concepts pour la mise en place d'un modèle adapté à un cadre de réunion collaborative colocalisé. Nous définissons alors que tout élément S peut être un espace ou une surface. Le contenu 3D et sa visualisation 3D est alors supporté. La notion de couplage - limité par Lachenal à la notion de surfaces physiques - est ici étendue à de multiples niveaux (détaillée plus loin dans ce chapitre).

Nous définissons notre environnement comme une composition de différents éléments hiérarchiques (figure 7.4) :
- *le référentiel (reference)* : Il définit un S contigu représenté par un ou plusieurs périphériques. Il correspond à une couche logique bas niveau (équivalent à l'espace d'affichage de *X Windows*) ;
- *l'espace de travail (workspace)* : inclus dans un référentiel, il définit un mode de gestion d'objets et des contrôles globaux des objets contenus (équivalent aux gestionnaires de fenêtre et environnement de bureau) ;
- *la fenêtre (window)* : elle définit un container applicatif, interface d'une application

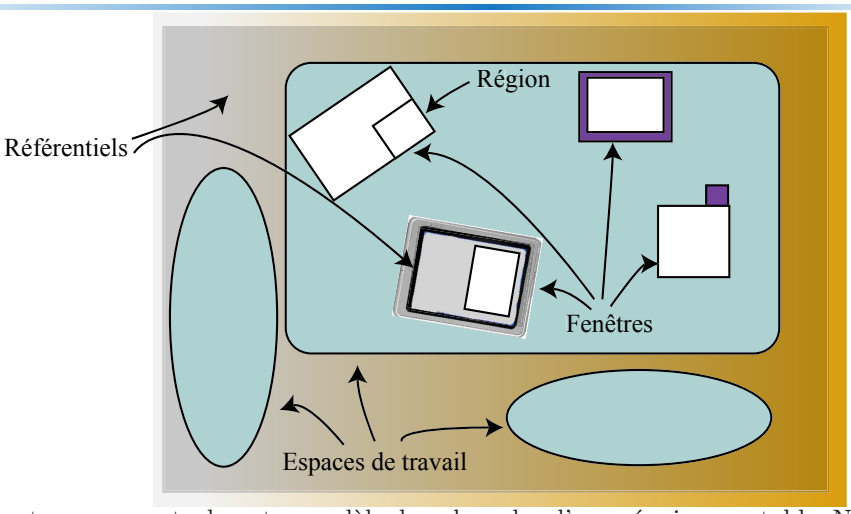


Fig. 7.4: Différents composants de notre modèle dans le cadre d'une réunion sur table. Nous distinguons les référentiels (en gris), les espaces de travails (en vert), les fenêtres (en blanc) et les régions (aussi en blanc). Les éléments violets désignent des manipulateurs physiques de ces fenêtres.

(équivalent aux fenêtres des bureaux numériques).
Chaque élément peut être défini relativement par rapport à sa catégorie supérieure. Cette relation peut être définie sur des critères spatiaux (e.g. coordonnées globales ou relatives), mais aussi sous forme symbolique (e.g. à gauche). Chacun des composants peut être dynamique ou statique. Chacun peut être partionné en sous-partie que nous nommons *Région*.

Référentiel

Un référentiel est un service procurant abstraction de la couche matérielle, du système d'exploitation ou du gestionnaire de fenêtre locale. Il fournit une couche bas niveau homogène chargé de la représentation de primitives graphiques bas niveau. Il peut être une surface (plane, courbe) ou un espace (cubique, sphérique, etc.). On peut facilement imaginer l'equivalent de la fenêtre root de X-Windows (afficher avec un quadrillage noir et blanc) sous la forme d'un espace 3D sur la table.

Un référentiel peut être constitué de plusieurs machines avec chacun une carte graphique ou d'une machine avec plusieurs cartes graphique, la zone d'affichage obtenu doit alors être équivalente. Le référentiel peut revêtir de multiples formes : assemblages d'un écran 2D, un carrelage de projecteur, un PDA, l'ensemble représentant alors une seule zone contigue.

Espace de travail

L'espace de travail définit un contexte spécifique à une tâche, un utilisateur ou une application. Il reprend les caractéristiques des espace publics et privés de réunions (e.g. taille, orientation, représentation spécifique du contenu, etc). Comme le propose *DiamondSpin* on peut associer naturellement des espaces à un utilisateur pouvant être configuré suivant ses préférences. Son rôle est alors la gestion de fenêtres, de panels d'outils ou de services d'adaptation dynamique du contenu.

Fenêtre

Une fenêtre peut être 2D (similaire à *Windows*©) ou 3D (similaire à *Studierstube*). La RA permet alors de définir des fenêtres dites mixtes contenant des objets réels et/ou virtuels. Nous reviendrons plus en détail sur ce concept dans la section suivante.

Region

Une région est similaire au *frame* défini dans *HTML* [W3Ca] et plus dernièrement *SMIL* [W3Cb]. Il correspond à la notion de segment dans *BEACH*. Elle définit une partition

spatiale, ses propriétés se limitant à sa taille et sa position. Par défaut l'absence de découpage de régions signifie l'existence d'une région unique. Une région n'a pas en général à communiquer avec la couche supérieure.

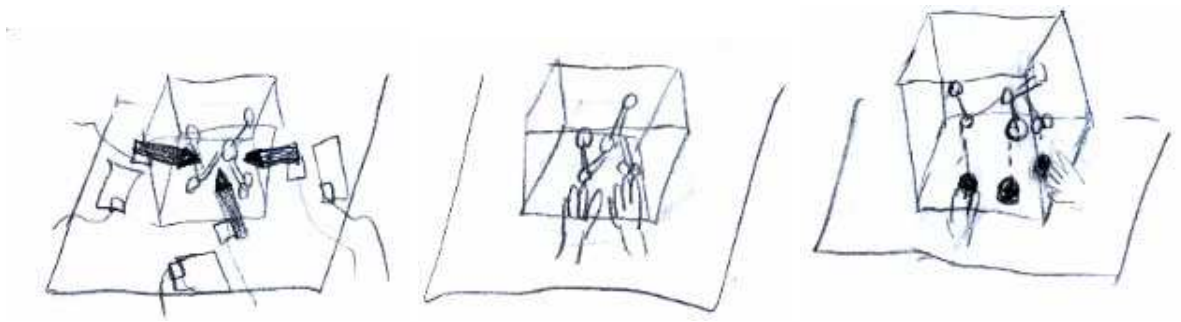
4.2. Description plus spécifique des fenêtres

Décomposition

Nous décomposons une fenêtre en trois types d'éléments :
- *le contenu* qui peut être réel/virtuel/mixte et 2D/3D ;
- *le conteneur* qui représente une enveloppe (visuelle) du contenu ;
- *le(s) manipulateur(s)* est(sont) un outil qui agit sur les propriétés d'un conteneur (décorateur dans la terminologie des environnements de bureau). Il peut alors être virtuel/réel, 2D/3D ou tangible/numérique. Les plus courants sont les manipulateurs de déplacement, de rotation, de taille et de visibilité.

La région d'une fenêtre représente une sous-partie du conteneur.

Un outil (dans la définition de Lachenal) peut alors être utilisé pour interagir sur le contenu et peut être représenté sous de multiples formes (R/V, 2D/3D). La figure 7.5 illustre des exemples d'interaction multi-utilisateurs sur une fenêtre.



(a) (b) (c)

Fig. 7.5: Interaction directe avec des outils sur la fenêtre : basées sur le concept du PIP, interaction gestuelle, interaction par TUI placée sur la table.

Conteneur

Il définit la gestion interne du contenu et représente l'interface standard accessible par une application. Il est associé à une zone allouée dans le référentiel. Le conteneur n'est pas limité à une représentation rectangulaire comme dans les environnements de bureau mais peut revêtir de multiples formes (figure 7.6).

L'introduction de la troisième dimension permet une extension de concept 2D, mais elle amène alors à ce poser des questions sur ces nouveaux paramètres :
- *le type d'enveloppe* : comme le souligne Schmalstieg [SFH00], l'enveloppe parallélépipédique supporte aujourd'hui la gestion du clipping comme sur les fenêtres 2D. Mais hormis ces limitations matérielles on peut alors utiliser n'importe quelle forme. Nous pouvons alors imaginer des solutions hybrides entre conteneur réel 3D et contenu virtuel ;
- *surface de l'enveloppe* : la surface de l'enveloppe n'est pas visible en 2D, le contenu de l'application étant positionné sur cette surface. Dans le cadre 3D, cette surface est visible, autant les faces devant le contenu que les faces arrières ;
- *la bordure de l'enveloppe* : la bordure est ici étendue à la troisième dimension. Nous pouvons alors nous demander si un aspect volumique peut apporter quelque chose ;
- *le positionnement* : il devient ici non contraint en rotation par rapport aux environnements classiques. La RA introduit de plus une dépendance vis-à-vis du réel : quelle

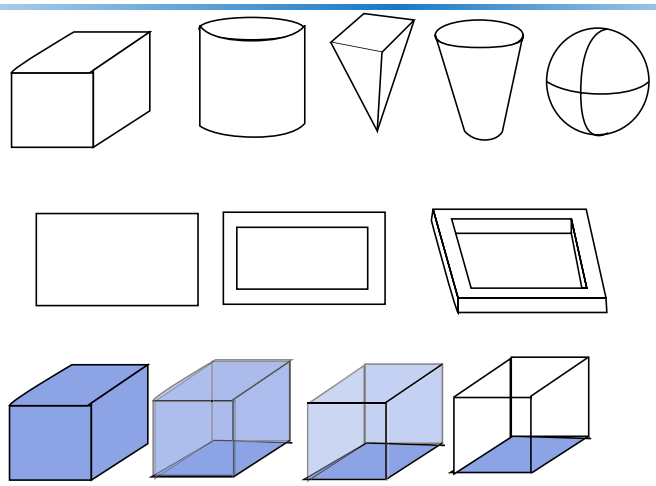


Fig. 7.6: Exemple de conteneurs (en bleu composant tangible).

relation spatiale ou physique avec l'environnement ?

Si on se place dans notre cadre, l'espace maximum est contraint par l'espace au dessus de la table. La manipulation libre de fenêtre dans cet espace semble difficile sans risque de chevauchement des éléments. Nous contraignons donc les fenêtres à avoir leur base (plan de dessous) sur la table, supposant qu'elle favoriserait l'interaction. La bordure des fenêtres induit lourdement sur la visibilité des éléments : nous choisissons dans ce but une representation différente pour les bordures des plans non en contact avec la table. Nous souhaitons pouvoir accéder directement aux fenêtres. Nous choisissons d'avoir une meilleur visibilité en utilisant des surfaces semi-transparente.

Document tangible

Nous introduisons ici un type spécialisé de fenêtre dédié à notre configuration sur table, nommé *fenêtre de RA tangible* (figure 7.7). Elles sont caractérisées par l'utilisation d'un manipulateur réel de déplacement. Contrairement au TUI, cette association est définie à un niveau plus haut que la simple association d'un élément virtuel à un élément réel. Elle représente l'association avec un contenu virtuel, supportant lui-même une interaction spécifique avec différentes métaphores de manipulation ou navigation.

Elle peut être vue comme un document papier lié avec du contenu numérique 2D, 3D. Ce support tangible doit être de préférence rigide pour favoriser les mouvements des objets et aide en complément à la gestion de la fenêtre. La table sert alors comme support pour placer ce document, le manipuler et ainsi fournir un retour durant l'interaction. Ce concept permet de reproduire les propriétés intéressantes des documents 2D (empiler grouper, comparer, échanger) tout en introduisant des nouvelles possibilités liées au virtuel (dynamicité, animation, modification interactive).

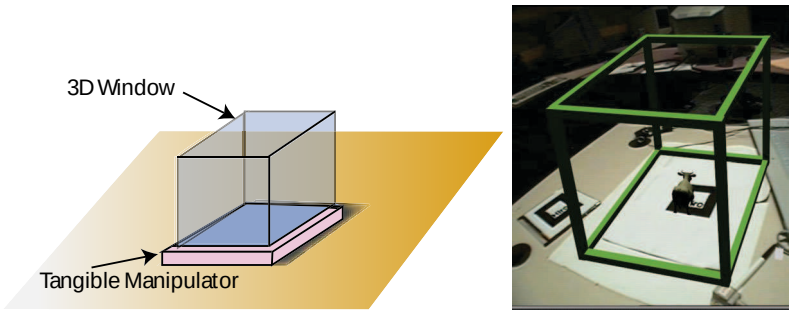
On privilégiera (pour des similitudes avec les réunions réelles) des formes simples (e.g. cube, plan, ligne, point), d'une taille proche du conteneur (figure 7.8). Diverses études sont nécessaires pour valider le choix approprié à un contexte.

Les variations sur le type de manipulateur et type de fênêtre d'une fenêtre RA tangible sont alors multiples : fenêtre 3D avec un manipulateur 2D, fenêtre 2D avec manipulateur 2D, fenêtre 3D avec du contenu réel 2D, etc.

Type de fenêtre

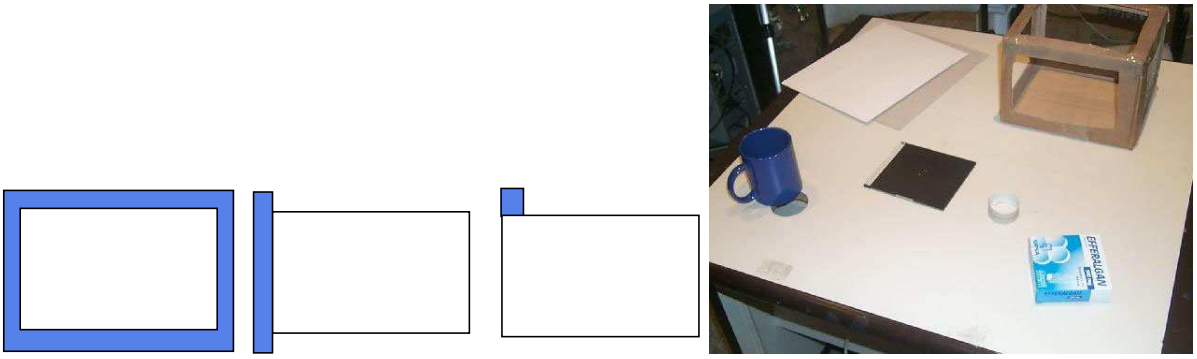
Dans le même esprit que Rekimoto [RUO01], nous identifions plusieurs types de fenêtres possibles :

– *contenu application* : définit la représentation visuelle du modèle d'une application ;



(a) (b)

Fig. 7.7: Fenêtre RA tangible : (a) concept (b) exemple de réalisation.



(a) (b)

Fig. 7.8: Exemple de manipulateurs : (a) manipulateurs 2D simples (b) exemple d'objets reels pouvant servir de manipulateurs.

– *outils* : définit un ensemble de fonctions, d'options et d'interfaces de manipulations lié à une application ;
– *pont* : elles sont des intermédiaires entre plusieurs mondes. Elles sont utilisées pour passer du réel au virtuel, de la 2D à la 3D et de l'accès distant à l'accès local ;
– *conteneur passif* : équivalent au *mediablock* elles servent à déplacer l'information entre sites distants.

Gestion de l'agencement

La métaphore de superposition des environnements de bureau numérique perd ici son sens : la superposition de fenêtre 3D n'est pas réalisable sans posés de lourdes difficultés d'utilisation et de gestion. Sur notre configuration sur table, nous choisissons alors respecter la gestion de documents 2D réels. Nous pouvons placer les éléments dans l'espace disponible de la table, autorisant l'intersection d'une fenêtre dans une autre dans le cas ou l'espace est rempli (on rappelle ici que nos fenêtres sont contraintes par leur base à être positionnées sur la table).

Nous proposons alors en complément la réutilisation des systèmes de fenêtrage classique, avec l'aide d'un élément de vue simplifié de l'environnement. Nous introduisons diverses possibilités d'arrangement de l'espace avec l'adjonction de cet élément (figure 7.9). La vue simplifie peut alors géré les éléments :
– sous forme linéaire, présent dans la plupart des GUI de bureau comme une barre. Elle peut être déplacée et orientée pour un utilisateur, ou un groupe d'utilisateurs. Elle peut être place sur une barre rectiligne ou courbe ;
– sous forme matricielle, présente sous forme d'un tableau. Elle permet d'avoir un meilleur point de vue sur un grand nombre d'applications et une gestion en classes des fenêtres ;
– sous forme circulaire, présente par des portions d'un disque. L'orientation peut alors être

dédie à la gestion particulière de fenêtres personnels d'un utilisateur. Nous proposons une interface rigide, et une interface sur un axe permettant une interaction à deux mains ;
- en 3D : grille 3D, cylindre 3D, sphère 3D ;
- sous une forme spécifiée par l'utilisateur.

La liberté de positionné le conteneur des fenêtre réduites devient ici orientable et tangible, permettant un positionnement dans une zone libre, un partage multi-utilisateurs.

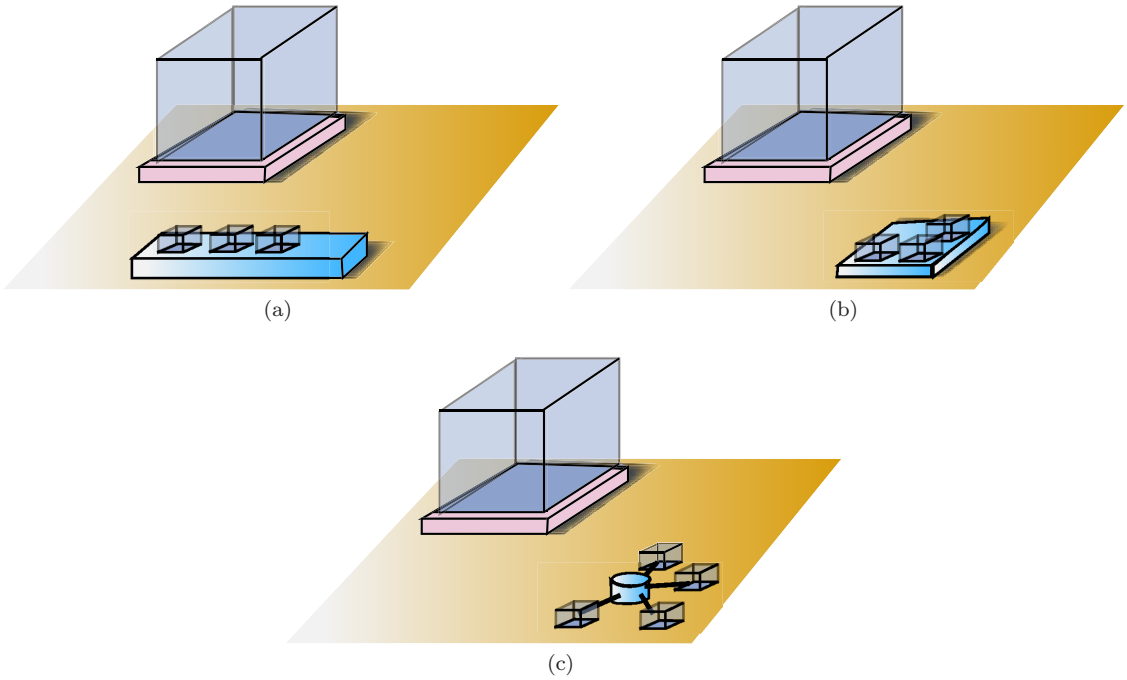


Fig. 7.9: Gestion de l'agencement des fenêtres : (a) gestion en ligne (b) gestion en grille (c) gestion circulaire.

Mais une problématique peut alors être soulevée dans le cas ou la fenêtre 3D est une fenêtre RA tangible. Comment réduire un document 3D virtuel associé à un support réel ? Une première solution pourrait être de considérer qu'un élément réel est contraint a un seul élément virtuel et la réduction de la fenêtre laisse l'élément en place. Cette approche nécessite alors de se souvenir de la relation ou de connaître explicitement la relation. De plus, l'espace physique est alors surchargé d'éléments réels non effectifs.

Une autre approche consiste à considérer qu'un élément réel est partageable entre des élément virtuels. Par exemple, si un élément virtuel A associé à un élément réel est réduit, et qu'un élément virtuel B, précédemment réduit, est maximisé il pourra alors être associé au même élément réel. Mais que faire alors si on maximise le document A ? une solution est alors de considère qu'un élément prend la place du prochain élément réel libre. Cette solution offre un gain de place, mais semble difficile d'usage avec de forte variation de gestion pouvant être peu naturel.

Enfin, une autre solution pourrait être de ne pas utiliser de zone de gestion d'arrangement virtuel mais utiliser une approche réelle. Plus précisément, nous pouvons considérer que la version réduite d'une fenêtre 3D est la représentation réelle de l'objet physique associé. La relation virtuel-élément réel est donc fixe, fortement corrélée et permet l'usage de techniques d'agencement réel. Les utilisateurs peuvent alors par exemple superposer en pile les manipulateurs pour agencer les différents contenus.

4.3. Couplage : approche multiniveaux et générique

Nous distinguons les types de couplage suivant :
- *couplage d'espaces physiques* provoquant le partage d'un référentiel ou d'un espace de travail. Hinckley [Hin03] et Lachenal [LC03] s'intéresse à fournir un nouveau type de métaphores pour réaliser le couplage dynamique par connections des éléments créant un espace de travail étendu ;
- *couplage de référentiels*, permettant d'étendre deux zones de gestion différente. Le prototype EMMIE illustre très bien ce cas, avec le couplage d'un espace 3D partagé avec différentes surfaces. Ce cas est aussi assimilable à la reprogrammation de sous-espace d'un mur d'écrans ;
- *couplage d'espace de travail*. Rekimoto [RAH98], ainsi que Shen [SKER03] et Tandler [TPMT+01] ont introduit la notion de multiples espaces de travail, des documents pouvant transiter entre ces espaces ;
- *couplage de fenêtres*. Un exemple est le prototype *DataTiles* permettant le couplage d'applications fenêtres 2D en RA ;
- *couplage d'éléments virtuels*. Poupyrev propose avec *Tiles*[PTB+02] le couplage d'éléments virtuels basés sur la mise en contact de deux éléments (en fait déclenchement de l'action basé sur un seuil de distance entre les deux éléments).

On revient plus précisément sur le cas le plus intéressant de notre environnement : le couplage de fenêtres.

couplage de fenêtres

Un couplage de fenêtres met en relation deux contextes applicatifs, analogie à l'utilisation de multiples documents papiers lors d'une réunion. Cette relation peut alors s'exprimer d'un point de vue logiciel comme la mise en relation de fonctions, propriétés ou messages entre deux objets.

Nous considèrons alors, en généralisant l'approche de Lachenal, que cette mise en relation constitue un nouveau type d'interaction pouvant alors être géré par l'application, mais aussi par l'espace de travail ou le référentiel (figure 7.10). La gestion de la relation pouvant alors générer des événements pour une application. Nous considérons de ce fait que la mise en relation peut être explicite ou implicite.

Dans un cadre implicite, la relation est définie automatiquement à base de règles. Nous avons identifié trois classes de relations :
- *relation géométrique* : se décompose en relation globale (définie dans un repère commun) ou locale. Les différents paramètres identifiés sont : la distance, la position relative, l'alignement, etc ;
- *relation topologique* : contact, inclusion, intersection, recouvrement, séparation ;
- *relation symbolique* : iconographique, caractéristiques morphologiques, etc.

De facon explicite, l'utilisateur spécifie interactivement la relation. Dans ce cadre, comme le définit Rekimoto et l'introduit plus récemment *ARWin*, nous pouvons utiliser un principe de slot qui est une représentation visuelle d'un point d'entrée d'un composant logiciel. Dans ce cadre, la tâche à un niveau utilisateur se ramène à fournir un mécanisme de sélection des éléments, de sélection des paramètres des documents et fournir un mécanisme de liaison des éléments. Chacune des étapes est complétée par un retour visuel pour l'utilisateur. Nous identifions plusieurs approches :
- connection directe par contact physique entre deux slots physiques ;
- connection directe par contact virtuel entre deux slots virtuels ;
- connection distante sous forme d'une liaison visuelle entre les éléments. Nous avons repris ici le concept défini par DiVerdi - et largement utilisé dans des interfaces graphiques d'applications de flots de données - en établissant la liaison par sélection sur une représentation visuelle des deux éléments en tirant une ligne entre les deux ports des éléments.

La notion de couplage permet de définir une association entre du contenu et un outil. La

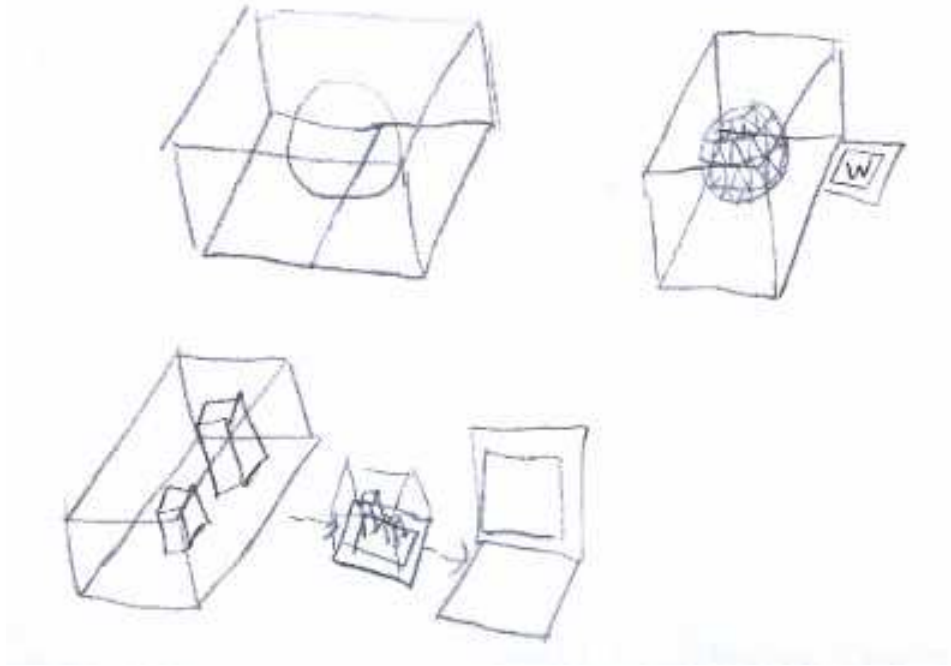


Fig. 7.10: Différents types de couplage. La première image présente deux fenêtres associées pour créer un espace plus grand. La seconde figure présente l'association d'un outil avec un espace 3D, changeant la représentation du contenu. La dernière figure représente le sauvegarde d'une application par l'utilisation d'un élément tangible de transfert.

contrainte de l'outil associé statiquement à une application dans les environnements de bureau numérique est ici levée en se basant sur les propriétés spatiales, pouvant séparer physiquement contenu et outils. Cette interaction indirecte peut être alors sous de multiples formes (figure 7.11). Nous avons identifié quatre types de relations dans le cadre d'un document tangible : relation implicite où l'outil est positionné dans la fenêtre (pas de couplage dans ce cadre), relation implicite où l'outil est positionné à l'extérieur de la fenêtre (pas de couplage dans ce cadre), relation implicite libre et relation explicite définie par l'utilisateur.

Cette approche va au delà du concept introduit dans Studierstube, en découplant l'outil du contenu, mais dans notre cas elle permet aussi d'avoir des outils décomposés physiquement (pas seulement un seul élément) et de créer dynamiquement le couplage durant la session. Différents utilisateurs avec un assistant digital personnel, des sliders 3D ou une interface physique peuvent interagir simultanément sur le même contenu.

5. Mise en œuvre

Nous nous basons sur notre architecture logicielle avec un suivi par *ARToolkit* pour les éléments. La description du contenu est réalisé sous *OpenGL Performer* et *Qt*.

Nous utilisons ici une architecture par briques logiciels permettant une identification, une accessibilité et une connection efficace et rapide des différents éléments (figure 7.12). Une fenêtre, un espace de travail, un référentiel ou un élément physique est alors défini sous forme d'objets (composant logiciel). Un module est en charge de la création, destruction et communication des objets. Il est aussi chargé de gestion des règles.

Nous décomposons les règles en deux composantes de façon similaire à SMIL [W3Cb] : une condition, qui peut être un ensemble de sous conditions et une liste d'actions exécutées lorsque la condition est vraie.

Le couplage est défini sous forme d'un service logiciel constitué de trois sous services. Le

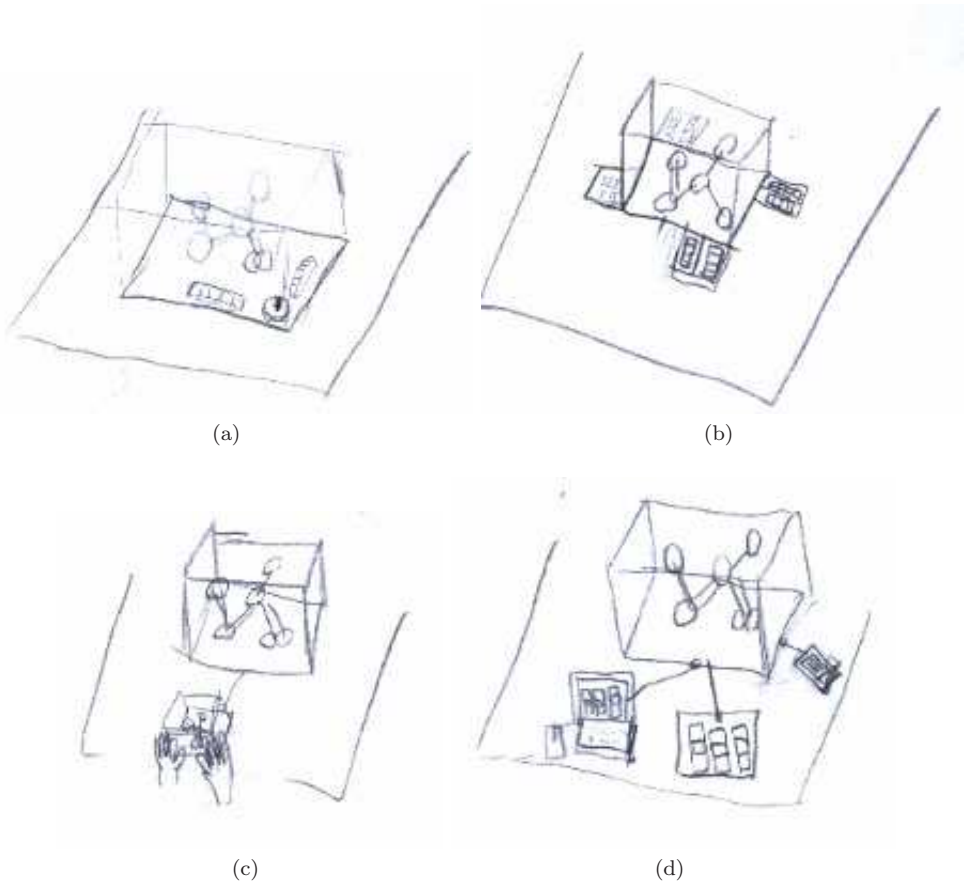


Fig. 7.11: Relation outils et fenêtre RA tangibles. Un widget peut être (a) statiquement positionné dans la fenêtre, (b) sur son bord extérieur (c) dans une autre fenêtre couplée physiquement ou (d) dans une autre fenêtre couplée à distance.

premier est le *service de couplage* qui vérifie à partir des données mesurées (depuils les capteurs) les conditions de chaque règle. Si les conditions sont vraies, on utilise alors *le service de connection*. Si l'action a une continuité temporelle, il établit la communication entre deux éléments. Sinon, il exécute alors les actions associées à la condition. Le *service de communication* est alors chargé d'assurer la mise à jour des relations établies.

Chacun de ces éléments peut être configuré à base d'un fichier *XML*. De même nous avons choisis une approche descriptive pour la définition de règles dérivée aussi de la syntaxe de *SMIL* (figure 7.13).

6. Résultats et applications

6.1. Premier prototype

Un premier prototype a été réalisé à partir de ces différents concepts. L'implantation est aujourd'hui partielle, la gestion dynamique de couplage de référentiel ou d'espace de travail et le couplage explicite de fenêtres ne sont pas encore réalisés. Une suite d'exemple illustre les possibilités de ce premier prototype.

Le point suivant est le support interactif avec le contenu d'une fenêtre. Nous avons d'une part réalisé la gestion multi-pointeurs dans une même fenêtre et d'autre part réalisé la gestion de l'interaction indirecte des boîtes à outils (figure 7.14). Pour la gestion et la cohérence de multiples interfaces d'un point de vue logiciel, nous nous basons sur le pattern de programmation *Observer* [GHJV95].

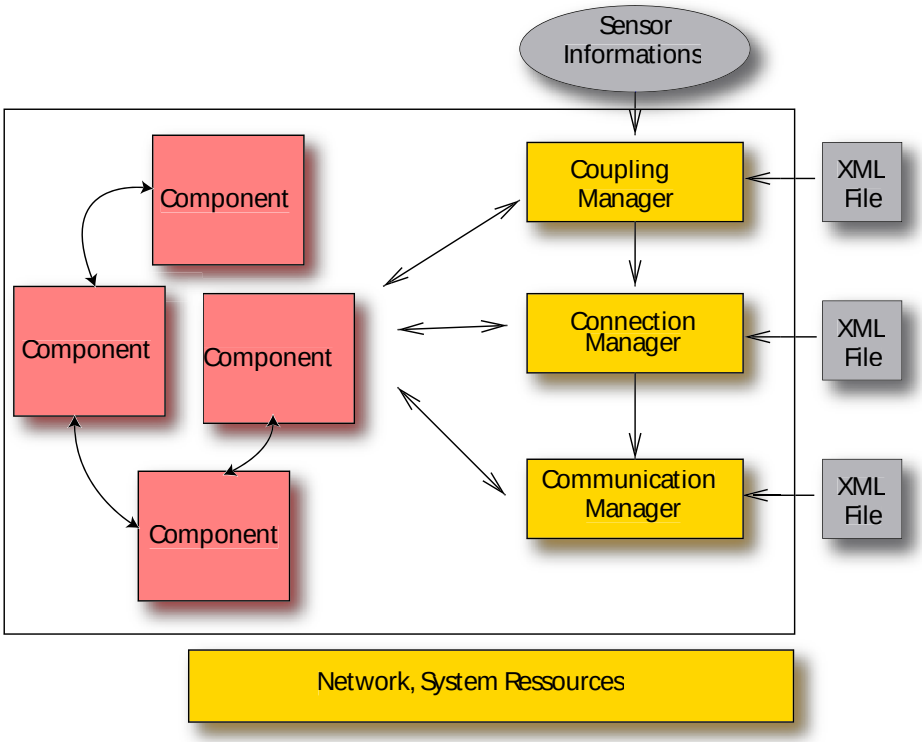
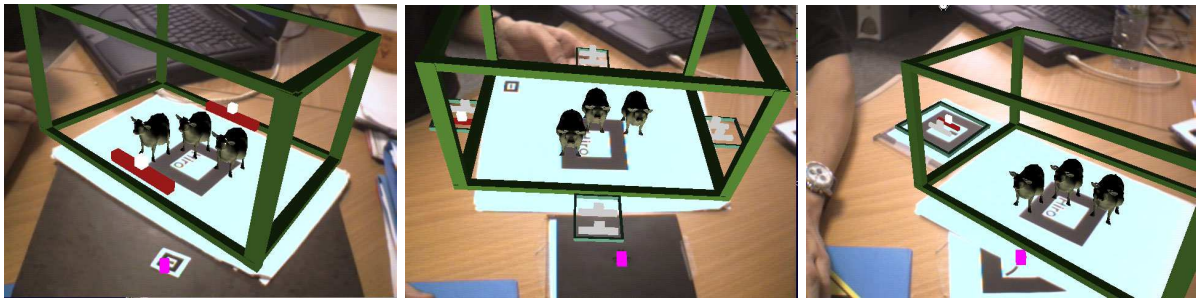


Fig. 7.12: Architecture pour le couplage.

```
<?xml version="1.0" encoding="iso-8859-1">
<rules>
  <rule id="CopyShared">
    <conditions>
      <condition>
        <event id="copyevent" type="binary" checking="onetime">
          <element id="globalSpace" type="workspace" slot="undefined">
          <element id="copymarker" type="physical" slot="undefined">
        </event>
      </condition>
    </conditions>
    <actions>
      <action type="Copy">
          <element id="globalSpace" method="Copy">
      </action>
    </actions>
  </rule>
</rules>
```

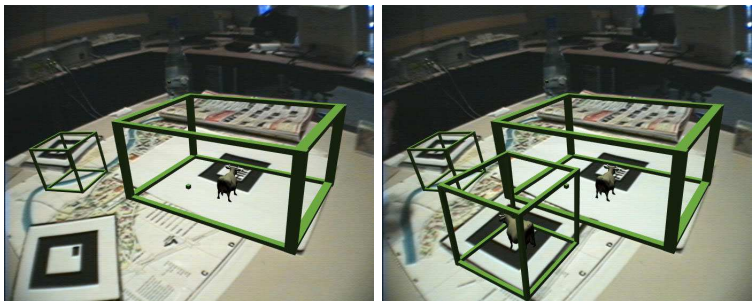
Fig. 7.13: Exemple d'un fichier de règle de couplage.

La figure 7.15 présente un couplage entre un document RA tangible et une surface 2D. La règle est ici la copie d'un élément 3D dans une nouvelle fenêtre associée à cet élément. On peut utiliser cette métaphore pour transiter de l'information entre différents espaces, telle que récupérer une sauvegarde dans un conteneur personnel. Les modes dérivés supportés sont clone et référence, où le contenu est alors partagé, mais pouvant être à différentes positions spatiales (le mode clone maintient juste une cohérence des propriétés spatiales).



(a) (b) (c)

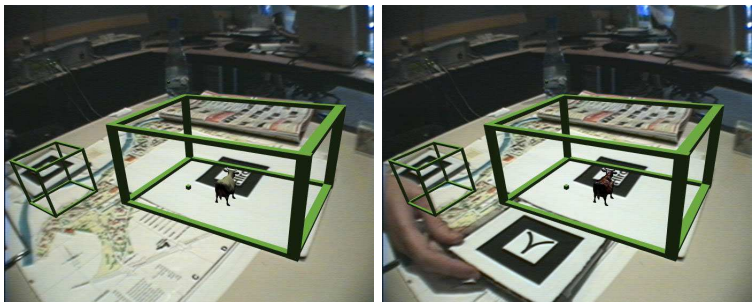
Fig. 7.14: Interaction indirecte avec un fenêtre : (a) widget dans la fenêtre (b) widget sur le bord (c) dans une autre fenêtre couplée physiquement (d) dans une fenêtre couplée à distance.



(a) (b)

Fig. 7.15: Copie d'un tangible AR document. Un utilisateur peut copier le contenu d'une fenêtre 3D par simple contact physique entre les éléments.

Le couplage peut aussi s'utiliser comme métaphore de mode de représentation de données, cet outil pouvant être alors utilisé de façon générique par de multiples contenus (figure 7.16). Il nous semble que ce type d'outil pourrait alors être défini à un niveau applicatif plus haut que la fenêtre.



(a) (b)

Fig. 7.16: changement de représentation par couplage de l'outil. Le modèle virtuel de vache passe ici d'un mode face caché basique à un mode de rendu haute qualité. L'effet n'est par persistant, la déconnexion physique supprimant la relation.

Une complémentarité entre surface haute définition et espace 3D peut être apportée pour le support d'informations textuelles (usage de deux référentiels). L'exemple suivant (figure 7.17) montre un couplage entre un outil et un objet 3D générant l'affichage automatique d'une page web sur la surface 2D.

Cette complémentarité peut être aussi définie en entrée, le référentiel 2D contrôlant le contenu de l'espace 3D. On peut imaginer le contrôle d'informations 3D à partir d'un

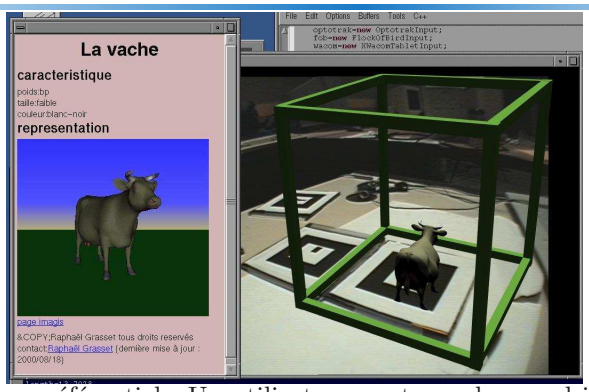


Fig. 7.17: Usage de plusieurs référentiels. Un utilisateur peut coupler un objet tangible à un espace 3D pour obtenir des informations affichées dans le référentiel de l'écran de l'ordinateur portable.

assistant digital personnel (figure 7.18). La réalisation de cet exemple utilise principalement un émulateur d'assistant PALM fonctionnant sur le portable et synchronisé avec l'application tournant sur l'Onyx.

Fig. 7.18: Interaction à partir d'un autre référentiel : interaction par PDA pour le contrôle de l'orientation du contenu d'un tangible AR document à l'aide de sliders.

6.2. Re-interpretation de prototypes de recherche

Notre canevas s'appuie sur tous les travaux présentés dans l'état de l'art de cette section. Il peut alors servir à reformaliser a-postériori les environnements existants, ceux-ci devenant des instances de notre modèle. Nous présentons un exemple détaillé ici sur le prototype *EMMIE*.

EMMIE peut être défini comme quatre référentiels ayant chacun leur propre espace de travail (touss défini globalement dans un même repère). Un référentiel 3D est défini dans l'espace de la pièce et repose sur l'utilisation de casques. Les référentiels 2D écran suivi et portable sont inclus dans ce référentiel. Il n'introduit pas de notions de fenêtre, les objets étant donc dans l'espace de travail. Il définit plusieurs règles de couplages implicites, géométriques, définis dans un repère absolu. Ces règles sont les suivantes :

- déplacement d'objets entre référentiel 3D, écran ou projection. Pour le transfert 2D portable vers espace, un région spécifique sur le portable permet ce transfert ;
- visualisation couplée entre espace 3D et portable ;
- écran suivi définissant une vue du référentiel 3D.

6.3. Applications

Nous présentons trois cadres applicatifs de notre nouveau concept : réunion sur table, pré-

sentation, activité en petit groupe. Le domaine applicatif peut être très variable (exemple pour l'architecture figure 7.19).

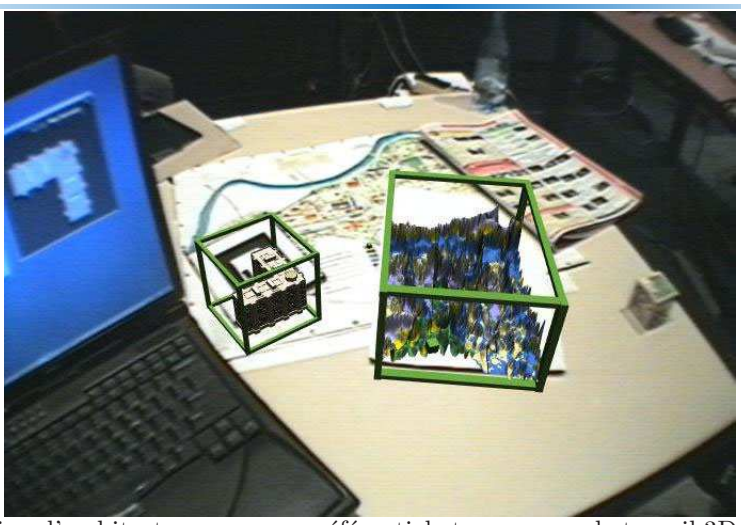


Fig. 7.19: Application d'architecture avec : un référentiel et un espace de travail 3D (espace de la table), un référentiel et un espace de travail 2D (portable), deux fenêtres 3D virtuelles, une fenêtre 2D mixte (avec carte réelle) et une fenêtre 2D virtuelle (sur le portable).

Réunion (figure 7.20) : notre système permet un enrichissement haut niveau des réunions. L'utilisateur retrouve la manipulation simple de documents réels, tout en unifiant contenu numérique et virtuel. Les métaphores de couplage permettraient de gérer ces multiples formats, d'échanger des informations, de sauvegarder et de gérer leur propre zone personnelle.

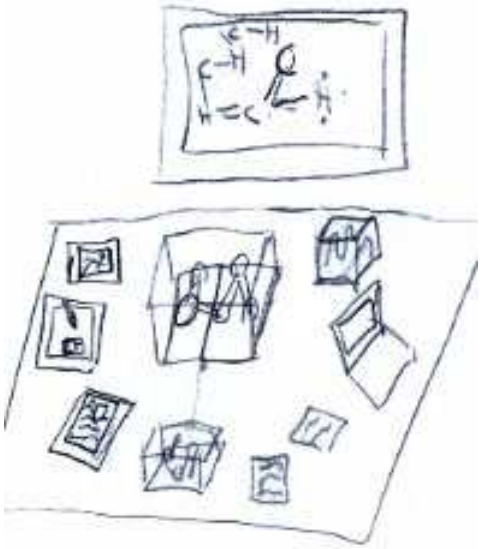


Fig. 7.20: Réunion. Les utilisateurs peuvent utiliser de multiples espaces 3D, assistants personnels, projection murale, interaction sur tablette digitale. Tout est unifié dans le même canevas logiciel facilitant la collaboration et les échanges durant la réunion.

Présentation sur multiples surfaces et espaces (figure 7.21) : nous pouvons imaginer l'extension d'une présentation numérique sur écrans à de multiples référentiels, espaces de travail, fenêtres. Cette solution permettrait de diffuser l'information appropriée sur un support dédié et d'utiliser les avantages d'une représentation spatiale 3D (complémentarité avec un espace large, hiérarchisation visuelle, lien hyperspace, arrangement dynamique sur des supports mobiles).

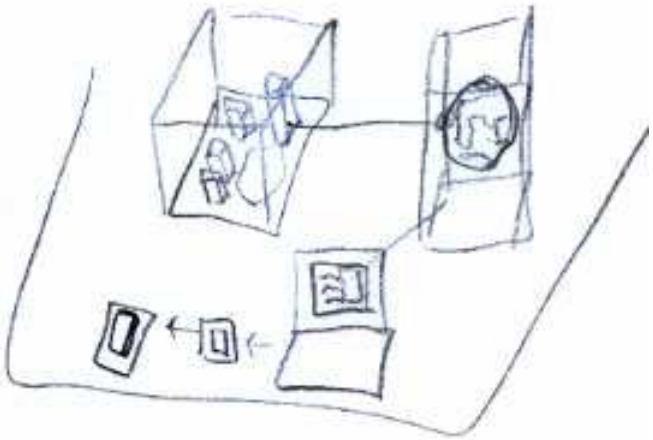


Fig. 7.21: Présentation Multi surfaces et multi espaces. De multiples conteneurs permettent un affichage de multiples informations et leurs transitions souples.

Activité en sous groupes (figure 7.22) : en complément des réunions en petit groupe, ce type d'interface permet une transition et gestion vers une activité en sous groupe à partir des régles de copie décrites précédemment. Dans le cadre d'un cours, l'activité en sous groupe se traduirait par la résolution d'un exercice. Notre solution permettrait alors une interaction en groupe sur plusieurs fenêtres qui pourraient alors être couplées pour comparer les résultats ou tester d'autres solutions.

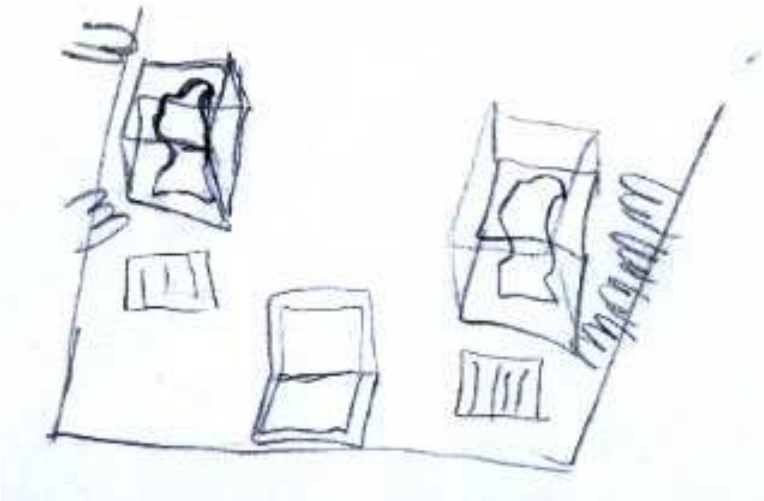


Fig. 7.22: Activité en sous groupes. Différents espaces peuvent être copiés et différents utilisateurs travaillent alors séparément sur le même contenu pouvant a posteriori être unifié.

On peut aussi imaginer l'application de ce canevas dans les musées. En effet, la mise en place d'événements nécessite généralement des configurations très différentes et complexes à mettre en œuvre. Notre environnement supportant des configurations hybrides permettrait alors la mise en œuvre de véritable scénario.

7. Limitations et perspectives

Le prototype présenté démontre la faisabilité du concept et le modèle supporte de nombreux cas. Son usage reste très limité d'un point de vue du coupage dynamique. L'utilisation de la librairie *ARToolkit* n pour le suivi des fenêtres RA tangibles n'est pas suffisamment robuste aux occultations, problématique dans un cadre de manipulation d'éléments

rigides de type cartons. La mise en œuvre de nouveaux algorithmes de vision dédiés permettrait de gérer ces problèmes.

Les critères d'utilisabilité n'ont pas été à ce jour évalués. Dans ce cadre une étude plus poussée doit être menée pour faciliter l'accessibilité de ce type d'interface.

8. Conclusion

Ce chapitre a présenté plusieurs techniques d'interactions dans le cadre d'interaction en RA 3D sur table. Une émergence des concepts de *Single Space Groupware (SSG)* et de *Multiple Space and Surface Groupware (MSSG)* nécessite comme pour les SDG des techniques appropriées que nous avons tenté de définir dans ce chapitre. L'ensemble des concepts peuvent être vus comme une extension 3D des travaux d'IHM ou comme une spécialisation à l'espace de la table des travaux de RA.

Sans aucun doute l'ensemble définit une proposition qu'il est nécessaire d'évaluer plus en détails au cours d'évaluations formelles et comparatives. La modélisation d'interface hybride nécessite une étude plus approfondie comme sur son usabilité. Comme on a pu le voir le réel nous offre de nombreuses possibilités mais induit en contrepartie de nouvelles problématiques à résoudre. Le concept d'interaction peut alors être poussé à des limites, telle que l'interaction sur le réel. Ce sujet constitue l'étude du prochain chapitre.

Interaction avec le réel

la réalité n'est qu'un point de vue.

—Philippe K Dick.

Le chapitre précédent s'est intéressé à l'interaction avec des éléments virtuels placés dans l'environnement réel, ainsi que la proposition d'un certain nombre de nouvelles techniques d'interaction. Nous avons essayé de démontrer l'intérêt d'utiliser des objets réels pour interagir avec les éléments virtuels. Nous souhaitons pousser ce concept plus loin ici en introduisant un nouveau type de relation en réalité augmentée : *interagir SUR le monde réel*.

Ce chapitre décrit un cadre conceptuel pour ce nouveau type de liaison (section 3) et la description d'un prototypes pour pouvoir modifier l'apparence d'objets réels (section 4).

Le contexte d'interaction collaboratif sur table nous amènera à nous restreindre à des situations de ce cadre, tout en gardant la possibilité d'un formalisme plus générique. Ceci nous conduit à soulever différentes questions que l'on va tenter de résoudre dans ce chapitre (figure 8.1) :
– Est ce que l'utilisateur peut modifier le réel dans le cadre d'une application sur table ?
– Est ce que l'utilisateur peut agir sur l'apparence d'un objet réel ? interactivement ?
– Comment formaliser la relation entre réel et virtuel ?

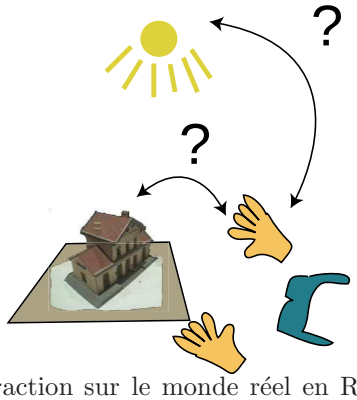
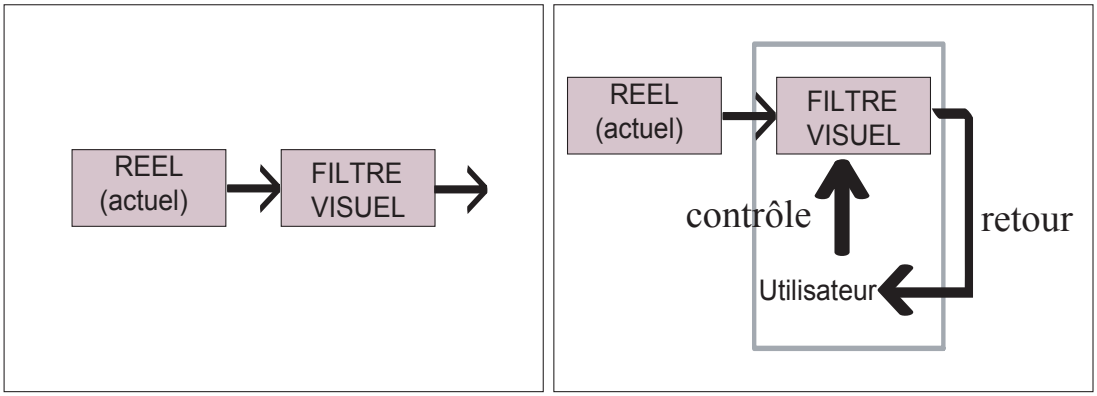


Fig. 8.1: Problématique lié à l'interaction sur le monde réel en RA sur table : quelles relations entre contenu réel et utilisateur ? quelles relations avec l'éclairage réel ambiant ?

1. Introduction et concept

Un grand nombre de travaux de réalité augmentée se sont orientés vers l'incrustation (basé image ou scène) d'objets virtuels dans le monde réel, tel que les travaux de Rekimoto [RN95], Feiner [FMS93], Bajura [BFO92] ou Poupyrev [PTB+02]. A ce jour, peu de travaux se sont penchés sur la problématique de modification directe du réel.

Comme brièvement présenté chapitre 2, la *Mediated Reality*, introduite par Steve Mann, se ramène à cette problématique (voir l'article de référence [Man94], ou [MF01]). Elle consiste à exprimer la modification du monde réel comme l'application d'un filtre linéaire ou non-linéaire basé sur des solutions d'architectures matérielles ou logicielles (figure 8.2(a)). Dans ce contexte la réalité augmentée peut être décrit comme un simple filtre additif. Jusqu'à



(a) (b)

Fig. 8.2: Concept de la *Mediated Reality* (a) et l'*Interactive Mediated Reality* (b) : l'utilisateur est fortement couplé avec la vision mixte réel/virtuel et peut contrôler le type d'éléments qu'il visualise.

ce jour les architectures proposaient des placements hors-ligne d'éléments virtuels, ou utilisaient comme Steve Mann des filtres matériels pour changer la vision du monde réel.

Nous généralisons l'approche et introduisons l' *Interactive Mediated Reality (IMR)* (figure 8.2(b)) : à l'aide d'outils interactifs, on change en *temps réel* la vision du réel, grâce à l'image de synthèse. Cette approche novatrice bénéficie de la complémentarité entre les images de synthèse et le monde réel.

En effet, les images de synthèse fournissent une représentation rapide et évolutive d'un objet[1] et peuvent produire des effets irréalistes (la solution n'est pas contrainte par les lois physiques comme le monde réel). Quand à l'utilisation du monde réel, il permet une manipulation intuitive d'éléments, un retour tactile et haptique, un environnement naturel d'activités collaboratives et des possibilités de perception non limitées par leur reproduction artificielle par l'informatique. Les derniers travaux de Lok [LJNW03], qui étudie la manipulation d'objets virtuels en RV, montre l'intérêt - dans un contexte d'utilisabilité et d'efficacité - d'utiliser des élément réels et de conserver leur visualisation propre. La figure 8.3 montre les résultats que l'on pourrait obtenir avec ce concept.

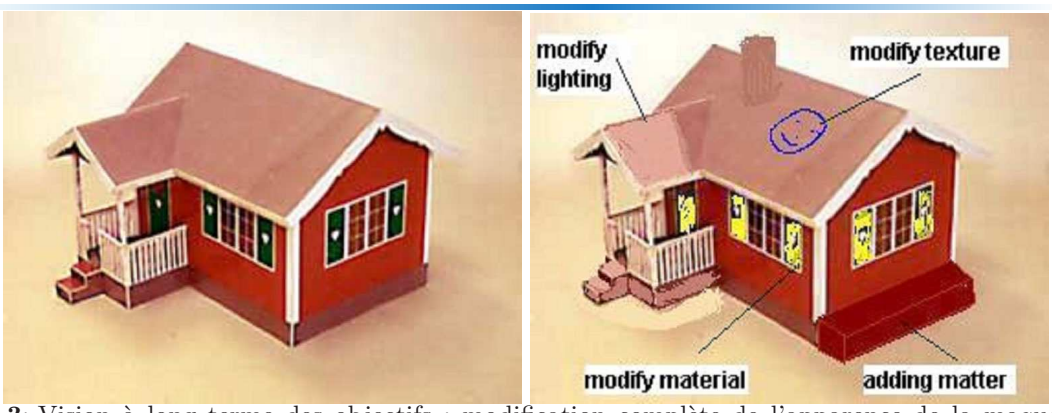


Fig. 8.3: Vision à long terme des objectifs : modification complète de l'apparence de la maquette de maison (ici photomontage du résultat escompté, avant (gauche) et après (droite)).

Comme énoncé par Raskar [BRF01] le concept de modification de l'apparence d'objets réels peut trouver son application (la figure 8.4 montre un exemple) dans le domaine de l'architecture (changer l'apparence d'un bâtiment réel), de l'industrie (changer la couleur/matériau d'une maquette, d'un prototype), du design (ajouter des propriétés pho-

[1] un faible coût en énergie, temps et de ressources comparé à la création et à la modification d'un objet réel

tométriques pour du packaging), ou dans le domaine culturel (créer des scènes mixtes à partir de maquette réels, définir de nouveau type de jeu, etc.).

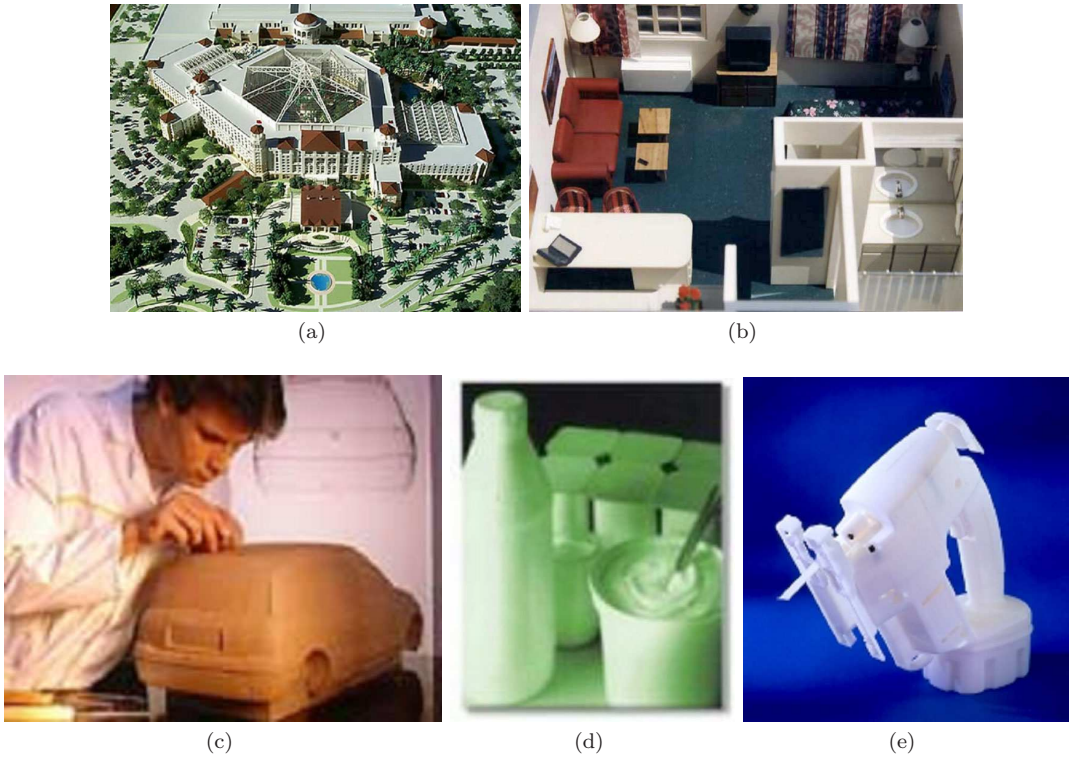


Fig. 8.4: Applications identifiées : (a) et (b) architecture (c) automobile (d) packaging (d) et (e) prototypage industriel.

2. Etat de l'art

Dans un contexte très général, «agir sur la réalité» peut désigner plusieurs type d'actions :
- utiliser des objets réel pour agir sur du contenu virtuel, ce sont les *tangible user interface*, déjà évoquées au chapitre 2 ;
- augmenter les propriétés d'un objet réel : c'est le concept d'*amplified reality* aussi présenté dans le chapitre 2 ;
- changer la vision de la réalité : cela peut être réalisé hors-ligne, par modification d'images vidéos avec des effets basés sur l'infographie (effets spéciaux ou retouche d'images), ou en ligne, en temps réel, par le concept de *mediated reality*.

On se limite ici à l'étude de la dernière sous catégorie, notre domaine étant dédié à la visualisation.

Le travail le plus intéressant pour modifier l'apparence d'objet réel est sous aucun doute le système introduit par Raskar, avec les *Dynamic Shader Lamps* [BRF01]. Basé sur une approche par projection, il a réalisé un prototype pour peindre sur des objets réels et aussi fournir l'illusion d'animation d'objets réels statiques. Malheureusement son système se limite à des surfaces ternes et unicolores d'objets réels (principalement dû au limitation de la plage de couleur dynamique du projecteur), souffre des difficultés d'utilisation dans un contexte réel (peu d'éclairage), des occultations du rayon du projecteur par l'utilisateur et est limité en terme des propriétés de mixage entre réel et virtuel.

Lee [LHS01] introduit un système pour digitaliser des objets réels à l'aide d'un système de réalité augmentée. Le système permet alors de déformer la forme résultante, mais le système est trop fortement manuel (construction point par point) et les exemples se

bornent à la création d'objet avec de la symétrie. Récemment Fiorentino [FdAMS02] a décrit un système pour esquisser et modéliser des objets virtuels en RA. Le système, basé sur un casque suivi, utilise principalement des outils classiques de CAO/CAD (extrusion, *skinning*, surface de Coons) pour modéliser de façon intuitive les éléments 3D. Bien que l'utilisateur peut ajouter des éléments virtuels 3D à un objet réel, le système - ciblé pour ingénieur – n'autorise pas un utilisateur à agir sur un objet. Les options de l'utilisateur sont limitées à l'alignement virtuel d'éléments réels et virtuels et à la superposition d'une nouvelle représentation virtuelle d'un objet réel (de façon globale).

Différents travaux nous ont aussi semblé intéressant par leur approches qui fournissent des métaphores intuitives pour modifier l'apparence d'objets virtuels. Le système *ArtNova* [FOL02] présente une approche interactive avec une interface par retour haptique pour peindre couleur et texture sur des objets, dans la continuité de travaux de même lignée Agrawala [ABL95], Hanrahan [HH90] (et repris dans des logiciels comme *Maya*© [may], *3DS Max*© [3ds] ou *Deep Paint 3D*© [3D.]). *Dab* [BSLM01] fourni des outils réalistes qui simule le mouvement d'un pinceau réel sur une toile (plus récemment repris par Yeh [YLO02]). Curtis [CAS+97] décrit la diffusion de peinture sur une surface virtuelle. Dans le domaine de la réalité virtuelle, nous pouvons retenir CavePainting[KFM+01] et les travaux de Schkolne[SMP01],[SIS02] qui proposent des outils pour créer de véritables surfaces 3D dans l'espace réel.

Le re-éclairage de scène réelle et de l'illumination inverse depuis des images réelles ont été étudiés par Fournier[Fou94], Loscos [LDR00], Boivin [BG01, **?**] avec des approches par radiosité, et Debevec [Deb98], Gibson [GM00], et Sato [SSI99] avec des approches basée image. Mais aucun de ces systèmes ne fournissent une solution temps réel excepté Gibson [GCHH03] qui nécessite un lourd pré-traitement.

A l'exception de l'approche de Raskar, peu de recherches se sont intéressées au développement d'un système interactif pour modifier l'apparence de modèles réels. Nous introduisons ici un modèle conceptuel permettant d'unifier ces différentes approches. Nous présentons ensuite nos différentes contributions avec ce nouveau canevas.

3. Le pipeline de l'interactive mediated reality

Agir sur la réalité pour changer notre vision du monde réel peut être décrit sous la forme d'un pipeline comprenant quatre niveaux (illustré sur la figure 8.5) :

- *acquisition des informations de la réalité* : nous utilisons la vue du monde, la géométrie et la réflectance d'objets réels, les conditions d'éclairage, des informations sur l'environnement et la position des objets. Chacun de ces éléments peut être acquis automatiquement (en utilisant par exemple un système de reconstruction 3D de la géométrie) ou manuellement (par digitalisation point par point) ;
- *modification des éléments réels et virtuels* : nous pouvons modifier les éléments automatiquement ou interactivement dans le monde réel, en fonction des paramètres précédemment obtenus (tel que la géométrie de l'objet réel) ;
- *alignement des objets virtuels et réels* : l'information virtuelle générée est alors alignée et calibrée avec les outils réels, les objets réels et l'environnement réel ;
- *affichage d'une nouvelle vision à l'utilisateur* : le résultat du filtrage est présenté à l'utilisateur à l'aide d'un dispositif de type casque, d'écran avec une caméra vidéo ou de type Virtual Showcase [BGW+02].

Dans cette boucle, nous nous intéressons principalement ici au premier et deuxième niveau, et plus particulièrement à une analyse plus détaillée du deuxième niveau. Dans ce cadre, nous pouvons distinguer différents types de paramètres : le contenu, la référence temporelle, la référence spatiale. Le contenu modifié peut être un objet réel ou un objet

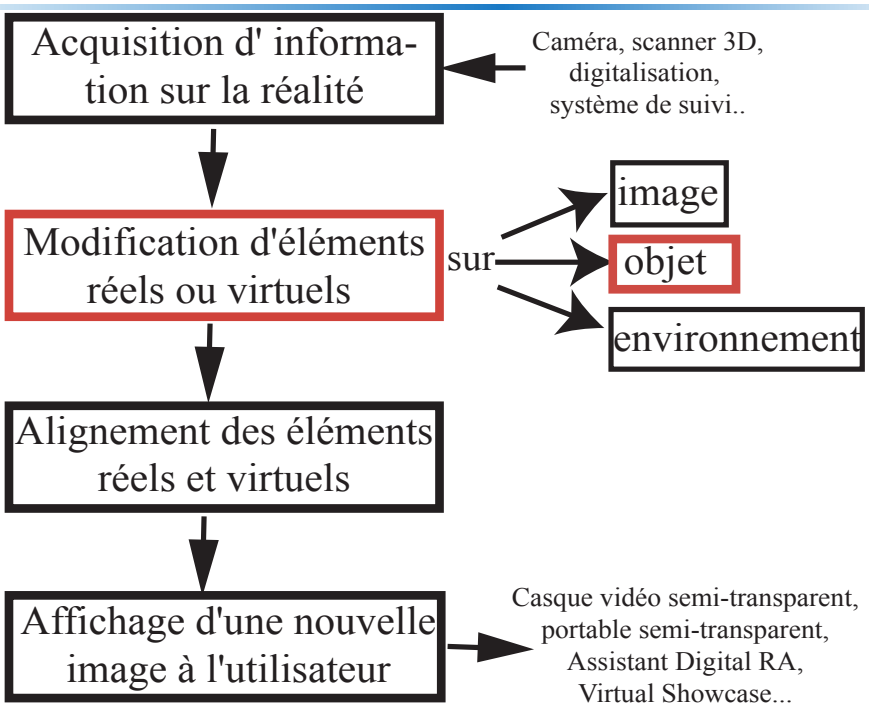


Fig. 8.5: Pipeline de l'*interactive mediated reality* : les différentes étapes pour la modification du réel (en rouge la cible de cette étude).

virtuel récemment introduit[2]. La référence temporelle qui défini la fréquence de synchronisation de perception de l'interaction : non-synchronisé, interactive ou temps réel. Et pour la référence spatiale, nous distinguons les possibilités suivantes dans la continuité des argument de Feiner [FMS93] :

- *sur l'image*. L'image finale perçu par l'utilisateur peut être modifiée. La référence spatiale 2D dépend principalement du type de technologie d'affichage utilisée, qui peut être une superposition optique, une composition vidéo ou une projection lumineuse. Avec un système temps réel, l'utilisateur peut placer des étiquettes, appliquer des effets sur l'image (telle qu'afficher l'image en noir et blanc), etc.
- *sur l'environnement*. L'utilisateur agit sur les propriétés globales de l'environnement perçu, tel que l'illumination, l'ajout et la suppression de contenu (virtuel ou réel) ou les attributs de rendu (e.g. représentation fil de fer) ;
- *sur l'objet*. Nous modifions les propriétés de l'objet réel. Un point intéressant sont les propriétés géométrique et d'apparence de l'objet que l'on peut modifier à un niveau global ou local à l'objet. Les modifications globales de la géométrie de l'objet désignent les déformations ou *morphing*. Les modifications locales sont l'ajout, la suppression ou la déformation d'une partie de l'objet. Les modifications locales de l'apparence sont le changement de couleur ou texture (avec un rendu de type Phong) à un niveau pixel/point de l'objet, ou une partie structurelle telle qu'un mur ou une porte pour une maquette d'architecture. Les modifications globales impliquent le style de rendu ou le type de matériau (non-photorealiste, brilliant, etc.).

A partir de ce canevas nous nous sommes restreint au cadre de cet thèse : collaboration sur table. Nous présentons donc maintenant nos travaux concernant des modifications sur le réel, sur la sous partie objet.

4. Modification local d'un objet : peindre, coller étaler

[2]suivant notre définition, il devient un objet réel, une fois inséré dans l'environnement

On s'intéresse à la modification d'éléments réels. Un système idéal s'adaptant à la description précédente devrait fonctionner sans phase d'acquisition hors-ligne, et dans un environnement non contraint et non préparé. Actuellement non techniquement réalisable, nous nous sommes donc restreint dans cette étude à des objets réels de *géométrie connues* acquises hors-ligne avec des méthodes robustes tel que par scanner laser, reconstruction 3D par vision, ou digitalisation manuelle. De plus, nous nous sommes limités dans notre cadre à des éléments de type maquettes de petites tailles telle qu'utilisé dans le travail journalier des architectures (et sur table).

Notre approche est dans la même catégorie que les travaux de Barrett [BC02] qui se concentre sur des technique d'éditions d'objets basé image, mais dans notre cas pour l'édition de modèle 3D (on peut aussi citer les travaux de Durand [OCDD01] avec une approche hybride 2D/3D). Nous nous intéressons aussi à fournir des outils dans l'esprit du logiciels *SketchUp3D*© [Ske] dédié au dessin d'esquisse en particulier pour des modèles architecturaux (réduisant alors la complexité de logiciels «usines à gaz», à des systèmes uniquement dédiés à la tâche des concepteurs).

4.1. Approche : métaphore du studio virtuel

On souhaite fournir aux utilisateurs une interface pour facilement modifier les objets réels. Notre approche s'inspire des techniques réelles et des environnement utilisés par les artistes, peintres, sculpteurs et designers dans leurs activités quotidiennes. Ils ont guidé notre choix d'environnement, qui se fixait pour but de reproduire le studio d'un artiste.

Dans un studio réel (figure 8.6(a) et (b)) nous avons pu identifier quatre éléments : l'espace de travail, les outils, les matériaux et le médium (l'objet sur lequel on travaille). L'adaptation de ces éléments à notre système a consisté à fournir à l'utilisateur, un espace de travail, un outil physique (*props*) avec des fonctionnalités reconfigurables dont le mouvement est suivi, une palette d'outils virtuels, et une zone de *scratch* pour esquisser ou expérimenter.

La surface de travail (figure 8.6(c)) est dans notre cadre une aire représentée par la table, où l'utilisateur peut positionner différents éléments. Les outils sont utilisés pour modifier le contenu et créer de nouveaux éléments. La palette permet d'accéder très simplement à des éléments virtuels (couleur, texture, options), avec de simples menus hiérarchiques. La zone de *scratch* sert à l'utilisateur à mélanger différents éléments avant son utilisation sur le médium.

D'après l'étude de Halper [HSS02], les designers commencent une réalisation à partir d'une idée initiale, esquissent de nouveaux éléments, procèdent ensuite par combinaison et réutilisation de différentes idées. Dans notre système, nous reprenons le même principe appliqué sur la modification interactive d'un modèle réel 3D ou d'une scène. Nous reprenons aussi les outils numériques courant (opérations d'éditions dit classiques comme le copier/coller, undo, etc.). Nous proposons pour cela trois métaphores :

- *peindre* : l'utilisateur peut interactivement peindre, déposer de la matière, de différents types : couleur, texture, matière 3D, vidéo, texte réel, etc ;
- *saisir* : l'utilisateur peut utiliser les possibilités d'un système vidéo temps réel pour acquérir des informations visuelles ;
- *coller* : l'utilisateur peut copier et coller du contenu, qui peut être réel ou virtuel, sur des modèles 3D de type étiquette planaire, objet 3D, etc.

On note ici que notre intérêt n'est pas dans une intégration parfaite, dans un sens alignement ou réalisme d'éclairage, mais plus particulièrement dans les possibilités de modification et de l'interface à fournir. Nous présentons dans la suite l'architecture spécifique utilisée.

4.2. Architecture

(a) peinture en studio

(b) peinture dans une salle

(c) on peut noter une palette d'outils sur la droite, un pinceau suivi au centre, une zone d'esquisse sur la gauche, et une maquette de maison sur la surface de travail

Fig. 8.6: Différents studio réels (a) (b) et notre studio virtuel (c).

Fig. 8.7: Photo du système en action. Equipé d'un casque suivi, un utilisateur peint sur un objet réel suivi.

Comme représenté figure 8.7, l'utilisateur est équipé d'un casque semi-transparent vidéo et agit sur du contenu placé sur la surface de travail. Le périphérique principal d'interaction est un pinceau suivi, et la palette d'outil est aussi suivie. La surface d'interaction de la table peut être individuellement reconfigurée pour placer des palettes en cartons suivies (utilisées pour les aires d'expérimentations) aux positions désirées par l'utilisateur.

Le système utilisé se base principalement sur notre architecture matérielle, avec l'Optotrak 3020 pour un suivi de haute précision du mouvement de la tête et des outils portables à la main, ce système étant complété par l'utilisation d'une librairie de suivi par vision pour certain nombre de palettes. L'usage du casque vidéo semi-transparent se justifie par la possibilité de nous fournir un contrôle totale de la vision de l'utilisateur. Cet élément est essentiel car le contrôle à un «niveau pixelique» nous offre véritablement la possibilité de modifier la *vision du monde réel*.

Dans le cadre d'une collaboration, l'application a été développée au dessus du canevas logiciel *Studierstube* [SFH+02] (bien que notre environnement logicielle convenait tout à fait aàce type d'application, des contraintes techniques nous ont conduit à ce choix). L'environnement de prototypage rapide de *Studierstube* nous a conduit au développement de fonctions d'édition interactive et à la définition de fonction associés et accessibles sur la palette PIP [SG97a].

Fig. 8.8: Menu principal de la palette fournissant des outils de peintures, étiquetage, déformations, ajout (à partir d'une illustration d'un modèle de grenouille modifié).

Le calibrage du système repose sur les méthode proposés dans chapitre 5 (pour l'Optotrak, le casque, la caméra, et le pinceau) comprenant aussi le calibrage des objets réels sur lesquelles on va interagir. Nous considérons donc à la fin de cette procédure un objet réel à une position connue, c.à.d. un modèle virtuel identique au réel aligné sur l'objet réel. Nous avons aussi un point d'interaction défini au bout du pinceau et des outils accessibles sur la palette. Nous décrivons maintenant les fonctionnalités de notre prototype.

4.3. Prototype

On présente successivement : la modification au niveau du pixel, du patch et de la matière.

4.3.1. Modification de l'objet : au niveau du pixel

L'utilisateur peut en premier lieu changer l'apparence d'un objet réel en peignant dessus (avec le pinceau suivi), comme on l'effectue en situation réel. Le déroulement de l'action est le suivant : le pinceau peut se deplacer librement, puis le pinceau rentre en collision avec un modèle virtuel (correspondant au modèle réel), on applique alors un modèle de peinture (i.d. forme du pinceau, contenu de peinture sur le pinceau, relation avec le support et propriétés de diffusion) et la peinture est diffusé et visualisée sur le modèle. Pour la mise en œuvre, nous nous sommes inspiré de l'approche du système *ArtNova* [FOL02] : détection de collision hiérarchique entre le pinceau et la géométrie de l'objet, extraction des polygones dans la zone d'influence du pinceau et modification de ces polygones en fonction des propriétés du pinceau et le type de peinture. Cette approche fonctionne dans l'espace des texture, supprimant ainsi les distorsions induites par les systèmes travaillant dans l'espace écran (voir Low [Low01] pour plus de détails).

L'algorithme de peinture est maintenant décrit. Nous considérons l'alignement effectué entre le modèle virtuel et l'objet reel. Nous utilisons une texture transparente qui sera modulée par la couleur de la peinture appliquée : un dépôt de peinture diminue le niveau de transparence laissant apparaître cet élement. Mais l'utilisation de transparence induit alors la visibilité des faces arrières d'un objet, ne correspondant pas du tout à la réalité dans le cas d'un objet réel opaque. Pour résoudre ce problème nous utilisons un modèle fantôme d'occultation[3] de l'objet réel qui est aligné avec l'objet virtuel (voir papier de Fuhrmann et al. [FHFG99] sur l'utilisation de fantômes en RA). L'éclairage virtuelle est défini arbitrairement. Contrairement à *ArtNova* notre système va beaucoup plus loin et propose une approche adaptée en fonction de l'information (acquise) disponible. Les critères retenus sont la géométrie, la réflectance et l'éclairage. Nous décrivons les différents cas d'adaptation traités.

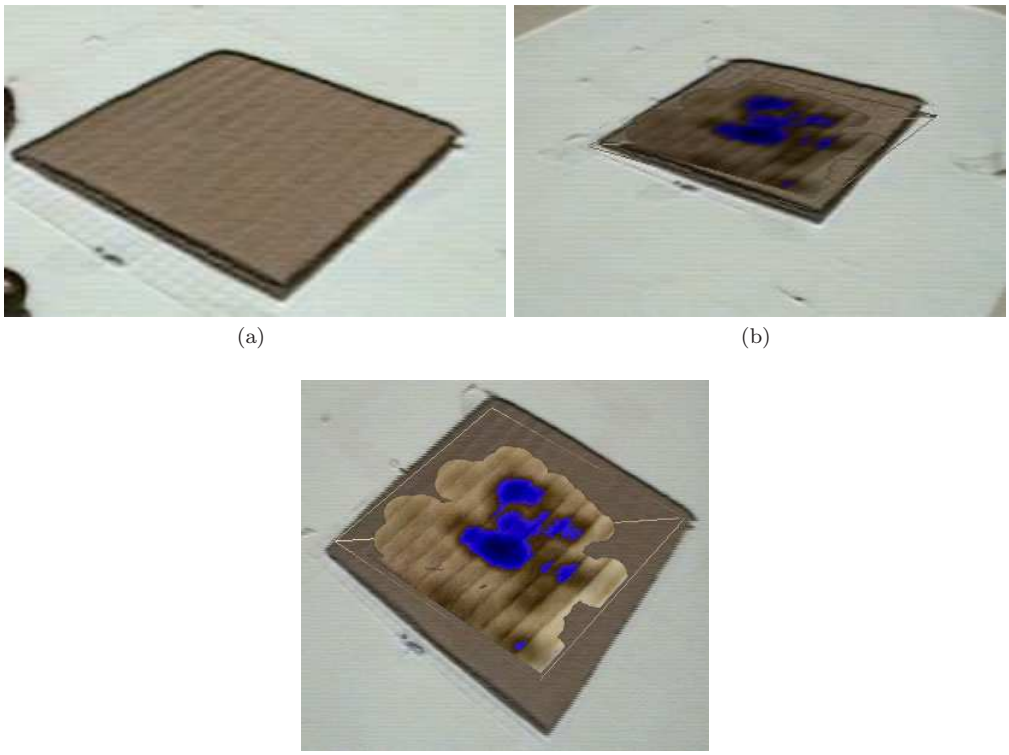
Connaissance unique de l'information de géométrie : la peinture est appliquée ici par un simple blending avec la couleur courante de la texture transparente (figure 8.9). En phase initiale, sans peinture, nous considérons juste un dépôt incrémentale de peinture, saturant le niveau de transparence, qui à son maximum passe alors en mode de fonctionnement normal. Nous remarquons ici qu'on a pas de mixage réel avec l'apparence de l'objet, mais plutôt le fait de repeindre un élément : si l'objet est rouge et qu'on dépose de la peinture bleu, on aura en sortie du bleu. Par contre ce bleu mélangé à du rouge virtuel donnera du violet.

Fig. 8.9: Peinture en mode simple : la peinture virtuelle est juste déposée sur l'élément réel.

Connaissance de l'information de géométrie et de réflectance : nous nous basons sur l'information complémentaire de l'apparence de l'objet pour réaliser un meilleur mixage (figure 8.10). Nous considérons ici un rendu temps réel, et nous nous basons alors sur un modèle d'éclairage de type Phong et de texture pour la réflectance. La texture est acquise à l'aide d'un système vidéo et les propriétés du modèle de Phong définies manuellement pour coller au mieux à l'apparence réelle. Nous observons une différence d'apparence réel/virtuel, qui reste un problème non traité ici (non résolu à notre connaissance dans le cadre génerique temps réel, voir Debevec [Deb98]). Le mixage est alors effectué entre la couleur courante du pinceau et la texture de l'objet réel. Nous utilisons aussi la propriété de transparence pour ne pas afficher la texture de l'objet réel lorsque l'objet n'est pas modifié, la vision de l'objet réel étaient alors celui de la texture réelle (le virtuel étant

[3]éroder du à des des problèmes de «flickering»

local).



(a) (b)

(c)

Fig. 8.10: Peinture avec l'information de réflectance (avant (a), après (b), (c)). Nous noterons la bonne intégration de la texture avec la peinture virtuelle, principalement les striures du cartons, mais aussi le problème d'interactions avec la différence de condition d'éclairage.

Grâce à l'application de «textures virtuelles» nous pouvons prendre des attributs difficilement modifiables dans le réel. L'utilisateur peut éclaircir (figure 8.11), assombrir, changer de contraste, appliquer la couleur opposée sur l'objet, en utilisant des outils similaires à ceux des logiciels de dessin 2D.

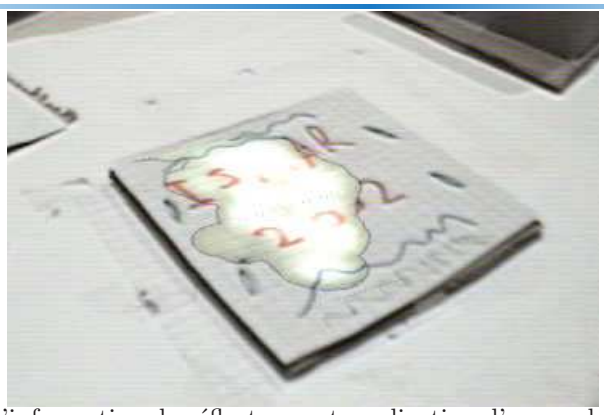


Fig. 8.11: Peinture avec l'information de réflectance et application d'un mode «éclaircir» sur un modèle 2D. Nous notons que le mode «éclaircir» supprime l'information pour de forte valeur de couleurs.

Connaissance de l'information de géométrie et acquisition de la réflectance à la volée : nous utilisons l'image fournie par la camera pour extraire une information de texture à partir du point de vue courant (figure 8.12). Nous calculons la projection du point d'action du pinceau et nous extrayons la couleur du pixel correspondant. L'erreur induit par la

perspective peut être minimisée par visualisation des éléments de façon perpendiculaire à la zone de travail. Pour éviter l'extraction de la couleur du pinceau au lieu de la couleur de l'objet, nous décalons le point 3D saisi, contraignant l'utilisateur à une direction lors de la peinture sur l'objet.

Fig. 8.12: Peinture avec acquisition de la texture à la volée : nous montrons la validité du système en re-appliquant la valeur saisie directement sur l'objet (mode de peinture copie). Nous pouvons noter que la grille réel-virtuel semble se coordonner de façon assez naturelle, avec toutefois quelques petites imprécisions..

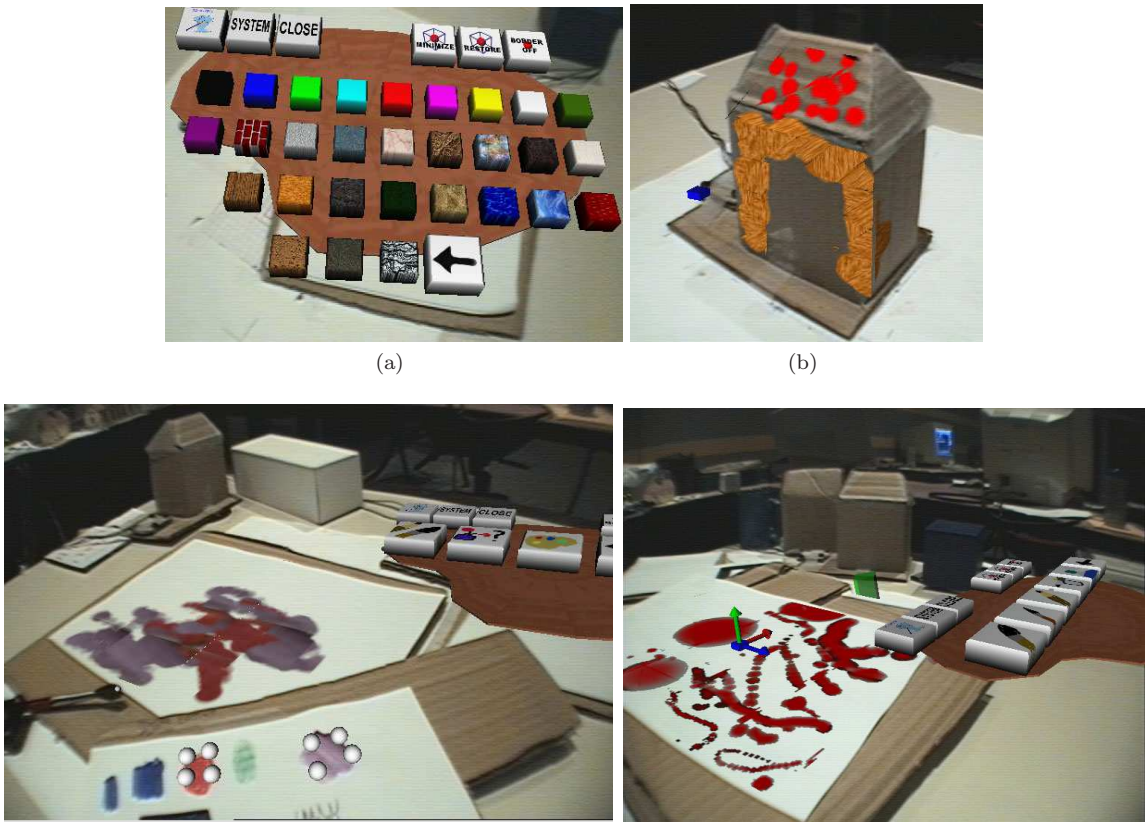
On introduit maintenant quelques fonctionnalités et liste d'outils et paramètres de notre système (figure 8.13). Au niveau des outils standards, nous reprenons la forme du pinceau,sa taille, son types, l'usage d'une palette de couleur et texture, etc.

On propose aussi à l'utilisateur une nouvelle fonctionnalité propre à l'IMR : la possibilité d'utiliser un patch réel qui sert alors comme couleur ou texture. Pour cela l'utilisateur sélectionne quatre points sur une surface réel, qui sont alors extrait depuis l'image de vue, la texture est transformé (suppression de l'effet de projection) et utilisable alors par le pinceau courant. Cette approche permet à l'utilisateur de créer interactivement une texture réel depuis une peinture réel, utilisé une texture depuis une ressource alternative, telle qu'un livre ou magazine. Si l'information de réflectance de la surface de destination est aussi disponible, il peut de la même façon que précédemment mixer la peinture en appliquant différents modes : copier, ajouter, soustraire, etc.

4.3.2. Modification de l'objet : au niveau du patch

Si la manipulation des points est trop inefficace, l'utilisateur peut sélectionner un élément spécifique sur l'objet. Un patch peut être prédéfini ou créé interactivement en sélectionnant individuellement des triangles ou en peignant un masque sur l'objet. Une fois le patch disponible, des opérations peuvent être appliquées sur l'ensemble, tel que changer la couleur ou les propriétés du matériaux. Ces éléments fournissent à l'utilisateur un haut niveau de contrôle sur l'objet.

Un autre outil puissant est la possibilité de placer des étiquettes textures (similaire à Kalnins et al. [KMM+02]). L'étiquette est une pièce texturée projetée sur la texture de destination de l'objet. Nous utilisons dans ce cadre les méthodes classiques de projection (planaire, cylindrique ou sphérique). Nous nous reposerons sur des manipulateurs virtuels pour configurer la projection sans s'attaquer aux problèmes de distorsion (exemple figure 8.14). Nous fournissons en complément la possibilité à l'utilisateur de pouvoir saisir des étiquettes depuis des objets réels et de pouvoir ensuite les coller sur un autre objet. Cette

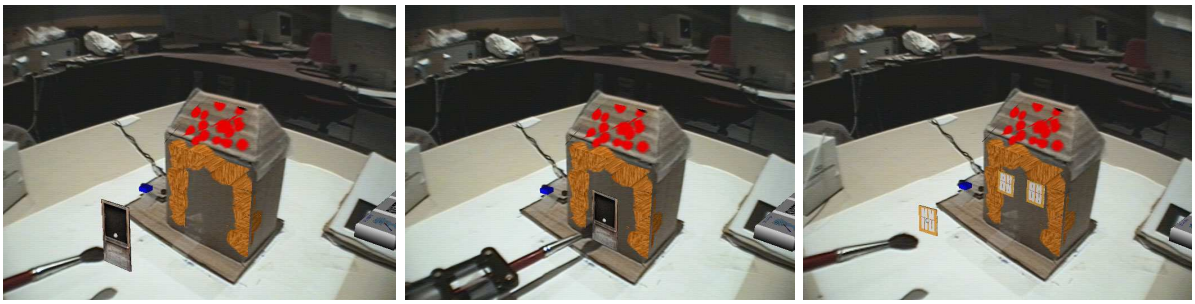


(a) (b)

(c) (d)

Fig. 8.13: Outils et fonctionnalités pour la peinture. (a) palette avec couleur et texture. (b) application de couleur et textures. (c) peinture avec une texture réel (d) résultats avec différents types de pinceaux.

possibilité offre le choix d'incorporation de texte réel, de photos ou d'images peintes à la main.



(a) (b) (c)

Fig. 8.14: Placement interactif d'étiquettes : Sur les deux premières images (a) et (b) on a le placement d'une étiquette de porte sur une maison réelle. Sur la dernière image (c), on a le placement d'une fenêtre avec l'outil «étiquette et copie».

4.3.3. Modification de l'objet : au niveau de la matière

L'utilisateur peut aussi modifier la géométrie de l'objet réel. Il peut étendre le modèle en ajoutant des éléments prédéfinis depuis une liste d'objets accessibles depuis la palette, qu'ils alignent manuellement avec la position de destination (figure 8.15).

La modification de la matière peut aussi être complétée par l'utilisation d'outils de sculp-

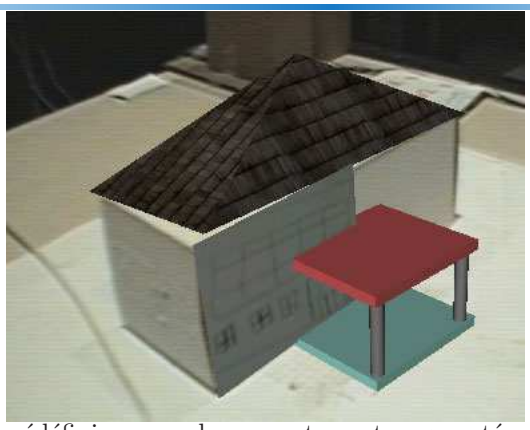
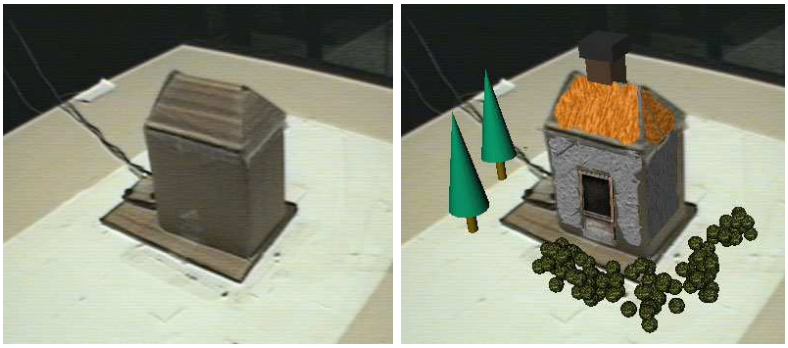


Fig. 8.15: Ajout de matière prédéfinie : un cube en carton est augmenté avec un toit virtuel, un porche, et un label réel dessiné.

ture qui lui permettent d'ajouter interactivement de la matière. La matière est déposée interactivement à la position du pinceau, et devient alors une partie de la scène qui peut être peinte ou modifier dans le même ordre que les autres objets. Nous voyons ici comment notre système permet de traiter de manière unifiée réel et virtuel, démontrant la richesse de l'IMR. Contraint par la non-connaissance du modèle de fond, nous nous sommes pas intéresser à la fonctionnalité de suppression de matière.

4.4. Résultats et applications



(a) (b)

Fig. 8.16: (a) Scène initiale et (b) résultat obtenu avec notre système

On présente maintenant les résultats de notre prototype. Les mesures sont réalisés sur notre plate-forme matériel basé sur l'Onyx 3400 avec seulement deux processeurs utilisés. Le système fonctionne alors à 30 Hz pour un objet de taille raisonable. Nous avons testé notre système avec de multiples types de géométrie : polygonale, surface courbe, surface complexe (élément scanné). Dans tous les cas notre système supporte parfaitement l'application des outils. Il y'a conflit si deux surfaces opposés sont proches, l'outil passant à travers : ce conflit peut être facilement détecté par calcul de direction des normales. par détection des normales au face.

Le contrôle de finesse de peinture est un paramètre décisif : quelle grain obtenir, quelle efficacité sur ce grain ? Il dépend principalement des limitations techniques, des possibilités du matériel graphique. Le contrôle d'un petit grain dépend principalement de la résolution de texture choisie et du bruit induit par le capteur. Dans notre cas nous limitons la taille du bout du pinceau à 4-5 mm. Pour obtenir un réalisme suffisant une texture minimum de 256x256 pixels avec des tailles d'éléments au maximum de 10 cm x 10 cm est nécessaire.

Si nous augmentons la taille des textures le fréquence d'affichage chute fortement, notre algorithme n'utilisant pas un pré-rendu en hard comme *Art Nova* [FOL02], et nécessitant un chargement des textures à chaque pas de simulation.

A l'exception du prototype *Dynamic Shader Lamps*, peu de travaux peuvent être comparés en termes de résultats à notre approche. Due à la difficulté de reproduire efficacement leur système, nous pouvons comparer empiriquement les deux approches :

- Pros :
  - peinture sur toutes matière ;
  - utilisable dans nombreuses conditions d'illumination ;
  - possibilité de se baser sur la vision de l'utilisateur pour saisir ou obtenir des informations ;
  - véritable mixage de la texture réel avec du contenu virtuel ;
  - véritable intégration d'objets virtuels 3D, impossible avec un système de projection 2D ;
  - canevas complet offrant un plus grand nombres d'outils ;
- Cons :
  - intrusion du système d'affichage ;
  - meilleur perception d'un mauvais alignement R/V et de la mauvaise stabilité due au suivi du mouvement de la tête de l'utilisateur ;
  - ccultation virtuel/réel moins naturelle qu'avec un système projectif ;
  - vision du monde réel limitée à celle de la camera, les systèmes optiques n'offrant pas suffisamment de bonne performances ;

4.5. Applications à des situations réeles

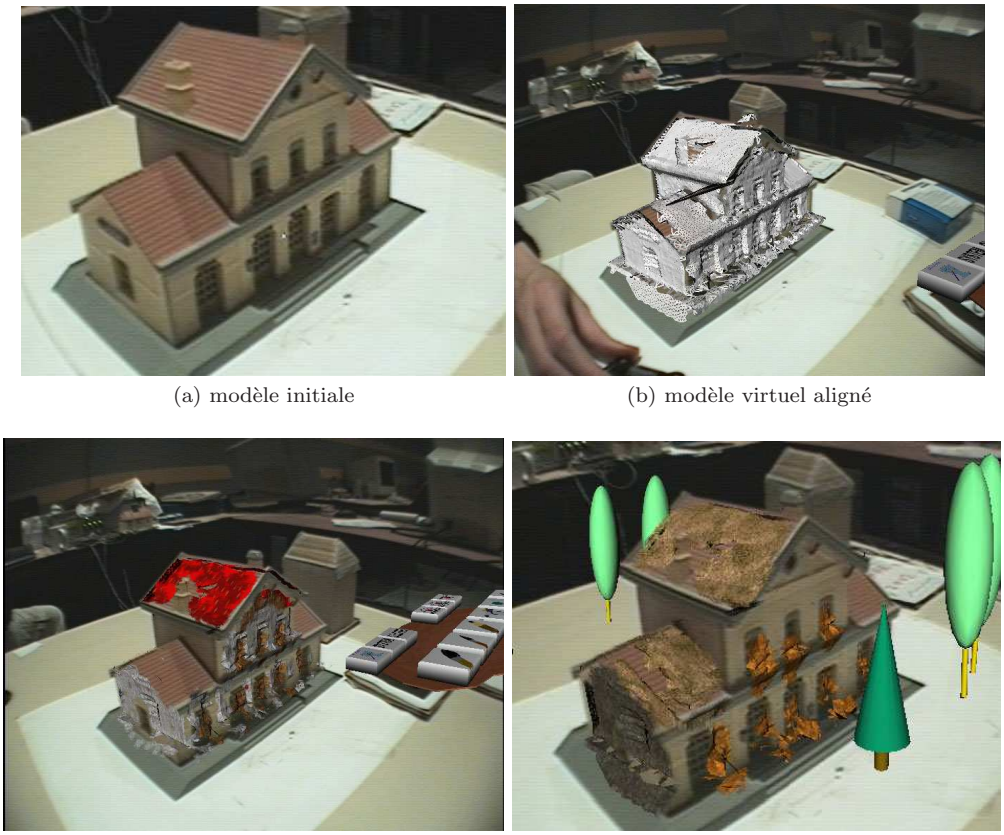
Architecture : l'utilisateur peut peindre une maquette réel pour expérimenter différents matériaux pour la future rénovation d'un bâtiment (figure 8.17). Nous avons choisi de scanner par une technique laser une maquette de jouet.

L'expérimentation montre que la mauvaise reconstruction fourni par le scanner induit un résultat non satisfaisant de façon esthétique. L'accès à des éléments de type bord de fenêtre, ornements, ne sont pas accessible et limité par la résolution du maillage et de notre système. Nous notons de plus que la texture obtenu est de mauvaise qualité, et l'auto-ombrage contraint beaucoup trop l'élément lorsqu'il est déplacé dans un espace différent de celui de la digitalisation.

Il nous semble que notre système peut aussi être utilisé avec une approche hybride 2D, pour peindre virtuellement sur un plan de sol réel, et «esquisser» les fondations (figure 8.18).

Cosmétique : un utilisateur peut utiliser notre système pour tester un nouveau type de rouge à lèvres, de style à la mode ou essayer des accessoires telles que lunette ou bijou (figure 8.19). Nous avons réalisé un prototype à partir d'une tête de mannequin en polystyrène dont le modèle a aussi été acquis. Ce prototype ne se basant pas sur un modèle de réflectance de peau, les résultats sont plutôt visuellement bon et le système facilement utilisable. Nous noterons qu'un maillage adapté aux éléments (lèvres, œil, joue) permettrai un meilleur réalisme. L'amélioration principale de ce type de système est le changement d'une configuration à casque à une approche par mirroir : un mirroir semi-transparent équipé d'une camera au dos permettrait d'augmenter indirectement l'image de l'utilisateur. Nous pourrions alors coupler la solution avec un système de suivi robuste telle qu'un suivi par vision Lepetit [LVF03] (étendu pour supporter les déformations du visage de l'utilisateur telle que dans Bartoli [BvTZ04]).

Le packaging : un styliste peut facilement modifier l'apparence du paquet d'un produit, en apposant des étiquettes ou l'assemblage d'objets annexes directement sur le prototype réel.



(a) modèle initiale

(b) modèle virtuel aligné

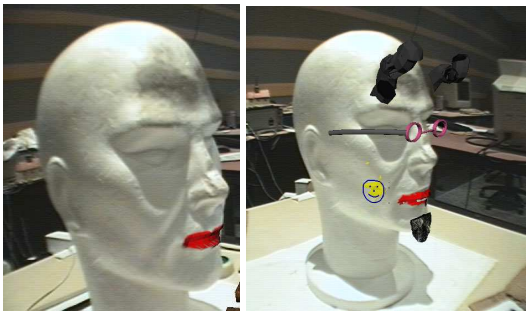
(c) résultat avec peinture sur différentes faces

(d) «scène mixte» avec intégration d'objet virtuels

Fig. 8.17: Architecture. A partir du modèle scanné d'une maquette, un architecte peu facilement tester une nouvelle représentation d'un bâtiment, tel qu'appliquer des changements à l'apparence du toit, aux portes ou redéfinir l'environnement extérieur.

Fig. 8.18: Architecture 2D. A partir d'un simple plan 2D, l'utilisateur peut tester différents matériaux pour le sol, et ébaucher une représentation volumique des fondations d'un nouveau bâtiment (ici une maison a partir du plan au sol).

Dans ce cas un prototype de couleur uniforme est suffisant, la problématique s'orientant vers des bonnes techniques de positionnement des objets annexes. Nous avons essayé le système avec une configuration d'une boite pour réaliser la création d'une boite de lait. La non finalisation de techniques de snapping limite fortement la sensation de collage des éléments sur la boîte. L'utilisation d'esquisse réel trouve ici tous sont intérêt le résultat étant facilement modifiable. Il nous semble aussi intéressant dans ce cas d'étude de fournir un mécanisme permettant de modifier interactivement depuis le dessin initiale l'apparence de celui-ci collé à la boîte.



(a) (b)

Fig. 8.19: Cosmétique. (a) application d'un rouge à levre virtuel (b) ajout d'accessoires : barbes, lunettes, tatouage.

Fig. 8.20: Packaging. Un auteur peut modifier l'apparence d'un produit, en appliquant de nouveau label sur un élément brut "non packagé" pour le présenter et le modifier devant des clients.

Le jeu : un enfant peu aisément utiliser de simple modèle d'objet réel, pour un nouveau type de jeu (figure 8.21). Les représentations cognitives qu'il se fait de certains modèle peuvent alors être plus imagé, et l'enfant dessiner son terrain de jeu virtuel telle qu'il l'imagine (rajoute des ailes virtuels a la boite, la peindre). Différentes expérimentations ont été réalisés dans ce sens (voir les images des résultats obtenus). Malheureusement les contraintes fortes de la configuration et la problématique de visualisation intrusive reste un fort problème pour un domaine où la robustesse, la fiabilité, l'utisabilité et la sécurité sont des points clefs. Néanmoins, nous croyons fortement dans ce nouveau type d'application, et nous envisageons des évaluations (hors du travail de cette thèse) à partir d'un système plus léger, définissant une aire de jeu virtuel.

D'un point de vue artistique nous avons expérimenté differents types de solutions, quelques résultats sont présentés figure 8.22.

4.6. Etude préliminaire

Une première évaluation a été réalisé dans un cadre très informelle, pour analyser l'utilisabilité, l'efficacité et les impressions des utilisateurs sur ce nouveau concept.

4.6.1. Tâche

Tâche et protocole : Le medium est une boîte en carton réel positionné au centre de la table. L'utilisateur ce sers de cette base pour réaliser sa propre «scène mixte». Un assistant présente les différents outils à l'utilisateur qui peut alors manipuler après chaque présentation d'un outil. A la fin, un temps libre lui permet de re-tester les outils qu'il

Fig. 8.21: Terrain de jeu virtuel. En utilisant de simple boite en carton, des enfant peuvent créer leur propre monde virtuel, et pouvoir alors interagir avec un nouveau type de jeu.

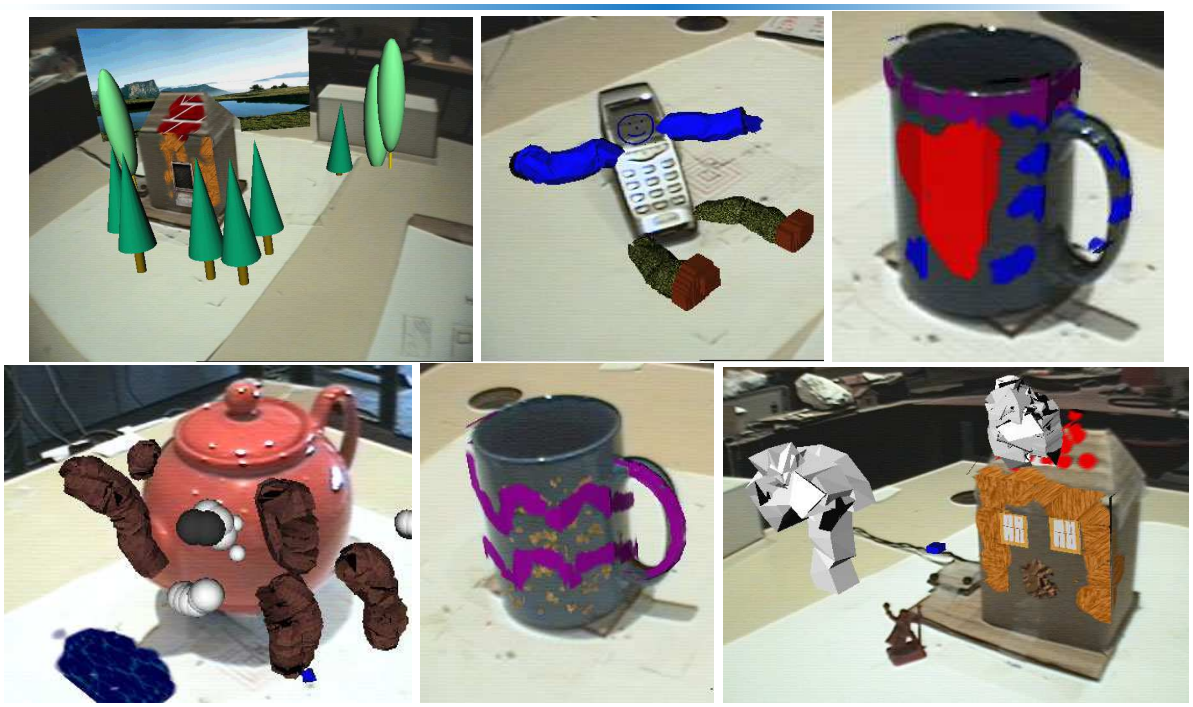


Fig. 8.22: Quelques résultats utilisateurs lors d'une expérimentation libre.

souhaitait (temps limité à 30 minutes). Nous nous restreignons à des outils de peinture sans utilisation de la réflectance et avec des outils de modification de la matière orientés dans la transformation de la boîte en «maison mixte» et des outils de modification de la scène pour en faire un espace d'habitation. Cette phase est alors suivie d'une discussion orale à partir d'une base de questions thématisées.

Sujets : L'expérimentation a été réalisée sur 6 étudiants, dont 5 appartenant à l'institut (âge approximatif de 23 ans, 4 hommes et 2 filles, droitiers, sans problèmes physiologiques spécifiques). Un seul travaillant dans le domaine des images de synthèse.

4.6.2. Résultats

Deux exemples de réalisations sont présentés figure 8.23. Nous résumons brièvement les commentaires et analyses presenter sous une forme d'avantages et d'inconvénients :

– *Pros* : Les utilisateurs ont la sensation que la peinture est réellement appliquée sur l'objet mais avec un petit décalage. Ils ont vraiment apprécié ce nouveau concept, et suggèrent de nouvelles applications. Ils n'ont pas eu de problème à comprendre le fonc-

tionnement de la palette, du pinceau, et à choisir des outils dans le menu hiérarchique. Ils ont apprécié les outils disponibles et la possibilité de choisir des éléments réels. L'environnement leur semblait familier (espace, outil, pinceau) et ils opéraient naturellement avec le pinceau.

– *Cons* : Le stylo est trop lourd et le suivi pas suffisamment stable. L'interprétation des outils difficile sans assistance, les boutons des menus trop petis. Quelques fois la perception de profondeur est difficile, rendant la peinture sur l'objet réel compliquée. Un utilisateur propose l'usage de techniques de snapping ou de grilles d'alignement favorisant la facilité d'interaction et de placement d'élément. Nous notons enfin que le mauvais alignement réel virtuel pose quelquefois des problèmes lors de l'utilisation de peinture : effet non désiré par l'utilisateur, débordement, etc.

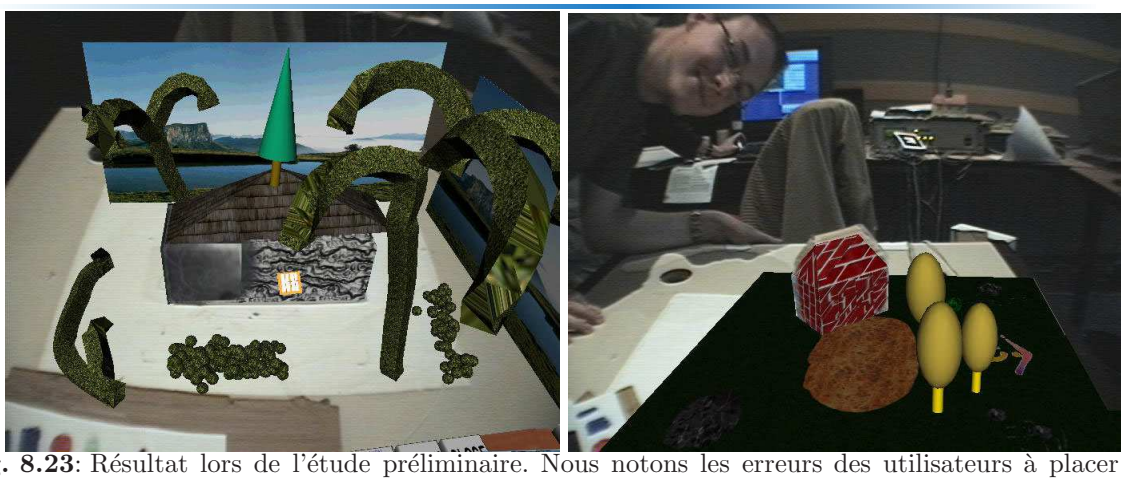


Fig. 8.23: Résultat lors de l'étude préliminaire. Nous notons les erreurs des utilisateurs à placer des éléments 3D dans l'espace sans respect de la gravité.

4.7. Etude principale

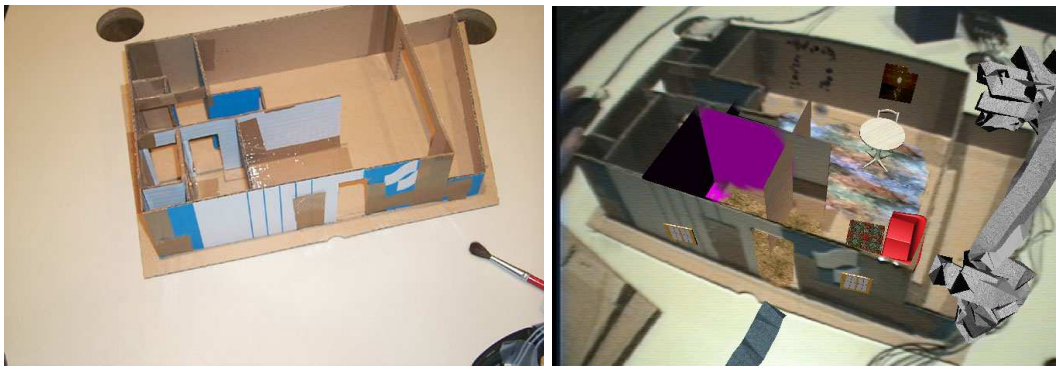
Une amélioration du système à donc été réalisée à la suite de ce premier prototype. Le stylo beaucoup trop lourd et souffrant beaucoup trop d'occultations a été remplacé par un suivi multisegment avec deux leds par segment. Un placement triangulaire - en se reposant sur les 110 degrés d'angle de vue des leds - et un choix de limitation du suivi à 5DDL nous offre alors un système plus robuste.

L'interface a été largement simplifiée les icones agrandies, leur nombres réduit et le contenu pictorial remplacé par des éléments textuel plus simple à comprendre. L'alignement a lui aussi été optimisé, mais sans l'élaboration de nouvelles techniques, principalement par affinage des procédures et résultats de calibrage.

4.7.1. Tâche

Le contenu de la tâche a été remplacé par un modèle moins générique et plus représentatif d'application réelle : une maquette réelle d'appartement (figure 8.24). L'étude complémentaire a été réalisée sur deux plans : dans un cadre grand public et dans un cadre professionnel. Notre choix s'est porté sur des étudiants d'architectures, ce domaine nous semblant très approprié aux fonctionnalités apportées par notre système. Quand au cadre grand public il nous sert à démontrer un intérêt pour ce type d'application et son utilisabilité par rapport à d'autres types de solutions de dessin (peinture réel, logiciels dessin 2D/ logiciels dessin 3D).

4.7.2. Protocole



(a) (b)

Fig. 8.24: Nouvelle configuration : (a) maquette avec le périphérique d'interaction et (b) résultat attendu, obtenu ici par un utilisateur avancé.

Pour l'évaluation grand public, les sujets sont choisis au seins de l'établissement avec un choix basé sur une variété de status/classe professionnelle. Pour l'évaluation professionnelle, les sujets proviennent de l'école d'architecture de Lyon, étudiant en 5ème année. Comme durant la première expérience l'utilisateur est assisté au cours de l'évaluation. Les mêmes outils sont présentés en donnant a l'utilisateur la possibilité d'expérimenter lui même les différents outils. A la fin de l'évaluation un questionnaire leur est présenté (voir annexe B), dépouillé à la suite des expériences. Le questionnaire est divisé en 5 parties : premières sentiments, acceptation de l'environnement, fonctionnalités des outils, applications et améliorations/commentaires. Une observation est réalisé au cours de l'expérimentation.

Sujets : L'étude s'est déroulé sur deux jours pour la partie grand public, et une journée pour la partie professionnelle. Chaque expérience durait approximativement 25 minutes (+ 5 minutes pour le questionnaire).

Étude pilote : nous avons réalisé une nouvelle étude pilote avec quatre utilisateurs. Elle a montrer une difficulté de conservation du suivi robuste du pinceau, les utilisateurs moins contraint que précédemment l'utilisait naturellement comme un pinceau réel, générant des occultations (expliciter aux utilisateurs dans la suite de l'expérience). La validation d'un choix des outils initialement avec un clavier a été remplacer par un système à pédale pour l'etude générale : l'étude pilote montre des difficultés d'utilisation (coordination globale), et le clavier à de nouveau été utilisé.

4.7.3. Résultats et Discussion

On réalise l'analyse sur 18 questionnaires. Ils comprend 10 cadres, 4 employés, 3 étudiants et une personne sans emploi. La moyenne d'age est de 30 ans. Aucune difficulté etait présente sur les sujets. Les critères qualitatifs sont convertis sur une échelle de 1 à 5.

Presque tous les utilisateurs ont trouvés ce concept interessant (16 réponse «intéressant», 1 réponse «assez intéressant»), trés simple et assez simple (4 réponses «très simple», 10 réponse «assez simple»), mais encore peu efficace (1 réponse «très efficace», 7 réponses «assez efficace», 5 réponses «mitigé»). Ils ont généralement trouvé le concept amusant.

Ils ont trés bien apppréciez le principe de peindre en virtuel dans le monde réel, et de pouvoir déplacer librement la tête pour voir sous différents angles. Ils ont aussi fortement appreciez le concept de manipulation libre de l'objet et d'utilisation de pinceau réel (on note 2 réponses «moyen»). Au vu des résultats le port du casque reste problèmatique (6 réponses «assez confortable», 7 réponses «moyen», 3 réponses «peu confortable»). Le principale défault noté est le mauvaise alignement réel/virtuel (10 oui, 8 non). Des

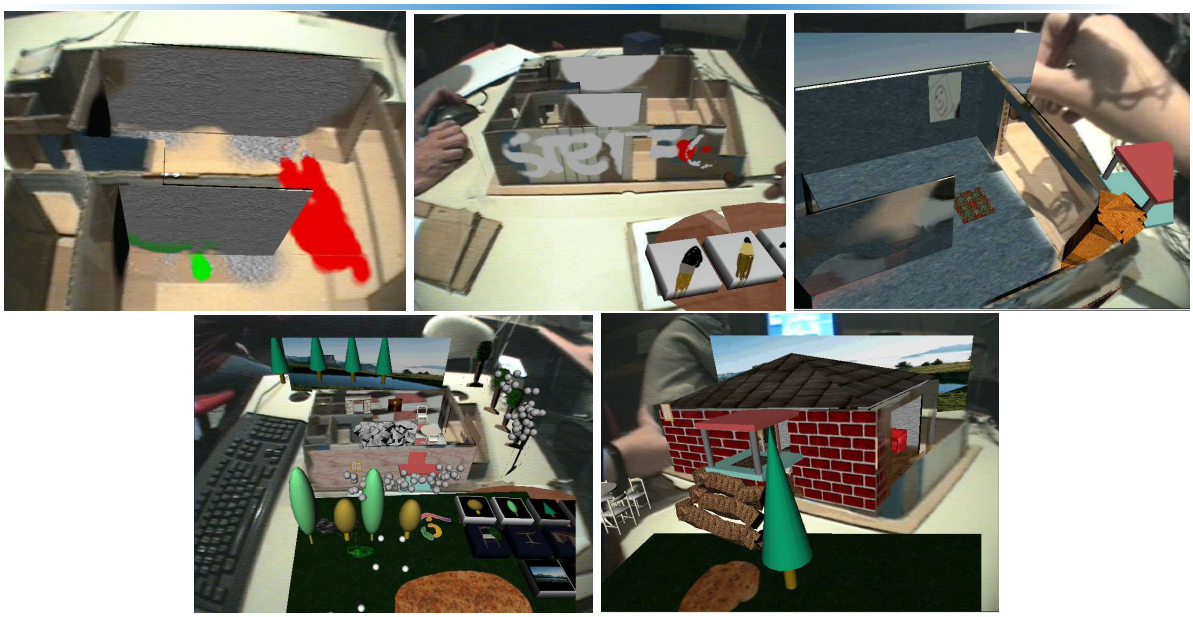


Fig. 8.25: Résultats obtenus lors de l'évaluation formelle par différents utilisateurs.

utilisateurs ont aussi noté une fatigue des yeux, une utilisation difficile à long terme.

L'usage de l'espace de travail a été trouvé bien approprié (seulement 3 réponses «moyen»). Les reproches notés étant un manque de place et la présence génante des fils. Ils ont trouvé l'usage du pinceau approprié, facile à utiliser, mais peu efficace. Le principale problème provenant d'un mauvais suivi. La palette par contre a semble être tres approprié et efficace. Par contre les zones de creations reste difficile à utiliser. Un utilisateur a propose une zone de sélection à partir de deux points au lieu des quatres. Un autre une sélection plus libre.

Les outils ont été globalement trouvés intéressantes (72% des réponses) et assez efficace (60 % des réponses). La possibilité de peindre a été une des fonctionnalités trouvés la plus intéressantes (88%), l'ajout d'éléments réels la moins intéressantes (60%). La possibilité d'ajouts d'élements virtuels a semblé peu efficace a utiliser (50 % d'avis favorables). Les utilisateurs auraient souhaités la possibilité d'effacer des éléments, d'avoir un historique des commandes ou de pouvoir zoomer.

D'un point de vue réaliste, 55% des utilisateurs ont peu la sensation que la peinture est sur la maquette réel lorsqu'ils déplacent la tête. De même les résultats sont mitigés lorsqu'ils déplacent la tête ou qu'il peigne. Par contre ils trouvent la diffusion de la peinture plutôt bonne (72 % de «bien»).

D'un point de vue comparatif aux outils traditionnels peu d'utilisateurs ont répondu. Parmis les réponses, on constate qu'ils trouvent le concept identique aux outils de dessin numérique 2D, moins réaliste que la peinture réel 2D ou 3D, et incertain dans une meilleur utilisation.

D'un point de vue applicatif, l'architecture arrive en tête (77% de oui) suivi par le jeu (50% de oui), la cosmétique (77 % de oui et peut-être), le packaging et l'art. Un grand nombre d'utilisateurs proposent ce produit pour du marketing, la décortation d'intérieur, de la présentation de produit ou du prototypage virtuel.

Nous noterons que si le casque était remplacé par de simples lunettes l'acceptation serait de 83% de l'intrusivité du système de visualisation. En remarque on note que les utilisateurs ont généralement trouvé ce concept intéressant mais le mauvais alignement induisant en complément le manque de précision reste la principale critique (manque de finesse).

Au vu de ces résultats, ils nous semblent qu'une limitation actuel est les propriétés vi-

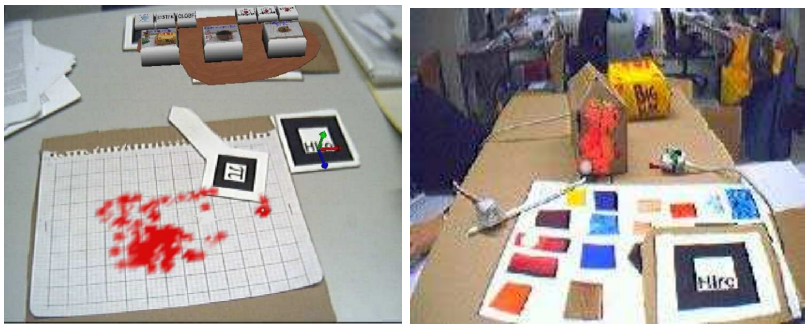
suelles insuffisantes du casque vidéo semi-transparent réduisant l'habilité de l'utilisateur de percevoir le monde réel à la même résolution que la vidéo. Nous espèrons que les prochains casque semi-transparent optiques supportant une plage de couleur plus grande et seront suffisament confortable pour un usage. Un autre obstacle semble lié au faible alignement qui detruit l'illusion de peinture sans retour haptique. L'utilisation de techniques de vision peuvent permettent de combler ces erreurs (par exemple avec une approche basée modèle).

5. Conclusion

L'interactive mediated reality présente un nouveau concept pour modifier interactivement notre vision du réel (publié dans [GGS03]). Le système basé sur des métaphores d'artistes autorise l'utilisateur à peindre, coller ou mettre des éléments sur des objets réels. Les utilisateurs ont accès à une large palette d'outils et peuvent complètement modifier la réalité. A la suite d'une première évaluation encourageante sur notre prototype, le système raffiné a suivi une évaluation plus complète au prés d'un public professionnelle et non-professionnelle. Les résultats ont montrés un engouement pour ce type de système, une approche intéressante, mais de nombreuses lacunes par rapport aux outils classiques, et la limitation des problèmes d'alignement.

De futurs travaux semble nécessaire principalement concernant les problèmes d'alignement mais aussi les problèmes d'intégration réaliste. De même une analyse plus poussée avec les professionnelles nécessite d'être menée, telle qu'en collaboration avec des architectes, des cosméticiens ou des enfants.

Dans un premier temps, nous pourrions mettre en œuvre une nouvelle architecture matérielle plus légère, et plus souple. Nous pourrions retenir une solution de bureau comme nous l'avons déjà expérimenté avec un suivi par vision et caméra de bureau sur un PC standard (voir figure 8.26). La principale limitation reste un suivi robuste et précis difficilement réalisable avec une solution de type suivi par marqueur avec la librairie ARToolkit. Une solution hybride ou une modification de la librairie pourrait être une première solution (on pourrait imaginer une sorte de compensation dynamique de la librairie par l'utilisation d'une table de correction). Cette solution permettrait alors de passer à des évaluations grand-public, qui permettrait de mieux spécifier les besoins pour une solution familiale.



(a) (b)

Fig. 8.26: Configuration légére mais souffrant de problèmes de précision : (a) peinture 2D (b) peinture 3D avec palette de couleurs.

Comme déjà cité la réalisation complète du pipeline fournirait alors un canevas complet permettant une évaluation avancée avec des professionnelles. Principalement, la modification de la matière nous semble une voie intéressante à étudier, ici limitée à l'ajout de matière qui aggrège un sous modèle géométrique au lieu de véritablement modifier le vrai

modèle. Connaissant a priori le modèle, des outils de déformation de type gonflement, extrusion ou torsion peuvent être mis en œuvre (en tenant compte des implication sur la matière et la continuité visuelle de la texture). Des outils qui peuvent conduire à supprimer de la matière nécessiteront une étude plus approfondi de l'impact sur l'utilisateur : Quelle perception a t'il en touchant de la matière qui peut être supprimée visuellement ?

Une voie intéressante est sans conteste la possibilité de peindre sur des objets avec une géométrie inconnu, autorisant un utilisateur à peindre et simultanément créer un modèle virtuel de l'objet en vol. Nous pourrions aussi fournir des outils plus tangibles, des méthodes indirectes (peinture par lancer de rayon), des techniques de rendu avancée (programmable shaders), etc. Enfin des nouvelles fonctionnalités de type rendu NPR sur le modèle réel semble une voie à étudier tout en s'assurant de la validité par des collaborations avec des artistes ou designers.

Hors du cadre de cette thèse, une mise en œuvre d'une solution mobile nous semble aussi intéressante. Un utilisateur pourrait se déplacer dans un bâtiment et pouvoir modifier l'apparence des murs et des couloirs de façon interactive. Il faudrait alors fournir un suivi hybride (GPS+vision) pour s'assurer une stabilité lors de l'interaction tout en ayant la possibilité de se déplacer librement dans tout le bâtiment.

Nous noterons pour finir que ce travail a principalement été réalisé durant 2 mois à l'institut technique de Vienne, au sein de l'*Interactive Media System (IMS) Group* sous la direction de Dieter Schmalstieg. Le système de peinture a été conçu initialement avec un système de bureau puis avec un système magnétique et finalement porté au seins de notre institut pour la continuité de ces travaux.

Aspects collaboratif

> [we] need to understand what participants actually do in a [meeting] activity in order to guide the development of technology (especially advanced computer tools) to support this activity.
>
> —Tang (1989)

1. Introduction

Nous nous intéressons dans ce chapitre à l'aspect coopératif de notre environnement. L'environnement précédemment proposé permet la gestion de plusieurs utilisateurs autour d'une table, manipulant de multiples périphériques d'interaction et visualisant simultanément des objets sur cette table. Les méthodes d'interaction définies dans le chapitre 6 ont été présentés dans un contexte que l'on peut définir par réplicatif et parallèlisable : n utilisateurs pouvant interagir simultanément et indépendamment avec du contenu virtuel. Les solutions proposées étaient, d'un point de vue collaboratif, nécessaires mais non suffisantes.

Pour cet aspect, des critères spécifiques doivent être respectés, basés sur les principes de communications et coopérations usuels, tels que la sensibilité et la connaissance des autres utilisateurs (*awareness*). La problématique (figure 9.1) a retenir au cours de ce chapitre peut alors se résumer à : « Quelles sont les besoins collaboratifs à respectés dans notre cadre ? Est ce que notre environnement supporte ces besoins collaboratifs ? ».

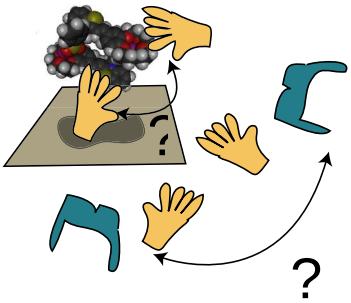


Fig. 9.1: Problématique du collaboratif en RA sur la table : quelles relations établir pour la communication entre les utilisateurs ? Quelles relations établir pour l'interaction avec le contenu applicatif entre les utilisateurs ?

Nous allons donc étudier les besoins requis pour une collaboration en RA 3D sur table (section 2), notre approche, sa mise œuvre et une analyse des résultats obtenus (section 3).

2. Contexte collaboratif

Pour fournir un support collaboratif, l'approche intuitive consiste en la réplication des métaphores, des moyens et des capacités mise en œuvre dans le cadre d'une réunion réel face à face. Cependant, ce passage semble être encore un large problème pour la réalisation de collecticiels. Quelle en est la cause ? La principale difficulté tiens au fait d'une connaissance actuelle partielle des mécanismes mise en jeu lors d'une réunion réelle.

Cette connaissance reste à ce jour une large sujet d'étude en TCAO et pourtant nécessaire pour la réalisation de collecticiels performants.

Comme notre compétence se situe dans le domaine des images de synthèse, nous avons choisi de ne pas mener nos propres études ethnographiques sur le sujet, mais plutôt de nous appuyer sur les travaux de ce domaine. Il nous a semblés plus judicieux d'analyser les résultats obtenus dans le domaine de la TCAO et de vérifier leur validité dans notre cadre : ces travaux étant généralement adaptés à des solutions de bureau ou d'informatique ubiquitaire, mais avec des concepts souvent découplés de leur mise en œuvre. Notre approche consiste alors à se baser sur ces concepts, les adapter et fournir une mise en œuvre dans un cadre de RA 3D sur table.

Nous allons présenter les critères retenus et étudier ceux qui nous ont semblés essentiels pour notre environnement, avec un classement en trois catégories : les besoins collaboratifs classique, les besoins collaboratifs en colocalisés sur table et les composantes d'activités pour du travail collaboratif.

Besoins collaboratifs standard

Comme déjà cité, les besoins collaboratifs classiques sont tout d'abord le support de communication face à face avec des métaphores naturelles, telle que verbale ou gestuelle. Elles guident alors des actions telles que la discussion, l'explication ou la désignation. Mais aussi l'utilisation de métaphores de communication non-verbale fourni par la posture, les expressions faciales où la direction du regard complètent la communication.

Goebbels [GAGL00] introduit une taxonomie pour l'interaction collaborative distribué dans les environnement virtuels. Il propose une modélisation sous la forme d'une boucle d'évolution du système collaboratif (*collaborative AAA loop*, pour *Awareness-Action-Feedback*). Visible figure 9.2, elle reprend sept éléments clefs : la proprioperception, la perception des interfaces (physique et virtuel) et des données, la perception de la co-présence, la perception des co-interfaces, la perception de la connaisance des autres utilisateur, la réalisation de l'action et le résultat (*feedback*). Démontré dans le cadre d'un système distant à deux utilisateurs ce concept semble être applicable à notre cadre.

Besoins collaboratifs sur table

Ces besoins ciblent les requis spécifiques aux cadre colocalisé, avec la notion de referentiel commun. Inkpen démontre dans le cadre d'activité sur écran sur table [IHMS01] que l'utilisation de périphérique personnels de type stylo favorise l'interaction et la communication. Kruger [KC03] étudie l'impact de l'orientation et des gestes sur une collaboration avec écrans horizontaux. Elle identifie trois types d'orientation usuels : besoin pour le groupe, besoin personnel et orientation comme symbolique du geste. Scott [Sco03] définie alors les caractéristiques conceptuelles des techniques d'orientation : rotation libre, légère, persistante, *feedthrough*, automatique et surchargeable.

Scott [Sco03] identifie aussi trois types d'espaces personnels généralement intuités dans de nos nombreux travaux de RA : espace personnel, espace de groupe mais aussi espace de stockage. Enfin Scott présente dans [SDM03] un guide pour la conception de système collaboratif sur écran sur table, dont on retiendra les principaux éléments suivants :

- gestion d'interaction interpersonnel naturel ;
- gestion de transition fluide entre activités ;
- gestion de transition entre travail/espace personnel et en groupe ;
- gestion de transition entre collaboration sur table et travail externe (information) ;
- gestion d'utilisation d'objets réels ;
- gestion d'accès au contenu partagée d'objets physiques ou digitales ;
- gestion de disposition flexible des utilisateurs ;
- gestion d'interaction simultanée des utilisateurs.

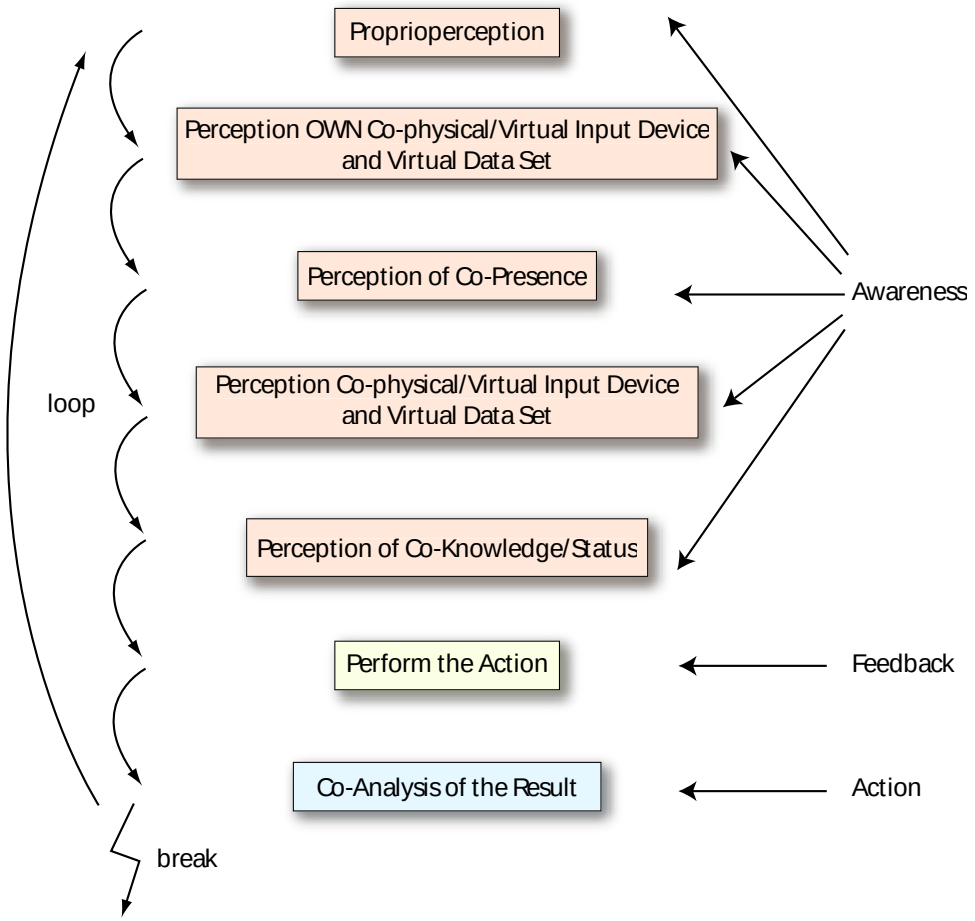


Fig. 9.2: Boucle AAA de Goebbels [GAGL00] pour les système collaboratifs.

Besoins collaboratifs pour l'activité

Nous retenons principalement les travaux de Gutwin [GG00] qui définissent les *mécanismes de collaboration* comme les tâches nécessaires au support du travail en groupe. Il décrit sept activités majeurs :

- *communication explicite* : fournir les mécanismes naturels de communication tels que verbal ou gestuel ;
- *communication conséquente* : fournir un support non direct d'informations sous la forme de retour sur la manipulation d'artefacts et la caractéristique des actions des utilisateurs ;
- *coordination des activités* : gérer l'ordonnancement et la faisabilité des tâches en empêchant la repétabilité ou le conflit ;
- *planification* : fournir des mécanismes bas niveau de division tels que la division de l'activité, de l'espace, la simulation d'une action ;
- *surveillance d'activités* : fournir de la connaissance d'informations sur les personnes (qui, comment) et leurs activités, mais aussi suivre les activités dans le cas d'apprentissage ;
- *assistance* : pouvoir fournir une aide aux autres utilisateurs de façon explicite ou implicite (nécessite d'avoir une compréhension du contexte de l'utilisateur) ;
- *protection* : consiste à éviter la destruction de données personnelles et prévenir l'accès aux autres utilisateurs.

Il introduit alors à partir de ces éléments trois critères (l'*effectiveness*, l'efficacité et la satisfaction) permettant de juger l'utilisabilité d'un collecticiel. Dans cette suite, Baker [BGG01] introduit alors une évaluation basé sur un canevas sous forme d'heuristiques reposant sur huit points :

– fournir les moyens appropriés pour la communication intentionnelle et verbale ;
– fournir les moyens appropriés pour la communication intentionnelle et gestuelle ;
– fournir les moyens appropriés pour la communication conséquente de la présence et l'«état» d'un individu ;
– fournir les moyens appropriés pour la communication conséquente des éléments partagés (*artefact feedthrough*) ;
– fournir des mécanismes de protection ;
– fournir des mécanismes de gestion de collaboration fortement et faiblement couplé ;
– fournir des mécanismes pour la coordination des utilisateurs et leurs actions ;
– faciliter la découverte des collaborateurs et l'établissement du contact.

L'ensemble de ces critères nous servent alors comme cadre théorique pour la validation de notre approche.

3. Approche et mise en œuvre

A partir de ce contexte, nous présentons nos choix et les solutions définies permettant la gestion de l'aspect collaboratif en RA 3D sur table. Nous nous fondons alors sur deux éléments présentés précédemment : les mécanismes de collaboration de Gutwin (section 3.1) et le guide de conception défini par Scott (section 3.2).

3.1. Respect des mécanismes de collaboration

Nous présentons successivement les différents éléments du cadre de Gutwin exposés précédemment et notre mise en œuvre dans ce cadre.

3.1.1. Communication explicite

Ce besoin est réalisé dans notre cadre à partir du réel, il est une propriété intrinsèque des systèmes de RA co-localisés. La communication verbale et gestuelle est conservée de façon similaire à une réunion réelle (malgré les périphériques). Notre environnement ne redéfini donc pas de métaphores dédiées mais maintient celles du réel (figure 9.3).

Fig. 9.3: Exemple enregistré en cours d'une démonstration : communication verbale et gestuelle.

Résultats : assez naturellement on note que les utilisateurs en cours de session communique facilement entre eux ou avec l'assistant démonstrateur. La communication gestuelle reste limitée par l'angle de vue du casque : la communication fonctionne mais nécessite un nombre de mouvement de la tête largement supérieur à la normale (testé lors des démonstrations par des indications gestuelles aux utilisateurs).

Comparaison avec les autres approches : notre système s'inscrit sur le même ordre que les travaux sur casques encore limités par leur faible champ de vue.

3.1.2. Communication conséquente

L'information fourni par la perception de présence des utilisateurs (*awareness*) et des éléments manipulés sont ici aussi fourni par le cadre réel. Mais l'utilisation de périphériques intrusifs limite fortement ces possibilités. Nous avons identifié en particulier trois points à améliorer : la perception des outils, du centre d'intérêt de l'utilisateur et des objets virtuels manipulés.

La perception des outils défini la bonne visibilité et compréhension du (des) point(s) d'interaction (e.g. doigt, bout du stylo) des interfaces utilisées par soi ou par d'autres utilisateurs. L'imprécision et la mauvaise connaissance du fonctionnement des outils en RA posent des difficultés pour les utilisateurs à identifier l'état de ces éléments et leurs point effectif d'interaction. La relation et superposition virtuel/réel n'est pas aujourd'hui une connaissance usuelle pour les utilisateurs. A part sous sa forme la plus connue pointeur-souris, voir baguette magique-point 3D pour les experts de RV, elle fonctionne souvent dans des espaces différents.

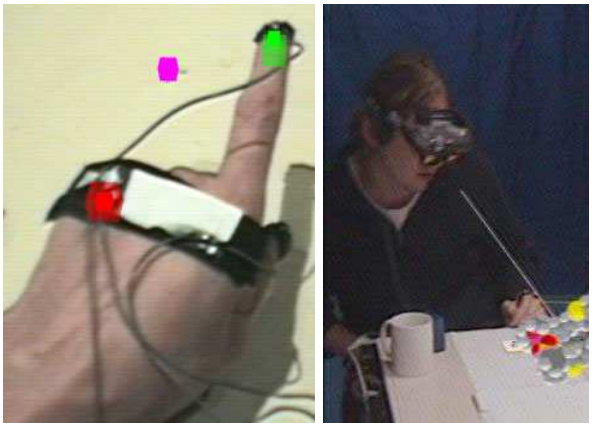
Dans ce cadre nous avons choisi d'augmenter visuellement le point d'interaction fournissant un retour visuel fort et sans ambiguïté pour soi et pour les autres utilisateurs. Pour cela nous associons au point d'interaction une représentation virtuelle définissant un pointeur ou un curseur (figure 9.4(a)). Cette représentation peut se faire dans notre environnement sous de multiples formes : objet virtuel de couleur quelconque, de couleur appropriée à un utilisateur ou avec une forme spécifique pour un utilisateur. Cette forme peut évoluer suivant l'outil manipulé et l'état actuel d'interaction. Dans notre cadre l'interaction est généralement direct. L'association entre interface-utilisateur est clairement identifiable limitant l'utilisation de mécanisme complexe (comme utilisés dans les collecticiels distribués ou avec des péripheriques à déplacement relatif [TG04]).

La mauvaise *perception du centre d'intérêt de l'utilisateur et contact visuelle entre utilisateurs* rend la coopération plus difficile. Ce type de retour est aujourd'hui fortement limité par le périphérique d'affichage, supprimant la visibilité de direction du regard visuelle et l'identification précise du point de regard de l'utilisateur. Dans ce cadre nous nous sommes basés sur les travaux de Kiyokawa [KTY99a] et nous proposons l'utilisation d'une demi-ligne de vue virtuelle défini entre la position de la tête de l'utilisateur et dirigé dans l'axe d'orientation de la tête (figure 9.4(b)). Cette solution permet alors à d'autres utilisateurs de percevoir une estimation du regard des autres utilisateurs lorsqu'on observe une tâche sur la table ou directement les autres utilisateurs. Cette technique ne nous procure évidemment pas la position des yeux (nécessite un système dédié) mais uniquement de la tête, réduisant son efficacité.

Le *retour sur les objets virtuels manipulés* joue aussi un rôle clef. Les fonctionnalités associés aux éléments virtuels et leurs états nécessitent d'être perçus globalement par tous les autres utilisateurs. Nous utilisons pour cela des retours visuels et sonores publics lors de la manipulation d'éléments publics. La sélection d'un élément, la modification de l'état du système sont indiqués à tous les utilisateurs à l'aide d'un son ou d'un effet visuel (l'affichage de la boite englobante, flash, etc.).

Résultats : La réalisation a été faite très facilement sous notre environnement, inspirés, pour la perception des outils, des travaux en SDG ([HB99], [TG04]). Une étude doit être menée pour vérifier l'efficacité de notre approche, les utilisateurs semblant très facilement assimilés la présence de ces concepts.

Comparaison avec les autres approches : *Studierstube* a été un des premiers système a augmenté la visibilité des points d'interactions mais leurs pointeur ne reflètent pas l'évo-



(a) (b)

Fig. 9.4: Résultats sur la communication conséquente : augmentation (a) du point d'interaction (b) et augmentation de la ligne de vue.

lution de l'état du système ou l'état d'action d'un utilisateur (utilisé aussi dans *EMMIE*, *MixDesign*). Les retours visuels sont généralement réalisés dans les différents systèmes et seule la démonstration *VOMAR* intègre ces mécanismes (de façon sonore). Enfin, aucun système ne demontre une utilisation de point de regard à part dans l'expérience pour *Seamless Design*. Notre système offre donc un panel unifié de solutions, garantissant une approche très souple d'utilisation.

3.1.3. Coordination des activités

Nous la d'une part réalisé par la conservation de la perception de l'espace, des autres utilisateurs et de leurs activités permettant alors de supporter fortement la coordination naturelle telle que visible lors d'une réunion. D'autre part, les outils utilisés sont adjoint de mécanismes de support de coordination (collaboration «médiatisé»). Le point étudié est l'accès aux ressources communes. Nous nous plaçons à un niveau de grain correspondant à un objet virtuel ou un document.

Nous distinguons ici deux types de relations : activités concurrentes et activités coopératives.

Dans un cadre concurrent, il faut résoudre le conflit d'accès à un élément entre deux ou plusieurs utilisateurs. Cet problématique a donné suite à de nombreux travaux principalement ciblés sur les éditeurs collaboratifs [BF91]. Nous distinguons deux approches : approche pessimiste et optimiste ([BLEE+99]).

Nous utilisons dans notre cadre l'utilisation d'une approche pessimiste, plus facile à mettre en œuvre et adaptée à notre contexte en forte interaction sociale. Notre approche est mixte, on utilise un *un protocole social couplé avec un protocole technique*. La conservation de la présence réel des autres utilisateurs et des voies de communications naturelles permettent de régler les conflits par les voix usuels (i.d. signaler à un autre utilisateur ce que l'on souhaite faire, visualiser et prédire la tache qu'il va effectuer). La solution technique consiste quand à elle à la réplication des protocoles logiciels usuels de gestions de conflits, par l'utilisation de verrous logiciels.

Lorsqu'un utilisateur sélectionne un élément nous évaluons le verrou, s'il est actif la sélection est refusé pour un autre utilisateur (représenté visuellement). La désélection de l'objet par le premier utilisateur permet de lever le verrou, l'autre utilisateur pouvant alors accéder à cet objet. Ce mode définit un mode standard généralement employé dans un cadre multi-utilisateurs.

Dans le cadre d'accés aux outils, nous proposons l'utilisation conjointe d'outils personnels (palette d'outils dans la zone personnel de l'utilisateur) et d'outils publics avec verrou.

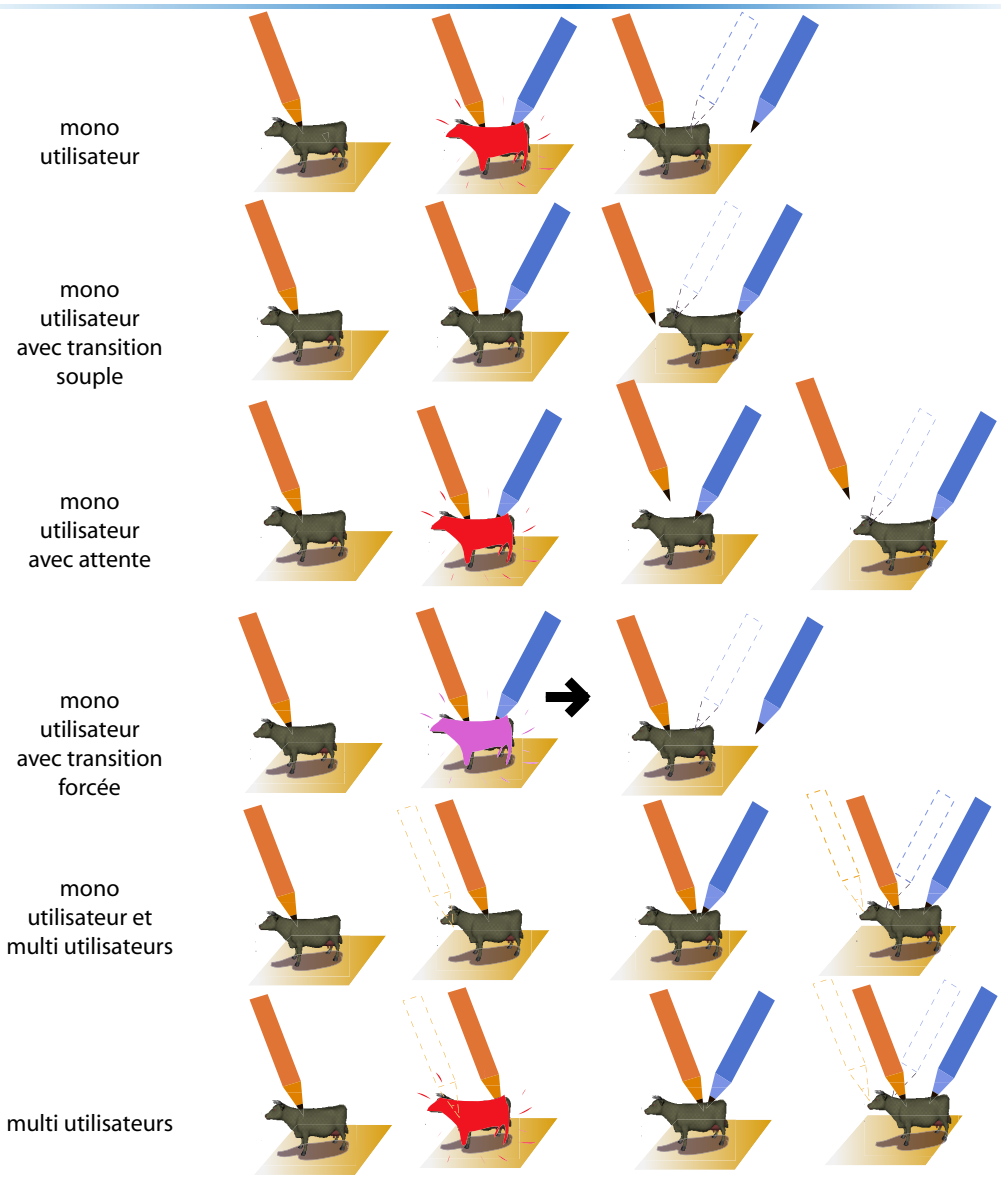


Fig. 9.5: Les différentes politiques de coordination proposées vue par un utilisateur.

Dans un cadre coopérative, plusieurs utilisateurs peuvent alors interagir sur le même objet et conduisant les techniques a être plus souple. Nous pouvons différencier plusieurs cas (représenté figure 9.5) d'après Vernier[VLS02] :

- l'objet ne supporte que l'interaction mono-utilisateur : nous retombons dans le cas présenté précédemment ;
- l'objet ne supporte que l'interaction mono-utilisateur mais supporte la transition entre deux utilisateurs. Nous sommes alors confrontés à différents sous-cas lorsque le deuxième utilisateur sélectionne l'élément :
  - le deuxième utilisateur prend alors obligatoirement possession de l'element, la technique est «prioritaire à l'arrivant» (transition souple),
  - le deuxième utilisateur doit attendre que le premier désélectionne l'élément, la technique est «prioritaire au courant» (politique d'attente),
  - le troisième cas est mixte : le deuxième utilisateur prend possession du document si le premier utilisateur ne le relâche pas (dépassement du temps d'attente, contrainte physique), la technique est alors prioritaire au plus actif ;

– l'objet supporte l'interaction mono-utilisateur et multi-utilisateurs : il faut alors définir un protocole d'interaction mono-utilisateur et son extension multi-utilisateurs, une interaction possible (à tous instant) par un autre utilisateur ;
– l'objet ne supporte que l'interaction multi-utilisateurs : dans ce cas la méthode d'interaction attend spécifiquement la sélection par deux utilisateurs de l'objet avant la mise en œuvre de la tache associé.

Résultats : notre mise en œuvre repose sur la définition de métaphores d'interaction spécifiquement collaboratives (accès à plusieurs valeurs par les interfaces des métaphores). Une métaphore peut réagir suivant la politique de coordination choisi dans l'application. Elles peuvent être attachées à un gros grain (applicatif) ou à un petit grain (objet) pour la gestion des objets.

Les classes de métaphores et de politiques peuvent facilement s'interconnecter dans le module *utilisateur* de notre architecture. La politique conserve l'état de l'objet vis à vis du support collaboratif (défini dans les propriétés de l'objet ou de façon externe). Nous utilisons un retour visuel de type clignotement autour de l'objet et changement de couleur pour signifier un conflit (couplé à un retour sonore de type *buzzer*).

Comparaison avec les autres approches : peu de travaux en RA ce sont véritablement intéressé à ce cadre, reposant alors naturellement et principalement sur des mécanismes sociales de coordination. En RV, on peut citer les travaux de Pinho et Bowman [PBF02] qui ont guidés notre choix de mise en œuvre, différents concepts se retrouvant avec leur proposition mais dédié à un cadre distant. Comme *Studierstube*, il nous semble qu'une association d'outils privées (le *PIP* dans *Studierstube* et la zone personnel dans *MARE*) favorise la coordination en supprimant les conflits d'accés aux outils applicatifs. Des mécanismes spécifiques sont démontrés dans notre cadre sur la notion d'accès à des éléments partagés pour la manipulation 3D, introduisant un certains nombres de solutions fondées sur l'étude préliminaire de Vernier.

3.1.4. Protection

La protection d'accès à des données peut être vue a deux niveaux : protection d'informations visibles aux autres utilisateurs et protection de modifications des propriétés des éléments de l'application. Dans ce cadre, nous nous sommes basés sur les concepts utilisés dans les systèmes d'exploitation de type UNIX. Lampson [Lam71] à défini un des premiers modèles nommé *modèle de matrice d'accès* (avec des modèles dérivés tels que [DS98], [BHST95]).

Le concept repose sur trois éléments : les utilisateurs (sujets ou processus), les objets (fichier ou ressources) et une matrice définissant la relation entre ces éléments : chaque case $A[i, j]$ de la matrice A, définit pour un objet i (colonne), les droit d'accès d'un utilisateur j (ligne). Le stockage de cette matrice creuse, peut être ramené à deux vues, stockage en lignes (*capabilities*) ou en colonnes (*acces control list*), réduisant alors l'information à sauvegarder (très éparse pour un grand nombre d'objet). L'intérêt étant la simplicité, l'accessibilité et l'extension et définition facile de droits. Le système UNIX propose une implémentation de ce principe par une approche à contrôle d'accés.

Nous réutilisons ce principe, en remplaçant les droits par les fonctions appropriées à notre contexte telles que *visible, deplaceable, resizable, annotable, effaceable* (cette liste étant extensible dans notre cas par une application donnée). Nous reprenons le concept d'utilisateur, de groupes, et de droits par défaut (contrairement à UNIX, on définit explicitement les droits pour chaque groupe, et utilisateurs d'un groupe). Nous considérons qu'un utilisateur ne peut avoir qu'un groupe terminal (un seul père dans une representation par graphe), mais les groupes peuvent être hiérarchiques (dans le même ordre que le modèle de Dewan [DS98]). Nous choisissons des règles d'inférence hiérarchique sur les groupes,

Utilisateurs	*Visible*	*Deplaceable*	*Annotable*
Defaut			
Architectes		✓	✓
Maurice			
Clients			✓
Joe			
Robert			

Fig. 9.6: Gestion du contrôle d'accès aux éléments virtuels : modèle par matrice de contrôle d'accès. Nous notons que les architectes peuvent déplacer et annoter les objets et Maurice a une vue spécifique en face caché. Les clients ont une vue haute définition et peuvent annoter les éléments. Par défaut, Nous voyons le modèle en fil de fer et nous avons aucun droit de manipulation.

l'héritage des droits pour un utilisateur est alors défini par la plus proche définition de ces droits au sens de la distance dans la graphe de structure des groupes (exemple figure 9.6).

Les droits tels que définis dans UNIX sont généralement à deux états : actif ou non-actif. Le cas du droit «visible» est plus complexe qu'un état binaire, pouvant être multi-variables. Smith [Smi96] définit une représentations de cette visibilité sous une forme matricielle, un axe définissant une apparence globale (définition géométrique) et un autre axe définissant des modificateurs (états éléments). Dans notre cas, on choisit de définir une liste de représentations possibles d'éléments, les paramètres nous semblant être représentable à une complexité plus élevée qu'une composition à deux variables. Nous pouvons alors associer à l'utilisateur : un modèle par défaut, une liste de modèles prédéfinis accessibles, une liste de paramètres ajustables par l'utilisateur. Le choix est alors laissé à l'application offrant le plus de souplesse par rapport à un modèle figé peu évolutif (le contenu de la case de visibilité peut être alors de différents types).

Un point souligné par Butz [BHF+99] et Szalavari [SEG98] concerne la modification dynamique du contrôle. Dans notre cadre nous considérons pas de propriétaire d'éléments, mais un ensemble de droits qui lui procurent ce statut. De plus nous choisissons que soit une personne responsable d'une session collaborative définisse ces droits («*root*» de session), soit un objet incorporé en cours de session apporte ses propres droits. La personne responsable peut alors aussi modifier ces droits en cours de session (à l'aide par exemple d'une console graphique).

Résultats : la mise en œuvre a été réalisé sous la forme d'un module pouvant être associés a l'application : il représente un filtre entre les techniques d'interaction et l'accès aux objets. Une structure spécifique permettant l'extension des droits est instanciée pour chaque objet et consultée à chaque accès sur cet objet (les droits sont définis sous forme textuelles, permettant une pérénité et une vérification aisée). Pour le droit de visibilité, on reprend l'approche de Szalavari [SEG98], par l'utilisation de valeur de masques associés à l'utilisateur et à l'objet. Chaque utilisateur et groupe sont identifiés par un champ dans une chaîne d'éléments. L'objet comprend alors lui aussi un champ (masque), et une combinaison par opération booléenne défini alors le choix de l'élément affiché, voir figure

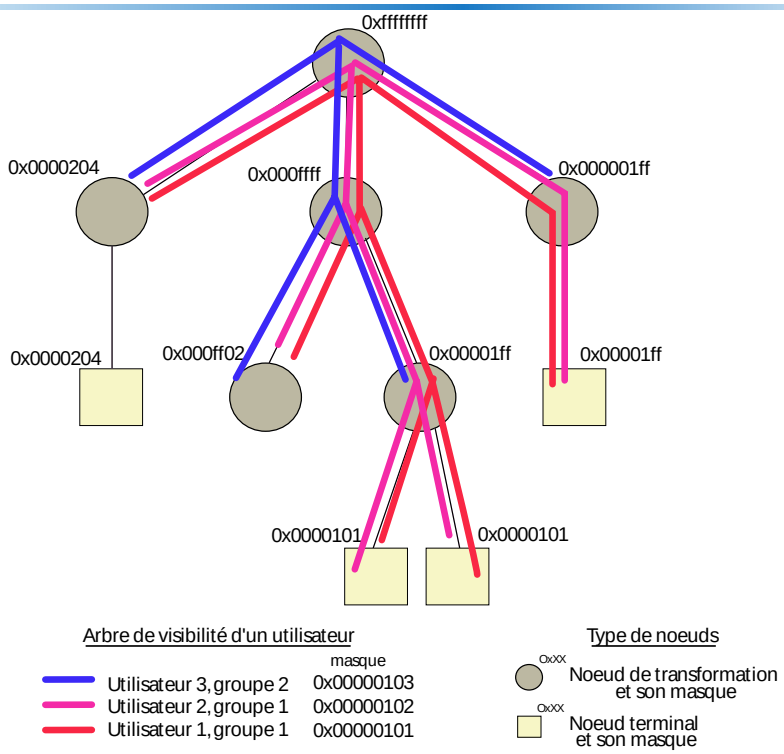


Fig. 9.7: Exemple de gestions de droits de visibilité pour trois utilisateurs.

9.7 (24 clefs possibles dans l'implémentation). Une procédure automatique permet de calculer la valeur de cet clef associé aux objets.

Si nous reprenons les critères définit par Dewan [DS98], notre système est simple (matrice d'accés), sans surcharge pour l'utilisateur, extensible, flexible et facilement administrable (fichier de configuration ou GUI). La figure 9.8 illustre un exemple de scénario résultant.

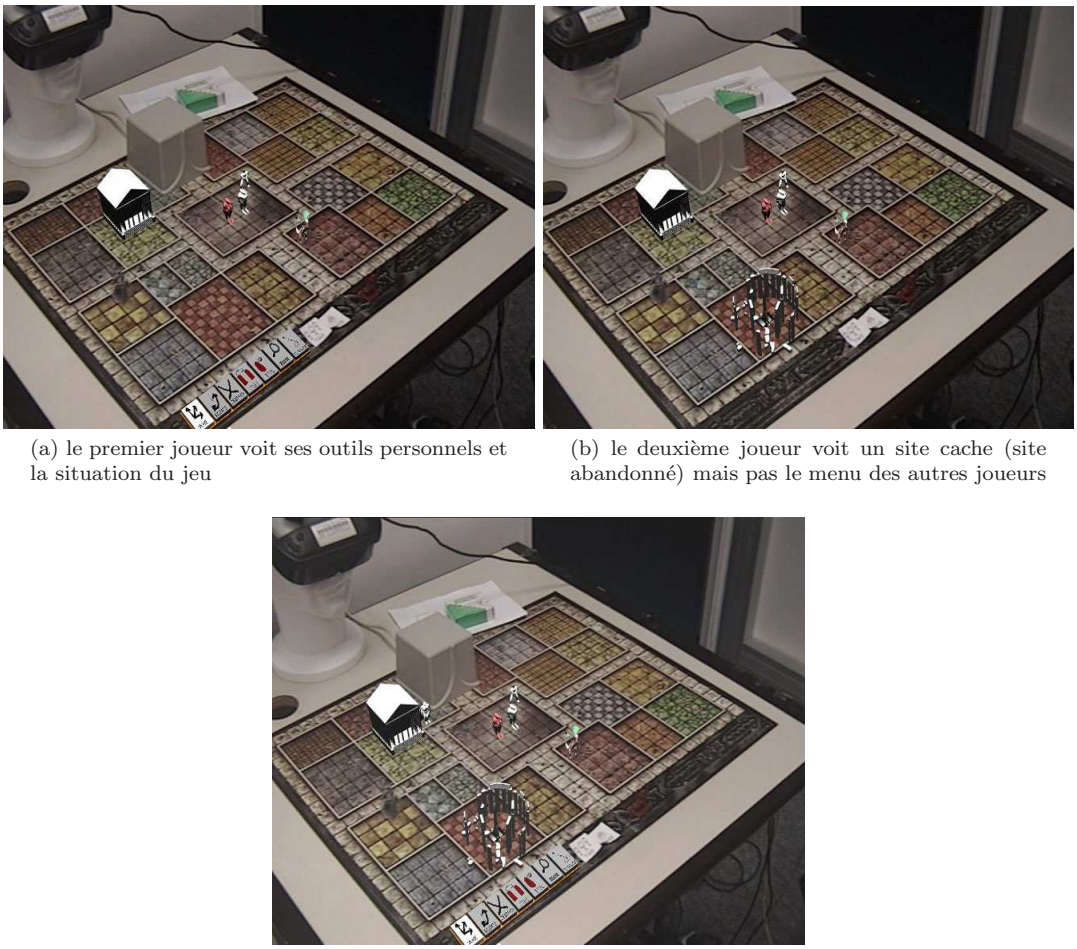
Comparaison avec les autres approche : *EMMIE* et *Studierstube* sont les seuls à proposer une gestion des droits des objets. *EMMIE* propose un mécanisme souple pour le contrôle en cours de session des droits, mais se limite à une configuration spatiale (un objet dans une zone à un droit spécifique, défini par une «lampe virtuel» et visible par un «mirroir virtuel»). Aucune information sur la mise en œuvre n'est fourni. *Studierstube* définit de même des zones de configurations. Chaque utilisateur a des droits dans ces zones, limitant l'usage pour un utilisateur d'un droit différent pour deux objets d'une même zone. Le modèle utilise *Open Inventor* pour la mise en œuvre des droits (sous forme de clefs et masque), notre solution reprend ce concept sous *OpenGL Performer* mais avec une indépedance du module de gestion.

3.1.5. Assistance

Notre architecture introduit différents services d'assistance à l'utilisateur, guidant l'utilisateur dans la compréhension, l'apprentissage ou l'usage d'un outil, d'un contenu de l'application. Nous introduisons une *aide publique*, présentation commune avant une session et une *assistance personnelle*, accessible par un utilisateur en cours de session.

L'assistance publique est défini sous forme de présentations interactives multimédias. Elle permet à l'ensemble des utilisateurs de visualiser et de coopérer pour la compréhension du système (au sens applicatif).

Pour l'aide personnel, nous reprenons les métaphores usuels des bureaux numériques que nous avons identifié sous trois formes. Le premier est l'accès à un contenu d'aide, accessible sous la forme d'un bouton graphique ou d'un élément de menu. L'utilisateur peut alors soit



(a) le premier joueur voit ses outils personnels et la situation du jeu

(b) le deuxième joueur voit un site cache (site abandonné) mais pas le menu des autres joueurs

(c) le maître du jeu voit tous les éléments du jeu

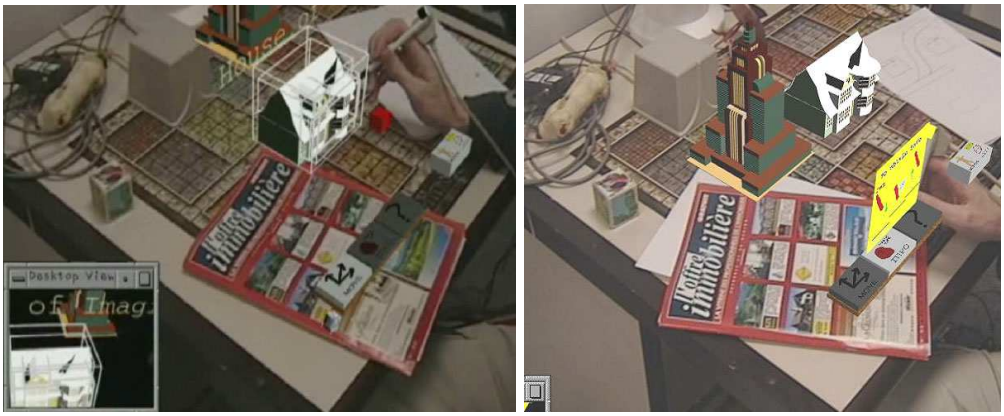
Fig. 9.8: Résultats : vue de trois utilisateurs de la même scène dans le cadre d'un jeu.

cliquer sur des éléments et obtenir une aide, soit naviguer dans une interface hypertexte pour trouver l'élément de réponse. Le deuxième type d'aide est l'aide contextuelle associée à une fonctionnalité, généralement représente sous formes de *tooltip* : il se base sur le concept de persistance d'un pointeur virtuel à une position donnée induisant alors un utilisateur en attente de réponses, et une réponse sous la forme d'une description textuelle réduite encadrée et positionnéesur l'élément.

Le troisième type d'aide proposée est l'agent conversationnel, actuellement encore peut présent dans des logiciels commerciaux, qui n'a pas été traité ici.

Résultats : les deux premières techniques ont été mise en œuvre (figure 9.9). Chacun de ces concepts ont été implémentés sous notre architecture dans l'espace 3D. Le *tooltip* sous forme textuelle 2D qui s'active au bout d'un certain temps de sélection constant en position statique. L'aide par boutons sous formes d'une plan indicatif comprenant explications textuelles et visuelles. Nous notons que l'orientation verticale choisit contraint fortement l'angle de lecture, une approche par *Bilboard* serrait une meilleur solution.

Comparaison avec les autres approches : *Tiles* intègre des mécanismes similaire aux nôtres sous la forme de *tooltip* affiché au cours du temps suivant l'orientation d'un outil (*tangible tooltip*) et propose un outil d'aide avec une information affiché sous forme d'avatar et de bulles de dialogue sur une carte réel (*tangible bubble help*). Ils ne fournissent pas de description précise du mécanisme est évoqué. Au niveau des agents, nous noterons l'agent Welbo [AKYT00] dans le cadre de support d'agencement de mobilier intérieur, la communication reposant sur un échange vocale avec des règles d'évolution de l'agent basé



(a) (b)

Fig. 9.9: Résultats : aide de bureau avec (a) tooltip (affichage texte au dessus de la boîte englobante) et (b) bouton d'aide (affichage plan d'aide au dessus du bouton activé).

sur observations vidéos.

3.1.6. Surveillance d'activités

Nous choisissons principalement dans notre cadre de surveillance d'activités en cours de session et post-session. En cours de session la forte corrélation des utilisateurs autour de la table limite considérablement le besoin d'un grand nombre d'informations sur l'activité des autres utilisateurs, nous conduisant à ne pas leur fournir explicitement des informations de surveillance.

Par contre, il nous a semblé judicieux de fournir des informations après la session sur l'activité déroulé durant la réunion. Dans la continuité des travaux en RV de Schäfer [SPOG01], nous proposons des outils permettant de «rejouer» la session, pour une post-analyse personnelle ou l'accès au déroulement de la session en cas d'absence de la réunion.

Résultats : dans ce cadre, un service spécifique reçoit et sauvegarde un ensemble d'éléments d'activités au cours de la session. Pour ce fait les différents services de l'application transmettent de façon asynchrone, un certain nombre d'événements (typé par le service), la tâche, et ses paramètres. Le choix retenu est basé sur une description textuelle (XML). Un lecteur graphique permet alors de rejouer la session contrôle par une *timeline*.

Notre système se limite dans notre cas à des événements de type discret et l'enregistrement de flux continu (audio, vidéo) n'est pas actuellement réalisé, pourtant nécessaire pour une bonne compréhension post-session (vcoir les travaux relatifs en informatique ubiquitaire). Dans ce cadre le support de RA nécessite alors un enregistrement vidéo externe car l'intégration d'objets réels (catalogue, maquette) non précisé au système est alors complètement ignoré par l'analyse post-session. De plus, on souhaiterait avoir des mécanismes plus haut niveau identifiant des tâches complexe (assemblage d'une pièce) permettant une navigation sémantique plus intéressante.

Comparaison aux autres approches : dans le cadre de RA 3D peu de travaux ce sont intéressés à ces problématiques, notre mise en œuvre initial reste fortement incomplète due à la complexité de gestion des deux mondes. En effet, la RA nécessite une prise en compte de l'activité réel et l'identification des critères importants reste à définir.

3.1.7. Planification

La planification de l'activité fixe une bonne répartition des tâches et une facilité de réalisation d'un but. De la même façon que précédemment nous ne fournissons pas de mécanismes

au cours de session, la définition de mécanismes génerique est difficile en considérant une abstraction de domaines. Nous nous sommes par contre intéressés à des mécanismes pre-session qui permettent de planifier le contenu de l'application et le rôle des utilisateurs, leur espaces avant le démarrage de la session.

Nous nous basons sur les outils définis dans le chapitre 4, le lecteur pourra s'y reporter. Nos outils proposant un ensemble avec des interfaces graphiques et une sortie sous format XML.

Comparaison avec les autres approches : *MixDesign* et *MagicPlatform* introduisent des outils de définitions de relations entre objets réels et virtuels pre-session mais pas de formalisme spécifique en cours de session. Nous noterons que la forte utilisation de format textuel sous *Studierstube* permet alors une configuration et planification aisée post-session des ressources utilisés et utilisable en cours de session (*OpenTracker*, *APRIL*, *Open Inventor*, etc.). Notre approche se situant donc dans la même lignée que les travaux précédents avec un ensemble plus homogène d'outils.

3.2. Respect des besoins pour un travail collaboratif co-localisé

Nous présentons successivement nos choix pour les différents éléments du cadre de Scott :

- *support d'interaction interpersonnel naturel* : notre système reproduit des analogies réels et supporte globalement les principes de Gutwin. Il fournit aussi une gestion du privé et du public inscrit dans un respect ergonomique collaboratif ;
- *support de transition fluide entre activités* : la complémentarité des interfaces (et outils) proposés permet un support par multiplexage temporelle mais aussi spatiale. Par contre peu de mécanismes explicites de transition ont été étudiés (changement d'activité par sélection virtuel d'un nouveau outil) ;
- *support de transition entre travail/espace personnel et en groupe* : la définition de multiples espaces et sous espaces virtuel assure une transition entre ces différents contenus gérés par notre interface haut niveau (chapitre 7). D'un point de vue ergonomique le respect des espaces personnels réels procure une bonne gestion de l'espace personnel. La transition est aussi possible sous la métaphore de copie de contenu permettant une transition physique simple entre les deux types d'activités ;
- *support de transition entre collaboration sur table et travail externe (information)* : notre système permet, d'une part par un cadre naturel d'interaction, une conservation des métaphores et outils traditionnels. D'autre part, les mécanismes explicites basés sur les méthodes d'ajout d'objets (chapitre 5, section 5), ou l'utilisation d'outil informatique classique (ordinateur portable) assurent un accès aisé à du contenu externe ;
- *support d'utilisation d'objets réels* : nous nous appuyons sur l'utilisation d'interfaces tangibles privées et publiques (section 2.6) ;
- *support d'accès au contenu partagée d'objets physiques ou digitales* : aussi garanti par notre architecture, les utilisateurs peuvent facilement manipuler de façon collaborative du contenu et réalisér une visualisation personnelle à partir d'un ensemble de copie (chapitre 7) ;
- *support de disposition flexible des utilisateurs* : les utilisateurs peuvent facilement redéfinir leur espace de travail, son orientation et sa position basé sur le concept d'interface haut niveau présenté chapitre 7) ;
- *support d'interaction simultanée des utilisateurs* : supporté intuitivement par notre architecture reposant sur une gestion de boucle temps réel, l'interaction simultanée est alors transparente pour les utilisateurs.

Notre environnement adhère donc fortement au requis et au guide pour l'interaction sur table colocalisé, ici en RA 3D sur table.

4. Conclusion

Nous avons proposé à partir de deux cadres théoriques les besoins collaboratifs assurés par notre système. En premier ordre, les besoins de communication, de protection, d'assistance fournissent des mécanismes avancés constituant une base solide pour le support de ces tâches. D'autres (coordination, planification, surveillance d'activités) nécessitent un large travail complémentaire même si un grand nombre d'éléments sont fortement dépendant d'un contenu applicatif.

En second ordre, nous respectons les besoins conceptuels définis dans le cadre de SDG et appliqué ici à notre environnement. Des mécanismes et interfaces de transition restent à définir pour l'adaptation et l'*awareness* aux utilisateurs (telle que la plasticité des interfaces [CCT01]).

En deçà d'avoir démontré la validité du respect du collaboratif en RA et l'identification complète des besoins spécifiques à ce contexte, notre analyse démontre et propose en préambule un support conceptuel complet de l'aspect collaboratif dans un cadre de RA. Une analyse plus complète est alors nécessaire pour une première validation : on l'expose au chapitre suivant.

Chapitre 10

Applications et retours d'éxpériences

A partir de notre cadre méthodologique défini section 6 nous désirons dans ce dernier chapitre évaluer l'intérêt de la RA dans un cadre d'applications collaboratives 3D sur table. En complément, nous nous intéressons à la validation de notre proposition d'environnement d'un point de vue de son intérêt, son utilisabilité et de son efficacité dans ce contexte. Nous noterons qu'il n'existe pas actuellement de méthodes spécifiques d'évaluations pour les environnements de RA, nous utiliserons alors principalement les approches d'IHM ou de RV.

Nous nous sommes orientés sur différents domaines qui nous semblaient correspondre à un échantillon suffisamment diversifié pour démontrer l'intérêt de notre système : nous avons choisi les domaines du jeu (section 2), de la chimie (section 3) et l'architecture (section 4). D'un point de vue générique, nous nous appuyons sur un certains nombre d'éléments recueillis au cours de diverses démonstrations (section 1).

Les résultats présentés ne sont que des *résultats préliminaires*, de nombreuses autres évaluations sont nécessaire pour démontrer une véritable validité du concept. mais ils constituent une première étape dans une approche centrée utilisateur [GHI99].

1. Retour d'éxpériences

Diverses évaluations informelles, discussions, présentations au cours de ces trois dernières années nous ont amenés à identifier quelques points sur l'utilisabilité de notre environnement. Nous présentons brièvement les points retenus pour ces expériences.

Intérêt et application : les utilisateurs trouvent en général le principe très intéressant mais l'aspect prototype les font considérer une utilisation dans un futur lointain. La présence de fils et d'un environnement trop intrusif leur laisse des doutes sur un usage à court terme. Un certain nombre est impressionné par l'aspect de manipulation d'objets virtuels, qu'ils associent au même principe que la réalité virtuelle (interaction 3D, navigation dans un monde virtuel). Quelques utilisateurs notent toutefois des principes spécifiques à la RA comme le mélange de maquette réelles et virtuelles ou l'intérêt de voir facilement les autres utilisateurs.

Utilisabilité de la visualisation : Les utilisateurs ont généralement une phase d'apprentissage dans laquelle ils découvrent la relation entre le casque suivi et le contenu visualisé. Après cette phase d'apprentissage leurs mouvements deviennent plus naturels, et les utilisateurs ont tendance à rapprocher leur tête du contenu ou le regarder sous différents angles. Par la suite, plusieurs utilisateurs font remarquer que les objets virtuels ne restent pas tout à fait en place. Avec le casque vidéo ils ont général du mal à ressentir la profondeur et l'occultation du contenu réel par le virtuel. Ceci leur pose des problèmes lorsque leurs mains ou leurs doigts disparaissent sous ce contenu. Avec le casque optique, nous notons en général, une critique de la mauvaise qualité d'image très loin de la qualité des écrans de bureau utilisés au quotidien. Nous noterons que la latence affecte aussi les mouvements, les utilisateurs ayant tendance à réduire la vitesse de mouvement qu'ils auraient aux naturels.

Utilisabilité de la manipulation : Les utilisateurs ont vite tendance à vouloir manipuler le contenu virtuel avec leur main, le toucher. Ceci explique peut être la plus forte facilité d'utilisation des interfaces de type digitales qui sont en général mieux perçu par les

utilisateurs. L'interface de type stylo trouve aussi une facilité de compréhension mais les problèmes de prototypes (erreur de précision de la technique, capteur défait de sa base) gêne souvent les utilisateurs. Les interfaces de type tangible n'ont pas été expérimentées. Les techniques d'interaction sont assez vite assimilées même si des erreurs de précisions affectent la réussite : les utilisateurs tentent alors d'autres essais jusqu'à la réussite.

Aspect collaboratif : les utilisateurs ont généralement tendance à s'observer à travers les casques dans une phase initiale. Par la suite, ils leur semblent très naturel de discuter, d'indiquer ou de regarder le même élément. Le faible champ de vue est souvent noté par les utilisateurs leur semblant trop restrictif pour la visualisation de contenu partagé. Le principe d'espaces privés et publics est en général peu exploité dans les prototypes présentés : néanmoins les utilisateurs ont appréciés le principe de menus virtuels privés dans leur espace personnel. L'interaction simultanée ne leurs posent pas de difficultés très bien gérer par le système.

Comme evoqué, ces premières remarques ne présentent que quelques détails génériques sur l'ensemble des éléments présentés dans les chapitre précédents. Une étude plus poussé sur chacun des points est nécessaire (comme l'aspect privé/public, le mélange réel/virtuel, l'aspect d'interaction tangible, etc.).

Nous présentons dans la suite l'étude de différents domaines spécifiques. Nous noterons que pour chacun des domaines choisis, une étude et une définition des besoins ont été effectuées. A partir de ces informations, notre environnement a été spécialisé, et un prototype réalisé. Nous complètons alors ces prototypes par des évaluations sous différentes formes pour démontrer la validité de notre approche.

2. Application : ludique

2.1. Contexte

Le domaine du jeu collaboratif nous a aussi semblé approprié à ce type d'environnement (figure 10.5). Un grand nombre de jeux de plateau, de rôle, tirent parti du support table, et d'un besoin d'interaction élevé entre utilisateurs (gestuelle, verbale, etc). Les premiers travaux réalisés dans le cadre de mon DEA (et à partir des travaux de Szalavari [SEG98]) s'étaient attachés à identifier les apports de la réalité augmentée dans ce type de jeu, orienté ssur un scénario de jeu de rôle. L'idée principale retenue n'était pas de fournir une version RA des jeux vidéos traditionnels, mais plutôt d'augmenter les jeux de plateau traditionnels avec les possibilités du numérique. Les principales caractéristiques qui avaient alors été identifiées étaient :

- conservation de l'espace de communication collaboratif et de l'espace de tâche présent dans les jeux de plateaux ;
- conservation du support tangible (pions, cartes) fournissant une facilité d'interaction et un support visuel efficace ;
- conservation des possibilités d'interactions naturelles, verbales ou gestuelles entre utilisateurs ;
- ajout des possibilités de calculs (état du jeu, stratégie) automatiques avec le système ;
- ajout des possibilités de persistance de l'informatique : accès distant, sauvegarde ou contrôle dynamique de l'état du jeu (état précédent, vitesse, etc.) ;
- ajout des possibilités de vues publiques et vues privées virtuelles ;
- ajout des possibilités d'animations et d'effets spéciaux présent dans les jeux vidéos : animation des personnages, des objets, effets pyrotechniques, etc.

2.2. Etat de l'art

(a) (b)

Fig. 10.1: Jeu : (a) exemple de jeu vidéo [Art03] et (b) jeu de plateau réel [dCTE].

Comme déjà évoqué, différentes études ont été menées sur le concept du jeu collaboratif devant un écran ([SSI00]), ou plus récemment sur PDA [SH03]. En interaction homme machine, l'apport numérique a permis de mettre en œuvre de nouveaux types de jeux hybrides tel que proposé par le *Tangible Media Group (Medialab)* du MIT ([GOI98], [IWO+99]).

Dans un cadre de RA sur table, Szalavari [SEG98] a proposé une version du Mah Jong basé sur *Studierstube* introduisant les avantages de conservation des métaphores naturelles de communication. Il propose l'utilisation de méthode de *snapping* pour la robustesse de l'interaction, de clef privée et de gestion des données en couche pour la protection et du *PIP* comme zone personnelle. Récemment, Schmalstieg *PIP* défini un jeu sur table basé sur des interfaces tangibles dans le cadre d'un jeu d'action (suivi par caméra placée sous une table semi-transparente). Il énonce alors la facilité d'utilisation des éléments réels tangibles permettant l'utilisation de métaphore d'interaction simple, sans fils, avec retour tactile. Starner [SLSP00] propose dans le cadre du projet *Perceptive Workbench* un jeu de RA distribué. Différents projets se sont plus particulièrement penchés sur l'analyse de ce type de jeu : Demeure et al. [DC02] sur l'étude d'un jeu de plateau 2D, Wichert [Wic02] introduit pour un jeu collaboratif distant basé sur un concept de *Tetris*© ou le groupe d'études *ARPE* [Tabb] avec des démonstrations de jeux sur table.

Le *Mixed Reality Laboratory* de Tokyo a largement étudié différents types de jeu en RA proposant et intégrant à chaque fois de nouveaux types de périphérique[1] : *AR2 Hockey* [OSYT98], *RV-Boarder Guards* (*AquaGauntlet*) [OSYT99] ou *Contact Water* [Mur01].

Un engouement récent a aussi trouvé place dans des projets de jeux mobiles. En intérieur, Cheok [CYY+02] a développé un jeu de recherche d'objets reposant sur une visualisation par casque et une interaction à l'aide d'une interface dérivée d'une manette de jeu. Lyons [LGS00] propose un concept de jeu basé sur le son. En extérieur, le projet *ARQuake* [TCD+00] s'intéresse à reproduire le jeu vidéo *Quake*© dans les rues d'une petite ville, basé sur un périphérique d'interaction 3D et d'un système porté par l'utilisateur dans le dos. Bouchet et al. [BNRP02] propose aussi un jeu en extérieur de recherche d'objets, tandis que Cheok [CWY+02] propose un *PACMAN* en RA.

2.3. Approche

L'approche retenue s'inscrit dans une continuité des travaux de DEA. Nous avons choisi un jeu de type jeu de rôle, qui fourni à l'utilisateur un fort niveau de collaboration et

[1] régulièrement présenté à l'*Emerging Technology* de *Siggraph*

d'interaction. L'application visée est le jeu Heroquest© qui est un jeu de plateau qui permet de respecter le contexte collaboratif sur table (notre version nommé *Heroquest AR*).

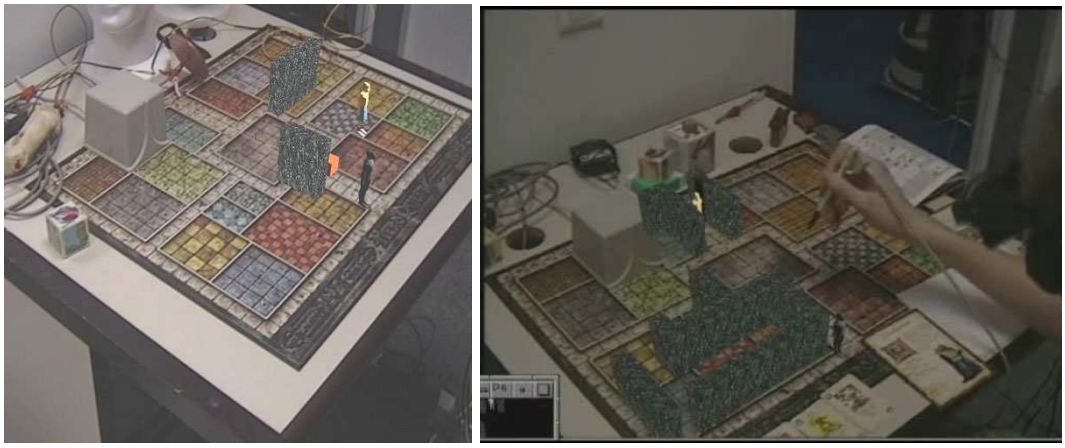
Le principe du jeu consiste à réaliser une quête en déplaçant son personnage sur un plateau, lancer des sorts, combattre des PNJ et chercher des pièges et trésors. Le jeu est coopératif, chaque joueur est assigné à un rôle et collabore avec les autres utilisateurs pour réaliser la quête.

Dans notre cadre chaque utilisateur est équipé de casques suivis. Ils peuvent interagir avec les personnages virtuels positionnés sur le plateau de jeu.

Les avantages identifiés du virtuel dans ce jeu sont : le calcul automatique des règles de combats, la représentation des personnages, des décors, mais aussi le concept de visibilité du point de vue d'un personnage[2]. Nous conservons le plateau de jeu réel, les pions des joueurs et ennemis et les cartes de jeu. Un utilisateur, maître de jeu, contrôle le déroulement d'une partie, à l'aide d'une console ou d'une vue spécifique dans le casque. L'utilisateur peut déplacer son personnage à l'aide d'un pion suivi ou de l'interface *mixed-pen* par l'intérmediaire de la métaphore *2D slide*.

2.4. Prototypes et résultats

Le prototype realisé se limite à la gestion d'un utilisateur, le deuxième guidant alors la quête. L'interaction est démontrée à l'aide de l'interface stylo. Dans sa quête l'utilisateur déplace son joueur dans un dédale de murs, ceux ci apparaissent au fur et à mesure de l'avancée du personnage. Nous proposons un scénario alors très basique faisant intervenir recherche d'objet, action avec PNJ et combat. Nous utilisons l'espace image du casque pour afficher des informations lors des différentes étapes importantes (initiale, finale, rencontre avec PNJm etc.). Deux exemples de l'application sont visibles figure 10.2. Le plateau est défini à partir d'une description textuelle, les règles du jeu sont décritent de façon procédurale.



(a) vue d'un joueur après quelques déplacements, les murs sont ajoutés dynamiquement suivant le champ de vue du personnage virtuel d'un joueur

(b) phase de fin de jeu, où le joueur à trouvé une clef après discussion avec un PNJ, et se dirige vers une porte. Nous noterons que les éléments réels (carte, feuille) ne servent ici que d'illustration

Fig. 10.2: Prototype du jeu *Heroquest AR*.

Les différentes démonstrations (sur une dizaine de personnes) ont prouvées un fort engouement pour cette application malgré la médiocrité du prototype. L'utilisation de retour sonore et l'apparition dynamique d'éléments ont prouvés une bonne complémentarité des jeux de plateau réels.

[2]dans le jeu les objets réels sont positionnés au fur et à mesure de leur apparition dans le champ de vue d'un personnage

La question de l'utilisation de pions réels pour les personnages virtuels reste largement posée : si les personnages deviennent tangibles, l'interaction est favorisée, mais les possibilités virtuelles réduitent (pas d'animation, de déplacement automatique), dans le cas contraire, les possibilités sont inversées.

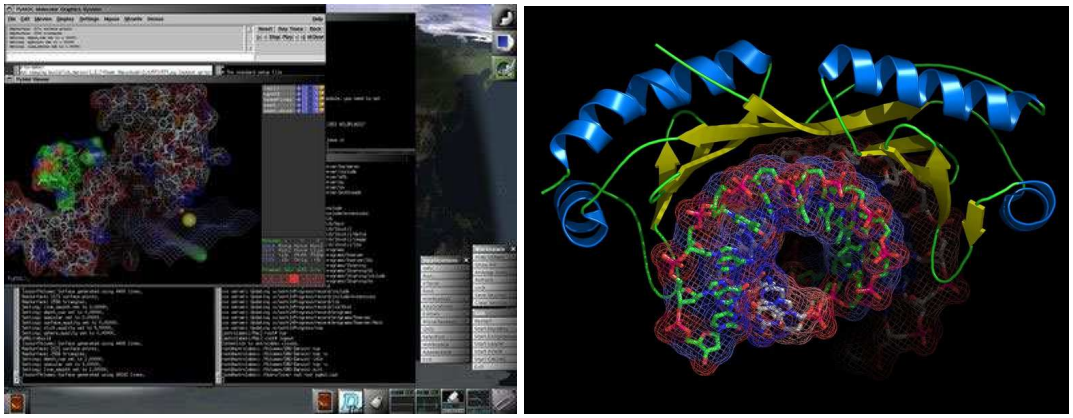
De même, les interfaces de type carte à jouer, offrent une aisance dans la manipulation mais nécessitent d'être identifiées par le système. Il nous semble qu'un système par vision permettrait d'identifier ces éléments tout en conservant les avantages du réel (une augmentation virtuelle des cartes retournant une information d'identification correcte par le système).

Une activité grand public serait l'occasion d'évaluer le système, mais la principale difficulté est la restitution de notre environnement sur des machines plus portables et tout aussi puissantes.

3. Application scientifique : chimie

3.1. Contexte

La chimie a elle aussi intégrée depuis plusieurs années l'outil informatique dans son cadre de travail. Au delà d'un outil d'analyse ou de modélisation, il est utilisé comme moyen de visualisation 3d de structure moléculaire (figure 10.3). Le chimiste peut de cette façon accéder à une représentation naturelle, directement interagir avec ; simuler des expériences ou confronter visuellement ses résultats à d'autres types de solutions.



(a) (b)

Fig. 10.3: Chimie : (a) outil de visualisation 3D et (b) rendu de structure chimique [PyM].

Dans ce cadre, il nous semble que la chimie et ses dérivées peuvent être un autre domaine applicatif de notre travail : des chercheurs autour d'une table (lors d'une réunion) pourraient facilement discuter, visualiser des structures moléculaires 3d placées au dessus de la table. Il pourrait se référer facilement à des documents textuels ou esquisser sur papier d'autres solutions. Ils pourraient utiliser des solutions tangibles pour la manipulation interactive de molécule 3D en place de solutions de bureau actuels ou des solutions en RV (stylo, pointeur, etc.).

3.2. État de l'art

La visualisation moléculaire est un sujet qui a conduit à de nombreuses collaborations avec des laboratoires de réalité virtuelle ; Les recherches visant surtout la visualisation temps réel de large structure moléculaire, la réalisation de nouvelles techniques d'immersion et de

manipulation, et la définition d'architecture couplant les environnements avec des logiciels de simulation.

Par exemple, le projet *Nanomanipulator* [TRC+93] s'est intéressé au retour haptique fourni par des structures nanomoléculaires à l'aide d'un bras télescopique couplé avec un système de visualisation stéréo. Le projet *PIT* (présenté chapitre 3) permet à deux utilisateurs de manipuler des molécules virtuelles visualisées au dessus d'une table.

Dans le cadre de la RA, Morton Fjeld [FV02] s'est intéressé au côté éducatif, il propose un système de modélisation de molécules à partir d'une interface tangible (*AR chemistry*) : un cube réel sert pour orienter une molécule, un cahier réel comme container d'atomes et une interface de type baguette magique pour la sélection et l'assemblage des atomes (la visualisation étant sous forme d'un large écran vertical en face de l'utilisateur). Slay [SPVT01] propose précédemment à ces travaux, une approche similaire avec un baguette magique pour la sélection d'un atome sur une structure moléculaire associée à un élément rigide 2D.

Weghorst [Weg03] s'est intéressé à l'aspect haptique des molécules et a réalisé un prototype couplant la superposition d'une représentation moléculaire virtuelle associé à une maquette réelle de molécule (base sur *ARToolkit* et sur *PyARTK* [pya]). Ces premières expériences démontrent un fort intérêt de cette approche, mais la nécessité d'un meilleur cadre logiciel d'utilisation. De même le projet *SPACE Cube* [Gos01] présente une démonstration ou l'utilisateur peut joindre (avec une approche spatiales) des cubes réels (conteneur de molécules) et ainsi assembler et créer des structures moléculaires complexes. Enfin, récemment Mulder et al. [MB04] propose dans le cadre du projet *PSS (Personal Space Station)* [pss], un environnement de bureau reprenant le concept de cube réel avec une interaction par stylet pour la sélection d'atomes ou l'accès à un menu (avec le portage de l'application *VMD* [VMD] dans leur environnement).

3.3. Approche

Nous nous intéressons dans notre cadre à reproduire un environnement similaire au PIT pour l'aspect collaboratif, mais avec une approche de RA. Dans un cadre de biochimie, nous avons choisi de réaliser une coopération avec le CERMAV (Centre de Recherche sur les Macromolécules Végétales)[3]. Au delà d'analyse et d'études basées du matériel de mesures (résonance magnétique nucléaire, spectrométrie de masse, microscopie électronique) une grande partie de leur travail nécessite une modélisation moléculaire tridimensionnelle, ainsi que sa visualisation.

Une prise de contact a donc été réalisé avec Alain Rivet (ingénieur de recherche responsable visualisation) et Karim Mazeau (chercheur CNRS actif en visualisation 3D). L'approche choisie a été de réaliser une étude à partir d'une entrevue sur l'applicabilité de nos travaux à ces domaines, et voir si notre système pouvait répondre à leur besoins (d'interaction, de collaboratif).

L'étude s'est montrée plutôt positive : l'absence actuelle d 'outils de visualisation 3D dédiés pour du travail collaboratif montre un intérêt pour notre type d'environnement. Le principal besoin se situe dans le couplage avec leurs outils de simulation, la visualisation servant à analyser le résultat et raffiner leur modèle pour relancer la simulation. Les techniques d'interactions se restreignant à une navigation basique (*trackball*) ou des sélections de sous-parties de structure moléculaires. La modélisation est généralement réalisé avec des outils annexes 2D. L'utilisation de documents papier est nécessaire pour esquisser des modèles, des idées sous forme de représentations schématiques 2D.

Les besoins retenus sont :

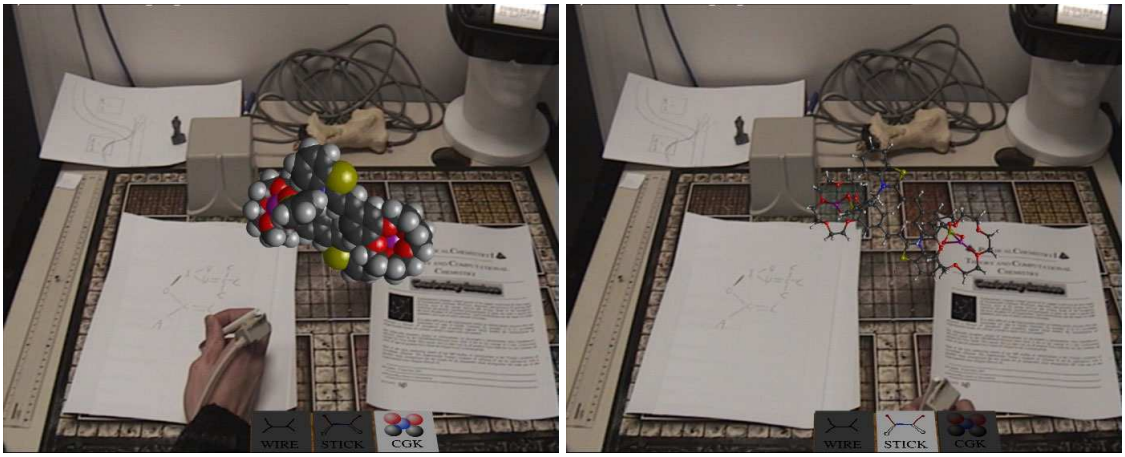
[3] laboratoire CNRS, rattaché à l'université Joseph Fourier qui travaille principalement sur l'étude de la biomasse et l'étude des oligos et polysaccharides

– la visualisation collaborative de molécule 3D («trackball» et immersion) ;
– la modification de la répresentation (modèle fil de fer, surfacique, affichage du solvant, etc) ;
– la possibilité de manipulation de molécule 3D ;
– la possibilité d'assembler interactivement des molécules 3D (*docking*) ;
– l'utilisation conjointe de documents et schémas scientifiques virtuels et réels ;
– un couplage avec les logiciels de simulation.

Comme précédemment, nous avons mis au point une suite de prototype pour l'interaction avec des molécules 3D.

3.4. Prototypes et résultats

La figure 10.4 montre un prototype démonstratif préliminaire. Un menu public permet le changement d'apparence d'une molécule sous trois modes : fil de fer, «bâtons et boules» et *CPK*. Deux utilisateurs peuvent alors manipuler cette molécule, la déplacer et s'échanger cette molécule, la navigation libre leur permet une visualisation sous plusieurs angles. Les modèles sont convertis depuis le format PDB (Protein DataBase) au format Inventor pour pouvoir être accessible dans notre environnement. Nous noterons que la faible résolution du casque limite fortement la visualisation avec la représentation en fil de fer.



(a) interaction mixte avec contenu réel (feuille, documentation) et virtuel (modèle, menu)

(b) changement d'apparence de la molécule au format stick and batons

Fig. 10.4: Illustrations du prototype de chimie.

Pour le prototype applicatif, un modèle de travail, assemblage de cellulose, nous a été fourni (au format PDB) et servait de base de test. Il a été converti et optimisé pour une visualisation à fréquence raisonnable dans notre environnement. Dans ce cadre la simulation se limite à une interpolation entre le résultat initial et final (obtenus précédemment par un logiciel de simulation propre au CERMAV).

Nous présentons brièvement le prototype mis en œuvre. Deux utilisateurs placés autour de la table peuvent visualiser des molécules 3D placées dans des documents tangibles (modèle fourni de cellulose et modèles simples obtenus sur le web). Ils ont accès à des articles scientifiques et feuilles réelles pour prendre des annotations. Chaque document tangible comprend un modèle réduit de la molécule d'étude. Ils peuvent les sélectionner, la molécule recouvrant alors l'espace de travail. Chaque utilisateur dispose de plusieurs outils accessibles dans sa zone personnelle sous forme d'une barre d'outils : déplacement, navigation, information, simulation, changement de représentation.

L'utilisateur peut déplacer et tourner la molécule à l'aide de l'interface tangible 2D+1D, mais aussi de l'interface digitale. La navigation se fait à l'aide de l'interface tangible

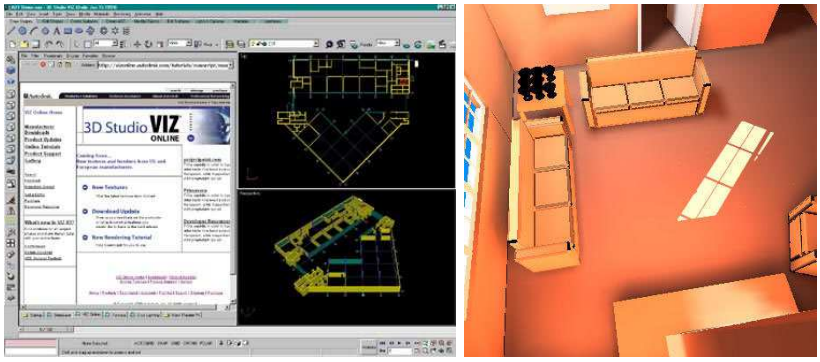
2D+1D. Ils accèdent sur le portable à des informations textuelles ou le contrôle de l'interface de simulation. La simulation permet de lancer un calcul de configurations stable ; un sous menu (2D ou 3D) permet de contrôler sa lecture après le calcul. Une fonction "tooltip" délivre aussi l'information sur un atome sélectionné (identification, propriétés).

Le deuxième prototype a été réalisé, l'évaluation n'a pas été faite. Notre système reste limité par une réalisation incomplète du couplage avec un logiciel de simulation ou de rendues similaires aux outils sandards de bureaux. Un grand nombre de pistes de recherches nous semble intéressante pour l'application de la RA en chimie : la définition précise de l'apport de la RA dans ce domaine, la place du tangible (une molécule = un objet réel ?) ou encore des méthodes de modélisation moléculaire telles que proposées dans *AR Chemistry*.

4. Application : architecture

4.1. Contexte

L'architecture a depuis quelques années intégrée les possibilités de l'informatique tant sur le plan du design (logiciel de dessin, modeleur 3D), que sur le plan de la réalisation (dessin industriel, prototypage virtuel). Les outils 3D actuels offrent les moyens de réaliser des maquettes virtuelles, d'étudier leur forme et apparence, de facilement changer de point de vue, de «naviguer» dans un modèle virtuel, de modifier l'éclairage ambiant ou de faire varier les conditions climatiques (figure 10.5).



(a) (b)

Fig. 10.5: Architecture : (a) logiciel de modélisation [Viz] et (b) rendu virtuel pour l'architecture intérieur [Gra01].

Cependant, il semble que la conservation des outils réels soit toujours d'actualité : le dessin, le plan technique offrent des outils connus, flexibles, et performants. La maquette réelle, comme introduite chapitre 8, offre les avantages d'une forte representation spatiale (compréhension cognitive aisée), un aspect collaboratif intéressant, et une modélisation et conception intuitive pour les designers (malgré le coût et le temps).

L'utilisation de notre environnement semblerait fournir une solution hybride appropriée : mélange de maquette réelle et virtuelle, modification interactive de maquette réelle, conservation des outils traditionnels, interaction 3D avec du contenu virtuel (animation, ajout d'éléments), etc. Son usage trouverait intérêt dans le cadre de réunions entre architectes, de présentations à des clients, de démonstrations grand public, etc. La couverture du domaine pourrait aller de la conception d'un bâtiment, d'un site, jusqu'à la planification urbaine.

4.2. État de l'art

De nombreux travaux se sont intéressés à l'architecture dans le cadre du rendu ([Gra01]), de la RV ([CJJ99] [SES99]) et plus récemment de la RA ; principalement dans le cadre de la visualisation, la modélisation ou de la manipulation de contenu virtuel architecturale.

Dans un cadre de visualisation, Webster et al. [WFM+96] a proposé la possibilité de voir à «l'intérieur des murs» à l'aide d'un dispositif optique de RA (repris par [Tri00]). En extérieur, Thomas [TPG99] propose l'augmentation des propriétés d'un bâtiment, quand à Reitmayr et Schmalstieg [RS04] ils proposent un système collaboratif extérieur pour annoter des bâtiments architecturaux. Enfin comme présenté chapitre 8 un intérêt va vers la modification de l'apparence de contenu réel architectural telle qu'avec les travaux de Loscos [LDR00] ou de Raskar [RWC99].

Pour la modélisation, Pietrasky [PT03] propose de digitaliser des bâtiments réels à l'aide d'une approche par contraintes CSG. Dernièrement, le prototype *sketchhand+* [Sei03] propose la modélisation de maquette virtuelle à partir d'une tablette digitale et la définition d'un profil et d'un chemin d'*extrusion*.

Dans le cadre d'une interaction sur table, Klinker [KSR99] propose des démonstrations pour la manipulation de maquettes virtuelles sur table ou l'intégration avec des maquettes réelles (ou encore l'intégration d'objets virtuels dans des séquences vidéos). Le projet *BUILD-IT* (déjà présenté Chapitre 2) cible leur environnement sur le cadre de manipulation de planification urbaine mais aussi architecturale. Le projet *Urp* [UI99] propose l'utilisation d'éléments tangibles couplés avec une projection 2D pour la simulation d'éclairage en architecture. En extension, Ishii [IUC+02] introduit alors le concept de maquette mixte couplant carte réelle, projection virtuel 2D et charpente d'un bâtiment 3D. Le projet *ARTHUR* [Rom01] s'intéresse à l'interaction collaboratif dédié à ce domaine (les premiers résultats se limitent à une interaction tangible avec un seul modèle virtuel).

Personnellement, nous nous concentrons sur un cadre d'interaction sur table donc principalement dans un contexte de manipulation. Nous nous placerons alors dans la continuité des travaux de *ARTHUR*, en étendant les concepts de Ishii (et Klinker) à une visualisation 3D.

4.3. Approche

Une collaboration a été réalisée avec le laboratoire de recherche en architecture de Lyon, ARIA. Un contact a été établi avec Xavier Marsault qui nous a amené à la proposition d'une évaluation avec des d'étudiants de dernière année d'école d'architecture de Lyon. De notre point de vue cette cible est fortement attrayante car elle constitue le panel possible d'utilisateurs de ce type d'environnement, qui pourrait voir le jour dans les années à venir.

En complément, un cabinet d'architecture de Mexico a pris contact avec nous, intéressé par ce type d'outils dans le cadre de démonstration et présentation à leur clients. Leur but est l'utilisation d'un espace de RA synchronisé avec le logiciel *3DS Viz*© [Viz]. Cette interoperabilité permettrait la conservation et l'édition dynamique du contenu 3D à partir d'une plate-forme connue. Suite à diverses discussions, nous avons identifier leurs besoins dans le cadre d'architecture en RA.

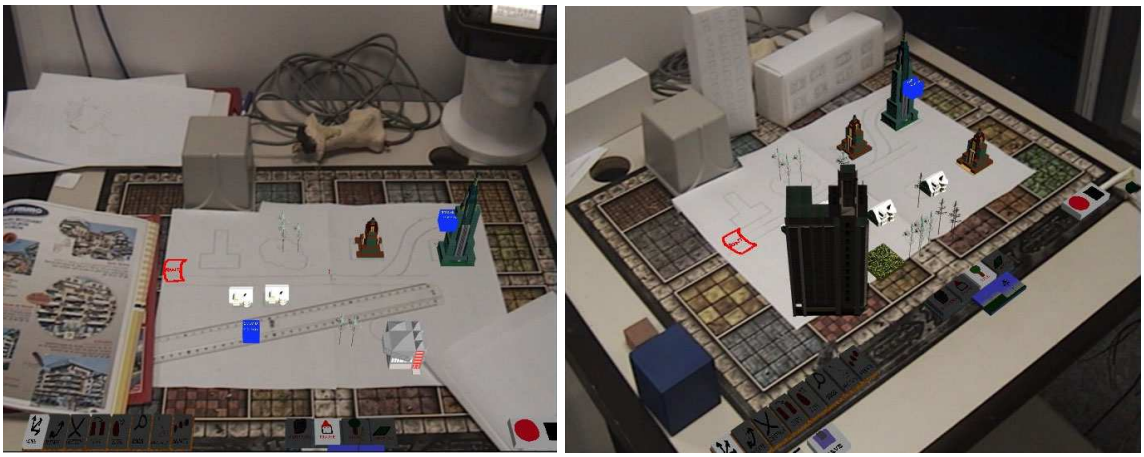
Nous avons choisi une présentation de type générale pour démontrer l'intérêt de notre environnement dans différentes situations liées à l'architecture. A la suite de ces différents éléments et diverses discussions complémentaires avec des étudiants en architecture, nous avons retenu les besoins suivants pour ce domaine :
– visualisation de contenu 3D dans l'espace réel ;
– navigation libre autour de maquette virtuelle 3D ;
– interaction, positionnement avec des objets virtuel 3D ;
– modification de contenu 3D virtuel (forme, géométrie) ;
– modélisation de contenu 3D virtuel (création de bâtiment) ;

– ajout, suppression de contenu 3D virtuel ;
– mixage de maquette 3D virtuelle/réelle ;
– support collaboratif d'interaction 3D sur maquette virtuel (désignation, annotations) ;
– modification des conditions extérieures (éclairage, climat) ;
– interopérabilité avec les applications standards d'architecture (*3DS Viz*©).

Nous avons donc mis au point une suite de prototypes pour la manipulation de contenu relié à cette thématique.

4.4. Prototypes et résultats

Un prototype préliminaire a été réalisé servant de cadre démonstratif sur la première version de MARE (figure 10.6). L'application propose la modification d'une maquette de planification urbaine reposant sur un plan réel. L'utilisateur peut à l'aide d'un menu ajouter des éléments virtuels (maison, tour, arbres), consulter et ajouter des annotations sonores et modifier le plan réel (par dessin ou adjonction d'éléments réels). Il peut en complément sauvegarder l'état de la scène pour la modifier à posteriori (prototype dérivé de l'outil MAREARSceneEditor).



(a) maquette mixte avec du contenu réel (plan, documentation, outils) et virtuel (modèle, menus)

(b) autre vue après l'ajout de différents éléments. nous noterons que les boutons sur la droite permettent d'enregistrer des annotations virtuels représentées sous forme de cube bleu dans la scène

Fig. 10.6: Illustrations du prototype d'architecture.

Les tests effectués par plusieurs personnes démontrent un intérêt confirmé pour ce genre d'application, une facilité d'utilisation des menus, des techniques de placement et de déplacement d'objets. Le problème d'alignement plutôt gênant, les utilisateurs considérant cette difficulté comme un problème à résoudre (nous noterons que les tests ont été effectués avec le premier prototype matériel, voir annexe A). Les annotation sonores ont montrés un grand intérêt même si des problèmes de synchronisation persistaient lors des démonstrations.

A partir de ces premiers résultats, nous avons réalisé d'autres prototypes pour l'évaluation globale. Nous avons pour cela réalisé la suite de prototypes démontrant : le concept d'alignement 3D en RA, le concept de visualisation de bâtiment 3D, le concept de visualisation coopératif, le concept d'interaction pour tâche 2D en planification urbaine et le concept d'application de peinture (repris du chapitre 8).

L'évaluation globale s'est déroulée durant une après-midi, les étudiants étaient encadrés de trois professeurs. Deux groupes d'une dizaine de personnes ont testé le système après une présentation générale (figure 10.8). Une évaluation complète de toutes les fonctionnalités pour tous les étudiants semblait difficiles (problème technique). Pour ce, la stratégie

a été de faire tester au moins un prototype (une fonction) à chaque étudiant, les autres pouvant visualiser sur des écrans de contrôle la vision et les actions du sujet. Chaque étape comprenait une explication générale, une manipulation guidée, puis une manipulation libre laissant la place ouverte aux questions (par les sujets ou les observateurs). Un questionnaire a été distribué en fin de séance aux deux groupes (annexe B).

(a) (b)

Fig. 10.7: Evaluation avec le groupe d'étudiants en architecture : (a) visualisation coopérative (b) interaction 3D.

Notre analyse repose sur 22 questionnaires retournés. Ils correspondent tous au étudiants de 5ème année. Une grande partie des sujets utilisent régulièrement des outils de modélisation 3D (17 sujets habitués, 3 initiés), ou des outils d'architecture (12 sujets utilisateurs), très peu sachant programmer (5 réponses positives avec 3 utilisateurs de VRML).

Fig. 10.8: Evaluation avec le groupe d'étudiants en architecture : peinture sur maquette réelle.

Un grand nombre des sujets ont trouvé le concept très intéressant (20/22) mais l'efficacité reste non démontré (2 réponses «très efficace», 9 réponses «efficace», 9 réponses «mitigé»). Il trouve en général ce concept très amusant (21/22). Ils ont le plus apprécier la possibilité de voir les objets dans le monde réel, de déplacer librement la tête, d'utiliser des objets réels en même temps que les objets virtuels (95 % de réponses «très bien» et «bien», 5 % de réponses «moyen»). Un grand nombre ont apprécié de pouvoir interagir naturellement avec les objets virtuels, de manipulation dans un cadre de réunion (moins de 15% de réponses moyennes). Le concept de peinture montre aussi un engouement pour cet approche (moins de 30% de réponses «moyen»).

Le casque a été perçu comme moyennement confortable (6 réponses «assez confortable», 11 «moyen», 4 «peu confortable»), son plus gros reproche reste la perte de liberté de mouvement (19 oui, 3 non). A la question «Si le casque était remplacé par de simples lunettes, avec une meilleur résolution vous jugeriez le fait d'avoir ces lunettes ?» ils ont répondus «confortable» (12) et «assez confortable» (9).

Ils considèrent la RA fortement applicable à leur domaines, même si 15% ne sont pas convaincus par les applications proposées. Concernant les avantages du systèmes, ils ont principalement choisi l'aspect de modification et d'expérimentation temps réel (11 réponses) de présentation (6 réponses) ou l'immersion (4 réponses). Les limitations principales se situant au niveau de la précision, de l'échelle et du matériel utilisé.

Les fonctionnalités proposées adhérent fortement avec la listes des besoins que nous avons identifiés. Une personne a proposé la visualisation en RA sur un site réel de maquette virtuel réalisé. Un autre a proposé l'adjonction des outils de mesures régulièrement utilisés en architecture. L'avenir du système est principalement perçu dans un cadre collaboratif (7 réponses). Cinq réponses lui donne un avenir indécis dû à la limitation de la taille des maquettes utilisables, les problèmes de fils, de lenteur du média par rapport au média traditionnel et des contraintes matérielles. De leur point de vue la plate-forme idéal est la 3D (10 réponses), les outils traditionnels (3) ou les deux (4 réponses).

Il semble évident qu'une étude plus concise nécessite d'être menée à partir d'un véritable scénario. De plus, il est nécessaire de réaliser une analyse plus détaillée des critères techniques, ergonomiques et applicatifs. Dans cette suite, il serrait intéressant d'étendre ce travail avec le laboratoire d'architecture ARIA et de réaliser un véritable projet avec un partenaire industriel.

Dans un cadre architectural il nous semble que la modélisation virtuelle est un sujet qui mériterait aussi d'être abordé ; Des approches par modélisation tangible ([SSWF00]) constitue une voie intéressante à explorer.

5. Conclusion

Ce chapitre présente une analyse globale du travail effectué et les résultats sur quelques domaines applicatifs choisis. Les premier résultats montrent un fort intérêt pour ce type d'application même si de nombreux problèmes restent persistants. Ils nous semblent que ces premiers travaux trouveront intérêt dans une continuité

Au niveau applicatif, un autre domaine d'étude serait la conception industrielle et le design ou prototypage virtuel collaboratif, tel que l'étude d'un moteur virtuel entre ingénieurs (figure 10.9). En complémentarité de la RV, ils offriraient un bon support d'échanges communicatifs collaboratifs.

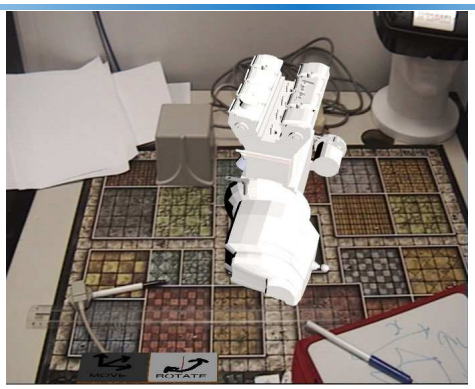


Fig. 10.9: Exemple de scénario de prototypage virtuel d'un moteur

Ils nous semblent que l'absence d'un guide de conception doit être défini, l'ensemble des résultats permettant déjà d'avancer quelques principes généraux. On les abord brièvement dans la conclusion.

Chapitre 11

Conclusion

La conclusion est l'endroit à partir duquel vous en aurez assez de penser.
—Arthur Block

Au cours de cette thèse, nous sommes partis du constat que les solutions actuelles pour interagir de façon collaborative sur des tâches 3D étaient non satisfaisantes. Nous nous sommes intéressés à l'utilisation de la réalité augmentée dans ce cadre et plus particulièrement au contexte colocalisé synchrone sur table. On résume ici les contributions, réflexions et perspectives de ces travaux.

1. Contributions

La contribution principale de cette thèse est **la démonstration de faisabilité** de notre approche : *utiliser la réalité augmentée dans un cadre de réunion sur table pour des tâches applicatives 3D*. Nous avons montré l'intérêt apporté par cette solution par rapport aux configuration existantes. Nous avons validé ces résultats par de nombreux prototypes et un intérêt certain des utilisateurs durant de nombreuses présentations et évaluations (majoritairement informelles).

Nous avons introduit une **nouvelle architecture** orientée vers des applications collaboratives sur table basée sur une approche hétérogène offrant un environnement générique. Nous avons en complémentarité proposé une **architecture logicielle** modulaire et extensible. Elle définit une plate-forme pour la conception de protocoles expérimentaux, la production de prototypes et d'applications dans un cadre de recherche de RA sur table.

Nous avons présenté un **canevas de calibrage**, dédié à notre environnement, offrant une solution simple et intuitive pour le calibrage statique. L'approche retenu reposait sur le concept du *plug'n calib and play* avec des tâche utilisateurs rapides et efficaces. Les résultats obtenus sont satisfaisants. Nos expérimentations sur la correction de l'erreur dynamique montrent les difficultés de mise en œuvre algorithmes de prédiction.

Nous avons proposé un **modèle d'interaction** pour de la RA collaborative sur table. A partir de la description de nouvelles interfaces, nous avons présenté une approche par décomposition en classes de tâches et des méthodes adaptées à chaque cas. Les premiers tests et évaluations informelles ont montré l'intérêt et l'utilisabilité de cette approche reposant sur un usage fort des propriétés de la table. Basée sur une optique d'interface hybride, nous avons introduit une interface de haut niveau dédiée à notre cadre d'interaction sur table. Nous avons proposé un canevas pour la gestion de documents virtuels à partir d'un formalisme d'espace/surface et l'importance de la notion de couplage. Un prototype «preuve-du-concept» montre la faisabilité de cette approche.

Nous avons proposé une nouvelle modélisation du concept de **modification du réel** à travers l'approche *Interactive Mediated Reality*. Un prototype de modification de l'apparence d'objets réel avec la notion de «studio virtuel» a été réalisé et évalué. Un deuxième prototype a présenté des techniques pour l'ajout d'objet réels de façon dynamique avec deux approches simples et intuitives.

Nous avons aussi introduit l'utilisation d'**heuristiques pour la validation** des critères collaboratifs dans un cadre de RA 3D (introduit dans des travaux en TCAO). A partir de ces éléments nous avons présenté les différentes méthodes mises en jeu pour les besoins collaboratifs.

Cette thèse contribue d'autre part à la recherche en RA par *une présentation exhaustive des divers travaux de RA réalisés à ce jour* et condensés dans la liste de références présente à la fin de ce document, mais aussi introduit tout au long des chapitres.

Par rapport aux solutions existantes notre solution innove par :
- l'utilisation du réel (contenu, outils et méthodes de travail) et le principe de **fusion forte** du monde réel et du monde virtuel, non limité à une simple intégration d'objets virtuels dans le monde réel ;
- une approche **«dédiée table»** qui se base fortement sur ces possibilités de ce support dans le même ordre que les travaux d'informatique évanescente ou les *SDG* ;
- l'usage fort des concepts d'**hétérogénéité** et de **générécité**. Nous souhaitions aussi introduire la nécessité de fournir une plate-forme de RA/RV/informatique évanescente introduisant une union de ces aspects.

2. Perspectives

En premier lieu, il nous semble qu'un grand nombre des concepts proposés dans cette thèse mérite de nombreuses validations complémentaires que les trois ans impartis ne nous ont pas permis d'effectuer. Il nous paraît indispensable de réaliser de plus amples évaluations quantitatives formelles ou des évaluations basés sur des tâches applicatives. Notre but initial étant orienté vers une approche centrée utilisateur, une analyse plus détaillée des résultats obtenus permettront d'en tirer des principes de conception dédiés aux environnements 3D collaboratifs sur table.

D'un point de vue de l'architecture, des études sur d'autres configurations spatiales telles que l'usage de tables avec d'autres caractéristiques (plus large, ronde, composable) doivent être menées. Elles permettront d'isoler les critères spécifiques de notre environnement et les propriétés génériques de ce type d'environnement.

D'un point de vue de la visualisation, des casques de meilleures résolutions et une projection sur table apporterait un système de meilleur qualité. Des travaux sur le suivi par vision en temps réel pourrait compléter notre canevas de suivi : une utilisation des travaux de vision en 2D telles que Berard [Ber03], Crowley [CC95] est envisageable. La généralisation de la notion de *cluster* hétérogènes et adaptatifs (de suivi, de caméras, de visualisation, etc.) permettrait d'avoir un système fortement générique.

Pour le calibrage, des techniques simples et efficaces doivent être réalisées, assistant l'utilisateur sans le remplacer (semi-automatique). Mais plus important concerne l'erreur dynamique. Il nous semble qu'une étude des mouvements corporels durant diverses tâches autour de la table permettrai la mise en œuvre de nouveaux modèles adaptés aux algorithmes de prédiction (tels que ceux initialement définis dans Shaw [SL92]).

D'autres interfaces doivent être expérimentées telles que les *tabletPC*, *SpaceBall*, les *phidgets* [GR01] ou les nouveaux systèmes haptiques sur table (telle que Rosenfeld [RZSP04]). De même une expérimentation quantitative plus formelle doit être conduite sur les aspects d'interaction 2D+1D vs 3D. Une étude sur des méthodes d'interaction 3D avec des espaces très hétérogènes (communication/interaction, relatif/absolu, autonome/complémentaire) constitue une prochaine étape de nos travaux sur les interfaces hybrides.

Des études plus précises sur l'aspect collaboratif sont aussi à compléter, limités dans notre cadre à un aspect statique (utilisateurs en position fixe) à courte durée en petit groupe. Des études sur d'autres cadres de collaboration doivent être réalisées : grande réunion, présentation (cadre mural), réunion spontanée ou collaboration mobile. La notion d'espace privé et public constitue également un cadre très intéressant à developper (forme, dimension, représentation).

L'aspect du positionnement spatial nous paraît intéressant : l'impact de l'orientation et la

comparaison de la collaboration face-à-face vs côte-à-côte méritent d'être observées. Des problèmes d'occultation (par des objets virtuels entre utilisateurs) et de point de vue très différents peuvent conduire à une étude de mise en valeur de certains éléments. On pourrait par exemple introduire la sélection «à travers les objets» d'éléments prépondérants ou de vue sémantique (dans le même ordre d'idée que Furmanski et al. [FAD02]).

Il nous semble aussi qu'une étude plus approfondie des besoins en mixage visuel réel/virtuel et de gestion des ombres doit être ménée. Des méthodes d'estimation rapide telles que Harmann et al. [WJH+03] constitue un bon point de départ pour ajouter l'éclairage direct temps réel en RA.

Enfin il nous semble qu'une fusion entre RA/RV et informatique évanescente doit être menée. La diversité des besoins tend aujourd'hui à se réduire par la convergence vers des problématique, un cadre de validation et des moyens techniques similaires (e.g. usage de la vidéo). Des collaborations avec des équipes (locales) de vision par support contextuel (PRIMA), d'IHM (IIHM) ou de RV (I3D) sont à mettre en œuvre à partir de cette plateforme générique.

3. Bilan et réflexion

Lors de cette thèse, une question a été souvent posée : Qu'est ce qui doit être réel ou virtuel ? La RA offrant un mixage des deux mondes, la part relative de chacun doit être toujours une question posée à partir de critères techniques, ergonomiques et conceptuels. De notre point de vue le réel offre un grand nombre de possibilités qu'il est actuellement impossible de reproduire dans le virtuel. Les métaphores d'écriture sont par exemple un des sujets fortement étudié en IHM. Les derniers périphériques telle qu'*Anoto*© [Ano] choisissent de conserver les métaphores réelles et d'adjoindre les possibilités du virtuel.

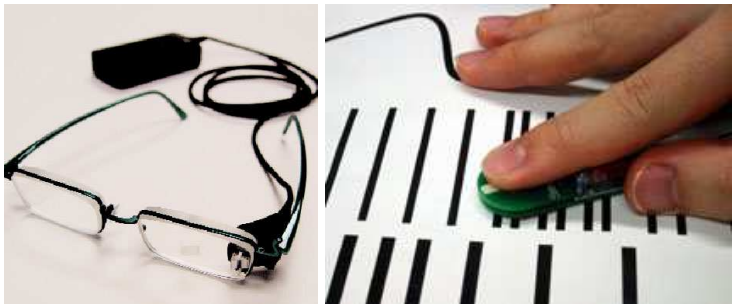
La RV a soulevé d'autres questions par rapport à l'importance du réel : le contrôle d'application doit-il être réel (menus 3D) ? L'immersion totale est-elle la solution ? l'immersion extrême a aujourd'hui trouvé un nouvel aspect par l'utilisation d'éléments tangibles (ou PDA) ou d'interfaces semi-immersives montrant aussi un retour au réel. De notre point de vue des analogies et l'observation des métaphores réelles peuvent apporter des pistes intéréssantes. L'usage du virtuel pourrait alors être utilisé en répondant à cette question : «Qu'est ce qui est nouveau, impossible en réel et utilisable ?».

L'autre point qui nous semble intéressant est l'aspect multidisciplinaire proposé par la RA. L'adaptation, l'unification et la raffinement de solutions existantes est une condition initiale à suivre. En effet, un grand nombre de travaux d'IHM et de RV est fortement disjoint, le cadre de chacun apporte pourtant des éléments complémentaires. Des concepts 2D peuvent simplement s'appliquer en 3D et ceci d'autant plus en RA.

Enfin, cette thèse et un grand nombre de travaux de RA ne peuvent s'exprimer aujourd'hui que par la suppression des contraintes technologiques. Des systèmes de visualisation, de suivi et d'interaction performants permettront de démontrer la validité et l'utilisabilité de la RA. On espère que l'accélération des prototypes industriels actuels apporteront des solutions (figure 11.1). Le suivi par vision peut aussi apporter une grande stabilité et robustesse de suivi pour la RA favorisant l'expérimentation de méthode d'interactions.

La définition d'interfaces 3D collaboratives futures restent alors ouvertes. Des questions telles que le choix de systèmes intrusifs (casques, capteurs 3D), faiblement intrusifs (lunettes, marqueurs au bout des doigts), ou non intrusifs (systéme projectif 3D, vision) peuvent se poser, la courbe de difficulté et de réalisation augmentant rapidement (figure 11.2). Est ce que l'utilisateur doit attendre l'interface idéale (non intrusif) ou s'adapter à une interface intermédiaire telle qu'on le fait avec la souris depuis de nombreuses années ?

Pour finir, nous considèrons que la RA offre une convergence ultime de toutes les interfaces



(a) (b)

Fig. 11.1: Prototypes actuels de lunettes de RA [Mic] et système tactile digitale [AMIM02].

(a) (b)

Fig. 11.2: Photomontage de futurs systèmes : peu intrusif vs non intrusif [Wora].

actuelles, intégrant de nombreux paramètres (2D, 3D, réel, virtuel, tangible, haptique, etc.). Elle offre un nouvel aspect d'interface, de visualisation et de support collaboratif ouvrant des perspectives de recherche dans la quête de l'environnement idéal.

> Le virtuel est ce qui nous aide à faire advenir ce que nous ne sommes pas encore.
>
> — Le Monde de l'Education. Philippe Queau.

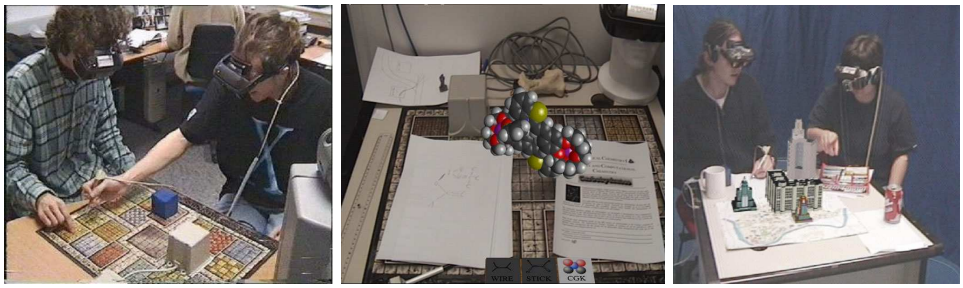
MARE : détails techniques de la réalisation

Cette annexe décrit plus en détails la réalisation et les analyses techniques de notre architecture. Nous présentons des résultats pouvant être utiles pour la reproduction de ce type de système. Nous introduisons en premier lieu un aperçu de l'architecture (section 1), puis l'architecture matérielle (section 2) et enfin l'architecture logicielle (section 3).

1. Historique

Le projet a initialement débuté au cours d'un stage de Magistère, en Octobre 1998, à partir d'un capteur magnétique et d'un casque de RA. Les premières configurations collaboratives ont véritablement vu le jour en 2001. Nous retenons quatre versions majeures réalisés (figure A.1) avec les caractéristiques suivantes :

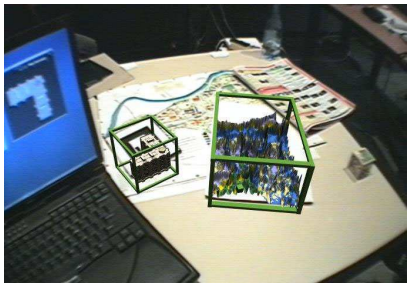
- Configuration 0 (1998-1999) : suivi par capteur magnétique pour un utilisateur. Expérimentation dans le cadre d'un jeu collaboratif ;
- Configuration 1 (2000-2001) : suivi par capteur magnétique et utilisation des deux casques semi-transparents, optique et opaque. Études des problèmes de calibrage et d'architecture. Développement de prototypes applicatifs ;
- Configuration 2 (2002) : suivi par capteur optique avec l'usage des deux casques semi-transparents optique et vidéo. Ce prototype a changé de salle en 2002 ce qui nous a conduit à améliorer la configuration dans le nouveau environnement. Études des problèmes d'interaction avec le virtuel et de modification du réel ;
- Configuration 3 (2003) : suivi avec de multiples capteurs et visualisation avec de multiples espaces et surfaces. Collaboration dans un cadre applicatif. Mise en œuvre de premières évaluations utilisateurs formelles.



(a) conf. 0 : configuration initiale

(b) conf. 1 : configuration magnétique

(c) conf. 2 : configuration optique



(d) conf. 3 : configuration finale

Fig. A.1: Les quatre configurations réalisées de MARE.

La suite de cet annexe s'intéressera principalement à la présentation de la dernière configuration.

2. Architecture matérielle

Cette section reprend le plan développé dans chapitre 4, section 2. Certaines images sont répliquées pour permettre une lecture indépendante de cet annexe du reste du document.

2.1. Mobilier : la table d'interaction

Étude préliminaire

Nous avons réalisé des observations informelles au sein de l'établissement de l'INRIA Rhône-Alpes sur les environnements de collaboration disponibles (figure A.2).

(a) amphithéâtre (b) grande salle de réunion (c) petite salle de réunion (d) grand bureau

(e) espace informel (f) cafétéria

Fig. A.2: Exemple d'environnement pour du travail collaboratif à l'INRIA Rhône-Alpes.

Ces observations nous ont aidés dans l'identification des critères et des relations des composant des environnements collaboratifs réels.

Mise en œuvre

La table choisie est visible sur les figures ci-après (figure A.3). Conçue initialement comme une table de travail personnel, elle nous a semblé adéquate pour convenir aux critères définis chapitre 4.

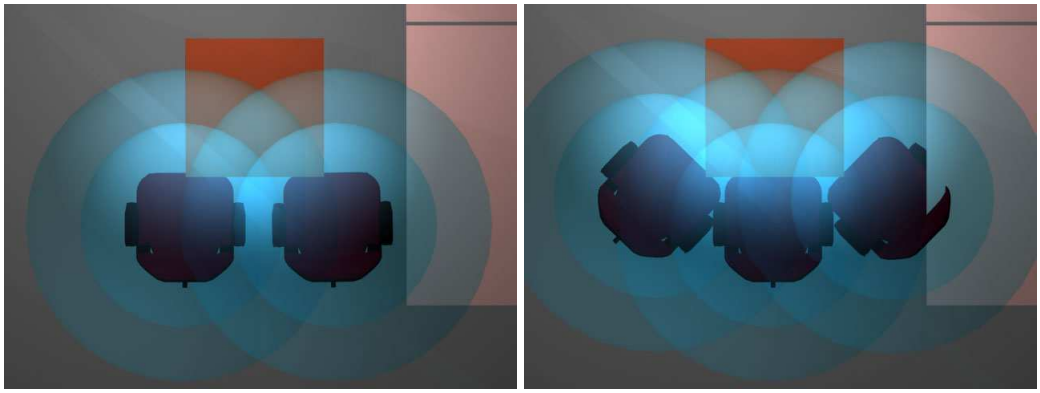
(a) (b) (c)

Fig. A.3: Table utilisée pour nos expériences : (a) configuration pour deux utilisateurs (b) configuration pour trois utilisateurs (c) configuration pour quatre utilisateurs.

La figure A.4 présente un scénario d'utilisation de la table avec des objets réels. Il nous donne une représentation réaliste des différents objets réels pouvant être présents dans une réunion : outils réels personnels (cahier de note, stylo), objets non fonctionnels, éléments réels 2D dans la zone centrale (papier, documentation, carte) et éléments réels 3D (maquette, artefact de maquette). Il fixe alors les contraintes pour le principe défini initialement dans cette thèse : conservation de l'environnement réel.

Fig. A.4: Contraintes d'interaction avec le réel : conservation des documents papiers, de la zone privée et des éléments réels non nécessaires à la tâche.

Évaluation



(a) (b)

Fig. A.5: Simulation des espaces ergonomiques avec notre configuration pour deux et trois utilisateurs. Le cercle interne représente l'espace personnel, le cercle externe l'espace personnel distant. On note le conflit d'espace intime pour trois utilisateurs.

Besoins ergonomiques : nous avons réalisé une simulation pour l'évaluation des espaces ergonomiques disponibles avec notre configuration. Elle a été réalisé pour deux et trois utilisateurs. Nous avons reproduit à partir d'une modélisation de l'environnement[1] un positionnement des espaces de communication sous forme de sphères géométriques. Nous avons alors étudié la configuration optimale (figure A.5). Elle montre que le cadre collaboratif est :

- satisfaisant pour deux utilisateurs côte à côte avec conservation d'un espace personnel suffisant ;
- mauvais pour trois utilisateurs côte à côte avec superposition des espaces intimes et un espace personnel réduit ;
- limite pour quatre utilisateurs (trois face à un) avec conservation d'un espace personnel réduit.

Nous notons de plus que la zone partagée se limitera au placement d'éléments de taille maximum A2 réduisant l'utilisation de carte de type A1 ou A0.

[1]basée sur les mesures et la modélisation réalisée sous *Discreet 3DS Max* par Laurence Boissieux

Limitations techniques : la forme physique de la table rend difficile le placement en configuration deux par deux face à face. Nous privilégierons donc une configuration côte à côte ou à 45 degrés avec deux utilisateurs.

Interface et suivi : les différentes utilisations de la table au cours de cette thèse ont permis de valider le support de suivi par vision, de suivi magnétique et la facilité d'interaction digitale (surface lisse, plane). La table répond donc aux contraintes définies.

2.2. Visualisation : casques, écrans et projection vidéo

Mise en œuvre

Pour les casques, la mise en œuvre a été réalisée à partir de deux casques Virtual I-O I-glasses. Le premier est un casque optique semi-transparent stéréoscopique, avec une résolution de 320x200 pixels, un angle de vue 31 degrés, une convergence fixée, 100% de superposition stéréo et une distance inter-pupillaire fixée à 62 mm. Le deuxième prototype est une configuration vidéo semi-transparente réalisée au laboratoire, comprenant une caméra Eon PAL placée sur le deuxième casque (écran opaque) au centre de la visière. La caméra a une résolution de 720x576 pixels, un écran 1/4 pouces, une focale de 6 mm. Un système adhésif a été choisi pour permettre un déplacement facile de la caméra et son réajustement de position. La connectique a été réalisée pour transporter l'image depuis la caméra vers la station d'accueil. Le positionnement des câbles et l'attache ont été ajustés pour favoriser les critères ergonomiques (liberté de mouvement et légèreté).

La figure A.6 et figure A.7 montrent les prototypes ainsi que les vues visibles à travers ces casques. L'image visible du casque semi-transparent optique a été filmée à l'aide d'une caméra positionnée à la position de l'œil. L'image visible du casque semi-transparent vidéo correspond à l'image finale envoyée dans ce casque.

(a) (b)

Fig. A.6: Le prototype de casque optique (a) et une vue de l'image perceptible à travers (b).

Le prototype de l'ordinateur portable vidéo semi-transparent a été réalisé à l'aide d'un ordinateur DELL Inspiron, Pentium III 800 Mhz, équipé d'un bus IEEE FireWire. La caméra choisie est une caméra Unibrain Fire-i, délivrant un signal compressé de type 4 :1 :1 en 640x480 à 30 images/secondes. Un support spécifique adhésif a été réalisé pour le positionnement de la caméra sur l'écran en conservant une certaine flexibilité (figure A.8).

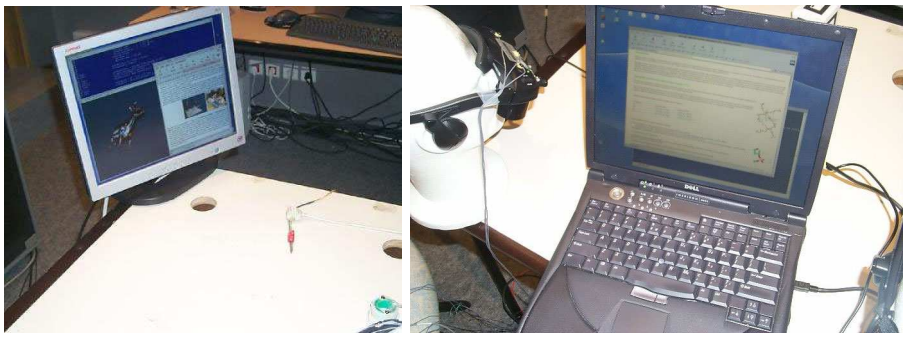
Pour la présentation de groupe, nous pouvons utiliser le portable introduit précédemment mais aussi un écran plan pouvant se positionner sur la table (figure A.9). C'est un écran orientable Compaq, 17 pouces, 1280x1024 pixels. Le placement retenu est sur le côté pour des limitations causés par les capteurs de mouvements utilisés, l'écran risquant de générer des occultations.

(a) (b)

Fig. A.7: Le prototype de casque vidéo (a) et une vue de l'image perceptible à travers (b).

(a) (b)

Fig. A.8: Vue personnelle complémentaire : (a) vue rapprochée du branchement du système vidéo sur portable et (b) vue de l'image perceptible à travers.



(a) (b)

Fig. A.9: Vue en groupe à courte distance : (a) écran plat et (b) écran de portable.

Pour la présentation publique, nous avons choisi une technologie par projection, avec un projecteur Nec 4760, 1000 lumens, 1280x1024 pixels. Il est positionné sur un support rigide derrière les utilisateurs pour une projection murale dans la direction de leur regard (figure A.10). La hauteur a été calculée pour éviter de masquer le cône de projection du système (par les utilisateurs). Elle a aussi été calculée a une hauteur minimisant les contraintes de mouvement de rotation de la tête (pour un utilisateur de taille "moyenne"). Nous avons positionné un écran en carton blanc peu réfléchissant au mur pour l'affichage.

(a) (b)

Fig. A.10: Vue publique complémentaire : (a) projecteur et (b) projection murale.

Résultats

Facteurs système : nous avons évalué les critères intrinsèques des dispositifs à partir de l'affichage d'une mire. Pour les casques, nous observons la trop faible plage de couleurs disponibles créant une énorme perte des couleurs rendues. Le casque optique souffre d'une mauvaise luminance et nécessite un éclairage externe suffisant pour restituer un bon contraste entre réel et virtuel. Un éclairage trop intense rend cependant l'image affichée complètement transparente. Pour le projecteur, nous observons une qualité suffisante sous éclairage naturel. Pour les écrans actifs, la qualité est très bonne avec aucune dégradation et une bonne luminance restituée. Pour les systèmes vidéos, une mire est placée devant chaque caméra à distance suffisante couvrant leur champ de vue. Nous observons une très forte distorsion pour la caméra placée sur le casque, une perte et une forte modification des couleurs et une restitution incomplète de l'image. La caméra sur le portable offre peu de distorsion et un bon rendu des couleurs en mode 4 :1 :1.

Nous avons testé la restitution d'une image avec du contenu 2D (texte, photo et illustration), 2D animé (animation, vidéo) ou 3D (modèle isolé, scène complexe) sur chacun des dispositifs. La lecture d'éléments textuels de petite taille est de mauvaise qualité sur la projection murale. Nous observons de plus une latence importante du casque optique.

Facteurs ergonomiques : de la même façon que précédemment nous avons simulé par une modélisation géométrique le champ de vue restitué par chacun des dispositifs avec une représentation basé sur des primitives de type cône ou pyramide tronqué. Nous observons alors leurs positions et la couvertures par rapport au modèle usuel humain. Comme prédit, le champ de vue des casques est très faible et de faible résolution[2]. Le casque optique semble plus intéressant car il permet une conservation de la vision périphérique. Comme observé le portable nécessite un recul important et prend une forte place dans l'espace de l'utilisateur. La «vision à travers» est très petite alors que nous souhaiterions une vision à travers plein écran. Pour la visualisation de documents 2D, une solution avec *tabletPC* semble intéressante mais elle perd l'aspect privé fourni par le portable (dû à son inclinaison dans le cadre d'une utilisation personnelle).

Pour les casques, la difficulté principale reste un bon placement initial nécessitant une assistance. Le côté intrusif induit par ce principe reste une question ouverte : au cours de nombreuses expérimentations peu d'utilisateurs ont critiqués l'approche par casque, notant principalement ses limitations technologiques.

2.3. Systèmes de suivi

[2] A distance standard de travail, il faudrait une résolution d'environ 4800x3400 pixels pour couvrir le champ visuel

Mise en œuvre

Nous avons choisi un capteur optique de type Northern Digital Optotrak 3020 basé sur une technologie infrarouge de suivi de leds avec fils (figure A.11). La résolution est de 0.01 mm, la précision 0.1 mm en x,y et 0.15 mm en z avec une fréquence de 5000 Hz maximum divisible par le nombre de leds utilisées (256 maximum). L'option temps réel garantit un suivi avec 24 leds simultanées à une fréquence d'au moins 30Hz. Le système permet à partir d'un fichier de définition (représentant les positions relatives de plusieurs leds dans un repère local) de fournir un suivi à 6 DDL d'un élément.

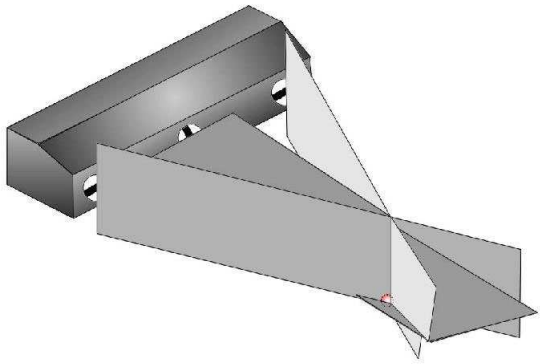


Fig. A.11: Fonctionnement du capteur Optotrak : triangulation à partir de trois caméras linéaires infrarouges.

Le positionnement du capteur a été étudié pour fournir un volume de suivi des utilisateurs et de l'espace au dessus de la table (les leds supportant un suivi avec un angle solide de 110 degrés). Nous avons choisi de positionner le capteur au mur, orienté vers le dessus de la table (figure A.12). L'inclinaison limite l'utilisation pour une collaboration face à face dans l'axe du capteur (les utilisateurs devant alors se mettre dans l'axe perpendiculaire au capteur).

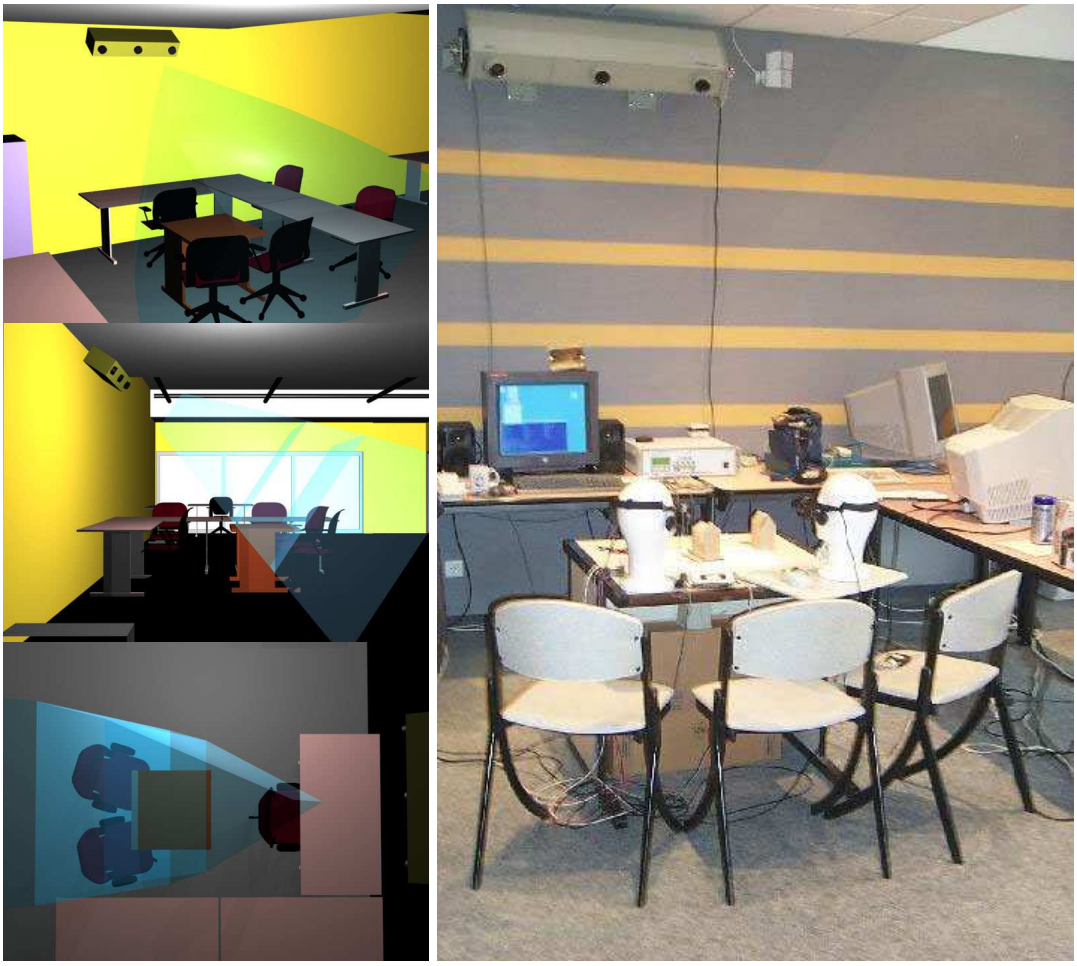
Nous avons choisi d'utiliser 3 leds pour le suivi du casque. En considérant un suivi pour 4 personnes, Ils restent ainsi 12 leds disponibles pour l'interaction.

Pour le suivi à 6 DDL robuste, nous avons choisi un capteur Ascension Technology Flocks of Birds délivrant une résolution de 0.76 mm et 0.1 degré (à 30 cm), une précision de 1.77 mm et 0.5 degrés (figure A.13). Il offre un champ de suivi sphérique de 90 cm de rayon à une fréquence de mesure de 100 Hz. Nous avons aussi étudié son positionnement (figure A.14) : il est placé sous la table sur un support rigide permettant une couverture hémisphérique au dessus de la table. Suite à des études réalisées dans Grasset [Gra98] le capteur est configuré pour fonctionner en mode continu avec des filtrages internes optimisés.

Pour le suivi externe, nous avons choisi une caméra SGI 02 (640x480, 30 Hz) couplée à la librairie ARToolkit [KB99] (figure A.15). Elle fournit un suivi par vision temps réel à partir de marqueurs 2D noir et blanc (6 DDL, avec une précision variable suivant la taille du marqueur). La fréquence du suivi est alors dépendante du nombre de marqueurs à reconnaître.

Résultats

Optotrak : Nous avons réalisé une évaluation des performances du système dans son contexte d'utilisation. Nous avons réalisé des mesures à différentes positions de l'espace de la table : la stabilité statique obtenue avec une led est de 0.006 mm. La même opération effectuée avec l'élément rigide PLATE nous donne 0.0052 mm. Nous avons ensuite déplacé l'élément manuellement sur une trajectoire rectiligne sur le support de la table : le bruit maximum engendré est de 2.24 mm pour la led et 2.24 mm pour l'élément rigide



(a) (b)

Fig. A.12: Étude du capteur optique : (a) simulation champ de vue et (b) positionnement.

(comprenant le bruit engendré par la main). Puis, nous avons réalisé une mesure de la fréquence obtenue avec 24 leds sous différentes configurations (avec ou sans rigide, leds statique ou en mouvement) : nous obtenons une fréquence de lecture de 155 hz et une fréquence de rafraîchissement en cohérence avec les valeurs théoriques (120 hz maximum avec 24 leds). La stabilité statique obtenue avec 24 leds est alors de 0.01 mm en X pour le PLATE. Pour finir nous avons réalisé un échantillonnage des bords du volume théorique de suivi présenté figure A.16 qui montre un volume suffisant autour de la table et une conformité à la valeur théorique estimée.

Nous avons ensuite évalué la robustesse du suivi par variation d'orientation. Nous l'avons d'abord mesurée sur l'objet rigide PLATE, par rotation sur les différents axes de l'élément, puis de façon aléatoire : nous remarquons peu de perte mais un volume trop limité (peu de rotation sur l'axe X). De part ces résultats, nous avons réalisé une évaluation sur le casque suivant les différentes configurations par mesure du déplacement de la tête dans la plage physique possible à cette position. Nous observons un suivi stable dans le volume possible (très peu de perte). La mesure de stabilité statique du casque sur la tête nous donne 0.2 mm dans une utilisation à 24 leds (0.024 mm si le casque est posé sur la table).

Nous notons que l'angle est très rasant en entrée de volume ce qui limite l'utilisation de la configuration quatre utilisateurs et l'augmentation des occultations. D' un point de vue ergonomique les leds sont légères, de petite taille et peu gênantes (même si elles restent intrusives). Le principal problème est le fait que les fils s'entremêlent facilement (avec des

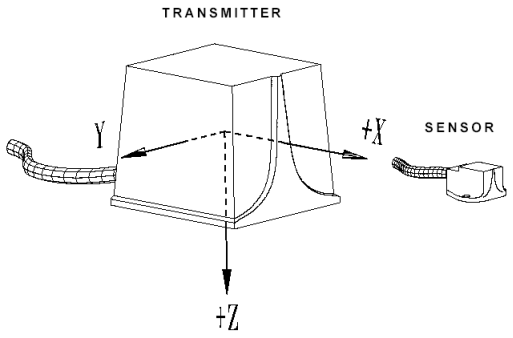


Fig. A.13: Composantes et repère du Flock of Birds : un émetteur (à gauche) associé à un récepteur magnétique (à droite) reliés à un boîtier de contrôle (non visible ici).

(a) (b)

Fig. A.14: Étude du capteur magnétique : (a) simulation du champ de vue et (b) positionnement.

(a) (b)

Fig. A.15: Caméra externe : (a) simulation du champ de vue et (b) positionnement.

connectiques peu robuste à un usage régulier).

Flock of Birds : Nous avons réalisé une mesure du fonctionnement du Flock of Birds dans notre environnement. L'étude initial déjà effectuée dans Grasset [Gra99a] a été renouvelée dans notre nouvel environnement. Basé sur le même protocole que précédemment, nous obtenons une stabilité statique de 0.05 mm lorsque l'élément est situé au centre de la

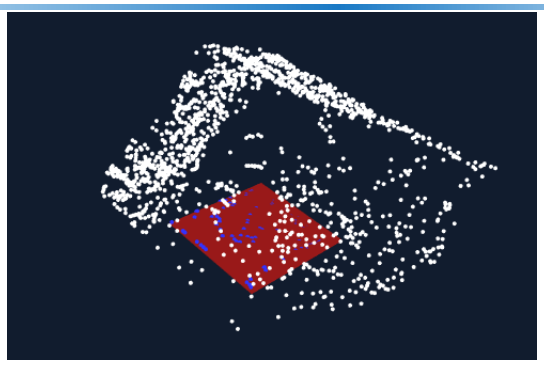


Fig. A.16: Échantillonnage du bord du volume de suivi de l'Optotrak autour de la table

table, de 0.12 mm sur sa surface et 0.5 mm en hauteur (56.5 mm). La fréquence de mesure obtenue est de 104 hz. Nous avons aussi fait une évaluation du volume de suivi confortant nos résultats théoriques. Pour évaluer la présence de distorsion du champ magnétique nous avons réalisés une mesure de précision comme précédemment à partir d'une grille réelle en positionnant l'élément à chacun de ces points, la grille déplacée verticalement à différentes hauteurs (0, 12.5, 16.5, 33, 45.5 cm) : aucune distorsion notable n'a été observée (voir figure A.17).

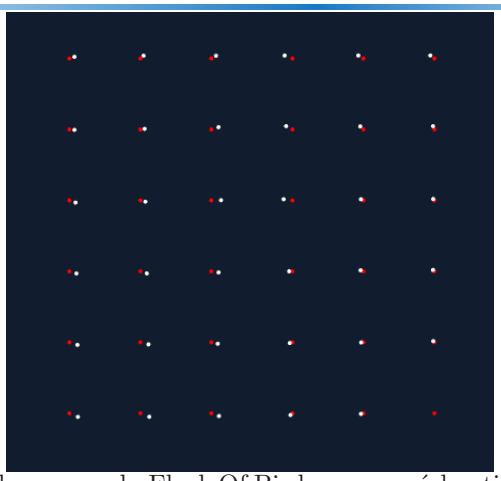
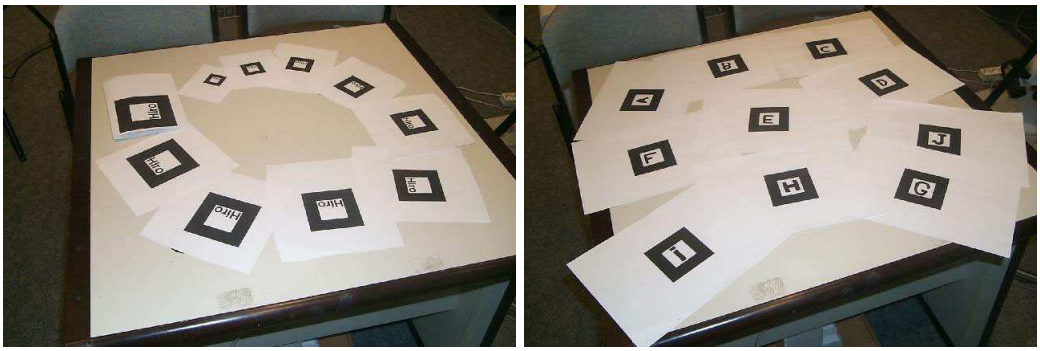


Fig. A.17: Representation de la mesure du Flock Of Birds pour un échantillonnage sur une grille régulière sur la table et à 12.5 cm en hauteur (en blanc les mesures et en rouge les valeurs idéales)



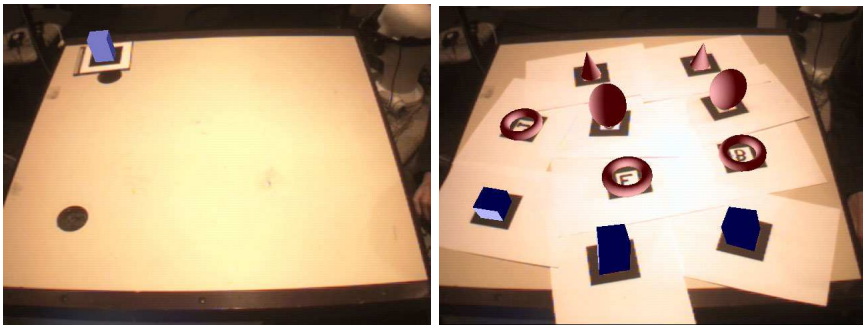
(a) (b)

Fig. A.18: Liste des patterns (a) de test pour le suivi par vision externe (b) utilisés pour le suivi.

ARTtoolkit : Nous avons réalisé une évaluation des performances de la librairie ARToolkit dans notre configuration. Nous avons tout d'abord défini la taille minimale des éléments

pouvant être suivi (figure A.18a). L'évaluation s'est faite par variation de la taille de l'élément et son positionnement à différents points clefs sur la table (angles et centre). En étudiant la robustesse du suivi, nous considérons que la taille raisonnable d'utilisation est 8 cm. Nous avons ensuite évalué la stabilité statique en positionnant les marqueurs à différentes positions de la table : Nous obtenons un écart-type de 0.86 à 2 mm pour X et une erreur de 1.2 mm à 10 mm (valeur croissante : le centre, angles devants et angles de derrière en cohérence avec l'information visuelle disponible). Pour finir nous avons évalués la fréquence obtenue avec le système avec un marqueur sur la table (9.1 Hz), un marqueur pour 10 connus par le système (1/10, 8.9 Hz), 5/10 (5.3 Hz) et 10/10 (3.4 Hz). Nous voyons que dix marqueurs semble une valeur maximale (300 ms de temps d'attente), couvrant aussi par ce fait un espace maximale sur la table (voir figure A.19). Nous notons que les bords de la table sont peu efficaces pour l'identification d'éléments mais supportent un suivi provenant d'une autre zone (on introduira donc les éléments au centre, ou en haut de sa zone personnelle).

Le choix des marqueurs a été réalisé en utilisant une symbolique iconographique et discriminante : les lettres de l'alphabet (avec l'utilisation dans certains cas des marqueurs standard). Nous noterons des conflits entre des marqueurs comme E et F (figure A.18b).



(a) (b)

Fig. A.19: Vue à travers la caméra (a) sur le bord (a) avec un grand nombre de patterns.

2.4. Interfaces

Interface standard

Au niveau des interfaces commerciales (figure A.20), nous avons choisi trois souris et claviers standard (3 boutons)[3] qui se positionne dans chaque espace personnel des utilisateurs. La tablette digitale choisi est une Wacom Intuos de format A4, offrant une résolution de 2540 dpi, une précision de +/- 0,25 mm, une fréquence de 200 pps et une détection de pression à 512 niveaux. Elle permet une interaction à deux mains avec un suivi du stylo de haute qualité (3 DDL position[4], 2 DDL orientation, pression et 3 boutons) et de la souris (2 DDL positions, 5 boutons).

Outil de pointage : le Mixed Pen

Mise en œuvre : différents prototypes ont été expérimentés au cours de cette thèse. Dans les différents cas nous utilisons un stylo bille comme interface avec le réel. Nous proposons alors deux versions étendu du stylo : naturel (permettant de fixer dynamiquement un capteur), ou modifié (avec un bouton apportant un élément de contrôle sur certaine tâches). Cette dernière version a été réalisée en coopération avec Laurence Boissieux (figure A.22).

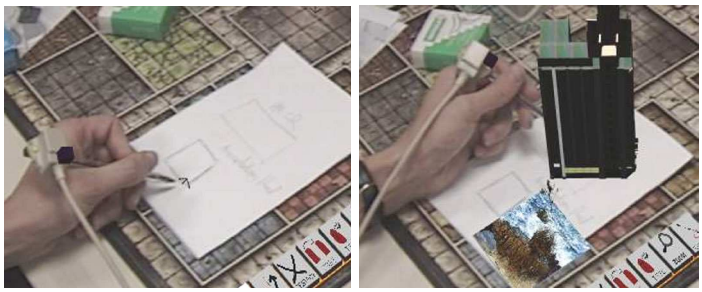
Nous décrivons brièvement les différentes approches étudiées pour le suivi ainsi que leurs limitations :

[3]nous pouvons ajouter l'interface du portable comprenant une entréé clavier et un *touchpad*

[4]Nous pouvons considérer en fait 3 DDL car la tablette fournit une détection magnétique jusqu'à 1 cm de hauteur.

(a) (b)

Fig. A.20: Configuration pour (a) une utilisation coopérative de souris et (b) une utilisation personnelle de la tablette.



(a) (b)

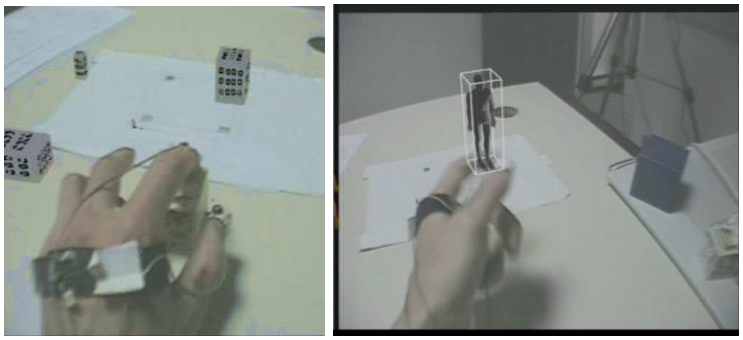
Fig. A.21: Le *Mixed Pen* : (a) utilisation en réel et (b) en virtuel.

Fig. A.22: Prototype du *Mixed Pen* avec un bouton.

- suivi par capteur magnétique. L'embout du stylo est associé rigidement à un récepteur magnétique. Le principal problème de ce type de système est le poids du stylo et la contrainte induite par le fil ;
- suivi par capteur optique. Un repère rigide constitué de plusieurs leds est associé à l'embout du stylo. Un suivi robuste à 6 DDL nécessite malheureusement un trop grand nombre de leds. Le faible poids des fils, contraint peu l'utilisateur, mais induit un certains nombre de torsions entre les fils, conduisant à une détérioration trop rapide du matériel ;
- suivi par vision. Un stylo est équipé d'un cube en carton associé à des marqueurs sur les différentes faces. L'avantage du sans fils induit par ce système souffre alors du faible angle de vue d'un fonctionnement mono-caméra. On est alors fortement limité une utilisation naturelle, contraint à un travail dans un volume toujours visible par l'œil de l'utilisateur (ce qui est problèmatique avec le faible angle de vue des casques) ;
- suivi par approche *Real Reality*. L'idée est de suivre les mouvements de la main, celle-ci associé au stylo. Ce type d'approche a montré de faible résultats principalement pour l'identification de la phase de sélection et l'obtention du point d'interaction (sans

contraindre l'utilisateur à prendre l'élément toujours dans la même position).

Outil d'interface gestuelle : le Lightweight AR Glove



(a) (b)

Fig. A.23: Le Lightweight AR Glove : (a) utilisation en réel et (b) en virtuel.

Mise en œuvre : comme précédemment nous avons étudiés plusieurs solutions et prototypes pour le *Lightweight AR Glove* (figure A.24) :

- suivi par capteur magnétique sur la main. Basé sur le même concept que Bruns [BB96] ou Veigl [VKL+02] nous fixons le capteur au dos de la main, considérant un point d'interaction à distance fixe de cet élément. Cette approche limite le type de mouvement réalisable, offre un seul point d'interaction et contraint ergonomiquement trop fortement l'utilisateur ;
- suivi par capteur magnétique sur le bout du doigt. Cette solution offre un meilleur contrôle du point d'interaction mais le capteur trop lourd limite la maniabilité et fatigue fortement l'utilisateur ;
- suivi par capteur optique fixé sur la main. A l'aide d'un adhésif spécial on fixe directement des leds sur la peau de l'utilisateur. Cette solution offre une bonne conservation des sensations tactiles avec les éléments réels mais les fils des leds sont gênants et la mise en place longue et avec perte (adhésif non réutilisable) ;
- suivi par vision sur le bout des doigts. Dans le même esprit que Dias [DSB+03] on fixe des marqueurs ARToolkit sur le bout des ongles à l'aide d'un fixant non persistant. Malheureusement les limitations de suivi contraignent trop fortement l'interaction, réduisant fortement l'intérêt de la main ;
- suivi par capteur optique sur un système attachable sur la main. C'est la solution choisie que nous présentons dans la suite.

(a) (b) (c)

Fig. A.24: Différents prototypes réalisés du *Lightweight AR Glove* : (a) capteur magnétique sur la main, (b) capteur optique fixé sur la main et (c) capteur sur une structure attachable à la main.

Le système choisi est constitué de trois parties enfilables à différentes positions : le pouce, l'index et la paume de la main. Un système par scratch permet de s'ajuster aux caractéris-

tiques morphologiques de l'individu (longueur des doigts, taille main et doigts, latéralité dominante). Sur chacun de ces éléments, nous avons choisi de positionner uniquement 3 leds. Cette solution restreint à un suivi à 3 DDL, mais permet de profiter pleinement d'une utilisation pour quatre personnes (ou deux utilisateurs pour une interaction à deux mains).

Le choix des points de placement des leds (pouce, index) se justifie par leurs plus fort potentiel d'interaction, doigts dominants dans de nombreuses tâches, les autres servant principalement de support. Si on observe les configurations d'interactions gestuelles sur des objets virtuels, un certain nombre peuvent être reproduite ici avec nos 3 points de suivi. Le positionnement des leds sur le support circulaire est défini pour maximiser le volume de suivi dans les postures usuels de la main en se basant sur le fort angle d'émission des leds (110 degrés).

Props et artefact physique : Tangible Augmented Reality

Mise en œuvre : Pour le deuxième type d'éléments nous nous sommes basés sur un suivi par vision à l'aide de la librairie ARToolkit qui est limité par un faible support des occultations (pas de masquage des éléments par la main de l'utilisateur) mais possède l'avantage d'être sans fils. Nous proposons en complément l'utilisation de positionnement de leds du système Optotrak pour les objets ou outils dans le cadre d'expérimentation offrant une forte précision.

Composants annexes

Comme illustré figure A.25, nous offrons en complément des interfaces «standard de RA» : nous définissons par exemple une version différente du concept du PIP [SG97a] avec l'utilisation d'une tablette réelle. Elle offre une interaction mixte : annotations réelles et manipulation d'éléments virtuels (voir les fonctionnalités du PIP pour plus d'information). Nous pourrons aussi utiliser un Joystick 3D (suivi par capteur magnétique) développé au cours dans nos précédents travaux [Gra98].

(a) (b)

Fig. A.25: Autres interfaces avec un suivi magnétique : (a) tablette mixte et (b) joystick 3D.

2.5. Micros et haut parleurs

Mise en œuvre

Nous avons choisi le moteur de rendu sonore 3D, OpenAL couplé à un système quadriphonique pour la restitution spatiale (configuration contrainte techniquement). Le système est distribué sur deux machines nécessitant une coordination entre les deux programmes de moteur de rendu sonore. Le système est défini de façon circulaire autour des utilisateurs et de la table (2 haut-parleurs devant et 2 derrière orientés vers le centre de la table). Pour la restitution d'ambiance sonore publique nous nous basons sur des librairies standards. Nous utiliserons la même configuration physique en privilégiant le système stéréophonique ayant de meilleures qualités. Pour la restitution d'éléments vocaux (dictation) nous utilisons le moteur de synthèse vocale *Festival Speech System* (moteur TTS, Text To Speech) couplé à *MBROLA* pour l'utilisation de voix françaises.

(a) (b)

Fig. A.26: Système quadraphonique : (a) vue devant et (b) derrière.

Pour les espaces sonores privés, nous utilisons 3 casques audio «bon marché» basés sur des oreillettes permettant la restitution d'un son très localisé. Notre système n'offre actuellement pas le support simultanées des espaces sonores publics et privés.

Pour l'enregistrement phonique on utilise 3 micro-cravates bon marché (figure A.27). Chaque utilisateur peut être équipé d'un de ces micros reliés aux machines de notre configuration (une machine par micro). Nous avons choisi d'expérimenter le logiciel *IBM ViaVoice 3.0* pour la reconnaissance vocale.

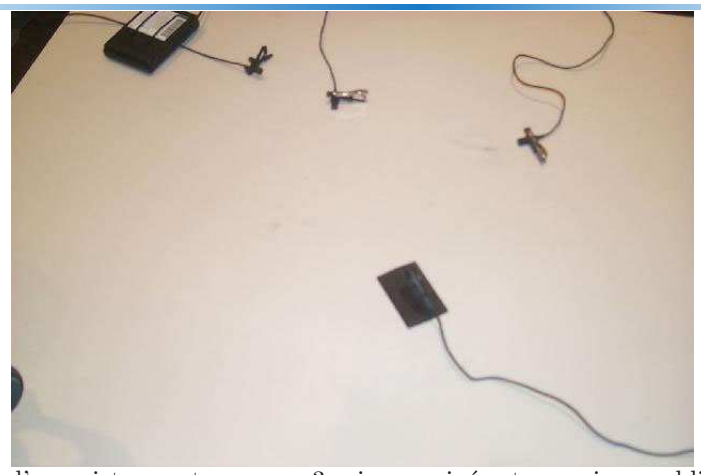


Fig. A.27: Ensemble d'enregistrement sonore : 3 micros privés et un micro public (3 éléments effectifs simultanément avec la configuration actuelle).

Résultats

Sortie sonore : Le système supporte aisément la diffusion de plusieurs sources sonores, de la musique et une restitution vocale. La synchronisation ad-hoc est suffisante pour une restitution spatiale entre les deux systèmes sonores, mais le temps de latence perceptible réduit son efficacité. Nous notons que le système quadriphonique est très limité, une approche avec un système Dolby Digital 5.1 ou un chaînage des haut parleurs USB serait plus efficace (sans problème de synchronisation). Une solution initiale serrait l'utilisant un protocole de type NTP. La restitution vocale sous *Festival Speech System* est de faible qualité : un enregistrement prédéfini de texte est alors préférable à sa restitution dynamique.

Entrée sonore : les essais réalisés à partir de librairie de reconnaissance vocale se montrent peu efficace avec le matériel standard même en mode public : l'environnement phonique est très bruité (par défaut dans l'espace de travail et dans un cadre de réunion). Les enregistrements sont de faible qualité mais suffisent pour des annotation sonores.

2.6. Support de calcul : architecture centralisé

Notre architecture actuelle repose sur l'utilisation d'une Onyx 3400 (8 processeurs, 8 Go de RAM, 2 cartes graphiques, IRIX 6.5), une O2 (1 processeur, 400 Mo de RAM, IRIX 6.5), et un portable Dell Inspiron 800 (PIII 800 Mhz, Linux). Ils sont réliés à partir du réséau de notre equipe, un réseau 100 Mbits.

3. Architecture Logicielle

Cette section présente de façon plus détaillée des modules de notre plate-forme logicielle.

Module `Périphérique`

`Périphérique` : ces composants définissent les sources et destinations (éléments terminaux en E/S). Nous distinguons les entrées et sorties :

- `entrées` : standard (clavier, souris), suivi (Optotrak, Flocks of Birds), tablette Wacom, SpaceMouse, vidéo, audio.
- `sorties` : affichage (écran de bureau, HMD, Reality Center), sonore (OpenAL). Dans le cadre de l'affichage nous gérons les différents modes stéréo (*Anaglyph, Line Interleaved, Field Interleaved, Frame Sequential, Dual Output*), le support multi-écrans (par Performer).

Ils respectent des noms de méthodes et une fonctionnement standardisé permettant une facilité de communication et d'extensions par de nouveaux périphériques (figure A.28).

`Filtres` : ces composants définissent des transformations du contenu en conservant le même format. Nous trouvons les composants :

- `Basique` : seuillage, seuil de confiance, lissage.
- `Transformation géométrique` : applique des opérations linéaires sur le contenu.
- `Prédiction` : basés sur des algorithmes prédictifs (actuellement seul Kalman est défini).

`Modificateur` : ces composants définissent des transformations du contenu en modifiant le format. Nous distinguons :

- `adaptateur` : réalise une transformation avec peu de coût en ressources, comme une réduction dimensionnelle ou changement de bases (conversion RGB->YUV).
- `reconnaissance` : effectue une transformation de contenu à fort coût souvent par analyse du contenu. Nous trouvons les approches par vision ou reconnaissance vocale. Notre architecture supporte *ARToolkit* et *Festival Speech System*.

Le lecteur notera que ARToolkit a été modifié pour être intégré dans notre environnement : nous avons dissocié vidéo et reconnaissance des patterns et supprimé le noyau GLUT. Nous avons intégré en entrée notre gestionnaire vidéo et en sortie une accessibilité vers Performer, offrant alors plus de flexibilité des flots de données.

`Fonctionnel` : ces composants offrent des analyses et des transformations du contenu mais sans propagation de l'information. Nous utilisons pour le moment uniquement la catégorie `calibrage` implémentant différents mécanismes de calibrages des éléments.

`Structurel` : nous retrouvons ici des composants non fonctionnels : développement (noeuds tests), structure de flots (sélection, fusion, opérations booléennes).

Les propriétés non-fonctionnelles supportées sont : différents modèles d'exécution (stream, queue), de supports exécutifs (callback, thread), d'interfaces (local, appel à distance, proxy), de développement (enregistrement des données (fichier de log), simulation (par fichier), affichage état (console)) et la synchronisation.

Module `Interaction`

nous reprenons les support des métaphores classiques de RV :

- `sélection` : directe (virtual hand, go-go), par rayon ou basée image.
- `manipulation` : directe (crayon), par rayon, basée image.

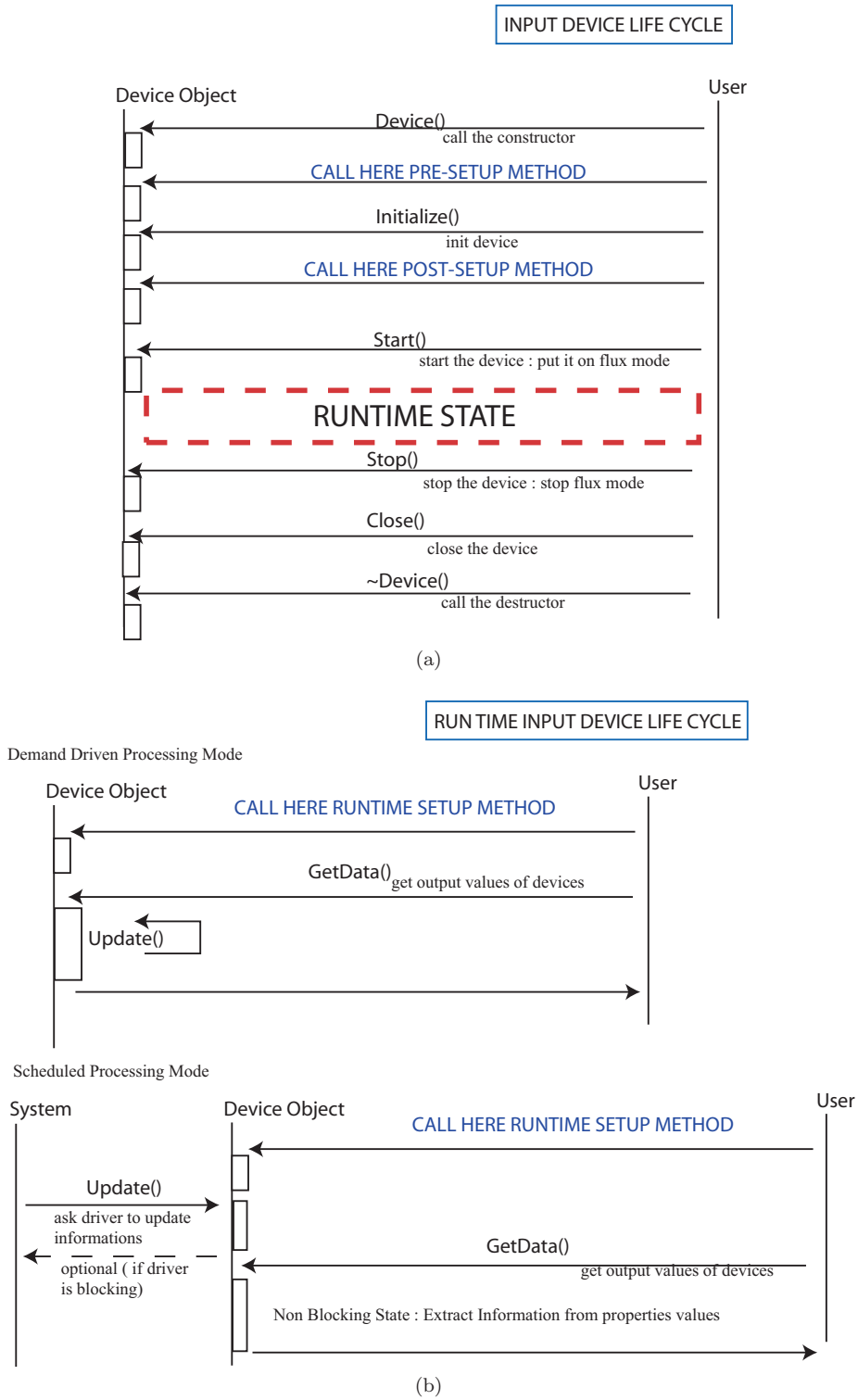


Fig. A.28: Périphérique et mode de communication : (a) cycle de vie d'un périphérique (b) cycle de vie en cours de session.

– `navigation` : mode standard à partir de position 2D (trackball, fly, drive), tête suivie par capteur (avec casque ou pour système projectif), modèles articulés (à partir de n capteurs). Nous intégrons la gestion de la navigation en RA (intégration vidéo, alignement à partir de fichier de calibrage) ;
– `contrôle d'application` : nous supportons la boîte à outils, la reconnaissance de gestes et les interfaces de menus par Qt et notre propre boîte à outils 3D.

Annexe B

Questionnaire pour l'évaluation de l'*interactive mediated reality* et l'application à l'architecture

Nous noterons que le format des questionnaires est ici modifié pour être intégré à la thèse (les 0 désignant des cases à cocher).

1. Questionnaire sur l'évaluation avec le système de peinture

```
Questionnaire sur le système de Peinture en Réalité Augmentée

Bonjour,

Quelques questions relatives à vos impressions suite a cette expérience.
Merci de votre participation.

0. Quelle est votre première impression ?
.................................................................
.................................................................

1. Avez vous trouvé ce concept :
0 intéressant   0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très simple   0 assez simple      0 mitigé 0 peu simple      0 pas simple
0 très efficace 0 assez efficace    0 mitigé 0 peu efficace    0 pas efficace
0 très amusant  0 assez amusant     0 mitigé 0 peu amusant     0 pas amusant

(efficace désigne le fait d une bonne réalisation de ce que l outil propose,
 sur cette question bien peindre et modifier le réel)

2. Comment avez vous apprécié :

Le concept de peindre en virtuel dans le monde réel :
0 très bien  0 bien 0 moyen  0 passable 0 nul
Le concept de déplacer librement la tête pour voir l'objet sous différents angles :
0 très bien  0 bien 0 moyen  0 passable 0 nul
Le concept de déplacer librement l'objet pour le voir sous différents angles :
0 très bien  0 bien 0 moyen  0 passable 0 nul
Le concept de peindre en virtuel avec un pinceau réel :
0 très bien  0 bien 0 moyen  0 passable 0 nul

3.Comment jugez-vous  :

-Le fait d avoir un casque :
```

0 confortable 0 assez confortable 0 moyen 0 peu confortable 0 pas confortable
Quels défauts lui trouvez-vous ?
Temps de retard trop perceptible ? 0 OUI 0 NON
Très mauvaise résolution ? 0 OUI 0 NON
Peu de liberté de mouvements ? 0 OUI 0 NON
Mauvaise alignement Réel et Virtuel ? 0 OUI 0 NON
Autres :

-L espace de travail (table avec le support des éléments et des outils, le fait d être assis) :
approprié : 0 très bien 0 bien 0 moyen 0 passable 0 nulle
Quels défauts lui trouvez vous ?
.....

-La manipulation du pointeur virtuel avec le pinceau réel:
outil approprié :
0 très bien 0 bien 0 moyen 0 passable 0 nul
outil facile à utiliser:
0 très bien 0 bien 0 moyen 0 passable 0 nul
outil efficace :
0 très bien 0 bien 0 moyen 0 passable 0 nul
Quels défauts lui trouvez-vous ?
Trop lourd et peu maniable ? 0 OUI 0 NON
Perte du suivi (le pinceau ne peint plus) ? 0 OUI 0 NON
Mauvaise alignement entre le bout du pinceau et le pointeur virtuel ? 0 OUI 0 NON
Autres : .....

-L utilisation de la palette :
outil approprié :
0 très bien 0 bien 0 moyen 0 passable 0 nul
outil facile à utiliser:
0 très bien 0 bien 0 moyen 0 passable 0 nul
outil efficace :
0 très bien 0 bien 0 moyen 0 passable 0 nul
Quels défauts lui trouvez-vous ?
.....

-L utilisation des zones de création, de saisie (digitalisation des éléments) :

4. Comment jugez vous les outils proposés :
outil approprié :
0 très bien 0 bien 0 moyen 0 passable 0 nul
outil facile à utiliser:
0 très bien 0 bien 0 moyen 0 passable 0 nul
outil efficace :
0 très bien 0 bien 0 moyen 0 passable 0 nul
Quels défauts lui trouvez-vous ?
.....

-D un point de vue global :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très complet 0 assez complet 0 mitigé 0 peu complet 0 pas complet
0 très efficace 0 assez efficace 0 mitigé 0 peu efficace 0 pas efficace

-La possibilité de peindre avec de la peinture virtuelle sur des éléments réels :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très simple 0 assez simple 0 mitigé 0 peu simple 0 pas simple
0 très efficace 0 assez efficace 0 mitigé 0 peu efficace 0 pas efficace

-La possibilité de choisir des couleurs et textures virtuelles :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très simple 0 assez simple 0 mitigé 0 peu simple 0 pas simple
0 très efficace 0 assez efficace 0 mitigé 0 peu efficace 0 pas efficace

-La possibilité de choisir des couleurs et textures réelles digitalisées:
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très simple 0 assez simple 0 mitigé 0 peu simple 0 pas simple
0 très efficace 0 assez efficace 0 mitigé 0 peu efficace 0 pas efficace

-La possibilité de peindre des éléments complets :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très simple 0 assez simple 0 mitigé 0 peu simple 0 pas simple
0 très efficace 0 assez efficace 0 mitigé 0 peu efficace 0 pas efficace

-La possibilité de placer des étiquettes (portes, fenêtres) sur
des éléments réels :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très simple 0 assez simple 0 mitigé 0 peu simple 0 pas simple
0 très efficace 0 assez efficace 0 mitigé 0 peu efficace 0 pas efficace

-La possibilité de placer des étiquettes réelles digitalisées
sur des éléments réels :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très simple 0 assez simple 0 mitigé 0 peu simple 0 pas simple
0 très efficace 0 assez efficace 0 mitigé 0 peu efficace 0 pas efficace

-La possibilité d'ajouter des éléments virtuels prédéfinis (mur)
sur des éléments réels :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très simple 0 assez simple 0 mitigé 0 peu simple 0 pas simple
0 très efficace 0 assez efficace 0 mitigé 0 peu efficace 0 pas efficace

-La possibilité d'ajouter de la matière virtuelle aux objets réels :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très simple 0 assez simple 0 mitigé 0 peu simple 0 pas simple
0 très efficace 0 assez efficace 0 mitigé 0 peu efficace 0 pas efficace

-La possibilité d'ajouter des éléments virtuels (mobilier) dans la scène :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très simple 0 assez simple 0 mitigé 0 peu simple 0 pas simple

0 très efficace 0 assez efficace 0 mitigé 0 peu efficace 0 pas efficace

Quels types de fonctionnalités souhaiteriez vous ajouter à ce système ?
.....

5. Comment jugez-vous d un point de vue réaliste :

La sensation que la peinture est sur la maquette réelle :
quand vous déplacez la tête :
0 très bien 0 bien 0 moyen 0 passable 0 nulle
quand vous déplacez les éléments :
0 très bien 0 bien 0 moyen 0 passable 0 nulle
La sensation de peindre :
0 très bien 0 bien 0 moyen 0 passable 0 nulle
La diffusion de la peinture sur la maquette :
0 très bien 0 bien 0 moyen 0 passable 0 nulle

6. Quelles différences pourriez-vous faire, principalement par rapport au concept, avec dautres outils (sans expérience, passez la question):

-Peinture sur papier :
d un point de vue global :
0 mieux 0 un peu mieux 0 identique 0 moins bien 0 inférieur 0 différent
d un point de vue réaliste :
0 mieux 0 un peu mieux 0 identique 0 un peu moins 0 inférieur
d un point de vue d utilisation :
0 mieux 0 un peu mieux 0 identique 0 un peu moins 0 inférieur

-Peinture sur maquette réelle :
d un point de vue global :
0 mieux 0 un peu mieux 0 identique 0 moins bien 0 inférieur 0 différent
d un point de vue réaliste :
0 mieux 0 un peu mieux 0 identique 0 un peu moins 0 inférieur
d un point de vue d utilisation :
0 mieux 0 un peu mieux 0 identique 0 un peu moins 0 inférieur

-Outils informatiques de dessin 2D:
d un point de vue global :
0 mieux 0 un peu mieux 0 identique 0 moins bien 0 inférieur 0 différent
d un point de vue réaliste :
0 mieux 0 un peu mieux 0 identique 0 un peu moins 0 inférieur
d un point de vue d utilisation :
0 mieux 0 un peu mieux 0 identique 0 un peu moins 0 inférieur

-Outils informatiques de peinture 3D sous logiciels commerciaux :
d un point de vue global :
0 mieux 0 un peu mieux 0 identique 0 moins bien 0 inférieur 0 différent
d un point de vue réaliste :
0 mieux 0 un peu mieux 0 identique 0 un peu moins 0 inférieur

d un point de vue d utilisation :
0 mieux 0 un peu mieux 0 identique 0 un peu moins 0 inférieur

7. Applications

Dans quels types d applications ce système trouverait-il un intérêt :
Architecture (modification maquette) ? 0 oui 0 peut-être 0 je ne sais pas 0 non
Cosmétique (essai maquillage virtuel) ? 0 oui 0 peut-être 0 je ne sais pas 0 non
Jeu (jeu mixte, avec cube R/V) ? 0 oui 0 peut-être 0 je ne sais pas 0 non
Packaging (designer des boîtes) ? 0 oui 0 peut-être 0 je ne sais pas 0 non
Art (nouveau type) ? 0 oui 0 peut-être 0 je ne sais pas 0 non

Quels types dapplication donneriez-vous à ce système :
.....

8. Remarques :
Si le casque était remplacé par de simples lunettes, avec une meilleure résolution,
vous jugeriez le fait d avoir ces lunettes :
0 confortable 0 assez confortable 0 moyen 0 peu confortable 0 pas confortable

Que pensez=vous du fait de pouvoir modifier la vision du monde réel ? Quel type d avenir lui voyez-vous ?
.....
.....

Avez vous des remarques générales sur l intérêt, le réalisme, les limitations, les avantages du système ?
.....
.....

9. Identité :
tranches dâge : 0 10-20 0 20-30 0 30-40 0 40-50 0 50-60
profil : 0 étudiant 0 employé 0 cadre 0 sans emploi 0 autre :.....
domaine: 0 artistique 0 designer 0 architecture 0 autre :.....

Merci encore d'avoir répondu à ce questionnaire.

2. Questionnaire pour l'évaluation en architecture

Questionnaire sur le système de Réalité Augmentée Collaboratif 3D :
Application à l Architecture

Bonjour,

Quelques questions relatives à vos impressions suite à ces démonstrations.
Merci de votre participation.

0. Quelle est votre première impression ?

.....

1. Avez vous trouvez ce concept (table + Réalité Augmentée + interaction 3D) :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
0 très efficace 0 assez efficace 0 mitigé 0 peu efficace 0 pas efficace
0 très amusant 0 assez amusant 0 mitigé 0 peu amusant 0 pas amusant

(efficace désigne le fait d une bonne réalisation de ce que l outil propose)

2. Comment avez-vous apprécié :
Le concept de visualiser des objets virtuels 3D dans le monde réel :
0 très bien 0 bien 0 moyen 0 passable 0 nul

Le concept de toujours pouvoir utiliser des objets réels en même tant que des virtuels :
0 très bien 0 bien 0 moyen 0 passable 0 nul

Le concept de déplacer librement la tête pour voir l'objet sous différents angles :
0 très bien 0 bien 0 moyen 0 passable 0 nul

Le concept de pouvoir interagir naturellement avec les objets virtuels sur une table :
0 très bien 0 bien 0 moyen 0 passable 0 nul

Le concept de peindre en virtuel dans le monde réel, mixer maquette réelle et maquette virtuelle :
0 très bien 0 bien 0 moyen 0 passable 0 nul

Le fait d interagir sur la table avec du contenu tridimensionnel dans un cadre de réunion :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant

3.Comment jugez-vous :

Le fait davoir un casque :
0 confortable 0 assez confortable 0 moyen 0 peu confortable 0 pas confortable

Quels défauts lui trouvez-vous ?
Temps de retard trop perceptible ? 0 OUI 0 NON
Très mauvaise résolution ? 0 OUI 0 NON
Peu de liberté de mouvements ? 0 OUI 0 NON
Mauvaise alignement Réel et Virtuel ? 0 OUI 0 NON

Si le casque était remplacé par de simples lunettes, avec une meilleure résolution,
vous jugeriez le fait d avoir ces lunettes :
0 confortable 0 assez confortable 0 moyen 0 peu confortable 0 pas confortable

4. Concernant votre domaine d application (architecture), vous pensez que ce système est :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant

utile pour du design collaboratif entre architectes : 0 OUI 0 NON
utile pour du travail collaboratif entre les différents
intervenants d'un projet: 0 OUI 0 NON
utile en chaîne finale pour faire des présentations au client : 0 OUI 0 NON

vous pensez que ce système est intéressant :
-pour la manipulation intuitive d éléments de maquette virtuelle :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
-pour la modification/création intuitive d une maquette virtuelle :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
-pour la conservation de l utilisation des outils classiques
(dessin, schéma, maquette réelle) :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant
-pour la collaboration naturelle remplaçant la discussion devant un écran :
0 intéressant 0 assez intéressant 0 mitigé 0 peu intéressant 0 pas intéressant

Quels sont les avantages de ce système concernant votre domaine?
.....
.....
.....

Quelles sont les limitations et inconvénients de ce système concernant
votre domaine?
.....
.....
.....

Quelle fonctionnalités vous sembleraient intéressantes concernant
votre domaine?
.....
.....
.....

Quel avenir voyez-vous à ce type de système concernant votre domaine ?
.....
.....
.....

Quelle est votre « plateforme idéale » (au niveau matériel, outils)
pour collaborer
concernant l architecture ? (3D ou 2D)
.....
.....
.....

Avez-vous des remarques générales ?
.....
.....
.....

5. Identité :
quelle pratique avez-vous des images de synthèse :
utilisation de logiciels de modélisation 3D (3ds Max, Maya) :
utilisation de logiciels professionnels du domaine :
outils de programmation (openGL) :

Merci encore d avoir répondu à ce questionnaire.

Bibliographie

[3D.] Right Hemisphere Deep Paint 3D. http ://www.righthemisphere.com/. 2

[3ds] Discreet 3ds max. http ://www.discreet.com/3dsmax/. 2

[AaSR02] Dzmitry Aliakseyeu, Jean-Bernard Martens andSriram Subramanian, and Mathias Rauterberg. Interaction techniques for navigation through and manipulation of 2d and 3d data. In *Eurographics Workshop on Virtual Environments*, May 2002. 3.1.2, 3.2.1

[AB94] Ronald Azuma and Gary Bishop. Improving static and dynamic registration in an optical see-through hmd. In *SIGGRAPH*, pages 197–204, 1994. 2, 3.2.4, 4.2

[ABB+01] Ronald Azuma, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier, and Blair MacIntyre. Recent advances in augmented reality. *IEEE Computer Graphics and Applications*, 21(6) :34–47, November/December 2001. 1.1

[ABC+94] K. Ahlers, D. Breen, C. Crampton, E. Rose, M. Tuceryan, R. Whitaker, and D. Greer. An augmented vision system for industrial applications. In *Telemanipulator and Telepresence Techonolgie*, volume 2351, pages 345–359. SPIE, Boston, USA, 1994. 1.3, 1.4.1, 3.3.2

[ABFH97] Maneesh Agrawala, Andrew C. Beers, Bernd Fröhlich, and Pat Hanrahan. The two-user responsive workbench : Support for collaboration through individual views of a shared space. In *SIGGRAPH*, 1997. 1.3, 2.1.5, 2.2.6

[ABL95] Maneesh Agrawala, Andrew C. Beers, and Marc Levoy. 3d painting on scanned surfaces. In *Symposium on Interactive 3D graphics*, April 1995. 6.2, 2

[ACI] ACIS. www.spatial.com/. 6.2.2

[ACK01] Nina Amenta, Sunghee Choi, and Ravi Kolluri. The power crust, unions of balls, and the medial axis transform. *Computational Geometry : Theory and Applications*, 19(2-3) :127–153, 2001. 6.2.3

[AKB+95] Klaus H. Ahlers, André Kramer, David E. Breen, Pierre-Yves Chevalier, Chris Crampton, Eric Rose, Mihran Tuceryan, Ross T. Whitaker, and Douglas Greer. Distributed augmented reality for collaborative design applications. *Computer Graphics Forum*, 14(3) :3–14, 1995. 1.3

[AKYT00] M . Anabuki, M. H. Kabuka, H. Yamamoto, and H Tamura. Welbo : An embodied conversational agent living in mixed reality space. In *(CHI), Videos*, 2000. 3.1.5

[AMD04] W. Yu. K. Abdel-Malek and J. Duncan. Human placement for maximum dexterity. In *SAE Digital Human Modeling and Simulation*, 2004. 2.2

[AMIM02] Hideyuki Ando, Takeshi Miki, Masahiko Inami, and Taro Maeda. Smartfinger : Nail-mounted tactile display. In *Siggraph. Emerging Technologies.*, 2002. 11.1

[AMS+01a] Dzmitry Aliakseyeu, Jean-Bernard Martens, Sriram Subramanian, Marina Vroubel, and Wieger Wesselink. Visual interaction platform. In *Interact*, pages 232–239, Tokyo, Japan, July 2001. 2.7

[AMS+01b] Dzmitry Aliakseyeu, Jean-Bernard Martens, Sriram Subramanian, Marina Vroubel, and Wieger Wesselink. Visual interaction platform. In *Interact*, pages 232–239, Tokyo, Japan, 2001. 1

[Amu] Amulet. http ://www-2.cs.cmu.edu/ amulet/. 2.2.9

[Ano] Anoto. www.anoto.com/. 3

[API+98] Kevin Arthur, Timothy Preston, Russell M. Taylor II, Frederick P. Brooks, Jr. Mary C. Whitton, and William V. Wright. Designing and building the pit : a head-tracked stereo workspace for two users. In *International Immersive Projection Technology Workshop*, 1998. 3.3, 2.1.4

[Art03] The Sims. Electronic Arts. http ://thesims.ea.com/, 2003. 10.1

[Azu93] Ronald Azuma. Tracking requirements for augmented reality. *Communications of the ACM*, 36(7) :50–51, July 1993. 1.4.1

[Azu97] Ronald T Azuma. A survey of augmented reality. *Presence : Teleoperators and Virtual Environments*, 1997. 1.1, 4.1

[BB96] W. Bruns and V. Brauer. Bridging the gap between real and virtual modeling - a new approach to human-computer interaction. In *IFIP5.10-Workshop Virtual Prototyping*, Arlington, USA, May 1996. 2.6, 2.4

[BBB03] Uli Bockholt, Alexander Bisler, and Mario Becker. Augmented reality for enhancement of endoscopic interventions. In *Virtual Reality (VR)*, Los Angeles, USA, March 2003. 1.3

[BBGK02] Mark Billinghurst, Daniel Belcher, Arnab Gupta, and Kiyoshi Kiyokawa. Experiments with face-to-face collaborative ar interfaces. *Virtual Reality Journal*, 2002. 2.4.4

[BBK+01] Martin Bauer, Bernd Bruegge, Gudrun Klinker, Asa MacWilliams, Thomas Reicher, Stefan Riss, Christian Sandor, and Martin Wagner. Design of a component-based augmented reality framework. In *Proceedings of ISAR 2001*, 2001. 2.4.8, 3.3.3, 3.4.5

[BBKF97] Ravin Balakrishnan, Thomas Baudel, Gordon Kurtenbach, and George Fitzmaurice. The rockin'mouse : integral 3d manipulation on a plane. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 311–318. ACM Press, 1997. 3.1.5, 3.1.5

[BC93] G. Burdea and P. Coiffet. *Virtual Reality Technology*. John Wiley Sons, 1993. 2.14

[BC02] William A. Barrett and Alan S. Cheney. Object-based image editing. *ACM Transaction Graphics*, 21(3) :777–784, July 2002. 4

[BCG+00] M. Büscher, M. Christensen, K. Grønbæk, P. Krogh, P. Mogensen, D. Shapiro, and P. Ørbæk. Collaborative augmented reality environments : Integrating vr, working materials, and distributed work spaces. In *Collaborative Virtual Environments*, 2000. 2.2.4

[BCG+01] M. Back, J. Cohen, R. Gold, S. Harrison, and S. Minneman. Listen reader : An electronically augmented paper-based book. In *Computer Human Interaction (CHI)*, Seattle, USA, March 2001. 1.3

[Bec99] Kent Beck. *Extreme Programming Explained : Embrace Change*. Addison-Wesley, 1999. 15

[Bed] Bedimo. www.bedimo.be. 1

[Ber99] Francois Berard. *Vision par ordinateur pour l'interaction homme-machine fortement couplée*. PhD thesis, Université Joseph Fourier, Grenoble, France, 1999. 2.2, 2.6

[Ber03] Francois Berard. The magic table : Computer-vision based augmentation of a whiteboard for creative meeting. In *workshop on Projector-Camera Systems (PROCAM)*, Nice, France., 2003. IEEE. 3.1.1, 2

[BES03] Oliver Bimber, L.M. Encarnação, and Dieter Schmalstieg. The virtual showcase as a new platform for augmented reality digital storytelling. In *Eurographics Workshop on Virtual Environemnts*, pages 87–95, 2003. 2.1.6, 4.7, 3.3.6

[BF91] Eric A. Bier and Steven Freeman. Mmm : a user interface architecture for shared editors on a single screen. In *Proceedings of the 4th annual ACM symposium on User interface software and technology*, pages 79–86. ACM Press, 1991. 3.1.3

[BF00] Blaine A. Bell and Steven K. Feiner. Dynamic space management for user interfaces. In *UIST*, pages 239–248, 2000. 3.1

[BFO92] M. Bajura, H. Fuchs, and R. Ohbuchi. Merging virtual objects with the real world : seeing ultrasound imagery within the patient. In *SIGGRAPH*, pages 203–210. ACM, 1992. 1.3, 2.4, 1

[BFSE01] O. Bimber, B. Fröhlich, D. Schmalstieg, and L.M Encarnação. The virtual showcase. *IEEE Computer Graphics and Applications*, 21(6) :48–55, 2001. 1.4.2, 2.8

[BG01] Samuel Boivin and André Gagalowicz. Image-based rendering of diffuse, specular and glossy surfaces from a single image. In *Proceedings of ACM SIGGRAPH 2001*, 2001. 2

[BG02] Patrick Baudisch and Nathaniel Good. Focus plus context screens : displays for users working with large visual documents. In *CHI '02 extended abstracts on Human factors in computing systems*, pages 492–493, 2002. 2.10

[BGG01] Kevin Baker, Saul Greenberg, and Carl Gutwin. Heuristic evaluation of groupware based on the mechanics of collaboration. *Lecture Notes in Computer Science*, 2001. 2

[BGW+02] Oliver Bimber, Gatesy, Witmer, Ramesh Raskar, and Miguel Encarnação. Merging fossil specimens with computer-generated information. *IEEE Computer*, 35(9) :45–50, September 2002. 3

[BH03] Leonard D Brown and Hong Hua. A widget framework for augmented collaboration in scape. In *User Interface Software and Technology*, 2003. 2.4.9, 3.1.1, 3.2.1

[BHF+99] A. Butz, T. Hollerer, S. Feiner, B. MacIntyre, and C. Beshers. Enveloping users and computers in a collaborative 3d augmented reality. In *International Workshop on Augmented Reality (IWAR)*, 1999. 3.9, 2.4.2, 3.1.1, 3.4, 3.1.4

[BHM+03] Martin Bauer, Otmar Hilliges, Asa MacWilliams, Christian Sandor, Martin Wagner, Joe Newman, Gerhard Reitmayr, Tamer Fahmy, Gudrun Klinker, Thomas Pintaric, and Dieter Schmalstieg. Integrating

studierstube and dwarf. In *International Workshop on Software Technology for Augmented Reality Systems (STARS)*, 2003. 3.3.3

[BHST95] R. Bentley, T. Horstmann, K. Sikkel, and J. Trevor. Supporting collaborative information sharing with the WWW : The BSCW shared workspace system. pages 63–74, 1995. 3.1.4

[BIJ+00] J.D. Brederson, M. Ikits, C.R. Johnson, C.D. Hansen, and J.M. Hollerbach. The visual haptic workbench. In *Proceedings of the Fifth PHANToM Users Group Workshop*, pages 46–49, October 2000. 2.1.5

[BK99] Ravin Balakrishnan and Gordon Kurtenbach. Exploring bimanual camera control and object manipulation in 3d graphics interfaces. In *ACM*, pages 53–63, 1999. 3.1.5

[BK00] A. Bicchi and V. Kumar. Robotic grasping and contact : A review. In *IEEE International Conference on Robotics and Automation*, April 2000. 2.6

[BKP01a] M. Billinghurst, H. Kato, and I. Poupyrev. The magicbook : Moving seamlessly between reality and virtuality. *IEEE Computer Graphics and Applications*, pages 2–4, 2001. 1.4.1, 3.2.5

[BKP01b] S. Bødker, P. Krogh, and M.G. Petersen. The interactive design collaboratorium. In *Interact*, pages 51–58, 2001. 2.2.4

[BKWF99] Mark Billinghurst, Hirokazu Kato, S. Weghorst, and T. A. Furness. A mixed reality 3dconferencing application. Technical report, Human Interface Technology Laboratory, University of Washington, 1999. 2.4.4, 3.3.5, 3.1.1

[BL01] Michel Beaudouin-Lafon. Novel interaction techniques for overlapping windows. In *Symposium on User interface Software and Technology (UIST)*, pages 153–154, 2001. 3.1

[BLEE+99] Michel Beaudouin-Lafon, K. Ehrlich, C.A. Ellis, W.E Mackay, H. Ishii, A. Prakash, S. Greenberg, M. Roseman, P. Dewan, P. Dourish, and C. Johnson. *Computer Supported Co-operative Work*. Wiley, 1999. 2.1, 2.3, 4.3, 3.1.3

[BLMP00] D. Bowman, E. Kruijff Joseph LaViola, Mark Mine, and Yvan Poupyrev. Three-dimensional user interfaces : Fundamental techniques, theory, and practice, 2000. Course presented at ACM SIGGRAPH. 2.9, 2.1, 2.3, 6.2

[BMS+99] M. Btischer, P. Mogensen, D. Shapiro, I., and Wagner. he manufaktur : Supporting work practice in (landscape) architecture. In *European Conference on Computer Supported Cooperative Work (ECSCW)*, pages 21–40, 1999. 2.2.4

[BMS00] Wolfgang Broll, Eckhard Meier, and Thomas Schardt. The virtual round table a collaborative augmented. In *International Conference on Collaborative Virtual Environments*, 2000. 2.4.5, 2.6

[BN95] Michael Bajura and Ulrich Neumann. Dynamic registration corrections in augmented-reality systems. In *Virtual Reality Annual International Symposium (VRAIS)*, March 1995. 3.2.4

[BNB00] Greg J. Badros, Jeffrey Nichols, and Alan Borning. Scwm : An intelligent constraint-enabled window manager. In *Spring Symposium on Smart Graphics*, March 2000. 3.1

[BNRP02] J. Bouchet, L. Nigay, P. Renevier, and L. Pasqualetti. Troc : un jeu collaboratif sur support mobile exploitant des techniques de réalité augmentée. In *groupe de travail 4.2 "Mobilité et Ubiquité", 2nd assises du GdR i3 (Information - Interaction -Intelligence) du CNRS*, Nancy, France, Décembre 2002. 2.2

[Bol80] Richard A. Bolt. «put-that-there» : Voice and gesture at the graphics interface. In *Proceedings of the 7th annual conference on Computer graphics and interactive techniques*, pages 262–270. ACM Press, 1980. 2.7, 3.1.3

[Bow99] Doug A. Bowman. *Interaction Techniques for Common Tasks in Immersive Virtual Environments : Design, Evaluation, and Application.* PhD thesis, Georgia Institute of Technology, 1999. 2.2, 2.3, 3.1.3, 3.2, 3.3

[BR86] S. Bly and J. Rosenberg. A comparison of tiled and overlapping windows. In *Computer Human Interaction (CHI)*, pages 101–106, New York, USA, 1986. 3.1

[BRF01] Deepak Bandyopadhyay, Ramesh Raskar, and Henry Fuchs. Dynamic shader lamps : Painting on real objects. In *The Second IEEE and ACM International Symposium on Augmented Reality (ISAR'01)*, October 2001. 1, 2

[Bro98] Wolfgang Broll. Dwtp-an internet protocol for shared virtual environments. In *Virtual Reality Modeling Language Symposium (VMRL)*, pages 49–56, February 1998. 2.4.5

[BRSB03] Rafael Ballagas, Meredith Ringel, Maureen Stone, and Jan Borchers. istuff : A physical user interface toolkit for ubiquitous computing environments. In *CHI*, 2003. 2.2.3

[BRTF02] Jan Borchers, Meredith Ringel, Joshua Tyler, and Armando Fox. Interactive workspaces : A framework for physical and graphical user interface prototyping. *IEEE Wireless Communications"*, 9(6) :64–69, 2002. 3.2

[BSLM01] William Baxter, Vincent Scheib, Ming Lin, and Dinesh Manocha. Dab : Interactive haptic painting with 3d virtual brushes. In *Proceedigns of ACM SIGGRAPH 01*, pages 461–468, 2001. 2

[Bux92] William Buxton. Telepresence : integrating shared task and person spaces. In *Graphics Interface*, pages 123–129, 1992. 2.3

[BvTZ04] A. Bartoli, E. von Tunzelmann, and A. Zisserman. Augmenting images of non-rigid scenes using point and curve correspondences. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, Washington, USA, June 2004. 4.5

[BWF98] Mark Billinghurst, S. Weghorst, and T. A. III Furness. Shared space : An augmented reality approach for computer supported collaborative work. In *Virtual Reality*, pages 25–36. IEEE, 1998. 2.4.4, 3.10

[Bér94] Francois Bérard. Vision par ordinateur pour la realité augmentée : Application au bureau numérique. Master's thesis, Joseph Fourier University, Grenoble, France, 1994. 1.2

[Cad] Barco CadWall. http ://www.barco.com/virtualreality/. 3.2

[Cal] Thrustmaster Calibration. www.thrustmaster.com. 5.3

[Can] MetaCreations Canoma. www.metacreations.com/. 5.21, 6.1

[CAS+97] Cassidy J. Curtis, Sean E. Anderson, Joshua E. Seims, Kurt W. Fleischer, and David H. Salesin. Computer-generated watercolor. *Computer Graphics*, 31(Annual Conference Series) :421–430, 1997. 2

[CAV] CAVELib. http ://www.vrco.com/products/cavelib/cavelib.html. 2.1.3

[CBC+99] J. Coutaz, F. Berard, E. Carraux, W. Astier, and J. Crowley. Comedi : using computer vision to support awareness and privacy in mediaspaces. In *CHI*, 1999. 2.3

[CC95] F. Crowley, J.L. Bérard and J. Coutaz. Finger tracking as an input device for augmented reality. In *International Workshop on Automatic Face and Gesture Recognition*, Zurich, Switzerland, June 1995. 3.1.1, 2

[CCM] v3.0 CORBA Component Model. *http ://www.omg.org*. OMG. 2.4.7

[CCRR02] J. L. Crowley, J. Coutaz, G. Rey, and P. Reignier. Perceptual components for context aware computing. In *UBICOMP 2002 - International Conference on Ubiquitous Computing*. Springer Verlag, September 2002 2002. 2.5

[CCT01] G. Calvary, J. Coutaz, and D. Thevenin. A unifying reference framework for the development of plastic user interfaces. In *EHCI*, Toronto, Canada, May 2001. 4

[CDOS95] B. Cameron, D.A DePalma, R. O'Herron, and N. Smith. Where does groupware fit ? *Forrester Report : Software Strategies*, 6(3), 1995. 2.2

[Cen] SGI Reality Center. http ://www.sgi.com/realitycenter/. 3.2, 2.1.1

[CFH97] Lawrence D. Cutler, Bernd Frolich, and Pat Hanrahan. Two-handed direct manipulation on the responsive workbench. In *Proceedings of the 1997 symposium on Interactive 3D graphics*, pages 107–ff. ACM Press, 1997. 3.1.3, 3.3.2

[CH93] C. Carlsson and O. Hagsand. Dive - a multi-user virtual reality system. In *Virtual Reality Annual International Symposium (VRAIS)*, pages 394–400, Los Alamitos, CA, 1993. IEEE. 3.2

[CJJ99] V. Coors, V. Jung, and U. Jasnoch. Using the virtual table as an interaction platform for collaborative urban planning. *Computers Graphics*, 23(4), 1999. 2.1.5, 4.2

[CLBB02] Joelle Coutaz, Christophe Lachenal, Francois Berard, and Nicolas Barralon. *Les cahiers du numérique : Ces objets qui communiquent*, chapter Quand les surfaces deviennent interactives., pages 101–126. Lavoisier, 2002. 2.3.2

[CLDC03] Joelle Coutaz, Christophe Lachenak, and Sophie Dupuy-Chessa. Ontology for multi-surface interaction. In *Interact*, 2003. 3.2, 4.1, 7.3

[CLPR98] Omer Casher, Christopher Leach, Christopher S. Page, and Henry S. Rzepa. Virtual reality modelling language (vrml) in chemistry. *Chemistry in Britain*, 34(26), 1998. 3.1

[CM92] T. Caudel and D. Mizell. Augmented reality : an application of heads-up display technology to manual manufacturing processes. In *Hawaii International Conference on System Sciences*, 1992. 1.1, 1.3

[CMBZ00] M. Capps, D. McGregor, D. Brutzman, and M. Zyda. Npsnet-v : A new beginning for dynamically extensible virtual environments. *IEEE Computer Graphics and Applications*, 20(5) :12–15, 2000. 3.2, 2.15

[CNSA+92] C. Cruz-Neira, D. J. Sandin, T. A.DeFanti, R. V. Kenyon, and J.C Hart. The cave : Audio visual experience automatic virtual environment. In *SIGGRAPH*, 1992. 1.3

[CNSD+92] C. Cruz-Neira, D.J. Sandin, T.A. DeFanti, R.V. Kenyon, and J.C. Hart. The cave : Audio visual experience automatic virtual environment. In *SIGGRAPH*, pages 65–72, 1992. 3.3, 2.1.3

[Cop77] LucasFilm Copyright. Star wars, l'empire contre attaque, 1977. 4.6

[Cor03] Brigitte Cord. Internet et pedagogie - etat des lieux, 2003. 2.1

[Cou00] Joelle Coutaz. Interaction homme-machine. cours de dea., 2000. 6.2

[Cry01] Cryopolis. www.cryopolis.com, 2001. 3.2

[Cur] Fakespace Curv. http ://www.fakespace.com/curv.shtml. 2.1.1

[CWY+02] A. Cheok, W. Weihua, X. Yang, S. Prince, F. Wan, M. Billinghurst, and H. Kato. Interactive theatre experience in embodied and wearable mixed reality space. In *International Symposium on Augmented Reality Environment (ISMAR)*, pages 59–68. IEEE and ACM, 2002. 1.3, 2.2

[CYY+02] Adrian David Cheok, Xubo Yang, Zhou Zhi Ying, Mark Billinghurst, and Hirokazu Kato. Touch-space : Mixed reality game space based on ubiquitous, tangible, and social computing. *Personal Ubiquitous Comput.*, 6(5-6) :430–442, 2002. 2.2

[Dal02] Laurent Da Dalto. Starmate : using augmented reality for maintenance, training and education. In *Virtual Reality International Conference (VRIC)*, Laval, France, 2002. 1.3

[DC02] Alexandre Demeure and Gaelle Calvary. Jeu et réalité mixte : retours d'expérience. In *Conférence Francophone sur l'Interaction Homme-Machine*, pages 89–96, Poitiers, France, 2002. 2.2

[dCTE] Les Colons de Catane. Tilsit Editions. http ://www.tilsit.fr/. 10.1

[DDC+99] C. Dumas, S. Degrande, C. Chaillou, G. Saugis, and P. Plénacoste et ML. Viaud. Spin : A 3-d interface for cooperative work. *Virtual Reality Journal*, (4) :15–25, 1999. 3.2, 2.15, 1, 1

[Deb98] Paul Debevec. Rendering synthetic objects into real scenes : Bridging traditional and image-based graphics with global illumination and high dynamic range photography. *Computer Graphics*, 32(Annual Conference Series) :189–198, 1998. 2, 4.3.1

[Deb00] Gilles Debunne. *Animation multirésolution d'objets déformables en temps-réel : Application à la simulation chirurgicale*. PhD thesis, Institut national Polytechnique de Grenoble, 2000. 2.2

[Dem] ATI Demo. http ://www.ati.com/developer/demos.html. 2.13

[DF01] Pierre Dragicevic and Jean-Daniel Fekete. Input device selection and interaction configuration with icon. In *IHM-HCI*, 2001. 3.4.5

[DGHP02] Ralf Dörner, Christian Geiger, Michael Haller, and Volker Paelke. Authoring mixed reality. a component and framework-based approach. In *International Workshop on Entertainment Computing (IWEC)*, Makuhari, Japan, May 2002. 3.3.3

[DGN02] Emmanuel Dubois, P. D. Gray, and Laurence Nigay. Asur++ : a design notation for mobile mixed system. In *international conference MobileHCI*, Pisa, Italy, September 2002. 1.2

[Div] Design Group Division. *Air Force Interior Design Guides. Chapitre 5 : Ergonomic and Environmental Guide.* http ://www.afcee.brooks.af.mil/dc/. 2.3.2, 4.4

[Dix94] A. J. Dix. *Design Issues in CSCW*, chapter Computer-supported cooperative work - a framework, pages 23–37. Springer Verlag, 1994. 2.1, 2.9, 2.2

[DJC+03] Miguel Dias, Joaquim Jorge, José Carvalho, Pedro Santos, and José Luzio. Usability evaluation of tangible user interfaces for augmented reality. In *ARtoolkit workshop*, 2003. 2.4.6

[DK02] Patrick Dähne and John N. Karigiannis. Archeoguide : System architecture of a mobile outdoor augmented reality system. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, 2002. 1.4.2

[DL01] P.H Dietz and D.L. Leigh. Diamondtouch : A multi-user touch technology. In *Symposium on User Interface Software and Technology (UIST)*, pages 219–226, November 2001. 3.7, 2.2.7

[DMS+02] Jose Miguel Salles Dias, Luis Monteiro, Pedro Santos, Rui Silvestre, and Nancy Diniz Rafael Bastos. Tangible interaction for conceptual architectural design. In *ART*, 2002. 3.11, 2.4.6, 3.1.1

[DNH03] Stephen DiVerdi, Daniel Nurmi, and Tobias Höllerer. Arwin - desktop augmented reality window manager. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, October 2003. 3.4

[DNT+99] Emmanuel Dubois, Laurence Nigay, Jocelyn Troccaz, Oliver Chavanon, and Lionnel Carrat. Classification space for augmented surgery, an augmented reality case study. In *Interact*, pages 353–359, Edinburgh, United Kingdom, September 1999. 1.2, 1.2

[dP03] Stéphane Louis dit Picard. *Plate-forme de communication distribuée pour les Environnements Virtuels Collaboratifs 3D à fort couplage d'activité synchrone.* PhD thesis, Université des Sciences et Technique de Lille, 2003. 3.2, 3.5

[DPBG03] Jeff Dyck, David Pinelle, Barry Brown, and Carl Gutwin. Learning from games : Hci design innovations in entertainment software. In *GI*, 2003. 2.1

[DS98] Prasun Dewan and Honghai Shen. Controlling access in multiuser interfaces. *ACM Transactions on Computer-Human Interaction*, 5(1) :34–62, 1998. 3.1.4, 3.1.4, 3.1.4

[DSAP03] F. Doil, W. Schreiber, T. Alt, and C. Patron. Augmented reality for manufacturing planning. In *Proceedings of the workshop on Virtual environments 2003*, pages 71–76. ACM Press, 2003. 2.2

[DSB+03] José Miguel Salles Dias, Pedro Santos, Rafael Bastos, Luis Monteiro, and Rui Silvestre. Developing and authoring mixed reality with mx toolkit. In *International Augmented Reality Toolkit Workshop*, Tokyo, Japan, October 2003. 2.4.6, 3.3.1, 3.1.1, 2.4

[Dub01] Emmanuel Dubois. *Chirurgie Augmentée : un Cas de Réalité Augmentée ; Conception et Réalisation Centrée sur l ?Utilisateur.* PhD thesis, Joseph Fourier University, Grenoble, 2001. 1.1, 1.2, 6.2

[Dum99] Cédric Dumas. *Un modèle d'interaction en 3D : Interaction homme-machine et homme-machine-homme dans les interfaces 3D pour le TCAO synchrone.* PhD thesis, Université des Sciences et Technique de Lille, 1999. 3.2

[Dup] Dupontavice. www.dupontavice.com. 1

[DUS01] Klaus Dorfmüller-Ulhaas and D. Schmalstieg. Finger tracking for interaction in augmented environments. In *International Symposium on Augmented Reality (ISAR)*, pages 55–64, October 2001. 3.1.1

[E] Evans and Sutherland Digistar 3. http ://www.es.com/products/. 3.2, 2.1.1

[EBG+92] S. Elrod, R. Bruce, R. Gold, D. Goldberg, F. Halasz, W. Janssen, D. Lee, K. McCall, D. Pedersen, K. Pier, J. Tang, and B. Welc. Liveboard : A large interactive display supporting group meetings. In *CHI*, pages 599–607, 1992. 2.2.5

[EBSC99] L. M. Encarnacao, O. Bimber, Dieter Schmalstieg, and S. D. Chandler. A translucent sketchpad for the virtual table exploring motion-based gesture recognition. In *Eurographics*, 1999. 3.3.1

[EGR91] Clarence A. Ellis, Simon J. Gibbs, and Gail Rein. Groupware : some issues and experiences. *Communication of ACM*, 34(1) :39–58, 1991. 2.2

[EKB99] H. Ernst, K. Schäfer K., and W. Bruns. Creating virtual worlds with a graspable user interface. interactions in virtual worlds. In *Workshop on Language Technology*, pages 45–57, 1999. 2.6

[EKEK93] Jannick P. Rolland Emily K. Edwards and Kurtis Keller. Video see-through design for merging of real and virtual environments. In *VRAIS*, 1993. 1.4.1

[Elm] Niklas Elmqvist. 3dwm : The three dimensional workspace manager. Chalmers Medialab, Chalmers University of Technology. 3.1

[Eve04] Everquest2. Sony copyright. http ://everquest2.station.sony.com/, 2004. 3.2, 2.15

[FAD02] Chris Furmanski, Ronald Azuma, and Mike Daily. Augmented-reality visualizations guided by cognition : Perceptual heuristics for combining visible and obscured information. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, 2002. 2

[FD02] Steven Spielberg. Twentieth Century Fox and Dreamwork, 2002. 1.1

[FdAMS02] Michele Fiorentino, Raffaele de Amicis, Giuseppe Monno, and Andre Stork. Spacedesign : A mixed reality workspace for aesthetic industrial design. In *International Symposium on Mixed and Augmented Reality (ISMAR'02)*, October 2002. 6.2.2, 2

[fFMoIS03] SmartSkin : An Infrastructure for Freehand Manipulation on Interactive Surfaces. Jun rekimoto. In *CHI*, 2003. 3.7, 2.2.7

[FHFG99] A. Fuhrmann, G. Hesina, F. Faure, and Michael Gervautz. Occlusion in collaborative augmented environment. In *EGVE*, 1999. 4.3.1

[FHP98] Eric Foxlin, Michael Harrington, and George Pfeifer. Constellation : a wide-range wireless motion-tracking system for augmented reality and virtual set applications. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 371–378. ACM Press, 1998. 1.4.1

[FIB95] George W. Fitzmaurice, Hiroshi Ishii, and William Buxton. Bricks : Laying the foundations for graspable user interfaces. In *Computer Human Interaction (CHI)*, pages 442–449, 1995. 1.2, 1.3, 2.5, 3.1.5, 3.3.2

[Fit96] George W. Fitzmaurice. Graspable user interfaces. In *Department of Computer Science, University of Toronto*, 1996. 3.1.1, 3.3.2

[FIV+99] M. Fjeld, N. Ironmonger, F. Voorhorst, M. Bichsel, and M. Rauterberg. Camera control in a planar, graspable interface. In *International Conference Applied Informatics (AI)*, pages 242–245. IASTED, 1999. 3.2.1

[FL03] Dieter Schmalstieg Florian Ledermann. Presenting past and present of an archaeological site in the virtual showcase. In *VAST. extended abstract.*, 2003. 2.4.3, 3.3.6

[FLS97] Anton Fuhrmann, Helwig Löffelmann, and Dieter Schmalstieg. Collaborative augmented reality : Exploring dynamical systems. In *Visualization*, pages 459–462. IEEE, 1997. 1.3

[FLSG98] Anton Fuhrmann, Helwig Lffelmann, Dieter Schmalstieg, and Michael Gervautz. Collaborative visualization in augmented reality. *IEEE Computer Graphics and Applications*, 18(4) :54–59, July/August 1998. 2.4.3

[FM01] Philippe Fuchs and Guillaume Moreau. *Le traité de la réalité virtuelle, Volume 1 : Fondements et interfaces comportementales*. Eyrolles, 2001. 3.1, 2.14

[FMS93] Steve Feiner, Blair MacIntyre, and D. Seligman. Knowledge-based augmented reality. *Communications of the ACM*, 36(7) :52–62, July 1993. 1.1, 1.3, 2.4, 3.4, 1, 3

[FOL02] Mark Foskey, Miguel A. Otaduy, and Ming C. Lin. Artnova : Touch-enabled 3d model design. In *IEEE Virtual Reality Conference 2002*, March 2002. 2, 4.3.1, 4.4

[Fou94] Alain Fournier. Illumination problems in computer augmented reality. TR 95-35, Department of Computer Science at the University of British Columbia, 1994. 2

[FPH+94] James D. Foley, Richard L. Phillips, John F. Hughes, Andries van Dam, and Steven K. Feiner. *Introduction to Computer Graphics*. Addison-Wesley Longman Publishing Co., Inc., 1994. 6.2.2

[FPTP01] Anton L. Fuhrmann, Jan Prikryl, Robert F. Tobler, and Werner Purgathofer. Interactive content for presentations in virtual reality. In *VRST*, 2001. 3.3.6, 3.1, 3.2.4, 3.2.5

[FRB99] J. Falk, Johan Redström, and Staffan Björk. Amplifying reality. *Lecture Notes in Computer Science*, 1707 :274–282, 1999. 1.2

[Fri02] Wolfgang Friedrich. Arvika ? augmented reality for development, production and service. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, 2002. 1.3

[FS91] Steven Feiner and Ari Shamash. Hybrid user interfaces : breeding virtually bigger interfaces for physically smaller computers. In *Proceedings of the 4th annual ACM symposium on User interface software and technology*, pages 9–17. ACM Press, 1991. 1

[FSP99] Anton Fuhrmann, Dieter Schmalstieg, and Werner Purgathofer. Fast calibration for augmented reality". In *Virtual Reality Software and Technology*, pages 166–167, London, United Kingdom, 1999. 3.2.4, 3.2.4

[FSSK00] Morten Fjeld, Sissel Guttormsen Schär, Domenico Signorello, and Helmut Krueger. Alternative tools for tangible interaction : A usability evaluation. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, Darmstadt, Germany, 2000. 3.1.1

[FV02] M. Fjeld and B. Voegtli. Augmented chemistry : An interactive educational workbench. In *International Symposium of Mixed and Augmented Reality (ISMAR)*, pages 259–260. IEEE, 2002. 1.3, 2.3.2, 2.6, 3.2

[FVB+99] M. Fjeld, F. Voorhorst, M. Bichsel, K. Lauche, M. Rauterberg, and H. Krueger. Exploring brick-based navigation and composition in an augmented reality. *Lecture Notes in Computer Science*, 1707, 1999. 2.3.2, 3.2.1

[Gab01] Joseph L. Gabbard. Researching usability design and evaluation guidelines for augmented reality (ar) systems, 2001. 6.2

[GAGL00] G.Goebbels, P. Th. Aquino, M. Göbel, and V. Lalioti. Supporting team work in collaborative virtual environments. In *International Conference on Artificial Reality and Tele-existence (ICAT)*, pages 25–27, Taipei, Taiwan, October 2000. 2, 9.2

[GB91] Saul Greenberg and R. Bohnet. Groupsketch : A multi-user sketchpad for geographically-distributed small groups. In *Graphics Interface*, pages 207–215, 1991. 2.3

[GC89] Saul Greenberg and E. Chang. Computer support for real time collaborative work. In *Conference on Numerical Mathematics and Computing*, Winnipeg, Manitoba, 1989. 4.1, 2.3.1

[GC01] Jérome Grosjean and Sabine Coquillart. Command and control cube : a shortcut paradigm for virtual environments. In *IPT-EGVE*, Stuttgart, Germany, 2001. 3.3

[GCHH03] S. Gibson, J. Cook, T.L.J. Howard, and R.J. Hubbold. Rapid shadow generation in real-world lighting environments. In *Eurographics Symposium on Rendering*, Leuven, Belgium, June 2003. 2

[GD01] T. Gleue and P. Daehne. Design and implementation of a mobile device for outdoor augmented reality in the archeoguide project. In *Virtual Reality, Archaeology, and Cultural Heritage International Symposium (VAST)*, Glyfada, Greece, November 2001. 5

[GDG01] Raphael Grasset, Xavier Decoret, and Jean-Dominique Gascuel. Augmented reality collaborative environment :calibration and interactive scene editing. In *VRIC 2001*, 2001. Laval Virtual. 7

[GEG+03] G.Goebbels, E.Kruijff, G.Galunic, T.Orthey, A.Ivanovic, and L. Sanfilippo. Ofuturo - a mixed reality desk environment for supporting creativity methods. In *Symposium on Virtual Reality (SVR)*, 2003. 3.8, 2.3.3

[GF03] S. Güven and S. Feiner. Authoring 3d hypermedia for wearable augmented and virtual reality. In *International Symposium on Wearable Computers (ISWC)*, pages 118–126, October 2003. 3.3.6

[GG00] C. Gutwin and S. Greenberg. The mechanics of collaboration : Developing low cost usability evaluation methods for shared workspaces. In *Workshops on Enabling Technologies : Infrastructure for Collaborative Enterprises (WETICE)*, pages 98–103, June 2000. 2

[GG02] Raphael Grasset and Jean-Dominique Gascuel. Mare : Multiuser augmented reality environment on real table setup. In *ACM SIGGRAPH Conference Abstracts and Applications*, 2002. 4

[GG03] Raphael Grasset and Jean-Dominique Gascuel. Réalité augmentée et environnement collaboratif : Un tour d'horizon. In *AFIG'03*, 2003. Actes des 16emes journeees de l'AFIG. 3.1, 4

[GGR96] Carl Gutwin, Saul Greenberg, and Mark Roseman. Workspace awareness support with radar views. In *CHI*, pages 210–211, 1996. 4.1

[GGS03] Raphael Grasset, Jean-Dominique Gascuel, and Dieter Schmalstieg. Interactive mediated reality. In *Poster at ISMAR (International Symposium on Augmented Reality) 2003*, 2003. 5

[GHI99] Joseph L. Gabbard, Deborah Hix, and J. Edward Swan II. User-centered design and evaluation of virtual environments. *IEEE Computer Graphics and Applications*, 19(6) :51–59, November-December 1999. 10

[GHJV95] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design Patterns : Elements od Reusable Object-Oriented Software*. Addison-Wesley Professional Computing Series. Addison-Wesley Publishing Company, New York, NY, 1995. 6.1

[GLGT99] Arthur D. Gregory, Ming C. Lin, Stefan Gottschalk, and Russel Taylor. H-collide : A framework for fast and accurate collision detection for haptic interaction. In *IEEE Virtual Reality*, pages 38–45, 1999. 3.4.4

[Glo] Gloss. http ://www.gloss.cs.strath.ac.uk/. 3.2

[GLPI+96] W.E.L. Grimson, T. Lozano-Perez, W.M. Wells III, G.J. Ettinger, S.J. White, and R. Kikinis. An automatic registration method for frameless stereotaxy, image guided surgery, and enhanced reality visualization. *Transactions on Medical Imaging*, 1996. 1.3, 2.4

[GM00] Simon Gibson and A. Murta. Interactive rendering with real-world illumination. In *11th Eurographics Workshop on Rendering*, 2000. 2

[GMS+03] Erik Granum, Thomas B. Moeslund, Mortiz Stoerring, Wolfgang Broll, and Michael Wittkaemper. Facilitating the presence of users and 3d models by the augmented round table. In *PRESENCE Conference*, Aalborg, Denmark, October 2003. 2.4.5, 3.11

[GMØ01] K. Grønbæk K.K. Gundersen, P. Mogensen, and P. Ørbæk. Interactive room support for complex and distributed design projects. In *Interact*, 2001. 3.6, 2.2.4

[GOI98] M. Gorbet, M. Orth, and H. Ishii. Triangles : Tangible interface for manipulation and exploration of digital information topography. In *CHI*, Los Angeles,USA, April 1998. 2.2

[Gol98] Benjamin Goldstein. Tandemvr : A framework for collaborative virtual environments. Master's thesis, University of Illinois at Chicago, USA, 1998. 3.1

[Gos01] SpaceCube. Jeremy Goslin. http ://spacecube.jeremygoslin.com/, 2001. 3.1.1, 3.2

[GR01] Saul Greenberg and Michael Rounding. The notification collage : posting information to public and personal displays. In *CHI*, pages 514–521, 2001. 2.3, 2

[GR03] Dieter Schmalstieg Gerhard Reitmayr. Location based applications for mobile augmented reality. In *Australasian User Interface Conference*, 2003. 1.4.2, 2.4.3

[Gra98] Raphael Grasset. Etude d'un environnement virtuel collaboratif basé sur un capteur magnétique et un casque de rv. Rapport de Magistère. Université Joseph Fourier., 1998. 2.3, 2.4

[Gra99a] Raphael Grasset. Etude d'un environnement de realité augmentée et son application aux jeux videos. Master's thesis, Institut National Polytechnique de renoble, 1999. 3.2.5, 4.1, 5.2, 6.2.1, 2.3

[Gra99b] Raphael Grasset. Extension d'un environnement de réalité augmentée. Rapport de Magistère. Université Joseph Fourier., 1999. 3.2.5

[Gra01] Xavier Granier. *Controle automatique de qualite pour l'eclairage global*. PhD thesis, UJF, Novembre 2001. 10.5, 4.2

[Gro] Mac Leonard Research Group. http ://mac-leonard4.embl-heidelberg.de/filnet/3d/actin.html. 2.13

[GS00] B. Garth and D. Shoemaker. Privacy and awareness in multiplayer electronic games. In *Western Computer Graphics Symposium (WCGS)*, pages 117–121, Panorama Mountain Village, Canada, March 2000. 2.1

[GSW01] François Guimbretière, Maureen Stone, and Terry Winograd. Fluid interaction with high-resolution wall-size displays. In *Symposium on User Interface Software and Technology (UIST)*, 2001. 2.2.3

[GTN02] Yackup Genc, Mihran Tuceryan, and Nassir Navab. Practical solutions for calibration of optical see-through devices. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 169–175, Darmstadt, Germany, September 2002. 3.1

[GV99] A. Guye-Vuillieme. Non-verbal communication interface for collaborative virtual environments. *Virtual Reality Journal*, 4 :49–59, 1999. 2

[GW00] Francois Guimbrètiere and Terry Winograd. Flowmenu : Combining command, text, and data entry. In *Symposium on User Interface Software and Technology (UIST)*, 2000. 2.2.3

[GWN+03] M. Gross, S. Wuermlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. Van Gool, S. Lang, K. Strehlke, A. Vande Moere, and O. Staadt. blue-c : A spatially immersive display and 3d video portal for telepresence. In *SIGGRAPH*. ACM, 2003. 2.1.3

[GWW00] François Guimbretière, Terry Winograd, and Sha Xin Wei. The geometer's workbench : An experiment in interacting with a large, high resolution display. Technical Report 05, Stanford University, 2000. 2.2.3

[Hal97] Michael Halle. Autostereoscopic displays and computer graphics. In *SIGGRAPH*, volume 31, pages 58–62, May 1997. 2.4.2

[HB99] J.P Hourcade and B. Bederson. Architecture and implementation of a java package for multiple input devices (mid). Technical Report 4018, Human-Computer Interaction Lab, University of Maryland, USA, 1999. 3.1.2

[HBGA03] Hong Hua, Leonard D. Brown, Chunyu Gao, and N. Ahuja. A new collaborative infrastructure : Scape. In *Virtual Reality*, Los Angeles, USA, March 2003. IEEE. 3.12, 2.4.9, 3.2.1

[HBP+01] N. Hedley, M. Billinghurst, L. Postner, R. May, and H. Kato. Explorations in the use of augmented reality for geographic visualization. *Presence*, 11(2) :119–133, 2001. 1.3

[HDD+92] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. In *SIGGRAPH*, pages 71–78, 1992. 6.2.3

[Hei00] John Heikkilä. Geometric camera calibration using circular control points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10) :1066–1077, October 2000. 3.2.2

[HFT+99] T. Höllerer, S. Feiner, T. Terauchi, G. Rashid, and D. Hallaway. Exploring mars : Developing indoor and outdoor user interfaces to a mobile augmented reality system. *Elsevier Publishers Computers and Graphics*, 23(6) :779–785, December 1999. 1.4.2, 5, 1, 2.4.2, 3.3.6

[HGB+02] Hong Hua, Chunyu Gao, Leonard Brown, Narendra Ahuja, and Jannick P. Rolland. Precise registration, natural occlusion and interaction in an augmented environment using a head-mounted projective display (hmpd). In *Virtual Reality (VR)*, pages 81–89, Orlando, USA, 2002. IEEE. 1.4.2, 2.8

[HGBR02] Hong Hua, Chunyu Gao, Leonard Brown, and Jannick P. Rolland. Design of an ultra-light head-mounted projective display (hmpd) and its applications in augmented collaborative environment. In *SPIE*, volume 4660, 2002. 1.4.2

[HGRT03] M. Hachet, P. Guitton, P. Reuter, and F. Tyndiuk. The cat for efficient 2d and 3d interaction as an alternative to mouse adaptations. In *Virtual Reality Software and Technology (VRST)*, pages 205–212, October 2003. 2.1.1

[HH90] P Hanrahan and P. Haeberli. Direct wysiwyg painting and texturing on 3d shapes. In *Proceedings of ACM SIGGRAPH 90*, pages 215–223, 1990. 2

[Hin03] Ken Hinckley. Synchronous gestures for multiple users and computers. In *Symposium on User Interface Software and Technology (UIST)*, 2003. 4.3

[HK03] Dieter Schmalstieg Hannes Kaufmann. Mathematics and geometry education with collaborative augmented reality. *Computers and Graphics*, 27(3), 2003. 1.3, 2.4.3

[HM97] Song Han and G&#233 ;rard Medioni. 3dsketch : modeling by digitizing with a smart 3d pen. In *Proceedings of the fifth ACM international conference on Multimedia*, pages 41–49. ACM Press, 1997. 6.2

[Hol] TAN Holobench. www.barco.com/virtualreality/. 3.4, 2.1.5

[Hol95a] Richard Holloway. Registration error analysis for augmented reality. *Presence*, 6(4), 1995. also UNC technical report TR95-001. 2, 4

[Hol95b] Richard Holloway. *Registration errors in augmented reality systems*. PhD thesis, University of North Carolina at Chapel Hill, 1995. 2

[Hou92] Stephanie Houde. Iterative design of an interface for easy 3-d direct manipulation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 135–142. ACM Press, 1992. 3.1.3

[HPF99] Tobias Hollerer, John V. Pavlik, and Steven Feiner. Situated documentaries : Embedding multimedia presentations in the real world. In *International Symposium on Wearable Computers (ISWC)*, pages 79–86, October 1999. 1.3, 3.3.6

[HPP+97] Ken Hinckley, Randy Pausch, Dennis Proffitt, James Patten, and Neal Kassell. Cooperative bimanual action. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 27–34. ACM Press, 1997. 3.1.5

[HSH+99] Ken Hinckley, Mike Sinclair, Erik Hanson, Richard Szeliski, and Matt Conway. The videomouse : a camera-based multi-degree-of-freedom input device. In *Proceedings of the 12th annual ACM symposium on User interface software and technology*, pages 103–112. ACM Press, 1999. 3.1.5

[HSS02] Nick Halper, Stefan Schlechtweg, and Thomas Strothotte. Creating non-photorealistic images the designer's way. In *International Symposium on Non Photorealistic Animation and Rendering (NPAR2002)*, pages 97–104, 2002. 4.1

[HTP+97] Ken Hinckley, Joe Tullio, Randy Pausch, Dennis Proffitt, and Neal Kassell. Usability analysis of 3d rotation techniques. In *Proceedings of the 10th annual ACM symposium on User interface software and technology*, pages 1–10. ACM Press, 1997. 2.4.1

[HZR+92] Kenneth P. Herndon, Robert C. Zeleznik, Daniel C. Robbins, D. Brookshire Conner, Scott S. Snibbe, and Andries van Dam. Interactive shadows. In *Proceedings of the 5th annual ACM symposium on User interface software and technology*, pages 1–6. ACM Press, 1992. 6.12, 3.1.3

[IBM] User-Centered Design. IBM. 6.2

[IHJ03] Milan Ikits, Charles D. Hansen, and Christopher R. Johnson. A comprehensive calibration and registration procedure for the visual haptic worbench. In *Eurographics Workshop on Virtual Environments*, pages 247–254, Zurich, Switzerland, May 2003. 3.1, 3.2.1, 3.2.4, 3.2.5

[IHMS01] K. M. Inkpen, M. Hancock, R. L. Mandryk, and S. D. Scott. Collaboration around a tabletop display : Supporting interpersonal interactions. Technical report, SFU, 2001. 2

[IKJG92] H. Ishii, M. Kobayashi, J., and Grudin. Integration of inter-personal space and shared workspace : Clearboard design and experiments. In *Computer-Supported Cooperative Work (CSCW)*, 1992. 2.3, 2.12

[Ima] RealViz ImageModeler. www.realviz.com/. 6.1

[Imma] Fakespace ImmersaDesk. http ://www.fakespace.com/. 3.4

[Immb] Fakespace ImmersaDesk2. http ://www.fakespacesystems.com/. 3.2

[IU97] H. Ishii and B Ullmer. Tangible bits : Towards seamless interfaces between people, bits and atoms. In *Computer Human Interaction (CHI)*, pages 234–241, March 1997. 1.2, 1.3, 2.5, 3.1.1

[IUC+02] H. Ishii, J. Underkoffler, D. Chak, B. Piper, E. Ben-Joseph, L. Yeung, and Z. Kanji. Augmented urban planning workbench : Overlaying drawings, physical models and digital simulation. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, Darmstadt, Germany 2002. 4.2

[IWO+99] H. Ishii, C. Wisneski, J. Orbanes, B. Chun, and J. Paradiso. Pingpongplus : Design of an athletic-tangible interface for computer-supported cooperative play. In *CHI*, Pittsburgh, USA, May 1999. 2.2

[JF02] Brad Johanson and Armando Fox. The event heap : A coordination infrastructure for interactive workspaces. In *Workshop on Mobile Computing Systems and Applications (WMCSA)*, Callicoon, USA, June 2002. 2.2.3

[JFW02] Brad Johanson, Armando Fox, and Terry Winogra. The interactive workspaces project : Experiences with ubiquitous computing rooms. *IEEE Pervasive Computing Magazine*, 1(2), 2002. 3.6, 2.2.3

[JG02] C. Matysczok J. Gausemeier, J. Fründ. Ar-planning tool - designing flexible manufacturing systems with augmented reality. In *Eurographics Workshop on Virtual Environments*, 2002. 1.3

[JHWS02] Brad Johanson, Greg Hutchins, Terry Winograd, and Maureen Stone. Pointright : Experience with flexible input redirection in interactive workspaces. In *Symposium on User Interface Software and Technology (UIST)*, 2002. 2.2.3

[Joh] Rob Johnson. http ://www.otterbein.edu/home/fac/rbrwjhns/. 2.13

[Joh91] R. Johansen. *Groupware : Computer Support for Business Teams*. Free Press, Macmillan Inc, 1991. 2.2

[JPSF01] Brad Johanson, Shankar R. Ponnekanti, Caesar Sengupta, and Armando Fox. Multibrowsing : Moving web content across multiple displays. In *Ubiquitous Computing Conference (UBICOMP)*, 2001. 2.2.3

[JS99] V. Raghavan Molineros J. and R. Sharma. Interactive evaluation of assembly sequences using augmented reality. *IEEE Transactions on Robotics and Automation*, 15(3), 1999. 1.3

[JSMM94] Robert J. K. Jacob, Linda E. Sibert, Daniel C. McFarlane, and M. Preston Mullen, Jr. Integrality and separability of input devices. *ACM Trans. Comput.-Hum. Interact.*, 1(1) :3–26, 1994. 2.4.1, 3.2.4

[Jug] VR Juggler. www.vrjuggler.org. 3.4.3

[Jul00] S. et al Julier. Battlefield augmented reality system. In *NATO Symposium on Information Processing Techniques for Military System*, Istanbul, Turkey 2000. 2.7

[JZ99] L. Joseph and R. Zeleznik. Flex and pinch : A case study of whole hand input design for virtual environmentinteraction. In *International Conference on Computer Graphics and Imaging*, 1999. 3.2.5

[KB94] G. Kurtenbach and W. Buxton. User learning and performance with marking menus. In *CHI*, pages 258–264, 1994. 2.2.3

[KB99] Hirokazu Kato and Mark Bilinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *International Workshop on Augmented Reality*, San Francisco, USA, October 1999. IEEE and ACM. 1.3, 1.4.1, 2.4.4, 2.3

[KBCW03] Kiyoshi Kiyokawa, Mark Billinghurst, B. Campbell, and E. Woods. An occlusion-capable optical see-through head mount display for supporting co-located collaboration. In *ISMAR*, 2003. 1.4.2, 2.8

[KBF+95] W. Krueger, C. Bohn, B. Frohlich, H. Scheuth, W. Strauss, and G. Wesche. The responsive workbench. *IEEE Computer*, 28(7), 1995. 3.4, 2.1.5

[KBP+00] H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto, and K. Tachibana. Virtual object manipulation on a table-top ar environment. In *International Symposium on Augmented Reality (ISAR)*, 2000. 1.3, 2.4.6, 2.2, 3.1.1, 3.4.1

[KC02] Russell Kruger and Sheelagh Carpendale. The e-table : Exploring collaborative interaction on a horizontal display. Technical report, Department of Computer Science, University of Calgary, 2002. 1, 2.2.7

[KC03] R. Kruger and S. Carpendale. Exploring orientation on a table display. Technical Report 2003-726-29, University of Calgary, 2003. 2

[kCRM91] Stuart k Card, George G. Robertson, and Jock D. Mackinlay. The information visualizer, an information workspace. In *Computer Human Interaction (CHI)*, pages 181–186, New Orleans, USA, 1991. 3.1

[KDB+02] Gudrun Klinker, Allen Dutoit, Martin Bauer, Johannes Bayer, Vinko Novak, and Dietma Matzke. Fata morgana - a presentation system for product design. In *International Symposium on Aumgented and Mixed Reality (ISMAR)*, 2002. 2.4, 1.3

[KFM+01] Daniel F. Keefe, Daniel Acevedo Feliz, Tomer Moscovich, David H. Laidlaw, and Joseph J. LaViola Jr. Cavepainting : a fully immersive 3d artistic medium and interactive experience. In *Symposium on Interactive 3D Graphics*, pages 85–93, 2001. 2

[KGH85] Myron W. Krueger, Thomas Gionfriddo, and Katrin Hinrichsen. Videoplace==an artificial reality. In *CHI*, pages 35–40, San Francisco, USA, 1985. 1.2

[KK01a] Hiroyuki Ohn Kiyoshi Kiyokawa, Yoshinori Kurata. Elmo : An enhanced optical see-through display using an lcd panel for mutual occlusion. In *International Scymposium on Mixed Reality (ISMR)*, pages 186–187, Yokohama,, 2001. 1.4.2

[KK01b] Hiroyuki Ohno Kiyoshi Kiyokawa, Yoshinori Kurata. An optical see-through display for mutual occlusion with a real-time stereo vision system. *International Journal of Systems and Applications in Computer and Graphics, Special Issue on "Mixed Realities - Beyond Conventions"*, 25(5) :765–779, 2001. 1.4.2

[Kli97] G. Klinker. Confluence of computer vision and interactive graphics for augmented reality. *Presence : Teleoperators and Virtual Environments*, 6(4), 433-451 1997. 1.4.1

[KMM+02] Robert D. Kalnins, Lee Markosian, Barbara J. Meier, Michael A. Kowalski, Joseph C. Lee, Philip L. Davidson, Matthew Webb, John F. Hughes, and Adam Finkelstein. Wysiwyg npr : drawing strokes directly on 3d models. In *Proceedings of ACM SIGGRAPH 2002*, pages 755–762, 2002. 4.3.2

[KS97] Eser Kandogan and Ben Shneiderman. Elastic windows : Evaluation of multi-window operations. In *Computer Human Interaction (CHI)*, pages 250–257, 1997. 3.1

[KSK04] Christian Kulas, Christian Sandor, and Gudrun Klinker. Towards a development methodology for augmented reality user interfaces. In *nternational Workshop on Exploring the Design and Engineering of Mixed Reality Systems (MIXER)*, 2004. 3.5

[KSR99] Gudrun Klinker, Didier Stricker, and Dirk Reiners. Optically based direct manipulation for augmented reality. *Computers and Graphics*, 23(6) :827–830, 1999. 4.2

[KTT+03] H. Kato, K. Tachibana, M. Tanabe, T. Nakajima, , and Y. Fukuda. A city-planning system based on augmented reality with a tangible interface. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, 2003. 3.1.1

[KTY99a] Kiyoshi Kiyokawa, Haruo Takemura, and Naokazu Yokoya. A collaboration supporting technique by integrating a shared virtual reality and a shared augmented reality. In *International Conference on Systems, Man and Cybernetics (SMC)*, pages 48–53, 1999. 2.4.1, 2.4.2, 3.1.2

[KTY99b] Kiyoshi Kiyokawa, Haruo Takemura, and Naokazu Yokoya. Seamlessdesign : A face-to-face collaborative virtual/augmented environment for rapid prototyping of geometrically constrained 3-d objects. In *ICMCS*, pages 447–453, 1999. 3.9, 2.4.1, 3.2.1, 3.2.2, 3.2.5

[KvL04] A.J.K. Kok and Robert van Liere. Co-location and tactile feedback for 2d widget manipulation. In *IEEE VR*, March 2004. 3.3.1

[Lam71] Butler Lampson. Protection. In *Proceedings of the 5th Annual Princeton Conference on Information Sciences and Systems*, pages 437–443, Princeton University, 1971. 3.1.4

[Lau02] Yann Laurillau. *Conception et réalisation logicielles pour les collecticiels centrées surl'atctivité de groupe : le modèle et la plate-forme Clover*. PhD thesis, Université Joseph Fourier, Grenoble, France, 2002. 2.2

[LB03] D. Larimer and D. Bowman. A framework for building a windowing interface in a virtual environment. In *INTERACT*, pages 809–812, 2003. 3.3

[LC03] Christophe Lachenal and Joelle Coutaz. A reference framework for multi-surface interaction. In *HCII*, 2003. 3.2, 4.1, 4.3

[LDR00] Céline Loscos, George Drettakis, and Luc Robert. Interactive virtual relighting of real scenes. *IEEE Transactions on Visualization and Computer Graphics*, 6(3), July-September 2000. 2, 4.2

[LDT+01] J. Leigh, G. Dawe, J. Talandis, E. He., S. Venkataraman, J. Ge, D. Sandin, and T. A. DeFanti. Agave : Access grid augmented virtual environment. In *AccessGrid Retreat*, Argonne, USA, 2001. 2.1.2

[Lev90] Pierre Levy. *Les technologies de l'intelligence*. Édition La Découverte, 1990. 2.2

[LGS00] Kent Lyons, Maribeth Gandy, and Thad Starner. Guided by voices : An audio augmented reality system. In *ntl. Conf. on Auditory Display (ICAD)*, Atlanta, USA, April 2000. 2.2

[LHS01] Joohi Lee, Gentaro Hirota, and Andrei State. Modeling real objects using video see-through augmented reality. *Presence - Teleoperators and Virtual Environments*, 11(2) :144–157, 2001. 6.1, 6.2, 2

[LJNW03] Benjamin Lok, Frederick Brooks Jr., Samir Naik, and Mary Whitton. Effects of handling real objects and self-avatar fidelity on cognitive task performance in virtual environments. In *VR*. IEEE, 2003. 1

[LJP+02] J. Leigh, A. Johnson, K. Park, A.R. Nayak, R. Singh, and V. Chowdry. Amplifed collaboration environments. In *Symposium VizGrid*, Tokyo, Japan, November 2002. 2.1.2, 3.3

[LLR+99] W. Luk, T. K. Lee, J. R. Rice, N. Shirazi, and P. Y. K. Cheung. Reconfigurable computing for augmented reality. In *Symposium on FPGAs for Custom Computing Machines*, pages 136–145, Los Alamitos, USA, 1999. 3.3.5

[Low01] Kok-Lim Low. Simulated 3d painting. TR 01–02, Department of Computer Science, University of North Carolina at Chapel Hill, 2001. 4.3.1

[LSH99] Robert W. Lindeman, John L. Sibert, and James K. Hahn. Handheld windows : Towards effective 2d interaction in immersive virtual environments. In *IEEE Virtual Reality*, pages 205–212, 1999. 2.4.1, 3.1.2

[LSR+00] Bastian Leibe, Thad Starner, William Ribarsky, Zachary Wartell, David Krum, Brad Singletary, and Larry F. Hodges. The perceptive workbench : Toward spontaneous and natural interaction in semi-immersive virtual environments. In *VR*, pages 13–20. IEEE, 2000. 2.1.5

[LVF03] V. Lepetit, L. Vacchetti, , and P. Fua. Fully automated and stable registration for augmented reality applications. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, Tokyo, Japan, Septetmber 2003. 4.5

[Mac] Blair MacIntyre. a mail on the ar mailing list : "subject : Re : [arforum] ar definition ? date : Mon, 19 may 200". 1.2

[Mac96] Wendy. Mackay. Réalité augmentée : le meilleur des deux mondes. *La Recherche, Special issue on L'ordinateur au doigt et à l'oeil*, 284, March 1996. 1.2

[MAC+02] David Margery, Bruno Arnaldi, Alin Chauffaut, Stéphane Donikian, and Thierry Duval. Openmask : (multi-threaded | modular) animation and simulation (kernel | kit) ; un bref survol. In *AFIG*, 2002. 3.5

[Mal02] G. Roth Malik, C. McDonald. Hand tracking for interactive pattern-based augmented reality. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, Darmstadt, Germany, 2002. IEEE and ACM. 3.1.1

[Man94] Steve Mann. Mediated reality. TR 260, M.I.T. Media Lab Perceptual Computing Section, 1994. 1.2, 1

[Man02] T. Manninen. Interaction forms in multiplayer desktop virtual reality games. In *Virtual Reality International Conference (VRIC)*, pages 223–232, 2002. 2.1

[Max] Discreet 3DS Max. www.discreet.com/products/3dsmax/. 2.2

[may] Alias wavefront maya. http ://www.alias.com/eng/products-services/maya/index.shtml. 2

[MB04] Jurriaan Mulder and Breght Boschker. Collaborative desktop virtual and augmented reality with a shared workspace. In *IEEE VR*, 2004. 4.7, 3.2

[MBE03] Manfred Weiler Michael Braitmaier and Thomas Ertl. Openmanip : An extensible cross-scene-graph framework for direct object manipulation. In *OpenSG*, 2003. 3.1.2

[MCN92] John Mylopoulos, Lawrence Chung, and Brian A. Nixon. Representing and using nonfunctional requirements : A process-oriented approach. *IEEE Transactions on Software Engineering*, 18(6) :483–497, 1992. 2.2

[MCW00] D. R. McGee, P. R. Cohen, and L. Wu. Something from nothing : Augmenting a paper-based work practice with multimodal interaction. In *Designing Augmented Reality Environments Conference (DARE)*, pages 71–80, Copenhagen, Denmark, April 2000. 3.1.1

[MEJ02] Blair MacIntyre, Enylton, and Simon Julier. Estimating and adapting to registration errors in augmented reality systems. In *IEEE Virtual Reality Conference*, pages 73–80, Orlando, USA, March 2002. 7

[MF96] B. MacIntyre and S. Feiner. Language-level support for exploratory programming of distributed virtual environments. In *Symposium on User Interface Software and Technology (UIST)*, pages 83–95, 1996. 2.4.2, 3.3.2

[MF01] Steve Mann and James Fung. Videoorbits on eye tap devices for deliberately diminished reality or altering the visual perception of rigid planar patches of a real world scene. In *International Symposium on Mixed Reality*, pages 48–55, March 2001. 1.2, 1

[MGB+03] Blair MacIntyre, Maribeth Gandy, Jay Bolter, Steven Dow, and Brendan Hannigan. Dart : The designer ?s augmented reality toolkit. In *International Symposium on Mixed and Augmented Reality (ISMAR). demo.*, Tokyo, Japan, October 2003. 3.3.4

[Mic] MicroOptical. http ://www.microopticalcorp.com/. 11.1

[Mil] Herman Miller. www.hermanmiller.com/. 1

[Mim] Mimio. http ://www.mimio.com/index.shtml. 2.2.4, 2.2.5

[Min93] Mark R. Mine. Characterization of end-to end delays in head-mounted display system. Technical Report TR93-001, Department of Computer Science, University of North Carolina at Chapel Hill, 1993. 2

[Min95] Mark Mine. Isaac : A virtual environment tool for the interactive construction of virtual worlds. Technical report, UNC Chapel Hill Computer Science, 1995. 2.2, 3.1.2

[MK94] P. Milgram and F. Kishino. A taxonomy of mixed reality visual displays. In *IEICE Transactions on Information and Systems (Special Issue on Networked Reality*, number E77-D(12) :1321-1329, December 1994. 1.2

[MKBP02] D. Mogilev, K. Kiyokawa, M. Billinghurst, and J. Pair. Ar pad : An interface for face-to-face ar collaboration. In *Computer Human Interaction (CHI)*, pages 654–655. ACM, 2002. 1.4.1

[Moda] Corba Component Model. www.corba.org. 3.3.3

[Modb] Photo Modeler. www.photomodeler.com/. 6.1

[MP02] Haller M. and Stampfl P. Asr - augmented sound reality. In *SIGGRAPH, Conference Abstracts and Applications*, 2002. 2.7

[MSG98a] B.A. Myers, H. Stiel, and R. Gargiulo. Collaboration using multiple pdas connected to a pc. In *Computer Supported Cooperative Work (CSCW)*, 1998. 2.2.9, 3.1

[MSG98b] Brad A. Myers, Herb Stiel, and Robert Gargiulo. Collaboration using multiple pdas connected to a pc. In *Conference on Computer supported cooperative work*, pages 285–294, Seattle, USA, 1998. ACM Press. 2.11, 2.3

[MSL+04] T.B. Moeslund, M. Störring, Y. Liu, W. Broll, I. Lindt, C. Yuan, , and M. Wittkaemper. Towards natural, intuitive and non-intrusive hci devices for roundtable meetings. In *workshop on Multi-User and Ubiquitous User Interfaces (MU3I)*, Funchal, Portugal, 2004. 2.4.5

[MSW92] R. Mander, G. Salomon, and Y. Wong. A pile metaphor for supporting casual organization of information. In *Computer Human Interaction (CHI)*, pages 627–634, April 1992. 3.1

[MT94] R. Mas and D. Thalmann. A hand control and automatic grasping system for synthetic actors. In *Eurographics*, Oslo, 1994. 2.6

[MT99] ”Erin McGarrity and Mihran Tuceryan. A method for calibrating see-through head-mounted displays for AR. In *International Workshop on Augmented Reality (IWAR)*, San Francisco, USA, 1999. 3.1

[mT+01] Erin mcGarrity, Mihran Tuceryan, , Yackup Genc, Charles Owen, and Nassir Navab. Evaluation of calibration for optical see-through augmented reality systems. In *International Symposium on Augmented Reality (ISAR)*, 2001. 2, 3.2.4, 5.9

[Mur01] Taisuke Murakami. Contact water. In *SIGGRAPH, Art Gallery*, 2001. 1.3, 2.2

[MVC+93] W.E. Mackay, G. Velay, K. Carter, C. Ma, and D. Pagani. Augmenting reality : Adding computational dimensions to paper. *Communications of the ACM*, 3(7), July 1993. 1.1

[MWLM01] Andrea H. Mason, Masuma A. Walji, Elaine J. Lee, and Christine L. MacKenzie. Reaching movements to augmented and graphic objects in virtual environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 426–433. ACM Press, 2001. 2.4.1, 3.1.3, 3.1.5

[MZDD93] P. Milgram, S. Zhai, and J. Grodski D. Drascic. Applications of augmented reality for human-robot communication. In *International Conference on Intelligent Robots and System (IROS)*, Yokohama, Japan, July 1993. 1.3

[NDRT03] Laurence Nigay, Emmanuel Dubois, Philippe Renevier, and Jocelyn Troccaz. Mixed systems : Combining physical and digital worlds. In *HCI International*, pages 1203–1207, 2003. 2.4.2

[Neta] NetMeeting. http ://www.microsoft.com/windows/netmeeting/. 2.3

[Netb] BUILD-IT Teaching Network. http ://caad.arch.ethz.ch/buildit/. 2.3.2

[NM98] U. Neumann and A. Majoros. Cognitive, performance, and systems issues for augmented reality applications in manufacturing and maintenance. In *Virtual Reality Annual International Symposium*, pages 4–11. IEEE, 1998. 3.3.6

[Nor98] D. Norman. *The Psychology of Everyday Things*. Basic Books, 1998. 2.6

[NSK01] Takahiro Nishi, Yoichi Sato, and Hideki Koike. Snaplink : Interactive object registration and recognition for augumented desk interface. In *IFIP Conference on Human-Computer Interaction (Interact)*, pages 240–24, July 2001. 3.1.1

[OBF03] A. Olwalm, H. Benko, and S. Feiner. Senseshapes : Using statistical geometry for object selection in a multimodal augmented reality system. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, 2003. 3.1.1

[OCDD01] Byong Mok Oh, Max Chen, Julie Dorsey, and Frédo Durand. Image-based modeling and photo editing. In *SIGGRAPH*, 2001. 4

[oEUSoA] Department of Energy. United States of Amercia. *Creating a Healthy Workstation environment : worksptation etgonomics made Easy*. 2.3.2, 4.3

[Off] Smith's Office. http ://www.smithsofficefurniture.com. 1

[Ope] OpenAL. www.openal.org. 2.9

[OSYT98] T. Ohshima, K. Satoh, H. Yamamoto, and H. Tamura. Ar2hockey : A case study of collaborative augmented reality. In *VRAIS*, pages 268–295. IEEE, 1998. 1.3, 2.2

[OSYT99] T. Ohshima, K. Satoh, H. Yamamoto, and H. Tamura. Rv-border guards : A multi-player entertainment in mixed reality space. In *IWAR, Poster*, 1999. 2.2

[OT96] Takashi Oishi and Susumu Tachi. Methods to calibrate projection transformation parameters for see-through head-mounted displays. *Presence*, 5(1) :122–135, 1996. 3.1

[OTX03] C. Owen, A. Tang, and F. Xiao. Imagetclar : A blended script and compiled code development system for augmented reality. In *International Workshop on Software Technology for Augmented Reality Systems (STARS)*, tokyo, Japan, October 2003. 3.3.1

[Pal97] I.S. MacKenzie R.W. Soukoreff C. Pal. A two-ball mouse affords three degrees of freedom. In *CHI*, pages 303–304, 1997. 3.1.5

[PBF02] Márcio S. Pinho, Doug A. Bowman, and Carla M.D.S. Freitas. Cooperative object manipulation in immersive virtual environments : framework and techniques. In *Proceedings of the ACM symposium on Virtual reality software and technology*, pages 171–178. ACM Press, 2002. 3.1.3

[PBWI96] Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. The go-go interaction technique : Non-linear mapping for direct manipulation in vr. In *Symposium on User Interface Software and Technology*, pages 79–80, 1996. 2.2.4

[PGT99] W. Piekarski, B. Gunther, and B. Thomas. Integrating virtual and augmented realities in an outdoor application. In *International Workshop on Augmented Reality*, pages 45–54, San Francisco, USA, October 1999. 1.4.2

[PHP+02] P. Grimm P., M. Haller, V. Paelke, S. Reinhold, C. Reimann, and J. Zauner. Amire - authoring mixed reality,. In *IEEE International Augmented Reality Toolkit Workshop*, Darmstadt, Germany, September 2002. 3.3.3, 3.4.5

[PIHP01] James Patten, Hiroshi Ishii, Jim Hines, and Gian Pangaro. Sensetable : a wireless object tracking platform for tangible user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 253–260. ACM Press, 2001. 3.1.1, 3.1.5

[Pin01] Claudio Pinhanez. Augmented reality with projected interactive displays. In *International Symposium on Virtual and Augmented Architecture*, Dublin, Ireland, 2001. 1.4.1

[Pla] Universal Plug'n Play. www.upnp.org/. 3.4.5

[Pow] Barco PowerWall. http ://www.barco.com/virtualreality/. 3.2

[PPM+03] M. Ponder, G. Papagiannakis, T. Molet, N. Magnenat-Thalmann, and D. Thalman. Vhd++ development framework : Towards extendible, component based vr/ar simulation engine featuring advanced virtual character technologies, i. In *Computer Graphics International (CGI)*, 2003. 3.3.3, 3.4.2

[PRI02] B. Piper, Ratti, and H. Ishii. Illuminating clay : A 3-d tangible interface for landscape analysis. In *Computer Human Interaction (CHI)*, Minneapolis, USA, April 2002. 2.5, 1.4.1

[pss] Pss et application à la chimie. http ://homepages.cwi.nl/ robertl/pss/pss.html. 3.2

[PT03] W. Piekarski and B. H. Thomas. An object-oriented software architecture for 3d mixed reality applications. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, 2003. 3.3.2, 4.2

[PTB+02] Ivan Poupyrev, Desney S. Tan, Mark Billinghurst, Hirokazu Kato, Holger Regenbrecht, and Tetsutani. Developing a generic augmented-reality interface. *IEEE Computer*, 35(3) :44–50, 2002. 3.10, 2.4.4, 3.1.1, 3.3.1, 3.4, 4.3, 1

[PW03] W. Pasman and C. Woodward. Implementation of an augmented reality system on a pda. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, Tokyo, Japan, October 2003. 1.4.2

[pya] Pyartk. http ://www.scripps.edu/pub/olson-web/pyartk/pyartk.html. 3.2

[PyM] PyMol. http ://pymol.sourceforge.net/. 10.3

[RAH98] Jun Rekimoto, Yuji Ayatsuka, and Kazuteru Hayashi. Augment-able reality : Situated communication through physical and digital spaces. In *International Symposium on Wearable Computing (ISWC)*, pages 68–75, 1998. 3.2, 4.3

[RCL+98] George G. Robertson, Mary Czerwinski, Kevin Larson, Daniel C. Robbins, David Thiel, and Maarten van Dantzich. Data mountain : Using spatial memory for document management. In *User Interface Software and Technology (UIST)*, pages 153–162, 1998. 3.1

[Rea] Reachin. www.reachin.se. 2.4.2

[Rek96] Jun Rekimoto. Transvision : A hand-held augmented reality system for collaborative design. In *Virtual Systems and Multi-Media (VSMM)*, 1996. 3.1.1

[RFK+97] M. Rauterberg, M. Fjeld, H. Krueger, M. Bichsel, U. Leonhardt, and M. Meier. Build-it : a video-based interaction technique of a planning tool for construction and design. In *Work With Display Units (WWDU)*, pages 175–176, 1997. 1.4.1, 3.8, 2.3.2, 3.1.1, 3.3.1

[RH94] Warren Robinett and Richard Holloway. The visual display transformation for virtual reality. *Presence*, 4(1), 1994. Also UNC technical report TR94-031. 3.2.1

[RHF94] Janick P. Rolland, Richard Holloway, and Henry Fuchs. A comparison of optical and video see-through head-mounted displays. In *SPIE Telemanipulator and Telepresence Technologies*, volume 2351, Boston, USA, October 1994. 1.4.1

[RLW01] Ramesh Raskar, Kok-Lim Low, and Greg Welch. Shader lamps : Animating real objects with image-based illumination. In *12th Eurographics Workshop on Rendering*, June 2001. 6.2

[RN95] Jun Rekimoto and Katashi Nagao. The world through the computer : Computer augmented interaction with real world environments. In *Symposium on User Interface Software and Technology*, pages 29–36, 1995. 1.3, 2.5, 1.4.1, 1

[Rod91] Tom Rodden. A survey of CSCW systems. *Interacting with Computers*, 3(3) :319–353, 1991. 2.2, 2.10

[Rom01] Odd Romell. Arthur (augmented round table for architecture and urban planning) - an augmented reality, task oriented collaboration environment. In *Conference on Applied Virtual Reality in Engineering and Construction Applications of Virtual Reality Current Initiatives and Future Challenges (AVR II and CONVR)*, October 2001. 4.2

[Rou02] Nicolas Roussel. Videoworkspace : une boîte à outils pour l'exploration de nouvelles techniques de gestion de fenêtres. In *Conférence Francophone sur l'Interaction Homme-Machine (IHM)*, pages 217–274, Poitiers, France, 2002. 3.1

[RP01] Nigay L Renevier P. Mobile collaborative augmented reality : the augmented stroll. In *EHCI*, pages 315–3, Toronto, Canada, May 2001. 1.3, 1

[RPBM03] I. Rodríguez, M. Peinado, R. Boulic, and D. Meziat. Bringing the human arm reachable space to a virtual environment for its analysis. In *IEEE Conf. on Multimedia and Expo*, 2003. 2.2

[RPF01] Miguel Ribo, Axel Pinz, and Anton L. Fuhrmann. A new Optical Tracking System for Virtual and Augmented Reality Applications. In *Proc. of IEEE Instrumentation and Measurement Technology Conference, IMTC 2001*, volume 3, pages 1932–1936, Budapest, Hungary, May 2001. 1.4.2

[RR01] John A. Robinson and Charles Robertson. The livepaper system : augmenting paper on an enhanced tabletop. *Computers Graphics*, 25(5) :731–743, 2001. 1, 3.4

[RS96] M. Rauterberg and P. Steiger. Pattern recognition as a key technology for the next generation of user interfaces. In *International Conference on Systems, Man and Cybernetics (SMC)*, volume 4, pages 2805–2810, 1996. 1.2

[RS99] Jun Rekimoto and M. Saitoh. Augmented surfaces : A spatially continuous work space for hybrid computing environment. In *CHI*, pages 378–385, 1999. 1.4.1, 2.3.1, 3.8, 2.4.2, 3.1.1

[RS00] Jun Rekimoto and Eduardo Sciammarella. Toolstone : effective use of the physical manipulation vocabularies of input devices. In *Proceedings of the 13th annual ACM symposium on User interface software and technology*, pages 109–117. ACM Press, 2000. 3.1.5, 3.3.2

[RS04] Gerhard Reitmayr and Dieter Schmalstieg. Collaborative augmented reality for outdoor navigation and information browsing. In *Symposium Location Based Services and TeleCartography*, 2004. 1, 4.2

[RUO01] Jun Rekimoto, Brygg Ullmer, and Haruo Oba. Datatiles : a modular platform for mixed physical and graphical interactions. In *Computer Human Interaction (CHI)*, pages 269–276, 2001. 3.4, 4.2

[RvBC02] Ramesh Raskar, J. van Baar, and J.X Chai. A low-cost projector mosaic with fast registration. In *Asian Conference on Computer Vision (ACCV)*, January 2002. 3.2.3

[RvDR+00] George G. Robertson, Maarten van Dantzich, Daniel C. Robbins, Mary Czerwinski, Ken Hinckley, Kirsten Risden, David Thiel, and Vadim Gorokhovsky. The task gallery : a 3d window manager. In *Computer Human Interaction (CHI)*, pages 494–501, 2000. 3.1

[RWB02] H. Regenbrecht, M. Wagner, and G. Baratoff. Magicmeeting - collaborative tangible augmented reality system. *Virtual Reality - Systems, Development and Applications*, 6(3), 2002. 3.11, 2.4.7, 3.3.3, 3.4

[RWC99] R. Raskar, G. Welch, and W. Chen. Tabletop spatially augmented reality : Bringing physical models to life using projected imagery. In *International Workshop on Augmented Reality (IWAR)*, San Francisco, USA, October 1999. 1.4.1, 4.2

[RZSP04] Dan Rosenfeld, Michael Zawadzki, Jeremi Sudol, and Ken Perlin. Physical objects as bidirectional user interface elements. *IEEE Computer Graphics and Applications*, 24(1), 2004. 2

[Sat01] M. Satyanarayanan. Pervasive computing : Vision and challenges. *IEEE Personal Communications*, August 2001. 1.2

[SATdV00] N.M. Segers, H.H. Achten, H.J.P. Timmermans, and B. de Vries. A comparison of computer-aided tools for architectural design. In *Design and Decision Support Systems*, pages 325–340, 2000. 3.1

[SBC+99] Valerie A. Summers, Kellogg S. Booth, Tom Calvert, Evan Graham, and Christine L. MacKenzie. Calibration for augmented reality experimental testbeds. In *Symposium on Interactive 3D Graphics*, pages 155–162, Atlanta, USA, April 1999. 3.1, 3.3

[SBD99] Jason Stewart, Benjamin B. Bederson, and Allison Druin. Single display groupware : A model for co-present collaboration. In *CHI*, pages 286–293, 1999. 3.7, 2.2.6

[SBF+87] M. Stefik, D. G. Bobrow, G. Foster, S. Lanning, and D. Tatar. Wysiwis revised : Early experiences with multi-user interfaces. In *Conference on Computer-Supported Cooperative Work*, pages 276–290, Austin, USA, 1987. 2.3, 2.11, 2.2.1

[Sco03] S.D. Scott. Territory-based interaction techniques for tabletop collaboration. In *Symposium on User Interface Software and Technology (UIST). Doctoral Symposium.*, pages 17–20, November 2003. 2

[SCP95] Richard Stoakley, Matthew J. Conway, and Randy Pausch. Virtual reality on a WIM : Interactive worlds in miniature. In *CHI*, 1995. 2.2.4, 3.2, 3.2.1, 3.2.5

[SDB00] Colin Swindells, John Dill, and Kellogg S. Booth. System lag tests for augmented and virtual environments. In *UIST*, pages 161–170, 2000. 4.1, 4.2

[SDM03] S. D. Scott, K. D., and R. L. Mandryk. System guidelines for co-located, collaborative work on a tabletop display. In *European Conference Computer-Supported Cooperative Work (ECSCW)*, pages 159–178, September 2003. 2

[SEG98] Zsolt Szalavári, Erik Eckstein, and Michael Gervautz. Collaborative gaming in augmented reality. In *VRST*, pages 195–204, 1998. 2.4.3, 3.1.4, 2.1, 2.2

[Sei03] H. Seichter. Collaborative augmented sketching. In *AR Toolkit Workshop*, Tokyo, Japan, 2003. 4.2

[SES99] Dieter Schmalstieg, L. Miguel Encarnacao, and Zslot Szalavari. Using transparent props for interaction with the virtual table. In *symposium on Interactive 3D graphics*, pages 147–153, Atlanta, USA, 1999. 4.2

[SFH00] D. Schmalstieg, A. Fuhrmann, and G. Hesina. Bridging multiple user interface dimensions with augmented reality. In *International Symposium on Augmented Reality (ISAR)*, Munich, Germany, October 2000. 2.7, 2.4.3, 3.4, 4.2

[SFH+02] Dieter Schmalstieg, Anton Fuhrmann, Gerd Hesina, Zsolft Szalavari, L. Miguel Encarnação, Michael Gervautz, and Werner Purgathofer. The studierstube augmented reality project. *Presence - Teleoperators and Virtual Environments*, 11(1) :33–54, February 2002. 3.9, 3.3.2, 3.2.5, 4.2

[SFSG96] D. Schmalstieg, A. Fuhrmann, Z. Szalavari, and M. Gervautz. An environment for collaboration in augmented reality. In *Collaborative Virtual Environments (CVE), Extended abstract*, Nottingham, United Kingdom, September 1996. 2.4.3, 3.1.1

[SG97a] Zsolt Szalavári and Michael Gervautz. The personal interaction panel — A two-handed interface for augmented reality. *Computer Graphics Forum*, 16(3) :C335–C346, 1997. 3.2.1, 3.3.1, 4.2, 2.4

[SG97b] Zsolt Szalavári and Michael Gervautz. The personal interaction panel - a two-handed interface for augmented reality. In *Eurographics*, pages 335–346, September 1997. 2.4.3

[SGHH94] Norbert A. Streitz, Jorg Geisler, Jorg M. Haake, and Jeroen Hol. DOLPHIN : Integrated meeting support across local and remote desktop environments and liveboards. In *Computer Supported Cooperative Work*, pages 345–358, 1994. 2.3

[SGI] Open Inventor. SGI. www.sgi.com/software/inventor. 3.1.2

[SH02] Brett Shelton and Nicholas Hedley. Using augmented reality for teaching earth-sun relationships to undergraduate geography students. In *International Augmented Reality Toolkit Workshop (ART)*, Darmstadt,Germany, 2002. 1.3

[SH03] Johan Sanneblad and Lars Erik Holmquist. Designing collaborative games on handheld computers. In *SIGGRAPH. Sketches and Applications*, 2003. 2.1, 2.2

[Sho] Microsoft Direct Show. www.microsoft.com/windows/directx/. 3.4.4

[SI01] Garth B. D. Shoemaker and Kori Inkpen. Single display privacyware : augmenting public displays with private information. In *CHI*, pages 522–529, 2001. 2.2.6

[SIS02] Steven Schkolne, Hiroshi Ishii, and Peter Schröder. Tangible + virtual ā flexible 3d interface for spatial construction applied to dna. Tr, Caltech Multi-Res Modeling Group, 2002. 2

[SKB97] Johann Schlichter, Michael Koch, and Martin Bürger. Workspace awareness for distributed teams. In *Workshop Coordination Technology for Collaborative Applications*, 1997. 2.1

[Ske] Sketchup3D. www.sketchup.com/. 4

[SKER03] C. Shen, K.M. K.M. Everitt, and K. Ryal. Ubitable : Impromptu face-to-face collaboration on horizontal interactive surfaces. In *International Conference on Ubiquitous Computing,*, October 2003. 2.2.7, 4.3

[SKH02] Jouni Smed, Timo Kaukoranta, and Harri Hakonen. A review on networking and multiplayer computer game. Technical report, Turku Centre for Computer Science, April 2002. 3.1

[SKT03] N. Sugano, H. Kato, and K. Tachi. The effects of shadow representation of virtual objects in augmented reality. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, 2003. 1

[SL92] Chris Shaw and Jiandong Liang. An experiment to characterize head motion in VR and RR using MR. In *Western Computer Graphics Symposium*, pages 99–101, 1992. 7, 2

[SLH+96] Andrei State, Mark A. Livingston, Gentaro Hirota, William F. Garrett, Mary C. Whitton, Henry Fuchs, and Etta D. Pisano (MD). Technologies for augmented-reality systems : realizing ultrasound-guided needle biopsies. In *SIGGRAPH*, pages 439–446, New Orleans, USA, August 1996. ACM. 1.3, 1.4.1

[SLM+01] C. Shen, N.B. Lesh, B. Moghaddam, P.A Beardsley, and R.S Bardsley. Personal digital historian : User interface design. In *CHI*, March 2001. 2.2.8

[SLSP00] Thad Starner, Bastian Leibe, Brad Singletary, and Jarrell Pair. Mindwarping : towards creating a compelling collaborative augmented reality game. In *Intelligent User Interfaces*, pages 256–259, 2000. 2.2

[Sma] SmartBoard. www.smarttech.com. 2.2.3, 2.2.5

[Smi96] Gareth Smith. Cooperative virtual environments : Lessons from 2d multi user interfaces. In *Computer Supported Cooperative Work*, pages 390–398, 1996. 3.1.4

[SMP01] S. Schkolne, Pruett M., and Schröder P. Surface drawing : Creating organic 3d shapes with the hand and tangible tools. In *Proceedings of CHI 2001*, 2001. 2

[SMW+02] Christian Sandor, Asa MacWilliams, Martin Wagner, Martin Bauer, and Gudrun Klinker. Sheep : The shared environment entertainment pasture. In *International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE and ACM, 2002. 1.4.1, 2.4.8, 3.12

[SPMT+02] N. A. Streitz, Th. Prante, C. Müller-Tomfelde, P. Tandler, and C. Magerkurth. Roomware : The second generation. In *International Conference on Ubiquitous Computing (UBICOMP)*, pages 77–78, Göteborg, Sweden, 2002. 3.6, 2.2.2

[SPOG01] Leonie Schäfer, Wolfgang Prinz, Amanda Oldroyd, and Lesley Gavin. Virtual storytelling of cooperative activities in a theatre of

work. In *Proceedings of the International Conference on Virtual Storytelling : Using Virtual Reality Technologies for Storytelling*, pages 191–200. Springer-Verlag, 2001. 3.1.6

[SPVT01] Hannah Slay, Matthew Phillips, Rudi Vernik, and Bruce Thomas. Interaction modes for augmented reality visualization. In Peter Eades and Tim Pattison, editors, *Australian Symposium on Information Visualisation, (invis.au 2001)*, Sydney, Australia, 2001. ACS. 3.1.1, 3.3.1, 3.2

[SR01] Dieter Schmalstieg and Gerhard Reitmayr. Open software architecture for virtual reality interaction. In *Symposium on Virtual Reality Software Technology*, 2001. 2.4.3, 3.3.5, 3.4.4, 3.5

[SRH02] Dieter Schmalstieg, Gerhard Reitmayr, and Gerd Hesina. Distributed applications for collaborative three-dimensional workspace. *Presence*, 12(1), 2002. 2.4.3

[SSI99] Imari Sato, Yoichi Sato, and Katsushi Ikeuchi. Acquiring a radiance distribution to superimpose virtual objects onto a real scene. *IEEE Transactions on Visualization and Computer Graphics*, 5(1) :1–12, 1999. 2

[SSI00] S. D. Scott, G. B. D. Shoemaker, and K. M. Inkpen. Towards seamless support of natural collaborative interactions. In *Graphics Interface*, pages 103–110, Montreal, Canada, May 2000. 2.2

[SSWF00] S. Sutphen, E. Sharlin, B.A. Watson, and John Frazer. Reviving a tangible interface affording 3d spatial interaction. In *Western Canadian Computer Graphics Symposium*, pages 155–166, Panorama, Canada, March 2000. 4.4

[STMTK01] Norbert Streitz, Peter Tandler, Christian Müller-Tomfelde, and Shin'ichi Konomi. Roomware : Towards the next generation of human-computer interaction based on an integrated design of real and virtual worlds. In J. Carroll, editor, *Human-Computer Interaction in the New Millenium"*, pages 553–578. Addison-Wesley, 2001. 3.2

[Sto] Yahoo Shop Store. shopping.yahoo.com/. 1

[Stu86] R. Stults. Media space. Technical report, Xerox PARC, 1986. 2.3

[Sut65] Yvan E. Sutherland. The ultimate display. In *IFIPS Congress*, volume 2, pages 506–508, New York, USA, May 1965. 1.3, 2.4, 1.4.1

[Sut68] Yvan E. Sutherland. A head-mounted three-dimensional display. In *AFIPS Conference*, volume 33, pages 757–764, 1968. 1.3

[SVFR04] C. Shen, Vernier, C. Forlines, and M. Ringel. Diamondspin : An extensible toolkit for around-the-table interaction. In *CHI*, April 2004. 2.2.7, 3.2

[Sys] Virtual Liver Surgery Planning System. http ://liverplanner.icg.tugraz.ac.at/. 2.4.3

[SZC+01] B. Spitzer, P. M. Zavracky, J. Crawford, P. Aquilino, and G. Hunter. Eyewear platforms for miniature displays. In *SID Symposium*, 2001. 1.4.2, 2.8

[Taba] AR Conferencing Table. http ://www.ims.tuwien.ac.at/ bara/project/. 2.4.3

[Tabb] Augmenting Reality Gaming Table. http ://www.etc.cmu.edu/projects/ar/argt.html. 2.2

[tAMHYCWK02] Introduction to Anthropometry. Myung Hwan Yun. Chul Woo Kim. In a course note. http ://www.postech.ac.kr/class/ie643/, 2002. 2.3.2

[Tan89] John C. Tang. *Listing, Drawing and Gesturing in Design : a Study on the use of Shared Workspaces by Design Teams.* PhD thesis, Stanford University, 1989. 4.2

[Tan91] J. Tang. Findings from observational studies of collaborative work. *International Journal of Man-Machine Studies*, 34(2) :143–160, 1991. 4.2

[Tan00] Peter Tandler. Architecture of beach : The software infrastructure for roomware environments. In *Workshop on Shared Environments to Support Face-to-Face Collaboration (CSCW)*, 2000. 2.6

[Tan03] P. Tandler. The beach application model and software framework for synchronous collaboration in ubiquitous computing environments. *Journal of Systems Software, Special Edition on Application Models and Programming Tools for Ubiquitous Computing*, 2003. 2.2.2

[TCD+00] Bruce Thomas, Ben Close, John Donoghue, John Squires, Phillip De Bondi, Michael Morris, and Wayne Piekarski. Arquake : An outdoor/indoor augmented reality first person application. In *ISWC*, pages 139–146, 2000. 1.3, 2.2

[TCH+03] Nicolas Tarrin, Sabine Coquillart, Shoichi Hasegawa, Laroussi Bouguila, and Makoto Sato. The stringed haptic workbench : a new haptic workbench solution. In *Eurographics*, 2003. 2.1.5

[TFK+02] Michael Tsang, George W. Fitzmaurice, Gordon Kurtenbach, Azam Khan, and Bill Buxton. Boom chameleon : simultaneous capture of 3d viewpoint, voice and gesture annotations on a spatially-aware display. In *Proceedings of the 15th annual ACM symposium on User interface software and technology*, pages 111–120. ACM Press, 2002. 4.7

[TG04] E. Tse and S. Greenberg. Rapidly prototyping single display groupware through the sdgtoolkit. In *Australasian User Interface Conference*, 2004. 2.2.6, 3.1.2

[TGW+95a] Mihran Tuceryan, Douglas S Greer, Ross T. Whitaker, David Breen, Eric Rose, Chris Crampton, and Klaus H. Ahlers. Calibration requirements and procedures for a monitor-based augmented reality system. *IEEE Transactions on Visualization and Computer Graphics*, 1(3) :255–273, September 1995. 3.2.1, 3.2.1

[TGW+95b] Mihran Tuceryan, Douglas S. Greer, Ross T. Whitaker, David E. Breen, Chris Crampton, Eric Rose, and Klaus H. Ahlers. Calibration requirements and procedures for a monitor-based augmented reality system. *IEEE Transactions on Visualization and Computer Graphics*, 1(3) :255–273, 1995. 3.2.5, 3.1.1

[TL97] J. Tsao and C. Lumsden. Crystal : Building multicontext virtual environments. *Presence*, 6(1) :55–72, 1997. 3.3

[TM02] Peter Tandler and C. Magerkurth. Augmenting tabletop design for computer-supported cooperative work. In *Workshop on Co-located Tabletop Collaboration : Technologies and Directions at CSCW'02*, 2002. 2.3.2

[TM03] T. Kujanpaa T and T Manninen. Non-verbal communication in computer game avatars. In *Digital Games Research Conference*, pages 220–233, 2003. 2.1

[TN00] Mihran Tuceryan and Nassir Navab. Optical see-through hmd calibration : A novel stereo method validated with a video see-through system. In *International Symposium on Augmented Reality (ISAR)*, pages 165–174, Munich,Germany, October 2000. 3.2.4

[TP02] B. H. Thomas and W. Piekarski. Glove based user interaction techniques for augmented reality in an outdoor environment. *Virtual Reality : Research, Development, and Applications*, 6(3), 2002. 3.1.1

[TPG99] H. Thomas, W. Piekarski, and B. Gunther. Using augmented reality to visualise architecture designs in an outdoor environment. *International Journal of Design Computing Special Issue on Design Computing on the Net (DCNet)*, 1999. 1.3, 4.2

[TPMT+01] P. Tandler, Th. Prante, C. Müller-Tomfelde, N. A. Streitz, and R. Steinmetz. Connectables : Dynamic coupling of displays for the flexible creation of shared workspaces. In *Symposium on User Interface Software and Technology (UIST)*, pages 11–20, 2001. 2.2.2, 4.3

[TRC+93] Russell M. Taylor, Warren Robinett, Vernon L. Chi, Frederick P. Jr. Brooks, William V. Wright, R. Stanley Williams, and Erik J. Snyder. The nanomanipulator : a virtual-reality interface for a scanning tunneling microscope. In *SIGGRAPH*, pages 127–134, 1993. 3.2

[Tri00] Anish Tripathi. Augmented reality : An application for architecture. Master's thesis, Faculty of the School of Architecture, University, of Southern California, 2000. 4.2

[Tro] Qt Toolkit. Trolltech. www.trolltech.org. 3.3.2

[Tsi98] Nicolas Tsingos. *Simulation de champs sonores de haute qualité pour des applications graphiques interactives*. PhD thesis, Université Joseph Fourier (Grenoble), Dec 1998. 2.7

[TZO03] Arthur Tang, Ji Zhou, and Charles Owen. Evaluation of calibration methods for optical see-through head-mounted displays. In *International Symposium on Mixed and Augmented Reality (ISMAR 2003)*, October 2003. 3.2.4

[UI98] John Underkoffler and Hiroshi Ishii. Illuminating light : An optical design tool with a luminous-tangible interface. In *CHI*, pages 542–549, 1998. 1.3

[UI99] J. Underkoffler and H. Ishii. Urp : A luminous-tangible workbench for urban planning and design. In *CHI*, pages 386–393, Pittsburgh, USA, May 1999. 3.1.1, 4.2

[Ulb03] Christiane Ulbricht. Tangible augmented reality for computer games. Master's thesis, Université Technique de Vienne, 2003. 2.4.3, 3.1.1

[Ull02] Brygg Ullmer. *Tangible Interfaces for Manipulating Aggregates of Digital Information*. PhD thesis, MIT Media Laboratory, 2002. 3.1.1

[UTS+02] Shinji Uchiyama, Kazuki Takemoto, Kiyohide Satoh, Hiroyuki Yamamoto, and Hideyuki Tamura. Mr platform : A basic body on which mixed reality applications are built. In *International Symposium on Mixed and Augmented Reality (ISMAR)*, 2002. 3.3.1

[UYKT97] S. Uchiyama, H. Yamamoto, A. Katayama, and H. Tamura. Presentation and interaction of virtual 3d objects without geometric model. In *HCI International*, volume 2, pages 869–872, 1997. 2.2

[VD] Seos V-DOME. http ://www.seos.com/dome.htm. 2.1.1

[Viz] 3D Studio Viz. www2.discreet.com/products/. 10.5, 4.3

[VKL+02] S. Veigl, A. Kaltenbach, F. Ledermann, G. Reitmayr, and D. Schmalstieg. Two-handed direct interaction with artoolkit. In *International Workshop on ARToolKit*. IEEE, 2002. 3.2.5, 3.1.1, 2.4

[vLM03] Robert van Liere and Jurriaan D. Mulder. Optical tracking using projective invariant marker pattern properties. In *Virtual Reality*, Los Angeles, USA, March 2003. IEEE. 1.4.2

[VLS02] F. Vernier, N.B Lesh, and C. Shen. Visualization techniques for circular tabletop interfaces. In *Advanced Visual Interfaces (AVI)*, 2002. 2.2.8, 3.1.3

[VMD] VMD. http ://www.ks.uiuc.edu/research/vmd/. 3.2

[VNC] VNC. http ://www.realvnc.com/. 2.1.2

[VPNF98] E. Viirre, H. Pryor, S. Nagata, and T. A. Furness. The virtual retinal display : A new technology for virtual reality and augmented vision in medicin. In *Medicine Meets Virtual Reality*, pages 252–257, 1998. 1.4.2

[W3Ca] HTML. W3C. http ://www.w3.org/. 4.1

[W3Cb] SMIL. W3C. http ://www.w3.org/. 3.3.6, 4.1, 5

[Wag03] Daniel Wagner. First steps towards handheld augmented reality. In *ISWC*, 2003. 1.4.2

[WB03] Michael Wu and Ravin Balakrishnan. Multi-finger and whole hand gestural interaction techniques for multi-user tabletop displays. In *Symposium on User Interface Software Technology (UIST)*, pages 193–202, 2003. 2.2.7, 3.1.1, 3.1.3

[WBV+01] Greg Welch, Gary Bishop, Leandra Vicci, Stephen Brumback, Kurtis Keller, and D'nardo Colucci. High-performance wide-area tracking -the hiball tracking system. *Presence : Teleoperators and Virtual Environments*, 2001. 1.4.1

[WC03] L. K. Wong and Y. S. Chee. Constructivist physics lerning in an immersive hot air balloon simulation program (ihabs). In *SIGGRAPH. Educators Program*, 2003. 3.5, 2.1.6

[Weg03] Suzanne Weghorst. Augmented tangible molecular models. In *International Workshop on Spatial Media*, 1003. 3.2

[Wei91] Mark Weiser. The computer for the twenty-first century. *Scientific American*, pages 94–10, 1991. 1.2

[Wel91] Pierre Wellner. The digitaldesk calculator : tangible manipulation on a desk top display. In *Symposium on User interface software and technology*, pages 27–33, Hilton Head, USA, 1991. ACM Press. 1.1, 1.3, 2.5, 1

[WFM+96] A. Webster, Steve Feiner, Blair MacIntyre, W. Massie, and T. Krueger. Augmented reality in architectural construction, inspection and renovation. In *Congress on Computing in Civil Engineering*, pages 913–919, Anaheim, USA, June 1996. ASCE. 1.3, 4.2

[Wic02] R. Wichert. Collaborative gaming in a mobile augmented reality environmen. In *EUROGRAPHICS Ibero-American Symposium in Computer Graphics*, pages 31–37, Guimarães, Portugal, 2002. 2.2

[Wil91] P. Wilson. *Computer Supported Cooperative Work*. UK : Intellect Books, 1991. 2.2

[WIM] WIMP. http ://www.cc.gatech.edu/classes/cs6751_97_winter/topics/dialog-wimp/. 4

[WJH+03] W. Hartmann W, J. Zauner J, M. Haller, T. Luckeneder, and W. Woess. Shadow catcher : A vision based illumination condition sensor using artoolkit. In *International Augmented Reality Toolkit Workshop*, Tokyo, Japan, October 2003. IEEE. 2

[WM00] Yanqing Wang and Christine L. MacKenzie. The role of contextual haptic and visual constraints on object manipulation in virtual environments. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 532–539. ACM Press, 2000. 2.4.1, 3.1.2

[WMSB98] Yanqing Wang, Christine L. MacKenzie, Valerie A. Summers, and Kellogg S. Booth. The structure of object transportation and orientation in human-computer interaction. In *CHI*, pages 312–319, 1998. 2.6

[WO90] C. Ware and S. Osborne. Exploration and virtual camera control in virtual three dimensional environments. In *Symposium on Interactive 3D Graphics*, pages 175–183, 1990. 3.2.3, 3.2.4

[Wora] How Stuff Works ? How Holographic Environment Will Work. http ://computer.howstuffworks.com/holographic-environment.htm. 11.2

[Worb] Haptic Workstation. Immersion. 2003. 1.1

[YLO02] Jeng-Sheng Yeh, Ting-Yu Lien, and Ming Ouhyoung. On the effects of haptic display in brush and ink simulation for chinese painting and calligraphy. In *Proceedings of Pacific Graphics 2002 (PG2002)*, pages 439–441, October 2002. 2

[ZFS97] Robert C. Zeleznik, Andrew S. Forsberg, and Paul S. Strauss. Two pointer input for 3d interaction. In *Symposium on Interactive 3D Graphics*, pages 115–120, 192, 1997. 3.1.5

[ZM94] Shumin Zhai and Paul Milgram. Input techniques for hci in 3d environments. In *Conference companion on Human factors in computing systems*, pages 85–86. ACM Press, 1994. 2.4.1

[ZM98] Shumin Zhai and Paul Milgram. Quantifying coordination in multiple dof movement and its application to evaluating 6 dof input devices. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 320–327. ACM Press/Addison-Wesley Publishing Co., 1998. 2.4.1, 6.3