HAL
open science

Nouvelles approches pour la portabilité, la non-intrusivité et l'accessibilité des interfaces de manipulation 3D

Paul Issartel

To cite this version:

Paul Issartel. Nouvelles approches pour la portabilité, la non-intrusivité et l'accessibilité des interfaces de manipulation 3D. Interface homme-machine [cs.HC]. Université Paris Saclay (COmUE), 2017. Français. NNT : 2017SACLS077 . tel-01515696

HAL Id: tel-01515696

https://theses.hal.science/tel-01515696v1

Submitted on 28 Apr 2017

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.

NNT : 2017SACLS077

Thèse de doctorat
de l'Université Paris-Saclay
préparée à l'Université Paris-Sud

École doctorale n° 580
Sciences et technologies de l'information et de la communication (STIC)
Spécialité de doctorat : Informatique

par

M. Paul Issartel

Nouvelles approches pour la portabilité, la non-intrusivité et l'accessibilité des interfaces de manipulation 3D

Thèse présentée et soutenue à Orsay, le 3 avril 2017.

Composition du Jury :

M.	Tobias Isenberg	Directeur de recherche Inria Saclay	(Président du jury)
M.	Martin Hachet	Chercheur Inria Inria Bordeaux	(Rapporteur)
M.	Anatole Lécuyer	Directeur de recherche Inria Rennes	(Rapporteur)
Mme	Sabine Coquillart	Directrice de recherche Inria Rhône-Alpes	(Examinatrice)
M.	Gilles Bailly	Chargé de recherche ISIR	(Examinateur)
M.	Mehdi Ammi	Maître de conférences HDR LIMSI-CNRS	(Directeur de thèse)
M.	Florimond Guéniat	Docteur Université de l'Illinois	(Co-encadrant)

Sommaire

Introduction générale **4**
1 Objectifs de la thèse 7
2 Contributions 7

1 Définitions **9**
1.1 Manipulation 3D 9
1.2 Efficacité 10
1.3 Portabilité 13
1.4 Non-intrusivité 14
1.5 Accessibilité 15

2 Systèmes existants **17**
2.1 Ordinateur personnel 17
2.2 Appareil mobile 22
2.3 Installations fixes de réalité virtuelle 25
2.4 Systèmes de réalité virtuelle portés par l'utilisateur 31
2.5 Conclusion 35

3 Manipulation par objets tangibles sur appareil mobile **37**
3.1 Introduction 37
3.2 Travaux existants 38
3.3 Concept 40
3.4 Implémentation 41
3.5 Application à l'exploration de données 42
3.6 Première étude : manipulation 3D 49
3.7 Deuxième étude : plans de coupe 56
3.8 Conclusion 62

4 Perception de forces en l'absence de dispositifs actifs **64**
4.1 Introduction 64
4.2 Travaux existants 66
4.3 Concept 69

4.4 Implémentation 71
4.5 Première étude : tri d'objets par masse 73
4.6 Deuxième étude : seuil différentiel 78
4.7 Conclusion 82

5 Appareil mobile comme support de manipulation 83
5.1 Introduction 83
5.2 Concept 84
5.3 Implémentation 86
5.4 Travaux existants 88
5.5 Formalisation des principaux *mappings* 92
5.6 Propriétés de compatibilité spatiale 98
5.7 Référentiel allocentrique ou égocentrique 106
5.8 Espace de manipulation 122
5.9 Conclusion 130

6 Le « volume tangible » : un dispositif portable et intégré pour une manipulation co-localisée 132
6.1 Introduction 132
6.2 Travaux existants 134
6.3 Concept 137
6.4 Implémentation 141
6.5 Première étude : sélection et manipulation 152
6.6 Deuxième étude : champ de vision 155
6.7 Conclusion 160

Conclusion générale 161

Annexes 166

Introduction

Les environnements virtuels 3D sont aujourd'hui au cœur de nombreux domaines d'activité. Dans un contexte scientifique ou médical, ils servent à visualiser et explorer des données volumiques complexes. Dans l'industrie et les services, le prototypage virtuel permet de concevoir des produits puis de les évaluer avant même de débuter la fabrication. Les environnements 3D sont également devenus incontournables dans le secteur du jeu vidéo. Toutes ces activités ne pourraient exister sans l'outil informatique et les interfaces homme-machine associées. En effet, au-delà de la seule visualisation, l'interface permet à l'utilisateur *d'agir* sur l'environnement virtuel et d'en observer immédiatement les effets.

Un aspect fondamental de cette interaction est la *manipulation 3D*, c'est-à-dire la sélection, le positionnement, et la rotation d'objets dans un espace tri-dimensionnel [Bowman *et al.* 2004]. Compte tenu du caractère fondamental de cette tâche, une interface destinée à interagir avec des objets virtuels 3D se doit d'être bien adaptée à la manipulation 3D.

À l'heure actuelle, la grande majorité de ces interfaces sont conçues pour une même plate-forme : l'ordinateur personnel (PC). Parmi les principaux logiciels de visualisation scientifique, ou les logiciels de conception et de modélisation, il est bien difficile d'identifier un exemple notable qui ne soit pas principalement destiné à être utilisé sur PC. Les jeux vidéo, un autre exemple majeur d'interaction avec des objets 3D, sont eux aussi largement présents sur cette plate-forme. En dehors du PC, on constate également une récente progression des appareils mobiles (smartphones, tablettes...) avec le portage de certains logiciels de conception et un nombre croissant de jeux 3D.

Cependant, tous ces logiciels sont soumis aux limitations de leur plate-forme. Ainsi, les interfaces utilisateur destinées au PC sont conçues selon le mode d'interaction standard pour cette plate-forme : clavier et souris. Mais ces dispositifs, qui ont fait leurs preuves pour l'interaction 2D, montrent leurs limites lorsqu'il s'agit d'interagir avec des objets 3D. La souris 2D (et ses équivalents : pavé tactile, *trackball*, *trackpoint*...) permet à l'utilisateur de contrôler deux degrés de liberté (DdL) simultanément. Le clavier n'offre lui qu'un contrôle binaire sur chaque touche. Il nécessite donc deux touches pour contrôler un seul degré de liberté, comme le montre l'exemple des touches directionnelles. Or, la manipulation 3D est une tâche qui requiert jusqu'à 6 degrés de liberté : outre la sélection, le positionnement d'un objet sur chaque axe nécessite 3 DdL et la rotation autour de ces mêmes axes néces-

site également 3 DdL. Il s'ensuit que la souris et le clavier sont inadaptés pour contrôler entièrement une tâche de manipulation 3D. La solution généralement retenue est de décomposer les déplacements en plusieurs sous-étapes, ce qui complique singulièrement l'interface utilisateur.

Les appareils mobiles, eux, disposent habituellement d'un écran tactile comme dispositif d'entrée. Un contact sur l'écran fournit deux degrés de liberté. Si beaucoup d'écrans tactiles sont multipoints, il n'en faut pas moins un minimum de trois doigts pour contrôler les 6 DdL que peut nécessiter une tâche de manipulation 3D. L'utilisation de trois doigts pose des problèmes ergonomiques sur un appareil mobile, et aboutit à des techniques d'interaction complexes et difficiles à maîtriser. De plus, les 3 DdL de positionnement et les 3 DdL de rotation se prêtent mal à une décomposition en sous-ensembles de 2 DdL contrôlés séparément. Toutes ces limitations se répercutent ici encore sur l'usabilité de l'interface.

On constate donc que ces deux plates-formes (PC et mobile), bien que majoritaires, sont peu adaptées à la manipulation 3D. Il existe pourtant d'autres approches, spécifiquement conçues pour interagir avec des environnements 3D. Les systèmes basés sur la réalité virtuelle en sont les meilleurs exemples. En plaçant l'interaction dans un espace lui-même en 3D, la réalité virtuelle facilite la manipulation d'objets. Ainsi, les installations de type « salle immersive » consistent en un espace semi-clos entouré d'une surface d'affichage et dont le volume intérieur est entièrement couvert par un système de suivi 6 DdL. L'interaction avec l'environnement virtuel se fait habituellement au travers de dispositifs appelés « wand », qui sont tenus dans la main et reconnus par le système de suivi afin de fournir un contrôle 6 DdL pleinement adapté à la manipulation 3D. L'interaction peut également se faire par le biais d'un bras robotisé, cette dernière solution étant en mesure de fournir un retour de force. Une approche différente consiste à équiper l'utilisateur d'un visiocasque (*head-mounted display*, HMD). Le casque produit l'illusion de se trouver au sein d'un espace virtuel, avec lequel il devient possible d'interagir directement. Pour cela, l'utilisateur peut notamment être équipé de gants instrumentés (*data gloves*) lui permettant de manipuler des objets virtuels avec la main.

Les approches ci-dessus ne sont pas des concepts nouveaux. La première salle immersive a été construite en 1993, tandis que le premier HMD remonte à 1968. Tous deux ont été largement exploités depuis lors aussi bien par la recherche académique que par l'industrie. De nos jours, ce type de matériel est même directement commercialisé. Pourtant, malgré l'existence de ces équipements, la majorité des tâches courantes de manipulation 3D sont encore réalisées sur des interfaces PC. On peut alors se demander pourquoi les approches telles que celles mentionnées ci-dessus, malgré leurs avantages évidents, demeurent sous-utilisées en pratique.

Plusieurs raisons peuvent être avancées pour expliquer cette faible utilisation. Une première raison est l'encombrement et l'absence de portabilité de certaines approches. Une salle immersive nécessite des installations fixes et volumineuses, tant pour l'affichage que pour le système de suivi. Il n'est donc pas envisageable d'en équiper chaque poste de travail, et

un laboratoire ou une entreprise ne dispose souvent au mieux que d'une seule installation de ce type. Par conséquent, les utilisateurs potentiels sont contraints de se rendre dans un lieu dédié pour s'en servir, et doivent se coordonner pour en partager l'usage. À cause du manque de portabilité, il n'est pas non plus possible d'en poursuivre l'utilisation chez soi ou en déplacement. L'encombrement et l'absence de portabilité présentent donc des obstacles à l'utilisation de ces systèmes au quotidien.

Une deuxième raison est l'intrusivité de certains dispositifs. Une interface basée sur un visiocasque et un gant instrumenté est théoriquement portable. Elle ne pose donc pas les difficultés décrites ci-dessus. Cependant, le visiocasque ainsi que le gant doivent impérativement être portés avant de pouvoir utiliser l'interface, et doivent ensuite être retirés. Ces dispositifs présentent aussi l'inconvénient de couper l'utilisateur du monde réel : une coupure visuelle dans le cas d'un visiocasque, et tactile dans le cas d'un gant. À l'inverse, les salles immersives sont immédiatement utilisables en pénétrant simplement dans l'espace concerné et en manipulant un « wand » à main nue, et ne coupent pas l'utilisateur du monde réel et des personnes environnantes. Ainsi, l'intrusivité de dispositifs tels qu'un visiocasque ou un gant instrumenté présente elle aussi des obstacles à une utilisation courante.

Une troisième raison est liée à l'accessibilité des systèmes eux-mêmes. Le très large déploiement de plates-formes comme le PC, et plus récemment des appareils mobiles, peut expliquer pourquoi les utilisateurs continuent à se servir majoritairement de ces systèmes malgré leurs limitations. En effet, la plupart des utilisateurs disposent personnellement de tels systèmes sur leur lieu de travail, ainsi qu'à leur domicile, et peuvent raisonnablement s'attendre à en trouver lors d'un déplacement. Confrontés à des systèmes potentiellement efficaces mais localisés et non portables, et à des systèmes portables mais intrusifs et contraignants, il est compréhensible que les utilisateurs choisissent de conserver leurs systèmes habituels malgré une moindre performance pour la manipulation 3D. Une conséquence est que les interfaces dotées des fonctionnalités les plus avancées (logiciels phares, logiciels métiers...) sont en premier lieu développées pour les plates-formes les plus répandues telles que le PC, renforçant la dépendance des utilisateurs pour des systèmes peu adaptés à ce type d'interaction.

Ces différentes contraintes (absence de portabilité, intrusivité et faible accessibilité) mènent à une situation aujourd'hui insatisfaisante : les utilisateurs qui pourraient bénéficier d'une interface efficace pour la manipulation 3D ne s'en servent pas réellement pour leurs tâches courantes. Bien que de nouveaux systèmes interactifs de plus en plus perfectionnés ne cessent d'apparaître, les trois aspects ci-dessus semblent trop souvent ignorés, ce qui conduit en pratique à une sous-utilisation de ces systèmes. Il existe donc un besoin pour de nouvelles formes d'interfaces, permettant non seulement une manipulation 3D efficace mais prenant également en compte les critères ci-dessus, afin de favoriser leur adoption par les utilisateurs concernés.

1 Objectifs de la thèse

Cette thèse vise à apporter des réponses au besoin identifié ci-dessus. Le premier axe de ce travail consiste ainsi à *proposer et étudier de nouvelles approches pour augmenter la portabilité, la non-intrusivité et l'accessibilité* de la manipulation 3D, tout en continuant de permettre une manipulation efficace. Plusieurs formes d'interfaces sont successivement proposées, chacune mettant un accent différent sur ces quatre aspects. Ces approches sont ensuite évaluées de manière analytique et expérimentale afin de vérifier leur adéquation avec le but recherché.

La poursuite des objectifs de portabilité, de non-intrusivité et d'accessibilité, souvent laissés de côté dans les interfaces existantes, conduit à l'apparition de nouveaux défis. Comment réaliser un suivi 3D fiable sans recourir à des dispositifs externes ? Est-il possible de générer un retour de force en se passant d'équipements intrusifs et/ou onéreux ? Une surface d'affichage de taille réduite, intégrée dans un système portable, représente-t-elle un inconvénient pour la manipulation ? Le deuxième axe majeur de cette thèse vise donc à *apporter des réponses aux difficultés spécifiques* que posent ces objectifs dans la conception d'une interface de manipulation 3D.

2 Contributions

Nous présentons tout d'abord, dans le chapitre 1, une définition détaillée des principaux points autour desquels s'organise cette thèse : la manipulation 3D et son efficacité, la portabilité, la non-intrusivité et l'accessibilité. Nous examinons ensuite dans le chapitre 2 les principales catégories de systèmes existants afin de déterminer comment ces derniers répondent aux critères que nous nous sommes fixés. Il ressort de cette analyse qu'aucun des principaux systèmes d'interaction ne répond simultanément à tous ces objectifs, mais que leurs avantages et inconvénients sont complémentaires.

Sur la base de ce constat, nous proposons dans le chapitre 3 une première approche pour obtenir une interface à la fois portable, non-intrusive, accessible et efficace pour la manipulation 3D. Celle-ci consiste à combiner un appareil mobile classique (pour l'accessibilité et la portabilité) avec des objets tangibles passifs (pour une manipulation 3D efficace et non-intrusive). Nous présentons une implémentation de ce concept dans un contexte d'exploration de données volumiques, avant d'étudier son usabilité à travers deux études expérimentales.

Cette configuration basée sur des objets tangibles passifs, dans un système visant à être portable, non-intrusif et accessible rend particulièrement difficile l'intégration de dispositifs à retour de force dans l'interface. L'absence de retour de force constitue un désavantage

en terme de manipulation par rapport aux systèmes existants. Nous proposons donc dans le chapitre 4 une technique basée sur le principe du « pseudo-haptique », visant à recréer une perception de force par un effet purement visuel. Notre technique permet ainsi de se passer de dispositifs actifs et des contraintes qui leur sont associées. Nous évaluons expérimentalement l'efficacité de cette technique pour la perception de masses virtuelles.

L'utilisation d'objets tangibles en combinaison avec un appareil mobile présente un certain nombre d'avantages, mais oblige l'utilisateur à toujours conserver et transporter ensemble les différents objets qui constituent l'interface. Une manière d'en augmenter encore la portabilité est de se servir de *l'appareil mobile* comme objet tangible, en déplaçant l'appareil mobile lui-même dans l'espace 3D pour manipuler des objets affichés sur son propre écran. Cette approche, dans laquelle un même dispositif sert à la fois de surface d'affichage et de support de manipulation, permet de se passer des objets tangibles externes. Cependant, elle présente également certaines particularités pour la manipulation. Dans le chapitre 5, nous examinons en profondeur les différentes questions que pose cette configuration spécifique, afin de déterminer une fonction de transfert adaptée entre les mouvements de l'appareil mobile et ceux des objets manipulés.

Une des principales difficultés de cette dernière approche est due à la distance de séparation entre les objets 3D et l'appareil mobile. Dans le chapitre 6, nous proposons donc le concept de « volume tangible », reposant là encore sur un dispositif unique pour l'affichage et la manipulation mais permettant cette fois de positionner ce dernier *sur* les objets à manipuler. Cette solution requiert cependant un nouveau type d'appareil mobile, ce qui la rend moins accessible que l'approche précédente, mais autorise en contrepartie une manipulation plus directe et plus efficace tout en restant portable et non-intrusive. Nous examinons dans un premier temps les aspects techniques de faisabilité d'un tel dispositif. Nous présentons ensuite une implémentation partielle de ce concept, que nous utilisons pour en étudier plusieurs critères d'usabilité.

Chapitre 1

Définitions

Dans ce chapitre, nous commençons par définir en détail les points essentiels qui seront abordés dans cette thèse : la *manipulation 3D* et les aspects pouvant influencer son *efficacité*, la *portabilité*, la *non-intrusivité* et enfin *l'accessibilité*.

1.1 Manipulation 3D

L'interaction 3D peut prendre de nombreuses formes, en fonction de l'application pour laquelle est conçue l'interface et de l'objectif visé par l'utilisateur. Pour être en mesure d'étudier et d'améliorer les systèmes d'interaction 3D, il est nécessaire de catégoriser ces différentes formes d'interaction en *tâches universelles*. Une solution développée pour répondre à l'une de ces tâches universelles peut ensuite profiter à des interfaces plus spécialisées.

Bowman *et al.* [2004] identifient cinq tâches universelles dans les interfaces 3D : la sélection, la manipulation, la navigation, le contrôle d'application et l'entrée symbolique.

— **Sélection :** action de définir sur quel objet virtuel appliquer la ou les action(s) suivante(s), notamment la manipulation ;
— **Manipulation :** action de positionner et d'orienter un objet virtuel dans l'espace 3D, généralement après l'avoir sélectionné ;
— **Navigation :** déplacement du point de vue de l'utilisateur dans l'environnement virtuel ;
— **Contrôle d'application :** changement de l'état du système par le biais de commandes ;
— **Entrée symbolique :** entrée d'informations telles que du texte dans l'interface.

Dans cette thèse, nous choisissons de nous focaliser sur la **manipulation**. En effet, cette tâche est l'une des plus fondamentales et des plus importantes dans le cas de l'interaction avec un environnement virtuel 3D. Un grand nombre d'actions effectuées dans un tel environnement, y compris des opérations de navigation ou de contrôle d'application, peuvent être assimilées à de la manipulation [Bowman *et al.* 2004].

En pratique, les tâches universelles de manipulation et de sélection sont intimement liées. La sélection est indissociable de la manipulation dès lors que l'environnement virtuel contient plusieurs objets. Dans la suite de ce travail, nous regroupons donc ces deux aspects sous un même terme :

> **Manipulation 3D :** sélection, positionnement et rotation d'objets virtuels dans un environnement tridimensionnel.

Le terme *positionnement* désigne le fait de modifier la position de l'objet manipulé dans l'espace 3D, c'est-à-dire ses coordonnées spatiales suivant les axes X, Y et Z de l'environnement virtuel. Le terme *rotation* désigne quant à lui le fait de modifier l'orientation de l'objet dans ce même espace, c'est-à-dire l'angle de rotation de l'objet autour des axes X, Y et Z.

On remarquera que cette formulation n'inclut pas explicitement d'autres formes d'interaction telles que la déformation et la mise à l'échelle, souvent considérées comme de la « manipulation » tant par certains auteurs [Mine 1995] que dans plusieurs logiciels de modélisation. Mais comme le notent Bowman *et al.* [2004], la déformation ou la mise à l'échelle sont généralement réalisées en *manipulant* des contrôles 3D [Bier 1987; Jankowski et Hachet 2013]. Une interface de « manipulation 3D » supporte donc indirectement ce type d'interaction. Nous pouvons ainsi compléter la définition ci-dessus en précisant que la manipulation 3D consiste à sélectionner, positionner et orienter des objets *rigides* dans un environnement virtuel.

1.2 Efficacité

Pour constituer un progrès par rapport aux systèmes existants, une interface se doit en premier lieu d'être *efficace* pour des tâches de manipulation 3D. Au sens strict, « l'efficacité » désigne la capacité « d'aboutir à un résultat utile » [Larousse 2016]. En l'occurrence, l'interface doit donc *au minimum* permettre à l'utilisateur de mener à bien une tâche de manipulation 3D. Ce seul critère n'est cependant pas suffisant. Une interface permettant de manipuler des objets 3D mais seulement au prix d'efforts considérables pourrait être qualifiée « d'efficace », tout en ayant peu de chances d'être adoptée en présence d'alternatives.

Nous préférerons donc employer le terme « efficacité » dans son sens plus courant : celui de « produire le maximum de résultats avec le minimum d'efforts » [Larousse 2016] (on rencontre parfois l'anglicisme « efficience » pour désigner ce rapport entre la qualité du résultat obtenu et l'effort entrepris). Une interface est ainsi efficace si elle tire suffisamment parti des *capacités de l'utilisateur*, de manière à limiter l'effort demandé. Cette définition présente l'intérêt de pouvoir comparer différentes interfaces selon leur *degré* d'efficacité. Plus

une interface tire parti des capacités de l'utilisateur, plus elle peut être considérée comme « efficace » pour la manipulation 3D.

Le degré selon lequel une interface exploite les capacités de l'utilisateur dépend de nombreux critères. En établir une liste exhaustive pourrait constituer un véritable axe de recherche. Nous choisissons ici de nous intéresser aux critères suivants, pouvant être quantifiés de manière objective :

— **Degrés de liberté :** à quel point l'interface tire parti des dimensions de mouvement dont dispose l'utilisateur pour contrôler les dimensions de mouvement de l'objet manipulé. Une capacité insuffisante à exploiter ces degrés de liberté oblige à décomposer la manipulation en plusieurs étapes, ce qui réduit l'efficacité de l'interface [Bowman *et al.* 2004].

— **Qualité de la trajectoire :** à quel point l'interface limite les mouvements superflus par rapport à la tâche de manipulation à effectuer [Zhai et Milgram 1998]. Une trajectoire plus directe, par définition plus efficace, permet notamment de retarder la survenue d'une fatigue chez l'utilisateur.

— **Retour d'information :** à quel point l'interface exploite les canaux de retour en direction de l'utilisateur. Un retour insuffisant (tel qu'une interruption fréquente de l'affichage) nuit à l'efficacité, tandis qu'un retour plus riche (tel qu'un retour de force en sus du retour visuel) exploite mieux les capacités de l'utilisateur [Marchal 2014].

Parmi les critères ci-dessus, la question des degrés de liberté revêt une importance particulière. En effet, comme nous allons le voir par la suite, celle-ci constitue la principale limitation des systèmes actuellement dominants. D'autres systèmes, répondant mieux à ce critère, présentent quant à eux l'inconvénient d'être non portables, intrusifs et/ou peu accessibles. Compte tenu de son importance dans la suite de cette thèse, il paraît nécessaire de définir cet aspect plus en détail.

Degrés de liberté (DdL)

L'emplacement (position et orientation) d'un objet rigide dans un espace 3D est défini par exactement six valeurs : trois valeurs de position suivant les trois dimensions de l'espace, et trois valeurs d'orientation autour de ces mêmes dimensions. Déplacer un tel objet, ce qui est l'objectif de la manipulation 3D, revient à changer ces valeurs pour l'amener à un nouvel emplacement. En l'absence de contrainte spécifique, ces six valeurs sont librement et indépendamment modifiables. Cette opération possède donc six paramètres indépendants, que l'on appelle « degrés de liberté » (DdL, ou « DOF » pour *« degrees of freedom »*). La manipulation d'un objet 3D nécessite ainsi de contrôler jusqu'à 6 degrés de liberté, soit « 6 DdL ».

Degré de liberté (DdL) : dimension indépendante et non contrainte dans le déplacement d'un objet. Un objet rigide dans l'espace 3D possède jusqu'à 6 DdL.

Ce nombre de degrés de liberté, relativement élevé, est un enjeu central dans la conception d'une interface efficace pour la manipulation 3D. En effet, l'interface doit permettre à l'utilisateur d'agir sur chacun de ces 6 DdL d'une manière ou d'une autre sans quoi certains déplacements de l'objet manipulé seraient impossibles. Un facteur déterminant est donc le *nombre de degrés de liberté des dispositifs d'entrée*. Par exemple, une souris de PC possède 2 DdL car son état dans l'interface est représenté par deux valeurs : les coordonnées 2D du pointeur. Un dispositif physique quelconque (rigide) associé à un système de suivi 3D possède 6 DdL : sa position et son orientation dans l'espace réel. La main humaine possède quant à elle jusqu'à 22 DdL [Sherman et Craig 2002].

Les dispositifs d'entrée possédant 6 DdL ou plus peuvent être directement associés aux 6 degrés de liberté de la manipulation. Mais les dispositifs d'entrée possédant moins de 6 DdL nécessitent de décomposer la tâche de manipulation en plusieurs sous-étapes, chacune ne contrôlant qu'une partie des degrés de liberté de l'objet. Cette décomposition complexifie et ralentit les tâches de manipulation sur 6 DdL, ce qui conduit par définition à une moindre efficacité de l'interface.

Si la manipulation 3D en tant que telle possède effectivement 6 degrés de liberté, toutes les tâches n'exigent pas forcément de contrôler autant de paramètres en même temps. Ainsi, un cube posé à plat sur un plan perd deux degrés de liberté en rotation et un en translation. Des contraintes peuvent également être appliquées afin de simplifier la manipulation [Gleicher 1993; Bowman et Hodges 1995], comme par exemple maintenir les meubles au niveau du sol dans un logiciel d'aménagement intérieur. Dans ces situations, un contrôle sur moins de 6 DdL pourrait être suffisant. Cependant, ce type de contrainte est généralement transitoire ou spécifique à une tâche précise. Un dispositif d'entrée fournissant au moins 6 degrés de liberté reste en mesure de supporter ces contraintes si besoin, tout en permettant de revenir à un contrôle 6 DdL lorsque cela est utile. À l'inverse, un dispositif limité à moins de 6 degrés de liberté n'offre pas la même souplesse. C'est pourquoi il est important que l'interface permette de contrôler au moins 6 DdL simultanément pour assurer une manipulation 3D efficace, même si tous les degrés de liberté ne seront pas forcément utilisés à chaque instant.

Un dispositif possédant plus de degrés de liberté, tel que la main, ne serait pas limitant mais pas non plus pleinement exploité car la manipulation d'objets rigides (tâche qui nous intéresse ici) est limitée à 6 DdL. Un tel dispositif poserait par ailleurs la question de quels degrés de liberté choisir pour contrôler la manipulation, tout en évitant que les degrés de liberté superflus n'affectent la précision de l'interaction [Bowman et Hodges 1995].

Par conséquent, pour permettre de manipuler efficacement un objet 3D, une interface devrait fournir à l'utilisateur la capacité de contrôler exactement 6 DdL.

1.3 Portabilité

Outre l'efficacité de l'interface pour la manipulation 3D, un critère important pour favoriser son adoption par les utilisateurs est la *portabilité*. Aujourd'hui, l'immense succès des appareils mobiles montre bien la demande des utilisateurs pour des systèmes portables. À l'heure où il est devenu normal de consulter sa messagerie électronique ou regarder des vidéos sur son appareil mobile en pleine rue, il devient difficile d'admettre qu'il soit nécessaire de se rendre dans un lieu dédié pour manipuler des données 3D. Ainsi, un chercheur souhaitant examiner interactivement le résultat d'une simulation numérique 3D se trouve face au choix suivant : se rendre dans une salle spécifique pour utiliser un équipement avancé de réalité virtuelle (si son laboratoire en possède), ou se servir du PC qui se trouve devant lui, voire de l'appareil mobile qui se trouve dans sa poche. Dans les faits, le système le plus proche sera favorisé, et un système portable présente l'avantage d'être toujours au plus près de l'utilisateur.

Une autre situation dans laquelle la portabilité est importante est de permettre à un utilisateur de poursuivre une tâche chez lui plutôt que d'être limité à son lieu de travail. De même, il peut être souhaitable d'emporter le système avec soi pour s'en servir en réunion ou en conférence. LaValle [2016] note ainsi une forme de progression dans les usages et les exigences des utilisateurs au fur et à mesure qu'une technologie évolue, citant l'exemple de la télévision (cinéma, poste de TV et aujourd'hui appareil mobile) :

> *« La progression est la suivante :*
> 1. *devoir se rendre dans un lieu dédié ;*
> 2. *y avoir accès chez soi ;*
> 3. *pouvoir l'emporter en tout lieu. »* [1]

Cependant, le fait qu'un système soit *transportable* n'est pas forcément suffisant. Un PC de bureau ou une console de jeu de salon, par exemple, sont assez compacts pour être considérés comme « transportables », et sont d'ailleurs occasionnellement déplacés d'un lieu à l'autre. Mais ces systèmes ne sont pas conçus pour être utilisés *durant* leur transport, car ils doivent être branchés à un écran et à une prise de courant pour fonctionner. Un « ordinateur portable » intègre lui une batterie et un écran, mais beaucoup de modèles nécessitent tout de même d'être posés sur une surface plane de taille suffisante avant d'être utilisés. À l'inverse, des systèmes tels que les appareils mobiles (smartphone, tablette...) ou les consoles de

1. *«The progression is from: 1) having to go somewhere to watch it, to 2) being able to watch it in your home, to 3) being able to carry it anywhere.»* [LaValle 2016]

jeu portables sont réellement conçus pour être immédiatement utilisables, sans exiger de branchement ou de surface adéquate.

Nous définirons donc le critère de *portabilité* comme suit :

Portabilité : capacité de pouvoir transporter le système, mais également de pouvoir l'utiliser pendant son déplacement sans nécessiter de phase d'installation (branchements, calibration, etc.).

1.4 Non-intrusivité

Un troisième aspect pouvant affecter le choix ou non d'utiliser un système au quotidien est son *intrusivité*. En termes absolus, un système est « intrusif » lorsqu'il nécessite que l'utilisateur s'équipe d'un ou plusieurs dispositifs d'interaction pour fonctionner, par opposition à un système qui serait utilisable directement sans devoir porter d'équipement spécifique (parfois qualifié de « transparent »). Un système intrusif nécessite donc une étape préalable consistant à revêtir un équipement, et une étape correspondante consistant à l'enlever ensuite.

Cependant, cette étape supplémentaire ne suffit pas à elle seule pour rendre un système moins pratique à utiliser. En effet, il est possible d'imaginer que l'utilisateur s'équipe d'un dispositif d'interaction en début de journée et continue à le porter jusqu'au soir, voire le porte en permanence. Dans ce cas, le dispositif serait déjà équipé au moment de commencer à se servir du système. L'intrusivité ne constitue donc un obstacle que si le système requiert un dispositif d'interaction *contraignant*, empêchant ainsi l'utilisateur de le porter en permanence.

Plusieurs facteurs peuvent empêcher cela. Premièrement, le dispositif peut réduire les sensations de l'utilisateur, voire le couper entièrement du monde extérieur. Dans ces conditions, il n'est pas envisageable de le porter en dehors des périodes spécifiques d'utilisation de l'interface. Le fait de perturber le contact avec l'extérieur peut de plus se révéler problématique pendant l'utilisation, par exemple lorsqu'il s'agit de communiquer avec un collègue. Deuxièmement, le dispositif peut gêner la liberté de mouvement de l'utilisateur, que ce soit à cause d'un poids important ou simplement en étant attaché mécaniquement à un point fixe. Là encore, l'utilisateur n'a d'autre choix que d'équiper puis d'enlever le dispositif à chaque usage du système.

Nous définirons par conséquent la propriété de *non-intrusivité* de la manière suivante :

> **Non-intrusivité** (ou transparence) **:** ne pas obliger l'utilisateur à porter d'équipement le coupant du monde réel et/ou limitant sa liberté de mouvement.

Dans la suite de ce travail, nous utiliserons l'expression « système non-intrusif » plutôt que « système transparent » afin de souligner l'absence d'équipements « intrusifs ».

1.5 Accessibilité

Un dernier critère que nous considérons jouer un rôle dans l'adoption d'un système d'interaction est son *accessibilité*. Il ne s'agit pas ici d'accessibilité dans le sens de la prise en compte du handicap (même si cet aspect a sa propre importance), mais plutôt de la capacité de l'ensemble des utilisateurs concernés à *pouvoir accéder au matériel* que requiert l'interface.

En effet, s'il est tentant de vouloir créer des systèmes puissants et performants pour la manipulation 3D en se basant sur des technologies très avancées, ou sur du matériel très haut de gamme, il ne faut pas oublier qu'un utilisateur n'ayant pas d'accès physique à un tel matériel ne pourra pas en bénéficier. Un nouveau système reposant en partie ou totalement sur du matériel déjà répandu ou abordable aura donc plus de chances d'apporter un bénéfice concret aux utilisateurs par rapport aux systèmes actuellement dominants.

Nous définirons donc ainsi le critère *d'accessibilité* :

> **Accessibilité :** capacité d'un système à fonctionner sur du matériel déjà largement répandu, et/ou de coût raisonnable.

On notera que, sur le plan pratique, l'accessibilité et la portabilité sont d'une certaine façon liées. Un système non portable, mais capable de fonctionner sur un matériel *accessible* comme un PC classique pourrait répondre à notre objectif d'en favoriser l'utilisation. C'est d'ailleurs la situation dont bénéficient les logiciels PC actuels (mais ceux-ci ne répondent pas à notre autre critère de permettre une manipulation 3D efficace). À l'opposé, un système nécessitant du matériel spécifique et/ou coûteux, donc *peu accessible*, mais par ailleurs totalement portable, répondrait également à notre objectif. En effet, même si le prix de départ du système est important, les utilisateurs peuvent l'avoir constamment avec eux ce qui compense l'inconvénient d'employer un matériel spécifique.

Cependant, un système *à la fois portable* et basé sur du matériel *accessible* (tout en restant efficace pour la manipulation 3D) aurait un avantage sur les deux systèmes précédents. Il y a donc un intérêt à poursuivre ces deux objectifs simultanément.

Chapitre 2

Systèmes existants

Dans ce chapitre, nous examinons les différents systèmes interactifs existant actuellement pour la manipulation 3D afin d'identifier de quelle manière ceux-ci répondent aux critères d'efficacité, de portabilité, de non-intrusivité et d'accessibilité définis précédemment.

À la différence d'autres études, l'objectif n'est pas ici de comparer un par un chaque dispositif d'interaction (affichage, entrée...) mais plutôt d'examiner les systèmes dans leur ensemble, tels qu'ils sont susceptibles d'être rencontrés sur le terrain. En effet, ce sont bien les caractéristiques du système entier qui déterminent en pratique son adoption (ou non) par les utilisateurs. Un dispositif d'interaction pris individuellement pourrait répondre aux critères que nous nous sommes fixés, mais sa combinaison avec d'autres dispositifs plus contraignants suffirait à annuler ces avantages. C'est donc en examinant les *systèmes* existants que nous pourrons comprendre pourquoi, malgré le développement d'approches de plus en plus avancées, les utilisateurs continuent à se servir d'interfaces sous-optimales pour la manipulation 3D.

2.1 Ordinateur personnel

Dans un contexte d'interaction homme-machine, les systèmes de type « ordinateur personnel » (*personal computer*, PC) correspondent à un ensemble de dispositifs d'interaction bien spécifiques. En plus d'une unité centrale, un ordinateur personnel est constitué d'au moins un écran, un clavier, et un dispositif de pointage 2D (souris, pavé tactile...), le tout étant dimensionné pour être posé sur un bureau.

Plusieurs sortes d'appareils répondent à cette définition (figure 2.1). Un « ordinateur de bureau » est une machine de taille moyenne, peu adaptée au transport mais largement répandue sur les lieux de travail. Une « station de travail » est un ordinateur spécialisé pour une tâche particulière, disposant par exemple de capacités graphiques accrues ou d'une connexion dédiée à un serveur de données. Un « ordinateur portable » est plus compact qu'un ordinateur de bureau de sorte qu'il peut être facilement transporté, même s'il n'est

(a) Ordinateur de bureau (b) Ordinateur portable (c) iMac G4

FIGURE 2.1 — Plusieurs types d'ordinateur personnel (PC). Malgré leurs différences, tous ces modèles sont conçus autour des mêmes dispositifs d'interaction : écran, clavier et pointage 2D.

généralement pas conçu pour être utilisé en plein déplacement. Selon le modèle considéré, le PC répond donc plus ou moins bien à notre critère de *portabilité*.

En dépit de ces différences, tous les modèles ci-dessus ont en commun de reposer sur les mêmes dispositifs d'interaction : écran, clavier et pointage 2D. C'est pourquoi nous emploierons le terme de « plate-forme PC » pour désigner ce type d'appareils :

Plate-forme PC :
— une unité centrale
— un écran (au minimum)
— un clavier
— un dispositif de pointage 2D (souris, pavé tactile, etc.)

Aucun des dispositifs ci-dessus ne nécessitant d'être porté en tant qu'équipement par l'utilisateur, la plate-forme PC répond entièrement à notre critère de *non-intrusivité*.

Dans les faits, cette plate-forme est aujourd'hui la plus utilisée pour la manipulation 3D. En effet, lorsqu'on examine les principaux domaines d'application de la manipulation 3D, on constate que la plupart des logiciels disposant d'une part de marché significative sont conçus pour la plate-forme PC. Ainsi, dans le domaine de la CAO (conception 3D), les 20 premiers logiciels les plus utilisés en production représentant 90 % des parts de marché [CNC Cookbook 2016] sont tous, sans exception, des logiciels PC (comme sans doute la plupart des logiciels restants). Même si certains ont pu être portés sur d'autres plates-formes, notamment sur appareil mobile [1], l'usage de ces versions alternatives demeure limité par rapport à la version PC d'origine. Une étude portant spécifiquement sur le type de matériel

1. `https://www.autocad360.com/`

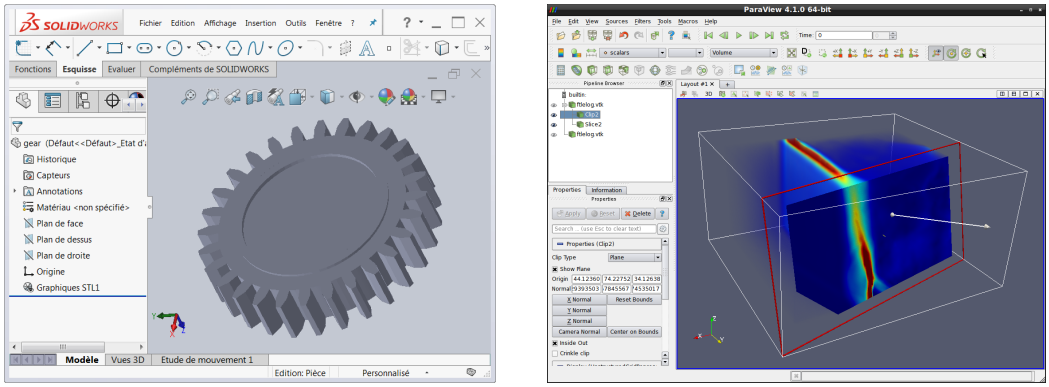
[Worldwide CAD Trends 2015] confirme que l'ordinateur personnel est bien actuellement la plate-forme la plus utilisée en CAO. On observe le même phénomène dans le domaine de l'infographie 3D : les logiciels les plus utilisés dans l'industrie [Birn 2004] sont, eux aussi, systématiquement conçus pour le PC. Une autre application de la manipulation 3D est l'exploration de données scientifiques. Là encore, les principaux outils de visualisation interactive [2] (ParaView, MATLAB, IDL...) sont avant tout des logiciels PC, avec quelques portages limités sur mobile. Enfin, le jeu vidéo 3D représente un domaine majeur sur la plate-forme PC. Même s'il existe des plates-formes dédiées (les consoles de jeu), on notera que la majorité des jeux 3D vendus à plus de 15 millions d'exemplaires sont présents sur PC [3].

Une des principales raisons de cette dominance est, vraisemblablement, le très large déploiement de la plate-forme PC. En France, la quasi-totalité des sociétés d'au moins 10 personnes sont aujourd'hui équipées d'ordinateurs [Insee 2012]. Plus des trois quarts des ménages disposent eux aussi de ce type de matériel à leur domicile [Insee 2015]. Il n'est donc pas surprenant que les éditeurs de logiciels privilégient cette plate-forme, et que les versions les plus abouties de leurs outils existent en premier lieu sur PC. En d'autres termes, un intérêt majeur du PC est d'être une plate-forme *accessible*.

Pourtant, du point de vue de *l'interaction*, cette plate-forme n'est pas véritablement adaptée pour la manipulation 3D. Le PC tel qu'il existe aujourd'hui a été développé pour répondre à certaines tâches bien précises. Par exemple, le clavier excelle pour les tâches d'entrée symbolique et de contrôle d'application. Le dispositif de pointage 2D est quant à lui parfaitement approprié pour la manipulation d'objets 2D dans les interfaces graphiques WIMP (« Fenêtres [*Windows*], Icônes, Menus, Pointeur », [Myers 1988]), qui constituent encore aujourd'hui la majorité des interfaces PC (figure 2.2).

Cependant, ces deux dispositifs d'entrée, qui caractérisent la plate-forme PC, présentent une importante limitation pour la manipulation 3D. Conçus pour des interfaces 2D, ils n'offrent pas suffisamment de *degrés de liberté*. Comme nous l'avons vu, la manipulation d'objets en trois dimensions requiert jusqu'à 6 degrés de liberté (3 pour les translations, et 3 pour les rotations). Chaque touche d'un clavier de PC ne possède que deux états : pressée ou relâchée. S'il est ainsi théoriquement possible de contrôler chaque degré de liberté avec deux touches, il reste pourtant difficile d'envisager contrôler 6 DdL de cette manière. Certains claviers possèdent quatre touches directionnelles, qui peuvent être utilisées pour effectuer soit des translations, soit des rotations sur deux axes. Dans ces conditions, l'utilisateur est amené à actionner jusqu'à deux touches simultanément. Or, un contrôle sur 6 DdL nécessiterait 12 touches directionnelles. Une telle configuration demanderait de plus à

2. `http://www.cscs.ch/a-display.php?id=57`
3. Source [Wikipédia 2016], en considérant uniquement les jeux vidéo 3D.



(a) SolidWorks (b) ParaView

FIGURE 2.2 — Copies d'écran de logiciels PC centrés sur la manipulation 3D. L'interface utilisateur et les techniques d'interaction sont entièrement pensés en fonction des deux dispositifs d'entrée qui caractérisent la plate-forme PC : le clavier et le dispositif de pointage 2D.

l'utilisateur d'actionner jusqu'à 6 touches simultanément. De toute évidence, le clavier n'est donc pas une solution adaptée pour effectuer ce type de tâche.

Le dispositif de pointage 2D permet, lui, de contrôler jusqu'à 2 DdL simultanément. Mais cela reste insuffisant pour effectuer des translations ou des rotations seules (3 DdL), et *a fortiori* pour un contrôle 6 DdL. Pour cette raison, utiliser le dispositif de pointage 2D pour la manipulation 3D nécessite de décomposer la manipulation en sous-étapes, chacune étant limitée à 1 ou 2 DdL (figure 2.3). En appuyant sur des boutons situés sur le dispositif de pointage ou sur des touches du clavier, l'utilisateur choisit quels degrés de liberté de la manipulation il souhaite associer aux 2 DdL du dispositif. Ainsi, dans le logiciel de visualisation scientifique ParaView[4], une pression sur le bouton gauche active un mode permettant de contrôler les rotations autour des axes X et Y, une pression sur le bouton central permet de contrôler les translations autour des axes X et Y, et l'appui sur certaines touches du clavier en plus d'un bouton de la souris permet de contrôler l'axe restant (Z) en rotation ou en translation. Dans le logiciel Blender[5], l'appui sur la touche g entre dans un mode permettant de contrôler les translations sur le plan de l'écran, la touche r permet de contrôler les rotations sur l'axe perpendiculaire à l'écran, et les touches x, y et z permettent de spécifier explicitement quel axe de translation ou de rotation contrôler par le biais du dispositif de pointage. Une telle décomposition, indispensable pour pallier au manque de degrés de liberté, réduit d'autant l'efficacité de la manipulation. Les limitations des dispositifs d'entrée font donc apparaître le PC comme une plate-forme *sous-optimale* pour ce type de tâche.

4. `http://www.paraview.org/`
5. `https://www.blender.org/`

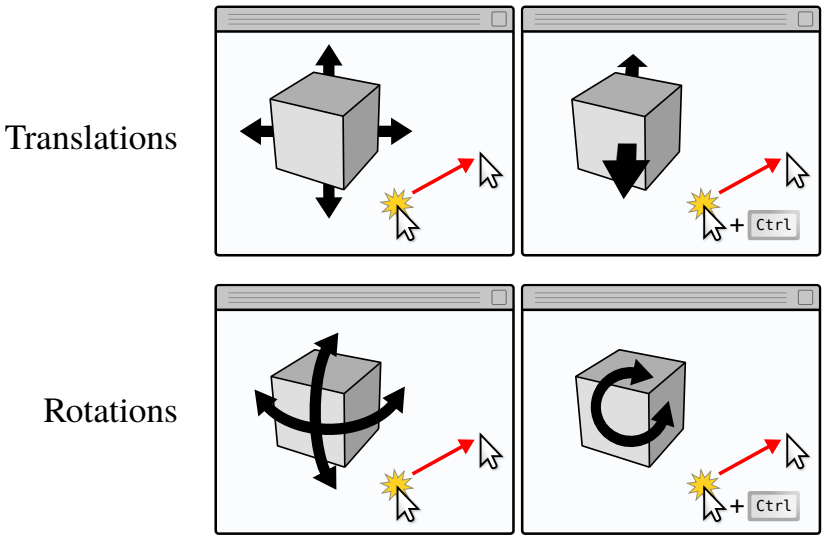


FIGURE 2.3 — Exemple typique de manipulation 3D sur PC utilisant les dispositifs standards de cette plate-forme. Le dispositif de pointage 2D (souris ou équivalent) ne fournissant que deux degrés de liberté, il est nécessaire de décomposer la manipulation en tâches de translation et de rotation, puis en sous-étapes de 2 DdL maximum. Le changement de mode se fait avec les boutons de la souris et/ou les touches du clavier (ici la touche « control »).

En principe, il serait bien sûr possible de répondre aux limitations ci-dessus en rajoutant à un PC classique des dispositifs d'interaction spécialisés. Certaines stations de travail pour la CAO sont effectivement équipées de « souris 3D », des dispositifs de contrôle 6 DdL. Cependant, la *plate-forme PC* en elle-même ne suppose que la présence d'un clavier et d'un dispositif de pointage 2D. Il s'agit de la configuration la plus répandue, celle qui donne justement au PC son intérêt en terme d'accessibilité. Les logiciels PC restent conçus en fonction de ce dénominateur commun car cette configuration, étant la plus répandue, est aussi (par définition) la plus susceptible d'être rencontrée par les utilisateurs.

En pratique, on constate donc que la majorité des tâches de manipulation 3D sont aujourd'hui réalisées sur PC par le biais d'un clavier et d'une souris 2D : une plate-forme *fondamentalement inadaptée* à cette tâche.

Le tableau suivant résume les avantages et inconvénients de la plate-forme PC pour répondre à nos quatre critères, au vu de l'analyse ci-dessus :

efficacité	**portabilité**	**non-intrusivité**	**accessibilité**
– –	+/–	+ +	+ +

2.2 Appareil mobile

Bien que le PC soit longtemps resté le principal outil informatique chez le grand public, ces dernières années ont vu se développer massivement une nouvelle forme de matériel : les *appareils mobiles*. À l'origine de simples « téléphones intelligents » (*smartphones*), les appareils mobiles se sont progressivement éloignés de ce seul rôle. Ils sont aujourd'hui suffisamment perfectionnés pour qu'on puisse envisager de s'en servir pour toutes sortes de tâches jusqu'alors réservées au PC, dont la manipulation 3D.

FIGURE 2.4 — Appareils mobiles : smartphone (à gauche) et tablette (à droite). Malgré des différences de format, la plupart des appareils mobiles actuels présentent de fortes similitudes.

Il existe plusieurs sortes d'équipements habituellement qualifiés d'appareils mobiles : les smartphones, les tablettes, les « phablettes »... Chaque type d'appareil est optimisé pour un certain usage. Cependant, le développement de ces différents modèles semble avoir convergé vers une même configuration présentant des caractéristiques communes (figure 2.4). Comme dans le cas du PC, on peut donc considérer qu'il existe une « plate-forme mobile ». Cette dernière est constituée des éléments suivants :

Plate-forme mobile :
- une unité centrale de taille compacte
- un écran tactile, généralement multipoints, recouvrant une face de l'unité centrale
- un nombre limité de boutons physiques sur le reste de sa surface

En plus de ces caractéristiques, on retrouve souvent un certain nombre de capteurs intégrés (gyroscopes, accéléromètres, ou encore une caméra miniature). Mais l'intérêt majeur de cette plate-forme par rapport au PC est que les composants ci-dessus sont totalement intégrés dans *un unique objet de taille réduite.* Un appareil mobile est donc véritablement conçu pour être *portable*. De plus, l'intégration des dispositifs d'interaction dans un même objet fait que

l'utilisateur n'a pas à porter d'équipement particulier, ce qui répond également à notre critère de *non-intrusivité*.

Aujourd'hui, le taux de déploiement des appareils mobiles est de plus en plus important. En 2015, plus d'un Français sur deux était équipé d'un smartphone [ARCEP 2015]. Si l'avantage principal du PC est d'être accessible, la plate-forme mobile semble donc de plus en plus combiner les avantages de *l'accessibilité* avec ceux de la portabilité (tout en restant non-intrusive). Pour cette raison, les appareils mobiles apparaissent comme une solution intéressante pour répondre à nos objectifs. Cependant, comme précédemment, le principal obstacle reste au niveau des capacités d'interaction.

Sur un appareil mobile typique, les boutons physiques sont en nombre bien insuffisant pour envisager de s'en servir pour la manipulation 3D. En pratique, le principal mode d'interaction utilisé actuellement est donc l'écran tactile multipoints. Chaque contact sur l'écran tactile fournit 2 DdL, ce qui est parfaitement adapté à la manipulation d'objets 2D [Steinicke *et al.* 2008] et correspond bien à la plupart des interfaces mobiles qui restent majoritairement en 2D. Mais la manipulation 3D requiert elle 6 DdL pour un contrôle complet, ou au minimum 3 DdL pour un contrôle séparé des translations et des rotations. Un seul contact est donc insuffisant pour contrôler la manipulation. Plusieurs solutions ont été proposées pour tenter de résoudre cette limitation.

La première consiste à décomposer la manipulation en sous-étapes de 1 ou 2 DdL maximum, chacune pouvant donc être contrôlée avec un seul doigt [Decle et Hachet 2009; Buda 2012]. Afin de couvrir tous les degrés de liberté, il est nécessaire de changer de « mode » [Besançon *et al.* 2016], par exemple par le biais de boutons physiques ou de zones sur l'écran tactile. Comme dans le cas de la souris 2D, cette décomposition rend la manipulation plus complexe et moins efficace.

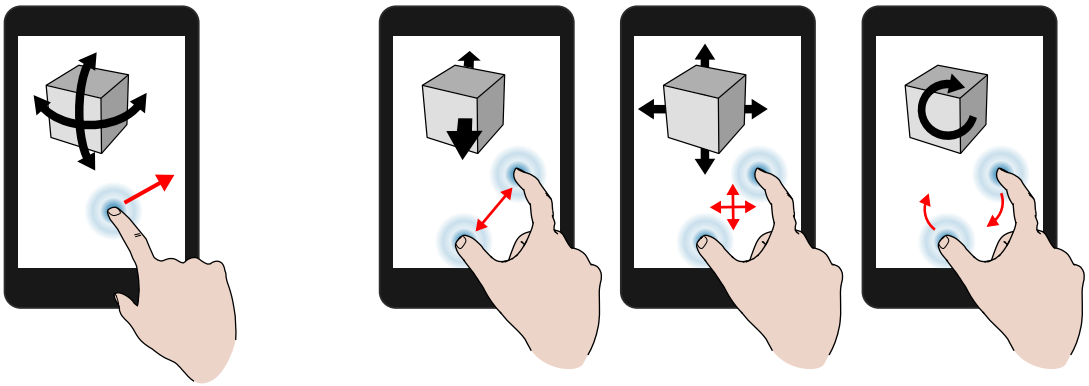


FIGURE 2.5 — Technique RST (*rotate-scale-translate*). Cette technique, la plus répandue dans les logiciels pour plate-forme mobile, permet de contrôler les 6 DdL de manipulation en utilisant au maximum deux doigts simultanément. Cependant, le fait que les deux doigts contrôlent à la fois des translations et des rotations rend cette technique difficile à maîtriser.

Une deuxième solution consiste à utiliser jusqu'à deux doigts simultanément. Un exemple de cette approche est la technique RST (*rotate-scale-translate*), qui se trouve également être la plus employée dans les logiciels existants [Besançon *et al.* 2016]. Dans cette technique, un mode utilisant un seul doigt permet de contrôler deux axes de rotation, et un autre mode utilisant deux doigts (soit 4 DdL) permet de contrôler l'ensemble des translations ainsi que l'axe de rotation restant (figure 2.5). Cependant, cette technique présente l'inconvénient d'associer à la fois des translations et des rotations à un même mode. Cela est dû au fait que les 3 DdL de positionnement et les 3 DdL de rotation sont répartis par groupes de 2 DdL, ce qui conduit nécessairement à un mode combinant les deux. Cette manière de contrôler translations et rotations au travers d'une même action est connue pour être difficile à maîtriser par les utilisateurs [Besançon *et al.* 2016].

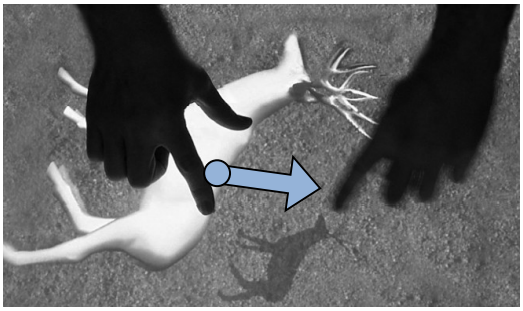


FIGURE 2.6 — « Sticky tools » [Hancock *et al.* 2009], un ensemble de techniques pour tables tactiles (*tabletops*) utilisant jusqu'à trois doigts simultanément. Bien que cela permette de contrôler effectivement 6 DdL, l'utilisation de trois doigts est peu adaptée à un appareil mobile.

Une troisième solution est d'employer jusqu'à trois doigts simultanément, ce qui permet ainsi de contrôler entièrement les 6 degrés de liberté [Hancock *et al.* 2007, 2009]. Mais l'utilisation de trois doigts est problématique dans le cas d'un appareil mobile. Les techniques 6 DdL ont été conçues pour des tables tactiles (*tabletop*) et supposent l'utilisation des deux mains durant l'interaction. Dans le cas d'un appareil mobile en situation de portabilité, il est fréquent que n'avoir qu'une seule main disponible car l'autre tient l'appareil. Par conséquent, les techniques à trois doigts ne semblent pas être une option viable sur appareil mobile.

Dans l'ensemble, toutes ces limitations font que la manipulation 3D sur 6 degrés de liberté reste aujourd'hui complexe et peu intuitive sur cette plate-forme [Steinicke *et al.* 2008].

Le tableau suivant résume les avantages et inconvénients de la plate-forme mobile pour répondre à nos quatre critères :

efficacité	**portabilité**	**non-intrusivité**	**accessibilité**
– –	+ +	+ +	+ +

2.3 Installations fixes de réalité virtuelle

Face aux limitations des plates-formes PC et mobile pour réaliser des tâches pour lesquelles elles n'ont pas été initialement prévues, des approches alternatives ont été développées avec comme objectif explicite de faciliter l'interaction avec des environnements virtuels 3D.

Une de ces approches les plus notables est la « réalité virtuelle ». Pour le grand public, ce concept renvoie immédiatement à l'idée « d'immersion » dans un monde virtuel. L'immersion complète n'est pourtant pas l'aspect le plus important ici. Il existe d'ailleurs une multitude d'implémentations et d'applications de la réalité virtuelle, certaines immersives et d'autres non. Il est donc utile de revenir aux bases de cette approche afin d'identifier ce qui la rend intéressante pour la manipulation 3D.

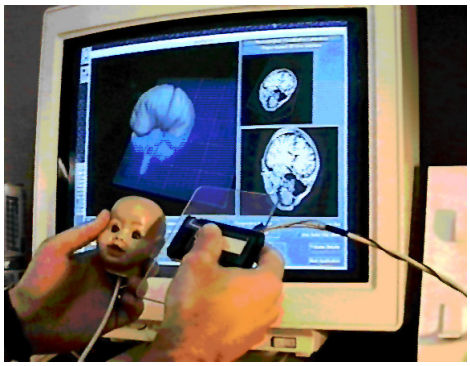
Fuchs et Moreau [2006] définissent la réalité virtuelle comme ayant pour but de « permettre à une personne (ou à plusieurs) une activité sensori-motrice et cognitive dans un monde artificiel, créé numériquement, qui peut être imaginaire, symbolique ou une simulation de certains aspects du monde réel ». Bowman *et al.* [2004] proposent quant à eux une définition plus directe : la réalité virtuelle serait « un monde artificiel (généralement un espace 3D) observé en vue subjective »[6].

Un point commun ressort de ces deux définitions : la réalité virtuelle consisterait à simuler la présence d'un environnement virtuel 3D *autour* de l'utilisateur (« vue subjective »), dans lequel peut alors se dérouler l'interaction (« activité sensori-motrice »). Indépendamment de leur caractère immersif, les systèmes de réalité virtuelle ont donc comme particularité d'amener *l'interaction* dans le *même espace que l'utilisateur*.

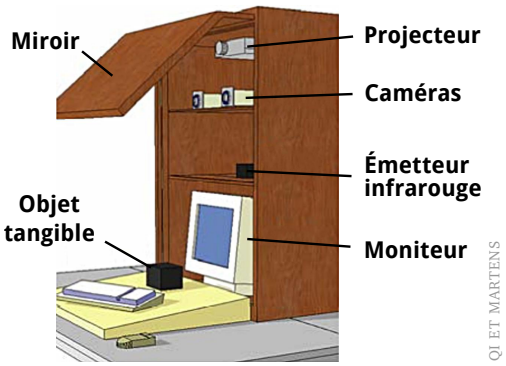
Pour une tâche de manipulation, pouvoir interagir directement dans un espace 3D présente d'importants avantages. Déplacer un objet virtuel peut ainsi être effectué par l'intermédiaire de véritables translations dans l'espace concerné. De même, tourner un objet virtuel peut être effectué par le biais de véritables rotations. Enfin, sélectionner un objet virtuel peut être aussi simple que de tendre la main vers son emplacement. En s'affranchissant des limitations des dispositifs d'interaction traditionnels, la réalité virtuelle autorise donc une manipulation 3D potentiellement aussi *efficace* et *naturelle* que pourrait l'être la manipulation d'objets dans le monde réel.

Afin d'accomplir cet objectif, la réalité virtuelle nécessite au minimum trois éléments : un système d'affichage, un système de suivi en trois dimensions (*tracking*) capable de couvrir

6. « *A synthetic, spatial (usually 3D) world seen from a first-person point of view.* » [Bowman *et al.* 2004]



(a) « PassProps » [Hinckley *et al.* 1994], reposant sur un suivi magnétique.



(b) Suivi optique d'objets tangibles autour d'un écran de PC [Qi et Martens 2005].

FIGURE 2.7 — Réalité virtuelle « de bureau » (*desktop VR*) : basée sur un PC classique, mais nécessitant de mettre en place un système de suivi 3D autour de celui-ci.

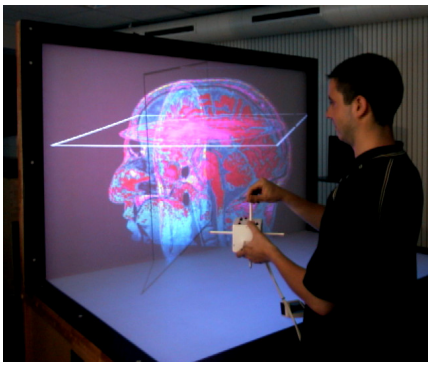
l'espace souhaité, et un moyen d'interagir avec cet espace. Pour cette raison, beaucoup de systèmes de réalité virtuelle prennent la forme « d'installations » combinant ces trois éléments.

Ce type d'installation peut s'organiser autour d'un simple moniteur de PC ou écran de télévision, à condition de mettre en place un système de suivi autour de cet écran. On parle alors de « *desktop VR* » (réalité virtuelle « de bureau »). Un exemple de telle installation est le système « PassProps » proposé par Hinckley *et al.* [1994] pour l'exploration de données médicales. L'utilisateur manipule des objets physiques devant un écran. Ces objets physiques, aussi appelés « objets tangibles » [Ishii 2008; Shaer et Hornecker 2010] ou « *props* » (accessoires), symbolisent chacun un objet virtuel. Ils sont reliés à un système magnétique capable de suivre leurs mouvements dans l'espace réel. Tout déplacement est ainsi immédiatement appliqué aux objets virtuels correspondants sur l'écran. D'autres systèmes basés sur ce principe ont ensuite été proposés, favorisant plutôt un système de suivi optique pour éliminer les câbles reliés aux objets physiques. Qi et Martens [2005] utilisent ainsi une combinaison de caméras infrarouges et de miroirs pour suivre plusieurs objets tangibles devant un moniteur. Ce système permet à l'utilisateur de manipuler une donnée volumique, un plan de coupe et un pointeur virtuel visibles à l'écran. Gillet *et al.* [2004] utilisent une caméra pour suivre des reproductions physiques de molécules dans l'espace 3D, et augmenter leur image sur l'écran avec des informations calculées dynamiquement. Jackson *et al.* [2013] se servent d'une caméra et d'un capteur de profondeur (Microsoft Kinect) pour suivre un *prop* en papier muni de marqueurs spécifiques, qui est utilisé pour manipuler des objets 3D sur l'écran.

Plutôt que de chercher à mettre en place des équipements spécifiques autour d'un moniteur de PC classique, il existe des installations complètes dédiées à la réalité virtuelle. On peut citer comme exemple les *workbenches* (« tables de réalité virtuelle »). Ces installations sont conçues pour reproduire le principe d'un plan de travail ou d'un établi, mais avec l'avantage pour l'utilisateur de pouvoir interagir directement avec des objets virtuels dans l'espace

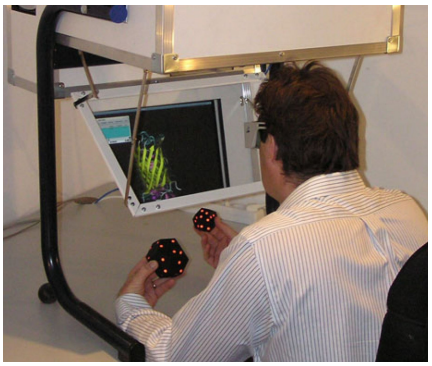
FAKESPACE SYSTEMS

(a) ImmersaDesk R2



FRÖHLICH ET PLATE

(b) Responsive Workbench et Cubic Mouse [Fröhlich et Plate 2000]



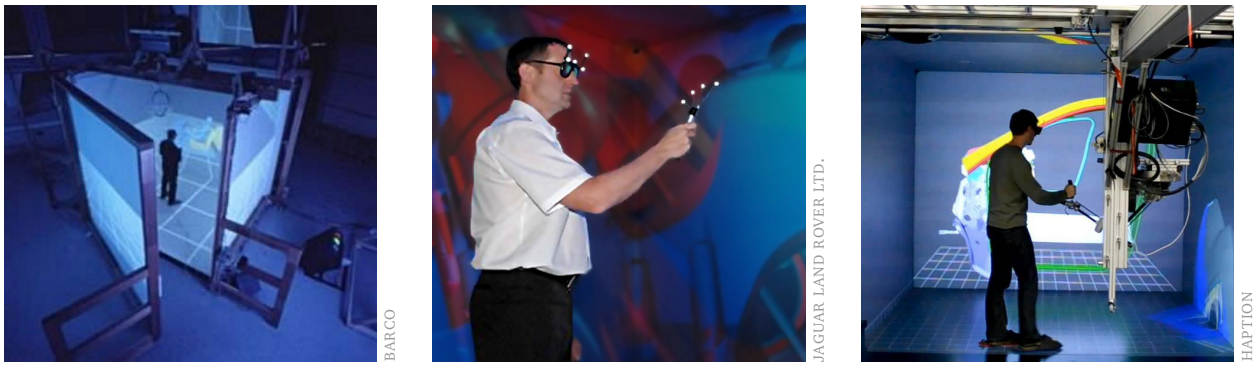
VAN RHIJN ET MULDER

(c) Personal Space Station [van Rhijn et Mulder 2006]

FIGURE 2.8 — Plusieurs exemples de *workbenches*, des installations complètes de réalité virtuelle constituées d'une ou plusieurs surfaces de projection permettant à l'utilisateur d'interagir dans l'espace 3D situé devant lui.

situé devant lui. Elles sont typiquement constituées d'une surface de projection au-dessus de laquelle se déroule la manipulation, ou alternativement de deux surfaces disposées à angle droit. L'« ImmersaDesk » [Czernuszenko *et al.* 1997] est un *workbench* comportant une unique surface de projection, ajustable de l'horizontale à la verticale. La projection est stéréoscopique, créant ainsi l'illusion visuelle que les objets virtuels se trouvent dans le même espace que l'utilisateur. L'interaction avec les objets s'effectue en tenant dans la main un dispositif appelé « *wand* » (manette). Ce dispositif, dont la position et l'orientation sont suivis par le système, permet aux utilisateurs de pointer ou de manipuler les objets virtuels directement dans l'espace 3D. Le « HoloBench »[7] est un autre exemple de *workbench*, muni de deux surfaces de projection disposés à angle droit délimitant un volume d'interaction. Cette configuration permet d'agrandir nettement l'espace dans lequel les objets virtuels peuvent être manipulés. Le « Responsive Workbench » [Krueger et Froehlich 1994; Fröhlich et Plate 2000] est une marque désignant différents types de *workbenches*, munis d'une ou de plusieurs surfaces de projection. L'interaction nécessite habituellement de s'équiper d'un gant instrumenté (*data glove*), mais il est également possible d'utiliser un dispositif appelé « Cubic Mouse » [Fröhlich et Plate 2000] qu'il suffit de tenir dans la main à la manière d'un *wand*. Le « Personal Space Station » [Mulder et Van Liere 2002], aujourd'hui commercialisé sous le nom de « PSS » et de « C-Station »[8], est une variante du concept de *workbench*. Cette installation prend la forme d'un bureau devant lequel est assis l'utilisateur. L'interaction se déroule cette fois *derrière* la surface d'affichage, par le biais d'objets tangibles (*props*) représentant les objets virtuels à manipuler. Le système « HUVR » [Margolis *et al.* 2011] repose sur un principe similaire, mais l'interaction se fait grâce à un bras robotisé, ou « bras haptique ».

7. `http://www.barco.com/fr/Produits-et-solutions/L-shaped-3D-projection-table-with-2-orthogonal-projection-surfaces.aspx/`
8. `http://www.ps-tech.com/3d-workstations/3d-workstation-pss`



(a) Exemple de salle immersive : le TAN VR-Cube [9].

(b) Interaction à travers un *wand*.

(c) Interaction à travers un bras robotisé (Haption Scale1).

FIGURE 2.9 — Les installations de type « salle immersive » entourent quasi entièrement l'utilisateur de parois de projection, délimitant ainsi un volume d'interaction.

L'intérêt du bras haptique, par rapport à un simple objet tangible, est de pouvoir également fournir un retour de force à l'utilisateur.

Un troisième type d'installation est la « salle immersive ». Celle-ci consiste en un espace clos ou semi-clos dans lequel se tient l'utilisateur. Chacune des parois de cet espace constitue une surface d'affichage, grâce à des projecteurs situés à l'extérieur. Le système de suivi 3D couvre l'intégralité du volume intérieur de la salle. La position de l'utilisateur dans cet espace est continuellement suivie par le système, qui adapte la projection sur les parois pour produire l'illusion que l'utilisateur est entouré par l'environnement virtuel. L'interaction avec les objets virtuels peut se faire de différentes façons. L'une des plus courantes consiste à tenir dans la main un dispositif *wand*, lui aussi suivi par le système et qui permet de pointer et de manipuler des objets 3D. Une autre approche est d'utiliser un bras robotisé, à la fois pour le suivi et pour fournir un retour de force à l'utilisateur. Il est enfin possible de se servir d'un gant instrumenté après en avoir équipé l'utilisateur. La plus connue de ces salles immersives est le « CAVE » (*Cave Automatic Virtual Environment*) inventé par Cruz-Neira *et al.* [1992, 1993]. Depuis, de nombreux systèmes inspirés du système CAVE ont été développés. Parmi les exemples les plus notables, on peut citer le « VR-Cube » [9] qui peut gérer jusqu'à 6 parois de projection, entourant ainsi complètement l'utilisateur. Le « RAVE » [10] (*Reconfigurable Automatic Virtual Environment*) permet quant à lui de disposer les surfaces d'affichage dans d'autres configurations que celle d'un cube.

Comme nous l'avons vu plus haut, la réalité virtuelle est une solution bien adaptée pour permettre une manipulation 3D efficace, grâce au principe d'amener l'interaction dans l'es-

9. `http://www.barco.com/en/Products/Cubic-immersive-virtual-environment.aspx`
10. `http://www.fakespacesystems.com/products/rave.html`

pace de l'utilisateur. Par ailleurs, il existe aujourd'hui un certain nombre d'installations de réalité virtuelle dont certaines sont même directement commercialisées. Il s'agit donc d'une technologie pouvant être considérée aujourd'hui comme mature.

Pourtant, la majorité des tâches de manipulation 3D sont aujourd'hui effectuées sur d'autres plates-formes beaucoup moins adaptées, notamment le PC. Comment expliquer alors que les installations ci-dessus soient *si peu utilisées* en pratique, y compris dans des professions pour lesquelles la manipulation 3D représente une tâche quotidienne (CAO, modélisation...) ?

Selon Roussou et Drettakis [2005], la *spécificité du matériel requis* et le *manque de portabilité* seraient les principales raisons pouvant expliquer la faible utilisation des installations de réalité virtuelle :

> « Les problèmes associés au déploiement de la réalité virtuelle dans le travail quotidien sont bien connus et s'expliquent toujours par les mêmes difficultés pratiques : devoir mettre en place un matériel spécifique et coûteux sous forme d'installations difficilement transportables, qui nécessitent des équipes spécialisées de développeurs et de techniciens [...] » [11]

Effectivement, toutes les installations citées plus haut requièrent un certain nombre d'équipements spécialisés, que ce soit pour l'affichage ou pour le suivi 3D. L'utilisation de matériel spécifique et parfois coûteux va à l'encontre du critère *d'accessibilité* établi dans le chapitre précédent. Le manque d'accessibilité pourrait être compensé par du matériel portable, car un système que l'on a toujours avec soi et que l'on peut utiliser immédiatement en tout lieu n'a besoin d'être acquis qu'une seule fois. Mais des installations « difficilement transportables » ne peuvent prétendre répondre à ce second critère de *portabilité*.

Les approches de type *« desktop VR »* pourraient à première vue sembler les plus accessibles car elles sont typiquement construites autour d'un PC classique. La réalité virtuelle nécessite cependant un système de suivi (magnétique, optique, bras robotisé...) pour amener l'interaction dans l'espace de l'utilisateur. Or, un tel système de suivi ne fait *pas* partie de la plate-forme PC telle que nous l'avons définie plus haut. Un « PC classique », comme on peut aujourd'hui en trouver dans la plupart des lieux de travail, n'est pas suffisant pour faire fonctionner ce type d'approche. La *desktop VR* ne peut donc pas bénéficier du large déploiement de la plate-forme PC. La portabilité de ces approches est également limitée. Afin de couvrir un volume d'espace suffisamment grand autour de l'écran, le système de suivi doit être placé à une certaine distance, ce qui peut nécessiter un système de miroirs [Qi et Martens 2005],

11. « *The issues regarding the deployment of VR in everyday work contexts have been discussed many times and continue to revolve around the familiar practical difficulties: setting up special and costly hardware within facilities that are not easily transportable, requiring special teams of developers and maintenance staff [...]* » [Roussou et Drettakis 2005]

ou être composé de plusieurs capteurs placés autour de l'utilisateur [Besançon *et al.* 2016] ce qui requiert alors une calibration préalable. Dans de nombreux cas, le système de suivi ne peut donc pas être simplement transporté d'un PC à un autre.

Les approches de type « *workbench* » visent en principe à être installées directement sur les lieux de travail pour en faire bénéficier les utilisateurs. Certains *workbenches* sont d'ailleurs aujourd'hui commercialisés comme n'importe quel autre produit. Cependant, leur coût demeure très élevé. Le prix du « Personal Space Station » est ainsi évalué à environ 13 000 € [Mulder et Van Liere 2002]. Un « ImmersaDesk » coûte quant à lui plus de 90 000 €[12]. L'idée de pouvoir en installer dans chaque bureau, comme des tables à dessin dans un cabinet d'architectes, est donc actuellement difficile à envisager. Une entreprise ou un laboratoire ne peut tout au plus en acquérir que quelques exemplaires, qui seront installés dans des lieux bien précis, réduisant alors leur accessibilité pour les utilisateurs concernés. Ce type d'installation n'est pas non plus portable. Les *workbenches* cités plus haut ont la taille d'un bureau ou d'une armoire. Ce format peut éventuellement être considéré comme « transportable » d'une pièce à une autre. Mais il ne permet pas, par exemple, à un utilisateur d'emporter l'équipement à son domicile pour terminer une tâche.

Les salles immersives sont elles aussi des installations extrêmement coûteuses, bien plus encore que les *workbenches*. Le prix du système « CAVE » original est évalué à 1 800 000 €[13]. Un modèle plus récent construit en 2012, le « CAVE2 », a quant à lui coûté 830 000 €[14]. Un coût si important est associé à des équipements complexes et fragiles, qui nécessitent la présence de techniciens spécialisés et que l'on hésite à confier directement aux utilisateurs pour de « simples » tâches quotidiennes. À ce sujet, Bouvier [2009] remarque effectivement que la complexité et le coût des salles immersives les rend en pratique *moins accessibles* pour les utilisateurs qui souhaiteraient pourtant en bénéficier :

> « Il est important aussi de veiller à ce que la démesure de ces systèmes ne bloque pas les initiatives car trop complexes à utiliser ou trop chères pour en donner l'accès. On laisse probablement moins facilement ce type de matériels à des étudiants pour des projets... »

De plus, le prix de ces installations fait qu'une entreprise ou un laboratoire, lorsqu'ils peuvent se permettre d'acquérir une salle immersive, n'en possèdent normalement qu'une seule. L'espace que celles-ci occupent représente aussi un inconvénient. Leur encombrement relègue ces équipements uniques dans des salles dédiées, ce qui les éloigne mécaniquement des utilisateurs concernés. Couture et Rivière [2006] rapportent ainsi une expérience de mise en place d'une salle immersive dans un contexte professionnel. Le coût de l'installation et

12. « *More than $100,000* » [Margolis *et al.* 2011]
13. Deux millions de dollars américains actuels [Grimes 2013].
14. Le prix mentionné est de 926 000 $ [Grimes 2013].

son encombrement s'avèrent finalement avoir des conséquences pratiques sur son utilisation par les personnes concernées, qui finissent par s'en détourner :

> « Une salle immersive de réalité virtuelle a déjà été aménagée [...]. Mais la salle n'est plus utilisée. Le coût important du dispositif et la place qu'il requiert (une très grande salle dédiée à un seul usage) ne permettent pas d'en posséder plus d'une. Les utilisateurs doivent donc se conformer à un planning de réservation. [...] »

Pour compenser la faible accessibilité des salles immersives traditionnelles, plusieurs équipes ont tenté de proposer des solutions « portables » et « peu coûteuses ». Cependant, il serait plus juste de parler de solutions « transportables » et « plus abordables ». La salle immersive décrite par Ryu et Ohno [2007] occupe 3×1,3×1,5 m lors de son transport, mais requiert une étape d'assemblage (de plusieurs heures) avant être utilisable. Miller *et al.* [2005] présentent eux aussi une installation « portable ». Celle-ci peut tenir dans un véhicule, mais nécessite six personnes et deux heures pour être réassemblée une fois à destination. Il s'agit donc en pratique d'une installation seulement « transportable ». Son prix est de 30 000 €, un coût nettement inférieur à celui d'une salle immersive classique mais qui reste hors de portée de beaucoup d'utilisateurs. De Sorbier *et al.* [2009] décrivent une salle immersive là encore transportable dans un véhicule et nécessitant une étape d'assemblage. Son coût est de 7500 €, ce qui est mieux que le système précédent mais reste un prix relativement élevé.

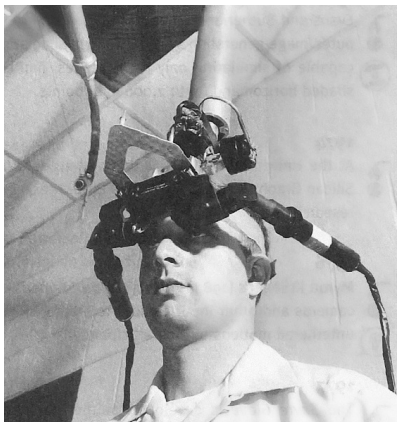
La réalité virtuelle représente donc une solution intéressante pour une manipulation 3D efficace, en amenant l'interaction dans l'espace de l'utilisateur. Une interface de réalité virtuelle peut également être non-intrusive, notamment lorsque des dispositifs tels que des *wands* ou objets tangibles sont utilisés, bien que cet objectif soit rarement une priorité. Cependant, les installations de réalité virtuelle restent aujourd'hui sous-utilisées face à d'autres plates-formes comme le PC. Cela s'explique principalement par l'emploi de matériel spécifique et/ou coûteux (manque *d'accessibilité*) et par l'absence de systèmes *véritablement portables*.

Le tableau suivant résume les avantages et inconvénients des installations fixes de réalité virtuelle par rapport à nos quatre critères :

efficacité	**portabilité**	**non-intrusivité**	**accessibilité**
++	–	+/–	– –

2.4 Systèmes de réalité virtuelle portés par l'utilisateur

Nous avons vu dans la section précédente que devoir se rendre dans une « installation » spécifique équipée de dispositifs d'affichage, de suivi et d'entrée représente un obstacle à



(a) Un des premiers exemples de visiocasques, surnommé « l'épée de Damoclès » en raison d'une imposante structure mécanique [Sutherland 1968].

(b) Le système « VIVED » [Fisher *et al.* 1987], permettant de manipuler les objets virtuels grâce à des gants instrumentés.

(c) Point de vue de l'utilisateur dans le système « VIVED ».

FIGURE 2.10 — Les dispositifs d'affichage, de suivi 3D et d'entrée peuvent tous trois être portés par l'utilisateur, réduisant ainsi l'encombrement par rapport à une installation fixe de réalité virtuelle. Ce type d'interface s'articule généralement autour d'un visiocasque (HMD).

l'utilisation de la réalité virtuelle en pratique. Afin de se passer d'un lieu dédié, d'autres systèmes choisissent *d'équiper l'utilisateur lui-même* de tels dispositifs.

Cette autre approche de la réalité virtuelle repose en premier lieu sur l'utilisation d'un visiocasque (*head-mounted display*, HMD) pour assurer l'affichage. Ce dispositif, porté par l'utilisateur, se place directement devant ses yeux et lui permet ainsi de voir des images virtuelles. Un visiocasque est généralement muni d'un système de suivi des déplacements. Le système de suivi adapte en continu l'image affichée par le visiocasque, de manière à produire l'illusion visuelle que l'utilisateur est entouré d'un environnement simulé. Il devient alors possible pour l'utilisateur d'interagir directement dans cet espace 3D, comme dans les installations citées précédemment. Par exemple, l'utilisateur peut porter des gants instrumentés afin d'interagir avec ses mains dans l'espace qui l'entoure.

L'idée de placer des équipements d'affichage et d'entrée *sur* l'utilisateur pour simuler un environnement virtuel autour de lui remonte aux premiers temps de la réalité virtuelle. L'un des premiers systèmes de ce type est probablement celui décrit par Sutherland [1968], une performance compte tenu des technologies de l'époque. Malgré un rendu visuel primitif sous forme de lignes uniquement, un suivi 3D basé sur une structure mécanique limitant fortement les déplacements de l'utilisateur (figure 2.10(a)), et l'absence de véritable interaction avec les objets virtuels, ce premier exemple de visiocasque a permis de concrétiser le concept. Un certain nombre de projets ont ensuite poursuivi et amélioré ce principe. On peut ainsi citer le système « VIVED » (*Virtual Visual Environment Display*), développé par la NASA [Fisher

et al. 1987], qui améliore la liberté de mouvement grâce à un suivi 3D magnétique et offre la possibilité de manipuler les objets virtuels grâce à des gants instrumentés (figure 2.10(b)).

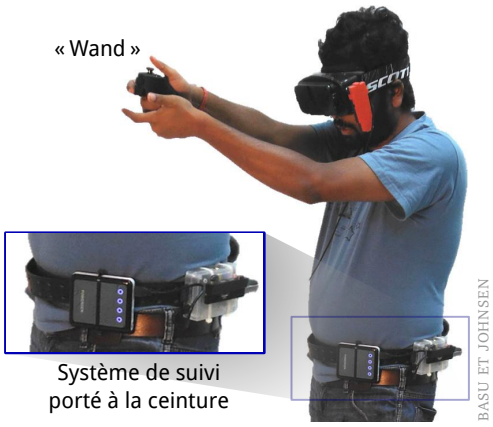
Deux approches différentes apparaissent alors. La première vise à remplacer totalement la vision de l'utilisateur par celle de l'environnement virtuel en plaçant des écrans opaques devant les yeux. On parle dans ce cas de systèmes « immersifs », comme l'est par exemple le système VIVED. La seconde vise à maintenir la vision de l'environnement réel, par-dessus lequel les objets virtuels sont donc superposés. Cela peut être accompli de manière optique (*optical see-through*) en utilisant des surfaces transparentes ou semi-transparentes [Chung *et al.* 1989], ou en équipant le visiocasque de caméras frontales (*video see-through*) qui retransmettent le flux vidéo vers un écran opaque placé devant les yeux de l'utilisateur [Rolland *et al.* 1995]. Cette seconde approche, dans laquelle l'environnement réel reste visible, est plus connue sous le nom de « réalité augmentée ».

Après une longue période pendant laquelle ces technologies sont restées relativement confidentielles, de récentes avancées technologiques ont permis à ce type de systèmes de rencontrer un véritable essor ces dernières années. L'Oculus Rift [15], dont les premiers modèles sont sortis en 2013, a démocratisé les visiocasques pour le grand public avec un prix de l'ordre de 300 € (modèle DK1). De nombreux concurrents ont suivi avec d'autres produits visant directement le consommateur : HTC Vive, Gear VR, PlayStation VR... Les visiocasques de réalité augmentée sont eux aussi en passe de devenir des produits grand public, avec notamment les Google Glass [16] (sorties en 2013) et le casque HoloLens [17]. Les systèmes basés sur des visiocasques sont donc de plus en plus *accessibles* pour les utilisateurs.

L'autre avantage de ces systèmes, par rapport aux installations de réalité virtuelle, est leur *potentielle portabilité*. Tous ne sont pas automatiquement portables car beaucoup dépendent d'équipements fixes : câbles d'alimentation ou de données, caméras externes pour le suivi... Mais plusieurs projets ont montré que l'on peut combiner un visiocasque avec un système de suivi et un dispositif d'entrée eux aussi portés par l'utilisateur. Feiner *et al.* [1997] proposent ainsi la « *Touring Machine* », un système complet de réalité augmentée tenant dans un sac à dos (figure 2.11(a)). L'utilisateur porte le sac et le visiocasque qui lui est relié, tout en interagissant avec l'espace environnant par le biais de dispositifs d'entrée tenus dans la main. Le suivi 3D est réalisé par une combinaison de capteurs d'orientation et de GPS. Il est donc possible pour l'utilisateur de se déplacer librement tout en utilisant le système, ce qui correspond à notre définition de la portabilité. Hodgson *et al.* [2012] décrivent un système similaire dans lequel l'utilisateur peut se déplacer librement à l'extérieur, en étant cette fois totalement immergé dans l'environnement virtuel. Foxlin et Harrington [2000]

15. `https://www.oculus.com/rift/`
16. `http://www.google.com/glass`
17. `http://hololens.com/`

(a) « Touring Machine » [Feiner *et al.* 1997]



(b) Visiocasque, *wand* et système de suivi entièrement portés par l'utilisateur [Basu *et al.* 2014]

FIGURE 2.11 — Les systèmes de réalité virtuelle équipant l'utilisateur peuvent être rendus portables. Ce type d'approche reste toutefois particulièrement intrusive.

présentent une approche pour suivre les déplacements 3D locaux de la main par rapport à un visiocasque (permettant ainsi la manipulation d'objets virtuels) de manière entièrement portable et autonome. Cette approche consiste à faire porter à l'utilisateur sur sa main un émetteur à ultrasons. L'émetteur est suivi par des capteurs placés sur le casque lui-même et connectés à une unité centrale portée à la ceinture. Une alternative est proposée par Basu *et al.* [2012], dans laquelle le porteur du visiocasque tient dans la main des *wands* reliés par fil à un système de suivi magnétique porté à la ceinture (figure 2.11(b)). Bien que les systèmes ci-dessus ne soient pas aussi facilement transportables qu'un simple appareil mobile, ils constituent tout de même une avancée importante par rapport aux installations essentiellement fixes décrites précédemment.

Cependant, ces approches souffrent d'un inconvénient majeur : leur *intrusivité*. Les dispositifs étant placés sur l'utilisateur, il est nécessaire de passer par une phase préalable d'équipement avant de pouvoir utiliser l'interface. Cela ne serait pas forcément un obstacle si les dispositifs en question étaient suffisamment transparents pour pouvoir être portés en permanence. Mais en l'état actuel des technologies, il est difficile d'envisager de porter un visiocasque au long terme. D'une part, la plupart des modèles ont un poids non-négligeable (490 g pour l'Oculus Rift), bien plus qu'une paire de lunettes classique, auquel il faut ajouter l'encombrement lié aux câbles de contrôle et d'alimentation. D'autre part, un visiocasque immersif coupe complètement l'utilisateur du monde extérieur (comme par exemple dans le système décrit par Hodgson *et al.* [2012]) et ne peut donc pas être porté le reste du temps. Un visiocasque de type *video see-through* offre lui une vision du monde réel, mais la latence et la résolution limitée de l'affichage par vidéo interposée font qu'un usage prolongé peut provoquer des nausées (*cybersickness*) [Lindeman 2016; LaViola 2000]. Les casques *optical see-through* sont les seuls à permettre une vision non-altérée du monde extérieur, mais l'encombrement reste

un facteur. En effet, la technologie actuelle ne permet pas encore de construire un visiocasque à la fois performant et aussi peu encombrant qu'une paire de lunettes (les « lunettes de réalité augmentée » si souvent imaginées dans la fiction). Un des modèles grand public le plus proche de cet idéal, les Google Glass, n'offrent pour l'instant qu'une image semi-transparente et qui ne couvre qu'une faible partie du champ de vision.

En plus des contraintes liées au port d'un visiocasque, il faut également prendre en compte celles causées par les autres éléments du système : les dispositifs de suivi et d'entrée. Les gants instrumentés sont ainsi des dispositifs particulièrement intrusifs car ils réduisent la sensibilité et la liberté de mouvement des mains mêmes de l'utilisateur. Cela suffit à rendre difficile des gestes aussi simples que le fait de saisir un stylo ou une tasse [Quek 1995]. Les systèmes qui combinent un visiocasque avec des gants instrumentés (tel le projet VIVED), bien que très puissants en matière de manipulation 3D, sont donc d'autant plus contraignants pour l'utilisateur (figure 2.10(b)). Dans le cas d'une interaction par le biais de *wands* suivis par un dispositif magnétique (figure 2.11(b)), les multiples câbles et le fait de devoir porter la base du dispositif à la ceinture rendent l'ensemble peu pratique à porter en permanence. Une approche un peu moins intrusive est celle de Foxlin et Harrington [2000] où un émetteur est fixé sur l'objet à suivre (en l'occurrence la main) et suivi par des capteurs situés sur le visiocasque, ce qui ne nécessite aucun câble. Une solution pour aller plus loin dans la non-intrusivité serait de placer l'émetteur sur un objet tangible pour ne rien avoir à porter sur la main, même si le casque lui-même resterait dans tous les cas intrusif.

Les approches de réalité virtuelle consistant à équiper l'utilisateur de dispositifs d'affichage (visiocasque), de suivi et d'entrée présentent donc dans l'ensemble une meilleure accessibilité et portabilité que les installations fixes, tout en offrant les mêmes possibilités d'interagir dans un véritable espace 3D. Cependant, ces systèmes sont fortement contraints par leur intrusivité, en particulier celle du visiocasque mais aussi celle des dispositifs qui lui sont habituellement associés.

Le tableau suivant résume les avantages et inconvénients des systèmes de réalité virtuelle portés par l'utilisateur pour répondre à nos quatre critères :

efficacité	**portabilité**	**non-intrusivité**	**accessibilité**
++	+	−−	+

2.5 Conclusion

Le tableau 2.1 résume les avantages et inconvénients des différents types d'approches existantes pour répondre aux critères que nous nous sommes fixés. La mention **++** indique qu'une approche donnée répond de manière satisfaisante au critère en question.

TABLEAU 2.1 — Avantages et inconvénients des approches existantes en terme de portabilité, non-intrusivité, accessibilité et efficacité pour la manipulation 3D.

		efficacité	**portabilité**	**non-intrusivité**	**accessibilité**
Plate-forme PC		– –	+/–	**++**	**++**
Plate-forme mobile		– –	**++**	**++**	**++**
RV	Installations fixes	**++**	–	+/–	– –
RV	Systèmes portés par l'utilisateur	**++**	+	– –	+

Sur la base de ces constatations, les appareils mobiles semblent représenter une solution intéressante pour répondre à la fois à nos objectifs de portabilité, de non-intrusivité et d'accessibilité. Cependant, la plate-forme mobile en tant que telle n'offre pas une bonne efficacité pour la manipulation 3D. Les systèmes de réalité virtuelle (sous forme d'installations fixes ou portés par l'utilisateur) répondent quant à eux moins bien à ces trois critères, mais autorisent une manipulation efficace en amenant l'interaction dans l'espace 3D.

Il apparaît donc pertinent de combiner le principe d'un appareil mobile avec le type d'interaction que permettent les approches de réalité virtuelle. Mais cela ne doit pas se faire au détriment des avantages de portabilité, de non-intrusivité et d'accessibilité de l'appareil mobile. Ainsi, un dispositif destiné à être porté par l'utilisateur (tel qu'un gant instrumenté), combiné à un appareil mobile, aurait pour effet de rendre l'ensemble du système intrusif. Un dispositif efficace, mais particulièrement onéreux aurait pour effet de rendre l'ensemble du système moins accessible. Un système de suivi fixe, comme dans la plupart des installations de réalité virtuelle, aurait pour effet de rendre l'ensemble du système non-portable en dépit de l'emploi d'un appareil mobile.

Dans cette thèse, nous étudions donc différentes approches visant à combiner les avantages des systèmes ci-dessus, avec pour objectif de répondre simultanément aux objectifs de portabilité, de non-intrusivité, d'accessibilité et d'efficacité de manipulation.

Chapitre 3

Manipulation par objets tangibles sur appareil mobile

3.1 Introduction

Sur la base des constatations précédentes, nous proposons dans ce chapitre une première approche visant à obtenir une interface portable, non-intrusive et accessible tout en étant efficace pour la manipulation 3D. Cette approche consiste à combiner un *appareil mobile* avec des *objets tangibles*, que l'utilisateur déplace dans l'espace réel autour de l'appareil pour agir sur les objets virtuels à manipuler.

Le choix d'un appareil mobile présente un certain nombre d'avantages pour répondre à nos objectifs. Celui-ci fournit en même temps des capacités de traitement (processeur), ce qui permet de s'affranchir d'une unité centrale externe, et une surface d'affichage (écran) nécessaire pour visualiser les objets à manipuler. La plupart des modèles disposent en plus d'une caméra intégrée qui peut servir de système de suivi optique. Ces trois fonctions indispensables peuvent donc être remplies par un unique dispositif complètement *portable*. De plus, le récent essor de ce type d'appareils (smartphones, tablettes...) fait que ces derniers sont aujourd'hui largement *accessibles* pour le grand public.

Cependant, la manipulation proprement dite ne devrait pas s'effectuer par l'intermédiaire de l'écran tactile de l'appareil mobile, car celui-ci présente d'importantes limitations comme on a pu le voir précédemment. Pour une manipulation *efficace*, il est souhaitable d'amener l'interaction dans l'espace 3D comme le permettent les systèmes de réalité virtuelle. Enfin, cette interaction devrait être *non-intrusive* afin de respecter notre quatrième critère.

Une manière de répondre à ces objectifs est d'employer des objets tangibles comme support de manipulation dans l'espace réel. En effet, les objets tangibles présentent des avantages qui complètent ceux de l'appareil mobile. Ils peuvent être manipulés efficacement sur 6 DdL, en les déplaçant et en les tournant naturellement dans l'espace réel. Ils sont non-intrusifs,

car ils peuvent être saisis à main nue sans devoir porter d'équipement spécifique. Par ailleurs, les objets tangibles sont également portables tout comme l'appareil mobile, ce qui permet au système entier de rester portable. Enfin, ils n'ont pas besoin d'être instrumentés si le système de suivi est contenu dans l'appareil mobile. Ces objets peuvent donc être totalement passifs et peu coûteux, ce qui permet au système entier de rester accessible.

L'utilisation d'objets tangibles pour la manipulation s'inspire de l'approche choisie par certaines installations de réalité virtuelle [Hinckley *et al.* 1994; Qi et Martens 2005]. Mais à la différence de ces installations, les capacités de traitement, d'affichage et de suivi sont ici fournies par l'appareil mobile. Notre approche permet donc de combiner l'efficacité et la non-intrusivité de la manipulation tangible avec la portabilité et l'accessibilité d'un appareil mobile.

3.2 Travaux existants

3.2.1 Interfaces tangibles

Le principe des « interfaces tangibles » (*Tangible User Interfaces*, ou « TUI » [Ishii 2008; Shaer et Hornecker 2010]) est de donner forme physique à des données numériques [Ishii et Ullmer 1997]. Pour cela, ce type d'interface emploie des objets physiques, appelés « objets tangibles », pour représenter ces données dans le monde réel.

Les *Graspable User Interfaces* [Fitzmaurice 1996], que l'on peut traduire par « interfaces saisissables », sont une catégorie de TUI dans laquelle les objets tangibles servent de « poignée » pour manipuler des objets virtuels. Chaque objet tangible est associé à un objet virtuel, qui suit ses translations et rotations dans l'espace réel. Les utilisateurs peuvent ainsi manipuler un objet virtuel en manipulant directement l'objet tangible correspondant. En tirant parti des capacités existantes de l'utilisateur pour la manipulation d'objets réels [Fitzmaurice 1996; Ishii 2008], cette forme d'interaction permet une manipulation 3D immédiatement efficace et naturelle.

Par ailleurs, Fitzmaurice [1996], citant Norman [1993], avance que les lois physiques qui affectent les objets tangibles peuvent assister l'utilisateur durant la manipulation. Par exemple, relâcher un objet tangible provoque sa chute sous l'effet de la gravité, et avec lui celle de l'objet virtuel correspondant. Ce type de comportement permet de mieux prévoir et interpréter les mouvements des objets manipulés.

Enfin, chaque objet tangible peut être associé à un objet virtuel différent (multiplexage spatial). L'étape de sélection (section 1.1) dans une tâche de manipulation correspond donc simplement au fait de manipuler un objet tangible plutôt qu'un autre [Fitzmaurice 1996].

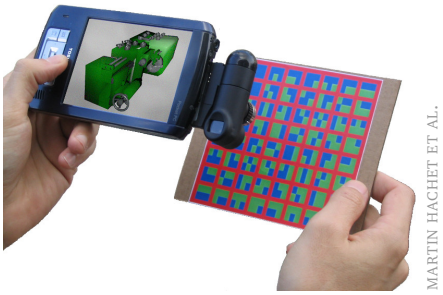
3.2.2 Interaction mobile « au-delà de l'écran »

Un certain nombre de travaux ont proposé des solutions pour amener l'interaction « au-delà » de l'écran tactile, dans l'espace 3D entourant l'appareil mobile.

Le projet « T(ether) » [Blackshaw *et al.* 2012] est une interface de manipulation d'objets sur appareil mobile, dans laquelle l'utilisateur interagit avec par le biais d'un gant instrumenté dans l'espace derrière une tablette (figure 3.1(a)). Cette approche est cependant intrusive, car elle nécessite de porter un gant. De plus, elle n'est pas réellement portable car le gant comme la tablette sont suivis par un système externe.

Une alternative au port d'un gant intrusif est d'utiliser des objets tangibles comme support de manipulation, reprenant ainsi le concept de *props* utilisé par Hinckley *et al.* [1994] ou Qi et Martens [2005] mais dans le cadre d'un appareil mobile. Certains travaux proposent ainsi d'employer des objets tangibles équipés d'aimants, en combinaison avec un capteur magnétique ajouté à l'appareil mobile. On peut citer comme exemples [Hwang *et al.* 2013] et [Liang *et al.* 2013]. Cette approche permet en théorie de suivre les objets tangibles dans tout l'espace 3D autour de l'appareil. Mais elle présente en pratique d'importants inconvénients. Premièrement, la distance de détection est limitée à seulement quelques centimètres. Deuxièmement, le suivi magnétique ne semble pas en mesure de suivre l'ensemble des degrés de liberté des objets eux-mêmes, notamment en terme d'orientation. Enfin, l'accessibilité

(a) « T(ether) » [Blackshaw *et al.* 2012]. Malgré l'utilisation d'un appareil mobile, l'interface demeure intrusive car l'utilisateur doit porter un gant. Le système de suivi externe limite par ailleurs sa portabilité.



(b) Marqueur tangible proposé par Hachet *et al.* [2005]. Ce système est réellement portable et non-intrusif. Cependant, il ne permet pas une manipulation 6 DdL.

FIGURE 3.1 — Différentes approches d'interaction « au-delà de l'écran » sur appareil mobile.

du système est fortement réduite lorsque celui-ci nécessite l'ajout d'un capteur spécifique à l'appareil mobile.

Une autre approche est le suivi optique d'objets dans l'espace entourant l'appareil mobile. Par exemple, « Portico » [Avrahami *et al.* 2011] est un système dans lequel des tiges extensibles munies de caméras sont rajoutées à une tablette, lui permettant de suivre des objets tangibles posés autour d'elle. Tous les degrés de liberté ne sont cependant pas pris en compte, ne permettant pas une manipulation 3D sur 6 DdL. L'ajout de caméras extensibles à une tablette classique rend en outre le système moins accessible.

Hachet *et al.* [2005] proposent une interface dans laquelle un marqueur tangible, tenu dans la main, est suivi par la propre caméra de l'appareil (figure 3.1(b)). Les mouvements du marqueur tangible sont utilisés pour contrôler des tâches de navigation 2D (déplacement et zoom sur une carte) ainsi que pour des tâches de manipulation 3D. À la différence de ce que nous proposons ici, leur marqueur tangible n'est suivi que sur 3 DdL et uniquement en position. La manipulation 3D doit donc être décomposée en translations et en rotations, ces dernières s'effectuant par un mouvement de translation. Cette limitation est justifiée par les auteurs sur la base de la puissance de calcul limitée des appareils mobiles de l'époque. Celle-ci a toutefois suffisamment augmenté depuis lors pour qu'un suivi 6 DdL devienne envisageable. Les auteurs mentionnent par ailleurs plusieurs points particulièrement intéressants en lien avec notre approche. D'une part, Hachet *et al.* notent que l'utilisation d'un objet tangible en combinaison avec un appareil mobile permet à l'interface d'être totalement portable. D'autre part, l'utilisation d'une caméra dirigée vers l'arrière, contrairement aux systèmes magnétiques capables de suivre des objets tout autour de l'appareil, ne semble pas représenter un inconvénient. En effet, les auteurs remarquent que le fait d'interagir à l'arrière de l'appareil mobile permet d'éviter d'occulter l'écran, ce qui est utile compte tenu de sa faible taille. Enfin, Hachet *et al.* notent que cette approche de suivi optique sur appareil mobile permet de contourner la faible extensibilité des plates-formes mobiles classiques. De fait, ce type d'approche ne nécessite rien d'autre que la caméra arrière déjà présente dans la plupart des appareils actuels. En d'autres termes, cette approche permet de se servir d'un appareil mobile classique sans le modifier, ce qui renforce son accessibilité pour les utilisateurs.

3.3 Concept

Dans notre approche, le système entier est constitué des seuls éléments suivants : un appareil mobile (muni d'une caméra), et un certain nombre d'objets tangibles. La première étape consiste à amener l'interaction au-delà de la surface de l'appareil mobile, dans l'espace réel. Pour cela, l'appareil mobile est transformé en « fenêtre » de réalité augmentée. L'image

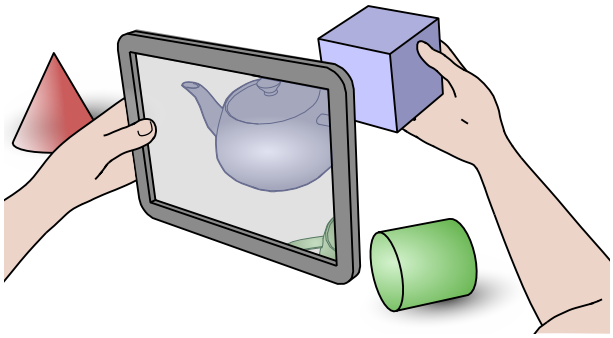


FIGURE 3.2 — Représentation schématique de notre concept. L'utilisateur manipule des objets tangibles derrière un appareil mobile. Chaque objet tangible est associé à un objet virtuel, qui suit tous ses déplacements dans l'espace 3D. En regardant « à travers » l'appareil mobile, l'utilisateur semble donc manipuler directement les objets virtuels sur 6 DdL.

captée par la caméra arrière est continuellement retransmise sur l'écran situé à l'avant. La portion d'espace réel se trouvant derrière l'appareil mobile devient donc visible « à travers » l'écran.

En parallèle, ce flux d'images est analysé par l'appareil mobile afin de détecter la présence d'objets tangibles dans ce volume d'espace. Lorsqu'un objet tangible est identifié, une représentation de l'objet virtuel associé lui est superposée sur l'écran. Le rendu de cet objet virtuel est effectué avec la même position et orientation que l'objet tangible. À travers la fenêtre de réalité augmentée, l'objet virtuel semble ainsi « prendre la place » de l'objet tangible et suivre tous ses déplacements (figure 3.2). L'utilisateur est donc en mesure de manipuler les objets virtuels sur 6 DdL *par l'intermédiaire* des objets tangibles.

3.4 Implémentation

L'appareil mobile utilisé pour implémenter cette interface doit être muni d'une caméra arrière. La plupart des smartphones et tablettes disponibles actuellement possèdent une telle caméra. Une tablette représente le meilleur choix grâce à son écran de taille plus importante, mieux adapté pour servir de fenêtre de réalité augmentée. C'est donc une tablette que nous avons retenue dans la suite de ce travail.

Le principal défi d'implémentation se trouve au niveau du suivi 3D des objets tangibles. Cette approche apporte ainsi un premier exemple de difficultés spécifiques que peuvent poser les objectifs de portabilité, de non-intrusivité et d'accessibilité. Pour respecter le critère de portabilité, ce suivi doit obligatoirement être réalisé depuis l'appareil mobile lui-même (*inside-out*

tracking) ; il n'est pas possible de se reposer sur un système de suivi externe. Ce suivi peut être réalisé de plusieurs façons : en fixant les objets sur un bras mécanique [Sutherland 1968], en les équipant d'un dispositif magnétique [Fisher *et al.* 1987] ou ultrasons [Foxlin et Harrington 2000], ou encore de manière optique [Gillet *et al.* 2004]. Or, les appareils mobiles disponibles sur le marché, et donc accessibles, ne disposent à ce jour ni de bras mécaniques, ni de dispositifs de suivi magnétique ou ultrasons. Pour respecter le critère d'accessibilité, le seul choix possible est donc celui du suivi optique, en tirant parti de la caméra arrière de l'appareil.

Une méthode courante de suivi optique est de placer un marqueur, c'est-à-dire un motif spécifique reconnaissable par la caméra, sur chaque objet à suivre. Cependant, dans notre approche, les objets tangibles sont destinés à être constamment manipulés et notamment tournés pour effectuer des rotations. La caméra utilisée pour le suivi se trouvant forcément sur l'appareil mobile (*inside-out tracking*, imposé par le critère de portabilité), le seul point de vue disponible sur les objets tangibles est celui de l'appareil lui-même. Au cours de la manipulation, le marqueur pourrait donc se retrouver dirigé à l'opposé de l'appareil mobile et devenir invisible depuis le point de vue de la caméra. Cela risquerait de provoquer une interruption du suivi et la perte de l'affichage virtuel.

Pour éviter cela, il est nécessaire de placer *plusieurs marqueurs* sur les objets tangibles afin de couvrir toutes les orientations qu'ils peuvent prendre par rapport au point de vue de l'appareil mobile (figure 3.3). Le suivi *inside-out* est alors maintenu durant toute la manipulation sans avoir besoin d'un système de suivi externe, et le système conserve ainsi sa portabilité.

Grâce au suivi optique, les objets tangibles dans notre approche peuvent être de simples objets passifs qu'il suffit de recouvrir de marqueurs. Cela contribue à la réduction de leur coût et donc à l'accessibilité du système.

3.5 Application à l'exploration de données

Afin d'évaluer l'intérêt et l'usabilité de notre approche en situation pratique, nous avons choisi d'implémenter une interface basée sur ce concept pour l'exploration de données volumiques. Par « exploration », nous entendons l'analyse interactive d'un volume de données. Cela implique non seulement de pouvoir le visualiser sous plusieurs angles, mais aussi des techniques plus poussées telles que la réalisation de coupes.

L'exploration de données est l'un des principaux domaines pouvant bénéficier d'une interface efficace pour la manipulation 3D, et constitue un cas d'application particulièrement intéressant grâce à la variété des techniques d'interaction que nécessite l'analyse de volumes de données complexes.

3.5.1 Objets tangibles

Nous avons conçu deux objets tangibles pour accompagner l'appareil mobile (la tablette) dans cette application de notre concept : *l'objet de référence* et le *stylet* (figure 3.3). Ces objets représentent le principal moyen d'interaction avec les objets virtuels dans notre interface, qu'il s'agisse de données 3D ou d'outils d'exploration.

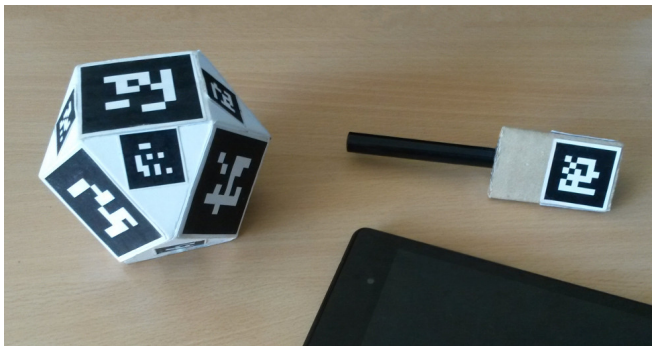


FIGURE 3.3 — Deux objets tangibles conçus pour cette première application de notre concept : l'objet de référence (à gauche) et le stylet (à droite). Les multiples marqueurs recouvrant la surface des objets tangibles permettent à la caméra de les suivre continuellement quelle que soit leur orientation. Ces objets tangibles en combinaison avec la tablette constituent l'intégralité de notre interface.

Objet de référence

L'objet de référence *représente* la donnée 3D à explorer (figure 3.5). Il constitue un « ancrage » physique et tangible pour le volume de données virtuel. Lorsque l'objet de référence est observé à travers la tablette, la donnée virtuelle apparaît en lieu et place de cet objet, dans la même position et orientation. Lorsque cet objet ou la tablette se déplacent l'un par rapport à l'autre, l'affichage est continuellement adapté pour maintenir l'illusion que l'objet et la donnée virtuelle ne font qu'un.

L'objet de référence joue deux rôles essentiels. Son premier rôle est de fournir un référentiel extérieur à l'écran. Un référentiel séparé, indépendant de l'écran, est indispensable pour amener les objets virtuels (tel que le volume de données) dans l'espace réel, autorisant ainsi une manipulation directe par l'utilisateur. À la différence des installations reposant sur un système de suivi externe et stationnaire, l'objet de référence constitue ici un *référentiel portatif*.

On pourrait penser qu'un système de suivi « sans marqueurs » de l'environnement (*markerless environment tracking*), souvent utilisé sur appareil mobile pour la réalité augmentée, serait en mesure de fournir un tel référentiel externe sans nécessiter d'objet spécifique. Mais l'objet de référence joue un deuxième rôle essentiel dans notre interface : il sert de *support physique*

pour la manipulation de la donnée. À tout moment, l'utilisateur peut agir sur cet objet pour déplacer et orienter la donnée virtuelle. Il devient donc possible de la saisir directement, à mains nues, et la manipuler comme s'il s'agissait d'un objet réel, par exemple pour l'observer sous différents angles.

La forme particulière de l'objet de référence dans notre implémentation répond à la fois à des critères fonctionnels, techniques et ergonomiques. Les données volumiques ont généralement une forme de pavé (cuboïde). De plus, leur contenu est souvent orienté de manière différente à 90° près. Une forme appropriée pour l'objet de référence est donc celle d'un cube, qui peut être facilement réorienté puis reposé sur une autre face, contrairement à une sphère par exemple. La forme cubique présente donc une « potentialité » (*affordance*, [Gibson 1977]) pour ce type d'action.

Cependant, une forme cubique présenterait un inconvénient pour le suivi d'objets à base de marqueurs. Selon son orientation, il se peut qu'une seule face du cube soit visible du point de vue de la caméra. Lorsque l'objet de référence est tenu dans la main, cette face peut être occultée par les doigts, interrompant alors le suivi. Nous avons donc choisi la forme la plus proche d'un cube qui ne présente pas cet inconvénient : le « cube tronqué », ou cuboctaèdre. Ce polyèdre possède des faces carrées qui correspondent aux faces du cube d'origine, et permettent donc toujours de tourner la donnée par intervalles de de 90°. Les faces triangulaires servent à maintenir le suivi lorsque les faces carrées sont masquées par les doigts (figure 3.4), un problème exacerbé par le fait que l'unique caméra, forcément intégrée à l'appareil mobile dans cette configuration portable, n'offre qu'un seul point de vue sur l'objet. Bien que ces marqueurs secondaires soient plus petits, ils ne sont amenés à prendre le relais que lorsque l'objet est tenu dans la main. Dans ce cas, la tablette étant tenue dans l'autre main, l'objet de référence se trouve nécessairement proche de la caméra. La plus petite taille des marqueurs dans les faces triangulaires ne constitue donc pas un inconvénient.

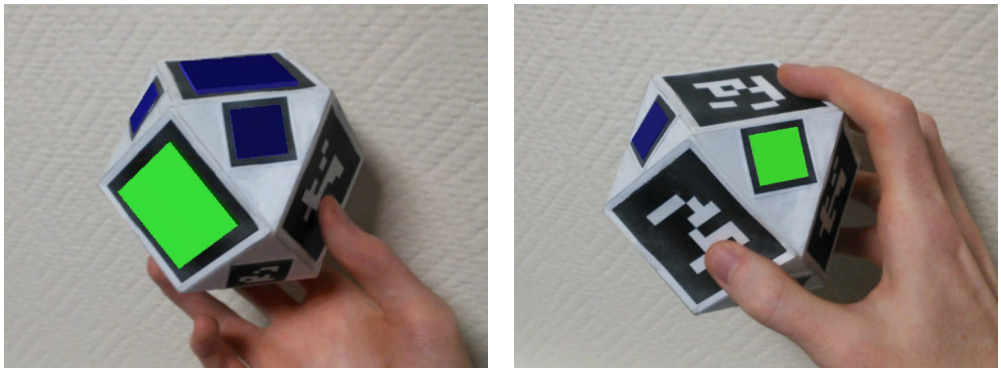


FIGURE 3.4 — La forme particulière de l'objet de référence, un « cuboctaèdre » plutôt qu'un simple cube, offre plus de faces sur lesquelles placer des marqueurs. Cela réduit le risque que tous les marqueurs soient masqués par les doigts depuis le point de vue de l'appareil mobile.

Stylet

Le stylet (figure 3.3) représente un outil d'exploration. Il est destiné à agir sur la donnée virtuelle, représentée par l'objet de référence. Sa forme allongée « suggère » son utilisation dans notre interface : tel un stylo pour des tâches de pointage, ou tel un couteau pour définir des coupes dans la donnée. Le stylet tangible profite donc de la dextérité acquise dans l'utilisation de ces outils réels pour la manipulation des outils virtuels.

Un problème inhérent aux objets tangibles est le fait que le stylet ne peut pas traverser l'objet de référence. Afin de rendre possible les tâches nécessitant de pointer à l'intérieur du volume de données, l'extrémité virtuelle de l'outil d'exploration est décalée vers l'avant par rapport à l'extrémité physique du stylet.

3.5.2 Techniques d'interaction

Le contexte d'exploration de données volumiques offre l'opportunité d'appliquer notre concept à différentes techniques d'interaction. Nous présentons ici plusieurs exemples tirant parti des particularités de notre approche.

Visualisation de la donnée

La première technique d'interaction est le fait de visualiser le volume de données, ce qui implique de l'observer sous différentes positions et différents angles. Pour cela, l'utilisateur peut manipuler soit l'objet de référence, soit la tablette, ou encore manipuler les deux simultanément. Le résultat est visible « à travers » la tablette, le volume de données suivant à chaque instant la position et l'orientation de l'objet de référence (figure 3.5).

Plans de coupe

Dans une interface d'exploration de données, la création et la manipulation de coupes planaires constitue une tâche importante dans l'analyse d'un volume de données. La combinaison d'un appareil mobile avec des objets tangibles autorise plusieurs possibilités pour manipuler de tels plans de coupe. Nous décrivons ici deux techniques reposant sur les particularités de notre interface, inspirées de métaphores différentes. Une évaluation de ces deux techniques est présentée en section 3.7.

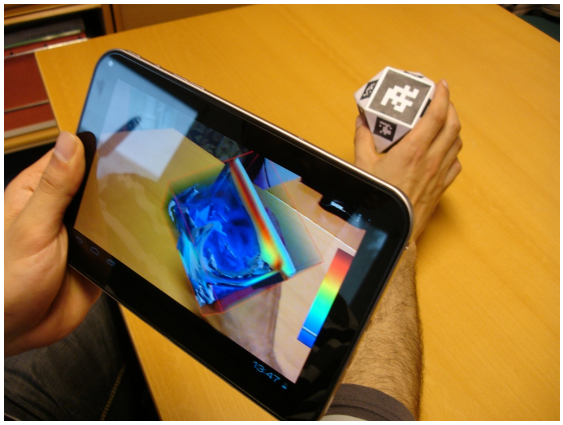


FIGURE 3.5 — Visualisation et manipulation sur 6 DdL du volume de données dans notre interface. L'utilisateur manipule la donnée virtuelle à travers l'objet de référence et observe le résultat à travers la tablette, comme si la donnée était réellement dans sa main.

Coupe contrôlée par le stylet — La première technique de coupe repose sur l'utilisation du stylet. La métaphore employée est celle d'une « lame » virtuelle qui serait attachée au stylet tangible. Cette lame définit un plan de coupe dans le volume de données. La coupe est mise à jour en temps réel sur l'écran, ce qui produit l'illusion de « couper » effectivement la donnée virtuelle avec un outil tenu dans la main (figure 3.6(a)).

Coupe contrôlée par la tablette — La seconde technique revient à considérer la *tablette* comme un objet tangible. En effet, celle-ci a une forme planaire et sa position et orientation par rapport à la donnée sont connues (grâce à l'objet de référence). Par conséquent, la tablette peut être vue comme un « plan tangible » directement contrôlé par l'utilisateur dans un espace 3D.

Dans ce mode, la surface de la tablette définit un plan de coupe dans le volume de données. Tout plan souhaité peut être obtenu en positionnant la tablette et l'objet de référence l'un par rapport à l'autre. Le résultat de la coupe est visible sur l'écran et mis à jour en continu. Ce processus est illustré par la figure 3.6(b).

Cependant, cette métaphore présente une limitation : la tablette et l'objet de référence ne peuvent s'intersecter [1]. Une stricte interprétation de cette métaphore risquerait donc de rendre certains plans de coupe inaccessibles. Une solution est de décaler le plan virtuel de plusieurs centimètres derrière la tablette. On retrouve ainsi la métaphore de « saisie d'objet »

1. Contrairement à l'approche présentée par Qi et Martens [2005] qui consiste en un « cadre » tangible pouvant « intersecter » le volume de données, la tablette est ici forcément un objet plein.

proposée précédemment par Henrysson *et al.* [2005] et Marzo *et al.* [2014], dans laquelle un objet 3D (ici le plan de coupe) est fixé à une certaine distance derrière l'appareil mobile.

(a) Plan de coupe contrôlé par le stylet. (b) Plan de coupe contrôlé par la tablette.

FIGURE 3.6 — Deux techniques de manipulation de plans de coupe tirant parti des particularités de notre concept : la première transforme le stylet en « lame virtuelle », tandis que la seconde utilise l'appareil mobile lui-même comme un plan tangible.

Un aspect particulièrement intéressant de cette seconde technique est qu'elle exploite l'appareil mobile lui-même comme « objet tangible », permettant ainsi de se passer d'un objet tangible distinct pour cette tâche.

Pointage 3D

Le fait d'amener l'interaction hors de l'écran offre la possibilité de réaliser des tâches de pointage 3D directement dans l'espace réel, en utilisant le stylet comme pointeur. Deux exemples sont présentés ici : la construction d'isosurfaces à partir d'un point et le lâcher de particules.

Construction d'isosurfaces — Pour faciliter l'exploration d'un champ scalaire, il peut être utile de construire une isosurface, c'est-à-dire une surface définie par tous les points partageant une même valeur. La plupart des logiciels proposent à l'utilisateur d'entrer lui-même la valeur souhaitée. Cependant, dans un contexte exploratoire, cette valeur n'est pas nécessairement connue. Il peut donc être souhaitable de générer une isosurface sur la base des structures visibles dans le volume.

Dans notre interface, le stylet permet à l'utilisateur de directement pointer vers une position 3D dans le volume (figure 3.7), puis de construire une isosurface à partir de la valeur scalaire à cette position.

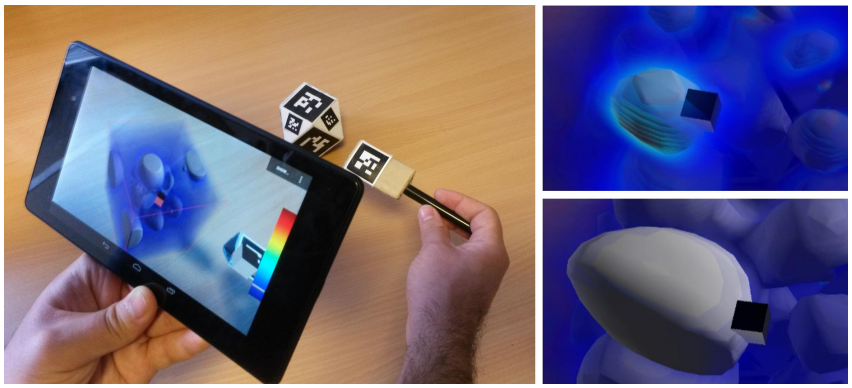


FIGURE 3.7 — Construction d'isosurfaces à partir d'une valeur scalaire. Notre interface permet à l'utilisateur de pointer directement la valeur souhaitée dans le volume, par l'intermédiaire du stylet.

Lâcher de particules — Certaines données sont en réalité des champs de vecteurs, par exemple un champ de vitesses. Le lâcher de particules est une technique pour explorer interactivement de telles données. Celle-ci consiste à générer un nuage de particules autour d'un emplacement donné dans le volume. La trajectoire de chaque particule est déterminée par le champ de vecteurs, comme si elle était entraînée par le courant. Le déplacement des particules permet alors d'identifier visuellement les structures dans le champ. Une technique similaire, appelée « ensemencement », est utilisée pour analyser de véritables écoulements (physiques) en y injectant de petites particules telles que de la fumée.

Ici, le stylet peut être utilisé comme source de particules. L'extrémité du stylet définit la position autour de laquelle les particules sont générées, et peut être directement positionné dans le volume de données en observant à travers la fenêtre de réalité augmentée (figure 3.8). Le geste devient ainsi équivalent à celui de l'ensemencement, à ceci près que le champ est virtuel.

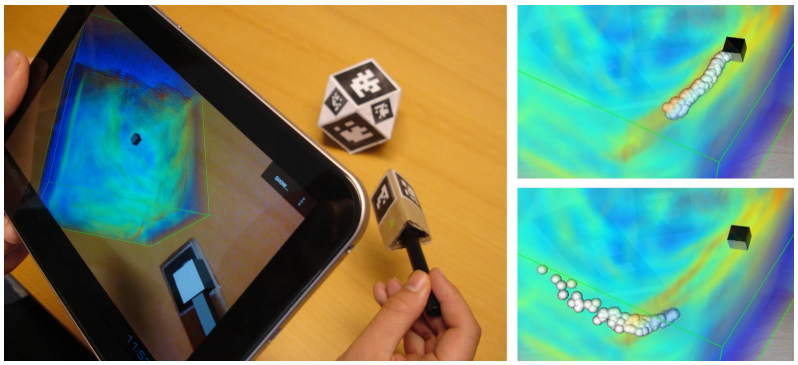


FIGURE 3.8 — Lâcher de particules dans notre interface, permettant d'explorer des champs vectoriels. Le stylet joue ici le rôle d'un outil d'ensemencement virtuel, tout en étant directement manipulable par l'utilisateur.

3.6 Première étude : manipulation 3D

Après avoir décrit notre concept d'interface et son application à l'exploration de données, nous présentons dans la suite de ce chapitre deux études visant à évaluer expérimentalement l'usabilité de cette interface.

Le but de cette première étude est de vérifier la pertinence de l'interaction tangible pour permettre une manipulation 3D efficace sur appareil mobile. Pour justifier ce choix, nous comparons notre approche tangible à une technique utilisant l'écran tactile de la tablette, qui représente le mode d'interaction « standard » sur appareil mobile. Nous comparons également notre approche à une « souris 3D », un dispositif d'interaction 6 DdL disponible dans le commerce. En tant que dispositif spécifiquement conçu pour la manipulation 3D, la souris 3D nous sert de référence pour évaluer l'efficacité de notre approche. Ces trois modes d'interaction sont comparés pour une même tâche de manipulation au travers de mesures de performance et d'usabilité.

3.6.1 Conditions

La première condition de cette expérience (**C1**) consiste à utiliser « l'objet de référence » pour manipuler un objet 3D de manière tangible, comme cela a été décrit précédemment.

La deuxième condition (**C2**) consiste à utiliser l'écran tactile multipoints de la tablette pour translater et tourner un objet 3D. À ce jour, la modalité tactile reste le mode d'interaction standard sur dispositifs mobiles, y compris dans les logiciels de manipulation 3D. Comme nous l'avons vu dans le chapitre 2, la technique RST (*rotate-scale-translate*) est la plus répandue dans les logiciels existants [Besançon *et al.* 2016]. Nous avons donc implémenté cette technique pour cette seconde condition, de manière identique à celle que l'on peut trouver dans KiwiViewer [2], un logiciel de visualisation scientifique pour tablettes et smartphones.

La troisième condition (**C3**) consiste à utiliser une « souris 3D », plus précisément un modèle SpaceNavigator [3] (figure 3.9). Il s'agit d'un dispositif d'interaction 6 DdL, similaire à un joystick, conçu pour remplacer ou compléter la souris classique pour une manipulation 3D efficace. Bien qu'un tel dispositif ne soit pas forcément le plus adapté pour accompagner un appareil mobile, et ne réponde pas particulièrement à nos objectifs de portabilité et d'accessibilité, il s'agit malgré tout d'un dispositif commercial spécifiquement conçu pour l'interaction 3D. De plus, il peut être directement connecté à la tablette à des fins de compa-

2. `http://www.kiwiviewer.org/`

FIGURE 3.9 — Le SpaceNavigator, un modèle de souris 3D utilisé dans cette expérience.

raison avec les autres conditions. Nous avons donc choisi d'utiliser ce dispositif comme point de référence pour mesurer l'efficacité de notre approche tangible.

Les trois conditions de cette étude sont donc les suivantes :
— **C1** : manipulation tangible à travers « l'objet de référence » ;
— **C2** : manipulation par écran tactile (technique RST) ;
— **C3** : manipulation par souris 3D.

3.6.2 Hypothèses

La raison nous ayant conduit à préférer l'interaction tangible à l'interaction tactile, qui est pourtant la modalité standard sur appareil mobile, est que l'interaction tactile ne permet pas une manipulation 6 DdL intégrée. En l'occurrence, la technique RST est limitée à 4 DdL simultanés, par ailleurs confondus entre translations et rotations. Notre première hypothèse **H1** est donc que la condition tangible offre une meilleure efficacité que la condition tactile pour la manipulation 3D.

Notre deuxième hypothèse est que l'interaction tangible est en mesure d'offrir une efficacité comparable à celles d'autres interfaces dédiées à la manipulation 3D (avec l'avantage supplémentaire de permettre à notre interface d'être portable, accessible et non-intrusive). Nous faisons donc l'hypothèse **H2** que la condition tangible présente une efficacité au moins comparable à celle de la condition souris 3D.

— **H1** : la condition tangible est plus efficace que la condition tactile pour la manipulation 3D ;
— **H2** : la condition tangible est au moins aussi efficace que la condition souris 3D.

3. `http://www.3dconnexion.eu/products/spacenavigator-for-notebooks.html`

3.6.3 Participants

Un groupe de 12 participants (2 femmes, 10 hommes), âgés de 20 à 35 ans (moyenne=23,3, σ=4,0) ont été recrutés pour cette expérience. Tous étaient droitiers. Parmi ces participants, 10 ont déclaré se servir fréquemment d'un smartphone et 5 utiliser régulièrement une tablette. Cinq d'entre eux étaient des utilisateurs réguliers de logiciels 3D. Un des participants avait une certaine expérience avec la réalité augmentée. Deux participants s'étaient déjà servi d'une souris 3D, sans pour autant en être des utilisateurs réguliers.

3.6.4 Plateforme expérimentale

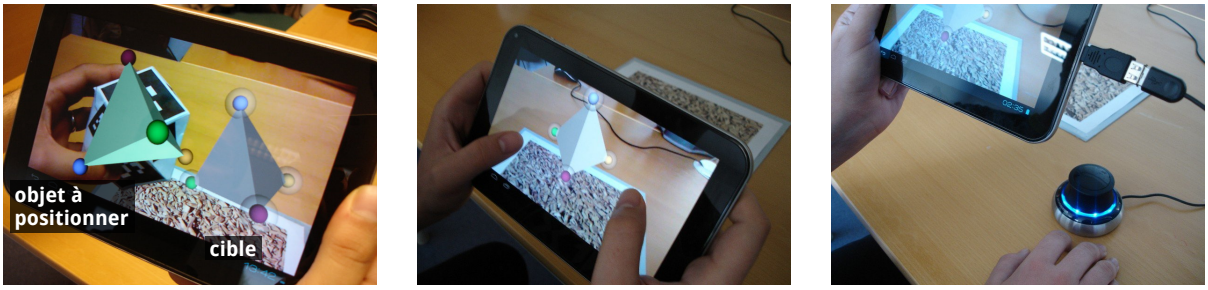
L'expérience est implémentée sur une tablette Toshiba AT270, d'une taille de 7 pouces et d'un poids de 332 g. Cette tablette sert de « fenêtre » de réalité augmentée, comme expliqué précédemment. Le taux de rafraîchissement est de 15 images par seconde. Afin d'évaluer les trois conditions dans le même contexte, toute l'expérience se déroule sur la même tablette. La souris 3D est connectée directement à la tablette durant la condition correspondante. Les trois conditions consistent à manipuler un objet 3D en réalité augmentée (figure 3.10).

Pendant l'expérience, les participants sont assis devant une table. Sur la table se trouve un marqueur texturé servant de référentiel pour la tâche de manipulation [4]. L'objet de référence et la souris 3D sont également sur la table, à portée de main. Les participants sont invités à tenir la tablette dans une main (gauche ou droite selon leur préférence) et à manipuler les dispositifs d'interaction avec l'autre main. Les participants doivent rester assis durant toute l'expérience, mais sont autorisés à déplacer librement la tablette pour changer de point de vue sur la scène virtuelle.

3.6.5 Procédure et tâche

La tâche expérimentale, inspirée de travaux précédents sur la manipulation 3D [Zhai et Milgram 1998; Hancock *et al.* 2007], consiste à aligner un objet virtuel (un tétraèdre) avec une cible (un tétraèdre de même taille), c'est-à-dire faire correspondre leur position et leur orientation. Ce type de tâche est appelé « amarrage » (*docking*). L'objet et la cible sont tous deux visibles en réalité augmentée « à travers » la tablette. L'objet lui-même est contrôlé par

4. Dans une utilisation normale de l'interface, un tel marqueur fixe n'est pas nécessaire : le référentiel portable que constitue l'objet de référence serait suffisant. Mais pour évaluer une tâche de manipulation impliquant l'objet de référence lui-même, il est nécessaire que la cible à atteindre soit indépendante de cet objet, d'où le besoin d'un référentiel supplémentaire. Lors d'un usage normal de l'interface, la « cible » des manipulations de l'utilisateur n'existe que dans son intention et n'a donc pas besoin d'être représentée visuellement.



C1 : Objet tangible **C2** : Écran tactile **C3** : Souris 3D

FIGURE 3.10 — Illustration des trois conditions de cette expérience. La condition **C1** représente notre concept d'interface tangible et mobile. La condition **C2** consiste à utiliser l'écran tactile pour manipuler l'objet 3D. La condition **C3** consiste à utiliser une souris 3D pour contrôler l'objet manipulé.

le participant par le biais de la méthode d'interaction à évaluer (**C1**, **C2** ou **C3**). La cible n'est pas contrôlable, et reste fixe au-dessus du marqueur texturé dans une position et orientation donnée (différente à chaque essai). Pour aider les participants à distinguer les deux objets, l'objet manipulé a une teinte différente (figure 3.10, image de gauche).

Bien que l'objet de référence serve normalement à manipuler un volume de données dans notre interface, nous avons préféré utiliser un tétraèdre comme « donnée 3D » générique. Le choix d'un objet simple permet d'éviter que d'éventuelles structures ou effets de transparence dans un véritable volume de données ne compliquent la tâche d'amarrage avec la cible. Par ailleurs, à la différence de précédents travaux [Zhai et Milgram 1998], les tetrahèdres sont ici *solides* au lieu d'être affichés sous forme de « fil de fer ». En effet, des essais préliminaires ont montré qu'un tétraèdre en fil de fer ne présentait pas suffisamment d'indices de profondeur en l'absence d'affichage stéréoscopique (comme c'est le cas sur la majorité des tablettes tactiles, dont celle que nous utilisons ici) et que cela pouvait gêner le déroulement de la tâche. Dans notre cas, l'ombrage solide et l'occultation entre l'objet et la cible fournissent les indices de profondeur nécessaires.

Les sommets des deux tétraèdres sont indiqués par des sphères colorées. Pour accomplir la tâche, les participants doivent aligner les sommets de l'objet manipulé avec les sommets *correspondants* de la cible, ce qui nécessite donc d'atteindre à la fois la même position et la même orientation. Une distance de tolérance est représentée par un « halo » autour de chaque sommet de la cible. Lorsque deux sommets correspondants sont suffisamment proches, ce halo devient bleu. Lorsqu'au moins trois sommets sont alignés, la tablette vibre pour indiquer que l'alignement est bon. Cet état doit être maintenu pendant 1 seconde pour que l'essai soit validé (pour éviter une éventuelle validation par chance en manipulant l'objet au hasard). Un message de succès est alors affiché sur l'écran.

Les trois conditions sont évaluées l'une à la suite de l'autre, dans un ordre différent entre chaque participant pour compenser de possibles effets d'apprentissage. Chaque condition commence par 2 essais d'entraînement, suivis de 8 essais réels. Les essais réels sont présentés dans un ordre aléatoire depuis un ensemble de 8 configurations prédéfinies, dans lesquelles la cible est distante de 2,5 cm à 13,5 cm d'une *position de base* (à 10 cm au-dessus du plateau) et de 40° à 180° d'une *orientation de base* (arbitraire). Les deux configurations d'entraînement sont les mêmes dans chaque condition : une tâche « simple » où la cible est placée à une distance et un angle faibles de l'emplacement de base et une tâche « complexe » où la cible est placée à une distance et un angle plus importants. Au début de chaque essai, l'objet manipulé est ramené à la position et l'orientation de base. Dans la configuration tangible, le participant doit ramener lui-même l'objet de référence à cet emplacement de base pour que la tâche commence. Il y a une limite de temps de 2 minutes pour terminer chaque essai, et un délai de 10 secondes entre les essais. Les participants sont autorisés à faire une pause entre chaque condition.

3.6.6 Mesures

La principale mesure objective est le temps d'accomplissement de la tâche. À la fin de l'expérience, les participants complètent également un questionnaire dans lequel ils doivent classer les trois conditions selon plusieurs critères subjectifs : vitesse ressentie (pour accomplir la tâche), précision ressentie, fatigue et préférence globale.

3.6.7 Résultats et discussion

Temps d'accomplissement de la tâche

Comme cela est souvent le cas pour ce type de mesure, les temps obtenus ne suivaient pas une distribution normale (Shapiro-Wilk : $W=0{,}53$, $p \ll 0{,}01$) mais plutôt une distribution log-normale (Kolmogorov-Smirnov : $D=0{,}04$, $p=0{,}78$). Nous avons donc appliqué une transformation *log* aux temps mesurés [5]. Les variances des valeurs transformées étant homogènes (Levene : $F=0{,}42$, $p=0{,}66$), remplissant ainsi les prérequis pour l'utilisation de tests paramétriques. Les valeurs mentionnées dans le texte et dans les figures restent non transformées.

La figure 3.11 montre les temps d'accomplissement de la tâche pour chaque condition. Un test ANOVA révèle une influence statistiquement significative de la condition sur le temps

5. Un exemple d'utilisation de cette procédure peut être trouvé dans Kane *et al.* [2008].

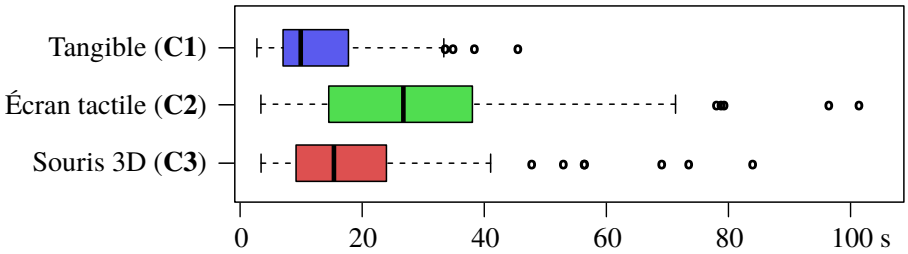


FIGURE 3.11 — Temps d'accomplissement de la tâche (secondes) dans l'expérience de manipulation d'objets. Les barres verticales représentent les médianes de chaque groupe [6].

d'accomplissement de la tâche (F(2,22)=40,02, p<0,01). Des tests *t* post-hoc deux-à-deux (*pairwise post-hoc t-tests*), avec une correction de Holm, montrent une différence statistiquement significative entre les temps moyens de chaque condition (p<0,01 dans chaque cas). Nous pouvons donc constater que les participants ont accompli la tâche le plus rapidement avec l'objet tangible (médiane=8,7 s, σ=8,5 s), suivi de la souris 3D (médiane=13,7 s, σ=29,7 s), l'écran tactile ayant montré les moins bonnes performances (médiane=24,2 s, σ=27,54 s).

Ce résultat confirme notre hypothèse **H1** que l'interaction tangible est plus efficace que l'interaction tactile pour cette tâche de manipulation 3D. Pour la plupart des participants, il s'agissait là de leur première expérience avec l'interaction tangible et la souris 3D, tandis qu'une grande majorité d'entre eux étaient déjà habitués à l'interaction tactile sur smartphone et/ou tablette. Malgré cela, les participants ont tout de même mis plus de temps à accomplir la tâche avec l'écran tactile qu'avec les autres conditions. Ce premier résultat montre donc que l'interaction tangible constitue une amélioration par rapport au mode d'interaction tactile traditionnel sur appareil mobile, pour les tâches de manipulation 3D.

L'objet tangible s'est également révélé plus performant que la souris 3D pour cette tâche. Cela montre que son efficacité est *au moins* comparable à celle d'un dispositif déjà reconnu et conçu pour l'interaction 3D, confirmant notre deuxième hypothèse **H2**. La moindre performance de la souris 3D peut trouver une explication dans le fait qu'il s'agit d'un dispositif isométrique à contrôle en vitesse (*isometric rate control*), contrairement à l'objet tangible qui est un dispositif isotonique à contrôle en position (*isotonic position control*). Une précédente étude par Zhai [1995] a montré que le premier type de dispositif présentait de moins bonnes performances que le second dans des tâches d'amarrage 3D, en l'absence de véritable entraînement des participants (dans notre expérience, les deux tâches d'entraînement ne permettaient que de se familiariser avec chaque dispositif). D'après Zhai, cette différence disparaîtrait après 20 minutes d'entraînement sur le dispositif isométrique. Mais

6. La valeur médiane est plus représentative que la valeur moyenne dans ce type de distributions asymétriques.

dans tous les cas, nos résultats montrent que l'interaction tangible présente effectivement des performances comparables à celles d'un autre dispositif dédié à la manipulation 3D.

Résultats subjectifs

La figure 3.12 montre les résultats du questionnaire destiné à comparer les trois conditions selon des critères subjectifs.

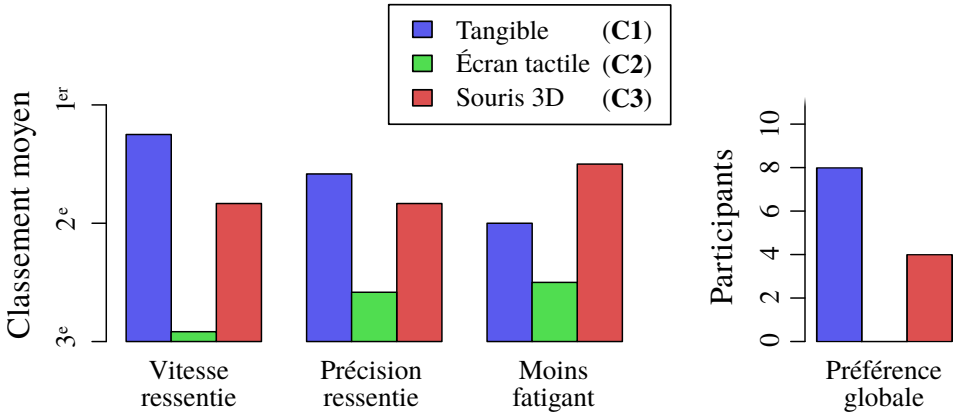


FIGURE 3.12 — Résultats du questionnaire de la première expérience, montrant l'ordre dans lequel les participants ont classé les trois conditions selon plusieurs critères subjectifs.

En terme de vitesse ressentie pour accomplir la tâche, nous retrouvons sans surprise un résultat similaire à celui des mesures objectives. Les participants ont globalement perçu l'objet tangible comme permettant d'accomplir plus rapidement la tâche. L'écran tactile a été presque systématiquement considéré comme le plus lent. Un test de Kendall montre une bonne cohérence parmi les classements des trois techniques par les participants (W=0,72, χ^2=17,2, p<0,01).

Pour la précision ressentie, l'objet tangible bénéficie d'un classement comparable, voire légèrement supérieur à celui de la souris 3D, alors que l'écran tactile est nettement en troisième position. Nous aurions pu nous attendre à ce que l'objet tangible, un dispositif isotonique tenu à bout de bras, soit perçu comme moins précis que les autres modes d'interaction. Cependant, les participants semblent avoir retenu la sensation de contrôle pour noter ce critère, un aspect pour lequel un dispositif à contrôle de position tel que l'objet tangible est mieux adapté qu'un dispositif à contrôle de vitesse comme la souris 3D (susceptible de provoquer un dépassement de la cible [Zhai 1995]), et mieux adapté qu'un contrôle non-6 DdL comme l'écran tactile. Les réponses des participants sont ici légèrement moins cohérentes que pour la vitesse (Kendall : W=0,27, χ^2=6,5, p<0,05).

En terme de fatigue, la souris 3D a été préférée (c'est-à-dire considérée comme *moins fatigante*) à l'objet tangible, tandis que l'écran tactile est classé dernier. Il existe toutefois un

certain désaccord entre les réponses des participants (Kendall : W=0,25, χ^2=6,0, p≈0,05). Il n'est pas surprenant qu'un dispositif isotonique (tel que l'objet tangible) soit perçu comme plus fatigant qu'un dispositif isométrique (tel que la souris 3D). Les autres avantages de l'interaction tangible peuvent cependant compenser cet inconvénient. Le score bien inférieur de l'écran tactile est lui plus inattendu, mais pourrait s'expliquer par le temps nettement supérieur nécessaire pour accomplir la tâche.

La préférence globale va à l'objet tangible pour les deux tiers des participants, ce qui constitue un résultat très encourageant quant à l'acceptation de ce type d'interface face aux plateformes actuellement dominantes.

3.7 Deuxième étude : plans de coupe

L'objet de cette deuxième étude est d'évaluer l'efficacité et l'usabilité de notre approche de manipulation tangible pour une tâche plus complexe, allant au-delà du seul positionnement de la donnée 3D. Cette tâche consiste à contrôler un outil d'exploration : le plan de coupe. Effectuer des coupes dans le volume est un aspect essentiel dans l'exploration d'une donnée 3D, mais il s'agit également d'un outil pouvant particulièrement bénéficier d'une manipulation 3D efficace. Qui plus est, cette tâche nécessite d'interagir avec plusieurs objets virtuels (le volume de données et l'outil de coupe). Cette étude vise donc à évaluer notre approche dans des conditions plus proches d'une utilisation réelle.

3.7.1 Conditions

La première condition (**C1**) consiste à utiliser le stylet (figure 3.6(a)) pour contrôler le plan de coupe dans la donnée virtuelle, représentée par l'objet de référence. Cette condition représente le fait d'utiliser un des objets tangibles pour agir sur un autre objet dans l'espace entourant l'appareil mobile, une configuration similaire à celle d'un dispositif « wand » en réalité virtuelle.

La deuxième condition (**C2**) consiste à utiliser la tablette pour contrôler le plan de coupe (figure 3.6(b)). Cette condition représente le fait d'utiliser l'appareil mobile *lui-même* comme objet tangible pour interagir dans l'espace 3D.

Enfin, pour servir de base avec laquelle comparer les performances et l'usabilité des techniques ci-dessus, la troisième condition (**C3**) consiste à utiliser une souris 2D classique pour contrôler le plan de coupe. La souris 2D représente le dispositif d'interaction standard pour manipuler des plans de coupe dans les logiciels PC, la plateforme la plus utilisée actuel-

lement. Notre objectif est de proposer une interface portable, accessible et non-intrusive offrant également une meilleure efficacité que la plateforme PC. La souris 2D constitue donc une comparaison pertinente pour cette étude.

Pour nous assurer que cette dernière condition soit représentative, nous avons reproduit à l'identique la technique d'interaction utilisée dans ParaView[7], un logiciel de visualisation scientifique largement utilisé sur PC. Déplacer la souris tout en maintenant le bouton droit appuyé permet d'orienter le plan de coupe (rotation « arcball »). Déplacer la souris en maintenant le bouton gauche appuyé permet de translater le plan de coupe le long de sa normale.

Les trois conditions de cette étude sont donc :
— **C1** : plan de coupe contrôlé par le stylet ;
— **C2** : plan de coupe contrôlé par la tablette ;
— **C3** : plan de coupe contrôlé par une souris 2D.

3.7.2 Hypothèses

Notre principale hypothèse est que les deux techniques tangibles (**C1** et **C2**), basées sur une manipulation 6 DdL directe dans un espace 3D, sont plus efficaces que la condition souris 2D (**C3**) pour cette tâche.

Mais au-delà de leur performance, cette étude vise également à évaluer les avantages et inconvénients en terme d'usabilité des deux techniques tangibles proposées.

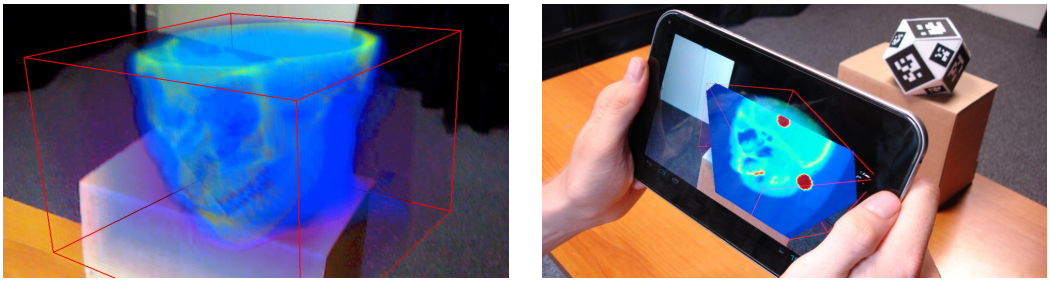
3.7.3 Participants

Un groupe de 9 participants (8 hommes, 1 femme), âgés de 23 à 29 ans (moyenne=25,0, σ=1,7) ont été recrutés pour cette expérience. Un des participants était gaucher.

3.7.4 Plateforme expérimentale

Comme dans l'expérience précédente, la tablette utilisée est un modèle Toshiba AT270, d'une taille de 7 pouces, servant de fenêtre de réalité augmentée (à 15 images par seconde).

7. `http://www.paraview.org/`



(a) Le volume de données à explorer, associé à l'objet de référence. L'objet de référence est ici posé sur une boîte de manière à être surélevé par rapport à la table.

(b) Déroulement de l'expérience. Les participants doivent localiser les trois cibles à l'intérieur de la donnée en manipulant des plans de coupe (exemple tiré de la condition **C2**).

FIGURE 3.13 — Illustration des deux conditions de la deuxième expérience, portant sur la manipulation de plans de coupe.

Les participants sont assis devant une table. Sur la table se trouvent l'objet de référence et une boîte en carton (16 cm × 12 cm × 12 cm). Cette boîte permet de surélever l'objet de référence, afin d'augmenter l'espace d'interaction en dessous du volume de données sans être limité par la surface de la table (figure 3.13). Notons que cela ne constitue pas un écart à notre concept d'interface, puisque rien n'empêche les utilisateurs de tirer parti de leur environnement dans une utilisation réelle. L'objet de référence est initialement placé sur la boîte, mais les participants sont expressément autorisés (et même encouragés) à le déplacer ailleurs s'ils le souhaitent, puisqu'il s'agit d'un aspect essentiel de notre concept.

Dans la condition souris 2D, une souris sans fil ambidextre est placée sur la table et connectée en Bluetooth à la tablette.

3.7.5 Procédure et tâche

Une donnée volumique (une tomographie crânienne) est chargée dans l'interface et associée à l'objet de référence (figure 3.13(a)). À l'intérieur de cette donnée, 3 cibles sphériques (⌀ 1 cm) sont générées. Leur couleur contraste fortement avec le reste de la donnée afin de les rendre immédiatement identifiables (figure 3.13(b)). Ces cibles sont invisibles depuis l'extérieur du volume, et ne peuvent être vues qu'en effectuant des coupes. Le plan de coupe est contrôlé par la technique à évaluer.

Les participants doivent *trouver les trois cibles* dans le volume grâce à des plans de coupe, puis *positionner le plan de coupe pour intersecter les trois sphères*. Cet état doit être maintenu pendant 0,8 s pour valider la tâche. Il y a une limite de temps de 5 minutes sur chaque essai.

Les trois conditions sont évaluées l'une après l'autre. Pour compenser d'éventuels effets d'apprentissage, l'ordre de présentation des conditions est permuté entre chaque participant.

Un ensemble de 18 configurations de cibles (triplets de positions) a été généré avant l'expérience, et vérifié pour exclure les cas aberrants (tels que des cibles superposées, trop proches l'une de l'autre, ou sur les bords extrêmes du volume). Tous les participants effectuent les mêmes 18 tâches, présentées dans un ordre aléatoire. Chaque bloc de 6 tâches consécutives est réalisé avec une condition différente. Chacun de ces blocs est précédé de 2 tâches d'entraînement. Les participants sont autorisés à effectuer une pause entre chaque bloc.

3.7.6 Mesures

La principale mesure objective est le temps d'accomplissement de la tâche. À la fin de l'expérience, les participants remplissent un questionnaire dans lequel ils doivent noter chaque technique selon la fatigue ressentie et leur préférence globale. Ils sont également invités à résumer, par des mots, les aspects positifs et négatifs de chaque technique.

3.7.7 Résultats et discussion

Temps d'accomplissement de la tâche

Comme pour l'expérience précédente, les temps obtenus ne suivaient pas une distribution normale (Shapiro-Wilk : W=0,79, p<0,01) mais une distribution log-normale (Kolmogorov-Smirnov : D=0,05, p=0,76). Après avoir appliqué une transformation *log* aux valeurs mesurées, et vérifié que les variances étaient homogènes (Levene : F=0,55, df=2, p=0,57), les données remplissaient les prérequis pour des tests paramétriques.

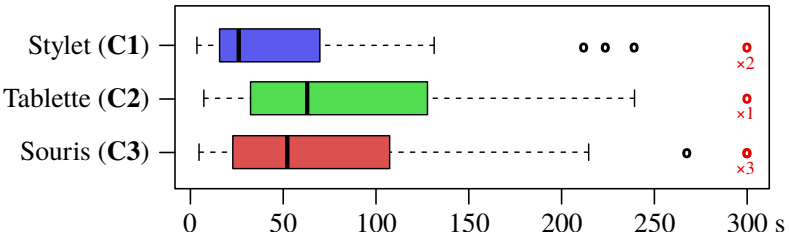


FIGURE 3.14 — Temps d'accomplissement de la tâche (secondes) dans l'expérience de manipulation de plans de coupe. Les barres verticales représentent les médianes.

La figure 3.14 montre les temps d'accomplissement de la tâche pour chaque condition. Le temps médian des essais menés à leur terme est de 24,0 s (σ=54,5 s) avec le stylet, de 62,6 s (σ=63,4 s) avec la tablette et de 50,7 s (σ=58,5 s) avec la souris. Un petit nombre d'essais ont dû être interrompus à cause de la limite de temps : 2 avec le stylet, 1 avec la

tablette et 3 avec la souris. Un test ANOVA révèle une influence statistiquement significative de la condition sur le temps d'accomplissement de la tâche ($F(2,16)=4,52$, $p<0,05$). Des tests *t* post-hoc deux-à-deux, avec une correction de Holm, montrent que le temps moyen pour accomplir la tâche avec le stylet est inférieur à celui de la condition tablette ($p<0,01$) et souris ($p<0,05$). Il semble toutefois ne pas y avoir de différence statistiquement significative entre les conditions tablette et souris ($p>0,05$).

Sur la base de ces résultats, le stylet tangible apparaît être plus performant que la technique actuellement la plus répandue utilisant une souris 2D, pour cette tâche consistant à effectuer des plans de coupe. Cela confirme notre hypothèse de départ qu'une interface basée sur le positionnement direct dans un espace 3D est plus efficace qu'une technique basée sur la souris. De plus, les temps dans la condition stylet présentent une nette asymétrie vers les faibles valeurs ($\gamma_1=2,16$), ce qui signifie que le stylet était bien plus efficace dans certains cas que dans d'autres (tout en restant dans l'ensemble plus performant que la souris). D'après nos observations, cela pourrait être dû au « débrayage » (*clutching*, [Zhai *et al.* 1996]). Lorsque le plan de coupe final pouvait être atteint avec de faibles mouvements des doigts, du poignet ou du bras, la tâche était terminée rapidement. Lorsque le plan de coupe final ne pouvait pas être directement atteint par le stylet tangible à cause de contraintes anatomiques, les participants devaient interrompre la manipulation pour changer de position (débrayage), ce qui rallongeait le temps d'accomplissement de la tâche.

De manière surprenante, la condition tablette n'a pas présenté de meilleures performances que la condition souris dans cette expérience, ce qui semble aller à l'encontre de notre hypothèse. Nos observations durant l'expérience permettent d'avancer une explication à ces performances décevantes. Les participants étaient assis durant l'expérience, et pointaient la tablette en direction de l'objet de référence. Le plan de coupe étant lié à l'orientation de la tablette, l'intervalle des orientations possibles pour le plan de coupe était donc limité. La solution à cette limitation est en principe de manipuler l'objet de référence, afin de faire tourner la donnée par rapport à un plan de coupe fixe. Cependant, les participants tentaient autant que possible d'atteindre l'orientation voulue sans toucher à l'objet de référence, allant parfois jusqu'à des positions extrêmes, mais souvent sans succès. Lorsque les participants se décidaient à manipuler l'objet de référence, ils rencontraient manifestement des difficultés pour interpréter le résultat de leurs actions. Il semblerait que faire tourner le volume de données par rapport à un plan de coupe fixe soit plus complexe à interpréter que de directement faire tourner le plan de coupe.

La technique consistant à utiliser la tablette elle-même comme objet tangible pour contrôler le plan de coupe semble donc poser des problèmes de manipulation. Pourtant, cette technique présentait l'avantage de réduire le nombre d'objets tangibles nécessaires à l'interface. Il serait donc intéressant de rechercher des solutions pour rendre cette approche utilisable.

Résultats subjectifs

La figure 3.15 montre les notes attribuées à chaque condition par les participants dans le questionnaire d'évaluation subjective.

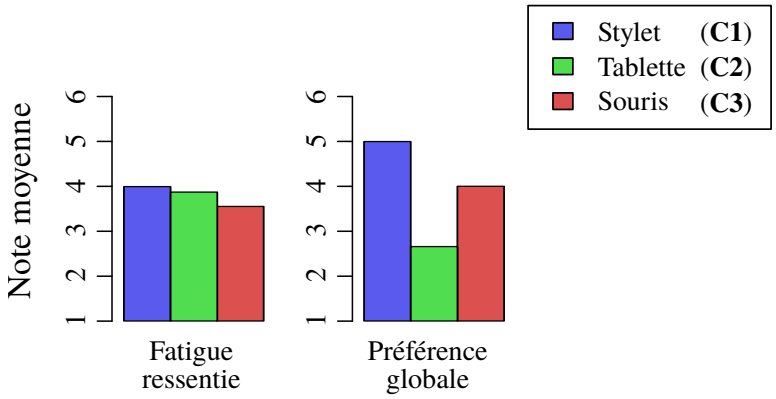


FIGURE 3.15 — Résultats du questionnaire de la deuxième expérience, montrant les notes attribuées par les participants aux trois conditions selon des critères subjectifs.

En terme de fatigue ressentie, les trois conditions ont été perçues comme comparables. La moyenne des scores, sur une échelle de Likert à 6 points (1 : pas fatigant, 6 : fatigant), était de 4,0 pour le stylet (σ=1,2), 3,9 pour la tablette (σ=1,7) et 3,6 pour la souris (σ=1,6). Il semble donc que la technique n'ait pas eu d'effet notable sur la fatigue. De fait, certains participants ont indiqué que tenir la tablette dans la main durant toute l'expérience leur paraissait avoir un effet plus important sur la fatigue que les techniques elles-mêmes.

Les scores de préférence étaient quant à eux plus contrastés. Sur une échelle de Likert à 6 points (1 : négatif, 6 : positif), les notes moyennes étaient de 5,0 pour le stylet (σ=0,7), 2,7 pour la tablette (σ=1,4) et 4,0 pour la souris (σ=1,7). Les faibles notes attribuées à la condition tablette s'expliquent vraisemblablement par les problèmes de manipulation rencontrés par les utilisateurs. La différence entre les scores des conditions stylet et souris est statistiquement faible. Cependant, la condition souris pourrait avoir été favorisée en raison de la familiarité des participants avec ce mode d'interaction, tandis que la préférence pour la condition stylet pourrait avoir été affectée par les erreurs occasionnelles du système de suivi.

En plus des notes subjectives, les 9 participants ont été invités à lister ce qui selon eux représentait les points positifs et négatifs de chaque technique. De manière intéressante, beaucoup ont employé des mots ou expressions similaires. On retrouve également certains commentaires plus spécifiques mais particulièrement éclairants. La condition souris, servant de base de comparaison dans cette étude, a été perçue comme « précise » par trois participants et « stable » par trois autres. Mais sur le plan négatif, cinq participants ont trouvé cette technique « peu intuitive pour contrôler le plan de coupe, en particulier pour les rotations ».

Concernant la condition stylet, cinq participants l'ont trouvée « intuitive » et trois autres l'ont trouvée « pratique pour les rotations du plan de coupe ». Cependant, trois participants ont rencontré des « difficultés pour maintenir le stylet dans le champ de vision » (derrière la tablette) et trois autres se sont plaints « d'interruptions dans le suivi ». Ces interruptions peuvent elles aussi s'expliquer par la sortie, même partielle, du stylet du champ de vision de la caméra. Un des participants a trouvé le stylet « trop sensible », ce qui pourrait être dû en partie à l'imprécision du système de suivi optique, mais peut-être aussi au principe même de l'interaction tangible. Concernant la condition tablette, trois participants ont apprécié le fait que la tablette pouvait ainsi être « tenue à deux mains ». Trois participants ont cependant trouvé cette technique « difficile et/ou imprécise pour positionner le plan de coupe ». Un autre participant s'est plaint du fait qu'il devait maintenir la tablette « trop proche des yeux ». Nous avons en effet remarqué ce problème chez certains participants, qui pourrait s'expliquer par la distance entre le plan de coupe et la tablette que nous avions fixée empiriquement. Comme indiqué plus haut, les participants ne semblent pas avoir tiré parti du fait qu'ils pouvaient déplacer l'objet de référence pour compenser les limitations liées à la tablette. Nous recommandons donc de permettre aux utilisateurs de modifier la distance entre l'outil manipulé et l'appareil mobile lors de l'implémentation d'une technique basée sur ce principe.

3.8 Conclusion

L'approche proposée dans ce chapitre, combinant un appareil mobile classique avec des objets tangibles passifs servant de support de manipulation, constitue une première solution pour répondre simultanément à nos objectifs de portabilité, non-intrusivité et accessibilité. Les études expérimentales menées sur cette interface ont permis de vérifier que celle-ci offrait des performances au moins équivalentes à celles d'un dispositif commercial dédié (une souris 3D) pour des tâches de manipulation, et des performances nettement supérieures à celles des dispositifs actuellement les plus utilisés (souris 2D et écran tactile). Ces résultats confirment donc que notre interface répond également au quatrième objectif d'efficacité.

Cependant, l'étude de cette approche a aussi mis en évidence un certain nombre d'aspects pouvant être améliorés. Il apparaît en effet que déporter la manipulation dans des objets tangibles extérieurs à l'appareil mobile n'est pas sans inconvénients. Sur le plan ergonomique, certains utilisateurs ont rencontré des difficultés pour amener et maintenir des objets tangibles, tels que le stylet, dans l'espace derrière l'appareil mobile. Le suivi optique reposant sur la caméra arrière de l'appareil (la seule solution qui puisse répondre à nos objectifs de portabilité, non-intrusivité et accessibilité), la manipulation est limitée par le champ de vision de cette caméra. Étant donné que l'utilisateur tient les objets tangibles à proximité de l'appareil mobile, l'espace couvert par la caméra est donc particulièrement étroit.

D'autre part, sur le plan pratique, le fait que cette approche nécessite des objets tangibles externes *en plus* d'un appareil mobile classique la rend forcément moins portable qu'une interface utilisant un appareil mobile seul. Il devient en effet nécessaire de conserver et de transporter plusieurs éléments au lieu d'un seul. Bien que les objets tangibles contrebalancent cet inconvénient en permettant une manipulation 3D plus efficace, il n'en reste pas moins une marge d'amélioration concernant la portabilité de l'interface.

Une solution permettant théoriquement de se passer d'objets tangibles distincts a été introduite dans la section 3.5.2 (« coupe contrôlée par la tablette ») : utiliser l'appareil mobile lui-même comme support de manipulation 6 DdL. Mais cette technique s'est avérée très peu performante à cause de difficultés de manipulation. Il semble donc important de chercher à approfondir cette approche, qui pourrait répondre à la fois aux problèmes d'ergonomie et aux problèmes pratiques que posent les objets tangibles séparés.

Une deuxième limitation du type d'approche étudié dans ce chapitre est la difficulté d'intégrer un retour de force dans des objets tangibles passifs. L'utilisation de dispositifs actifs à retour de force est difficilement envisageable dans cette configuration sans aller à l'encontre de nos objectifs. En effet, ces dispositifs sont généralement intrusifs, encombrants et onéreux, tandis que l'intérêt majeur des objets tangibles passifs est d'être non-intrusifs, portables et peu coûteux. L'absence de retour de force constitue un désavantage par rapport aux autres systèmes de réalité virtuelle, qui ne présentent pas cette limitation. Dans le chapitre suivant, nous étudions donc une solution pour retrouver ce retour de force dans une interface portable avec objets tangibles passifs.

Chapitre 4

Perception de forces en l'absence de dispositifs actifs

4.1 Introduction

L'utilisation d'objets tangibles comme support de manipulation, comme cela a été présenté dans le chapitre précédent, permet de faciliter la manipulation 3D en « matérialisant » les objets virtuels associés. Cependant, le nombre d'objets tangibles disponibles dans l'interface est nécessairement fini, tandis que le nombre d'objets virtuels est potentiellement illimité. Ainsi, dans l'exemple de l'exploration de données, l'utilisateur dispose de deux objets tangibles (l'objet de référence et le stylet) mais les objets virtuels pouvant faire l'objet d'une manipulation sont bien plus nombreux (régions d'intérêt dans la donnée, isosurfaces, particules...).

Les objets tangibles sont donc amenés à interagir avec des objets purement virtuels, c'est-à-dire n'étant pas eux-mêmes associés à des objets physiques. Cette situation conduit à un manque de réalisme, car les objets tangibles et les objets purement virtuels peuvent s'interpénétrer.

Un moyen d'améliorer le réalisme de cette interaction est d'implémenter une simulation physique [Buchanan *et al.* 2008; Marchal 2014]. En repoussant les objets virtuels lors d'une collision, ceux-ci apparaissent alors comme « solides » et non plus immatériels. De plus, un comportement physique permet de mieux prévoir et interpréter les réactions des objets manipulés [Fitzmaurice 1996; Teather et Stuerzlinger 2007]. Cependant, la simulation physique à elle seule n'est pas suffisante. En effet, si les objets virtuels sont ainsi affectés par les collisions, les objets tangibles, eux, ne le sont pas. Les objets virtuels n'ayant pas de présence physique, ils ne peuvent pas repousser les objets tangibles. L'interaction est à sens unique. Selon la troisième loi de Newton [1687], un objet entrant en contact avec un autre objet devrait subir une force de réaction équivalente à la force de collision, dans une direction opposée. En l'absence de force de réaction s'exerçant sur les objets tangibles, ces

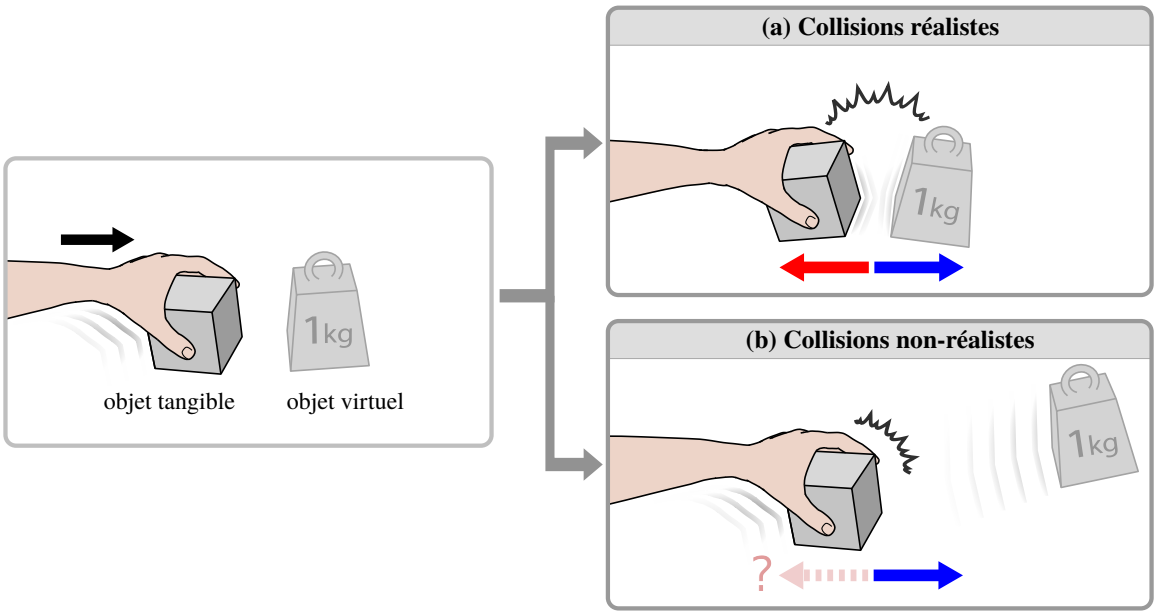


FIGURE 4.1 — Lorsqu'un objet entre en collision avec un autre objet, il subit en principe une force de réaction égale à la force de collision dans la direction opposée (flèche rouge, illustration (a)). Or, un objet tangible passif entrant en collision avec un objet virtuel ne subit aucune force de réaction (illustration (b)). L'interaction entre objets tangibles et virtuels n'est donc pas réaliste.

derniers semblent posséder une inertie infinie tandis que les objets virtuels semblent n'avoir aucune masse. Cette situation souffre donc encore d'un défaut de réalisme (figure 4.1).

Le problème ci-dessus est généralement résolu en intégrant dans l'interface des dispositifs à retour de force, tels que les bras haptiques ou des gants. De fait, de nombreuses installations de réalité virtuelle sont pourvues de ces dispositifs. La question du retour de force dans ces installations est donc facilement réglée.

Cependant, de tels dispositifs présentent d'importantes contraintes qui vont à l'encontre de nos objectifs de portabilité, de non-intrusivité et d'accessibilité. Les bras haptiques et dispositifs similaires requièrent un point d'attache fixe, ce qui les rend fondamentalement non portables. Les gants à retour de force nécessitent d'être portés par l'utilisateur, et les mécanismes qu'ils contiennent les rendent particulièrement intrusifs. Enfin, ces dispositifs à retour de force restent des équipements spécialisés et coûteux, ce qui les rend peu abordables et donc peu accessibles pour les utilisateurs. Afin de répondre à nos objectifs, il faudrait donc pouvoir exercer un retour de force sur les objets tangibles sans recourir à ces dispositifs.

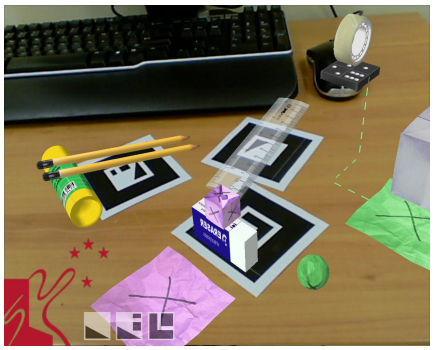
Dans ce chapitre, nous proposons une approche pour recréer une perception de force en l'absence de dispositifs actifs, grâce à un effet purement visuel : le « retour pseudo-haptique ». Nous présentons une nouvelle technique appelée « découplage » visant à reproduire les forces de réaction s'exerçant sur les objets tangibles. Nous évaluons ensuite la capacité de

cette technique à représenter des forces d'intensité différentes, à travers plusieurs études de perception de masses virtuelles.

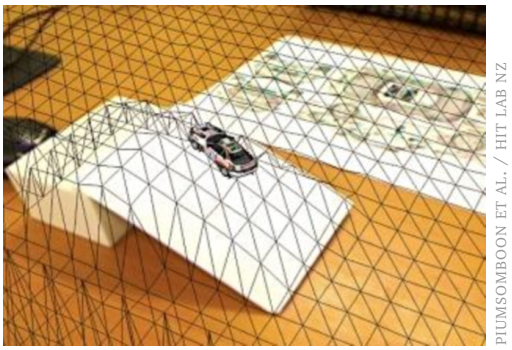
4.2 Travaux existants

4.2.1 Simulation physique

Un nombre croissant d'interfaces font le choix d'employer une simulation physique, pour améliorer l'usabilité et rendre l'interaction avec des objets virtuels plus naturelle. Buchanan *et al.* [2008] décrivent un jeu éducatif en réalité augmentée, dans lequel les objets virtuels sont manipulés par l'intermédiaire de marqueurs et peuvent s'affecter mutuellement d'une manière physiquement réaliste (figure 4.2(a)). Cependant, cette interface ne propose pas d'interaction entre objets physiques et objets virtuels. Wilson [2007] et Piumsomboon *et al.* [2011] utilisent un capteur de profondeur pour analyser l'environnement en temps réel, permettant aux objets virtuels de réagir physiquement à la présence d'objets réels (figure 4.2 (b)). Les objets réels ne sont cependant pas affectés par les objets virtuels. Du point de vue de la simulation physique, ils sont considérés comme inamovibles. Kim *et al.* [2011] mentionnent les problèmes causés par des objets réels inamovibles. Ils montrent que cela conduit à un comportement physique non-réaliste, qu'ils appellent « artéfacts physiques » (*physical artifacts*). En particulier, un de ces artéfacts est le fait que les objets réels ne subissent pas de force de réaction opposée lors d'une collision avec des objets virtuels, et ne sont donc pas repoussés en arrière.



(a) Interaction physique entre objets virtuels [Buchanan *et al.* 2008].

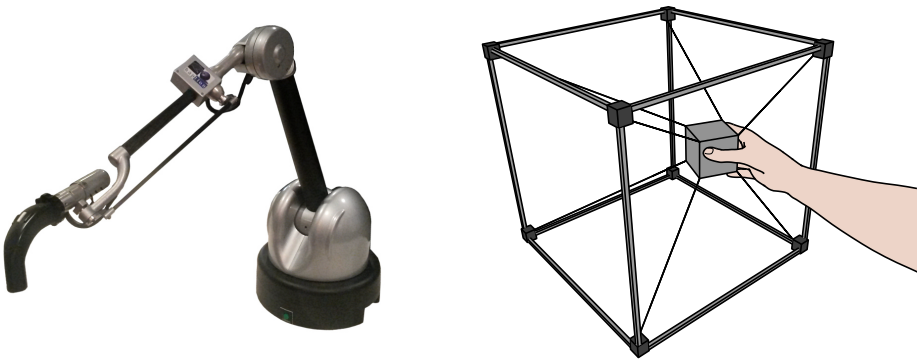


(b) Prise en compte des objets réels dans la simulation [Piumsomboon *et al.* 2011].

FIGURE 4.2 — Deux exemples de simulation physique en réalité augmentée. Le cas (a) ne simule toutefois que les interactions entre objets virtuels, tandis que le cas (b) permet aux objets réels d'affecter les objets virtuels, mais pas dans l'autre sens.

4.2.2 Dispositifs actifs à retour de force

Une solution permettant à des objets réels d'être affectés par des forces virtuelles est de les équiper d'actuateurs. Par exemple, Kang et Woo [2011] proposent « ARMate », un prototype dans lequel un véhicule miniature peut être poussé par un personnage virtuel. Pour cela, le véhicule est équipé de moteurs contrôlés à distance par le système. Bien que cette solution paraisse la plus évidente, équiper tous les objets tangibles de moteurs serait complexe et coûteux en pratique. Par ailleurs, de tels moteurs n'auraient aucun effet lorsqu'un objet tangible est tenu en l'air. Yokokohji *et al.* [1996] et Jeon *et al.* [2011] proposent des solutions basées sur un bras haptique, permettant de moduler les forces s'appliquant sur un objet lors de sa manipulation. Le « SPIDAR » [Kim *et al.* 2000] est un autre type de dispositif pouvant générer un retour de force, qui consiste à fixer l'objet manipulé à un cadre avec plusieurs câbles. L'inconvénient principal des deux approches ci-dessus est qu'elles requièrent des dispositifs non portables, encombrants et onéreux. Par ailleurs, ajouter un tel retour de force à plusieurs objets tangibles nécessiterait un dispositif haptique par objet tangible. Une approche différente est proposée par Minamizawa *et al.* [2007], dans laquelle un gant haptique génère une sensation de force en appliquant une pression et un cisaillement à la peau de la main. Cette approche pourrait effectivement permettre de percevoir, au moins partiellement, une force de réaction à travers n'importe quel objet tangible manipulé. Mais l'utilisation d'un gant que l'utilisateur doit porter rend cette solution intrusive.



(a) Un bras haptique : le VIRTUOSE 6D. (b) Principe du « SPIDAR ».

FIGURE 4.3 — Dispositifs actifs capables de fournir un véritable retour de force. Bien que performants pour remplir cette fonction, ces dispositifs sont particulièrement coûteux, non portables et parfois intrusifs.

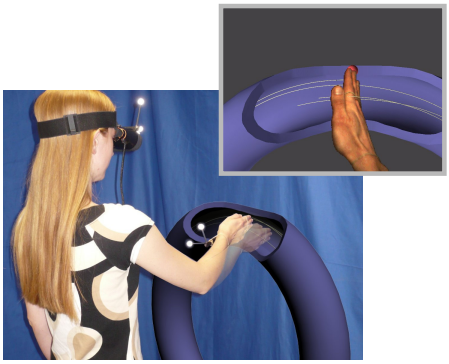
4.2.3 Pseudo-haptique

L'usage de dispositifs actifs à retour de force étant difficile compte tenu des critères de portabilité, de non-intrusivité et d'accessibilité que nous nous sommes fixés, il serait intéressant de

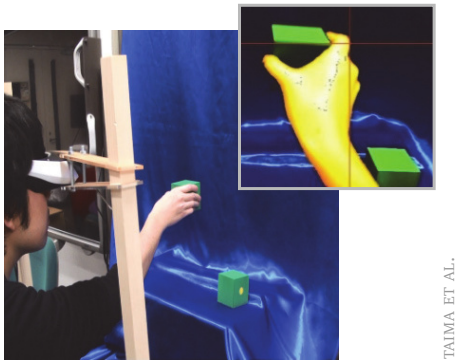
pouvoir simuler des forces par d'autres moyens. Un système basé sur la réalité augmentée, tel que celui employé dans le chapitre précédent, repose sur le principe d'altérer le retour visuel pour afficher des objets virtuels. Or, il est connu que la modalité visuelle domine la modalité haptique dans des tâches d'interaction 3D [Hatwell *et al.* 2003]. En l'absence de « véritables » forces, la modalité visuelle pourrait donc être exploitée pour simuler un retour de force.

Le « retour pseudo-haptique » (*pseudo-haptic feedback*) est une technique qui consiste à modifier le déplacement apparent d'un objet virtuel par rapport au déplacement réel du dispositif d'entrée afin de produire une illusion de force [Lécuyer 2009]. Dans une expérience appelée « marais », Lécuyer *et al.* [2000] montrent que la diminution du gain visuel (rapport entre le mouvement exercé par l'utilisateur et le déplacement apparent de l'objet manipulé sur l'écran) d'un dispositif entièrement passif peut produire l'illusion d'un changement de viscosité. Dans une seconde expérience, ils montrent que cette approche peut également simuler différentes raideurs pour un même ressort. Crison *et al.* [2004] se servent d'un principe similaire dans une simulation de fraisage, en faisant varier visuellement la vitesse de l'outil afin de reproduire la résistance de différents matériaux. Dominjon *et al.* [2005b] rapportent que la modification du gain visuel peut être utilisée pour influencer la perception de masse de deux objets pourtant identiques.

Cependant, les exemples précédents ne sont pas directement transposables à la configuration de réalité augmentée que nous employons ici. En effet, il est bien plus difficile d'altérer le gain visuel d'un objet tangible lorsque celui-ci (ainsi que la main qui le tient) est directement visible à travers une fenêtre de réalité augmentée. Pusch *et al.* [2008] proposent malgré tout une solution pour produire un retour pseudo-haptique en réalité augmentée. Celle-ci



(a) Pusch *et al.* [2008]



(b) Taima *et al.* [2014]

FIGURE 4.4 — Techniques pseudo-haptiques consistant à déplacer visuellement l'image de la main de l'utilisateur pour simuler une force virtuelle. Ces approches ont cependant d'importantes contraintes : l'interaction doit se faire devant un fond uni afin d'isoler l'image de la main, et l'utilisateur doit être équipé d'un visiocasque *video see-through*.

consiste à altérer l'image entière de l'environnement, en extrayant la partie contenant la main de l'utilisateur puis en modifiant sa position apparente, afin de recréer la sensation de pression exercée par un flux virtuel. Taima *et al.* [2014] présentent une technique similaire pour amplifier visuellement le déplacement d'un objet manipulé. Ils montrent qu'il est ainsi possible d'influencer le poids ressenti par l'utilisateur et la fatigue résultante. Ce type d'approche est très puissant car il autorise un contrôle total du mouvement apparent de n'importe quel objet réel. Mais son inconvénient majeur est sa difficulté d'implémentation. Les deux projets ci-dessus ne fonctionnent que lorsque la manipulation se déroule devant un fond bleu, de manière à pouvoir extraire l'image de la main et de l'objet manipulé. Ces approches imposent de plus l'emploi d'un visiocasque de type *video see-through*. Un visiocasque *optical see-through* ou un dispositif non-immersif tel qu'un appareil mobile ne pourraient pas fonctionner, car les utilisateurs resteraient en mesure de voir l'environnement réel.

4.3 Concept

Dans cette section, nous décrivons une technique, appelée « découplage », permettant de reproduire la force de réaction se produisant lorsqu'un objet réel entre en collision avec un objet virtuel. Basée sur le pseudo-haptique, celle-ci permet de se passer de dispositifs à retour de force actifs. Mais à la différence des approches précédentes qui impliquent de modifier de larges portions du retour visuel et requièrent pour cela un visiocasque et un arrière-plan spécifique, notre solution est bien plus accessible et moins intrusive.

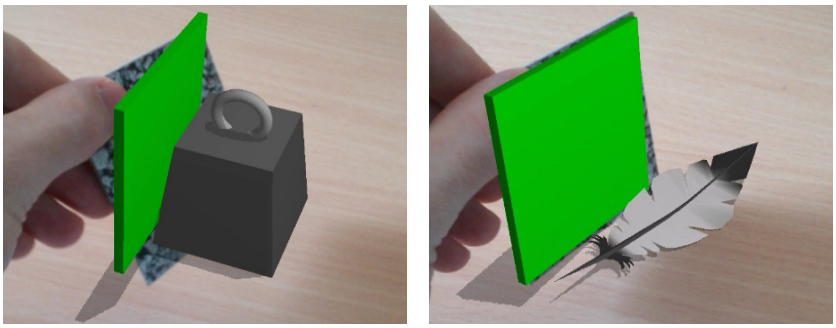


FIGURE 4.5 — Le clone virtuel est visuellement découplé de la position et de l'orientation de l'objet réel qu'il représente. L'utilisation d'une couleur distincte pour le clone permet de renforcer la visibilité du découplage. L'intensité de l'effet dépend de la masse de l'objet virtuel en train d'être poussé.

Notre approche consiste à créer un *clone virtuel* de même forme et de même taille que l'objet tangible (réel), et de l'attacher à cet objet réel par le biais d'un ressort virtuel. Ce clone joue le rôle *d'intermédiaire physique* pour l'objet réel dans l'espace virtuel. Puisque le clone est virtuel, il devient possible de le contrôler entièrement sans avoir besoin de dispositifs additionnels. Il peut donc faire partie intégrante de la simulation physique et interagir

correctement avec les autres objets virtuels. En particulier, il peut être *affecté par les forces de réaction* lors du contact avec un autre objet virtuel, comme le prédit la troisième loi de Newton. L'intensité de cet effet dépend de la masse de l'objet virtuel rencontré. Un objet léger n'affectera que très peu le clone et sera lui-même fortement repoussé, tandis qu'un objet lourd affectera fortement le clone et ne bougera lui-même que très peu (figure 4.5). Ainsi, le déplacement du clone permet de percevoir et de différencier la masse des objets virtuels.

Cependant, le clone virtuel ne doit pas être repoussé trop loin de l'objet virtuel qu'il représente, sans quoi le lien entre les deux objets pourrait ne plus être apparent pour l'utilisateur. Le clone doit rester aussi proche que possible de l'objet réel, tout en restant affecté par les forces virtuelles. Cela est réalisé en reliant les deux objets par une articulation de ressort 6 DdL (*spring joint*). Ce type d'articulation limite le mouvement sur chaque degré de liberté par le biais de ressorts virtuels (figure 4.6). Une telle articulation tend à ramener le clone virtuel vers la position et l'orientation de l'objet réel avec une force qui augmente en fonction de la distance. Le clone est donc bien attaché à l'objet réel, mais reste en mesure de dévier de sa position et orientation lors du contact avec d'autres objets virtuels.

En créant un clone virtuel associé à un objet réel, pouvant être visuellement affecté par les forces de réaction virtuelles, notre technique fournit un *retour pseudo-haptique* des forces virtuelles à travers la *seule modalité visuelle*. Notre approche permet ainsi de se passer de dispositifs spécifiques pour le retour de force, car elle ne nécessite rien d'autre qu'un moyen d'affichage (en principe déjà présent). Ni les objets tangibles, ni l'utilisateur lui-même n'ont besoin d'être instrumentés ou attachés à des dispositifs actifs pour que cette technique fonctionne.

Nous pouvons également remarquer que notre technique ne requiert pas forcément que l'objet réel soit tenu par l'utilisateur, car l'information pseudo-haptique passe par le canal visuel. Il est donc possible de percevoir la masse d'un objet virtuel entrant en collision avec un objet réel quelconque de l'environnement, à partir du moment où ce dernier dispose de son clone virtuel.

Enfin, notre technique ne nécessite pas de déplacer visuellement l'objet réel, ce qui la rend bien plus accessible et moins intrusive que les précédentes approches basées sur des dispositifs immersifs (telles que [Pusch *et al.* 2008; Taima *et al.* 2014]). Puisque la vue réelle n'est pas altérée, il peut arriver que deux versions du même objet soient visibles simultanément : l'objet réel et son clone virtuel, ce qui ne se produisait pas dans les approches citées. Mais l'articulation de ressort permet de s'assurer que les deux restent toujours suffisamment proches pour être considérés comme un même objet. Notre technique relâche le couplage entre la représentation physique de l'objet réel dans la simulation, matérialisée par son clone virtuel, et l'objet réel lui-même. Ils sont liés, sans être strictement couplés. Pour cette raison, nous appelons cette technique le « découplage ».

4.4 Implémentation

Un des principaux avantages de notre technique de découplage est qu'elle repose entièrement sur le rendu visuel, et ne nécessite donc pas d'équipements supplémentaires. L'aspect logiciel présente également peu de difficulté, car il consiste simplement à simuler une articulation de ressort 6 DdL. La plupart des moteurs physiques, en plus de gérer les collisions entre objets virtuels, offrent les fonctionnalités appropriées pour définir ce type d'articulation inter-objets.

4.4.1 Modèle d'articulation de ressort

Une articulation de ressort 6 DdL peut être modélisée comme un système masse-ressort sur chacun des 6 degrés de liberté (figure 4.6), dans lequel le clone virtuel (la « masse ») serait relié à l'objet réel par le biais de ressorts. Ces derniers peuvent être vus comme des ressorts de traction sur les axes de translation, ou comme des ressorts de torsion sur les axes de rotation. Chaque ressort possède un paramètre de *raideur*. Comme nous allons le voir par la suite, cette raideur a une influence cruciale sur l'efficacité de la technique de découplage pour la perception de masse.

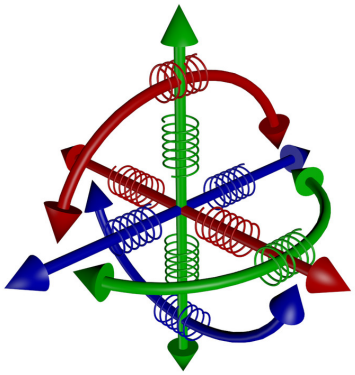


FIGURE 4.6 — Représentation schématique d'une articulation de ressort 6 DdL.

En tant que tels, de simples ressorts présenteraient un risque d'instabilité. Lorsqu'une force virtuelle cesse de s'appliquer sur le clone virtuel, celui-ci pourrait se mettre à osciller autour de la position et de l'orientation de l'objet réel, ce qui serait perturbant pour l'utilisateur. Pour éviter cela, il est nécessaire de rajouter un terme *d'amortissement* qui ralentit le mouvement du clone virtuel durant le retour à sa configuration de base. Ce coefficient d'amortissement doit être bas pour que le clone retourne aussi rapidement que possible vers l'objet réel, mais doit également être suffisamment grand pour empêcher toute oscillation. La valeur précise qui satisfait au mieux ces deux critères est appelée le coefficient d'amortissement *critique*.

L'articulation de ressort est réalisée en appliquant continuellement une force et un couple au clone virtuel de manière à le ramener constamment vers la position et l'orientation de l'objet réel. Selon la loi de Hooke, la force exercée par un ressort est égale à $-k\vec{d}$ où k est la raideur du ressort et $\vec{d}$ représente l'allongement (distance entre le clone virtuel et l'objet réel). La force d'amortissement est égale à $-cv$, où c est le coefficient d'amortissement et $\vec{v}$ est la vélocité du clone virtuel par rapport à l'objet réel. Le coefficient d'amortissement critique c_c est égal à $2\sqrt{mk}$ dans le cas des ressorts de traction, m représentant la masse du clone virtuel. La force $\vec{F}$ à appliquer sur le centre de masse du clone pour simuler la partie traction est donc obtenue par :

$$c_c = 2\sqrt{mk}$$
$$\vec{F} = -k\vec{d} \ + \ -c_c\vec{v}$$

- $\vec{d}$ distance entre le clone virtuel et l'objet réel
- $\vec{v}$ vélocité du clone virtuel par rapport à l'objet réel
- k raideur des ressorts de traction
- m masse du clone virtuel
- c_c coefficient d'amortissement critique pour m et k

Le couple $\vec{\tau}$ à appliquer pour simuler la partie torsion est obtenu d'une manière similaire, mais la masse du clone virtuel est remplacée par son *moment d'inertie* (ou « tenseur d'inertie »). Cette grandeur est en principe calculée par le moteur physique.

$$C_c = 2\sqrt{I\kappa}$$
$$\vec{\tau} = -\kappa\vec{\theta} \ + \ -C_c\vec{\omega}$$

- $\vec{\theta}$ angles (sur chaque axe) entre le clone et l'objet réel
- $\vec{\omega}$ vélocité angulaire du clone virtuel par rapport à l'objet réel
- κ raideur des ressorts de torsion
- I moment d'inertie du clone virtuel
- C_c coefficient d'amortissement critique pour I et κ

4.4.2 Paramétrage

D'après le modèle ci-dessus, nous pouvons constater que la technique de découplage est contrôlée par quatre paramètres : la raideur en traction k, la raideur en torsion κ, la masse m du clone virtuel, et son moment d'inertie I. Le moment d'inertie peut être calculé à partir de la masse et de la forme de l'objet. Ce calcul est généralement effectué par le moteur physique.

Les paramètres restants doivent être fixés en fonction des masses que l'on *prévoit de rencontrer* dans l'environnement virtuel. La masse m doit tout particulièrement être comparable aux autres masses virtuelles. Si m est bien plus grande que la masse des autres objets virtuels, le clone aura tant d'inertie qu'il sera à peine affecté par les forces de réaction et aucun découplage visible ne se produira. À l'inverse, si m est bien plus faible, le clone n'aura pas assez d'inertie pour déplacer les autres objets. Si les objets virtuels ont des masses en adéquation avec leur taille apparente en réalité augmentée, un bon point de départ pour choisir la masse m est d'utiliser celle de l'objet réel correspondant.

Les deux paramètres de raideur k et κ doivent être fixés de manière à maximiser l'amplitude visuelle du découplage entre un objet virtuel léger et un objet lourd. Idéalement, un découplage quasi-imperceptible devrait se produire lors du contact avec l'objet virtuel le plus léger, et un découplage considérable devrait se produire lors du contact avec l'objet virtuel le plus lourd. Étant donné que la technique de découplage repose exclusivement sur la modalité visuelle, le choix des paramètres de raideur est déterminant pour l'efficacité de la perception de masse.

Un dernier paramètre implicite se doit d'être mentionné : la friction. Un certain degré de friction est indispensable dans toute simulation physique pour éviter qu'un objet virtuel ne glisse indéfiniment après une collision avec un autre objet. Mais la friction a également un effet sur le découplage. En tant que force, elle modifie l'inertie d'un objet virtuel posé au sol. Selon la troisième loi de Newton, la force de réaction produite en rentrant en contact avec cet objet se trouve donc aussi modifiée. Il faut donc s'assurer que cela ne fausse pas la perception de masse. Dans le modèle de Coulomb (implémenté par les moteurs physiques), la friction dépend de la masse des objets et d'un facteur de multiplication constant. L'accroissement de la force de réaction due à la friction est donc proportionnel à la masse des objets. Par conséquent, la friction ne change pas le fait qu'un objet virtuel sera perçu comme plus léger ou plus lourd qu'un autre objet virtuel.

4.5 Première étude : tri d'objets par masse

L'objectif de cette première étude est de vérifier notre hypothèse de base, selon laquelle interagir avec des objets virtuels par le biais d'un objet tangible passif (sans retour de force actif) ne permet pas de percevoir (et de différencier) les masses virtuelles, et que notre technique de découplage rend cette perception possible.

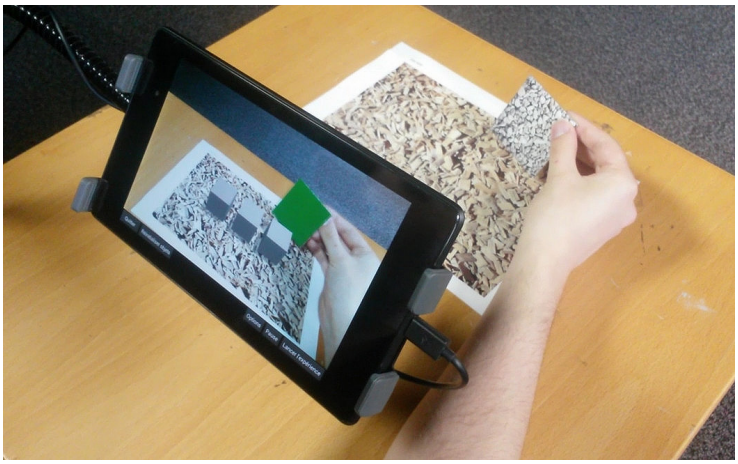


FIGURE 4.7 — Plateforme expérimentale pour l'étude de la technique de découplage. La tâche représentée ici est celle de la première expérience (tri d'objets).

4.5.1 Conditions et hypothèses

Dans cette expérience, un objet tangible appelé « effecteur » (visible dans les figures 4.5 et 4.7) est utilisé pour déplacer des objets virtuels. L'objectif est de comparer deux conditions : l'une où l'effecteur repousse les objets virtuels sans aucun découplage, l'autre où un découplage est visible. Pour éviter tout biais dû à d'éventuelles différences visuelles entre les deux conditions, tel que des différences d'ombrage ou d'occultation, nous avons fait le choix de toujours afficher le clone virtuel de l'effecteur. Dans la première condition, le clone n'est pas affecté par les forces virtuelles et reste donc strictement couplé à l'effecteur réel. Dans l'autre, il se découple comme expliqué précédemment. Les deux conditions de cette expérience sont donc :

— **C1** : l'effecteur virtuel reste strictement couplé à l'effecteur réel ;
— **C2** : l'effecteur virtuel peut se découpler de l'effecteur réel.

Notre hypothèse (**H1**) est que la condition **C1** ne permet pas aux utilisateurs de différencier des masses virtuelles. Nous supposons également (**H2**) que la condition **C2** permet de percevoir cette différence, et plus spécifiquement de reconnaître si un objet virtuel est plus léger ou plus lourd qu'un autre. Enfin, nous faisons l'hypothèse (**H3**) que le découplage ne nécessite pas d'entraînement préalable pour arriver à cette perception. Nos hypothèses pour cette expérience sont donc :

— **H1** : la condition **C1** ne permet pas de percevoir les différences de masse entre objets virtuels ;
— **H2** : la condition **C2** permet de reconnaître qu'un objet est plus léger ou plus lourd qu'un autre ;
— **H3** : la condition **C2** est efficace sans entraînement préalable.

4.5.2 Participants

Un total de 13 participants (3 femmes, 10 hommes), âgés de 22 à 37 ans (moyenne=27,0, σ=4,9) ont participé à cette expérience. Un des participants était gaucher. Aucun d'entre eux n'avait de connaissance préalable de la technique de découplage.

4.5.3 Plateforme expérimentale

Les participants sont assis devant une table, sur laquelle se trouvent l'objet servant d'effecteur, un plateau texturé pour la réalité augmentée, et un support flexible. Une tablette 7 pouces est fixée sur le support, de manière à se trouver entre les yeux de l'utilisateur et le plateau texturé. Les objets virtuels apparaissent en réalité augmentée (à une cadence de 15 images par seconde) à travers la tablette, sur le plateau qui se situe derrière. Le plateau et l'effecteur sont suivis dans l'espace 3D en utilisant le système Vuforia [1]. Les participants sont invités à tenir l'effecteur derrière la tablette et à s'en servir pour interagir physiquement avec les objets virtuels (figure 4.7).

Nous utilisons le moteur physique Bullet [2] pour gérer les collisions entre objets et pour implémenter notre articulation de ressort comme cela est décrit plus haut. Le paramètre m définissant la masse de l'effecteur est fixé à 10 g, et la simulation physique est conçue [3] pour des masses virtuelles allant d'environ 10 g à 1000 g. Sur la base de cette masse m et de celles des objets virtuels que les participants seront amenés à rencontrer, nous avons fixé les autres paramètres de façon à maximiser l'amplitude de découplage. La raideur en traction k est fixée à 50 N·m^{-1} et la raideur en torsion κ est fixée à 150 N·m·rad^{-1}. Le coefficient de friction est de 0,8 entre chaque objet, dont le sol virtuel.

4.5.4 Protocole

Préalablement à l'expérience, nous montrons aux participants une courte vidéo de démonstration leur indiquant comment tenir l'effecteur et comment l'utiliser pour interagir avec les objets virtuels. Cette démonstration se justifie par la liberté d'interaction qu'autorise une telle interface tangible. Il est possible de tenir et d'utiliser l'effecteur de multiples manières, mais certaines configurations aberrantes risqueraient de rendre le découplage difficile à percevoir (par exemple lorsque l'effecteur est vu sur la tranche). La vidéo de démonstration aide

1. `http://www.vuforia.com/`
2. `http://www.bulletphysics.org/`
3. Il reste possible d'interagir avec des masses hors de cet intervalle, mais la simulation est alors moins stable.

donc les participants à adopter un comportement cohérent pendant l'expérience. Dans cette vidéo, les objets virtuels ont tous la même masse et *aucun découplage* n'est visible. Ainsi, les participants restent dans l'ignorance de la technique de découplage jusqu'au commencement véritable de l'expérience.

L'expérience elle-même débute par une tâche d'introduction dans la configuration de découplage (**C2**). Le but de cette première tâche est d'évaluer l'efficacité de la technique de découplage sur des participants entièrement novices. Deux cubes virtuels, de 3,5 cm d'arête, sont affichés côte à côte en réalité augmentée. Ils sont visuellement indifférenciables mais ont des masses virtuelles différentes : 15 g et 800 g. Les participants doivent identifier le cube le plus lourd. Pour cela, ils sont invités à pousser chaque objet avec l'effecteur, avant d'indiquer lequel est le plus lourd en le sélectionnant sur l'écran de la tablette. Les deux cubes sont visibles en même temps et les participants peuvent donc les manipuler librement dans l'ordre qu'ils souhaitent, l'objectif étant effectivement de les comparer. Il n'y a pas de limite de temps. La tâche se termine dès qu'une réponse correcte est donnée, mais les réponses incorrectes sont aussi enregistrées.

Après cette tâche d'introduction, une série d'essais est lancée. Celle-ci consiste en 10 essais, répartis de manière égale entre les conditions **C1** et **C2**. Il y a maintenant *trois* cubes virtuels (figure 4.7), de 3,5 cm d'arête, toujours visuellement identiques mais avec des masses virtuelles différentes. Les masses virtuelles sont de 15 g, 200 g et 800 g, représentant des objets légers, moyens et lourds dans l'intervalle pour lequel est optimisée la simulation (section 4.5.3). À chaque essai, ces masses restent les mêmes mais les cubes correspondants sont disposés côte à côte dans un ordre aléatoire. La tâche est d'indiquer quel cube est le plus léger, le plus lourd, et de masse intermédiaire. Avec 3 objets à trier, il n'y a qu'une probabilité de 1/6 (≈17 %) de donner une bonne réponse par pure chance. Les participants sont invités à manipuler les cubes avec l'effecteur, dans n'importe quel ordre, aussi longtemps que nécessaire pour répondre à la question posée. À tout moment, les participants peuvent marquer un cube comme étant le plus léger, le plus lourd ou de masse intermédiaire (ou encore modifier une marque déjà placée) en le sélectionnant sur l'écran de la tablette. Ces marques sont mutuellement exclusives. Lorsque les trois cubes sont marqués, les participants pressent un bouton sur l'écran tactile pour valider leur choix et terminer la tâche.

Un bouton « je ne sais pas » est également disponible pour passer directement à l'essai suivant. Les participants sont informés qu'ils peuvent utiliser ce bouton dans l'éventualité où ils ne se sentent pas capables de répondre, c'est-à-dire lorsqu'ils ne perçoivent pas suffisamment les masses virtuelles pour les différencier. Plutôt que d'obliger les participants à répondre au hasard, cette option permet d'avoir une indication claire que le participant lui-même n'était pas en mesure de percevoir les masses virtuelles.

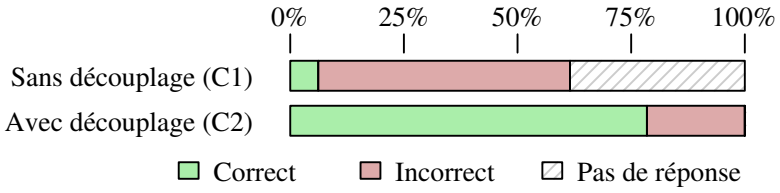


FIGURE 4.8 — Pourcentages moyens de réponses correctes, incorrectes et de non-réponses dans la tâche de tri d'objets.

4.5.5 Résultats et discussion

La figure 4.8 montre les résultats obtenus lors de cette tâche de tri d'objets. Dans la condition **C1** (sans découplage), les participants ont pu trier correctement les trois objets dans **6,2 %** des cas en moyenne (σ=9,6 pp entre participants). Le pourcentage moyen de réponses incorrectes est de 55,4 % et le pourcentage moyen de non-réponses est de 38,5 %. Le taux de succès s'avère donc être très bas dans cette condition. Les participants ont souvent choisi de ne pas répondre, ce qui montre qu'ils avaient conscience de ne pas percevoir de différence de masse dans de nombreux cas. Même en ne considérant que les réponses exprimées, le taux de succès résultant (10,1 %) reste en deçà de la probabilité de 1/6 (17 %) de donner la bonne réponse par pure chance. Par conséquent, ces résultats confirment notre hypothèse **H1** que la condition **C1** ne permet pas de percevoir de différence de masse entre les objets virtuels.

À l'inverse, dans la condition **C2** (découplage), les participants ont pu trier correctement les trois objets dans **78,5 %** des cas en moyenne (σ=20,8 pp entre participants). Le pourcentage moyen de réponses incorrectes est de 21,5 % et le pourcentage de non-réponses est de 0 %. Ce taux de succès est nettement au-dessus de la probabilité de 17 % de donner une bonne réponse par pure chance. Il apparaît donc que le découplage permet effectivement de percevoir la différence entre un objet virtuel léger et un objet lourd, ce qui confirme l'hypothèse **C2**. Par ailleurs, un taux de succès aussi important montre que les participants ont pu distinguer clairement et sans ambiguïté les différences entre les objets, du moins dans les conditions (masses virtuelles et paramètres de découplage) de cette expérience. Le fait que les participants ne se soient jamais servi du bouton « je ne sais pas » montre bien qu'ils étaient confiants dans leur perception.

En ce qui concerne l'hypothèse **H3** que le découplage est efficace sans le moindre entraînement, nous pouvons examiner les résultats de la première tâche. Comme expliqué dans le protocole, les tentatives de réponses correctes comme incorrectes étaient toutes deux enregistrées. Pourtant, nous avons observé que la totalité des 13 participants ont répondu correctement du premier coup. Étant donné qu'aucun d'entre eux n'avait été exposé à la technique de découplage avant le début de l'expérience, nous pouvons en conclure que cette

technique est efficace sur des sujets novices. La métaphore de découplage est immédiatement comprise sans entraînement préalable, ce qui confirme notre hypothèse **H3**.

4.6 Deuxième étude : seuil différentiel

La première étude a montré que le découplage est efficace pour percevoir la différence entre trois masses virtuelles à travers les forces de réaction générées, et cela sans dispositif actif de retour de force. Cependant, ces résultats restent spécifiques aux trois masses testées dans cette étude (et aux paramètres de découplage choisis). Bien qu'il soit maintenant acquis que la technique de découplage permet effectivement de distinguer certaines masses virtuelles, deux masses relativement proches pourraient être plus difficiles à différencier.

Cette deuxième étude a donc pour objectif de mesurer la plus petite différence de masse pouvant être perçue grâce au découplage. Cette mesure se nomme le « seuil différentiel » (*just noticeable difference, JND*). En tant que différence, ce seuil est exprimé par rapport à une valeur de référence, dans notre cas une *masse de référence*. D'après la loi de Weber [Ross et Murray 1996], le seuil différentiel entre deux poids réels est proportionnel à la valeur de référence. Pour cette raison, il serait utile de déterminer comment le seuil différentiel du découplage varie en fonction de la masse de référence. Dans cette expérience, nous avons donc choisi de mesurer ce seuil pour trois masses de référence représentatives couvrant l'intervalle optimisé pour la simulation : 15 g, 200 g et 800 g. La loi de Weber ayant été vérifiée dans une précédente configuration pseudo-haptique [Dominjon *et al.* 2005a], nous pouvons faire l'hypothèse que le seuil différentiel augmente avec la masse de référence (**H1**) et que cette augmentation est proportionnelle (**H2**) :

— **H1** : le seuil différentiel augmente avec la masse de référence ;
— **H2** : le seuil différentiel est proportionnel à la masse de référence.

4.6.1 Participants et plateforme expérimentale

Les mêmes 13 participants que dans la première étude ont pris part à cette expérience, ce qui nous a permis de nous assurer qu'ils étaient déjà familiarisés avec la technique de découplage. La plateforme expérimentale est elle aussi inchangée (figure 4.7).

4.6.2 Protocole

La tâche expérimentale consiste à comparer la masse de deux objets virtuels, et à indiquer s'ils paraissent différents ou non. Dans chaque essai, deux cubes virtuels visuellement identiques, de 3,5 cm de côté, sont affichés côte à côte en réalité augmentée. Un des cubes sert de masse de référence tandis que l'autre a une masse de comparaison plus importante. Les deux cubes sont disposés dans un ordre aléatoire. Les participants manipulent les cubes avec l'effecteur de manière à comparer leurs masses. Ils doivent ensuite sélectionner le cube le plus lourd sur l'écran s'ils perçoivent une différence, ou appuyer sur un bouton « pas de différence » dans le cas contraire. Il n'y a pas de limite de temps pour chaque essai, mais les participants sont invités à se fier à leur première impression afin de ne pas passer trop de temps sur chaque essai.

Nous employons une procédure dite « en escalier » (*staircase procedure*, [Leek 2001]) pour estimer le seuil différentiel. Le premier essai de « l'escalier » débute avec une différence initiale entre les masses de référence et de comparaison, et un pas initial. Ces valeurs doivent être bien adaptées pour que le premier essai soit sans ambiguïté, sans pour autant allonger inutilement l'expérience. Nous utilisons donc des valeurs initiales différentes pour chaque masse de référence (tableau 4.1).

Tableau 4.1 — Paramètres initiaux de la procédure en escalier.

Masse de référence	Différence initiale	Pas initial
15 g	175 g	50 g
200 g	600 g	150 g
800 g	700 g	200 g

Lorsque le participant indique avoir perçu une différence (en sélectionnant l'un des cubes), la masse de comparaison est diminuée du pas actuel. À l'inverse, lorsque le participant indique ne pas avoir perçu de différence (en appuyant sur le bouton), la masse de comparaison est augmentée du pas actuel. Si la réponse est différente de celle de l'essai précédent, il s'agit d'un *renversement*. Lorsque deux renversements se produisent successivement, le pas est divisé par deux. Après chaque réponse de l'utilisateur, un nouvel essai est lancé avec les nouveaux paramètres. L'expérience se termine après 10 renversements. À la fin de la procédure, le seuil différentiel est calculé comme la moyenne des différences de masse dans tous les essais où s'est produit un renversement, à l'exception des deux premiers (considérés comme antérieurs à la phase de convergence).

Puisqu'il y avait trois masses de référence, nous avons mené simultanément trois procédures en escalier en alternant de manière aléatoire entre les essais de chaque procédure. Cela présente l'avantage supplémentaire de rendre moins prévisibles les masses virtuelles rencontrées au cours de l'expérience.

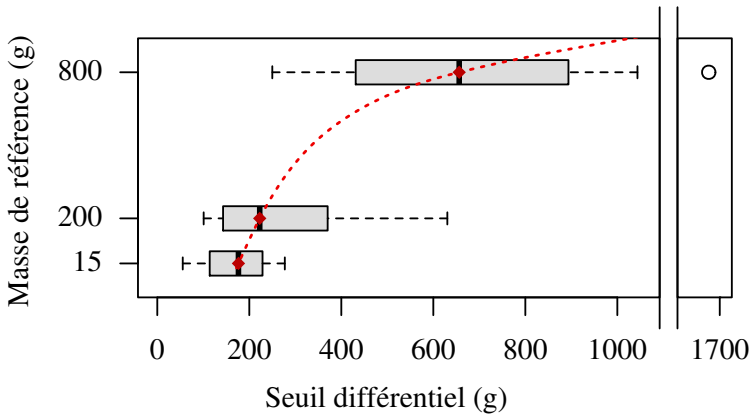


FIGURE 4.9 — Seuils différentiels mesurés par rapport aux trois masses de référence. Un possible modèle non-linéaire est ajusté aux trois points médians.

4.6.3 Résultats et discussion

La figure 4.9 montre les seuils différentiels obtenus. Les valeurs moyennes sont :

— pour 15 g : **172,3 g** (σ=72,4 g)
— pour 200 g : **271,5 g** (σ=155,3 g)
— pour 800 g : **696,2 g** (σ=394,7 g)

Nous pouvons déjà constater que les résultats présentent une certaine dispersion. Cela peut s'expliquer en partie par l'incertitude inhérente à ce type de mesure : les participants n'ont simplement pas tous les mêmes seuils de perception. Un autre facteur pourrait être l'aspect non-contraint de l'environnement expérimental. Par exemple, les participants étaient libres de manipuler les cubes dans n'importe quel ordre. De plus, bien que nous leur ayons montré comment tenir l'effecteur et comment s'en servir pour manipuler les objets (section 4.5.4), les participants restaient en mesure d'employer différentes stratégies pour comparer les masses virtuelles. Enfin, certains passaient plus de temps que d'autres pour se faire une « première impression ». Cependant, l'un des principaux avantages de l'interaction physique par le biais d'objets tangibles est de rendre la manipulation plus naturelle et spontanée. Un environnement expérimental très encadré n'aurait donc pas été représentatif de l'utilisation réelle. En accordant délibérément une certaine liberté aux participants, nous favorisons donc la validité externe, avec comme contrepartie une plus grande dispersion des résultats.

Malgré cela, il apparaît clairement que le seuil différentiel augmente effectivement avec la masse de référence. Cela est en accord avec les précédentes études sur la perception de masse, et confirme notre hypothèse **H1**. Pourtant, il semble aussi que la relation entre le seuil différentiel et la masse de référence ne soit pas proportionnelle, comme le suggère la courbe tracée dans la figure 4.9. Cela contredirait donc l'hypothèse **H2**. Bien entendu, trois

points sont loin d'être suffisants pour établir un modèle fiable. Mais cette non-linéarité, si elle existe réellement, pourrait s'expliquer par la manière dont fonctionne la technique de découplage.

Lorsque l'effecteur entre en contact avec un objet léger, tel qu'une masse autour de 15 g, la force nécessaire pour le déplacer est assez faible. La force de réaction exercée sur le clone virtuel de l'effecteur est donc également faible. Celui-ci étant très peu affecté, le découplage est presque invisible sur l'écran. Comme nous l'avons montré dans la première étude, l'absence de découplage rend alors imperceptibles les différences entre les deux masses. Cette différence ne devient perceptible que lorsque la force de réaction est suffisamment grande pour provoquer un découplage plus net, ce qui semble se produire autour de 190 g d'après la figure 4.9. Cela pourrait expliquer pourquoi le seuil différentiel semble avoir une valeur plancher.

Avec des masses virtuelles plus importantes, telles que 800 g et plus, le phénomène opposé se produit. La force nécessaire pour déplacer l'objet est bien plus grande, par rapport à la force générée par l'articulation de ressort. Au-delà d'un certain point (rarement atteint pendant l'expérience mais qui existe malgré tout), la force requise est si élevée que l'objet virtuel semble devenir immuable : le clone de l'effecteur appuie sur l'objet sans effet apparent. Cela pourrait expliquer pourquoi le seuil différentiel semble se diriger vers un point de saturation lorsque les masses virtuelles deviennent importantes.

La combinaison de l'effet plancher pour les faibles masses et du point de saturation pour les masses importantes constitue donc une explication possible à la non-proportionnalité observée dans nos résultats.

Il est important de garder à l'esprit que les valeurs précises des seuils obtenus dans cette expérience restent spécifiques aux paramètres de découplage (raideur et friction) de notre implémentation. Comme expliqué précédemment, ces paramètres doivent être ajustés de manière à maximiser les différences de découplage entre les objets virtuels les plus légers et les objets les plus lourds. Mais la vision humaine, le dispositif d'affichage et le moteur physique ont tous une précision finie. Par conséquent, ajuster ces paramètres pour un intervalle de masse plus large mène nécessairement à un seuil différentiel moins bon (plus haut). Inversement, un meilleur seuil différentiel (plus bas) peut toujours être obtenu en ajustant les paramètres pour un intervalle de masse plus restreint, produisant ainsi un découplage plus facilement visible entre deux masses proches dans cet intervalle.

4.7 Conclusion

Nous avons présenté dans ce chapitre une nouvelle technique dite de « découplage », basée sur le principe du pseudo-haptique, permettant de percevoir des forces virtuelles à travers la seule modalité visuelle. Cette technique permet ainsi d'améliorer la richesse du retour d'information dans une interface de manipulation 3D tout en se passant des contraintes de non-portabilité, intrusivité et coût que pose l'utilisation de dispositifs actifs à retour de force.

Une première expérience a montré l'efficacité de cette technique pour différencier des forces de réaction traduisant des masses virtuelles différentes, avec un taux de succès très nettement supérieur à celui obtenu sans retour pseudo-haptique. Dans une deuxième expérience, nous avons mesuré le seuil différentiel (JND) que permet de percevoir notre technique. Nous avons pu observer que ce seuil augmente bien avec la valeur de référence, mais de manière non proportionnelle contrairement aux prédictions de la loi de Weber. En particulier, le JND présente une valeur plancher pour des masses (et donc forces) faibles, et une saturation pour des forces importantes.

Il serait donc intéressant de rechercher d'autres manières de représenter les forces virtuelles exercées sur l'objet effecteur, par exemple en simulant une déformation non rigide, en provoquant son changement de couleur, ou encore en émettant du son. Cela pourrait ainsi apporter un complément au découplage, afin d'aider l'utilisateur à mieux percevoir et différencier des forces virtuelles très faibles ou très importantes.

Dans le prochain chapitre, nous nous intéressons à la deuxième question mise en évidence dans le chapitre 3 : comment remédier aux inconvénients, en terme de portabilité, que pose l'emploi d'objets tangibles externes et séparés de l'appareil mobile.

Chapitre 5

Appareil mobile comme support de manipulation

5.1 Introduction

Les approches décrites jusqu'à présent, bien que répondant effectivement aux critères de portabilité, de non-intrusivité, d'accessibilité et d'efficacité de manipulation, présentent une caractéristique notable. Elles sont constituées de plusieurs éléments distincts : des objets tangibles pour la manipulation, et un appareil mobile pour l'affichage et le traitement.

Or, cette séparation en plusieurs éléments peut s'avérer être un inconvénient. On a ainsi pu observer dans le chapitre 3 certains problèmes ergonomiques : les utilisateurs avaient parfois des difficultés à manipuler simultanément plusieurs objets, et des difficultés pour aligner correctement la « fenêtre » de l'appareil mobile avec les objets tangibles se trouvant derrière. Mais cette séparation peut également poser des problèmes pratiques. Bien que l'interface en elle-même soit portable, il demeure indispensable de toujours *transporter ensemble* les éléments qui la constituent (appareil mobile et objets tangibles). L'oubli ou la perte de l'un d'entre eux peut perturber, voire empêcher totalement le fonctionnement de l'interface. Le fait d'être composé de plusieurs éléments présente donc un frein à la portabilité.

Pour surmonter ces inconvénients, une solution serait de faire en sorte que l'interface soit constituée d'un seul élément intégré. Dans ce chapitre, nous étudions donc une nouvelle approche visant à augmenter l'intégration et donc la portabilité de l'interface, tout en conservant les avantages de la manipulation tangible sur appareil mobile.

5.2 Concept

Les interfaces étudiées dans les chapitres 3 et 4 consistaient à combiner un appareil mobile avec des objets tangibles. L'appareil mobile fournissait l'affichage et les capacités de traitement, tandis que les objets tangibles servaient de support pour la manipulation.

Cependant, l'appareil mobile peut lui aussi être vu comme un objet tangible. Il est tenu dans la main et peut être déplacé et orienté dans l'environnement sur 6 degrés de liberté. Une solution pour rendre l'interface plus intégrée est donc de se servir de l'appareil mobile *lui-même* comme support de manipulation. Cette approche consiste à utiliser les mouvements mêmes de l'appareil pour contrôler des objets 3D affichés sur son propre écran (figure 5.1). L'interface tout entière n'est ainsi plus constituée que d'un seul élément : l'appareil mobile. Cette solution élimine la nécessité d'avoir des objets distincts, tout en conservant l'aspect direct de la manipulation tangible.

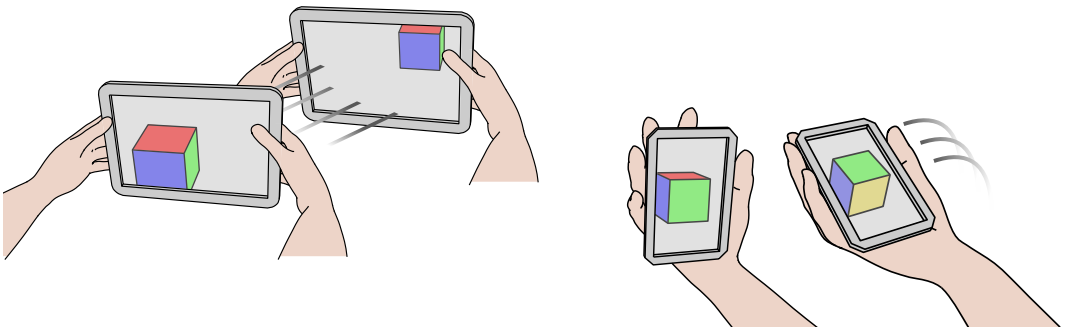


FIGURE 5.1 — Utilisation des mouvements (translation et rotation) d'un appareil mobile pour manipuler des objets 3D affichés sur son propre écran.

Cette configuration désormais entièrement intégrée présente toutefois une particularité importante. L'écran, sur lequel sont affichés les objets manipulés, est couplé avec le dispositif d'entrée. Il s'agit ainsi d'une configuration « localement couplée » (*locally coupled configuration*, [Rahman *et al.* 2009]). Par conséquent, l'écran sur lequel sont affichés les objets 3D *se déplace et tourne en même temps que le support de manipulation.*

Cette particularité soulève des questions spécifiques, qui ne se posaient pas dans le cas où les objets tangibles étaient séparés de l'écran. La première question est de savoir comment faire correspondre au mieux les mouvements sur l'écran avec les mouvements de l'appareil mobile. En effet, si les objets 3D manipulés se déplacent sur l'écran, l'écran lui-même se déplace aussi durant l'opération. Du point de vue de l'utilisateur, il y a donc une *composition* de déplacements. Les techniques d'interaction classiques supposent l'existence d'une surface d'affichage séparée, qui ne produit pas un tel effet. Il est donc indispensable d'adapter les fonctions de transfert (*mappings*) traditionnelles pour tenir compte de cette composition.

Il est également important d'examiner le comportement de ces mappings dans cette nouvelle configuration afin de de déterminer, pour chacun d'entre eux, à quel point les mouvements apparents des objets sont cohérents avec les mouvements de l'appareil mobile.

Une deuxième question concerne le référentiel dans lequel les utilisateurs situent la manipulation. En effet, cette configuration localement couplée se prête à deux interprétations opposées. L'appareil mobile peut être vu comme une « poignée » qui contrôle des objets 3D depuis un point de vue fixe, à la manière des objets tangibles dans le chapitre 3. Ou, à l'inverse, l'appareil peut être vu comme une « fenêtre » qui contrôle le *point de vue* sur des objets fixes, ce qui se rapproche de son utilisation comme fenêtre de réalité augmentée dans les chapitres précédents. Bien que fonctionnellement équivalentes, ces deux interprétations correspondent à des mappings opposés. Il est donc essentiel de déterminer quelle interprétation est privilégiée par les utilisateurs.

Enfin, une troisième question porte sur l'espace de manipulation de l'appareil mobile. Comme pour tout dispositif d'entrée, cet espace est limité par la portée de mouvement de l'utilisateur. Mais dans cette configuration localement couplée, l'écran lui-même se déplace aussi pendant la manipulation. Le retour visuel peut donc être perturbé ou interrompu avant même d'atteindre la portée maximale de mouvement. Il importe donc de proposer des solutions permettant de retarder autant que possible la perte du retour visuel.

Pour répondre à ces questions, il est nécessaire d'étudier en détail les possibilités de *mapping* entre les mouvements de l'appareil mobile et les objets manipulés. Pour cette raison, nous consacrons la majeure partie de ce chapitre à l'étude des mappings qu'autorise cette configuration particulière. La section 5.4 présente ainsi une revue des techniques existantes basées sur un principe similaire. Nous introduisons ensuite dans la section 5.5 un formalisme commun, avec lequel nous exprimons les principaux mappings dans une même notation. Ce formalisme nous permet alors d'analyser méthodiquement chaque mapping, afin d'identifier ses propriétés et ainsi déterminer comment celui-ci répond aux questions précédentes. La section 5.6 examine la première question, c'est-à-dire la cohérence entre les mouvements des objets manipulés sur l'écran et ceux de l'appareil mobile. La section 5.7 vise à répondre à la deuxième question sur le référentiel de manipulation, à travers une étude expérimentale pour déterminer l'interprétation favorisée par les utilisateurs. Enfin, la section 5.8 aborde la troisième question sur l'espace de manipulation, et propose plusieurs solutions pour y remédier.

Mais avant de commencer à examiner ces questions fondamentales, nous nous intéressons tout d'abord dans la section suivante à la manière d'implémenter ce nouveau mode d'interaction, et aux défis particuliers que pose cette configuration entièrement intégrée pour le suivi 3D et la sélection d'objets.

5.3 Implémentation

5.3.1 Suivi 3D

Le fait que l'interface entière soit constituée d'un seul élément, l'appareil mobile, présente un intérêt pour la portabilité mais pose également de nouvelles difficultés en terme d'implémentation. La première est celle du suivi 3D. Dans l'approche décrite dans le chapitre 3, les objets tangibles constituaient un « référentiel portable » pour l'appareil mobile. Connaître les déplacements relatifs entre les objets tangibles et l'appareil mobile suffisait au fonctionnement de l'interface. Dans le cas présent, cependant, il n'y a plus d'objets tangibles externes. Les déplacements de l'appareil mobile lui-même servent maintenant de modalité d'entrée. Il est donc nécessaire de pouvoir suivre les déplacements de l'appareil par rapport à l'environnement réel.

L'emploi d'un système de suivi externe, ou de marqueurs placés dans l'environnement, sont tous deux exclus car notre objectif est que l'interface soit entièrement portable et contenue dans un seul dispositif. La seule option restante est donc celle d'un suivi *inside-out* et *markerless* (sans marqueurs) des mouvements de l'appareil mobile par rapport à son environnement.

Le projet Tango [1] a pour objectif de concevoir un appareil mobile capable de modéliser son environnement et de suivre ses propres déplacements 6 DdL dans l'espace. Cette technologie repose sur la combinaison de plusieurs capteurs : une caméra couleur, une caméra grand-angle, un capteur de profondeur, et enfin une centrale inertielle (figure 5.2). Le suivi des mouvements de l'appareil mobile est réalisé par odométrie visuo-inertielle [2] (*visual-inertial odometry*), une technique qui ne nécessite pas de marqueurs dans l'environnement mais demande une certaine puissance de calcul. La caméra détecte et suit des points-clés dans l'environnement visuel, et ces informations sont combinées avec celles de la centrale inertielle pour une meilleure précision et stabilité. Le capteur de profondeur établit quant à lui un modèle 3D de l'environnement, permettant ainsi de compléter le suivi visuel des mouvements notamment suite à des rotations importantes [3]. Ce capteur peut intégrer un émetteur infrarouge (approche par « temps de vol »), ou ce dernier peut être distinct du capteur de profondeur (approche par « lumière structurée »).

Bien qu'une caméra couleur et une centrale inertielle soient aujourd'hui présents sur une majorité d'appareils mobiles, l'utilisation d'autres capteurs plus spécifiques pose la ques-

1. https://get.google.com/tango/
2. https://developers.google.com/tango/overview/motion-tracking
3. https://developers.google.com/tango/overview/depth-perception

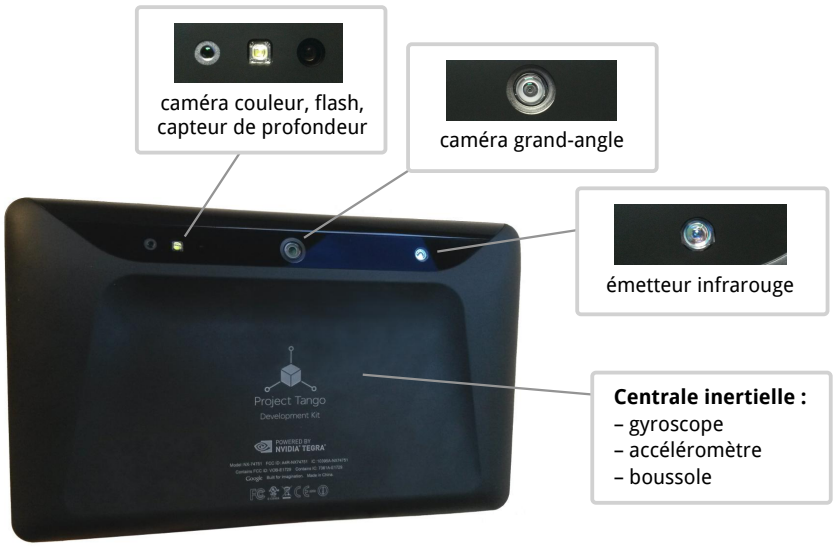


FIGURE 5.2 — La tablette « Yellowstone », une implémentation fonctionnelle de la technologie Tango. Ce type d'appareil mobile est capable de suivre ses propres mouvements sur 6 DdL dans l'espace réel.

tion de *l'accessibilité* d'une telle technologie pour les utilisateurs. Or, le projet Tango a récemment abouti à la commercialisation de plusieurs produits fonctionnels, dont la tablette « Yellowstone » illustrée dans la figure 5.2. Cette technologie est de plus amenée à être intégrée par des constructeurs tiers dans un nombre croissant d'appareils mobiles, dont deux modèles ont d'ores et déjà été annoncés [4]. La technologie Tango représente donc une possibilité d'implémenter notre approche dès aujourd'hui, sur un type de dispositif de plus en plus accessible pour le grand public.

5.3.2 Sélection d'objets

Un deuxième défi que pose cette nouvelle approche est celui de la sélection. En effet, comment interagir avec plusieurs objets virtuels à travers un unique support de manipulation ? Dans une interface constituée de multiples objets tangibles telle que celle du chapitre 3, chaque objet tangible peut être associé à un objet virtuel différent. Cela permet à l'utilisateur d'interagir avec plusieurs objets virtuels en même temps, en manipulant simplement les objets tangibles correspondants. On parle dans ce cas de « multiplexage spatial » [Fitzmaurice 1996]. Mais dans la configuration présentée ici, un seul objet tangible (l'appareil mobile) est disponible pour manipuler un nombre quelconque d'objets virtuels. Par consé-

4. `https://get.google.com/tango/`

quent, il devient nécessaire de *sélectionner* quel objet, ou groupe d'objets, doit être associé au support de manipulation à un instant donné. On parle alors de « multiplexage temporel » [Fitzmaurice 1996].

Cette étape de sélection peut être implémentée en tirant parti de l'écran tactile de l'appareil mobile. Si l'écran tactile est peu adapté à la manipulation 6 DdL, il reste cependant tout à fait approprié pour l'interaction 2D. Une technique 2D reconnue [Bowman *et al.* 2004] pour la sélection d'objets dans un espace 3D est le lancer de rayon (*raycasting*). L'utilisateur indique un point sur l'écran tactile à partir duquel un rayon perpendiculaire au plan de l'écran est généré, et le premier objet virtuel touché par ce rayon est sélectionné [Bolt 1980; Bowman *et al.* 2004]. L'utilisateur peut par la suite sélectionner un autre objet par une nouvelle opération de *raycasting*, ou désélectionner un objet en pointant un espace vide.

5.4 Travaux existants

Dans une première étape visant à répondre aux trois grandes questions identifiées précédemment (section 5.2), nous examinons les techniques d'interaction existantes utilisant l'écran lui-même comme dispositif d'entrée. Nous nous intéressons tout particulièrement au *mapping* choisi entre les mouvements de l'écran et ceux de l'objet manipulé. Bien qu'une majorité de ces techniques soient prévues pour une interaction 1D ou 2D, il existe malgré tout quelques exemples destinés à la manipulation 3D.

5.4.1 Techniques exploitant l'inclinaison de l'écran

L'un des premiers travaux sur ce sujet [Rekimoto 1996] propose d'exploiter l'inclinaison d'un ordinateur de poche (PDA) pour naviguer dans un menu affiché sur l'écran même de l'appareil (*tilt-based interaction*). D'après la description du mapping, la position du curseur dans le menu est fonction de l'angle d'inclinaison du PDA. Weberg *et al.* [2001] décrivent également une interface utilisant l'inclinaison d'un PDA pour déplacer un curseur dans un menu, mais l'angle d'inclinaison contrôle cette fois la *vitesse* de déplacement du curseur. Oakley et O'Modhrain [2005] étudient ces deux approches pour la navigation de menu. Nous pouvons donc déjà identifier deux types de mapping entre les mouvements de l'appareil mobile et ceux de l'objet manipulé, tel qu'un curseur ou un menu : le premier consiste à contrôler la position de l'objet (contrôle en position) tandis que le second consiste à contrôler la vitesse de déplacement de l'objet (contrôle en vitesse).

De nombreux autres travaux se sont intéressés à cette forme d'interaction basé sur l'inclinaison. Le défilement, qu'il s'agisse de listes, de documents ou d'images reste l'une des

tâches les plus fréquemment étudiées. Les premiers exemples de ce type [Small et Ishii 1997; Harrison *et al.* 1998; Bartlett 2000] semblent favoriser un contrôle en vitesse. Cependant, les mappings proprement dits ne sont décrits qu'informellement. Des travaux plus récents sur ce sujet [Hinckley *et al.* 2000; Cho *et al.* 2007; Eslambolchilar et Murray-Smith 2008] fournissent eux une meilleure description des mappings utilisés : dans les trois cas un contrôle en vitesse. L'inclinaison de l'écran est également employée pour des tâches de déplacement et de zoom dans un environnement 2D. La technique « RotoView » [Feinstein 2002], par exemple, consiste à incliner l'écran pour naviguer dans une carte. Joshi *et al.* [2012] décrivent un mapping hybride, combinant contrôle en position et en vitesse, pour visualiser des images panoramiques à 360°. L'inclinaison peut aussi être utilisée pour des tâches de pointage. Tsandilas *et al.* [2013] s'intéressent au contrôle en vitesse, au contrôle en position et à un contrôle hybride pour du pointage 1D. Teather et MacKenzie [2014] étudient quant à eux à le contrôle en position et le contrôle en vitesse pour une tâche de pointage 2D. Ce dernier cas se rapproche plus de la manipulation 3D que les exemples précédents, car le pointage 2D requiert la manipulation précise d'un objet (pointeur) sur l'écran sur plusieurs degrés de liberté.

5.4.2 Techniques exploitant la position de l'écran

Les techniques présentées dans la section précédente n'utilisaient que l'orientation de l'écran comme modalité d'entrée. Il est cependant possible d'exploiter de même la *position* de l'écran. Ce type d'interface est généralement qualifié de *spatially-aware display* (écran sensible à la position). Par exemple, Small et Ishii [1997] décrivent un système pour visualiser de longs journaux ou parchemins, qui consiste à fixer un écran sur des roues et à faire défiler son contenu lorsque l'ensemble est déplacé par l'utilisateur. Le mapping n'est pas décrit formellement, mais semble être un contrôle en position. Yee [2003] propose le « *peephole display* », dans lequel les translations d'un PDA (suivies en attachant ce dernier à des câbles) permettent de naviguer dans des espaces 2D plus grands que l'écran. Là encore, le mapping semble être un contrôle en position mais n'est pas décrit formellement. Spindler *et al.* [2014] présentent une approche similaire mais avec un suivi infrarouge. Wang *et al.* [2006] utilisent la caméra interne de l'appareil mobile pour suivre ses propres mouvements (suivi *inside-out*) et les appliquer à plusieurs tâches d'interaction 2D. Hansen *et al.* [2006] est un des rares exemples mentionnant à la fois un contrôle en position et un contrôle en vitesse, dans une configuration reposant également sur une caméra interne pour le suivi.

Dans l'ensemble, il apparaît qu'un nombre limité d'interfaces choisissent de se servir de la position de l'écran comme modalité d'entrée, si on le compare aux exemples bien plus nombreux qui se servent de l'orientation de l'écran. Cela pourrait s'expliquer par la plus grande complexité du suivi en position par rapport au suivi de l'orientation. L'orientation d'un appareil mobile peut facilement être obtenue avec des capteurs intégrés et peu onéreux,

tels que des gyroscopes et des accéléromètres. Ces capteurs sont aujourd'hui présents dans la plupart des appareils mobiles. À l'inverse, suivre la position de l'appareil est plus difficile. Certains des travaux cités plus haut utilisent des roues, des câbles, ou des caméras infrarouge externes, qui sont des dispositifs peu compatibles avec un objectif de mobilité. D'autres travaux exploitent la caméra de l'appareil lui-même pour ce suivi en position. Avec un nombre croissant d'appareils mobiles désormais équipés de caméras et l'augmentation de leur puissance de calcul, le suivi optique *inside-out* semble devenir la solution la plus prometteuse pour suivre les translations de l'écran. La récente sortie de plusieurs appareils mobiles Tango (section 5.3.1), capables de suivre leurs propres mouvements dans l'espace, pourrait ouvrir la voie à une plus grande popularisation de cette forme d'interaction.

5.4.3 Manipulation 3D par déplacement d'un appareil mobile

Bien que les mappings ci-dessus utilisés pour des tâches 1D ou 2D puissent servir de base pour des tâches de manipulation 3D, seuls les travaux portant réellement sur ces dernières tâches peuvent mettre en évidence les défis et contraintes spécifiques à l'interaction 3D.

Fitzmaurice *et al.* [1993] décrivent le système « Chameleon », dans lequel la position d'un écran portable permet de contrôler le point de vue sur une scène 3D affichée sur ce même écran. D'autres travaux [Tsang *et al.* 2002] ont ensuite étendu ce concept en ajoutant le contrôle de l'orientation, permettant ainsi une manipulation 6 DdL du point de vue. Ce type d'approche, cependant, vise avant tout à simuler une « fenêtre portable » sur la scène virtuelle. Le seul mapping possible est donc un contrôle en position isomorphique [5]. Aucune mention n'est faite d'autres mappings qui pourraient pourtant être utiles dans le cas de la manipulation d'objets 3D.

D'autres projets se servent, eux, des déplacements d'un appareil mobile pour de véritables tâches de manipulation 3D. Certains de ces projets se placent dans une configuration de réalité augmentée. Henrysson *et al.* [2005] et Marzo *et al.* [2014] décrivent une métaphore de « saisie d'objet » (*grasping*) dans laquelle l'objet manipulé est « fixé » à l'appareil mobile durant la manipulation. Un inconvénient de cette approche est qu'il est difficile de faire tourner l'objet manipulé sur lui-même sans lui appliquer aussi une translation. La scène virtuelle se situant dans un référentiel externe, et l'objet manipulé se situant dans le référentiel de l'appareil mobile mais à une certaine distance de son centre, il est nécessaire de déplacer l'appareil suivant un arc pour faire pivoter l'objet sur lui-même. Le même inconvénient apparaissait dans la technique de plan de coupe par tablette que nous avons étudiée dans le chapitre 3. La technique HOMER-S [Mossel *et al.* 2013] élimine ce problème en appliquant

5. Dans lequel l'échelle des mouvements est conservée.

séparément les rotations à l'objet manipulé. En conséquence, cependant, l'objet n'est plus fixe dans le référentiel de l'appareil mobile et peut donc sortir du champ de vision au cours des rotations. Aucune de ces deux approches ne peut éviter à la fois ces deux problèmes, à cause de la séparation intrinsèque entre l'objet manipulé et l'appareil mobile.

Une alternative à la manipulation en réalité augmentée est de placer également la scène virtuelle dans le repère de l'appareil mobile (c'est-à-dire dans le repère de l'écran), permettant ainsi de tourner librement l'appareil sans que l'objet manipulé ne sorte du champ de vision. Neale *et al.* [2013] présentent une interface de ce type pour visualiser des objets de musée sur une tablette. L'orientation 3D de l'objet sur l'écran peut être contrôlée soit par l'écran tactile, soit par l'inclinaison de l'appareil (contrôle en position ou contrôle en vitesse). Les translations ne sont cependant pas prises en compte. Daiber *et al.* [2012] proposent une interface pour appareil mobile permettant d'effectuer des rotations et des translations. Les rotations reposent sur l'inclinaison de l'appareil, mais les translations restent réalisées à travers l'écran tactile. Le dispositif « PDDM » proposé par Noma *et al.* [1996] constitue l'un des seuls exemples utilisant les déplacements d'un appareil mobile pour effectuer à la fois des translations et des rotations dans le repère de l'écran. Ainsi, cette configuration est la plus proche de celle que nous nous proposons d'étudier dans ce chapitre. Ce dispositif consiste en un PDA fixé sur un bras mécanique pour le suivi 3D. Les auteurs décrivent quatre mappings différents, tous de type contrôle en position. Les mappings sont illustrés mais pas formalisés. L'étude des mappings eux-mêmes n'est cependant pas l'objectif principal des auteurs. Les questions qui nous intéressent ici, telles que le référentiel dans lequel les utilisateurs situent la manipulation, sont uniquement mentionnées comme travaux futurs.

Comme nous pouvons le voir, il existe quelques rares travaux proposant d'utiliser les déplacements d'un appareil mobile pour manipuler des objets 3D dans le repère de l'écran. Mais ceux-ci ne traitent généralement que d'un sous-ensemble des mappings possibles, et ne se focalisent pas sur l'étude des mappings eux-mêmes. De plus, l'absence de formalisation rend difficile la généralisation de leurs résultats.

Dans la suite de ce chapitre, nous présentons donc dans un premier temps une formalisation des principaux mappings pour ce type de manipulation (section 5.5). Sur la base de ce formalisme, nous étudions ensuite en profondeur la manière dont ces mappings peuvent répondre aux trois questions que nous avons identifiées plus haut : en terme de cohérence entre les mouvements de l'appareil et ceux de l'objet (section 5.6), de référentiel de manipulation (section 5.7) et enfin d'espace de manipulation (section 5.8).

5.5 Formalisation des principaux *mappings*

Un *mapping*, aussi appelé « fonction de transfert », décrit la manière dont les mouvements d'un dispositif d'entrée (en l'occurrence l'appareil mobile) sont appliqués à un objet virtuel sur l'écran. Dans cette section, nous présentons trois des principales manières de réaliser ce transfert. Nous exprimons chaque mapping dans un formalisme commun (s'inspirant des notations de [Poupyrev *et al.* 2000]) afin de pouvoir les comparer et examiner leurs propriétés dans la section suivante. À la différence de beaucoup de travaux précédents, notre formalisme prend en compte à la fois les translations et les rotations.

5.5.1 Notation

Les termes pc_t et qc_t représentent respectivement la position et l'orientation de l'appareil mobile au temps t. Ces termes constituent les valeurs de ***c**ontrôle* du mapping, obtenues à partir du système de suivi et mesurées dans un repère arbitraire spécifique à ce système (le « repère de suivi »). La position pc_t est un vecteur 3D. L'orientation qc_t est un quaternion qui indique la rotation de l'appareil mobile par rapport à une orientation de base arbitraire.

Les termes pd_t et qd_t représentent respectivement la position et l'orientation de l'objet 3D manipulé au temps t. Il s'agit des valeurs de ***d**isplay* (affichage) du mapping, exprimées dans le repère de l'écran. La position pd_t est un vecteur 3D, et l'orientation qd_t est un quaternion qui indique la rotation de l'objet par rapport à une orientation de base sur l'écran. Les valeurs de pd_t et qd_t sont obtenues en appliquant la fonction de mapping aux valeurs de contrôle.

Le temps $t=0$ désigne le début de la manipulation, c'est-à-dire l'instant où l'utilisateur commence la manipulation [6]. Ainsi, les termes pc_0 et qc_0 représentent la position et l'orientation initiales de l'appareil mobile. De même, pd_0 et qd_0 représentent la position et l'orientation initiales de l'objet manipulé sur l'écran. Chaque étape de temps t marque l'instant où une nouvelle mesure est obtenue depuis le système de suivi. Pour simplifier la notation, on considère que cette mesure s'effectue à intervalles réguliers.

6. Il est préférable que les utilisateurs puissent enclencher et désenclencher le mode de manipulation, par exemple avec un bouton sur l'appareil mobile, afin de leur permettre de déplacer l'appareil dans un autre but que la manipulation (une opération appelée « débrayage », voir section 5.8.2).

5.5.2 Conversion des mouvements dans le repère de l'écran

Les valeurs de contrôle (positions pc_t et qc_t de l'appareil mobile) sont générées par le système de suivi dans un repère arbitraire qui lui est propre. Ainsi, les mouvements de l'appareil mobile sont eux aussi exprimés dans ce repère. Cependant, l'objet 3D manipulé se trouve lui dans le repère de l'écran. Il est donc nécessaire de convertir les mouvements de l'appareil mobile dans le repère de l'écran.

Dans la configuration « localement couplée » que nous étudions ici, l'écran et le dispositif d'entrée ne font qu'un. Appliquer une rotation à l'appareil mobile dans le repère de suivi produit également une rotation de *l'écran* dans le repère de suivi. Par conséquent, le repère de l'écran et le repère de suivi ne sont pas alignés. Si les mouvements de l'appareil mobile étaient *directement* appliqués à un objet sur l'écran, cet objet risquerait de se déplacer dans une direction inattendue (figure 5.3). Convertir les mouvements de l'appareil dans le repère de l'écran implique donc de rectifier cet alignement.

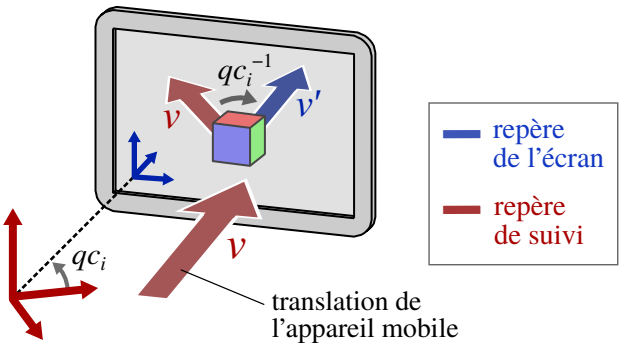


FIGURE 5.3 — Appliquer directement à un objet sur l'écran une translation v de l'appareil mobile présente le risque de translater l'objet dans une direction inattendue, qui varie selon l'orientation qc_i de l'appareil dans le repère de suivi. Pour obtenir le résultat souhaité (v'), il est nécessaire d'appliquer au vecteur v la rotation inverse qc_i^{-1}. La même opération doit être effectuée pour les mouvements de rotation.

Considérons une translation v et une rotation r de l'appareil mobile, mesurées dans le repère de suivi entre les temps i et j. Au début du mouvement, l'orientation de l'appareil mobile dans le repère de suivi est qc_i. Puisque l'écran est fixé sur l'appareil mobile, l'orientation de l'écran est elle aussi qc_i (par souci de simplicité, on suppose que l'écran est aligné avec le reste de l'appareil). Pour ramener v et r dans le repère de l'écran, cette orientation doit être *annulée* de manière à ré-aligner les deux repères (figure 5.3). On applique pour cela à v et r la rotation *inverse* qc_i^{-1}. Dans le cas du quaternion r, cette rotation inverse est appliquée à sa composante vectorielle (l'axe de rotation) par le biais de l'opération de conjugaison $qc_i^{-1}\, r\, (qc_i^{-1})^{-1}$, que l'on peut simplifier en $qc_i^{-1}\, r\, qc_i$. Dans le cas du vecteur v, celui-ci peut

être considéré comme un quaternion dont la composante réelle serait 0 et la même opération peut alors être appliquée.

La nouvelle translation v' et la nouvelle rotation r', correspondant à v et r ramenées dans le repère de l'écran, sont donc obtenues de la manière suivante :

$$v' = qc_i^{-1}\ v\ qc_i$$
$$r' = qc_i^{-1}\ r\ qc_i$$

Cette transformation permet d'exprimer les translations et les rotations de l'appareil mobile dans un repère stable quelle que soit l'orientation de l'appareil pendant la manipulation. Les mouvements v' et r' ainsi obtenus peuvent dès lors être appliqués à un objet 3D affiché sur l'écran.

Mais comme nous avons pu le voir dans les travaux précédents, il existe plusieurs manières d'appliquer les mouvements de l'appareil mobile à l'objet 3D manipulé.

5.5.3 Contrôle en position

Dans un mapping de type « contrôle en position » (*position control*), le dispositif d'entrée contrôle directement la position et l'orientation de l'objet manipulé [Zhai 1995]. Dans la configuration présente, cela signifie que les translations appliquées à l'appareil mobile contrôlent la position de l'objet 3D manipulé, et les rotations de l'appareil contrôlent l'orientation de l'objet.

Deux formes de contrôle en position peuvent être distinguées. La première, dite *absolue*, consiste à assigner directement à l'objet la position et l'orientation de l'appareil mobile telles que mesurées dans un repère donné. La seconde, dite *relative*, consiste à appliquer progressivement à l'objet les mouvements de l'appareil mobile (c'est-à-dire ses changements de position et d'orientation entre chaque instant t et $t-1$). Cette distinction s'inspire de la terminologie proposée par Poupyrev *et al.* [2000] pour les mappings de rotation, que nous appliquons également ici aux translations.

Contrôle en position absolu

Un mapping absolu [Poupyrev *et al.* 2000; Bowman *et al.* 2004] applique directement la position et l'orientation de l'appareil mobile (pc_t et qc_t) à la position et l'orientation de l'objet manipulé (pd_t et qd_t). Afin de permettre une manipulation en plusieurs étapes (débrayage), il est nécessaire de prendre en compte dans le mapping la position et l'orientation initiales de l'objet (pd_0 et qd_0). Pour que l'objet manipulé reste fixe tant que l'appareil mobile n'a pas été

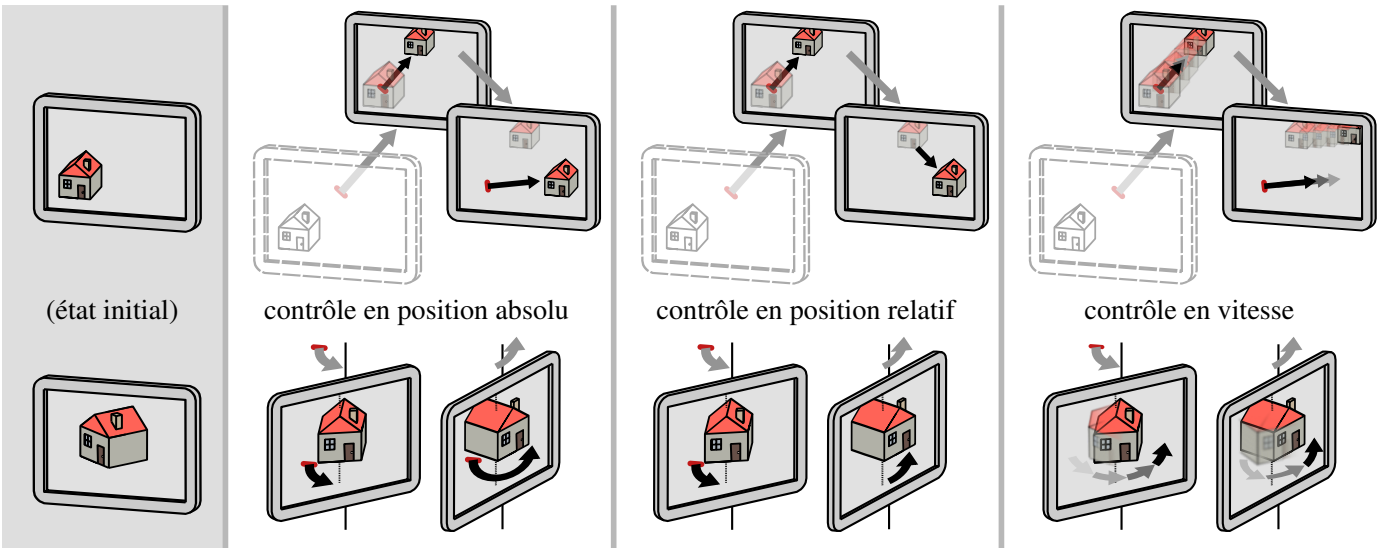


FIGURE 5.4 — Trois principales manières d'appliquer les mouvements d'un appareil mobile à un objet 3D sur l'écran. Contrôle en position absolu : le déplacement de l'appareil mobile *par rapport à sa position initiale* est appliqué à l'objet. Contrôle en position relatif : le déplacement *incrémental* de l'appareil mobile est appliqué à l'objet. Contrôle en vitesse : le déplacement de l'appareil mobile par rapport à sa position initiale agit sur la *vitesse* de déplacement de l'objet.

déplacé de son emplacement initial, il faut également soustraire la position et l'orientation initiales de l'appareil mobile (pc_0 et qc_0). Enfin, comme expliqué dans la section 5.5.2, les translations et rotations de l'appareil mobile doivent être converties dans le repère de l'écran. Puisque ces dernières sont mesurées par rapport à l'emplacement initial de l'appareil à $t=0$, une rotation de qc_0^{-1} doit leur être appliquée. Un mapping absolu peut donc être exprimé de la manière suivante :

$$\begin{aligned} \Delta pc_t &= qc_0^{-1}\,(pc_t - pc_0)\,qc_0 \\ \Delta qc_t &= qc_0^{-1}\,(qc_t\,qc_0^{-1})\,qc_0 \\ pd_t &= \Delta pc_t + pd_0 \\ qd_t &= \Delta qc_t\,qd_0 \end{aligned} \tag{5.1}$$

Contrôle en position relatif

Au lieu d'appliquer directement à chaque étape la position et l'orientation de l'appareil mobile, une alternative consiste à appliquer les déplacements *incrémentaux* de l'appareil. Un mapping relatif [Poupyrev *et al.* 2000; Bowman *et al.* 2004] applique les translations et rotations incrémentales de l'appareil mobile, mesurées entre les temps $t-1$ et t (c'est-à-dire $pc_t - pc_{t-1}$ et $qc_t\,qc_{t-1}^{-1}$), à la position et l'orientation actuelles de l'objet manipulé (pd_{t-1} et qd_{t-1}). Comme précédemment, les translations et rotations de l'appareil mobile

doivent être converties dans le repère de l'écran. Puisque celles-ci sont mesurées par rapport au temps $t-1$, une rotation de qc_{t-1}^{-1} doit être appliquée. Un mapping relatif peut donc être exprimé de la manière suivante :

$$\begin{aligned} \Delta pc_t &= qc_{t-1}^{-1}\,(pc_t - pc_{t-1})\,qc_{t-1} \\ \Delta qc_t &= qc_{t-1}^{-1}\,(qc_t\,qc_{t-1}^{-1})\,qc_{t-1} \\ pd_t &= \Delta pc_t + pd_{t-1} \\ qd_t &= \Delta qc_t\,qd_{t-1} \end{aligned} \tag{5.2}$$

Afin d'unifier la notation entre les différents mappings, l'expression ci-dessus suppose l'existence d'un système de suivi *absolu* capable de mesurer pc_t et qc_t par rapport à un repère fixe. Cependant, certains appareils mobiles (non-Tango) ne disposent que de capteurs relatifs, tels que les gyroscopes, qui mesurent des déplacements incrémentaux. Puisque les valeurs incrémentales retournées par ces capteurs sont déjà exprimées par rapport à l'appareil lui-même, il n'est pas nécessaire de les convertir dans le repère de l'écran. Par conséquent, celles-ci peuvent directement être utilisées à la place des termes Δpc_t et Δqc_t dans l'équation 5.2. Un corollaire est que le mapping relatif est le seul à être utilisable avec des capteurs relatifs, qui ne permettent pas de connaître pc_t et qc_t.

5.5.4 Contrôle en vitesse

Dans un mapping de type « contrôle en vitesse » (*rate control*), le dispositif d'entrée contrôle cette fois-ci la vitesse (linéaire ou angulaire) de l'objet manipulé [Zhai 1995]. Après avoir commencé la manipulation, l'utilisateur déplace l'appareil mobile depuis sa position initiale pc_0 et son orientation initiale qc_0. Dans un mapping de type contrôle en vitesse, la vitesse linéaire de l'objet manipulé augmente au fur et à mesure que l'appareil s'éloigne de sa position initiale, et diminue lorsque l'appareil revient vers ce point. La vélocité de l'objet est donc fonction du vecteur de translation $pc_t - pc_0$. De même, la vitesse angulaire de l'objet dépend de la rotation de l'appareil par rapport à son orientation initiale, soit $qc_t\,qc_0^{-1}$. Ces deux termes étant exprimés par rapport à l'emplacement initial de l'appareil (à $t=0$), une rotation de qc_0^{-1} doit être appliquée pour les convertir dans le repère de l'écran. Appliquer une vélocité à un objet équivaut[7] à l'ajouter à sa position et orientation actuelles (pd_{t-1} et qd_{t-1}) à chaque instant t. Un mapping de type contrôle en vitesse peut donc être

7. Comme indiqué dans la section 5.5.1, les intervalles de temps sont supposés constants dans notre notation.

obtenu de la manière suivante :

$$\begin{aligned} \Delta pc_t &= qc_0^{-1}\,(pc_t - pc_0)\,qc_0 \\ \Delta qc_t &= qc_0^{-1}\,(qc_t\,qc_0^{-1})\,qc_0 \\ pd_t &= \Delta pc_t + pd_{t-1} \\ qd_t &= \Delta qc_t\,qd_{t-1} \end{aligned} \tag{5.3}$$

5.5.5 Mapping d'ordre supérieur

Le contrôle en position est un mapping « d'ordre zéro » : il contrôle directement les positions et orientations de l'objet manipulé. Le contrôle en vitesse est un mapping « d'ordre un » : il contrôle la vitesse de l'objet, c'est-à-dire la dérivée première par rapport au temps de sa position et de son orientation.

Il est possible d'implémenter des mappings d'ordre supérieur, tels que le contrôle en accélération [8] [Massimino *et al.* 1989]. Cependant, de tels mappings s'avèrent être moins performants que le contrôle en position ou en vitesse [Massimino *et al.* 1989; Zhai 1995]. Par conséquent, nous ne nous intéresserons pas à de tels mappings dans la suite de ce travail.

5.5.6 Fonction de gain

Les mappings tels que nous les avons formalisés jusqu'ici ne modifient pas l'échelle des mouvements de l'appareil mobile avant de les appliquer à l'objet manipulé. Ces mappings sont dits *isomorphiques* [Poupyrev *et al.* 2000; Zhai 1995]. Cependant, il est possible d'étendre leur formulation pour amplifier ou atténuer les translations et les rotations. Les mappings deviennent alors *non-isomorphiques*.

Pour cela, nous introduisons une fonction de gain qui calcule un *facteur de gain* k_t (scalaire) pour chaque instant t :

$$k_t = \text{gain}(t)$$

Ce facteur de gain permet de modifier l'échelle des translations et des rotations de l'appareil mobile avant qu'elles ne soient appliquées à l'objet manipulé . Ainsi, l'objet sur l'écran peut se déplacer plus ou moins rapidement que l'appareil mobile. Dans le cas d'une translation,

8. Le contrôle en accélération, ou « d'ordre deux », peut être comparé à un environnement en impesanteur : un objet mis en mouvement ne peut être stoppé qu'en appliquant une action égale de direction opposée.

le facteur de gain doit en modifier l'amplitude sans changer sa direction. Ce résultat est facilement obtenu en multipliant le vecteur de translation Δpc_t par le facteur de gain k_t, ce qui donne un nouveau vecteur de translation $\Delta pc'_t$:

$$\Delta pc'_t = k_t \, \Delta pc_t$$

Dans le cas d'une rotation d'angle θ autour d'un axe donné, le facteur de gain doit en changer l'angle dans en modifier l'axe. Si la rotation Δqc_t est exprimée en tant que quaternion, il est possible d'effectuer une interpolation « slerp » [Shoemake 1985] depuis le quaternion identité $\mathbf{1}$ pour construire une nouvelle rotation $\Delta qc'_t$ autour du même axe mais avec un angle multiplié par k_t. Nous noterons cette opération sous la forme $\Delta qc_t^{k_t}$. Pour une rotation Δqc_t non-nulle, la nouvelle rotation $\Delta qc'_t = \Delta qc_t^{k_t}$ est ainsi obtenue comme suit :

$$\begin{aligned} \Delta qc'_t &= \text{slerp}(\mathbf{1}, \Delta qc_t, k_t) \\ &= \Delta qc_t^{k_t} \end{aligned}$$

En remplaçant les termes Δpc_t et Δqc_t par $k_t \Delta pc_t$ et $\Delta qc_t^{k_t}$ dans les mappings formalisés plus haut, il devient possible de modifier dynamiquement le facteur de gain qui s'applique aux translations et aux rotations.

Dans notre modèle, le facteur de gain est fonction de t. Contrairement à certains travaux précédents [Teather et MacKenzie 2014; Poupyrev *et al.* 2000; LaViola et Katzourin 2007] qui utilisaient un facteur de gain constant durant la manipulation, nous tenons à souligner l'importance que le gain puisse *changer dynamiquement*. Un facteur de gain variable est particulièrement utile pour augmenter à la fois la portée et la précision de la manipulation (section 5.8). Dans la suite de ce travail, nous nous assurons de toujours vérifier la validité de nos résultats en présence d'un facteur de gain variable.

5.6 Propriétés de compatibilité spatiale

Grâce à la formalisation introduite dans la partie précédente, nous pouvons maintenant examiner de manière rigoureuse les propriétés que présente chaque mapping dans cette configuration localement couplée. Cette analyse a pour objectif de répondre à la première question : à quel point les différents mappings assurent-ils la cohérence entre les mouvements de l'objet manipulé et ceux de l'appareil mobile ?

Lorsqu'un utilisateur se sert des déplacements d'un appareil mobile pour manipuler un objet 3D sur l'écran, il bénéficie d'un retour d'information (*feedback*) sous plusieurs formes. Premièrement, le fait de translater et de tourner l'appareil mobile lui-même fournit un retour kinesthésique en tant qu'objet tangible. Deuxièmement, les mouvements de l'objet manipulé

sur l'écran fournissent un retour visuel. Afin d'assurer une bonne usabilité de l'interface, notamment en terme de performance et de confort, il est essentiel d'assurer la cohérence de ces différentes formes de retour d'information [Smith et Smith 1987]. On nomme ce principe la « compatibilité du retour d'information » (*feedback compliance*, [Bowman *et al.* 2004]). Ici, nous nous focalisons plus spécifiquement sur la « compatibilité spatiale » du retour d'information (*spatial feedback compliance*) qui concerne la position et les déplacements des objets dans l'interface.

Dans cette section, nous analysons de manière formelle les propriétés de compatibilité spatiale de chaque mapping. Bien que certaines de ces propriétés aient fait l'objet d'analyses dans de précédents travaux, nous nous attachons ici à les réexaminer dans cette configuration localement couplée, en translation comme en rotation, et en présence d'un facteur de gain variable. Nous nous intéressons aux deux propriétés mentionnées par Bowman *et al.* [2004] et Poupyrev *et al.* [2000] : la compatibilité directionnelle et la remise à zéro (*nulling compliance*), auxquelles nous ajoutons la propriété de transitivité [Bade *et al.* 2005]. Enfin, nous examinons la question du sens de déplacement, qui mène directement à celle du référentiel de manipulation que nous traitons dans la section suivante.

5.6.1 Compatibilité directionnelle

La compatibilité directionnelle (*directional compliance*[9], [Bowman *et al.* 2004; Poupyrev *et al.* 2000]) signifie que l'objet virtuel manipulé se déplace *dans la même direction* que le dispositif d'entrée. « Même direction » signifie ici « suivant le même axe », le *sens* de déplacement étant l'objet d'une propriété différente étudiée par la suite.

Dans notre cas, l'objet manipulé se déplace sur l'écran et est contrôlé par les mouvements de l'appareil mobile. Cependant, l'écran est fixé sur l'appareil mobile et se déplace lui aussi pendant la manipulation. Contrairement à une configuration où l'écran serait fixe par rapport à l'environnement, il importe donc de préciser par rapport à quoi nous considérons les mouvements. Dans cette configuration localement couplée, la compatibilité directionnelle signifie que l'objet 3D se déplace *sur l'écran* dans la même direction que les mouvements de l'appareil mobile *par rapport à l'écran*.

Remarquons que la conversion décrite dans la section 5.5.2 permet d'aligner les déplacements de l'appareil mobile avec le repère de l'écran à $t=0$, mais ne garantit aucunement la compatibilité directionnelle pour la suite de la manipulation.

9. Aussi appelée « correspondance kinesthésique » (*kinesthetic correspondence*, [Britton *et al.* 1978]).

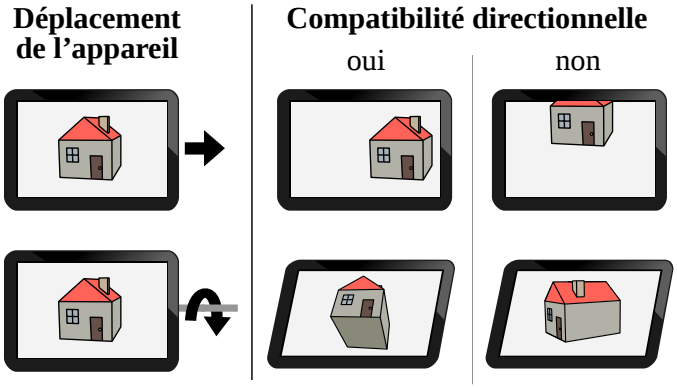


FIGURE 5.5 — Illustration de la propriété de compatibilité directionnelle.

Les déplacements de l'objet virtuel correspondent à son changement de position et d'orientation sur l'écran entre les temps $t-1$ et t : respectivement $pd_t - pd_{t-1}$ et $qd_t\, qd_{t-1}^{-1}$. De même, les déplacements de l'appareil mobile correspondent à son changement de position et d'orientation dans le repère de suivi entre les temps $t-1$ et t : respectivement $pc_t - pc_{t-1}$ et $qc_t\, qc_{t-1}^{-1}$. Comme nous l'avons vu, une rotation de qc_{t-1}^{-1} doit être appliquée aux mouvements de l'appareil pour les convertir dans le repère de l'écran. Formellement, la compatibilité directionnelle signifie que les translations de l'objet sont colinéaires avec les translations de l'appareil mobile rapportées à l'écran, et que les rotations de l'objet se font autour du même axe que les rotations de l'appareil rapportées à l'écran. Ainsi, un mapping présente une compatibilité directionnelle au temps t si et seulement s'il peut être exprimé de la manière suivante :

$$\begin{aligned} &\exists\,(\alpha,\beta) \in \mathbb{R}^2 : \\ &\qquad pd_t - pd_{t-1} = \alpha\,\left(qc_{t-1}^{-1}\,(pc_t - pc_{t-1})\,qc_{t-1}\right) \\ &\qquad qd_t\, qd_{t-1}^{-1} = \left(qc_{t-1}^{-1}\,(qc_t\, qc_{t-1}^{-1})\,qc_{t-1}\right)^{\beta} \end{aligned} \tag{5.4}$$

Contrôle en position relatif — Un mapping de type « contrôle en position relatif » présente *toujours* une compatibilité directionnelle, aussi bien en translation qu'en rotation. À partir de la formulation du mapping (équation 5.2) et en prenant en compte le facteur de gain k_t (section 5.5.6), nous obtenons $pd_t - pd_{t-1} = k_t\,\Delta pc_t = k_t\,\left(qc_{t-1}^{-1}\,(pc_t - pc_{t-1})\,qc_{t-1}\right)$ et $qd_t\, qd_{t-1}^{-1} = \Delta qc_t^{k_t} = \left(qc_{t-1}^{-1}\,(qc_t\, qc_{t-1}^{-1})\,qc_{t-1}\right)^{k_t}$; ce qui est équivalent à l'expression 5.4. Par conséquent, les mappings relatifs garantissent la compatibilité directionnelle tant pour les translations que pour les rotations.

Contrôle en position absolu — Un mapping de type « contrôle en position absolu » ne garantit *pas* la compatibilité directionnelle *dans le cas général*. Cependant, cette compatibilité peut être assurée dans certaines conditions. Comme nous le montrons dans l'annexe A.1,

les translations dans un mapping absolu peuvent présenter cette propriété *à condition* que le facteur de gain soit resté constant entre $t{=}0$ et $t{-}1$ *et* que l'appareil mobile n'ait pas changé d'orientation depuis $t{=}0$. Les rotations dans un mapping absolu ne présentent quant à elles cette propriété qu'autour de l'axe de rotation initial de l'appareil mobile (sauf si $k_t = -1$, un cas particulier traité par la suite).

Contrôle en vitesse — Un mapping de type « contrôle en vitesse » ne garantit *pas* non plus la compatibilité directionnelle, en translation comme en rotation. En effet, si l'on considère les translations de l'objet manipulé entre $t{-}1$ et t, tout en tenant compte du facteur de gain k_t, nous obtenons $pd_t - pd_{t-1} = k_t\,\Delta pc_t = k_t\left(qc_0^{-1}\,(pc_t - pc_0)\,qc_0\right)$; ce qui n'est pas équivalent à l'équation 5.4 dans le cas général. De même, pour les rotations, nous obtenons $qd_t\,qd_{t-1}^{-1} = \Delta qc_t^{k_t} = \left(qc_0^{-1}\,(qc_t\,qc_0^{-1})\,qc_0\right)^{k_t}$; ce qui n'est pas non plus équivalent à l'expression 5.4 dans le cas général. Cependant, nous montrons dans l'annexe A.2 que la compatibilité directionnelle peut être maintenue en translation aussi longtemps que l'axe de déplacement de l'appareil mobile *et* son orientation ne changent pas depuis le début de la manipulation. Les rotations ne conservent pour leur part cette propriété qu'autour de l'axe de rotation initial de l'appareil mobile.

5.6.2 Transitivité

La transitivité [Bade *et al.* 2005] est une propriété intuitive dans le monde réel : déplacer un objet d'un point A à un point B puis un point C, ou directement de A à C, a pour effet d'amener l'objet au même endroit. On est donc en droit de s'attendre qu'un système interactif présente la même propriété : translater ou tourner l'appareil mobile de A vers B puis C, ou directement de A à C, devrait amener l'objet virtuel manipulé vers la même position ou orientation. Cela permet notamment à l'utilisateur de ramener facilement l'objet virtuel à son emplacement initial, par exemple en cas d'erreur de manipulation, en ramenant simplement l'appareil mobile à sa configuration de départ. Cette propriété particulière de « remise à zéro » est appelée *nulling compliance* [Poupyrev *et al.* 2000]. La propriété de transitivité est une généralisation de la *nulling compliance* à n'importe quel emplacement visé.

Contrôle en position absolu — Un mapping de type « contrôle en position absolu » est *généralement* transitif, en translation comme en rotation. Les seuls termes dans la formulation de base du mapping (équation 5.1) pouvant varier au cours de la manipulation sont pc_t et qc_t. Le résultat d'un mapping absolu ne dépend donc que de la position et orientation *actuelles* de l'appareil mobile, quelles que soient les étapes précédentes de déplacement.

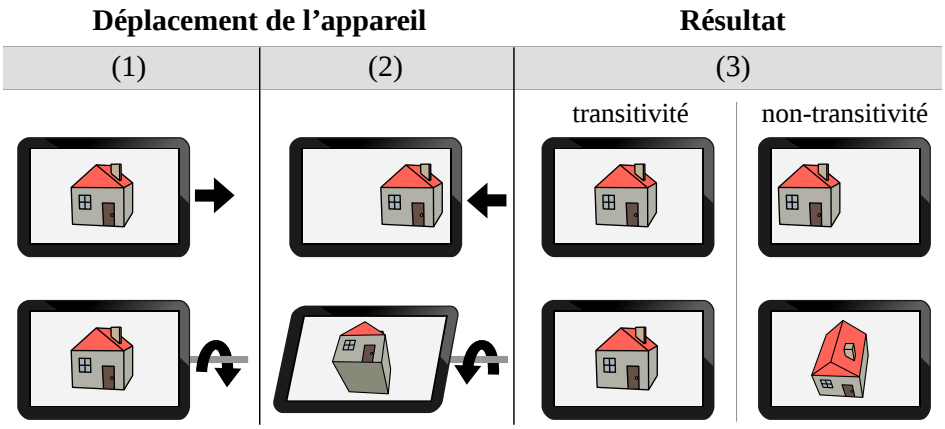


FIGURE 5.6 — Illustration de la propriété de transitivité, ici dans le cas particulier de la *nulling compliance*.

Cependant, il est nécessaire de prendre en compte le facteur de gain. Un mapping absolu *isomorphique* (sans facteur de gain) est toujours transitif. Un mapping non-isomorphique, quant à lui, n'est transitif que si la fonction de gain elle-même ne dépend pas d'autres variables que pc_t et qc_t. Un facteur de gain constant répond bien entendu à cette définition. Mais l'ajout de tout paramètre non-constant à la fonction de gain (autre que pc_t ou qc_t) ne permet plus de garantir la transitivité.

Dans tous les cas, les mappings isomorphiques comme non-isomorphiques sont tous deux transitifs vis-à-vis de la position et orientation initiales de l'appareil mobile (pc_0 et qc_0), car une translation ou une rotation nulle n'est pas affectée par le facteur de gain. Les mappings absolus présentent donc *toujours* la propriété de *nulling compliance*. Ce dernier point a été initialement mis en évidence par Poupyrev *et al.* [2000] dans le cas des rotations, mais, comme nous venons de le voir, reste aussi vrai pour les translations.

Contrôle en position relatif — Un mapping de type « contrôle en position relatif » n'est *pas* transitif (ou *nulling compliant*) dans le cas général. Cependant, nous pouvons identifier certaines conditions dans lesquelles un tel mapping peut être transitif. Comme nous le montrons dans l'annexe A.3, les translations dans ce mapping peuvent être transitives à condition que le facteur de gain *et* l'orientation de l'appareil mobile soient restés constants depuis le début de la manipulation. Les rotations, cependant, ne présentent pas cette propriété (sauf si $k_t = -1$, un cas particulier traité dans la section suivante).

Contrôle en vitesse — Un mapping de type « contrôle en vitesse » n'est *pas* transitif (ni même *nulling compliant*) tant en translation qu'en rotation, car il est intrinsèquement dépendant du temps. La trajectoire de l'objet manipulé à travers un tel mapping dépend du temps passé par l'appareil mobile entre les points A, B et C. Il n'est donc pas possible de prédire

l'emplacement final de l'objet à partir des seules positions et orientations intermédiaires de l'appareil mobile.

5.6.3 Sens de déplacement

Si la propriété de compatibilité directionnelle signifie que l'objet manipulé se déplace « suivant le même axe » que l'appareil mobile, il n'en demeure pas moins deux possibilités de déplacement le long de cet axe : dans le *même sens* que l'appareil mobile, ou dans le sens *opposé*. Ainsi, un mapping peut très bien présenter une compatibilité directionnelle alors même que l'objet manipulé se déplace dans le sens opposé à celui de l'appareil mobile, à partir du moment où ces déplacements se produisent suivant le même axe. Une telle situation, loin d'être anecdotique, permet notamment de placer la manipulation dans un référentiel différent comme nous l'étudions plus en détail dans la section 5.7.

En utilisant le formalisme introduit plus haut, le sens de déplacement de l'objet manipulé peut être directement modifié en changeant le signe du facteur de gain k_t. Un facteur de gain positif, ou aucune valeur explicite (c'est-à-dire un gain de 1) a pour effet de faire bouger l'objet dans le même sens que l'appareil mobile. Un facteur de gain négatif fait, quant à lui, bouger l'objet dans le sens opposé. Les deux alternatives sont *fonctionnellement* équivalentes : entre deux fonctions de gain de signe opposé, la trajectoire de l'objet manipulé est identique à une symétrie près. Dans le cas général, les propriétés de compatibilité spatiale de chaque mappings ne sont donc pas modifiées.

Cependant, il existe une valeur de gain spécifique qui a un effet notable dans les mappings de type « contrôle en position » : lorsque le gain a une valeur constante **égale à -1**. Cette valeur inverse le sens des translations et des rotations, sans modifier leur amplitude. Autrement dit, un facteur de gain de -1 applique l'exact opposé des mouvements de l'appareil mobile sur l'objet manipulé.

Cela produit un effet particulier dans le cas des rotations. L'objet est affiché sur l'écran de l'appareil mobile, qui tourne lui-même durant la manipulation. En appliquant l'exact opposé des rotations de l'appareil mobile sur l'objet, ces rotations sont annulées du point de vue de l'utilisateur. L'objet manipulé semble donc conserver une *orientation fixe* par rapport à l'environnement réel. Plus intéressant encore, l'effet est identique tant dans un mapping absolu que dans un mapping relatif. Comme nous le montrons dans l'annexe A.4, cette valeur rend les mappings absolus et relatifs strictement *équivalents* en terme de rotation. Étant équivalents, les deux mappings partagent les mêmes propriétés. Le résultat est donc un mapping en rotation qui garantit à la fois la compatibilité directionnelle (en tant que mapping relatif) et la transitivité (en tant que mapping absolu de gain constant).

Mais cette équivalence ne s'applique toutefois pas aux translations (cf. annexe A.4). Pour cette raison, on remarquera qu'effectuer *simultanément* des translations et des rotations avec un facteur de gain de -1 n'est *pas* équivalent à une interface de type « réalité augmentée » (où l'objet aurait à la fois une position et une orientation fixes par rapport à l'environnement réel), car la formulation des translations telle que nous l'avons présentée dépend aussi de l'orientation de l'appareil mobile.

5.6.4 Tableau récapitulatif

Le tableau 5.1 résumé les propriétés de compatibilité spatiale des trois types de mapping étudiés jusqu'ici. La mention de $k_t = -1$ indique que la propriété n'est garantie qu'en présence d'un facteur de gain k_t resté constant et égal à -1 depuis le début de la manipulation.

Tableau 5.1 — Propriétés de compatibilité spatiale pour chaque mapping dans cette configuration localement couplée.

	compatibilité directionnelle		**transitivité**	
	translations	rotations	translations	rotations
absolu	non	$k_t = -1$	oui	oui
relatif	oui	oui	non	$k_t = -1$
vitesse	non	non	non	non

5.6.5 Choix des propriétés de compatibilité

Comme le montre le tableau 5.1, aucun des trois types de mappings ne garantit à la fois la compatibilité directionnelle et la transitivité dans tous les cas. Les rotations ne présentent ces deux propriétés qu'à condition que le mapping soit de type contrôle en position, strictement isomorphique, et de sens opposé ($k_t = -1$). Les translations ne présentent ces deux propriétés que si le mapping est de type contrôle en position, sans rotations simultanées, et avec un facteur de gain constant. Il s'agit là de contraintes importantes, qui peuvent ne pas être acceptables dans de nombreuses situations. En pratique, il faudra donc déterminer laquelle de ces deux propriétés est la plus importante selon la situation.

Compatibilité directionnelle — La compatibilité directionnelle garantit que la direction de déplacement de l'objet manipulé correspond à celle du dispositif d'interaction. Cela permet à l'utilisateur d'anticiper les mouvements de l'objet et de planifier sa trajectoire, facilitant ainsi la manipulation [Bowman *et al.* 2004]. Il existe en effet des résultats expérimentaux indiquant que la compatibilité directionnelle joue un rôle dans l'usabilité et la performance.

Fitts et Seeger [1953] ont montré que les utilisateurs avaient des réactions plus lentes et moins précises lorsque le retour visuel n'était pas aligné spatialement avec leurs actions. De plus, cette réduction de performance persistait même après entraînement. Ware et Arsenault [2004] se sont intéressés aux effets d'une rotation entre les référentiels d'entrée et d'affichage (c'est-à-dire une non-compatibilité directionnelle) dans une tâche de rotation 3D. Leurs résultats révèlent une nette réduction de performance pour des angles importants, même si les effets étaient plus limités avec des rotations plus faibles. Van Rhijn et Mulder [2006] ont montré que les performances dans une tâche de manipulation 3D (translation et rotation) étaient meilleures lorsque les mouvements de l'objet étaient cohérents avec ceux du dispositif d'interaction dans le référentiel de l'objet (ce qui correspond à notre définition de la compatibilité directionnelle). Dans le cas contraire, le temps d'accomplissement de la tâche augmentait de manière importante.

La compatibilité directionnelle apparaît donc comme essentielle pour une manipulation efficace dans le cas général. Il peut être envisageable de s'en passer si l'orientation du dispositif d'entrée ne change que très peu durant l'interaction, de sorte que la différence d'alignement avec le référentiel de l'objet reste faible [Poupyrev *et al.* 2000].

Transitivité — La transitivité, ou la *nulling compliance*, peut être utile dans certaines situations. Comme expliqué précédemment, cette propriété permet de rectifier une erreur de manipulation en exploitant la mémoire musculaire [Bowman *et al.* 2004] pour ramener l'objet à son emplacement initial (*nulling compliance*) ou à n'importe quel emplacement intermédiaire. La transitivité peut également s'avérer utile lorsque l'objet présente une orientation privilégiée, comme l'axe vertical dans le cas d'un bâtiment, d'un paysage, ou encore d'un visage [Buda 2012]. L'objet manipulé peut ainsi être remis à l'endroit de manière prévisible en ramenant simplement l'appareil mobile vers une orientation connue.

Selon Poupyrev *et al.* [2000], la forme du dispositif d'interaction lui-même est également importante. Si celui-ci possède une orientation « de base », l'absence de *nulling compliance* peut être perçue par l'utilisateur et perturber l'usabilité. Ici, le dispositif d'interaction est l'appareil mobile. La plupart des appareils mobiles actuels ont une forme plate destinée à faire face à l'utilisateur. Ils ont ainsi une orientation de base que l'utilisateur perçoit visuellement et haptiquement. L'absence de *nulling compliance* (et *a fortiori* de transitivité) est donc apparente durant la manipulation. D'autres appareils, tels que le pCubee [Stavness *et al.* 2010], n'ont quant à eux pas d'orientation privilégiée. L'absence de transitivité des rotations est donc moins perceptible sur de tels dispositifs.

Malgré l'intérêt de la transitivité dans les cas ci-dessus, cette propriété reste pourtant incompatible avec la compatibilité directionnelle dans le cas général comme nous l'avons vu plus haut. Il existe au moins une étude [Buda 2012] montrant que des mappings présentant une compatibilité directionnelle offraient une meilleure performance et étaient notés plus

favorablement que des mappings transitifs. Cela suggère donc que la transitivité ne devrait être préférée à la compatibilité directionnelle que dans certains cas spécifiques.

5.7 Référentiel allocentrique ou égocentrique

Le fait de translater et tourner un objet 3D peut être interprété de deux manières [Klatzky 1998a]. Une interprétation possible est que l'objet lui-même se déplace par rapport à un point de vue fixe. On appelle cela un référentiel *allocentrique* (ou « exocentrique »). Une autre interprétation est que l'objet reste fixe tandis que le point de vue se déplace dans le sens opposé. On parle alors de référentiel *égocentrique*.

Or, la configuration localement couplée que nous étudions ici présente une particularité. L'appareil mobile, qui sert à manipuler les objets, se confond avec l'écran qui sert de point de vue. Pour cette raison, cette configuration se prête à la fois aux deux interprétations (figure 5.7) :

— soit l'appareil mobile peut être vu comme une « poignée » servant à contrôler les objets sur l'écran (interprétation allocentrique) ;

— soit l'appareil mobile peut être vu comme une « fenêtre » servant à contrôler le *point de vue* sur des objets fixes (interprétation égocentrique).

Dans l'interprétation allocentrique, les objets manipulés se déplacent dans le *même sens* que la « poignée » représentée par l'appareil mobile. Dans l'interprétation égocentrique, les objets se déplacent dans le sens *opposé*, comme si le point de vue était contrôlé par le biais d'une « fenêtre » tenue dans la main.

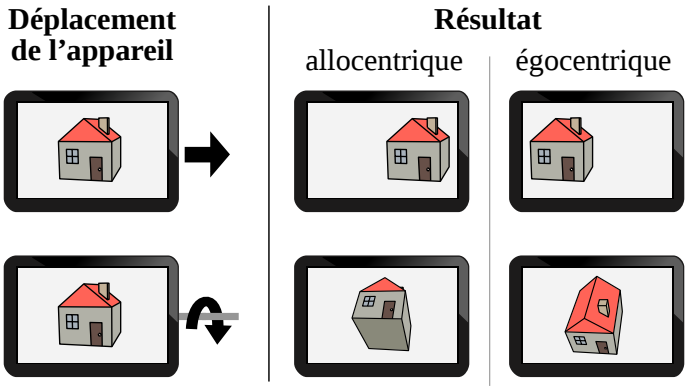


FIGURE 5.7 — Comparaison entre un mapping allocentrique et son équivalent égocentrique.

L'une et l'autre de ces alternatives peuvent être simplement obtenues en inversant le signe de la fonction de gain lors de l'implémentation du mapping, comme indiqué dans la section 5.6.3, et les mappings résultants sont fonctionnellement équivalents.

Cependant, les deux options étant mutuellement exclusives, il demeure nécessaire de déterminer *laquelle proposer* aux utilisateurs. Ce choix est particulièrement important si l'interface ne prévoit pas la possibilité de passer d'un mapping à l'autre. Mais dans tous les cas, le choix effectué impactera les utilisateurs lors de la première utilisation.

Puisque la configuration localement couplée se prête tout autant aux deux possibilités, ce choix ne peut donc reposer que sur l'interprétation favorisée subjectivement par une majorité d'utilisateurs. Selon le principe de « compatibilité stimulus-réponse »[10] [Chan *et al.* 2003], établi dans le domaine de l'ergonomie, l'alternative retenue devrait être celle à laquelle *s'attendent le plus souvent* les utilisateurs. Choisir un mapping compatible est essentiel pour garantir une bonne usabilité tout en limitant la fatigue [Chan *et al.* 2003], ainsi que pour éviter les erreurs de manipulation telles que les inversions accidentelles [Diaz et Sims 2005]. Il a été montré que, même après entraînement, un mapping non-compatible entraîne toujours une réduction de performance [Chan *et al.* 2003]. Dans cette section, nous cherchons donc à déterminer quelle interprétation est favorisée par les utilisateurs afin d'être en mesure de choisir un mapping approprié.

5.7.1 Travaux existants

Référentiels allocentrique et égocentrique

Chacune des interprétations ci-dessus correspond à une manière différente de se représenter la relation entre les objets dans l'espace, ce que l'on appelle un *référentiel*. La littérature en matière de cognition spatiale en distingue principalement deux [Klatzky 1998b] : le référentiel « allocentrique » (ou exocentrique) et le référentiel « égocentrique ». La majorité des études existantes se focalise cependant sur la relation entre une personne et les objets qui l'entourent. Dans notre cas, il existe à la fois une relation entre l'utilisateur et l'appareil mobile, et une relation entre l'appareil mobile et les objets virtuels affichés sur son écran. Il est donc nécessaire d'examiner de plus près comment les deux termes s'appliquent à cette configuration spécifique.

10. Abrégé en « compatibilité » dans le reste de cette section. Il est important toutefois de ne pas confondre les propriétés de « compatibilité spatiale » de la section précédente, qui sont des propriétés liées aux mappings eux-mêmes, et la « compatibilité avec les attentes des utilisateurs » qui relève elle du domaine subjectif.

Klatzky [1998b] définit ces deux termes dans le contexte de la navigation par déplacement du corps : le référentiel égocentrique serait lié au « point de vue du sujet »[11] et le référentiel allocentrique serait lié à un « cadre externe et indépendant du sujet »[12]. Burgess *et al.* [2004] s'intéressent à ces deux référentiels dans une tâche d'actualisation spatiale (*spatial updating*, le fait de déduire le mouvement relatif de plusieurs objets à partir d'une configuration initiale et d'une configuration finale), qui présente une certaine similarité avec la question étudiée ici. Ils définissent le référentiel égocentrique comme étant lié au « mouvement propre » (*« self-motion »*) et le référentiel allocentrique comme « associant la position des objets à un repère extérieur »[13]. Poupyrev *et al.* [1998] définissent ces deux référentiels dans le contexte des environnements immersifs : les techniques d'interaction égocentriques seraient celles liés au point de vue de l'avatar, tandis que les techniques « exocentriques » (c'est-à-dire allocentriques) seraient celles effectuées depuis un endroit externe. Dans une étude sur les inversions accidentelles [Diaz et Sims 2005], la condition égocentrique était là encore vue par les yeux de l'avatar virtuel, et la condition allocentrique était vue depuis l'extérieur de son corps.

Dans l'ensemble, le terme « égocentrique » semble être associé à l'idée du point de vue, et le terme « allocentrique » à l'idée d'un point extérieur fixe. Dans notre cas, lorsque l'objet manipulé se déplace dans la même direction que l'appareil mobile, il semble être contrôlé directement par les déplacements de l'appareil mobile et donc bouger comme lui par rapport à l'espace extérieur. Nous qualifions donc ce type de mapping « d'allocentrique ». Lorsque l'objet manipulé se déplace dans la direction opposée, l'appareil mobile semble à l'inverse contrôler le *point de vue* sur cet objet. Nous qualifions donc ce type de mapping « d'égocentrique ».

Stéréotypes de population

La relation entre le mouvement d'un dispositif d'entrée et le déplacement résultant sur l'écran a depuis longtemps fait l'objet d'études dans le domaine de l'ergonomie. Bien que cette relation (*mapping*) puisse prendre de nombreuses formes, certaines d'entre elles correspondent mieux que d'autres au modèle mental de l'utilisateur. De tels mappings sont alors dits « compatibles » [Fitts et Seeger 1953; Chan *et al.* 2003]. Notre objectif ici est de déterminer quel mapping est le plus compatible avec la population ciblée.

Lorsqu'il existe plusieurs alternatives fonctionnellement équivalentes pour réaliser une même tâche, comme les deux mappings que nous étudions ici, l'option la plus souvent *attendue*

11. *« In an egocentric reference frame, locations are represented with respect to the particular perspective of a perceiver »*
12. *« An allocentric reference frame locates points within a framework external to the holder of the representation and independent of his or her position. »*
13. *« The association of object locations to external landmarks »*

par les utilisateurs est appelée « stéréotype de population » [Fitts 1951; Wiebe et Vu 2009]. Un certain nombre de ces stéréotypes ont été identifiés dans la littérature. Par exemple, le principe de Warrick [Warrick 1947; Wiebe et Vu 2009] énonce que l'objet manipulé devrait suivre la même direction que le point le plus proche du dispositif d'entrée. Le principe « sens horaire augmente » (*clockwise-to-increase*, [Wiebe et Vu 2009]) énonce que l'objet manipulé devrait « augmenter », dans le cas d'un curseur, ou plus généralement se déplacer « vers le haut », lorsque le dispositif d'entrée est tourné dans le sens des aiguilles d'une montre. Cependant, ces principes partent de l'hypothèse que le dispositif d'entrée peut être déplacé indépendamment de l'écran. Dans la configuration localement couplée étudiée ici, les deux fonctions sont remplies par un même objet : l'appareil mobile. Les résultats existants concernant les stéréotypes de population ne peuvent donc pas être appliqués tels quels, à moins qu'ils n'aient été validés dans le cas d'une manipulation localement couplée.

Mapping favorisé dans une configuration localement couplée

Comme nous l'avons vu précédemment, une certaine quantité de travaux ont proposé d'utiliser le déplacement d'un appareil mobile pour contrôler des objets affichés sur son propre écran. Mais seul un petit nombre de ces travaux mentionnent explicitement l'existence de deux mappings possibles (direct/allocentrique et inversé/égocentrique) et la raison derrière l'alternative finalement retenue.

Dans l'interface décrite par Rekimoto [1996], les deux possibilités sont mentionnées mais le choix retenu (déplacer le menu derrière un curseur fixe, équivalent à notre interprétation égocentrique) repose uniquement sur des critères techniques au lieu d'être basé sur une étude utilisateur. Weberg *et al.* [2001] font le choix opposé (déplacer le curseur dans un menu fixe, équivalent à notre interprétation allocentrique) dans leur interface, sur la base du fait que cela « semblait très intuitif et naturel »[14]. Bartlett [2000] mentionne l'existence de deux groupes d'utilisateurs avec des modèles mentaux différents, chacun s'attendant à ce que l'image manipulée se déplace dans une direction opposée lorsqu'ils inclinent le dispositif. Hinckley et Song [2011] mentionnent également qu'un peu plus de la moitié de leurs participants semblaient avoir un modèle mental opposé aux autres au moment d'incliner l'appareil mobile pour contrôler le niveau de zoom. Bien que les travaux ci-dessus aient été conduits dans une configuration localement couplée, ils ne fournissent cependant pas suffisamment d'arguments pour montrer une réelle préférence entre l'interprétation allocentrique et égocentrique. De plus, ces études ont été menées sur des tâches de manipulation 1D ou 2D, et ne sont donc pas forcément généralisables à la manipulation 3D.

14. *« [...] felt very intuitive and natural. »*

Il n'existe que très peu d'études dans ce domaine portant sur de véritables tâches de manipulation 3D, mais aucune n'est placée dans une configuration localement couplée. Kaminaka et Egli [1985] étudient le mapping favorisé par les utilisateurs pour translater et orienter un cube à travers un levier. Le levier contrôle alternativement chacun des trois axes de translation ou de rotation. Bien qu'il s'agisse d'une tâche de manipulation 3D, le dispositif d'entrée 1D et la configuration qui n'est pas localement couplée rend difficile la réutilisation des résultats de cette étude dans notre situation. Diaz et Sims [2005] étudient les « inversions accidentelles » pour des rotations 3D, c'est-à-dire l'erreur qui se produit lorsque les utilisateurs rencontrent un mapping opposé à leur modèle mental. De telles inversions permettent d'identifier le stéréotype de population pour les rotations. Cependant, cette étude repose sur une souris 2D comme dispositif d'entrée, ainsi qu'un écran externe, ce qui rend là encore difficile la généralisation des résultats à la configuration qui nous intéresse ici.

Aucun des travaux existants ne nous permettant de connaître le stéréotype de population dans notre configuration, nous avons mené une étude expérimentale pour le déterminer.

5.7.2 Étude : détermination de l'interprétation attendue

Déterminer l'interprétation (allocentrique ou égocentrique) généralement *attendue* par les utilisateurs est une tâche délicate : beaucoup de biais peuvent en effet affecter le résultat d'une telle étude, comme une exposition antérieure à l'interface, des effets d'apprentissage, et même la façon de décrire le mode d'interaction aux utilisateurs. Nous avons donc envisagé plusieurs protocoles expérimentaux pour déterminer l'interprétation favorisée.

La première, inspirée par de précédents travaux sur les stéréotypes de population [Wiebe et Vu 2009] et sur l'actualisation spatiale [Burgess *et al.* 2004], serait de montrer aux participants une description non-interactive d'une tâche de manipulation. Par exemple, une image représentant un objet dans une position initiale, et une autre image du même objet dans une position cible. Les participants devraient alors indiquer dans quelle direction ils déplaceraient l'appareil mobile pour obtenir le même résultat. Ce protocole présente l'avantage de ne fournir que le minimum d'information nécessaire pour répondre à la question qui nous intéresse, réduisant ainsi le risque de variables parasites. Cependant, ce protocole a également un important défaut : à aucun moment les participants n'utilisent réellement l'interface. Cet inconvénient ne serait pas forcément gênant dans le cas d'une interface basée par exemple sur une souris 2D, un dispositif familier pour beaucoup d'utilisateurs. Mais dans le cas présent, il est probable que les participants aient des difficultés à imaginer les mouvements à effectuer sur une forme d'interface qu'ils n'ont vraisemblablement jamais rencontrée auparavant.

Une autre manière serait de demander aux participants d'effectuer une série de tâches de manipulation 3D dans les deux modes allocentrique et égocentrique, puis d'analyser les trajectoires effectuées afin d'y détecter des inversions accidentelles [Diaz et Sims 2005]. De telles inversions constituent une indication objective que le mapping utilisé ne correspond pas au modèle mental de l'utilisateur. Ce protocole présente aussi l'avantage de laisser les participants se servir réellement de l'interface, même si des effets d'apprentissage peuvent alors apparaître après un certain temps d'utilisation. Cependant, détecter des inversions accidentelles dans des trajectoires 3D totalement libres, produites par des utilisateurs non entraînés, reste difficile et peu fiable. Il serait possible d'entraîner les utilisateurs à générer des trajectoires plus « propres », mais un tel apprentissage risquerait d'altérer leur préférence entre les deux mappings par rapport au reste de la population.

Une troisième manière serait de faire effectuer aux participants des tâches de manipulation dans les modes allocentrique et égocentrique, puis de leur demander de *noter* à quel point chaque mapping leur semble « naturel » ou « non naturel ». Pour cela, les participants indiquent, à travers un score attribué à chaque mapping, à quel point l'objet manipulé se déplace et tourne *dans la direction qu'ils attendaient*. Là encore, ce protocole permet aux participants d'utiliser réellement l'interface. Mais l'évaluation est désormais purement subjective. par rapport au protocole précédent, cette mesure présente l'avantage d'être réalisable en pratique même avec des utilisateurs novices, ce qui évite les biais associés à un entraînement préalable. De plus, il se peut que certains participants considèrent les deux alternatives acceptables (ou les trouvent toutes deux peu naturelles). En laissant les participants noter chaque mapping au lieu de simplement choisir le « meilleur » des deux, ce protocole offre une vision plus détaillée de la question à étudier. Par conséquent, nous avons choisi ce dernier protocole pour notre étude expérimentale.

Indices contextuels

De précédents travaux [Burgess *et al.* 2004; Diaz et Sims 2005] ont montré que le référentiel attendu par les utilisateurs n'est pas seulement une question d'interprétation personnelle, mais peut aussi être affecté par des indices présents dans l'environnement, en d'autres termes des *indices contextuels*. Dans une interface totalement portable (répondant à notre critère de portabilité), l'environnement extérieur à l'appareil mobile n'est pas contrôlable. C'est pourquoi nous nous intéressons plutôt aux indices contextuels fournis par *l'environnement virtuel*. Celui-ci pouvant être radicalement différent d'une application à l'autre, il est important de déterminer à quel point la scène virtuelle affichée à l'écran influence le référentiel favorisé par l'utilisateur.

Premièrement, nous pouvons supposer que le type d'objet virtuel manipulé peut avoir une influence. Si cet objet représente quelque chose qui serait tenu dans la main dans le monde

réel (par exemple une figurine, un fruit...), il est possible que les utilisateurs s'attendent à pouvoir le manipuler directement, favorisant ainsi l'interprétation allocentrique. Si, à l'inverse, l'objet représente quelque chose qui pourrait faire partie du décor (tel qu'une maison, un paysage...), il se peut que les utilisateurs s'attendent à *se déplacer eux-mêmes* par rapport à l'objet au lieu de le manipuler directement, favorisant alors l'interprétation égocentrique.

Deuxièmement, les relations géométriques entre les objets de la scène pourraient également jouer un rôle. Si l'objet manipulé se déplace visiblement sur l'écran tandis que les autres objets de la scène restent fixes (par rapport à l'écran), alors ces autres objets pourraient être perçus comme un environnement extérieur au sein duquel l'objet manipulé se déplace, favorisant ainsi l'interprétation allocentrique. Si l'objet manipulé est vu de l'intérieur, alors il pourrait lui-même être perçu comme entourant l'appareil mobile, renforçant alors l'interprétation selon laquelle l'appareil se déplace *dans* l'objet (interprétation égocentrique) au lieu de déplacer l'objet (interprétation allocentrique).

Hypothèses

Sur la base des considérations ci-dessus, nous pouvons formuler les hypothèses suivantes sur l'interprétation favorisée par les utilisateurs dans plusieurs situations :

— **H1** : Lorsque l'objet manipulé est vu de l'intérieur, l'interprétation favorisée est égocentrique ;
— **H2** : Lorsque l'objet manipulé se déplace au sein d'un environnement virtuel fixe, l'interprétation favorisée est allocentrique ;
— **H3** : Lorsque le type d'objet manipulé représente un élément typiquement fixe (par exemple une maison), l'interprétation favorisée est égocentrique.

Afin de tester les hypothèses ci-dessus, nous avons conçu quatre scènes virtuelles (figure 5.8) contenant chacune des indices contextuels différents :

— **Scène 1** : un objet générique [15] sur fond noir, destiné à servir de scène « neutre » sans indice contextuel particulier ;
— **Scène 2** : un objet étant susceptible d'être perçu comme fixe (une maison), également sur fond noir ;
— **Scène 3** : la même maison qu'en scène 2, mais vue de l'intérieur ;
— **Scène 4** : le même objet qu'en scène 1, mais entouré d'un environnement fixe.

Les travaux précédents sur ce sujet (tels que [Kaminaka et Egli 1985]) étudiaient séparément translations et rotations. Cependant, l'utilisation des mouvements d'un appareil mobile

15. En l'occurrence le « Stanford Bunny », un modèle standard souvent utilisé comme objet de démonstration dans l'infographie 3D.

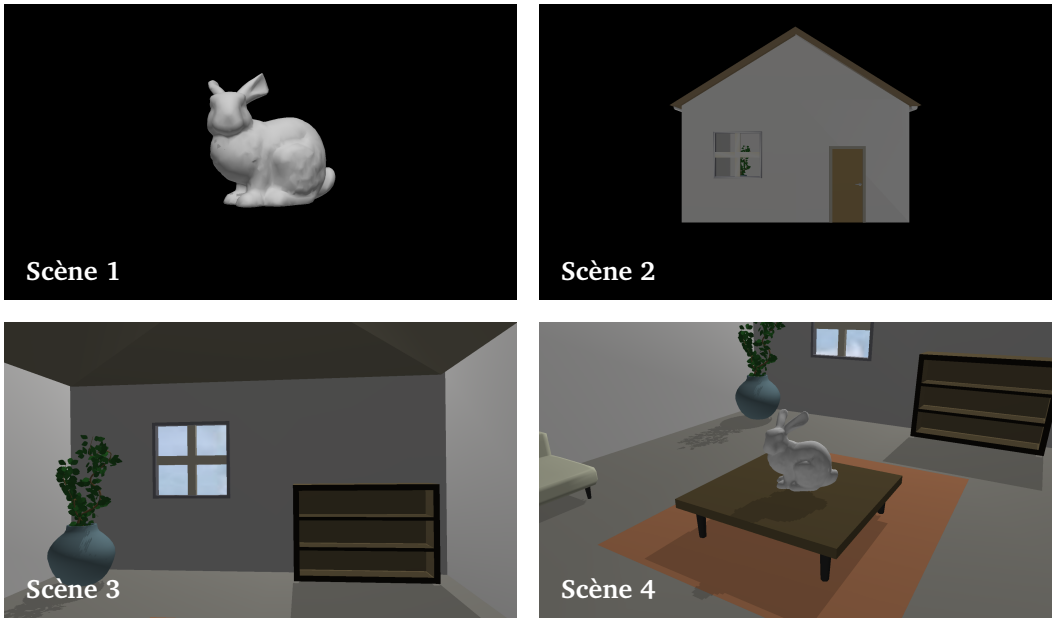


FIGURE 5.8 — Les quatre scènes virtuelles utilisées dans cette expérience. Les participants doivent atteindre le résultat ci-dessus en se servant des différents mappings à évaluer : tantôt allocentriques, tantôt égocentriques. Dans la scène 3, le modèle de maison est manipulé comme en scène 2 mais le point de vue est situé à l'intérieur. Dans la scène 4, seul l'objet sur la table est contrôlé par l'utilisateur tandis que le reste de l'environnement demeure fixe.

comme modalité d'interaction autorise une manipulation pleinement 6 DdL. Il est donc important de s'intéresser aussi aux translations et rotations effectuées *simultanément*. Afin d'étudier le rôle de chaque composante dans un contexte de manipulation 6 DdL, nous formulons l'hypothèse nulle suivante :

— **H4** : Choisir le « bon » mapping (correspondant à ce qu'attend l'utilisateur) est tout aussi important dans le cas des translations que des rotations. En d'autres termes, lorsque l'interface permet d'effectuer des translations et des rotations simultanément, l'appréciation des utilisateurs est autant affectée par le choix du mapping en translation que par le choix du mapping en rotation.

Plateforme expérimentale

L'expérience est entièrement contenue dans un unique appareil mobile servant à la fois à la manipulation, à l'affichage et au traitement, conformément à l'approche intégrée qui est l'objectif de ce chapitre. La scène virtuelle est affichée sur l'écran de l'appareil, ainsi que plusieurs boutons tactiles permettant de contrôler le déroulement de l'expérience. L'appareil mobile utilisé est une tablette Tango « Yellowstone » : un modèle 7 pouces équipé d'un

ensemble de capteurs lui permettant de suivre ses propres mouvements par rapport à l'environnement (section 5.3.1). Ces mouvements sont exprimés dans un repère fixe défini par la position et l'orientation de la tablette lors de sa mise sous tension.

À partir de ces informations de suivi, nous implémentons un mapping de type « contrôle en position relatif » tel qu'il est décrit dans la figure 5.5.3. Ce mapping a été choisi pour garantir la compatibilité directionnelle, afin que les participants puissent se concentrer sur le *sens* de mouvement (allocentrique ou égocentrique) sans qu'il y ait de confusion possible avec la *direction* de mouvement (qui correspond toujours à celle de l'appareil mobile dans le cas de la compatibilité directionnelle). Un mécanisme de « débrayage » est également présent : les mouvements de l'appareil mobile ne sont appliqués à l'objet manipulé que lorsqu'un doigt est pressé sur l'écran. Les utilisateurs peuvent ainsi interrompre la manipulation à tout moment pour repositionner la tablette si nécessaire.

Pendant toute l'expérience, les participants sont assis sur une chaise et tiennent la tablette en position horizontale (paysage) à deux mains. La chaise se trouve au centre de la pièce, pour éviter que la présence d'objets proches, tels qu'une table, ne crée un biais en faveur d'une interprétation égocentrique. La chaise n'est pas pivotante, pour inciter les participants à tourner l'appareil mobile *sur lui-même* durant les tâches de rotation.

Participants

Un groupe de 30 participants (18 hommes, 12 femmes) a été recruté pour cette expérience. Leur âge s'échelonnait de 20 ans à 53 ans (moyenne=30,3, σ=10,5). Parmi eux, 22 disposaient d'un diplôme universitaire et 8 d'un diplôme de baccalauréat ou inférieur. Tous avaient une vue normale ou corrigée.

Procédure et tâche

Avant de débuter l'expérience, nous commençons par présenter la tablette au participant et lui annonçons qu'il devra « effectuer des translations et des rotations dans plusieurs types d'environnements 3D ». Dans chacune de nos explications, nous veillons avec le plus grand soin à éviter des termes tels que « translater/tourner un objet » ou « déplacer la scène virtuelle » qui risqueraient de biaiser les résultats en faveur d'une interprétation allocentrique ou égocentrique. Aucune phase d'entraînement n'est prévue, afin d'éviter tout biais causé par une utilisation antérieure de l'interface. Pour remplacer l'entraînement, nous effectuons une démonstration en déplaçant et tournant la tablette devant le participant pour lui montrer en quoi consistent les mouvements qu'il faudra effectuer. Le principe du débrayage est lui aussi

présenté, en appuyant et relâchant un doigt sur l'écran. L'écran de la tablette reste éteint pendant ce tutoriel pour ne pas exposer les participants à un mapping particulier.

Les participants sont ensuite invités à commencer l'expérience. Celle-ci consiste en une série de *conditions*. Dans chaque condition, deux mappings ou plus doivent être évalués dans une scène virtuelle donnée. Sur l'écran de la tablette se trouvent plusieurs boutons tactiles intitulés C1, C2, etc. qui représentent les mappings à évaluer. Ceux-ci sont assignés au hasard à chaque bouton d'une condition à l'autre, de sorte que les participants ne peuvent pas reconnaître un mapping à partir du seul intitulé du bouton qui lui est associé. L'appui sur un bouton active le mapping correspondant, permettant aux participants de passer d'un mapping à l'autre pour les évaluer.

Pour aider les participants à se forger une opinion, nous leur demandons d'accomplir des tâches de manipulation 3D en utilisant chaque mapping. Ces tâches consistent à translater/tourner un objet 3D jusqu'à un emplacement cible, c'est-à-dire une tâche d'amarrage (*docking*). Habituellement, dans ce type de tâche, la cible est représentée visuellement au sein de la scène virtuelle. Cependant, nous ne pouvions pas afficher une telle représentation dans la scène car la présence d'un objet fixe aurait mené à la situation décrite dans l'hypothèse **H2**. De plus, nous ne pouvions pas non plus demander aux participants « d'amener l'objet manipulé vers la cible » car cela aurait produit un biais en faveur de l'interprétation allocentrique. Nous avons donc choisi une approche légèrement différente du *docking* classique. Sur une feuille de papier, nous imprimons une illustration de l'objectif à atteindre pour chaque scène (figure 5.8). Cette feuille est fixée sur un mur en face du participant pour qu'il puisse s'y référer à tout moment. Au début de chaque essai, l'objet manipulé est placé dans une position/orientation différente. Nous demandons au participant de « tenter d'obtenir le même résultat » que sur les illustrations « en effectuant des translations et des rotations », évitant ainsi toute formulation qui pourrait favoriser une interprétation allocentrique ou égocentrique. Lorsque cet emplacement cible est atteint, l'objet manipulé change de couleur sur l'écran pour indiquer que la tâche est réussie. L'objet est ensuite placé dans une nouvelle position/orientation dès que le participant relâche son doigt de l'écran. Nous invitons chaque participant à répéter cette tâche suffisamment de fois pour se faire une opinion précise avant d'attribuer une note à un mapping.

Pour attribuer une note, le participant appuie sur un bouton tactile sur l'écran de la tablette et voit s'afficher une échelle de Likert allant de « pas du tout naturel » à « très naturel ». La signification du terme « naturel » a été préalablement décrite au participant de la façon suivante : « à quel point vos actions produisent une translation ou une rotation dans la direction à laquelle vous vous attendiez ». Là encore, cette définition est conçue pour éviter toute interprétation allocentrique ou égocentrique.

Le système de suivi de la tablette Tango présentant parfois une légère dérive, cela risquait d'amener les participants à réduire leurs notes compte tenu de la définition ci-dessus. Nous

précisons donc aux participants que « si un léger mouvement se produit sans action de votre part, il s'agit d'une limitation technique que vous ne devez pas prendre en compte dans la notation ». Lorsqu'une note a été attribuée à tous les mappings d'une condition donnée, l'expérience passe automatiquement à la condition suivante.

Partie 1 : translation et rotation — Dans la première partie de l'expérience, les translations et les rotations sont évaluées séparément. Les conditions consistent donc en des tâches de translation seule ou de rotation seule dans chacune des quatre scènes, et avec deux mappings à évaluer : allocentrique et égocentrique.

La scène 1, conçue pour être un environnement « neutre », est présentée en premier pour servir de référence lors du premier contact des participants avec l'interface. Les deux premières conditions de l'expérience sont donc des tâches de translation dans la scène 1 et des tâches de rotation dans la scène 1, présentées dans un ordre qui alterne entre chaque participant. Les conditions suivantes sont les 6 combinaisons de translation ou de rotation avec chacune des trois scènes restantes, ces conditions étant elles-mêmes présentées dans un ordre aléatoire.

Les notes sont attribuées sur une échelle de Likert à 4 points. Cette échelle a été choisie spécifiquement car elle ne contient pas de point neutre, ce qui incite les participants à décider s'ils considèrent un mapping naturel ou non.

Partie 2 : translations/rotations simultanées — Dans la deuxième partie de l'expérience, les translations et les rotations sont effectuées simultanément (c'est-à-dire une manipulation 6 DdL), mais chaque composante (translation ou rotation) est alternativement allocentrique ou égocentrique. Il y a donc 4 mappings à évaluer : les quatre combinaisons de translations allocentriques ou égocentriques avec des rotations allocentriques ou égocentriques. Ces mappings sont eux-mêmes évalués dans chaque scène virtuelle, pour un total de 4 conditions présentées dans un ordre aléatoire. L'échelle de Likert est là encore sur 4 points, afin de pouvoir comparer les résultats avec ceux de la première partie.

5.7.3 Résultats et discussion

Dans cette analyse, nous nous intéressons principalement à la *taille d'effet* (c'est-à-dire à quel point les scores attribués à un mapping diffèrent de ceux attribués à un autre mapping) pour déterminer quel mapping est préféré aux autres dans chaque condition. Les valeurs obtenues à partir d'une échelle de Likert étant ordinales, nous utilisons des tests statistiques non-paramétriques pour quantifier les tailles d'effet. Dans chaque condition, nous appliquons

un test de type *Wilcoxon-Pratt signed-rank* aux scores attribués à chaque mapping proposé. Nous calculons ensuite une taille d'effet normalisée r à partir de la statistique z générée par le test précédent, d'après la méthode [16] décrite par Fritz *et al.* [2012].

On considère qu'une taille d'effet r>0,5 représente un effet important, r>0,3 un effet moyen, et r>0,1 un effet léger [Fritz *et al.* 2012]. Cependant, ces seuils ne doivent pas être vus comme une classification stricte : une taille d'effet de 0,49 est ainsi plus proche d'un effet important que d'un effet moyen. Pour évaluer la fiabilité statistique des résultats, nous indiquons également l'erreur type σ_r pour chaque taille d'effet, obtenue par un processus d'amorçage (*bootstrapping*).

Translations

La figure 5.9 présente les scores attribués par les participants à chaque mapping dans les tâches de translation de la partie 1.

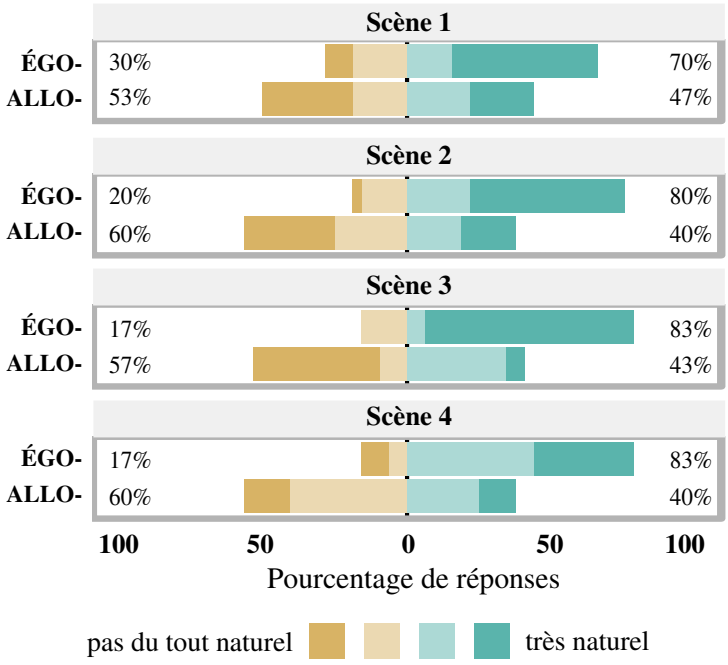


FIGURE 5.9 — Scores attribués à chaque mapping dans les tâches de translation seule.

La scène 1 apparaissait toujours en premier afin de minimiser d'éventuels effets d'apprentissage causés par l'utilisation de l'interface, et était prévue comme un environnement neutre ne présentant pas les indices contextuels des autres scènes. Par conséquent, les résultats

16. $r = z/\sqrt{N}$, où N est le nombre total de réponses.

obtenus dans cette scène peuvent être considérés comme se rapprochant le plus du modèle mental des participants. Le protocole de l'expérience ayant été conçu pour éliminer tout biais en faveur de l'une ou l'autre des interprétations (allocentrique ou égocentrique), nous n'avions pas de raison *a priori* de nous attendre à ce qu'un mapping soit préféré à l'autre.

Nous observons pourtant un effet léger voire moyen (r=0,27; σ_r=0,11) en faveur des translations égocentriques dans la scène 1. La répartition des réponses laisse penser que le mapping égocentrique a effectivement été trouvé naturel par la majorité des participants, tandis que le mapping allocentrique a fait l'objet d'opinions plus mitigées. Le mapping égocentrique a donc clairement été préféré lors de ce premier contact avec l'interface, même si le mapping allocentrique n'a pas non plus été entièrement rejeté.

La scène 2 présente une tendance similaire à la scène 1 : les translations égocentriques ont été préférées aux translations allocentriques. L'effet semble même plus important (r=0,41; σ_r=0,09), bien que l'erreur type dans les deux conditions rende cette distinction incertaine. Cette deuxième scène était particulièrement conçue pour tester l'hypothèse **H3** qu'un objet typiquement fixe favoriserait un mapping égocentrique. La préférence plus marquée pour les translations égocentriques dans cette scène pourrait donc appuyer cette hypothèse. Mais même si une telle différence d'effet avec la scène 1 était avérée, celle-ci paraît malgré tout assez limitée. Par ailleurs, étant donné que la préférence « de base » pour les translations semble être égocentrique (comme on a pu le voir dans la scène 1), un éventuel effet qui renforcerait l'interprétation égocentrique n'aurait que peu d'intérêt en pratique.

Dans la scène 3, la préférence était très clairement en faveur du mapping égocentrique (r=0,53; σ_r=0,05), plus encore que dans les scènes 1 et 2. Ce résultat accrédite donc fortement l'hypothèse **H1** qu'un objet vu de l'intérieur favorise une interprétation égocentrique.

La scène 4 présente elle aussi une préférence pour le mapping égocentrique (r=0,30; σ_r=0,12). Ce résultat est surprenant, car nous nous attendions à ce que la manipulation d'un objet par rapport à un environnement fixe favorise l'interprétation allocentrique (hypothèse **H2**). Ayant remarqué au cours de l'expérience que certains participants attribuaient des scores élevés au mapping égocentrique, nous avons saisi cette opportunité pour leur demander d'expliquer leur choix à la fin de la première partie. Leurs réponses laissent penser qu'ils étaient surtout concentrés sur la tâche et qu'ils ne prêtaient pas grande attention à la scène virtuelle elle-même. Étant donné que les participants avaient déjà rencontré deux conditions ou plus (au minimum les deux conditions de la scène 1) avant d'atteindre la scène 4, il se peut en effet que la tâche ait commencé à devenir une « routine ». D'un autre côté, le net contraste entre les résultats de la scène 3 et ceux des scènes 1 et 2 montre que les participants pouvaient encore être influencés par les indices contextuels à ce stade de l'expérience. Une autre explication possible est que les participants auraient simplement considéré l'environnement virtuel fixe de la scène 4 comme une image d'arrière-plan, et n'auraient donc pas adopté le modèle mental que nous attendions.

Rotations

La figure 5.10 présente les scores attribués par les participants à chaque mapping dans les tâches de rotation de la partie 1. La condition « de référence » dans la scène 1 révèle

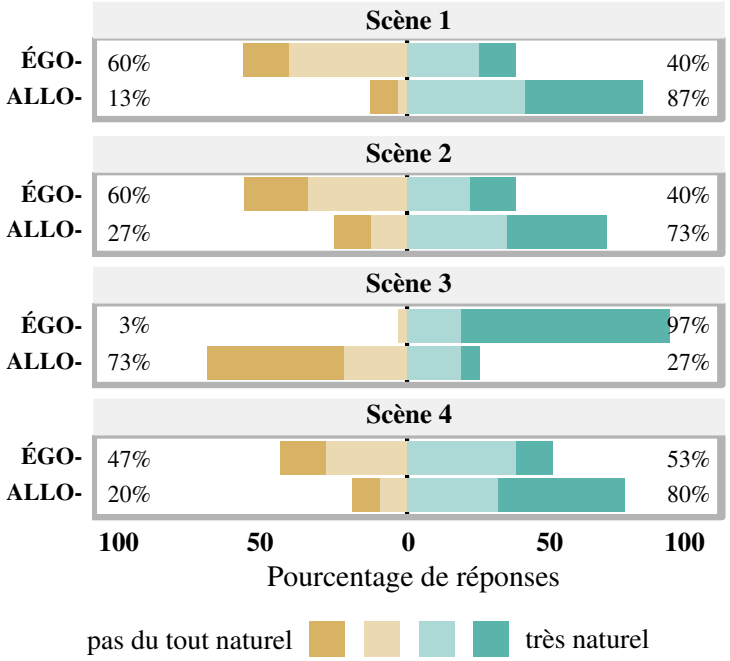
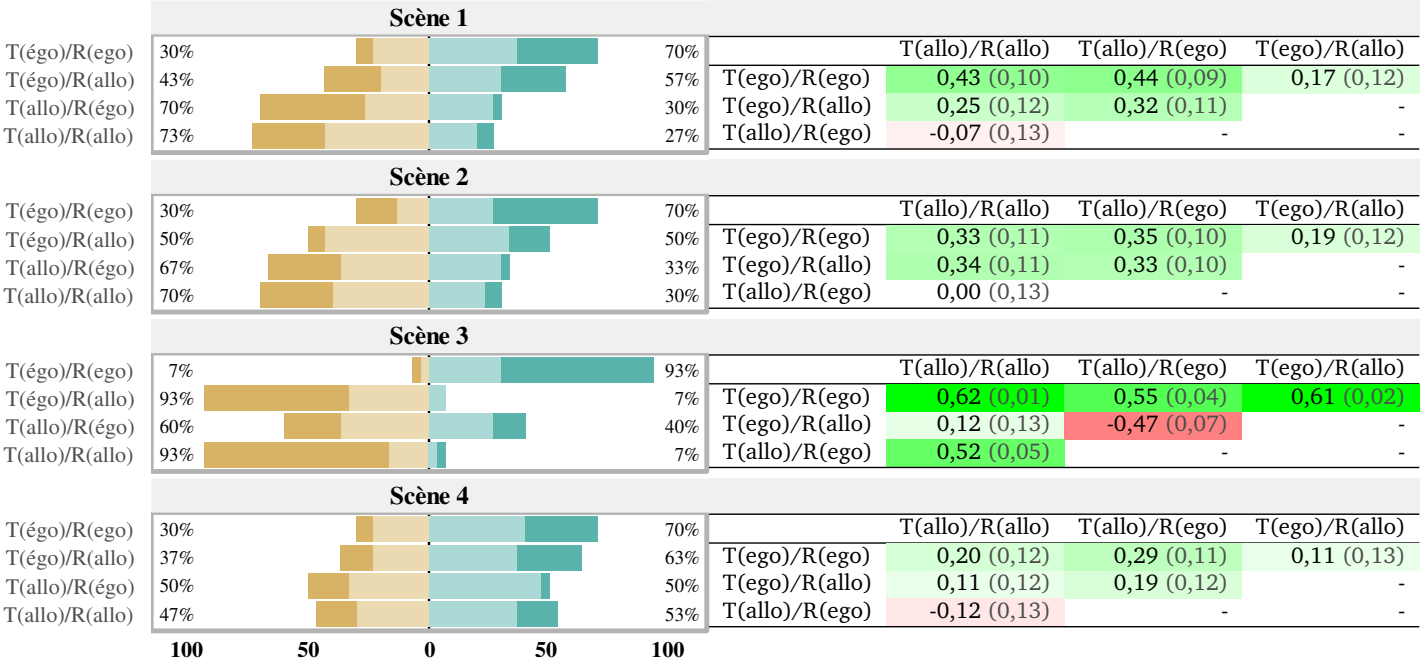


FIGURE 5.10 — Scores attribués à chaque mapping dans les tâches de rotation seule.

un effet moyen (r=0,37; σ_r=0,10) en faveur des rotations allocentriques. La majorité des participants ont considéré les rotations allocentriques comme naturelles, avec des opinions plus mitigées sur les rotations égocentriques.

Le mapping allocentrique a également été préféré dans la scène 2, mais de façon apparemment moins marquée (r=0,24; σ_r=0,12) que dans la scène 1. Bien que cette distinction reste incertaine en raison de l'erreur type, si elle était avérée cela appuierait l'hypothèse **H3** qu'un objet typiquement fixe favorise un mapping égocentrique. Comme pour les translations, l'effet semble assez faible. Mais contrairement aux translations, il tendrait à amener les utilisateurs vers le mapping *opposé* à celui « de base » identifié dans la scène 1. Cependant, cet effet, s'il existe, est apparemment trop faible pour renverser la préférence globale pour les rotations allocentriques.

À l'inverse des autres scènes, la scène 3 révèle une très nette préférence pour les rotations égocentriques (r=0,57; σ_r=0,04). À nouveau, il s'agit là d'un résultat qui conforte l'hypothèse **H1** qu'un objet vu de l'intérieur favorise une interprétation égocentrique.



Scène 1	T(allo)/R(allo)	T(allo)/R(ego)	T(ego)/R(allo)
T(ego)/R(ego)	0,43 (0,10)	0,44 (0,09)	0,17 (0,12)
T(ego)/R(allo)	0,25 (0,12)	0,32 (0,11)	-
T(allo)/R(ego)	-0,07 (0,13)	-	-

Scène 2	T(allo)/R(allo)	T(allo)/R(ego)	T(ego)/R(allo)
T(ego)/R(ego)	0,33 (0,11)	0,35 (0,10)	0,19 (0,12)
T(ego)/R(allo)	0,34 (0,11)	0,33 (0,10)	-
T(allo)/R(ego)	0,00 (0,13)	-	-

Scène 3	T(allo)/R(allo)	T(allo)/R(ego)	T(ego)/R(allo)
T(ego)/R(ego)	0,62 (0,01)	0,55 (0,04)	0,61 (0,02)
T(ego)/R(allo)	0,12 (0,13)	-0,47 (0,07)	-
T(allo)/R(ego)	0,52 (0,05)	-	-

Scène 4	T(allo)/R(allo)	T(allo)/R(ego)	T(ego)/R(allo)
T(ego)/R(ego)	0,20 (0,12)	0,29 (0,11)	0,11 (0,13)
T(ego)/R(allo)	0,11 (0,12)	0,19 (0,12)	-
T(allo)/R(ego)	-0,12 (0,13)	-	-

FIGURE 5.11 — Scores (et tailles d'effet) attribués par les participants dans les tâches de manipulation 6 DdL (translations et rotations simultanées). La notation T(tmap)/R(rmap) désigne un mapping « tmap » en translation associé à un mapping « rmap » en rotation.

La scène 4 montre une tendance similaire aux scènes 1 et 2, c'est-à-dire une préférence pour les rotations allocentriques (r=0,28; σ_r=0,11). Cependant, les rotations égocentriques ont malgré tout été considérées comme naturelles par un peu plus de la moitié des participants. Puisque le mapping préféré dans la scène 1 était déjà allocentrique, ces résultats ne permettent donc pas d'appuyer l'hypothèse **H2**. La taille d'effet plus faible que dans la scène 1 pourrait même aller à l'encontre de **H2**, puisque le mapping égocentrique a été plus facilement accepté dans cette configuration.

Translations/rotations simultanées

La figure 5.11 présente les scores que les participants ont attribués à chaque combinaison de translations et de rotations allocentriques et égocentriques, pour les tâches de manipulation 6 DdL. Dans un but de concision, nous utiliserons la notation T(tmap)/R(rmap) pour décrire la combinaison d'un mapping « tmap » en translation et « rmap » en rotation.

Les résultats pour les scènes 1 et 2 vont tous deux dans le même sens : les différences de score sont comparativement plus importantes (taille d'effet moyenne) entre les combinaisons T(égo)/* et T(allo)/* qu'entre les combinaisons */R(égo) et */R(allo) (taille d'effet faible voire nulle). Il semble donc que, dans ces deux scènes, le mapping en translation soit

plus important que le mapping en rotation dans le ressenti des utilisateurs lorsque les deux sont effectués simultanément. Ce résultat réfute donc l'hypothèse **H4**.

Comme nous pouvions nous y attendre compte tenu des résultats précédents en translation seule, les combinaisons T(égo)/* ont été préférées aux combinaisons T(allo)/* dans toutes les scènes, à l'exception de la scène 3. Dans ces trois scènes, cependant, la combinaison T(égo)/R(égo) a été préférée à la combinaison T(égo)/R(allo). Ce résultat est inattendu car les résultats de la partie 1 montraient une préférence pour les rotations allocentriques dans ces mêmes scènes. Par ailleurs, le mapping T(allo)/R(allo) semble également avoir été préféré au mapping T(allo)/R(égo) dans les scènes 1 et 4, avec une égalité dans la scène 2. Bien que ces différences restent inférieures à l'erreur type, elles laissent cependant penser qu'une même tendance pourrait exister dans ces trois scènes. En plus de la dominance du mapping en translation, il pourrait donc y avoir une préférence pour un mapping identique en translation et en rotation.

Plus globalement, les quatre combinaisons ont chacune obtenu des scores comparables entre les scènes 1 et 2. Les différences entre les combinaisons restent donc elles aussi comparables. Cependant, les tailles d'effet semblent plus uniformes dans la scène 2. Cela pourrait s'expliquer par le plus petit nombre de scores fortement négatifs pour les mappings T(égo)/R(allo) et T(allo)/R(égo) : deux combinaisons qui contiennent un mapping égocentrique. Cela est cohérent avec le léger effet observé précédemment qui tendrait à renforcer la préférence pour le mapping égocentrique dans la scène 2. Il s'agit également d'un nouvel indice, bien que limité, en faveur de l'hypothèse **H3**.

La scène 3 présente une forte différence entre le mapping pleinement égocentrique (T(égo)/R(égo)) et les autre combinaisons. Presque tous les participants ont considéré le mapping pleinement égocentrique comme naturel, alors que les trois autres combinaisons ont été largement considérées non-naturelles. Pourtant, parmi ces trois dernières combinaisons, T(allo)/R(égo) (la seule à présenter des rotations égocentriques) a été comparativement mieux notée, bien qu'elle soit constituée de deux mappings opposés. Nous pouvons supposer que cela est dû à la plus grande influence des rotations sur le *flux optique* lorsque l'objet manipulé est vu de l'intérieur. Dans cette situation, l'effet positif d'avoir un mapping « compatible » en rotation (égocentrique, d'après les résultats de la partie 1) semble pallier à l'effet négatif d'un mapping non-compatible (allocentrique) en translation, en dépit du fait que les mappings sont différents en translation et en rotation. Dans tous les cas, ces résultats confirment à nouveau l'importance d'avoir un mapping égocentrique lorsque l'objet manipulé est vu de l'intérieur (hypothèse **H1**).

Résumé des résultats

La préférence « de base », c'est-à-dire l'alternative considérée comme la plus naturelle sans expérience préalable avec l'interface et sans indices contextuels spécifiques, est un mapping *égocentrique pour les translations* et un mapping *allocentrique pour les rotations*.

En ce qui concerne les indices contextuels, l'hypothèse **H1** est largement appuyée par nos résultats : lorsqu'un objet est vu de l'intérieur, le mapping devrait être égocentrique en translation et en rotation. De façon surprenante, l'hypothèse **H2** n'a pas été validée par les résultats, et pourrait même être réfutée. Il semble donc que manipuler un objet au sein d'un environnement virtuel fixe ne crée pas de préférence pour un mapping allocentrique. Un certain support est apporté à l'hypothèse **H3**, selon laquelle un objet typiquement fixe créerait une préférence pour un mapping égocentrique. Les résultats laissent entrevoir un possible effet, mais celui-ci est de toute évidence faible et insuffisant pour peser face à d'autres facteurs (tels que la préférence de base).

Nous recommandons donc que les tâches de translation soient effectuées avec un mapping égocentrique dans tous les cas, et que les tâches de rotation soient effectuées avec un mapping allocentrique dans tous les cas *sauf* lorsque l'objet manipulé est vu de l'intérieur, auquel cas les rotations devraient être égocentriques.

Dans le cadre d'une manipulation 6 DdL, où les utilisateurs effectuent des translations et des rotations simultanément, le choix d'un « bon » mapping en translation (égocentrique, d'après les résultats de la partie 1) apparaît être plus important que le choix du mapping en rotation, ce qui réfute l'hypothèse **H4**. L'exception est là encore lorsque l'objet manipulé est vu de l'intérieur, auquel cas il est important que les *deux* mappings soient égocentriques. Par conséquent, nous recommandons d'utiliser un mapping 6 DdL de type T(égo)/R(égo) (c'est-à-dire pleinement égocentrique) dans tous les cas.

Dans l'annexe B, nous présentons un certain nombre de résultats supplémentaires constituant un complément à cette expérience.

5.8 Espace de manipulation

Le fait d'utiliser les mouvements d'un appareil mobile comme modalité d'interaction présente un autre défi important : celui de l'espace de manipulation. Selon les applications, l'espace virtuel peut être potentiellement très vaste. L'utilisateur peut donc avoir à déplacer un objet 3D sur une distance arbitrairement grande. D'autres situations peuvent quant à elles nécessiter des rotations arbitrairement larges, par exemple pour simuler une action de vissage dans une tâche d'assemblage.

Cependant, l'utilisateur n'est pas forcément en mesure d'appliquer des translations ou des rotations arbitrairement importantes à l'appareil mobile. D'une part, celui-ci est tenu dans la main et est donc soumis à des contraintes anatomiques et ergonomiques. D'autre part, dans cette configuration localement couplée, l'écran de l'appareil se déplace et tourne également durant la manipulation. En fonction des caractéristiques de l'appareil mobile, l'écran peut devenir invisible ou illisible dans certaines positions et orientations (notamment lorsque celui-ci n'est plus orienté vers l'utilisateur), interrompant ainsi le retour visuel. Cela restreint encore davantage l'espace utilisable pour l'interaction.

Nous appellerons *espace de manipulation* cet espace réellement utilisable (tant en translation qu'en rotation), défini comme l'intersection entre l'espace accessible en mouvement par l'utilisateur et celui dans lequel l'appareil mobile peut fournir un retour visuel.

Parce que cet espace de manipulation est fondamentalement limité, il est essentiel de proposer des solutions pour l'exploiter au mieux et éviter autant que possible que l'utilisateur en atteigne les frontières durant la manipulation. Dans cette section, nous examinons tout d'abord les différents facteurs qui déterminent l'étendue réelle de l'espace de manipulation. Nous proposons ensuite plusieurs solutions pour surmonter cette portée limitée, en examinant les avantages et inconvénients de chaque alternative.

5.8.1 Facteurs influençant l'espace de manipulation

Contraintes anatomiques

Puisque l'utilisateur tient l'appareil mobile pendant la manipulation, les mouvements possibles dépendent en premier lieu de l'espace accessible par les mains. Les translations sont limitées par la longueur des bras [17], tandis que les rotations sont affectées par les articulations du poignet [Rahman *et al.* 2009]. Cependant, l'étendue exacte de ces contraintes dépend de la manière dont l'utilisateur tient l'appareil mobile. D'une part, celui-ci peut être tenu dans une ou deux mains. D'autre part, deux formes de préhension peuvent être identifiées [Fitzmaurice 1996] : la « préhension en force » (*power grip*) qui consiste à tenir l'appareil mobile fermement dans la (les) main(s), et la « préhension précise » (*precision grip*) qui consiste à tenir l'appareil avec les doigts.

Dans une situation de « préhension en force », les mouvements de l'appareil sont plus contraints. Lorsque celui-ci est tenu à une main, les translations et les rotations sont limitées par les distances accessibles par la main et les orientations autorisées par le poignet.

17. Sauf si l'utilisateur lui-même se déplace par rapport au système de suivi (par exemple en marchant tout en tenant l'appareil mobile), une situation que nous choisissons de ne pas considérer ici.

Avec deux mains, les mouvements de l'appareil mobile deviennent sujets aux limitations des deux mains et poignets en même temps, même si certaines rotations plus importantes peuvent être réalisées en se servant des deux mains comme pivots. Dans le cas de la « préhension précise », les doigts peuvent compenser ces contraintes et offrir ainsi un plus grand espace de manipulation. En utilisant les doigts des deux mains, la liberté de manipulation augmente encore davantage.

Retour visuel

Sur un appareil mobile classique, la surface d'affichage ne couvre qu'une face de l'appareil. La seconde restriction affectant l'espace de manipulation est donc la visibilité de l'écran. Manipuler un objet virtuel nécessite un retour visuel ininterrompu afin de contrôler sa trajectoire. L'écran doit donc rester visible (et lisible) pendant toute la manipulation. Cela réduit l'espace de manipulation aux positions et orientations de l'appareil mobile qui maintiennent ce retour visuel. Étant donné que la plupart des appareils mobiles actuels ne disposent que d'un écran plat sur une seule face, l'intervalle de rotation possible correspond donc aux orientations où l'écran est tourné vers l'utilisateur [Rahman *et al.* 2009]. Les dernières technologies d'affichage (IPS, OLED...) offrent des angles de vision proches de 180°. Malgré cela, un unique écran sur un seul côté, quelle que soit sa technologie, devient forcément invisible lorsqu'il n'est plus tourné vers l'utilisateur.

Espace de manipulation *confortable*

Bien que l'espace de manipulation proprement dit puisse en théorie s'étendre jusqu'à la longueur du bras tendu et jusqu'aux limites des angles de vision, de telles positions et orientation ne sont pas forcément exploitables *en pratique* pour un usage courant. Tenir l'appareil mobile à bout de bras pendant des périodes longues ou répétées génère de la fatigue et réduit la précision. L'écran lui-même peut également être « visible », sans pour autant être « lisible ». Lorsque l'appareil mobile est déplacé loin des yeux, le contenu de l'écran devient plus petit ce qui réduit la lisibilité. Lorsque l'appareil mobile est tourné, la distorsion de perspective peut rendre le contenu de l'écran illisible, interrompant de fait le retour visuel. Par conséquent, nous pouvons faire une distinction supplémentaire entre *l'espace de manipulation total*, théoriquement exploitable par l'utilisateur, et *l'espace de manipulation confortable*, plus réduit, à l'intérieur duquel l'appareil mobile peut être manipulé confortablement. Un mapping approprié ne devrait pas obliger l'utilisateur à quitter l'espace de manipulation confortable. Cela dit, il reste toujours possible de tirer parti de l'espace additionnel pour des besoins ponctuels.

L'étendue précise des espaces de manipulation total et confortable peut être déterminée expérimentalement. Mais les résultats seraient nécessairement spécifiques à chaque modèle d'appareil mobile. Pour cette raison, nous faisons le choix dans cette section de rester dans un cadre générique. Une telle étude expérimentale pourrait être menée dans le futur, afin de déterminer l'étendue de l'espace de manipulation pour les appareils mobiles les plus répandus et ainsi mieux optimiser les mappings aux modèles actuels.

5.8.2 Surmonter les limitations de l'espace de manipulation

La limitation posée par l'espace de manipulation peut être surmontée de plusieurs manières. Dans cette section, nous présentons tout d'abord quelques-unes des solutions classiques, avec leurs avantages et inconvénients. Nous profitons ensuite du cadre formel introduit en section 5.5 pour proposer des solutions plus avancées, tirant parti des particularités de cette configuration localement couplée.

Contrôle en vitesse

Une manière de contourner presque entièrement ce problème est d'utiliser un mapping de type « contrôle en vitesse ». Dans un tel mapping, l'objet manipulé se déplace continuellement aussi longtemps que l'appareil mobile se trouve hors de son emplacement initial, avec une vitesse qui dépend de la distance à cet emplacement initial. Ainsi, avec un facteur de gain approprié et/ou suffisamment de temps, des translations et des rotations arbitrairement grandes peuvent être appliquées à l'objet manipulé tout en maintenant l'appareil mobile largement à l'intérieur de l'espace de manipulation.

La capacité à effectuer des mouvements « illimités » est sans aucun doute un avantage important du contrôle en vitesse. Cependant, ce type de mapping n'est pas sans inconvénients par rapport au contrôle en position. D'une part, ses propriétés de compatibilité spatiale sont moins désirables, comme nous l'avons vu dans la section 5.6. D'autre part, contrôler la vitesse de l'objet manipulé peut présenter certaines difficultés. Ainsi, changer la direction de déplacement de l'objet exige de ramener *d'abord* l'appareil mobile à son emplacement initial, *puis* de l'amener vers la nouvelle direction. Dans un contrôle en position, ce changement serait instantané. Terminer la manipulation à une position ou orientation précises est également difficile avec le contrôle en vitesse. Cela nécessite de ramener l'appareil mobile à son emplacement initial au moment précis de terminer la manipulation. Un appareil mobile étant portable et librement manipulable (ce qu'on appelle un dispositif *isotonique*), aucun mécanisme n'est présent pour aider l'utilisateur à le ramener à son emplacement initial [Zhai et Milgram 1998]. Le grand nombre de degrés de liberté dans l'espace 3D augmente d'autant la difficulté de cette tâche. Il existe des résultats expérimentaux allant dans ce sens :

Zhai [1998] a montré que le contrôle en vitesse était moins performant que le contrôle en position lorsqu'un dispositif isotonique était utilisé pour une tâche d'amarrage 3D (*docking*).

La difficulté de retrouver l'emplacement initial dans un espace libre peut être atténuée en créant une « zone nulle » (*deadband*) dans le mapping. Dans le formalisme établi en section 5.5, cela peut être réalisé en assignant la valeur 0 au facteur de gain lorsque la distance (ou l'angle) entre l'emplacement actuel de l'appareil mobile et son emplacement initial est en dessous d'un certain seuil. Nous introduisons pour cela une fonction « dist » qui calcule la distance ou l'angle entre deux emplacements de l'appareil mobile aux temps t_1 et t_2 (dans l'expression ci-dessous, $(q)_w$ représente la partie réelle d'un quaternion q) :

$$\mathrm{dist}(t_1,t_2) = \begin{cases} \|pc_{t_1} - pc_{t_2}\| & \text{(translations)} \\ 2\ \arccos\big((qc_{t_1}\ qc_{t_2}^{-1})_w\big) & \text{(rotations)} \end{cases} \tag{5.5}$$

La distance entre les emplacements actuel et initial de l'appareil mobile est ainsi dist(t,0). Une zone nulle de taille *seuil* peut être obtenue grâce à la fonction de gain suivante :

$$d = \mathrm{dist}(t,0)$$
$$\mathrm{gain}(t) = \begin{cases} 0 & \text{lorsque d} < \text{seuil} \\ (d - \text{seuil})\ /\ d & \text{sinon} \end{cases}$$

Cependant, une telle zone nulle occupe une portion précieuse de l'espace de manipulation, repoussant la partie active de la manipulation vers des zones potentiellement moins confortables. Par ailleurs, le changement de direction devient encore moins efficace.

Par conséquent, l'utilisation du contrôle en vitesse pour répondre au problème de l'espace de manipulation est un choix qui doit être fait avec précaution. Il convient de mettre en balance l'intérêt du mouvement illimité avec le risque d'une moindre précision et efficacité, un choix qui dépendra de l'usage envisagé pour l'interface et notamment de la fréquence des translations et rotations d'amplitude importante.

Débrayage

Le contrôle en position n'autorise pas de mouvement « illimité », mais il existe toutefois des solutions pour surmonter les limitations de l'espace de manipulation.

Le *débrayage* (*clutching*, [Zhai *et al.* 1996]) consiste à désactiver temporairement le mapping entre les mouvements de l'appareil mobile et ceux de l'objet manipulé, ramener l'appareil à un emplacement plus central dans l'espace de manipulation, puis reprendre la manipulation à partir de cet endroit. Cela permet d'appliquer des translations et des rotations arbitrairement

grandes à l'objet manipulé, en décomposant celles-ci en plusieurs étapes pouvant chacune entrer dans l'espace de manipulation.

Le débrayage nécessite un moyen d'enclencher et désenclencher explicitement le mapping, par exemple un bouton sur l'appareil mobile. Indépendamment de la question de l'espace de manipulation, un mécanisme de débrayage est de toute manière fortement conseillé pour que les utilisateurs puissent déplacer l'appareil mobile dans un autre but que la seule manipulation.

Cependant, le débrayage est déconseillé *durant* la manipulation. En interrompant le processus d'interaction, il ralentit la tâche de manipulation et réduit l'efficacité [Jellinek et Card 1990]. Plus l'objet virtuel doit être déplacé (ou tourné) loin, plus il faut recourir au débrayage et moins la manipulation est efficace. Le débrayage génère également des mouvements « inutiles » [Zhai et Milgram 1998]. Déplacer un appareil mobile dans l'espace 3D demandant déjà un certain effort, un débrayage fréquent peut donc rapidement conduire à de la fatigue.

En règle générale, le débrayage est indispensable dans le cas du contrôle en position pour rendre possible des translations ou des rotations exceptionnellement larges. Une utilisation occasionnelle du débrayage est normale dans un mapping de type contrôle en position. Mais un mapping efficace devrait être conçu pour éviter au maximum le recours au débrayage dans une utilisation courante.

Augmentation du facteur de gain

Il est possible de limiter le recours au débrayage en appliquant sur l'objet manipulé des déplacements *plus grands* que ceux de l'appareil mobile. Dans le formalisme introduit dans ce chapitre, cet effet peut être obtenu en choisissant un facteur de gain k_t (section 5.5.6) supérieur à 1. En augmentant le facteur de gain, le même mouvement de l'appareil mobile produit un mouvement plus important de l'objet manipulé. Plus ce facteur de gain est grand, moins l'utilisateur risque d'être limité par l'espace de manipulation durant la manipulation.

Cependant, le facteur de gain ne peut pas être infini. Le recours au débrayage ne peut donc pas être entièrement éliminé par cette seule méthode. Par ailleurs, la motricité humaine a elle-même une résolution finie [Bérard *et al.* 2011]. Il existe une limite au-delà de laquelle l'augmentation du facteur de gain dépasse les capacités de l'utilisateur à contrôler finement la manipulation, ce qui provoque une baisse de la précision. Le seuil précis où cela se produit dépend à la fois des propres capacités motrices de l'utilisateur et de la tâche de manipulation à effectuer. À l'inverse, réduire le facteur de gain à une valeur inférieure à 1 peut augmenter artificiellement la précision au-delà de la précision humaine (dans les limites du système de suivi), mais nécessite alors d'appliquer des mouvements plus importants sur l'appareil

mobile pour déplacer l'objet virtuel sur une même distance, augmentant de même le recours au débrayage.

L'augmentation du facteur de gain est donc une approche efficace pour éviter le débrayage dans des mappings de type « contrôle en position », mais mène à un compromis entre l'espace de manipulation et la précision. Pour éviter un tel compromis, il faut employer des stratégies plus avancées pour gérer le facteur de gain.

Facteur de gain dynamique

L'inconvénient principal d'un facteur de gain constant est que celui-ci s'applique à toutes les tâches de manipulation effectuées avec l'interface. Une meilleure approche serait d'adapter dynamiquement le facteur de gain au type de manipulation souhaitée : l'augmenter pour des déplacements importants ne nécessitant pas une grande précision, et le réduire pour une manipulation précise sur de courtes distances. Cependant, adapter ainsi le facteur de gain demanderait de connaître à l'avance le type de manipulation que l'utilisateur s'apprête à effectuer.

Une solution serait de laisser l'utilisateur contrôler explicitement le facteur de gain (un exemple de tel contrôle est décrit par Ware et Baxter [1989]). Mais cette solution exigerait une modalité d'entrée supplémentaire, étant donné que tous les degrés de liberté de l'appareil mobile sont déjà utilisés pour la manipulation. De plus, cela créerait une charge cognitive supplémentaire pendant l'interaction. Ware et Baxter [1989] évoquent effectivement la nécessité d'une période d'apprentissage spécifique avant de maîtriser cette fonction.

Une autre solution serait de mettre en place un contrôle *implicite* du facteur de gain, c'est-à-dire faire varier le gain en fonction du comportement de l'utilisateur. Nous proposons donc ici deux méthodes pour contrôler implicitement le facteur de gain dans des mappings de type « contrôle en position ».

Gain dépendant de la distance — Cette première méthode consiste à augmenter le gain lorsque l'appareil mobile s'éloigne de son emplacement initial, aussi bien en position qu'en orientation. Elle repose sur plusieurs constatations. Premièrement, les utilisateurs commencent généralement la manipulation à partir d'une position et une orientation confortables de l'appareil mobile. Si un utilisateur ne déplace ou tourne l'appareil que faiblement autour de cet emplacement initial, nous pouvons supposer que cela a pour but de manipuler précisément l'objet virtuel. Il est donc justifié de maintenir le facteur de gain faible dans cette zone. Cela n'augmente pas le risque de débrayage car l'appareil mobile se trouve encore largement dans l'espace de manipulation confortable. À l'inverse, si l'utilisateur déplace ou tourne l'appareil loin de son emplacement initial, nous pouvons supposer que son intention

est de déplacer ou tourner l'objet virtuel de façon importante. Il devient donc utile d'augmenter le facteur de gain dans cette situation, pour favoriser les mouvements importants tout en réduisant le risque d'atteindre les limites de l'espace de manipulation. Puisqu'il n'est pas possible de prédire quel mouvement l'utilisateur va effectuer, le facteur de gain effectue une transition en douceur entre les deux situations. Ce comportement est obtenu en faisant dépendre le facteur de gain de la distance depuis l'emplacement initial de l'appareil mobile. Nous réutilisons pour cela la fonction « dist » introduite plus haut (équation 5.5) :

$$\text{gain}(t) = a + b\ \text{dist}(t,0)^c$$

De précédents travaux ont proposé des méthodes similaires augmentant le gain avec la distance, telles que le mapping proposé par Poupyrev *et al.* [1999] dans le cas des rotations, ou la partie non-linéaire de la « Go-Go Interaction Technique » [Poupyrev *et al.* 1996] dans le cas des translations. Dans la fonction de gain formalisée ci-dessus, les termes a (facteur de gain minimum), b (facteur d'échelle) et c (exponentiation) peuvent être ajustés afin de recréer ces mappings.

Gain dépendant de la vitesse — Une approche différente, inspirée par les techniques « d'accélération de pointeur » utilisées avec les souris 2D, est de rendre le facteur de gain fonction de la *vitesse* de l'appareil mobile. En d'autres termes, plus l'appareil mobile est déplacé et tourné rapidement, plus vite (et donc plus loin) l'objet manipulé se déplace et tourne lui-même. Dans le monde réel, les tâches de manipulation précises doivent être effectuées lentement et précautionneusement, tandis que les déplacements plus larges peuvent être effectués plus rapidement. Nous pouvons donc nous baser sur cette métaphore pour adapter implicitement le facteur de gain aux besoins des utilisateurs durant la manipulation.

La vitesse de l'appareil mobile correspond à la distance ou à l'angle parcouru par l'appareil entre les temps $t-1$ et t, divisé par l'intervalle de temps Δt entre ces deux instants (dans notre formalisme, cet intervalle de temps est supposé constant). Il est important de noter que cette méthode se base sur les translations et rotations incrémentales de l'appareil mobile, et n'est donc applicable qu'aux mappings *relatifs*. Dans ces mappings, le gain dépendant de la vitesse peut être obtenu par la fonction suivante :

$$\text{gain}(t) = a + b\left(\frac{\text{dist}(t, t-1)}{\Delta t}\right)^c$$

Comme dans la méthode précédente, les termes a, b et c peuvent être utilisés pour ajuster la forme de la fonction, par exemple pour reproduire une des fonctions existantes d'accélération de pointeur.

5.9 Conclusion

Comme nous avons pu le voir dans ce chapitre, se servir de l'appareil mobile lui-même comme support de manipulation permet de répondre au principal inconvénient des approches par objets tangibles externes : celui d'être constituées de plusieurs éléments. Cette solution améliore donc encore la portabilité de l'interface, tout en conservant les avantages d'une manipulation non-intrusive et d'un matériel accessible. L'approche proposée ne requiert en effet qu'un appareil mobile de type Tango, une technologie que l'on peut d'ores et déjà trouver sur le marché et qui est amenée à se développer.

L'intégration du support de manipulation avec le dispositif d'affichage, une configuration dite « localement couplée », pose cependant de nouvelles questions. Ainsi, le fait de n'avoir plus qu'un support unique (l'appareil mobile) pour interagir avec un nombre quelconque d'objets virtuels nécessite désormais de sélectionner explicitement quel objet virtuel manipuler, et empêche d'en manipuler plusieurs simultanément (multiplexage temporel). Par ailleurs, le mapping entre le support de manipulation et l'objet virtuel n'est plus aussi direct que dans le cas des objets tangibles externes. Là où le seul mapping adapté avec un objet tangible séparé était de directement transférer sa position et orientation sur l'objet virtuel correspondant, la configuration étudiée ici se prête elle à d'autres interprétations. Il est ainsi possible de faire varier le sens de déplacement de l'objet, le facteur de gain de la manipulation, ou encore de contrôler sa vitesse plutôt que sa position.

Ces possibilités prennent tout leur sens en raison de la particularité du couplage local : l'écran, sur lequel sont affichés les objets virtuels, est confondu avec le support de manipulation. Cette composition de mouvements pose donc la question de la cohérence entre les mouvements des objets sur l'écran et ceux de l'appareil mobile. Comme nous l'avons montré dans notre analyse, chacun des principaux mappings répond à ce problème d'une façon différente. Une deuxième question est celle du référentiel de manipulation, allocentrique ou égocentrique, correspondant à deux sens de déplacement strictement opposés. Nous avons montré à travers une étude expérimentale que le mapping favorisé par les utilisateurs dépend en partie du type de tâche à effectuer (translation ou rotation), mais aussi du type d'objet manipulé sur l'écran. Enfin, une troisième question est celle de l'espace de manipulation. Comme nous l'avons vu, passer à un contrôle en vitesse ou faire varier le facteur de gain permet d'éviter ou de retarder la perte du retour visuel au cours d'une tâche de manipulation. Selon les choix effectués, un mapping donné répond donc plus ou moins bien à ces trois questions. Chaque mapping représente un compromis différent. Le gain en portabilité de cette approche, entièrement contenue dans l'appareil mobile, se fait donc au prix de certains compromis en terme de manipulation.

Dans ce chapitre, nous nous sommes attachés à répondre en détail aux questions ci-dessus, afin de pouvoir faire des choix pertinents lors de l'implémentation d'une interface basée sur

cette approche. Dans le prochain chapitre, nous abordons le problème initial sous un angle différent : comment modifier *l'appareil mobile* pour contourner les obstacles ci-dessus, de manière à retrouver un mapping aussi direct qu'avec les objets tangibles externes, mais cette fois dans un unique dispositif intégré ?

Chapitre 6

Le « volume tangible » : un dispositif portable et intégré pour une manipulation co-localisée

6.1 Introduction

Dans le chapitre précédent, nous avons étudié une approche d'interface consistant à se servir des déplacements d'un appareil mobile pour manipuler des objets 3D affichés sur son propre écran. Cette approche présente l'intérêt majeur de ramener le système entier à un seul élément (l'appareil mobile), ce qui permet d'améliorer sa portabilité tout en conservant une capacité de manipulation 6 DdL.

Cependant, cette configuration pose également des difficultés supplémentaires par rapport aux approches par objets tangibles décrites dans les chapitres 3 et 4. Une première difficulté est que se passer des objets tangibles externes n'élimine pas pour autant la *séparation* entre l'appareil mobile et les objets virtuels. Dans une projection 3D classique, perspective ou orthographique, les objets virtuels visibles sur l'écran se trouvent par définition en retrait de l'écran. Un objet se situant à la même position que l'écran serait partiellement ou totalement invisible (*clipping*) en raison de l'opération de projection, comme le montre la figure 6.1(c). Il existe donc une certaine distance entre le support de manipulation (l'appareil mobile avec son écran) et les objets virtuels que manipule l'utilisateur.

À cause de cette séparation, il n'est plus suffisant d'appliquer à l'identique les déplacements du support de manipulation sur un objet virtuel, comme nous pouvions le faire dans le cas des objets tangibles externes. Le principal obstacle se trouve au niveau des rotations. Si l'on suppose que l'objet manipulé est « fixé » au support de manipulation que représente l'appareil mobile, ainsi que le proposent par exemple Henrysson *et al.* [2005] et Marzo *et al.* [2014], alors il devient difficile de faire tourner efficacement l'objet sur lui-même (figure 6.1(a)). Si l'on cherche à appliquer séparément les translations et les rotations de l'appareil mobile sur

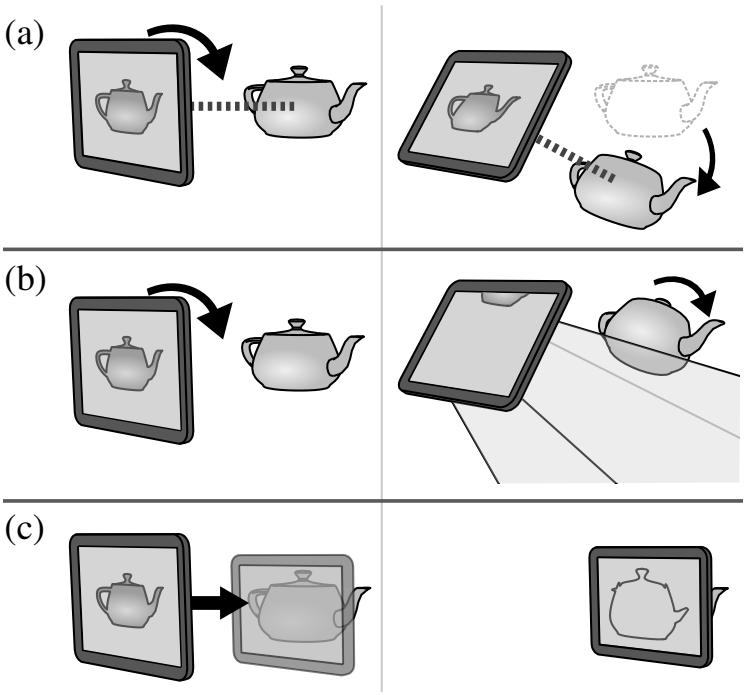


FIGURE 6.1 — Utiliser les déplacements d'un appareil mobile pour manipuler un objet 3D sur son propre écran présente un certain nombre de difficultés, principalement dues à la séparation entre l'appareil mobile et l'objet manipulé. **(a)** Si l'on choisit de « fixer » l'objet à l'appareil mobile, il devient difficile de faire tourner l'objet sur lui-même ; **(b)** Si l'on choisit de faire tourner l'objet sur lui-même, celui-ci risque alors de sortir de l'écran ; **(c)** Si l'on tente de réduire cette séparation en positionnant l'appareil mobile *sur* l'objet à manipuler, ce dernier disparaît partiellement ou totalement (*clipping*).

l'objet, ainsi que le proposent par exemple Mossel *et al.* [2013], celui-ci peut alors tourner sur lui-même mais risque également de sortir de l'écran durant la manipulation (figure 6.1(b)).

Pour contourner ce problème, nous avons choisi dans le chapitre précédent de fixer la scène virtuelle dans le repère de l'écran. Cependant, comme nous avons pu le voir, cette solution conduit à des mappings plus complexes qu'un simple transfert des mouvements sur l'objet. Cette approche crée par ailleurs une ambiguïté concernant le référentiel de manipulation, qui ne se posait pas dans les approches par objets tangibles externes.

Une deuxième difficulté liée à cette configuration est que la plupart des appareils mobiles actuels ont une forme plane, et ne disposent d'une surface d'affichage que sur un seul côté. Même si le problème de la séparation et de *clipping* était résolu, cette forme présenterait toujours un inconvénient : la perte du retour visuel lors de rotations importantes. Dans le chapitre précédent, nous avons proposé plusieurs manières de modifier le facteur de gain, ou encore l'utilisation d'un contrôle en vitesse, afin réduire l'éventualité d'une perte de

l'affichage. Mais cette solution rend la manipulation encore plus complexe et indirecte par rapport aux interfaces par objets tangibles externes.

Serait-il donc possible de retrouver la simplicité de manipulation des interfaces basées sur des objets tangibles, tout en conservant l'avantage en terme de portabilité d'un système entièrement contenu dans un unique dispositif ?

Dans ce chapitre, nous présentons une solution pour aboutir à ce résultat. Notre solution repose sur un nouveau type d'appareil mobile, le « volume tangible », conçu pour être directement positionné sur les objets virtuels à manipuler. Cette *co-localisation* élimine ainsi les problèmes de séparation identifiés ci-dessus. Le volume tangible fournit de plus un retour visuel sur la totalité de sa surface, ce qui permet de maintenir le retour visuel durant toute la manipulation.

6.2 Travaux existants

6.2.1 Fish-Tank Virtual Reality

Pour éliminer le problème de séparation entre l'écran et les objets virtuels, il faudrait pouvoir positionner l'appareil mobile directement *sur* l'objet à manipuler. Autrement dit, il faudrait que l'interface permette une co-localisation [Teather *et al.* 2009] entre le support de manipulation et les objets virtuels. Cependant, dans le cas d'une projection 3D classique de type perspective ou orthographique, l'objet en question serait coupé par le plan de projection (figure 6.1(c)). L'objet deviendrait alors partiellement ou totalement invisible (*clipping*), ce qui le rendrait difficile à manipuler convenablement. Afin de prévenir cette situation, il est donc nécessaire d'employer une autre forme de projection.

Le concept de *Fish-Tank Virtual Reality* (réalité virtuelle de type « aquarium »), ou « FTVR », apporte une solution au problème de clipping. Là où une projection classique confond l'écran avec le point de vue sur la scène virtuelle, la FTVR considère au contraire l'écran comme un simple « cadre » observé depuis le point de vue de l'utilisateur [Deering 1992; Ware *et al.* 1993]. L'objectif de départ de la FTVR est de produire l'illusion que les objets virtuels se trouvent « à l'intérieur » de l'écran, d'où le terme « d'aquarium ». Mais le fait de lier le point de vue à la position de l'utilisateur, plutôt qu'à la position de l'écran, permet surtout d'éviter que ces derniers ne soient coupés par le plan de l'écran. Il devient alors possible de faire apparaître des objets virtuels derrière, devant, ou sur le même plan que l'écran [Francone et Nigay 2011].

FIGURE 6.2 — Illustration d'un rendu *fish-tank* sur appareil mobile, basé sur le changement de parallaxe [Francone et Nigay 2011].

Il existe plusieurs manières d'arriver à cet effet. La première est d'utiliser un rendu stéréoscopique, comme cela est fait par exemple au cinéma. Une autre solution est de simuler le changement de parallaxe généré par les mouvements de la tête (*head-coupled perspective*), comme cela a été montré notamment par Lee [2008] et Yim *et al.* [2008]. Il est également possible d'employer une combinaison des deux. Selon Ware *et al.* [1993], qui ont étudié de telles combinaisons, la parallaxe joue un rôle plus important que le rendu stéréoscopique dans la perception spatiale, faisant donc de celle-ci une option à privilégier.

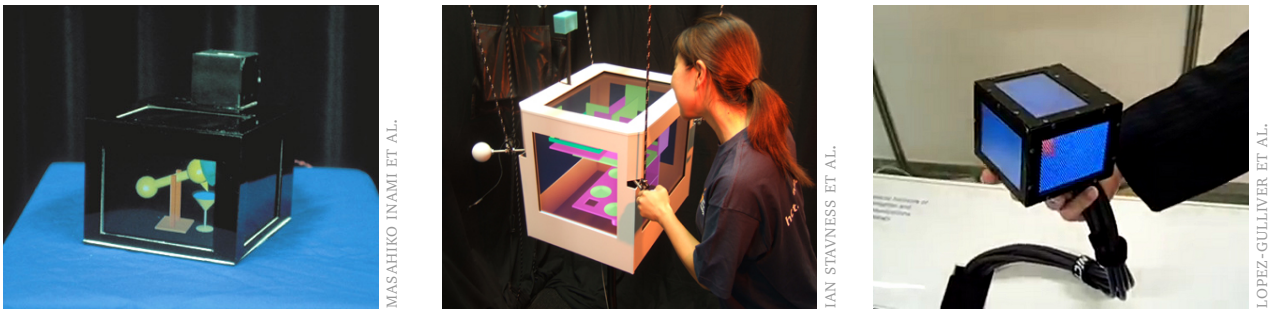
6.2.2 Écrans géométriques

La FTVR permet ainsi de positionner un appareil mobile directement sur un objet à manipuler sans que ce dernier ne soit coupé par le plan de l'écran. Cette co-localisation élimine donc les problèmes causés par la séparation entre le support de manipulation et l'objet virtuel.

Pourtant, un appareil mobile classique, de forme plate et muni d'un écran sur une seule face, présente toujours un inconvénient : celui de restreindre l'intervalle de rotations possibles avant de perdre le retour visuel. Même un deuxième écran situé sur l'autre face ne pourrait empêcher la perte du retour visuel sur les côtés de l'appareil [Spindler *et al.* 2012].

À ce niveau, il est important de rappeler que notre objectif est d'avoir un système entièrement contenu dans un unique dispositif portable et intégré. Pour cette raison, la seule surface d'affichage disponible est la surface du dispositif lui-même. Il s'ensuit que la seule manière de maintenir le retour visuel sur l'ensemble de l'espace de rotation serait d'avoir un dispositif non plus plat mais *en volume*, dont la surface serait entièrement recouverte d'écrans.

Un certain nombre de travaux ont proposé des dispositifs en volume équipés d'écrans, capables de fournir un retour visuel sous tous les angles. Certains de ces travaux exploitent la FTVR afin de produire l'illusion d'un espace 3D à l'intérieur de ce volume. De tels dispositifs sont connus sous le nom « d'écrans géométriques » (*geometric displays*, [Tang *et al.* 2014]), en référence aux transformations géométriques qui s'appliquent au rendu de chaque écran. Un des premiers exemples est le MEDIA cube [Inami 1997], dans lequel plusieurs



(a) MEDIA^3 [Inami 1997] (b) Cubee [Stavness *et al.* 2006] (c) gCubik [Lopez-Gulliver *et al.* 2009]

FIGURE 6.3 — Plusieurs exemples d'écrans géométriques, dispositifs volumiques recouverts d'écrans exploitant le rendu *fish-tank* pour représenter un espace 3D se trouvant à l'intérieur.

écrans LCD sont disposés de manière à former une boîte, combinés à un système de suivi de la tête pour produire l'illusion qu'une scène virtuelle se trouve à l'intérieur. Le CoCube [Brown *et al.* 2003] est un objet tangible en forme de cube, produisant ce type d'illusion lorsqu'il est observé à travers un visiocasque. Contrairement au MEDIA cube, le CoCube est conçu pour être manipulable par l'utilisateur. Le Cubee [Stavness *et al.* 2006] est un dispositif cubique capable du même résultat sans nécessiter de visiocasque, grâce à des écrans intégrés et un système de suivi de la tête. Le pCubee [Tang *et al.* 2014] est une évolution du Cubee, de taille réduite et visant à être plus portable. Un dispositif similaire est le gCubik [Lopez-Gulliver *et al.* 2009], qui repose sur des écrans autostéréoscopiques au lieu d'un suivi de la tête pour produire l'effet de FTVR. La plupart des dispositifs ci-dessus ont une forme cubique, vraisemblablement car il s'agit du volume le plus simple à construire à partir d'écrans en forme de dalle. D'autres formes sont malgré tout envisageables, comme un polyèdre quelconque [Harish et Narayanan 2013] ou même une sphère [Ferreira *et al.* 2014].

Les écrans géométriques semblent répondre à tous nos critères. Ils peuvent en théorie être rendus portables. Ils fournissent un retour visuel sur toute leur surface, sans interruption quel que soit l'angle sous lequel ils sont observés. Enfin, ils ne présentent pas de problème de clipping grâce à la FTVR, ce qui permet à un objet 3D d'apparaître à l'intérieur du volume (autorisant ainsi une manipulation co-localisée).

Malgré cela, les travaux cités plus haut se sont jusqu'à présent surtout focalisés sur les aspects techniques de réalisation de tels dispositifs. Leur potentiel pour la manipulation 3D apparaît encore peu exploité. Les rares techniques d'interaction explicitement mentionnées se limitent généralement à la seule visualisation d'objets, ou à quelques actions (roulements, secousses...) confinées au volume intérieur du dispositif.

Dans ce chapitre, nous proposons d'utiliser un écran géométrique pour effectuer une véritable manipulation d'objets sur 6 DdL, dans un espace s'étendant au-delà du dispositif lui-même. Nous avançons l'argument qu'un tel dispositif peut constituer en soi un système portable et

intégré pour la manipulation 3D, capable à la fois d'afficher une scène virtuelle et de servir de support de manipulation *co-localisé* dans cette même scène.

6.3 Concept

Sur la base du raisonnement ci-dessus, nous proposons le concept de « volume tangible ». Un volume tangible est un dispositif d'interaction constitué d'un objet physique unique, de taille suffisamment réduite pour être tenu dans la main. La surface de cet objet est entièrement couverte d'écrans, sur lesquels est affichée une scène virtuelle. Le rendu de chaque écran est continuellement ajusté selon la position de la tête de l'utilisateur (FTVR). Ainsi, une partie [1] de la scène virtuelle peut être vue « au travers » du dispositif. Ce dernier réagit également à son propre mouvement dans l'espace réel. L'utilisateur peut naviguer dans la scène en déplaçant le dispositif dans l'espace 3D (figure 6.4). À tout instant, le volume physique manipulé par l'utilisateur délimite donc un volume correspondant dans l'espace virtuel. Un tel dispositif constitue ainsi une *représentation tangible* d'un *volume* d'espace virtuel, tenu dans la main, et pouvant être directement positionné dans la scène virtuelle.

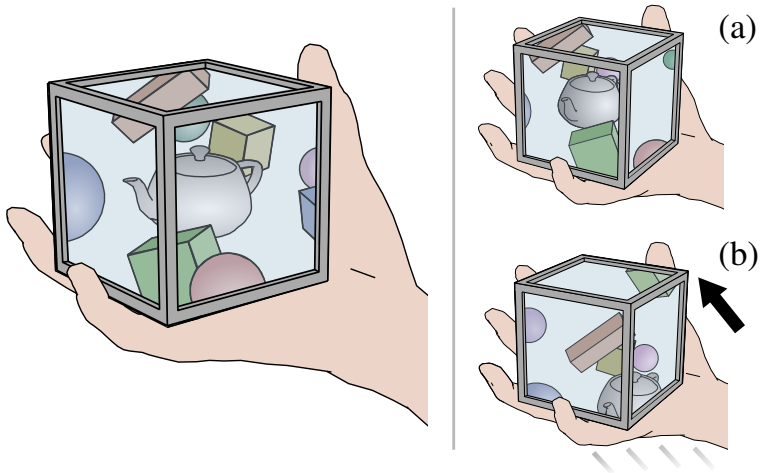


FIGURE 6.4 — Représentation de notre concept. Le « volume tangible » est un dispositif portable et intégré entièrement recouvert d'une surface d'affichage. **(a)** Un rendu *fish-tank* produit l'illusion que ce dispositif entoure un volume correspondant d'espace virtuel [1]. **(b)** En déplaçant le dispositif dans l'espace réel, il est possible de naviguer dans la scène virtuelle et de positionner ce volume directement *sur* l'objet que l'on souhaite manipuler.

1. Précisons que si notre dispositif délimite bien *physiquement* un volume correspondant d'espace virtuel, la partie *visible* de la scène virtuelle s'étend quant à elle à l'ensemble du cône de vision traversant le dispositif.

Ce volume peut dès lors être utilisé pour la manipulation 3D. Premièrement, le volume tangible est positionné *sur* l'objet virtuel à manipuler, l'entourant complètement. Cela est rendu possible par la technique de rendu FTVR comme expliqué précédemment. L'objet virtuel peut alors être « attaché » au volume (voir section 6.3.1) qui devient ainsi le support de manipulation. Une fois attaché, l'objet manipulé suit à l'identique tous les déplacements du volume tangible, en translation et en rotation, suivant le principe illustré par la figure 6.1(a). Mais contrairement à un appareil mobile classique, le volume tangible ne présente plus les inconvénients liés à la séparation avec l'objet manipulé. En effet, le volume tangible a été positionné *sur* l'objet virtuel au début de la manipulation. La séparation reste donc nulle durant toute la manipulation. D'autre part, le fait que le dispositif soit entièrement recouvert d'écrans et soit en volume, au lieu d'être plat, permet à l'utilisateur d'effectuer librement des rotations arbitrairement grandes sans jamais perdre le retour visuel.

Comme dans le cas d'un appareil mobile, ce concept combine affichage et support de manipulation dans un unique dispositif portable et intégré, permettant à la fois de visualiser la scène virtuelle et de manipuler des objets sur 6 DdL dans cette même scène. Mais contrairement à l'approche du chapitre précédent, cette nouvelle interface élimine entièrement la séparation avec les objets manipulés et offre un retour visuel ininterrompu sur l'ensemble de l'intervalle de rotations. Notre concept constitue donc une interface tout aussi portable et non-intrusive que l'approche précédente, mais sans les problèmes de manipulation qui lui sont associés.

Nous pouvons également établir un parallèle avec les approches basées sur des objets tangibles externes (chapitres 3 et 4). Le chapitre 5 visait à rendre ces interfaces plus portables en attribuant à la surface d'affichage (l'appareil mobile) le rôle de support de manipulation, de manière à se passer des objets tangibles externes. Le volume tangible peut être vu comme la solution opposée : rajouter à un objet tangible une surface d'affichage, afin de se passer de l'appareil mobile externe et des inconvénients associés à sa position distante.

Mais les bénéfices pour la manipulation par rapport à l'approche du chapitre précédent ne sont pas sans contrepartie : ils sont contrebalancés par une moindre *accessibilité* de l'interface. En effet, les écrans géométriques ne sont encore que des prototypes à l'heure actuelle, tandis que les appareils mobiles équipés de la technologie Tango (section 5.3.1) sont déjà des produits complets et commercialisés. Les obstacles technologiques à la démocratisation des écrans géométriques ne sont pourtant pas insurmontables à court terme, comme nous le verrons par la suite (section 6.4.1).

Dans la suite de cette section, nous décrivons plus en détail la manière dont les objets virtuels sont manipulés dans notre concept.

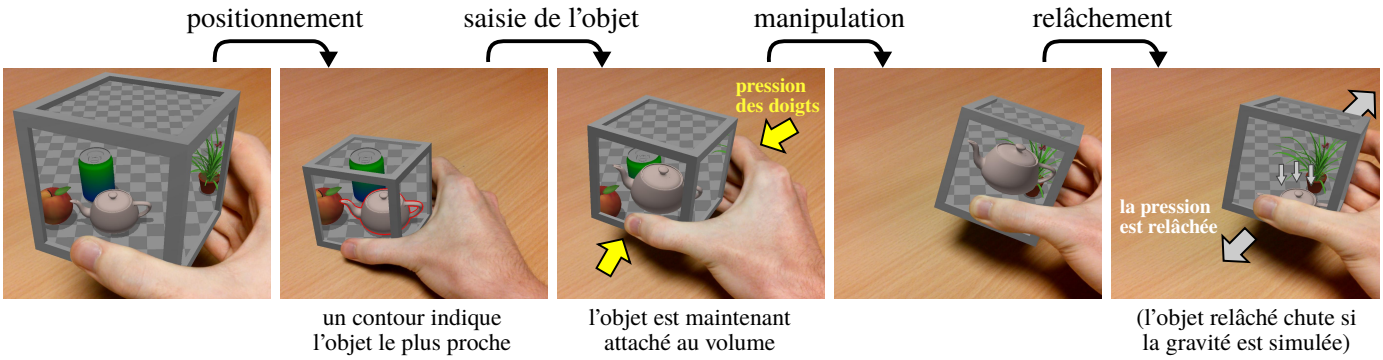


FIGURE 6.5 — Démonstration des différentes étapes de la manipulation d'objets dans notre concept. Premièrement, le volume tangible est positionné de manière à entourer un objet virtuel. Pour désambiguïser l'objet voulu parmi plusieurs objets rapprochés, un contour est affiché autour de l'objet le plus proche du centre du volume. Cet objet peut être saisi en exerçant une pression avec les doigts sur le volume. L'objet est alors « attaché » au volume et suit ses déplacements (translations et rotations) dans l'espace 3D. Il peut être détaché en relâchant la pression sur le volume. Si une simulation physique est implémentée, l'objet retombe au sol, sinon il conserve sa position et orientation.

6.3.1 Métaphore de saisie d'objets

Comme dans l'approche du chapitre précédent, l'utilisation d'un unique dispositif intégré nécessite de pouvoir sélectionner le ou les objets virtuels devant être attachés au support de manipulation à un instant donné (multiplexage temporel). À l'inverse, les interfaces à base d'objets tangibles externes ne nécessitent pas de sélection explicite car chaque objet tangible peut être associé à un objet virtuel différent (multiplexage spatial). Lorsque le choix est possible, le multiplexage spatial est préférable au multiplexage temporel car l'absence d'étape de sélection améliore l'efficacité et réduit la charge cognitive [Fitzmaurice 1996].

Examinons cependant ce que signifie vraiment la sélection dans le cas des objets tangibles externes. Bien que de telles approches permettent d'interagir avec plusieurs objets tangibles, il reste nécessaire à l'utilisateur de saisir chaque objet avec les mains avant de pouvoir interagir avec l'objet virtuel correspondant. Il existe donc tout de même une étape de sélection *implicite*, qui correspond au moment où l'utilisateur saisit un objet tangible avec la main. De ce point de vue, l'un des principaux intérêts des interfaces tangibles à multiplexage spatial n'est pas en réalité l'absence d'une étape de sélection, mais le fait que cette sélection est implicite et ne nécessite pas d'y penser consciemment. En d'autres termes, avancer la main vers un objet et le saisir constitue une façon *naturelle* de le sélectionner.

Notre concept offre l'opportunité de reproduire le même type de sélection implicite et naturelle que permettent des objets tangibles externes, mais dans un unique dispositif intégré. Comme expliqué plus haut, le volume tangible peut être positionné dans la scène 3D de manière à entourer un objet virtuel. Puisque le volume est tenu dans la main, la main se retrouve elle aussi positionnée autour de l'objet virtuel.

Nous proposons donc une technique de sélection consistant à *exercer une pression avec les doigts* sur le volume tangible afin de « saisir » l'objet virtuel ainsi entouré par la main (figure 6.5). Dans les faits, le volume tangible étant déjà tenu dans les doigts, l'action de sélection ne se déclenche que si la pression dépasse un certain seuil. Ce seuil permet de faire la différence entre une action volontaire de sélection et la simple manipulation du volume tangible. À partir de ce moment, l'objet virtuel devient fixé au volume et suit tous ses mouvements de translation et de rotation dans l'espace 3D. Relâcher l'objet, c'est-à-dire le désélectionner, s'effectue de la manière inverse : en relâchant la pression des doigts en dessous du seuil tout en maintenant le volume tangible lui-même dans la main. Après avoir été désélectionné, l'objet virtuel cesse de suivre les mouvements du volume.

Cette métaphore de sélection permet donc de saisir un objet virtuel de la même manière que l'on saisirait un objet physique : en plaçant la main autour de lui, puis en serrant les doigts pour le saisir.

6.3.2 Désambiguïsation entre plusieurs objets virtuels

Une difficulté qui apparaît dans la métaphore ci-dessus est le fait que les doigts ne peuvent pénétrer à l'intérieur du volume. Lorsque certains objets virtuels sont proches les uns des autres, plusieurs d'entre eux peuvent se retrouver en même temps à l'intérieur du volume. Les doigts ne pouvant pas être directement utilisés pour séparer l'objet voulu des autres, il est nécessaire de pouvoir contrôler quel objet sera sélectionné suite à une pression sur le volume.

Une solution possible est d'afficher un contour autour de l'objet qui se trouve le plus près du centre du volume, comme le montre la figure 6.5.

6.3.3 Manipulation bimanuelle

Une interface supportant le multiplexage spatial, telle qu'une interface à base d'objets tangibles externes, permet à l'utilisateur de manipuler des objets virtuels différents dans chaque main. L'approche décrite ici ayant pour vocation d'être contenue dans un unique dispositif, cette forme de manipulation bimanuelle n'est donc pas possible. Une solution pourrait être d'utiliser deux volumes tangibles (un pour chaque main), ce qui permettrait de se rapprocher de cette forme de manipulation. Mais cette option signifierait que l'interface serait à nouveau constituée de plusieurs éléments, ce que nous souhaitons précisément éviter.

En tant que dispositif unique, le volume tangible autorise cependant une autre forme d'interaction bimanuelle : manipuler un même objet virtuel avec les deux mains. En effet, le volume

tangible peut lui-même être tenu avec deux mains, ce qui permet par exemple d'appuyer avec les doigts de chaque main sur des faces opposées.

6.3.4 Simulation physique

Dans les interfaces à base d'objets tangibles, les lois physiques affectent les objets manipulés. Ainsi, relâcher un objet provoque sa chute sous l'effet de la gravité (section 3.2.1).

Cependant, dans notre interface, le volume tangible (support de manipulation) reste toujours dans la main. Relâcher un objet ne fait que détacher l'objet virtuel du volume. Si rien de plus n'est fait, l'objet virtuel reste alors là où l'utilisateur l'a laissé : flottant en l'air. Il est possible d'obtenir un comportement plus réaliste en rajoutant une simulation physique. Avec une telle simulation, un objet virtuel retombe jusqu'au sol s'il est relâché en l'air (figure 6.5). La simulation physique présente également l'avantage d'empêcher les objets virtuels de passer les uns au travers des autres. Cela renforce l'impression qu'ils sont solides et peuvent être directement saisis et manipulés par l'utilisateur.

Il faut pourtant noter qu'une telle simulation physique n'est pas forcément appropriée dans tous les cas. Par exemple, certaines tâches complexes de manipulation peuvent se révéler plus faciles à réaliser en les décomposant en plusieurs étapes (*clutching*). Avec une simulation de gravité, relâcher l'objet entre deux étapes aurait pour effet de le faire chuter. La simulation physique peut donc améliorer le réalisme, mais n'est pas forcément souhaitable en toutes circonstances.

6.4 Implémentation

À la différence des approches étudiées dans les chapitres précédents, le concept décrit ici nécessite un dispositif différent d'un appareil mobile standard. Or, parmi les dispositifs accessibles au grand public, il ne semble pas en exister aujourd'hui qui combine à la fois un écran géométrique et un suivi du point de vue de l'utilisateur. *A priori*, le concept de volume tangible semble donc s'éloigner de notre objectif d'accessibilité en tentant de répondre aux problèmes de manipulation identifiés plus haut.

Cependant, il est important de garder à l'esprit que le matériel est en constante évolution et de nouveaux dispositifs apparaissent sans cesse. Même si les technologies requises pour implémenter ce concept ne sont pas *aujourd'hui* accessibles au grand public, elles pourraient le devenir par la suite.

Dans cette section, nous examinons donc comment un tel dispositif pourrait être implémenté tout en respectant nos autres objectifs de portabilité et de non-intrusivité, de manière à évaluer sa faisabilité technique. Nous décrivons le processus de réalisation d'un premier démonstrateur, qui n'est pas encore entièrement fonctionnel à ce jour, illustrant ainsi concrètement quelles technologies sont déjà disponibles et lesquelles restent à développer. Nous présentons enfin un prototype fonctionnel mais partiellement simulé, que nous utilisons dans le reste de ce chapitre pour étudier expérimentalement l'usabilité de notre concept.

6.4.1 Faisabilité technique

Notre concept de volume tangible repose sur trois principales technologies. Premièrement, un dispositif de forme volumique, dont la surface serait entièrement couverte d'écrans et de capteurs de pression, et qui fonctionnerait de manière autonome. Deuxièmement, un suivi de la position de la tête de l'utilisateur (point de vue), indispensable pour le rendu *fish-tank*. Troisièmement, un suivi des mouvements du dispositif dans l'espace réel, afin de rendre possible la manipulation d'objets 3D à l'intérieur d'une scène plus large.

Écran géométrique

Pour ce qui est du premier aspect, les prototypes existants d'écrans géométriques montrent qu'il est déjà possible de fabriquer un dispositif volumique recouvert d'écrans. Les capteurs de pression pourraient ensuite être insérés en dessous de ces écrans, ou par-dessus ces derniers sous forme de couche tactile transparente.

Cependant, la principale limitation des prototypes existants est qu'ils sont encore dépendants d'une unité centrale externe pour le traitement et le rendu. Dans notre cas, les objectifs de portabilité et d'avoir une interface intégrée dans un unique élément exigent que le dispositif soit totalement autonome. Celui-ci doit donc contenir, en plus des écrans et des capteurs de pression, un processeur de puissance suffisante pour afficher une scène 3D en temps réel, et une batterie lui permettant de fonctionner pendant une durée acceptable.

Au vu des avancées actuelles dans le domaine des smartphones et tablettes, la question de la puissance de calcul et de l'autonomie dans un dispositif mobile semble être une question d'ores et déjà résolue.

GOPRO INC.

FIGURE 6.6 — Le GoPro Omni, un appareil capable de filmer à 360°. Mais contrairement à l'approche que nous proposons, ce dispositif n'a pas vocation à être couvert d'une surface d'affichage ce qui simplifie le placement des caméras.

Suivi du point de vue de l'utilisateur

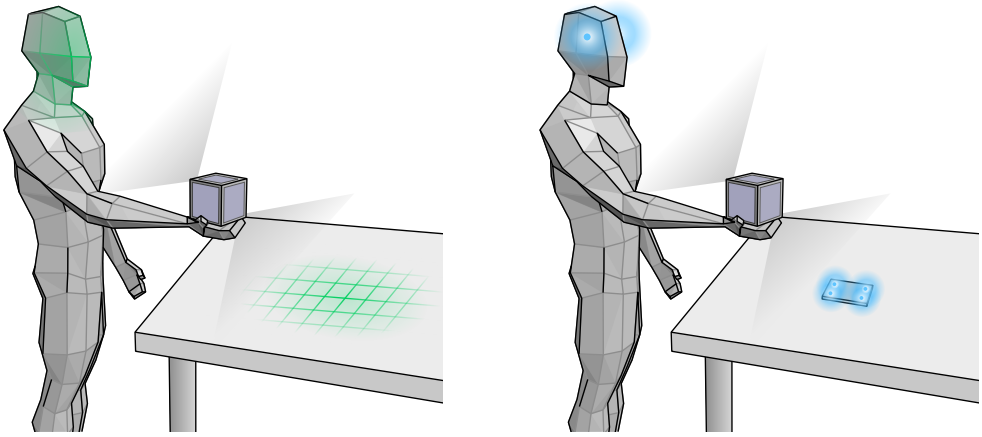
Le deuxième aspect concerne le suivi de la tête de l'utilisateur par rapport au dispositif. Dans notre concept, ce suivi est rendu plus ardu par le fait que l'interface devrait être portable et entièrement intégrée dans le dispositif. Certains des prototypes existants d'écrans géométriques (par exemple [Tang *et al.* 2014]) emploient des balises magnétiques pour effectuer ce suivi : l'une est fixée sur la tête de l'utilisateur, une autre sur le dispositif, et une base externe sert d'émetteur. Si la base pouvait être intégrée au dispositif ou portée par l'utilisateur, cette approche pourrait devenir portable. Cependant, elle serait aussi relativement intrusive car elle obligerait l'utilisateur à porter un équipement spécifique.

Afin de répondre entièrement à nos objectifs de portabilité et de non-intrusivité, il faudrait pouvoir placer un certain nombre de caméras (les seuls capteurs à même de suivre la tête de l'utilisateur sans instrumentation préalable) sur toute la surface du dispositif, de manière à couvrir l'ensemble des directions possibles sans obstruction. Il faudrait ensuite traiter simultanément ces flux vidéo pour détecter et suivre la position de la tête.

Il existe bien aujourd'hui des dispositifs munis de caméras « 360° » (figure 6.6), mais ceux-ci n'ont pas comme contrainte supplémentaire d'être recouverts d'écrans. La première difficulté est donc d'intégrer toutes ces caméras dans un écran géométrique. Le volume tangible étant déjà recouvert d'écrans, le seul espace restant pour placer des caméras se trouve sur les angles ou sur les arêtes. La seconde difficulté est de traiter autant de flux vidéo en parallèle. Il n'est pas certain que les processeurs mobiles actuels possèdent suffisamment de puissance pour cela, même si leur performance va continuer à augmenter. Enfin, le dernier défi est que le suivi optique de visage n'est pas totalement fiable à ce jour. Les algorithmes actuels nécessitent encore des conditions bien précises (luminosité, contraste, bonne résolution, faible rotation du visage [Rabhi *et al.* 2015]) qui sont difficiles à garantir en toutes circonstances,

et d'autant plus dans un contexte de portabilité. Un moyen d'améliorer cette fiabilité serait de faire porter à l'utilisateur un accessoire peu contraignant, tel que des lunettes, conçu pour faciliter le suivi indépendamment des conditions extérieures (figure 6.7(b)). Cette solution représenterait un compromis sur l'intégration, mais permettrait d'implémenter ce concept dans l'état actuel des technologies de suivi optique.

Malgré certaines difficultés qui restent à surmonter, parvenir à suivre le point de vue de l'utilisateur depuis un tel dispositif intégré ne paraît pas impossible à moyen terme. Compte tenu des limitations des technologies actuelles, un compromis consistant à faire porter un accessoire à l'utilisateur semble toutefois la solution la plus réaliste dans l'immédiat.



(a) Implémentation idéale : le suivi 3D est entièrement intégré dans le dispositif.

(b) Compromis sur l'intégration, facilitant l'implémentation avec les technologies actuelles.

FIGURE 6.7 — En principe, notre concept suppose que le suivi est entièrement contenu dans le dispositif lui-même (a). Mais compte tenu des limitations des technologies actuelles, une solution à court terme pourrait être de placer sur l'utilisateur ainsi que dans l'environnement des marqueurs permettant de faciliter le suivi, comme des diodes lumineuses (b).

Suivi des mouvements du dispositif

Le troisième aspect que requiert notre concept est de suivre les déplacements du volume tangible par rapport à l'environnement extérieur. Ce suivi est nécessaire pour placer la scène virtuelle dans un référentiel indépendant de celui du volume, ce qui permet ainsi à l'utilisateur de manipuler des objets dans un espace co-localisé avec l'espace réel. Dans une interface qui se veut portable et intégrée, il ne peut y avoir de système de suivi externe pour fournir ce référentiel. Comme pour le suivi de la tête, la seule option est donc de réaliser ce suivi depuis le dispositif lui-même (suivi *inside-out*).

Nous avons vu dans le chapitre précédent que les appareils mobiles Tango sont capables de suivre leurs propres mouvements dans l'environnement, et ce de manière totalement autonome. Il est donc possible d'imaginer placer des capteurs similaires à ceux de la technologie Tango sur la surface du volume tangible. Dans l'éventualité où le suivi du point de vue est effectué par des caméras placées sur toute la surface du dispositif, ces mêmes caméras pourraient être utilisées pour suivre également l'environnement dans toutes les directions.

Cependant, effectuer les mêmes traitements que la technologie Tango mais sur un nombre plus important de capteurs risque de demander une puissance de calcul nettement plus importante. Compte tenu de l'évolution en puissance des processeurs mobiles, cet objectif paraît envisageable dans un futur relativement proche. Une alternative pour faciliter une première implémentation serait de placer dans l'environnement un marqueur moins coûteux à suivre, même si cela contreviendrait à notre objectif d'intégration.

6.4.2 Démonstrateur

Nous décrivons ici le travail d'implémentation d'un premier démonstrateur de notre concept. Celui-ci n'est à ce jour pas encore complet, en raison notamment du délai nécessaire à la réalisation de certains composants spécifiques. Malgré cela, le travail effectué jusqu'à présent offre d'ores et déjà un aperçu concret des possibilités comme des difficultés d'implémentation de ce concept en pratique.

Processeur

L'affichage d'objets 3D sur plusieurs écrans ainsi que le pilotage simultané de multiples capteurs requiert une capacité de traitement non négligeable. Toutes ces tâches doivent de plus être réalisées par un équipement de taille réduite, pouvant être intégré dans un dispositif portable et autonome. Enfin, un tel équipement doit pouvoir être programmable, car la combinaison de traitements qui caractérise notre concept ne se rencontre pas aujourd'hui dans les dispositifs existants.

Ces dernières années ont vu un développement important de cartes électroniques programmables disponibles sur le marché, et dont la puissance de calcul ne cesse d'augmenter. On peut notamment citer les cartes « Arduino »[2] et les cartes « Raspberry Pi »[3]. Pour ce démonstrateur, nous avons sélectionné une carte Raspberry Pi modèle « Zero » (figure 6.8) car elle combine plusieurs caractéristiques essentielles. Premièrement, elle est de très petite

2. `https://www.arduino.cc/`
3. `https://www.raspberrypi.org/`

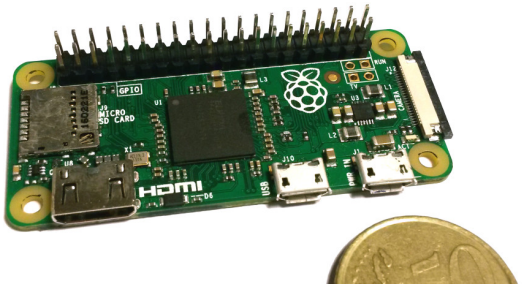


FIGURE 6.8 — Le Raspberry Pi Zero, une carte électronique miniature et programmable, servant au traitement et au rendu graphique dans notre démonstrateur.

taille par rapport aux autres cartes (65 mm × 30 mm × 5 mm), ce qui facilite son intégration dans un dispositif tenant dans la main. Deuxièmement, à la différence des Arduino, elle est munie d'un processeur graphique (Broadcom VideoCore) ce qui la rend utilisable pour des applications 3D. Enfin, elle dispose de nombreux pilotes de périphériques, en particulier pour des écrans miniatures.

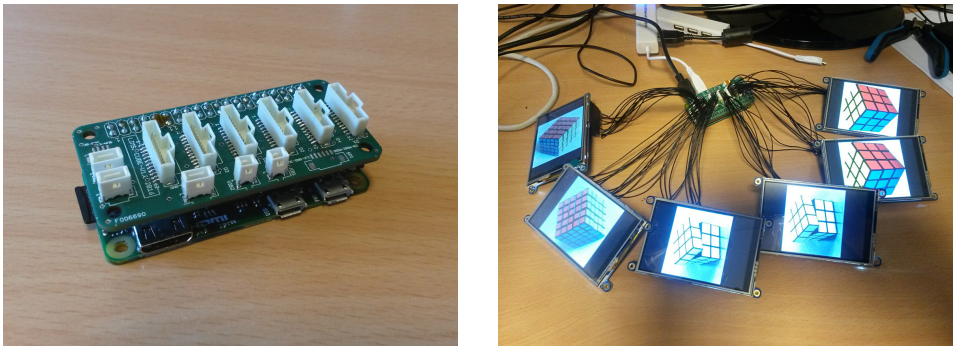
Cette carte Raspberry Pi Zero, qui constitue le cœur de notre implémentation, est alimentée par une batterie elle aussi placée dans le dispositif. Le système reste donc entièrement portable et autonome.

Écrans et capteurs de pression

Comme la plupart des prototypes d'écrans géométriques décrits jusqu'à présent, nous avons fait le choix d'une forme cubique pour notre dispositif. Un cube étant constitué de six faces, il nous faut intégrer six écrans et six capteurs de pression pour recouvrir l'ensemble de la surface du volume tangible.

Une première difficulté que nous avons rencontrée est qu'il ne semble pas exister d'écrans de forme carrée dans les dimensions que nous recherchons, à la fois suffisamment compact pour tenir dans la main mais pas trop pour pouvoir intégrer une carte électronique à l'intérieur du volume. La grande majorité des modèles carrés disponibles sur le marché sont aux dimensions de montres connectées, ce qui est trop petit pour notre implémentation. Nous avons donc choisi, dans un premier temps, d'utiliser des écrans de la dimension souhaitée mais rectangulaires, en les entourant d'une bordure pour former des faces carrées (de la même manière que les prototypes de la figure 6.3).

Pour piloter ces écrans, nous avons conçu une carte électronique secondaire (figure 6.9(a)) de même dimensions que la carte principale. Cette carte permet de rediriger les entrées/sorties du processeur vers six connecteurs, chacun étant relié à une face différente du cube. Par



(a) Carte d'extension (b) Pilotage simultané de six écrans

FIGURE 6.9 — Carte d'extension conçue pour piloter six écrans simultanément avec une même carte Raspberry Pi Zero. On notera que les écrans sont câblés par paires (figure (b)), ce qui permet de réduire de moitié la puissance de calcul nécessaire en ne considérant que les trois écrans qui font face à l'utilisateur.

ailleurs, le pilote logiciel des écrans a été modifié de manière à pouvoir contrôler plusieurs écrans simultanément. Le résultat est présenté dans la figure 6.9(b).

Une nouvelle difficulté qui s'est posée à ce niveau est que le processeur ne dispose pas d'une fréquence d'horloge suffisante pour contrôler six écrans en parallèle. Cependant, dans le cas d'un dispositif en forme de cube, nous pouvons remarquer que l'utilisateur ne pourra voir au maximum que trois faces en même temps. Nous avons donc câblé ensemble les écrans se trouvant sur des faces opposées, ce qui permet au processeur de n'avoir à contrôler que trois images au lieu de six. Les écrans opposés à l'utilisateur affichent ainsi le même contenu que les écrans lui faisant face, mais cela ne présente pas d'inconvénient car ces écrans ne sont pas visibles. Il est par ailleurs prévu de piloter séparément le rétroéclairage des écrans afin d'éteindre ceux qui ne font pas face à l'utilisateur, ce qui permettra par la même occasion d'économiser la batterie du dispositif.

Outre les écrans, notre concept nécessite également d'intégrer des capteurs de pression sur chaque face. Nous avons pour cela utilisé une combinaison de technologies (figure 6.10). D'une part, un film résistif transparent placé devant chaque écran permet de récupérer une information tactile. D'autre part, un second capteur de pression de type FSR (*force-sensing resistor*) est placé sous chaque écran. La rigidité des écrans situés au-dessus fait que ce capteur FSR ne répond qu'à des forces plus importantes, à l'inverse du film tactile qui ne détecte que des pressions faibles. Ensemble, ces deux capteurs permettent donc de mesurer la pression exercée par l'utilisateur sur chaque face.

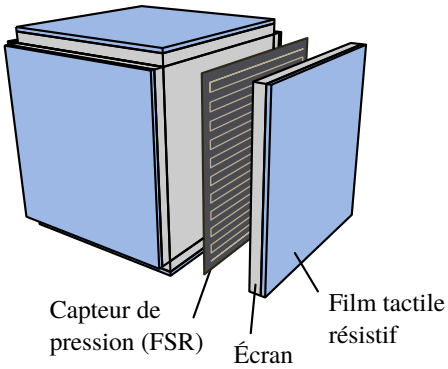


FIGURE 6.10 — La détection de pression dans notre démonstrateur repose sur une combinaison de deux technologies : un film tactile résistif, qui peut être placé devant les écrans mais ne répond qu'à des pressions faibles, et des capteurs de type FSR insérés sous chaque écran, répondant à des pressions plus importantes.

Système de suivi

Dans notre concept, le système de suivi est lui aussi censé être intégré au dispositif pour que le système reste portable et constitué d'un seul élément. Comme nous l'avons vu plus haut, l'approche répondant le mieux à ces objectifs est de placer des caméras sur la surface du dispositif afin de suivre optiquement des objets dans l'espace environnant : la tête de l'utilisateur pour le suivi du point de vue, et les déplacements relatifs de l'environnement pour le suivi du dispositif lui-même.

Un premier défi important est de couvrir l'ensemble des directions autour du dispositif. Une solution serait d'intégrer un grand nombre de caméras, mais cela est rendu difficile par la présence des écrans sur chaque face et par la puissance de calcul limitée du processeur intégré. Le nombre total de caméras nécessaire peut être réduit en utilisant des lentilles « grand angle » pour élargir leur champ de vision.

Une deuxième difficulté est qu'il reste aujourd'hui difficile de réaliser un suivi du visage fiable et en temps réel, *a fortiori* avec de multiples caméras et sur un processeur de puissance limitée. Un problème similaire se pose pour le suivi de l'environnement. Comme expliqué dans la section 6.4.1, une solution à court terme serait d'équiper l'utilisateur d'un dispositif (aussi peu intrusif que possible) facilitant le suivi, ce qui représenterait un compromis sur l'intégration. Une même approche peut être envisagée pour le suivi de l'environnement.

Pour ce démonstrateur, nous avons donc substitué aux caméras de « simples » capteurs infrarouge capables de détecter des points brillants. Nous équipons par ailleurs l'utilisateur d'une paire de lunettes munie de diodes électroluminescentes (DEL) infrarouge, fournissant aux

capteurs une cible plus facile à suivre. L'environnement est suivi de la même manière, grâce à un « marqueur » fixe constitué de plusieurs DEL infrarouge immédiatement identifiables par les capteurs. Nous faisons ainsi le choix dans ce premier démonstrateur d'implémenter l'approche illustrée dans la figure 6.7(b), bien que l'objectif à terme soit toujours d'avoir un suivi entièrement intégré.

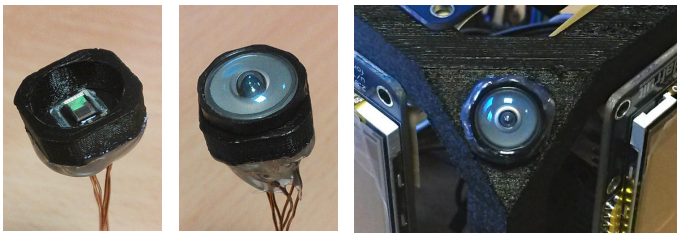


FIGURE 6.11 — Capteurs utilisés pour le suivi dans notre démonstrateur. Nous rajoutons sur chaque capteur une lentille grand-angle pour élargir son champ de vision. Les capteurs ainsi assemblés sont ensuite placés sur chaque sommet de la structure.

Chaque « caméra » est donc constituée d'un capteur infrarouge surmonté d'une lentille « fisheye » (grand-angle) pour élargir son champ de vision (figure 6.11). Ces caméras sont destinées à être placées sur chacun des sommets du cube, soit au total 8 capteurs. À ce jour, trois capteurs ont été réalisés. L'avancement actuel de notre prototype est illustré dans la figure 6.12.

Il est cependant apparu au cours de l'assemblage que le recouvrement du champ de vision de ces capteurs était insuffisant pour maintenir le suivi tout autour du dispositif, malgré la présence des lentilles. Actuellement, le suivi n'est donc pas encore entièrement fonctionnel dans notre démonstrateur. Nous envisageons de placer des capteurs supplémentaires sur chaque face du cube, en plus de ceux équipant déjà chaque sommet, en tirant parti de l'espace restant autour des écrans rectangulaires.

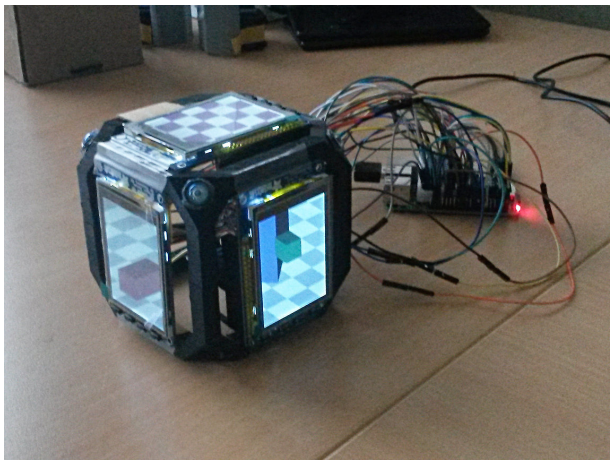


FIGURE 6.12 — Démonstrateur en partie assemblé. Le suivi 3D n'est pas encore satisfaisant, en raison du champ de vision des capteurs plus limité que prévu et de leur nombre insuffisant pour couvrir l'ensemble des directions autour du dispositif.

6.4.3 Prototype partiellement simulé

Le démonstrateur ci-dessus n'étant pas encore fonctionnel, il ne peut être utilisé pour mener des études expérimentales sur l'usabilité de notre concept. Nous avons donc choisi d'implémenter un prototype *partiellement simulé* pour nos évaluations. Ce prototype utilise la réalité augmentée pour simuler certains aspects matériels de notre concept.

La simulation d'une partie d'un dispositif d'interaction a déjà été employée dans de précédentes études [Baricevic *et al.* 2012]. Cette approche présente l'avantage d'offrir un rendu parfait et un suivi 3D fiable, ce qui n'est pas forcément le cas d'un démonstrateur de laboratoire mais est pourtant essentiel pour la validité des études expérimentales.

Notre prototype simulé repose sur un objet cubique, qui représente le volume tangible. Ce cube est dimensionné de manière à être facilement tenu dans une main (6,4 cm de côté), et chacune de ses faces est couverte d'un marqueur texturé. Une tablette est placée entre l'utilisateur et le cube, jouant le rôle de fenêtre de réalité augmentée. Lorsque le cube est observé à travers la tablette, ses faces sont remplacées par des « écrans simulés » au travers desquels apparaît la scène virtuelle (figure 6.13). Une bordure est rajoutée autour de ces écrans, afin de simuler le fait que de véritables écrans auraient eux aussi une bordure.

L'utilisation de la réalité augmentée permet de résoudre implicitement la question du suivi du point de vue, représenté ici par la tablette : les faces du cube étant suivies par la tablette, la transformation inverse équivaut à suivre le « point de vue ». Les déplacements du cube lui-même par rapport à l'environnement sont suivis grâce à un marqueur fixe supplémentaire

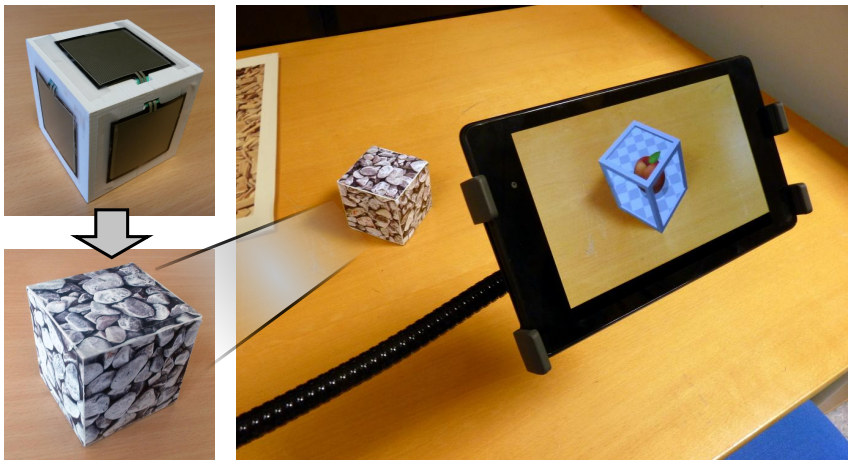


FIGURE 6.13 — Prototype fonctionnel mais partiellement simulé. L'affichage et le suivi sont simulés en réalité augmentée, par le biais d'une tablette placée devant les yeux de l'utilisateur. La détection de pression est quant à elle implémentée matériellement, grâce à des capteurs (situés sous les marqueurs texturés) communiquant sans fil avec la tablette.

posé derrière la tablette. Tous ces marqueurs sont suivis en utilisant le système Vuforia [4]. La tablette est maintenue par un support de manière à se situer dans l'axe du regard de l'utilisateur, lui permettant de manipuler le cube devant lui comme s'il s'agissait d'un véritable dispositif équipé d'écrans.

La technique de saisie d'objets est implémentée matériellement, grâce à six capteurs de pression (Interlink® FSR 406) placés en dessous des marqueurs sur chaque face du cube. Ces capteurs sont donc invisibles pour les utilisateurs. Ils sont connectés à un microcontrôleur (RFduino RFD22102) situé au centre du cube. Ce microcontrôleur est alimenté par une batterie elle aussi située à l'intérieur du cube, avec un port de rechargement se trouvant sous l'un des marqueurs. Le microcontrôleur transmet continuellement à la tablette les valeurs de pression par une liaison Bluetooth à une fréquence de 10 Hz. Enfin, la simulation physique est implémentée sur la tablette en utilisant le moteur Bullet [5].

En dépit des avantages que présente la simulation, ce prototype partiel n'est pas équivalent à une véritable implémentation. Il présente même d'importantes différences sur certains points. Premièrement, l'existence de la tablette, qui ne fait pas partie de notre concept, réduit nécessairement la fidélité de ce prototype. Le suivi du « point de vue » étant effectué depuis la tablette, les propres mouvements de la tête de l'utilisateur ne sont ainsi pas pris en compte. Cela n'empêche pas les objets d'apparaître « dans » le volume du cube, le rendu *fish-tank* étant quand même réalisé (depuis le point de vue de la tablette), mais l'intensité de cet effet est réduite. Par ailleurs, certains mouvements, comme le fait de rapprocher le volume tangible des yeux, sont rendus plus difficiles par la présence de la tablette. Enfin, l'image des écrans étant superposée à celle obtenue par la caméra arrière de la tablette, les doigts de l'utilisateur sont occultés par la scène virtuelle (figure 6.14) alors qu'ils devraient rester visibles si le dispositif possédait de véritables écrans. Malgré cela, nous considérons que les avantages en terme de qualité de rendu et de suivi d'un prototype partiellement simulé sont plus importants, et que ce dernier devrait être suffisant pour nos études expérimentales.

Dans le reste de ce chapitre, nous utilisons donc ce prototype partiel pour étudier deux importantes questions concernant l'usabilité de notre concept. Premièrement, nous évaluons l'intuitivité de la métaphore de saisie et de manipulation d'objets à travers le volume tangible. Dans une deuxième étude, nous évaluons l'impact que peut avoir un champ de vision restreint (limité à la surface apparente du volume) sur la perception spatiale de la scène virtuelle.

4. `http://www.vuforia.com/`
5. `http://www.bulletphysics.org/`

6.5 Première étude : sélection et manipulation

Dans le monde réel, saisir un objet en appuyant les doigts dessus puis le déplacer tout en maintenant la pression est une action parfaitement naturelle. Cependant, il reste à déterminer si cette même action est toujours aussi naturelle lorsqu'elle est réalisée *à travers* un volume tangible. L'objectif de cette première étude est donc d'évaluer l'intuitivité de la métaphore de saisie pour la sélection et la manipulation d'objets dans notre interface.

Plus spécifiquement, le but de cette étude est de vérifier si les utilisateurs parviennent à découvrir *d'eux-mêmes*, avec le moins d'explications possible, comment saisir et déplacer des objets virtuels avec notre interface. Bien sûr, si certains utilisateurs n'y parvenaient pas, cette seule mesure ne permettrait pas de comprendre ce qui a posé problème. C'est pourquoi un deuxième objectif de cette étude est de déterminer à quel point de la manipulation ces utilisateurs auraient besoin d'une explication lors de leur premier contact avec l'interface.

6.5.1 Participants et plateforme expérimentale

Cette expérience a été menée avec 36 participants (15 femmes, 21 hommes) âgés de 20 à 52 ans (moyenne=29,5, σ=9,4). Aucun d'entre eux n'avait de connaissance préalable de notre interface ou de la métaphore de saisie.

La plateforme expérimentale était constituée du prototype partiellement simulé en réalité augmentée décrit dans la section précédente. Pour cette expérience en particulier, nous avons légèrement relevé le seuil de pression afin de nous assurer que la saisie d'objets ne se déclenche que sur une action consciente et volontaire de l'utilisateur.

6.5.2 Protocole

Avant de commencer l'expérience, nous expliquons aux participants que celle-ci consistera à « manipuler le cube ». Nous leur expliquons également pourquoi une tablette se trouve devant eux : « pour simuler la présence d'écrans qui auraient dû couvrir les faces du cube, mais qui n'y sont pas pour des raisons techniques ». Nous leur montrons ensuite comment le dispositif se transforme en un écran cubique lorsqu'il est placé derrière la tablette, et comment une scène virtuelle devient visible au travers. Si notre prototype avait possédé de véritables écrans avec suivi du point de vue, les participants auraient vraisemblablement constaté cela immédiatement. Cette phase d'explications, nécessaire pour clarifier le rôle de la réalité augmentée dans notre prototype partiellement simulé, n'introduit donc pas de biais par rapport à une potentielle implémentation complète d'un volume tangible. Cependant,

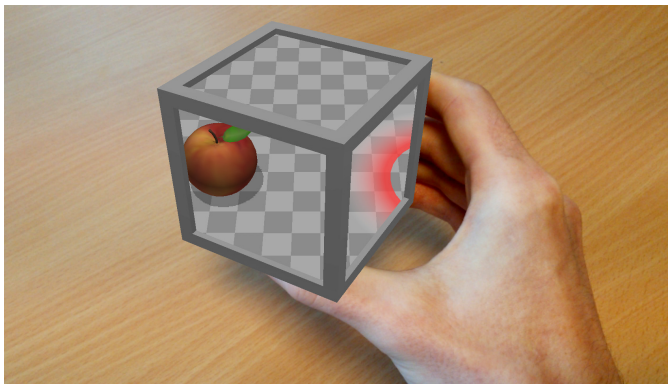


FIGURE 6.14 — Copie d'écran de la première expérience, correspondant à ce que les participants pouvaient voir à travers la tablette. On remarquera l'occultation des doigts due à la simulation des écrans en réalité augmentée, mais qui n'impacte pas le déroulement de l'expérience. La tâche consiste à saisir la pomme virtuelle et à la déposer dans le cercle sur la droite.

nous ne donnons aux participants *aucune* explication quant à la manière d'interagir avec les objets virtuels.

Nous présentons ensuite la tâche à effectuer : déplacer un objet virtuel (une pomme) vers une cible proche (représentée par un cercle au sol, voir la figure 6.14). Les participants sont informés que cette tâche doit être réalisée « en utilisant le cube ». Nous leur précisons notamment que la tablette n'intervient pas dans la manipulation. Enfin, nous leur expliquons qu'ils doivent découvrir comment faire « par eux-mêmes, autant que possible ».

Comme mentionné plus haut, il est possible que certains participants ne parviennent pas à terminer la tâche sans explication. Afin de déterminer précisément de quel type d'aide auraient besoin ces participants, nous avons établi une liste d'indices textuels ordonnés de manière à être de plus en plus explicites. Durant l'expérience, un bouton peut être pressé pour obtenir un indice. Les indices sont successivement révélés sur l'écran de la tablette, dans l'ordre suivant :

1. « Placer le cube sur la pomme »
2. « Appuyer les doigts sur le cube pour saisir la pomme »
3. « Déplacer le cube tout en maintenant la pression »

Ces indices sont spécifiquement conçus pour couvrir les différentes étapes de la manipulation d'un objet virtuel dans notre interface (figure 6.5) : positionnement, saisie et déplacement (l'étape de relâchement est implicite car il s'agit d'un quasi-mode). Par ailleurs, le choix d'utiliser des indices *textuels* à la place de représentations visuelles, telles que les flèches représentant la pression dans la figure 6.5, permet d'éviter toute ambiguïté d'interprétation qui pourrait perturber les conclusions de l'expérience.

Les participants sont informés de l'existence de ce bouton d'indice, mais nous les encourageons à tenter d'accomplir la tâche en utilisant le moins d'indices possible.

6.5.3 Résultats et discussion

La figure 6.15 donne le pourcentage de participants selon le nombre d'indices dont ils ont eu besoin pour terminer la tâche. Un total de 19 participants (53 %) ont pu accomplir la tâche sans demander aucun indice. Parmi les participants restants, tous sauf un (45 %) ont eu besoin des deux premiers indices pour terminer la tâche. Le troisième indice n'a jamais été utilisé. Nous rapportons également le temps d'accomplissement de la tâche pour les participants qui n'ont utilisé aucun indice. Le temps moyen était de 63,5 s (σ=34,3 s).

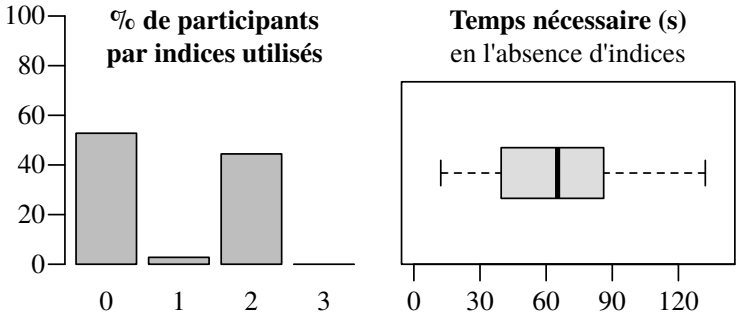


FIGURE 6.15 — Nombre d'indices ayant été nécessaires aux participants pour découvrir comment manipuler un objet virtuel avec notre interface. Le temps d'accomplissement de la tâche est également représenté, uniquement pour les 19 participants n'ayant demandé aucun indice.

Une majorité des participants ont donc pu découvrir *par eux-mêmes* comment saisir et manipuler des objets virtuels avec le volume tangible, sans la moindre explication préalable. Pour une interface si différente de la manière dont la plupart des utilisateurs interagissent actuellement avec des objets 3D, et plus généralement avec les systèmes informatiques, il s'agit là d'un résultat particulièrement encourageant.

À première vue, le temps d'accomplissement de la tâche (autour d'une minute en moyenne) peut paraître long. Cependant, il s'agit là du temps mis pour découvrir seul comment se servir d'un dispositif totalement nouveau *et* accomplir la tâche demandée pour la première fois. Il semble donc que l'idée de saisir un objet virtuel « à travers » un volume tangible a été spontanément envisagée par une majorité des participants, et ce en un temps raisonnable.

Pour ceux qui n'ont pas pu accomplir la tâche sans aide, le nombre d'indices demandés permet de mieux comprendre quelles étapes de la manipulation ont posé problème. Le premier indice était conçu pour mettre en évidence de potentielles difficultés dans le positionnement du volume tangible autour de l'objet virtuel. Quasiment aucun des participants ayant eu

besoin d'assistance n'a été aidé par ce premier indice. Par conséquent, l'étape de positionnement ne semble pas être celle qui les a empêchés de terminer la tâche. Effectivement, nous avons constaté que tous les participants remarquaient rapidement qu'un contour apparaissait autour de l'objet virtuel lorsque celui-ci se trouvait à l'intérieur du volume. Mais certains d'entre eux ont manifestement interprété ce contour comme signifiant que l'objet était sélectionné, car ils tentaient alors de déplacer l'objet sans presser les doigts.

Parmi les autres stratégies adoptées, un nombre étonnamment élevé de participants ont tenté de « pousser » l'objet virtuel avec le cube. Cela pourrait indiquer qu'ils supposaient que les bords du volume tangible seraient solides dans la scène virtuelle. Beaucoup ont également tenté de taper ou effleurer un doigt sur la surface du volume. Dans certains cas, il s'agissait manifestement d'une tentative de reproduire un « clic » de souris ou d'écran tactile, particulièrement lorsque les participants posaient la main sur le cube comme s'il s'agissait d'une souris et tentaient de cliquer (voire de double-cliquer) sur la face supérieure. Dans d'autres cas, il s'agissait plus clairement d'une tentative d'agir sur l'objet virtuel au travers du volume, notamment lorsque le doigt frappait une des faces latérales. Enfin, certains participants ont tenté d'entourer totalement le cube avec leurs mains. Cela pourrait montrer une volonté de saisir l'objet virtuel, mais avait aussi pour conséquence d'interrompre le retour visuel. Il était donc impossible de savoir si la pression exercée était suffisante pour déclencher la technique de saisie.

Presque tous les participants ayant demandé le premier indice ont aussi eu besoin du deuxième indice pour terminer la tâche. Ce deuxième indice était conçu pour mettre en évidence de possibles difficultés avec la métaphore de saisie par pression des doigts. Aucun participant n'a demandé le troisième indice. Pour ceux ayant eu besoin d'aide, le principal obstacle était donc de découvrir qu'ils devaient presser les doigts sur le volume tangible. Le troisième indice couvrait l'action de déplacer l'objet sélectionné pendant qu'il est fixé au volume tangible. Aucun participant n'ayant eu besoin de cet indice, nous pouvons en déduire qu'aucun d'entre eux n'a rencontré de problème avec cette dernière étape.

6.6 Deuxième étude : champ de vision

Une difficulté potentielle posée par le concept de volume tangible est le champ de vision restreint durant la manipulation. Un de nos principaux objectifs étant d'avoir une interface totalement intégrée, la seule surface disponible pour l'affichage est la surface du volume tangible lui-même. Le volume sert donc à la fois pour l'affichage et pour la manipulation. Lorsque le volume tangible est placé sur un objet virtuel pour débuter la manipulation co-localisée, il s'éloigne également des yeux de l'utilisateur. La taille apparente de la surface d'affichage devient donc plus petite. En raison du rendu *fish-tank*, le champ de vision sur

la scène virtuelle devient lui aussi plus étroit. Inversement, rapprocher le volume des yeux élargirait le champ de vision, mais rendrait aussi impossible la manipulation co-localisée.

Un champ de vision limité ne permet de voir qu'une petite partie de la scène virtuelle à la fois. Cela peut donc gêner les utilisateurs pour se constituer une représentation mentale de l'ensemble de la scène, ou pour retrouver un objet virtuel situé hors du champ de vision. La perception spatiale (*spatial awareness*) est donc un enjeu important en présence d'un champ de vision restreint.

Dans une étude sur la réalité virtuelle mobile, Hwang *et al.* [2006] ont observé que les mouvements de l'appareil mobile dans la scène pouvaient compenser un champ de vision limité. Cependant, dans notre situation le volume tangible est plus petit que nombre d'appareils mobiles traditionnels (pour faciliter la métaphore de saisie) et est généralement tenu plus loin des yeux pendant la manipulation (car il est positionné sur l'objet à manipuler). Le champ de vision effectif lors de l'utilisation du volume tangible est donc en moyenne plus limité que ceux testés dans l'étude sus-citée.

Si ce champ de vision restreint s'avérait avoir un impact négatif sur la perception spatiale de la scène par l'utilisateur, cela pourrait constituer un inconvénient de notre approche visant à intégrer affichage et manipulation co-localisée dans un seul objet. Il est cependant possible que le mouvement du volume tangible dans la scène et l'effet *fish-tank* soient suffisants pour compenser la limitation du champ de vision dans notre cas. Pour répondre à cette question, nous avons donc mené une seconde étude mesurant l'impact du champ de vision sur la perception spatiale.

6.6.1 Participants et plateforme expérimentale

Cette expérience a été menée avec 32 participants (20 hommes, 12 femmes) âgés de 20 à 52 ans (moyenne=29,9, σ=10,0). Tous avaient pris part à la première étude, et bénéficiaient donc d'une expérience préalable quant à l'utilisation de notre interface.

La plateforme expérimentale était ici encore constituée de notre prototype partiellement simulé en réalité augmentée, dont le seuil de pression a été ramené à sa valeur normale.

6.6.2 Protocole

La tâche consiste à naviguer dans une scène virtuelle contenant 6 objets virtuels différents, et à rechercher une cible apparaissant après un certain délai. Chaque cible contient la silhouette

FIGURE 6.16 — Les deux conditions de la seconde expérience, visant à mesurer l'impact d'un champ de vision restreint sur la perception de la scène. En haut : la condition normale dans laquelle la visibilité des objets est limitée à la surface apparente du volume tangible. En bas : une condition « artificielle » de comparaison dans laquelle la visibilité s'étend en dehors du volume.

d'un objet virtuel de la scène (figure 6.16), et l'objet en question doit être retrouvé et amené à l'intérieur de la cible. L'expérience se termine après trois tâches de manipulation.

Les participants sont alors invités, sans y avoir été prévenus, à indiquer le dernier emplacement de chaque objet tel qu'ils s'en souviennent. Ils disposent pour cela de vignettes qu'ils placent devant eux pour reconstituer la scène de mémoire, en observant leur main à travers la tablette s'ils le souhaitent. L'objectif de cette procédure est d'évaluer à quel point les participants se souviennent de la configuration de la scène, en comparant les positions rapportées aux positions réelles des objets à la fin de l'expérience. Les tâches de manipulation servent quant à elles à reproduire une utilisation normale de l'interface, alternant des phases de navigation (pour trouver les cibles) et de manipulation.

Le nombre d'objets virtuels dans la scène (6) a été choisi pour être suffisamment grand pour évaluer la mémoire spatiale, mais pas trop élevé ce qui aurait rendu la scène trop difficile à reconstituer. Les objets eux-mêmes, avec leurs formes et couleurs distinctes, ont été sélectionnés pour être faciles à reconnaître et à mémoriser. Les objets virtuels sont initialement positionnés de manière apparemment aléatoire, mais en réalité prédéfinie (identique pour chaque participant).

Trois cibles apparaissent au cours de l'expérience, correspondant à trois objets virtuels différents. Ainsi, certains objets sont manipulés et d'autre non, de manière à simuler une utilisation typique de l'interface. Ces trois cibles apparaissent à des emplacements en apparence aléatoires, mais en réalité prédéfinis. Après 15 s, la première cible apparaît. Lorsque la

première tâche de manipulation est accomplie, la deuxième cible apparaît après 10 s. Après la fin de la deuxième tâche de manipulation, la troisième cible apparaît au bout de 20 s. Enfin, à l'issue de la troisième tâche, l'écran de la tablette devient noir avec un message invitant les participants à retrouver la dernière position de chaque objet. Pour cela, les participants placent des vignettes aux emplacements (dans le monde réel) qu'ils pensent être ceux des objets virtuels. L'écran de la tablette est réactivé en tant que simple « fenêtre transparente » pour aider les participants à positionner les vignettes dans le même contexte visuel.

Afin d'évaluer la perception spatiale avec le volume tangible, nous l'avons comparé à une situation artificielle dans laquelle les objets virtuels seraient visibles *en dehors* du volume (figure 6.16). Pour cette condition de comparaison, nous avons simplement modifié notre prototype en réalité augmentée de sorte que les objets virtuels ne soient plus limités à la surface apparente du cube. Environ la moitié des participants (17) ont passé cette expérience dans la condition « champ de vision restreint », dans laquelle la visibilité des objets virtuels était limitée à la surface du volume tangible. Les autres (15) ont passé l'expérience dans la condition de comparaison « champ de vision large », dans laquelle la visibité s'étendait en dehors du volume pendant toute l'expérience. Deux participants additionnels ont dû être éliminés de la condition de comparaison en raison de réponses inexploitables.

Pour quantifier la correspondance entre les positions indiquées par les participants et la véritable position des objets virtuels, nous nous servons de la métrique de *Distance* proposée par Sharlin *et al.* [2009] :

$$\text{Distance} = 1 - \sum_i \left(\| \text{pos}_i^{\text{mém}} - \text{pos}_i^{\text{réel}} \| \,/\, \text{dist}_{\max} \right) / \, i_{\max}$$

$\text{pos}_i^{\text{mém}}$	position mémorisée (indiquée par le participant) de l'objet i
$\text{pos}_i^{\text{réel}}$	position réelle de l'objet i à la fin de l'expérience
$\text{dist}_{\max}$	distance maximale possible entre une position mémorisée et une position réelle, soit la diagonale de la zone accessible pour la manipulation (ici 44 cm)
$i_{\max}$	nombre total d'objets (6)

6.6.3 Résultats et discussion

La figure 6.17 montre les scores de correspondance calculés à partir des réponses des participants, groupées par condition. Le score moyen dans la condition « champ de vision restreint » (limité au volume tangible) est de 0,845 (σ=0,080). Le score moyen dans la condition « champ de vision large » (s'étendant au-delà du volume) est de 0,869 (σ=0,057).

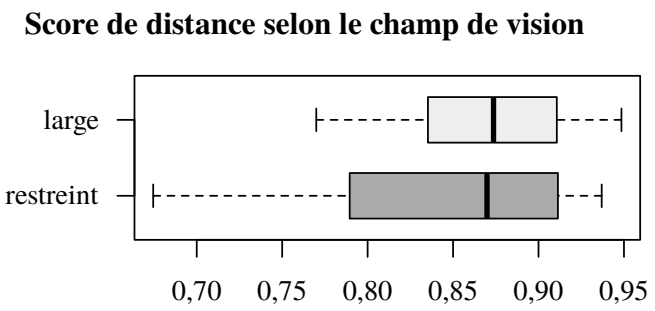


FIGURE 6.17 — Scores de correspondance entre les positions indiquées de mémoire par les utilisateurs et la véritable position des objets virtuels, d'après la métrique de distance introduite par Sharlin *et al.* [2009]. Les barres verticales représentent les médianes de chaque groupe.

Bien que quelques scores inférieurs apparaissent dans la condition « champ de vision restreint », les résultats globaux ne semblent pas être très différents entre les deux conditions. Pour confirmer cette impression, nous avons utilisé un test *t* à deux échantillons (*two sample t-test*) pour comparer les moyennes des deux conditions. Les résultats de ce test (t(30)=0,99 ; p=0,32) traduisent une différence entre les deux moyennes située dans un intervalle de [-0,075, 0,026] (à 95 %). Par conséquent, nous pouvons considérer qu'aucune différence pratique entre les deux conditions n'a été observée dans cette expérience.

Il apparaît donc que le champ de vision « restreint » n'a pas désavantagé les participants par rapport à la condition de comparaison. Ce résultat suggère qu'avoir un retour visuel limité à la surface du volume tangible ne présente pas d'inconvénient pour la perception de la scène virtuelle. Il s'agit d'un point important en faveur de la portabilité et de l'intégration, car cela signifie qu'il est possible de se passer d'une surface d'affichage externe même dans le cas d'une manipulation co-localisée.

Ce résultat confirme également ce qui a été observé précédemment par Hwang *et al.* [2006] sur des appareils mobiles classiques : le fait de déplacer l'appareil dans la scène virtuelle durant la manipulation peut compenser la limitation du champ de vision. Fait intéressant, certains participants dans la condition « champ de vision restreint » ont spontanément pensé à rapprocher le volume tangible de leurs yeux pour obtenir un plus grand champ de vision. Si notre implémentation avait été complète, nous aurions pu nous attendre à observer ce comportement. Mais la présence de la tablette, servant à simuler les écrans dans notre prototype préliminaire, n'a pas dissuadé certains participants d'employer cette stratégie.

6.7 Conclusion

Nous avons présenté dans ce chapitre une nouvelle approche d'interface, le « volume tangible », permettant une manipulation 3D sur 6 DdL à travers un unique dispositif portable et non-intrusif. À la différence du chapitre précédent, l'approche décrite ici offre en plus une co-localisation de la manipulation, ce qui simplifie grandement le mapping entre le dispositif d'interaction et les objets manipulés. La manipulation au travers du volume tangible se rapproche ainsi de celle que permettent les interfaces à base d'objets tangibles (chapitre 3 et 4), mais avec l'avantage supplémentaire pour l'interface d'être intégrée dans un unique dispositif sans éléments externes, ce qui améliore sa portabilité.

Sur la base de cette approche, nous avons proposé une métaphore de « saisie » d'objets visant à reproduire la manipulation d'objets réels (tangibles). Une première étude d'usabilité a montré que cette métaphore était immédiatement comprise par plus de la moitié des participants, et que la manipulation elle-même ne posait pas de difficulté une fois l'objet saisi. Une seconde étude d'usabilité nous a permis de vérifier que placer l'affichage sur le support de manipulation, malgré un champ de vision réduit causé par son éloignement, ne présentait pas non plus de problème pour la perception spatiale de la scène virtuelle.

L'approche du volume tangible présente pourtant un inconvénient important par rapport aux interfaces que nous avons étudiées jusqu'ici : elle nécessite le développement d'un nouveau type de dispositif, différent d'un appareil mobile classique. Cette approche répond donc moins bien à notre critère d'accessibilité. Afin d'évaluer l'effort nécessaire pour rendre un tel dispositif accessible au grand public, nous avons examiné en détail sa faisabilité technique. Il apparaît que l'implémentation de ce concept demeure complexe à l'heure actuelle, mais que les obstacles technologiques semblent pouvoir être levés à court ou moyen terme. Le principal défi reste le suivi 3D, tant celui du point de vue de l'utilisateur que celui de l'environnement, qu'il faut pouvoir effectuer depuis un unique dispositif portable et non-intrusif.

Conclusion

Au cours de cette thèse, nous avons proposé plusieurs approches visant à répondre aux inconvénients des systèmes existants pour la manipulation 3D. Ces inconvénients conduisent en effet les utilisateurs à préférer des systèmes peu adaptés à la manipulation 3D, au détriment d'interfaces plus avancées mais aussi plus contraignantes. Nous nous sommes ainsi concentrés sur quatre objectifs : améliorer la portabilité de l'interface, limiter son intrusivité, favoriser son accessibilité, tout en continuant de permettre une manipulation efficace.

L'interface proposée dans le chapitre 3, combinant un appareil mobile avec des objets tangibles, représente une première approche pour répondre simultanément à ces quatre objectifs. L'appareil mobile comme les objets tangibles sont des dispositifs fondamentalement portables, non-intrusifs et accessibles. Cependant, la principale limitation de la plate-forme mobile (reposant sur l'interaction tactile) est son manque d'efficacité pour la manipulation 3D. La combinaison d'un appareil mobile avec des objets tangibles permet de pallier à cette limitation, ce que confirment les études expérimentales que nous avons menées, répondant ainsi à notre dernier critère. Cette nouvelle configuration pose toutefois de nouveaux défis. L'objectif de portabilité interdit l'utilisation d'un système de suivi 3D externe. L'objectif d'accessibilité exclut certaines technologies telles que le suivi magnétique. Le suivi optique *inside-out* est rendu plus difficile par l'absence de capteurs externes et par la présence des doigts de l'utilisateur sur les objets tangibles durant la manipulation. Pour cette raison, nous décrivons dans le chapitre 3 une solution basée sur plusieurs marqueurs couvrant toute la surface des objets tangibles, permettant de maintenir un suivi acceptable malgré les contraintes liées à la portabilité et à l'accessibilité.

Un autre défi que pose l'approche ci-dessus est la difficulté d'intégrer un retour de force dans les objets tangibles, sans passer par des dispositifs actifs généralement intrusifs, non portables et onéreux. Nous avons donc proposé dans le chapitre 4 une nouvelle technique appelée « découplage », basée sur le principe du « pseudo-haptique », capable de recréer une illusion de force à travers de simples objets tangibles passifs. Nous avons montré expérimentalement que notre technique permet effectivement aux utilisateurs de différencier des masses virtuelles, sans le moindre apprentissage préalable, avec un taux de succès très nettement supérieur à la condition de contrôle. Afin de mieux caractériser les performances de notre technique, nous avons également mesuré le seuil différentiel entre deux forces virtuelles perceptible par l'utilisateur. Cette technique améliore la richesse du retour d'information de

TABLEAU 6.1 — Avantages et inconvénients des approches proposées dans cette thèse par rapport aux quatre critères établis initialement. Bien que chacune de ces approches représente un progrès par rapport aux principaux systèmes actuels (comparer avec le tableau 2.1), on notera qu'aucune ne répond *entièrement* aux quatre objectifs à la fois.

	efficacité	**portabilité**	**non-intrusivité**	**accessibilité**
Appareil mobile et objets tangibles (chapitre 3)	**++**	+	**++**	**++**
Appareil mobile seul (chapitre 5)	+	**++**	**++**	**++**
Volume tangible (chapitre 6)	**++**	**++**	**++**	–

la manipulation 3D sans recourir à des dispositifs actifs à retour de force, permettant ainsi de se passer des contraintes associées.

L'approche combinant un appareil mobile avec des objets tangibles présente cependant une autre limitation : l'interface n'est pas aussi portable qu'elle pourrait l'être. En effet, le fait d'être constituée de plusieurs éléments, l'appareil mobile et les objets tangibles, oblige l'utilisateur à toujours transporter ces éléments ensemble. La perte ou l'oubli de l'un d'entre eux suffit à rendre l'interface non fonctionnelle. La manipulation de plusieurs éléments nécessite de plus d'utiliser les deux mains, ce qui peut mener à des difficultés ergonomiques (longueur des bras, etc.). Nous avons donc choisi dans le chapitre 5 de nous intéresser à une approche différente : considérer l'appareil mobile lui-même comme support de manipulation tangible. Cette solution permet à l'interface d'être entièrement intégrée dans un seul et même dispositif, améliorant donc sa portabilité. Pourtant, ce gain en matière de portabilité n'est pas sans contreparties. La surface d'affichage étant désormais confondue avec le support de manipulation (« couplage local »), il ne suffit plus de seulement transférer les mouvements du support sur l'objet 3D manipulé. Il devient nécessaire de prendre en compte la *composition* des mouvements, afin que les déplacements de l'objet manipulé sur l'écran correspondent bien aux attentes des utilisateurs. Les déplacements importants doivent également faire l'objet d'une attention particulière, afin d'éviter que l'écran de l'appareil mobile ne sorte du champ de vision en pleine manipulation. Apporter une réponse à ces questions implique d'agir sur la fonctions de transfert, ou « *mapping* », entre les mouvements de l'appareil mobile et ceux des objets manipulés. Nous avons pour cela conduit une analyse détaillée des mappings possibles et et leurs propriétés respectives. Cependant, cette analyse a permis de montrer qu'aucun mapping unique ne peut répondre parfaitement à toutes les questions que soulève cette configuration. Il s'avère donc indispensable de choisir quelles propriétés privilégier et quelles propriétés abandonner. Cette configuration localement couplée améliore donc la portabilité par rapport à l'approche introduite dans le chapitre 3, mais pose en contrepartie certains problèmes de manipulation.

Afin de retrouver les avantages de la manipulation par objets tangibles, tout en conservant l'intérêt en terme de portabilité d'une interface intégrée dans un unique dispositif, nous avons proposé dans le chapitre 6 une dernière approche : le « volume tangible ». Celle-ci consiste à recouvrir un objet tangible d'une surface d'affichage. Il devient ainsi possible d'interagir *à travers* cet objet avec un volume équivalent de la scène virtuelle. La présence d'une surface d'affichage sur l'objet tangible lui-même permet de se passer d'un appareil mobile séparé, et donc d'avoir un système intégré dans un unique dispositif. Mais contrairement à l'approche précédente, cette solution permet également de positionner le support de manipulation *sur* les objets 3D (co-localisation), ce qui rapproche cette forme de manipulation de celle du chapitre 3. Nous avons ainsi proposé une technique de saisie d'objets consistant à positionner le volume tangible sur l'objet virtuel souhaité et à presser les doigts pour le saisir, ce qui permet de le manipuler sur 6 DdL comme s'il était directement tenu dans la main. Nous avons montré expérimentalement que cette métaphore de saisie pouvait être comprise par une majorité d'utilisateurs sans instructions préalables, et que le champ de vision restreint dû à la co-localisation ne constituait pas un inconvénient. Cependant, si le volume tangible semble répondre particulièrement bien aux objectifs d'efficacité, de portabilité et de non-intrusivité, le principal inconvénient de cette approche reste sa moindre accessibilité. En effet, il n'existe pas aujourd'hui d'appareil mobile accessible au grand public disposant de toutes les capacités nécessaires à l'implémentation d'une telle interface. Nous nous sommes donc attachés dans le chapitre 6 à examiner la faisabilité technique d'un tel dispositif. Cette analyse nous a permis de constater que les obstacles à sa fabrication (et donc à sa mise à disposition pour le grand public) existent bel et bien, mais paraissent surmontables à court ou moyen terme.

En mettant spécifiquement l'accent sur les quatre critères d'efficacité, de portabilité, de non-intrusivité et d'accessibilité, les approches introduites dans cette thèse représentent toutes une amélioration par rapport aux systèmes actuels. Cependant, aucune d'entre elles ne répond *entièrement* à ces quatre objectifs, comme le montre le tableau 6.1. Chacune de ces approches représente en effet un compromis différent. La combinaison d'un appareil mobile avec des objets tangibles (chapitre 3) n'est pas aussi portable qu'elle devrait l'être. L'utilisation d'un appareil mobile seul (chapitre 5) améliore cette portabilité, mais souffre de certains problèmes de manipulation. Le volume tangible (chapitre 6) répond à ces deux inconvénients mais au prix d'une moindre accessibilité, du moins en l'état actuel des technologies.

Ce travail ouvre donc un certain nombre de perspectives. La plus importante d'entre elles porte, bien entendu, sur la capacité à pouvoir répondre simultanément à nos quatre objectifs. Les différentes approches introduites dans cette thèse semblent à chaque fois pallier aux limitations des autres solutions en renonçant partiellement à un autre objectif. Il est donc légitime de se demander s'il serait possible de concevoir une interface réunissant chacun des avantages des approches proposées, sans pour autant en sacrifier certains aspects.

Cette question peut être abordée de deux manières. La première serait de continuer à imaginer de nouvelles approches, si possible radicalement différentes de celles proposées ici. En effet, les compromis successifs que nous avons dû réaliser pourraient n'être qu'une conséquence du fait que chaque approche s'inspire de la précédente. Élargir la variété des approches considérées, tout en maintenant nos objectifs de portabilité, de non-intrusivité, d'accessibilité et d'efficacité, augmenterait les opportunités d'arriver à une solution répondant pleinement à ces quatre critères.

Une deuxième manière de répondre à cette question serait de tenter d'améliorer les approches proposées ici, afin de combler leurs points faibles respectifs. De ce point de vue, chacune des interfaces que nous avons présentées constitue une perspective. Ainsi, l'une des principales limitations de l'approche combinant un appareil mobile avec des objets tangibles est d'être constituée de plusieurs éléments distincts, ce qui nuit à sa portabilité. Une piste d'amélioration pour conserver intact le principe de cette approche (avec ses autres avantages) mais sans avoir à transporter des objets tangibles dédiés, serait de tirer parti des objets présents dans l'environnement. En supposant que l'appareil mobile soit capable de scanner, puis de reconnaître et suivre en 3D un objet quelconque pris dans l'environnement, il deviendrait possible de transformer provisoirement tout objet en « poignée tangible » permettant une manipulation 6 DdL. Cette solution, qui reste à implémenter, pourrait en théorie combler le principal inconvénient de cette approche et ainsi aboutir à une interface répondant mieux à l'ensemble de nos critères.

Le principal obstacle de l'approche par appareil mobile seul est celui de ses mappings de manipulation, plus complexes et indirects que dans la manipulation par objets tangibles. Une telle complexité accrue est inhérente à cette configuration localement couplée, et le choix du mapping doit se faire en premier lieu en fonction des propriétés spatiales souhaitées, suivant les recommandations que nous avons établies dans le chapitre 5. Pourtant, les paramètres du mapping lui-même restent à spécifier, et peuvent avoir une importante influence sur l'usabilité. Une des pistes d'amélioration de cette approche serait donc de mener de nouvelles études expérimentales ayant pour but de déterminer les valeurs à donner à ces paramètres. Ainsi, l'étendue exacte de l'espace de manipulation (section 5.8.1) pourrait être mesurée expérimentalement afin d'optimiser les mappings pour chaque type d'appareil mobile. Les deux fonctions de gain proposées dans la section 5.8.2, bien qu'elles s'inspirent toutes deux de métaphores plausibles, pourraient également être comparées expérimentalement pour déterminer laquelle est la plus facilement comprise par les utilisateurs. Enfin, il se pourrait que certains mappings en translation se combinent mieux avec des mappings *différents* en rotation (par exemple un contrôle en position relatif pour les translations et un contrôle en vitesse pour les rotations). En menant de nouvelles études visant spécifiquement à optimiser les mappings de manipulation, il serait possible de réduire l'impact de l'unique inconvénient de cette approche.

Le volume tangible répond presque entièrement à nos objectifs, à l'exception notable de son manque d'accessibilité : le dispositif lui-même n'existe pas en l'état à l'heure actuelle. Pour que cette approche puisse effectivement bénéficier au plus grand nombre, il est donc nécessaire de travailler sur l'implémentation matérielle de ce concept. Comme nous l'avons montré dans le chapitre 6, un certain nombre des technologies requises existent déjà et ne demandent qu'à être assemblées dans un dispositif intégré. Cependant, un défi important reste à surmonter : le suivi 3D, aussi bien celui du point de vue que celui de l'environnement. Comme dans nos autres approches, ce suivi est contraint par les critères que nous nous sommes fixés. Il doit être intégré au dispositif afin de ne pas dépendre d'installations extérieures non-portables, et ne doit pas obliger l'utilisateur à porter d'équipement intrusif. Mais à la différence d'une approche par objets tangibles externes (chapitre 3), il n'est pas possible ici de se baser sur des marqueurs connus qui pourraient faciliter le suivi. À la différence de l'approche par appareil mobile seul (chapitre 5), la présence d'une surface d'affichage entourant la totalité du dispositif complexifie nettement l'intégration de capteurs, alors même que ceux-ci devraient être présents en plus grand nombre pour couvrir toutes les directions. Résoudre ces difficultés lèverait le principal obstacle à l'implémentation d'un tel dispositif, ce qui permettrait au volume tangible de répondre correctement à l'ensemble de nos objectifs. Même si une réponse sera sans doute apportée à terme par le développement des technologies de suivi, ce n'est cependant pas la seule manière d'aborder cette question. Nous pourrions en effet chercher à préciser, par le biais d'études expérimentales, le degré de tolérance des utilisateurs à un suivi 3D « imparfait ». Par exemple, il serait intéressant de déterminer à quel point une discordance entre le rendu *fish-tank* et la position réelle de la tête de l'utilisateur devient perturbante, ou encore à quel point une dérive dans le suivi de l'environnement durant la manipulation devient perceptible. Ces informations pourraient potentiellement autoriser l'utilisation de capteurs moins performants mais plus facilement accessibles, car déjà existants ou moins onéreux à développer. Déterminer la tolérance des utilisateurs à des erreurs de suivi aurait par ailleurs un intérêt allant au-delà du seul volume tangible. En réduisant les contraintes technologiques liées au suivi 3D, ces informations pourraient faciliter la conception de systèmes portables, non-intrusifs et accessibles en général.

Les approches que nous avons proposées dans cette thèse, malgré leurs limitations, apportent d'ores et déjà une réponse à la sous-utilisation actuelle des systèmes dédiés à la manipulation 3D. Que ce soit par une portabilité accrue, une moindre intrusivité, ou l'utilisation de matériel plus accessible, tout en conservant une bonne efficacité de manipulation, ces interfaces représentent un progrès par rapport aux systèmes actuels. Chacune d'entre elles constitue un compromis différent sur ces quatre objectifs, offrant ainsi plusieurs alternatives selon l'importance relative que l'on accorde à chaque critère. Les nouvelles approches présentées dans cette thèse devraient donc contribuer à favoriser l'adoption par les utilisateurs d'interfaces plus performantes pour leurs tâches quotidiennes.

Annexe A

Démonstrations

A.1 Compatibilité directionnelle du mapping absolu

Dans un mapping de type contrôle en position absolu (équation 5.1), et en prenant en compte le facteur de gain k_t, nous pouvons exprimer les translations de l'objet manipulé entre $t-1$ et t de la manière suivante :

$$\begin{aligned} pd_t - pd_{t-1} &= (k_t\ \Delta pc_t + pd_0) - (k_{t-1}\ \Delta pc_{t-1} + pd_0) \\ &= k_t\ \Delta pc_t - k_{t-1}\ \Delta pc_{t-1} \\ &= k_t \left(qc_0^{-1}\,(pc_t - pc_0)\,qc_0\right) - k_{t-1}\left(qc_0^{-1}\,(pc_{t-1} - pc_0)\,qc_0\right) \\ &= \left(qc_0^{-1}\,k_t\,(pc_t - pc_0)\,qc_0\right) - \left(qc_0^{-1}\,k_{t-1}\,(pc_{t-1} - pc_0)\,qc_0\right) \\ &= qc_0^{-1}\left(k_t\,(pc_t - pc_0) - k_{t-1}\,(pc_{t-1} - pc_0)\right)qc_0 \end{aligned} \tag{A.1}$$

Cette expression n'étant pas équivalente à l'équation 5.4, les translations de l'objet dans ce mapping ne présentent donc pas de compatibilité directionnelle dans le cas général. Mais si le facteur de gain k_t est supposé constant ($k_t = \alpha\ \ \forall t > 0$), alors l'expression A.1 peut être simplifiée ainsi :

$$\begin{aligned} &= qc_0^{-1}\left(\alpha\,(pc_t - pc_0) - \alpha\,(pc_{t-1} - pc_0)\right)qc_0 \\ &= qc_0^{-1}\ \alpha\left((pc_t - pc_0) - (pc_{t-1} - pc_0)\right)qc_0 \\ &= qc_0^{-1}\ \alpha\,(pc_t - pc_{t-1})\,qc_0 \\ &= \alpha\left(qc_0^{-1}\,(pc_t - pc_{t-1})\ qc_0\right) \end{aligned} \tag{A.2}$$

Si, par ailleurs, l'orientation de l'appareil mobile à $t-1$ (c'est-à-dire qc_{t-1}) est égale à son orientation initiale qc_0, alors l'expression ci-dessus peut être réécrite de la façon suivante :

$$= \alpha\left(qc_{t-1}^{-1}\,(pc_t - pc_{t-1})\,qc_{t-1}\right)$$

Nous retrouvons alors l'équation 5.4 qui définit la compatibilité directionnelle. Les translations dans un mapping absolu ne présentent donc cette propriété que si le facteur de gain est demeuré constant entre $t{=}0$ et $t{-}1$ *et* si l'orientation actuelle de l'appareil mobile est égale à son orientation initiale.

Les rotations incrémentales de l'objet manipulé peuvent quant à elles être exprimées ainsi :

$$\begin{aligned} qd_t\, qd_{t-1}^{-1} &= (\Delta qc_t^{k_t}\, qd_0)\,(\Delta qc_{t-1}^{k_{t-1}}\, qd_0)^{-1} \\ &= (\Delta qc_t^{k_t}\, qd_0)\,(qd_0^{-1}\, \Delta qc_{t-1}^{-k_{t-1}}) \\ &= \Delta qc_t^{k_t}\, \Delta qc_{t-1}^{-k_{t-1}} \\ &= \left(qc_0^{-1}\,(qc_t\, qc_0^{-1})\, qc_0\right)^{k_t} \left(qc_0^{-1}\,(qc_{t-1}\, qc_0^{-1})\, qc_0\right)^{-k_{t-1}} \end{aligned} \tag{A.3}$$

Dans le cas général, cette expression n'est pas équivalente à l'équation 5.4 et les rotations ne présentent donc pas de compatibilité directionnelle. Cependant, si $qc_t\, qc_0^{-1}$ (la rotation de l'appareil mobile au temps t par rapport à son orientation initiale) et $qc_{t-1}\, qc_0^{-1}$ (même chose au temps $t{-}1$) partagent le même axe de rotation, alors leur différence $qc_t\, qc_{t-1}^{-1}$ a elle aussi le même axe. Dans ce cas, il existe deux facteurs de gain a_1 et a_2 tels que $qc_t\, qc_0^{-1} = (qc_t\, qc_{t-1}^{-1})^{a_1}$ et $qc_{t-1}\, qc_0^{-1} = (qc_t\, qc_{t-1}^{-1})^{a_2}$, et l'expression A.3 peut être réécrite de la façon suivante :

$$\begin{aligned} &= \left(qc_0^{-1}\,(qc_t\, qc_{t-1}^{-1})^{a_1}\, qc_0\right)^{k_t} \left(qc_0^{-1}\,(qc_t\, qc_{t-1}^{-1})^{a_2}\, qc_0\right)^{-k_{t-1}} \\ &= \left(qc_0^{-1}\,(qc_t\, qc_{t-1}^{-1})\, qc_0\right)^{a_1+k_t} \left(qc_0^{-1}\,(qc_t\, qc_{t-1}^{-1})\, qc_0\right)^{a_2-k_{t-1}} \\ &= \left(qc_0^{-1}\,(qc_t\, qc_{t-1}^{-1})\, qc_0\right)^{a_1+k_t+a_2-k_{t-1}} \end{aligned}$$

En posant $a_1 + k_t + a_2 - k_{t-1} \equiv \beta$, nous pouvons simplifier l'expression ci-dessus :

$$= \left(qc_0^{-1}\,(qc_t\, qc_{t-1}^{-1})\, qc_0\right)^{\beta} \tag{A.4}$$

Si $qc_{t-1}\, qc_0^{-1} = (qc_t\, qc_{t-1}^{-1})^{a_2}$, alors $qc_0^{-1} = qc_{t-1}^{-1}\,(qc_t\, qc_{t-1}^{-1})^{a_2}$, ce qui permet de réécrire l'expression A.4 ainsi :

$$\begin{aligned} &= \left(qc_{t-1}^{-1}\,(qc_t\, qc_{t-1}^{-1})^{a_2}\,(qc_t\, qc_{t-1}^{-1})\left(qc_{t-1}^{-1}\,(qc_t\, qc_{t-1}^{-1})^{a_2}\right)^{-1}\right)^{\beta} \\ &= \left(qc_{t-1}^{-1}\,(qc_t\, qc_{t-1}^{-1})^{a_2}\,(qc_t\, qc_{t-1}^{-1})\,(qc_t\, qc_{t-1}^{-1})^{-a_2}\, qc_{t-1}\right)^{\beta} \\ &= \left(qc_{t-1}^{-1}\,(qc_t\, qc_{t-1}^{-1})^{a_2+1-a_2}\, qc_{t-1}\right)^{\beta} \\ &= \left(qc_{t-1}^{-1}\,(qc_t\, qc_{t-1}^{-1})\, qc_{t-1}\right)^{\beta} \end{aligned}$$

Nous retrouvons là encore l'équation 5.4. Ainsi, les rotations dans un mapping absolu ne présentent de compatibilité directionnelle qu'autour de l'axe de rotation initial. En pratique, cela signifie que seule la première rotation est compatible, les rotations suivantes ne le sont plus dès lors qu'elles ont lieu sur un axe différent. Par conséquent, un utilisateur souhaitant faire

tourner l'objet manipulé autour d'un axe différent doit au préalable ramener l'appareil mobile à son orientation initiale pour continuer à bénéficier de la compatibilité directionnelle.

Un cas particulier apparaît toutefois dans ce mapping lorsque le facteur de gain k_t est constant et égal à -1, qui permet de garantir la compatibilité directionnelle dans tous les cas (voir section 5.6.3).

A.2 Compatibilité directionnelle du contrôle en vitesse

Dans un mapping de type contrôle en vitesse (équation 5.3), et en prenant en compte le facteur de gain k_t, nous pouvons exprimer les translations de l'objet manipulé entre $t-1$ et t de la manière suivante :

$$\begin{aligned} pd_t - pd_{t-1} &= (k_t \Delta pc_t + pd_{t-1}) - pd_{t-1} \\ &= k_t \Delta pc_t \\ &= k_t \left(qc_0^{-1} (pc_t - pc_0)\, qc_0 \right) \end{aligned} \tag{A.5}$$

Cette expression n'étant pas équivalente à l'équation 5.4, les translations de l'objet dans ce mapping ne présentent donc pas de compatibilité directionnelle dans le cas général. Cependant, si l'on fait l'hypothèse que les translations de l'appareil mobile s'effectuent le long du même axe depuis le début de la manipulation, alors $pc_t - pc_{t-1}$ est colinéaire avec $pc_t - pc_0$. Il existe donc un scalaire a tel que $pc_t - pc_0 = a\,(pc_t - pc_{t-1})$, ce qui permet de réécrire l'expression A.5 de la façon suivante :

$$\begin{aligned} &= k_t \left(qc_0^{-1} \left(a\,(pc_t - pc_{t-1}) \right) qc_0 \right) \\ &= (a\, k_t) \left(qc_0^{-1} (pc_t - pc_{t-1})\, qc_0 \right) \\ &= \alpha \left(qc_0^{-1} (pc_t - pc_{t-1})\, qc_0 \right) \end{aligned}$$

Nous retrouvons l'équivalent de l'expression A.2, ce qui permet d'aboutir à l'équation 5.4 si l'on suppose que l'orientation de l'appareil mobile à chaque instant $t-1$ est égale à son orientation initiale qc_0 (cf. démonstration précédente). Les translations présentent donc uniquement une compatibilité directionnelle le long de l'axe initial de déplacement de l'appareil mobile *et* en supposant que l'orientation actuelle de l'appareil mobile reste égale à son orientation initiale.

Les rotations présentent la même propriété :

$$\begin{aligned} qd_t\, qd_{t-1}^{-1} &= (\Delta qc_t^{k_t}\, qd_{t-1})\, qd_{t-1}^{-1} \\ &= \Delta qc_t^{k_t} \\ &= \left(qc_0^{-1} (qc_t\, qc_0^{-1})\, qc_0 \right)^{k_t} \end{aligned} \tag{A.6}$$

Dans le cas général, cette expression n'est pas équivalente à celle de l'équation 5.4. Mais si l'on suppose que les rotations de l'appareil mobile se font autour du même axe depuis le début de la manipulation, alors il existe un scalaire b tel que $qc_t\, qc_0^{-1} = (qc_t\, qc_{t-1}^{-1})^b$, ce qui permet de réécrire l'expression A.6 ainsi :

$$\begin{aligned}
&= \left(qc_0^{-1}\, (qc_t\, qc_{t-1}^{-1})^b\, qc_0\right)^{k_t} \\
&= \left(qc_0^{-1}\, (qc_t\, qc_{t-1}^{-1})\, qc_0\right)^{k_t+b} \\
&= \left(qc_0^{-1}\, (qc_t\, qc_{t-1}^{-1})\, qc_0\right)^{\beta}
\end{aligned}$$

Nous retrouvons alors l'équivalent de l'expression A.4, qui permet d'aboutir à l'équation 5.4 puisque nous avons fait l'hypothèse que les rotations de l'appareil ont toutes eu lieu autour du même axe (cf. démonstration précédente). Comme dans le cas du mapping absolu, les rotations présentent donc une compatibilité directionnelle uniquement autour de l'axe de rotation initial de l'appareil mobile.

A.3 Transitivité du mapping relatif

Cette section examine les conditions particulières pouvant permettre à un mapping de type contrôle en position relatif (équation 5.2) de présenter la propriété de transitivité.

Soit pd_t^{incr} la position de l'objet manipulé résultant du passage de l'appareil mobile par plusieurs positions intermédiaires entre pc_0 et pc_t. Soit pd_t^{direct} la position de l'objet résultant d'un déplacement *direct* de l'appareil de pc_0 à $pc_1 = pc_t$. De même, qd_t^{incr} représente l'orientation de l'objet manipulé après plusieurs étapes de rotation de l'appareil entre qc_0 et qc_t, tandis que qd_t^{direct} représente l'orientation de l'objet après une rotation directe de l'appareil de qc_0 à $qc_1 = qc_t$.

Dans le cas des positions, nous avons :

$$\begin{aligned}
pd_t^{\text{incr}} &= k_t\, \left(qc_{t-1}^{-1}\, (pc_t - pc_{t-1})\, qc_{t-1}\right) + pd_{t-1} \\
&= \left(qc_{t-1}^{-1}\, k_t\, (pc_t - pc_{t-1})\, qc_{t-1}\right) + pd_{t-1} \\
&= \left(qc_{t-1}^{-1}\, k_t\, (pc_t - pc_{t-1})\, qc_{t-1}\right) \\
&\quad + \ldots + \left(qc_0^{-1}\, k_1\, (pc_1 - pc_0)\, qc_0\right) + pd_0 \\
pd_t^{\text{direct}} &= k_t\, \left(qc_0^{-1}\, (pc_t - pc_0)\, qc_0\right) + pd_0 \\
&= qc_0^{-1}\, k_t\, (pc_t - pc_0)\, qc_0 + pd_0
\end{aligned}$$

À cause des transformations (conversions dans le repère écran et facteur de gain) appliquées aux translations incrémentales $pc_i - pc_{i-1}$ dans pd_t^{incr}, le résultat n'est pas équivalent à pd_t^{direct}. Cela montre que les translations ne sont pas transitives dans un tel mapping dans

le cas général. Cependant, si l'orientation de l'appareil mobile n'a pas changé depuis le début de la manipulation ($qc_t = qc_0 \;\; \forall t{>}0$), alors pd_t^{incr} peut être simplifié ainsi :

$$\begin{aligned} pd_t^{\text{incr}} = qc_0^{-1} \big(& k_t \, (pc_t - pc_{t-1}) \\ & + \ldots + k_1 \, (pc_1 - pc_0) \big) \, qc_0 + pd_0 \end{aligned} \tag{A.7}$$

L'expression A.7 applique des facteurs de gain arbitraires ($k_1, k_2, \ldots, k_t$) à chaque étape de translation. Pour que l'expression A.7 soit équivalente à pd_t^{direct}, il est nécessaire que ces facteurs de gain soient tous identiques ($k_i{=}k_t \;\; \forall i{<}t$) :

$$\begin{aligned} pd_t^{\text{incr}} &= qc_0^{-1} \, k_t \, (pc_t - pc_0) \, qc_0 + pd_0 \\ &= pd_t^{\text{direct}} \end{aligned}$$

Par conséquent, les translations dans un mapping relatif ne sont transitives qu'à condition que le facteur de gain *et* l'orientation de l'appareil mobile soient restés constants depuis le début de la manipulation.

Dans le cas des rotations, nous avons :

$$\begin{aligned} qd_t^{\text{incr}} &= \big(qc_{t-1}^{-1} \, (qc_t \, qc_{t-1}^{-1}) \, qc_{t-1} \big)^{k_t} \; qd_{t-1} \\ &= \big((qc_{t-1}^{-1} \, qc_t) \, (qc_{t-1}^{-1} \, qc_{t-1}) \big)^{k_t} \; qd_{t-1} \\ &= (qc_{t-1}^{-1} \, qc_t)^{k_t} \, qd_{t-1} \\ &= (qc_{t-1}^{-1} \, qc_t)^{k_t} \, (qc_{t-2}^{-1} \, qc_{t-1})^{k_{t-1}} \ldots (qc_0^{-1} \, qc_1)^{k_1} \, qd_0 \\ qd_t^{\text{direct}} &= \big(qc_0^{-1} \, (qc_t \, qc_0^{-1}) \, qc_0 \big)^{k_t} \; qd_0 \\ &= (qc_0^{-1} \, qc_t)^{k_t} \, qd_0 \end{aligned}$$

Même si le facteur de gain restait constant, qd_t^{incr} ne serait toujours pas équivalent à qd_t^{direct} dans le cas général. Les rotations ne sont donc pas transitives dans un mapping relatif (sauf lorsque $k_t = -1$, un cas particulier traité dans la section 5.6.3).

A.4 Effets d'un facteur de gain constant égal à –1

Cette valeur a une signification particulière dans le cas d'un mapping de type contrôle en position. En effet, elle rend les rotations *équivalentes* dans les mappings absolus et relatifs. Pour un facteur de gain constant $k_t = -1 \;\; \forall t{>}0$, un mapping absolu produit les orientations

suivantes :

$$\begin{aligned}
qd_t &= \Delta qc_t^{-1}\, qd_0 \\
&= \left(qc_0^{-1}\,(qc_t\, qc_0^{-1})\, qc_0\right)^{-1}\, qd_0 \\
&= \left(qc_0^{-1}\,(qc_t\, qc_0^{-1})^{-1}\, qc_0\right)\, qd_0 \\
&= \left(qc_0^{-1}\,(qc_0\, qc_t^{-1})\, qc_0\right)\, qd_0 \\
&= (qc_t^{-1}\, qc_0)\, qd_0
\end{aligned}$$

Et un mapping relatif produit :

$$\begin{aligned}
qd_t &= \Delta qc_t^{-1}\, qd_{t-1} \\
&= \left(qc_{t-1}^{-1}\,(qc_t\, qc_{t-1}^{-1})\, qc_{t-1}\right)^{-1}\, qd_{t-1} \\
&= (qc_{t-1}^{-1}\, qc_t)^{-1}\, qd_{t-1} \\
&= (qc_{t-1}^{-1}\, qc_t)^{-1}\,(qc_{t-2}^{-1}\, qc_{t-1})^{-1} \ldots (qc_0^{-1}\, qc_1)^{-1}\, qd_0 \\
&= (qc_t^{-1}\, qc_{t-1})\,(qc_{t-1}^{-1}\, qc_{t-2}) \ldots (qc_1^{-1}\, qc_0)\, qd_0 \\
&= (qc_t^{-1}\, qc_0)\, qd_0
\end{aligned}$$

Par conséquent, les rotations deviennent strictement équivalentes dans des mappings absolus et relatifs pour un facteur de gain constant $k_t = -1$. Cela implique que les deux types de mappings partagent désormais les *mêmes propriétés*. Les rotations du mapping absolu acquièrent la compatibilité directionnelle, comme dans le mapping relatif équivalent. Les rotations du mapping relatif deviennent transitives, comme dans le mapping absolu équivalent. Un facteur de gain de -1 est donc l'unique moyen de bénéficier *à la fois* de la compatibilité directionnelle et de la transitivité dans un mapping de type « contrôle en position ».

Cependant, la même équivalence n'est pas garantie pour les translations. Comme on a pu le voir dans les démonstrations précédentes, les propriétés de compatibilité spatiale des translations ne dépendent pas seulement de la position de l'appareil mobile, mais également de son orientation. Dans un mapping absolu, un facteur de gain constant $k_t = -1$ produit les positions suivantes :

$$\begin{aligned}
pd_t &= -\Delta pc_t + pd_0 \\
&= -\left(qc_0^{-1}\,(pc_t - pc_0)\, qc_0\right) + pd_0 \\
&= \left(qc_0^{-1}\,(pc_0 - pc_t)\, qc_0\right) + pd_0
\end{aligned}$$

Le vecteur $pc_0 - pc_t$ représente l'opposé de la translation totale de l'appareil mobile depuis le début de la manipulation. On pourrait penser qu'appliquer ce vecteur à l'objet manipulé aurait pour effet d'annuler la translation de l'appareil mobile, faisant paraître l'objet « fixe » par rapport à l'environnement réel. Mais ce vecteur doit préalablement être converti dans le repère de l'écran. Dans un mapping absolu, cette conversion s'effectue par rapport à

l'orientation initiale qc_0 de l'appareil mobile. L'objet manipulé ne paraîtra donc fixe que si l'orientation de l'appareil mobile se maintient à qc_0.

Dans un mapping relatif, un facteur de gain constant $k_t = -1$ produit les positions suivantes :

$$\begin{aligned} pd_t &= -\Delta pc_t + pd_{t-1} \\ &= -\left(qc_{t-1}^{-1}\left(pc_t - pc_{t-1}\right) qc_{t-1}\right) + pd_{t-1} \\ &= \left(qc_{t-1}^{-1}\left(pc_{t-1} - pc_t\right) qc_{t-1}\right) + pd_{t-1} \\ &= \left(qc_{t-1}^{-1}\left(pc_{t-1} - pc_t\right) qc_{t-1}\right) \\ &\quad + \ldots + \left(qc_0^{-1}\left(pc_0 - pc_1\right) qc_0\right) + pd_0 \end{aligned}$$

Bien que l'opposé de chaque sous-étape de translation $pc_t - pc_{t-1}$ soit appliqué à l'objet manipulé, chacune de ces sous-étapes est préalablement convertie dans le repère de l'écran selon les orientations *intermédiaires* de l'appareil mobile. La translation totale de l'objet dépend ainsi des orientations intermédiaires de l'appareil. L'objet ne paraîtra donc fixe par rapport à l'environnement que si l'orientation de l'appareil mobile ne change pas pendant la manipulation (c'est-à-dire reste égale à qc_0).

Annexe B

Résultats supplémentaires sur les mappings allocentrique et égocentrique

B.1 Influence de la familiarité avec le jeu vidéo 3D

Une question additionnelle que nous souhaitions étudier dans l'expérience présentée en section 5.7.2 est l'influence de l'utilisation de logiciels 3D sur la préférence entre les mappings allocentrique et égocentrique. Cependant, seuls 7 participants (sur un total de 30) considéraient avoir une expérience préalable avec un logiciel de modélisation 3D, ce qui est un échantillon relativement faible pour mener ce type d'analyse. D'un autre côté, 14 participants, soit près de la moitié, déclaraient jouer fréquemment à des jeux vidéo 3D. Nous avons donc choisi de nous focaliser sur l'expérience de jeu vidéo comme indicateur de familiarité avec des logiciels 3D. Sans surprise, le groupe des « joueurs » était largement corrélé avec le jeune âge et le sexe masculin. Toutefois, nous avons supposé que la familiarité avec des jeux 3D en elle-même aurait plus d'impact sur les résultats que l'âge ou le sexe des participants.

Nous avons donc mené une seconde analyse des résultats de la partie 1 de l'expérience (translations et rotations seules, cf. section 5.7.2) en séparant les participants en deux groupes : ceux déclarant utiliser de manière régulière des jeux vidéo 3D (groupe « joueurs », 14 participants) et ceux ne pratiquant que rarement ou jamais cette activité (groupe « non-joueurs », 16 participants). Même si, dans les deux groupes, le mapping préféré dans chaque condition s'avère être le même que dans la partie 1 de l'expérience, cette deuxième analyse met en évidence dans plusieurs cas des tailles d'effet sensiblement différentes (figure B.1).

Résultats

Dans la scène 1, les non-joueurs ont manifestement eu une préférence moins forte pour les translations égocentriques (r=0,14; σ_r=0,17) que les joueurs (r=0,37; σ_r=0,16). Cette scène servant de référence pour déterminer le modèle mental « de base », nous pouvons donc

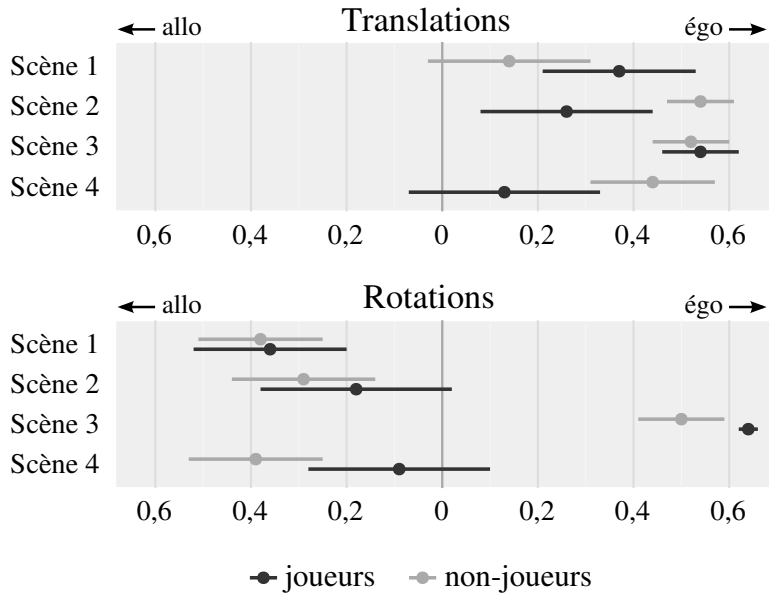


FIGURE B.1 — Tailles d'effet et erreur type pour la préférence entre les deux mappings parmi les groupes de « joueurs » et de « non-joueurs ».

observer (avec certaines réserves compte tenu de l'erreur type) que les non-joueurs semblent avoir une préférence *a priori* moins marquée pour les translations, et/ou que la familiarité avec le jeu vidéo 3D créerait un biais en faveur du mapping égocentrique. Pour ce qui est des rotations, la préférence était quasiment identique entre les joueurs (r=0,36; σ_r=0,16) et les non-joueurs (r=0,38; σ_r=0,13). Il apparaît donc que la familiarité avec le jeu vidéo 3D ne produit pas de tel biais pour les rotations.

Dans la scène 2, la préférence des joueurs pour les translations égocentriques était d'un niveau comparable à la scène 1 (r=0,26; σ_r=0,18). Cependant, la préférence des non-joueurs était bien plus marquée que dans la scène 1 (r=0,54; σ_r=0,07). Les joueurs semblent donc avoir été moins affectés que les non-joueurs par les changements dans la scène virtuelle. Cela pourrait s'expliquer par une plus grande tolérance envers l'utilisation de divers mappings dans différents environnements virtuels, avec malgré tout une préférence latente pour un mapping égocentrique. La différence entre les deux groupes était moins marquée pour les rotations, mais laisse entrevoir une tendance similaire (joueurs : r=0,18; σ_r=0,20, non-joueurs : r=0,29; σ_r=0,15). Ces résultats appuient donc l'hypothèse **H3**, mais seulement pour les non-joueurs.

Concernant la scène 3, la préférence pour les translations égocentriques était tout aussi forte chez les joueurs (r=0,54; σ_r=0,08) que chez les non-joueurs (r=0,52; σ_r=0,08). Nous pouvons donc en déduire que la configuration de la scène 3 (voir l'objet manipulé de l'intérieur) produit un effet suffisamment grand pour passer outre la tolérance présumée des joueurs envers le mapping non-préféré (allocentrique). Les rotations montrent également une forte

préférence pour le mapping égocentrique dans les deux groupes (joueurs : r=0,64; σ_r=0,02, non-joueurs : r=0,50; σ_r=0,09), ce qui confirme encore une fois l'hypothèse **H1**.

Dans la scène 4, comme dans la scène 2, les non-joueurs ont montré une nette préférence pour les translations égocentriques (r=0,44; σ_r=0,13) tandis que les joueurs avaient semblaient avoir plus de tolérance pour les deux mappings (r=0,13; σ_r=0,20). La même tendance est observée pour les rotations : les non-joueurs montrent une préférence pour les rotations allocentriques (r=0,39; σ_r=0,14) alors que les joueurs ont une opinion moins tranchée (r=0,09; σ_r=0,19). Bien que ces résultats aillent à nouveau à l'encontre de l'hypothèse **H2**, ils semblent tout de même confirmer notre conjecture précédente selon laquelle les joueurs seraient plus tolérants envers l'utilisation de divers mappings dans différents environnements virtuels.

Résumé des résultats

Dans le groupe de « joueurs » familiers avec le jeu vidéo 3D, la préférence de base (l'alternative considérée comme plus naturelle sans indices contextuels spécifiques) est moins marquée que dans le groupe des « non-joueurs » en ce qui concerne les translations. Cependant, la préférence de base pour les rotations est similaire.

Les joueurs semblent être moins influencés par le contenu de la scène virtuelle que les non-joueurs, sauf lorsque l'objet manipulé était vu de l'intérieur. Cela pourrait signifier que les joueurs sont plus tolérants face des mappings différents dans des contextes variés. Malgré cela, les mappings favorisés dans chaque scène restent les mêmes dans les deux groupes.

B.2 Inversion sur chaque axe

L'expérience présentée dans la section 5.7.2 était constituée d'une troisième partie, visant à évaluer l'influence de chaque axe de manipulation sur la préférence entre les mappings allocentrique et égocentrique. Certaines études [Diaz et Sims 2005; Kaminaka et Egli 1985] ont en effet mis en évidence des stéréotypes de population différents entre les trois axes de manipulation. Nous avons donc émis l'hypothèse de retrouver une influence différente de chaque axe dans cette configuration localement couplée :

— **H5** : Certains axes ont plus d'importance que d'autres dans l'appréciation des utilisateurs pour un mapping donné.

Cette hypothèse a donc été testée au cours d'une troisième partie optionnelle de l'expérience. Les parties 1 et 2 étant relativement longues (environ 30 minutes pour les deux), nous avons

proposé aux participants de poursuivre l'expérience uniquement s'ils le souhaitaient. Un total de 14 participants ont choisi de continuer.

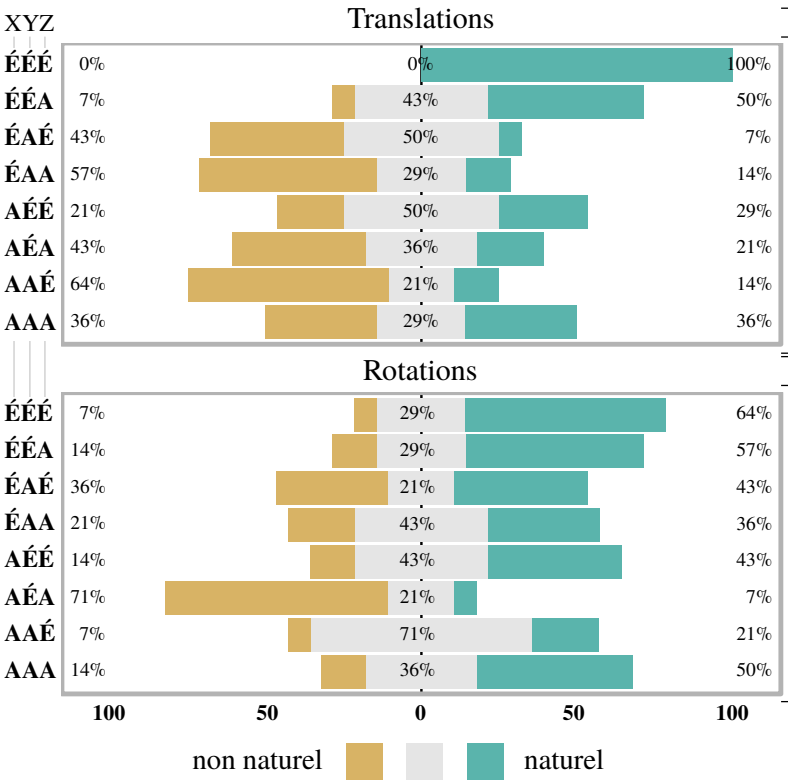
Cette troisième partie consistait en deux conditions : une pour les translations et une pour les rotations, présentées dans un ordre différent d'un participant à l'autre. Les deux conditions se déroulaient dans la scène 1. Dans chaque condition, 8 mappings différents devaient être évalués dans lesquels chaque axe de manipulation (x, y et z) était alternativement inversé. Ces 8 permutations étaient présentées dans un ordre aléatoire.

L'échelle de notation était sur 3 points, ce qui revenait donc à choisir entre « non naturel », « neutre » et « naturel ». Nous avons délibérément réduit le nombre de choix et autorisé une réponse neutre pour éviter de surcharger les participants, étant donné le grand nombre de mappings à comparer et les faibles variations d'un mapping à l'autre.

Résultats

La figure B.2 présente les scores attribués par les participants dans la troisième partie de l'expérience, dans laquelle la direction de mouvement était alternativement inversée le long de chaque axe de manipulation. L'axe x représente la direction gauche-droite de l'écran de la tablette (elle-même tenue en mode paysage), l'axe y représente la direction haut-bas de l'écran, et l'axe z est orthogonal au plan de l'écran. À la différence des deux parties précédentes, les scores étaient ici attribués sur une échelle de Likert à 3 points.

Dans le cas des translations, une combinaison a été très clairement préférée à toutes les autres : le mapping pleinement égocentrique. Ce résultat est cohérent avec les parties 1 et 2 de l'expérience qui ont montré une préférence pour les translations égocentriques dans la scène 1. Compte tenu de la force de cet effet (100 % des participants ont perçu ce mapping comme naturel, bien au-dessus des autres combinaisons), il ne semble pas y avoir d'intérêt à utiliser une combinaison mixte. Cependant il reste encore à déterminer si certains axes de manipulation ont plus d'influence que d'autres sur l'appréciation d'un mapping donné (hypothèse **H5**). Une tendance de ce type apparaît dans les résultats en translation : les combinaisons où l'axe y était inversé (c'est-à-dire égocentrique) ont été systématiquement notées plus haut que les autres combinaisons, à l'exception du mapping pleinement allocentrique. Nous avons donc conduit une nouvelle analyse en groupant les scores en deux catégories « allocentrique » et « égocentrique » selon l'état (inversé ou non) de chaque axe de manipulation. Les résultats ont effectivement révélé un effet plus important de l'axe y (r=0,35; σ_r=0,07) que des axes x (r=0,18; σ_r=0,09) ou z (r=0,05; σ_r=0,10). Le fait que le mapping pleinement allocentrique ait été noté comparativement plus haut, malgré son axe y non-inversé, pourrait s'expliquer par une préférence pour un mapping cohérent entre tous les axes ; de la même manière que la préférence supposée pour un mapping identique



	AAA	AAÉ	AÉA	AÉÉ	ÉAA	ÉAÉ	ÉÉA
ÉÉÉ	0,55 0,05	0,63 0,03	0,60 0,03	0,58 0,04	0,62 0,03	0,63 0,02	0,50 0,07
ÉÉA	0,26 0,17	0,49 0,10	0,41 0,12	0,31 0,15	0,55 0,05	0,53 0,06	-
ÉAÉ	-0,20 0,17	0,09 0,18	-0,12 0,19	-0,27 0,15	0,10 0,18	-	-
ÉAA	-0,24 0,17	0,05 0,19	-0,17 0,19	-0,33 0,15	-	-	-
AÉÉ	0,03 0,20	0,34 0,16	0,17 0,18	-	-	-	-
AÉA	-0,17 0,18	0,21 0,18	-	-	-	-	-
AAÉ	-0,30 0,15	-	-	-	-	-	-

	AAA	AAÉ	AÉA	AÉÉ	ÉAA	ÉAÉ	ÉÉA
ÉÉÉ	0,14 0,18	0,31 0,17	0,58 0,04	0,23 0,17	0,29 0,15	0,30 0,16	0,10 0,19
ÉÉA	0,11 0,19	0,27 0,18	0,51 0,09	0,09 0,19	0,18 0,18	0,19 0,18	-
ÉAÉ	-0,15 0,19	0,00 0,19	0,49 0,07	-0,11 0,19	-0,04 0,19	-	-
ÉAA	-0,13 0,18	0,00 0,19	0,39 0,14	-0,15 0,18	-	-	-
AÉÉ	-0,06 0,19	0,12 0,19	0,47 0,11	-	-	-	-
AÉA	-0,55 0,05	-0,56 0,05	-	-	-	-	-
AAÉ	-0,21 0,17	-	-	-	-	-	-

FIGURE B.2 — Scores (et tailles d'effet) relevés dans la troisième partie de l'expérience, qui consiste à inverser alternativement la direction de mouvement sur chaque axe de manipulation. Les mappings sont représentés sous la forme « **XYZ** », où chaque axe est soit **A**=allocentrique (direct) soit **É**=égocentrique (inversé).

entre translations et rotations mentionnée dans la section précédente. Quoi qu'il en soit, l'influence plus importante de l'axe y tend à valider l'hypothèse **H5** en translation.

Concernant les rotations, aucune combinaison n'a été unanimement préférée et les différences entre chaque combinaison étaient globalement plus faibles que pour les translations. La séparation des scores en deux groupes selon l'état de chaque axe de manipulation a confirmé que l'influence de chaque axe est effectivement limitée (x : r=0,17; σ_r=0,09, y : r=0,00; σ_r=0,09, z : r=0,11; σ_r=0,09). Pourtant, une combinaison paraît avoir été largement considérée peu naturelle : le mapping allo-égo-allo (AÉA). Ce résultat est surprenant, et nous ne pouvons pas pour l'instant expliquer un telle baisse dans les scores pour cette combinaison précise.

Résumé des résultats

Aucun avantage n'a été observé au fait d'inverser sélectivement certains axes de manipulation par rapport à un mapping « pleinement » égocentrique ou allocentrique. Si un tel mapping doit malgré tout être implémenté, il apparaît que l'axe y (l'axe vertical de l'écran de l'appareil mobile) a plus d'importance que les autres dans l'appréciation du mapping (un axe y « inversé » étant préférable), appuyant ainsi l'hypothèse **H5** mais seulement dans le cas des translations. Les rotations, quant à elles, ne montrent pas d'axe privilégié. Pour une raison qui reste encore à déterminer, une des combinaisons dans le cas des rotations (allo-égo-allo) a été perçue par les participants comme bien moins naturelle que les autres.

Bibliographie

ARCEP (2015). Baromètre du numérique, résultats 2015. http://www.arcep.fr/.

Avrahami, D., Wobbrock, J. O. et Izadi, S. (2011). Portico : tangible interaction on and around a tablet. *In Proceedings of the 24th annual ACM symposium on User interface software and technology*, page 347–356.

Bade, R., Ritter, F. et Preim, B. (2005). Usability comparison of mouse-based interaction techniques for predictable 3D rotation. *In Proc. Smart Graphics*, pages 138–150. Springer, Berlin/Heidelberg.

Baricevic, D., Lee, C., Turk, M., Hollerer, T. et Bowman, D. (2012). A hand-held AR magic lens with user-perspective rendering. *In 2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 197–206.

Bartlett, J. (2000). Rock 'n' Scroll is here to stay. *IEEE Computer Graphics and Applications*, 20(3):40–45.

Basu, A. et Johnsen, K. (2014). Ubiquitous virtual reality 'To-Go'. *In Virtual Reality (VR), 2014 IEEE*, page 161–162. IEEE.

Basu, A., Saupe, C., Refour, E., Raij, A. et Johnsen, K. (2012). Immersive 3dui on one dollar a day. *In 3D User Interfaces (3DUI), 2012 IEEE Symposium on*, page 97–100. IEEE.

Bérard, F., Wang, G. et Cooperstock, J. R. (2011). On the limits of the human motor control precision : The search for a device's human resolution. *In Proc. INTERACT*, pages 107–122. Springer, Berlin/Heidelberg.

Besançon, L., Issartel, P., Ammi, M. et Isenberg, T. (2016). *Usability Comparison of Mouse, Touch and Tangible Inputs for 3D Data Manipulation*.

Bier, E. A. (1987). Skitters and jacks : Interactive 3D positioning tools. *In Proceedings of the 1986 Workshop on Interactive 3D Graphics*, I3D '86, page 183–196, New York, NY, USA. ACM.

Birn, J. (2004). CG industry job outlook. http://www.3drender.com/.

Blackshaw, M., Lakatos, D., Ishii, H. et Perlin, K. (2012). T(ether). http://kiwi.media.mit.edu/.

Bolt, R. A. (1980). “Put-that-there” : voice and gesture at the graphics interface. *In Proceedings of the 7th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH ’80, page 262–270, New York, NY, USA. ACM.

Bouvier, P. (2009). *La présence en réalité virtuelle, une approche centrée utilisateur*. Thèse de doctorat, Université Paris-Est.

Bowman, D. A. et Hodges, L. F. (1995). User interface constraints for immersive virtual environment applications.

Bowman, D. A., Kruijff, E., LaViola Jr., J. J. et Poupyrev, I. (2004). *3D User Interfaces : Theory and Practice*. Addison-Wesley.

Britton, E. G., Lipscomb, J. S. et Pique, M. E. (1978). Making nested rotations convenient for the user. *ACM SIGGRAPH Computer Graphics*, 12(3):222–227.

Brown, L. D., Hua, H. et Gao, C. (2003). A widget framework for augmented interaction in SCAPE. *In Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, UIST ’03, page 1–10, New York, NY, USA. ACM.

Buchanan, P., Seichter, H., Billinghurst, M. et Grasset, R. (2008). Augmented reality and rigid body simulation for edutainment : The interesting mechanism - an AR puzzle to teach newton physics. *In Proc. ACE ’08*, page 17–20. ACM.

Buda, V. (2012). *Rotation Techniques for 3D Object Interaction on Mobile Devices*. Thèse de doctorat, Utrecht University.

Burgess, N., Spiers, H. J. et Paleologou, E. (2004). Orientational manoeuvres in the dark : dissociating allocentric and egocentric influences on spatial memory. *Cognition*, 94(2):149–166.

Business Advantage Group (2015). Worldwide CAD trends 2015 survey report – CAD hardware current and future usage. http://www.business-advantage.com/.

Chan, A. H. S., Shum, V. W. Y., Law, H. W. et Hui, I. K. (2003). Precise effects of control position, indicator type, and scale side on human performance. *The International Journal of Advanced Manufacturing Technology*, 22(5–6):380–386.

Cho, S.-J., Choi, C., Sung, Y., Lee, K., Kim, Y.-B. et Murray-Smith, R. (2007). Dynamics of tilt-based browsing on mobile devices. *In CHI Extended Abstracts*, pages 1947–1952, New York. ACM.

Chung, J. C., Harris, M. R., Brooks, F. P., Fuchs, H., Kelley, M. T., Hughes, J., Ouh-young, M., Cheung, C., Holloway, R. L. et Pique, M. (1989). Exploring virtual worlds with head-mounted displays. volume 1083, pages 42–52.

CNC Cookbook (2016). CAD survey results, part 1 : Market share.

Couture, N. et Rivière, G. (2006). Faisabilité d'une interface tangible pour la validation d'Hypothèses en géosciences. *In 3è Journées Francophones Mobilité et Ubiquité (Ubimob'06)*, page 163–164. ACM.

Crison, F., Lécuyer, A., Savary, A., Mellet-d'Huart, D., Burkhardt, J.-M. et Dautin, J.-L. (2004). The use of haptic and pseudo-haptic feedback for the technical training of milling. *In EuroHaptics Conference poster, Munich, Germany*.

Cruz-Neira, C., Sandin, D. J. et DeFanti, T. A. (1993). Surround-screen projection-based virtual reality : The design and implementation of the CAVE. *In Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '93, page 135–142, New York, NY, USA. ACM.

Cruz-Neira, C., Sandin, D. J., DeFanti, T. A., Kenyon, R. V. et Hart, J. C. (1992). The CAVE : audio visual experience automatic virtual environment. *Commun. ACM*, 35(6):64–72.

Czernuszenko, M., Pape, D., Sandin, D., DeFanti, T., Dawe, G. L. et Brown, M. D. (1997). The ImmersaDesk and infinity wall projection-based virtual reality displays. *SIGGRAPH Comput. Graph.*, 31(2):46–49.

Daiber, F., Li, L. et Krüger, A. (2012). Designing gestures for mobile 3D gaming. *In Proc. MUM*, pages 3 :1–3 :4, New York. ACM.

de Sorbier, F., Bouvier, P., Herubel, A. et Biri, V. (2009). From research on the virtual reality installation. *In* Hippe, Z. S. et Kulikowski, J. L., éditeurs : *Human-Computer Systems Interaction*, numéro 60 de Advances in Intelligent and Soft Computing, pages 335–345. Springer Berlin Heidelberg.

Decle, F. et Hachet, M. (2009). A study of direct versus planned 3D camera manipulation on touch-based mobile phones. *In Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '09, page 32 :1–32 :5, New York, NY, USA. ACM.

Deering, M. (1992). High resolution virtual reality. *SIGGRAPH Comput. Graph.*, 26(2): 195–202.

Diaz, D. D. et Sims, V. K. (2005). Accidental inversion during three-dimensional orientational control. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 49(13):1248–1250.

Dominjon, L., Lecuyer, A., Burkhardt, J.-M., Andrade-Barroso, G. et Richir, S. (2005a). The "Bubble" technique : Interacting with large virtual environments using haptic devices with limited workspace. *In Proc. World Haptics*, pages 639–640, Los Alamitos. IEEE Computer Society.

Dominjon, L., Lecuyer, A., Burkhardt, J.-M., Richard, P. et Richir, S. (2005b). Influence of control/display ratio on the perception of mass of manipulated objects in virtual environments. *In Proc. IEEE VR '05*, pages 19–25.

Eslambolchilar, P. et Murray-Smith, R. (2008). Control centric approach in designing scrolling and zooming user interfaces. *International Journal of Human-Computer Studies*, 66(12):838–856.

Feiner, S., MacIntyre, B., Höllerer, T. et Webster, A. (1997). A touring machine : Prototyping 3D mobile augmented reality systems for exploring the urban environment. *Personal Technologies*, 1(4):208–217.

Feinstein, D. (2002). View navigation and magnification of a hand-held device with a display. US Patent No. 6,466,198.

Ferreira, F., Cabral, M., Belloc, O., Miller, G., Kurashima, C., Lopes, R. d. D., Stavness, I., Anacleto, J., Zuffo, M. et Fels, S. (2014). Spheree : A 3D perspective-corrected interactive spherical scalable display. *In Proceedings of the 41st Conference on Computer Graphics and Interactive Techniques ; Posters*, SIGGRAPH'14, page 86 :1–86 :1, Vancouver, Canada. ACM.

Fisher, S. S., McGreevy, M., Humphries, J. et Robinett, W. (1987). Virtual environment display system. *In Proceedings of the 1986 Workshop on Interactive 3D Graphics*, I3D '86, page 77–87, New York, NY, USA. ACM.

Fitts, P. M. (1951). Engineering psychology and equipment design. *In* Stevens, S. S., éditeur : *Handbook of Experimental Psychology*, pages 1287–1340. Wiley.

Fitts, P. M. et Seeger, C. M. (1953). SR compatibility : Spatial characteristics of stimulus and response codes. *Journal of Experimental Psychology*, 46(3):199–210.

Fitzmaurice, G. W. (1996). *Graspable user interfaces*. Thèse de doctorat, University of Toronto, Canada.

Fitzmaurice, G. W., Zhai, S. et Chignell, M. H. (1993). Virtual reality for palmtop computers. *ACM Transactions on Information Systems (TOIS)*, 11(3):197–218.

Foxlin, E. et Harrington, M. (2000). Weartrack : A self-referenced head and hand tracker for wearable computers and portable vr. *In Wearable Computers, The Fourth International Symposium on*, page 155–162. IEEE.

Francone, J. et Nigay, L. (2011). Using the user's point of view for interaction on mobile devices. *In 23rd French Speaking Conference on Human-Computer Interaction*, IHM '11, page 4 :1–4 :8, New York, NY, USA. ACM.

Fritz, C. O., Morris, P. E. et Richler, J. J. (2012). Effect size estimates : Current use, calculations, and interpretation. *Journal of Experimental Psychology : General*, 141(1):2–18.

Fröhlich, B. et Plate, J. (2000). The cubic mouse : A new device for three-dimensional input. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '00, page 526–531, New York, NY, USA. ACM.

Fuchs, P. et Moreau, G. (2006). *Le traité de la réalité virtuelle*, volume 2. Presses des MINES.

Gibson, J. J. (1977). Perceiving, acting, and knowing : Toward an ecological psychology. *chap. The Theory of Affordances). Hillsdale, NJ : Lawrence Erlbaum*.

Gillet, A., Sanner, M., Stoffler, D., Goodsell, D. et Olson, A. (2004). Augmented reality with tangible auto-fabricated models for molecular biology applications. *In Visualization, 2004. IEEE*, page 235–241.

Gleicher, M. (1993). A graphics toolkit based on differential constraints. *In Proceedings of the 6th annual ACM symposium on User interface software and technology*, page 109–120. ACM.

Grimes, B. (2013). University of Illinois at Chicago : Virtual Reality's CAVE Pioneer. http://www.edtechmagazine.com/.

Hachet, M., Pouderoux, J. et Guitton, P. (2005). A camera-based interface for interaction with mobile handheld computers. *In Proceedings of the 2005 Symposium on Interactive 3D Graphics and Games*, I3D '05, page 65–72, New York, NY, USA. ACM.

Hancock, M., Carpendale, S. et Cockburn, A. (2007). Shallow-depth 3d interaction : design and evaluation of one-, two-and three-touch techniques. *In Proceedings of the SIGCHI conference on Human factors in computing systems*, page 1147–1156.

Hancock, M., ten Cate, T. et Carpendale, S. (2009). Sticky tools : full 6DOF force-based interaction for multi-touch tables. *In Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ITS '09, page 133–140, New York, NY, USA. ACM.

Hansen, T. R., Eriksson, E. et Lykke-Olesen, A. (2006). Mixed interaction space – Expanding the interaction space with mobile devices. *In People and Computers XIX — The Bigger Picture*, pages 365–380. Springer, London.

Harish, P. et Narayanan, P. (2013). Designing perspectively correct multiplanar displays. *IEEE Transactions on Visualization and Computer Graphics*, 19(3):407–419.

Harrison, B. L., Fishkin, K. P., Gujar, A., Mochon, C. et Want, R. (1998). Squeeze me, hold me, tilt me ! An exploration of manipulative user interfaces. *In Proc. CHI*, pages 17–24, New York. ACM.

Hatwell, Y., Streri, A. et Gentaz, E. (2003). *Touching for knowing : cognitive psychology of haptic manual perception*, volume 53. John Benjamins Publishing.

Henrysson, A., Billinghurst, M. et Ollila, M. (2005). Virtual object manipulation using a mobile phone. *In Proc. ICAT*, pages 164–171, New York. ACM.

Hinckley, K., Pausch, R., Goble, J. C. et Kassell, N. F. (1994). Passive real-world interface props for neurosurgical visualization. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '94, page 452–458, New York, NY, USA. ACM.

Hinckley, K., Pierce, J., Sinclair, M. et Horvitz, E. (2000). Sensing techniques for mobile interaction. *In Proc. UIST*, pages 91–100, New York. ACM.

Hinckley, K. et Song, H. (2011). Sensor synaesthesia : Touch in motion, and motion in touch. *In Proc. CHI*, pages 801–810, New York. ACM.

Hodgson, E., Bachmann, E., Waller, D., Bair, A. et Oberlin, A. (2012). Virtual reality in the wild : A self-contained and wearable simulation system. *In 2012 IEEE Virtual Reality Workshops (VRW)*, page 157–158. IEEE.

Hwang, J., Jung, J. et Kim, G. J. (2006). Hand-held virtual reality : A feasibility study. *In Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '06, page 356–363, New York, NY, USA. ACM.

Hwang, S., Ahn, M. et Wohn, K.-y. (2013). MagGetz : customizable passive tangible controllers on and around conventional mobile devices. *In Proceedings of the 26th annual ACM symposium on User interface software and technology*, UIST '13, page 411–416, New York, NY, USA. ACM.

Inami, M. (1997). Media3 : The virtual hologram. *In ACM SIGGRAPH 97 Visual Proceedings : The Art and Interdisciplinary Programs of SIGGRAPH '97*, SIGGRAPH '97, page 107, New York, NY, USA. ACM.

Insee (2012). Entreprises – Enquête sur les technologies de l'information et de la communication et le commerce électronique 2011. http://www.insee.fr/.

Insee (2015). Conditions de vie-Société – De plus en plus de foyers équipés de biens électroniques. http://www.insee.fr/.

Ishii, H. (2008). The tangible user interface and its evolution. *Commun. ACM*, 51(6):32–36.

Ishii, H. et Ullmer, B. (1997). Tangible bits : towards seamless interfaces between people, bits and atoms. *In Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, page 234–241.

Jackson, B., Lau, T. Y., Schroeder, D., Toussaint, K. C. et Keefe, D. F. (2013). A lightweight tangible 3D interface for interactive visualization of thin fiber structures. *Visualization and Computer Graphics, IEEE Transactions on*, 19(12):2802–2809.

Jankowski, J. et Hachet, M. (2013). A survey of interaction techniques for interactive 3D environments.

Jellinek, H. D. et Card, S. K. (1990). Powermice and user performance. *In Proc. CHI*, pages 213–220, New York. ACM.

Jeon, S., Metzger, J.-C., Choi, S. et Harders, M. (2011). Extensions to haptic augmented reality : Modulating friction and weight. *In Proc. WHC '11*, pages 227–232. IEEE.

Joshi, N., Kar, A. et Cohen, M. (2012). Looking at you : Fused gyro and face tracking for viewing large imagery on mobile devices. *In Proc. CHI*, pages 2211–2220, New York. ACM.

Kaminaka, M. S. et Egli, E. A. (1985). Lever controls on specialised farm equipment : Some control/response stereotypes. *Applied Ergonomics*, 16(3):193–199.

Kane, S. K., Bigham, J. P. et Wobbrock, J. O. (2008). Slide rule : making mobile touch screens accessible to blind people using multi-touch interaction techniques. *In Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*, page 73–80.

Kang, C. et Woo, W. (2011). ARMate : an interactive AR character responding to real objects. *In Edutainment Technologies. Educational Games and Virtual Reality/Augmented Reality Applications*, Lecture Notes in Computer Science. Springer Berlin Heidelberg.

Kim, S., Ishii, M., Koike, Y. et Sato, M. (2000). Development of tension based haptic interface and possibility of its application to virtual reality. *In Proc. VRST '00*, page 199–205. ACM.

Kim, S. W., Treskunov, A. et Marti, S. (2011). DRIVE : directly reaching into virtual environment with bare hand manipulation behind mobile display. *In 2011 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 107–108.

Klatzky, R. L. (1998a). Allocentric and egocentric spatial representations : Definitions, distinctions, and interconnections. *In Spatial Cognition*, pages 1–17. Springer, Berlin/ Heidelberg.

Klatzky, R. L. (1998b). Allocentric and egocentric spatial representations : Definitions, distinctions, and interconnections. *In Spatial Cognition*, pages 1–17. Springer, Berlin/ Heidelberg.

Krueger, W. et Froehlich, B. (1994). The responsive workbench. *IEEE Computer Graphics and Applications*, 14(3):12–15.

Larousse (2016). Définitions : efficacité - dictionnaire de français larousse. http://www.larousse.fr/.

LaValle, S. M. (2016). Virtual reality. http://vr.cs.uiuc.edu/.

LaViola, Joseph J., J. (2000). A discussion of cybersickness in virtual environments. *SIGCHI Bull.*, 32(1):47–56.

LaViola, J. J. et Katzourin, M. (2007). An exploration of non-isomorphic 3D rotation in surround screen virtual environments. *In Proc. 3DUI*, pages 49–54, Los Alamitos. IEEE Computer Society.

Lee, J. C. (2008). Hacking the nintendo wii remote. *IEEE Pervasive Computing*, 7(3):39–45.

Leek, M. R. (2001). Adaptive procedures in psychophysical research. *Perception & psychophysics*, 63(8):1279–1292.

Liang, R.-H., Cheng, K.-Y., Chan, L., Peng, C.-X., Chen, M. Y., Liang, R.-H., Yang, D.-N. et Chen, B.-Y. (2013). GaussBits : magnetic tangible bits for portable and occlusion-free near-surface interactions. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, page 1391–1400.

Lindeman, R. W. (2016). A low-cost, low-latency approach to dynamic immersion in occlusive head-mounted displays. IEEE.

Lopez-Gulliver, R., Yoshida, S., Yano, S. et Inoue, N. (2009). gCubik : Real-time integral image rendering for a cubic 3D display. *In SIGGRAPH Emerging Technologies*, page 11, New York. ACM.

Lécuyer, A. (2009). Simulating haptic feedback using vision : A survey of research and applications of pseudo-haptic feedback. *Presence : Teleoperators and Virtual Environments*, 18(1):39–53.

Lécuyer, A., Coquillart, S., Kheddar, A., Richard, P. et Coiffet, P. (2000). Pseudo-haptic feedback : can isometric input devices simulate force feedback ? *In Proc. IEEE VR '00*, pages 83–90.

Marchal, M. (2014). *3D Multimodal Interaction with Physically-based Virtual Environments*. Habilitation à diriger des recherches, Université de Rennes 1.

Margolis, T., DeFanti, T. A., Dawe, G., Prudhomme, A., Schulze, J. P. et Cutchin, S. (2011). Low cost heads-up virtual reality (HUVR) with optical tracking and haptic feedback. *In IS&T/SPIE Electronic Imaging*, page 786417–786417. International Society for Optics and Photonics.

Marzo, A., Bossavit, B. et Hachet, M. (2014). Combining multi-touch input and device movement for 3D manipulations in mobile augmented reality environments. *In Proc. SUI*, pages 13–16, New York. ACM.

Massimino, M. J., Sheridan, T. B. et Roseborough, J. B. (1989). One handed tracking in six degrees of freedom. *In Proc. Systems, Man and Cybernetics*, volume 2, pages 498–503, New York. IEEE.

Miller, S. A., Misch, N. J. et Dalton, A. J. (2005). Low-cost, portable, multi-wall virtual reality. Aalborg, Denmark.

Minamizawa, K., Fukamachi, S., Kajimoto, H., Kawakami, N. et Tachi, S. (2007). Gravity grabber : Wearable haptic display to present virtual mass sensation. *In SIGGRAPH '07 Emerging Technologies*. ACM.

Mine, M. (1995). Virtual environment interaction techniques. Rapport technique, UNC Chapel Hill CS Dept.

Mossel, A., Venditti, B. et Kaufmann, H. (2013). 3DTouch and HOMER-S : Intuitive manipulation techniques for one-handed handheld augmented reality. *In Proc. VRIC*, pages 12 :1–12 :10, New York. ACM.

Mulder, J. D. et Van Liere, R. (2002). The personal space station : Bringing interaction within reach. *In Proceedings of the virtual reality international conference, VRIC*, volume 2002, page 73–81.

Myers, B. A. (1988). A taxonomy of window manager user interfaces. *IEEE Computer Graphics and applications*, 8(5):65–84.

Neale, S., Chinthammit, W., Lueg, C. et Nixon, P. (2013). RelicPad : A hands-on, mobile approach to collaborative exploration of virtual museum artifacts. *In Proc. INTERACT*, pages 86–103. Springer, Berlin/Heidelberg.

Newton, I. (1687). *Philosophiae naturalis principia mathematica*. J. Societatis Regiae ac Typis J. Streater.

Noma, H., Miyasato, T. et Kishino, F. (1996). A palmtop display for dextrous manipulation with haptic sensation. *In Proc. CHI*, pages 126–133, New York. ACM.

Norman, D. A. (1993). *Things that make us smart : Defending human attributes in the age of the machine*.

Oakley, I. et O'Modhrain, S. (2005). Tilt to scroll : Evaluating a motion based vibrotactile mobile interface. *In Proc. World Haptics*, pages 40–49, Los Alamitos. IEEE Computer Society.

Piumsomboon, T., Clark, A. et Billinghurst, M. (2011). Physically-based interaction for tabletop augmented reality using a depth-sensing camera for environment mapping. *In Proc. Image and Vision Computing New Zealand (IVCNZ-2011)*, pages 161–166, Auckland.

Poupyrev, I., Billinghurst, M., Weghorst, S. et Ichikawa, T. (1996). The Go-Go interaction technique : Non-linear mapping for direct manipulation in VR. *In Proc. UIST*, pages 79–80, New York. ACM.

Poupyrev, I., Ichikawa, T., Weghorst, S. et Billinghurst, M. (1998). Egocentric object manipulation in virtual environments : Empirical evaluation of interaction techniques. *Computer Graphics Forum*, 17(3):41–52.

Poupyrev, I., Weghorst, S. et Fels, S. (2000). Non-isomorphic 3D rotational techniques. *In Proc. CHI*, pages 540–547, New York. ACM.

Poupyrev, I., Weghorst, S., Otsuka, T. et Ichikawa, T. (1999). Amplifying spatial rotations in 3D interfaces. *In CHI Extended Abstracts*, pages 256–257, New York. ACM.

Pusch, A., Martin, O. et Coquillart, S. (2008). HEMP-Hand-Displacement-Based pseudo-haptics : A study of a force field application. *In Proc. 3DUI '08*, pages 59–66.

Qi, W. et Martens, J.-B. (2005). Tangible user interfaces for 3D clipping plane interaction with volumetric data : a case study. *In Proceedings of the 7th international conference on Multimodal interfaces*, ICMI '05, page 252–258, New York, NY, USA. ACM.

Quek, F. K. (1995). Eyes in the interface. *Image and vision computing*, 13(6):511–525.

Rabhi, A., Sadiq, A. et Mouloudi, A. (2015). Face tracking : State of the art. *In Complex Systems (WCCS), 2015 Third World Conference on*, page 1–8. IEEE.

Rahman, M., Gustafson, S., Irani, P. et Subramanian, S. (2009). Tilt techniques : Investigating the dexterity of wrist-based input. *In Proc. CHI*, pages 1943–1952, New York. ACM.

Rekimoto, J. (1996). Tilting operations for small screen interfaces. *In Proc. UIST*, pages 167–168, New York. ACM.

Rolland, J. P., Holloway, R. L. et Fuchs, H. (1995). Comparison of optical and video see-through, head-mounted displays. *In Photonics for Industrial Applications*, page 293–307. International Society for Optics and Photonics.

Ross, H. E. et Murray, D. J. (1996). *E.H. Weber on the tactile senses*. Erlbaum (UK) Taylor & Francis Hove.

Roussou, M. et Drettakis, G. (2005). Can VR be useful and usable in real-world contexts ? observations from the application and evaluation of VR in realistic usage conditions. *In HCI International 2005 - 1st International Conference on Virtual Reality*, page 9, Las Vegas, United States.

Ryu, J. et Ohno, R. (2007). A novel room re-layout design tool for preparedness of earthquake disaster using real-time physics simulation and force-feedback interface. *In* Dong, A., Moere, A. V. et Gero, J. S., éditeurs : *Computer-Aided Architectural Design Futures (CAADFutures) 2007*, pages 541–554. Springer Netherlands.

Shaer, O. et Hornecker, E. (2010). Tangible user interfaces : past, present, and future directions. *Foundations and Trends in Human-Computer Interaction*, 3(1–2):1–137.

Sharlin, E., Watson, B., Sutphen, S., Liu, L., Lederer, R. et Frazer, J. (2009). A tangible user interface for assessing cognitive mapping ability. *International Journal of Human-Computer Studies*, 67(3):269–278.

Sherman, W. R. et Craig, A. B. (2002). *Understanding virtual reality : Interface, application, and design*. Elsevier.

Shoemake, K. (1985). Animating rotation with quaternion curves. *ACM SIGGRAPH Computer Graphics*, 19(3):245–254.

Small, D. et Ishii, H. (1997). Design of spatially aware graspable displays. *In CHI Extended Abstracts*, pages 367–368, New York. ACM.

Smith, T. J. et Smith, K. U. (1987). Feedback-control mechanisms of human behavior. *In* Salvendy, G., éditeur : *Handbook of Human Factors*, pages 251–293. Wiley, New York.

Spindler, M., Büschel, W. et Dachselt, R. (2012). Use your head : Tangible windows for 3D information spaces in a tabletop environment. *In Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces*, ITS '12, page 245–254, New York, NY, USA. ACM.

Spindler, M., Schuessler, M., Martsch, M. et Dachselt, R. (2014). Pinch-drag-flick vs. spatial input : Rethinking zoom & pan on mobile displays. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, page 1113–1122, New York, NY, USA. ACM.

Stavness, I., Lam, B. et Fels, S. (2010). pCubee : A perspective-corrected handheld cubic display. *In Proc. CHI*, pages 1381–1390, New York. ACM.

Stavness, I., Vogt, F. et Fels, S. (2006). Cubee : A cubic 3D display for physics-based interaction. *In ACM SIGGRAPH 2006 Sketches*, SIGGRAPH '06, New York, NY, USA. ACM.

Steinicke, F., Hinrichs, K. H., Schöning, J. et Krüger, A. (2008). Multi-touching 3D data : Towards direct interaction in stereoscopic display environments coupled with mobile devices. *In Proc. AVI Workshop on Designing Multi-Touch Interaction Techniques for Coupled Public and Private Displays*, page 46–49.

Sutherland, I. E. (1968). A head-mounted three dimensional display. *In Proceedings of the December 9-11, 1968, Fall Joint Computer Conference, Part I*, AFIPS '68 (Fall, part I), page 757–764, New York, NY, USA. ACM.

Taima, Y., Ban, Y., Narumi, T., Tanikawa, T. et Hirose, M. (2014). Controlling fatigue while lifting objects using pseudo-haptics in a mixed reality space. *In IEEE Haptics Symposium*, pages 175–180.

Tang, Y., Stavness, I. et Fels, S. S. (2014). The new pCubee : multi-touch perspective-corrected cubic display. *In CHI '14 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '14, page 419–422, New York, NY, USA. ACM.

Teather, R. J., Allison, R. S. et Stuerzlinger, W. (2009). Evaluating visual/motor co-location in fish-tank virtual reality. pages 624–629. IEEE.

Teather, R. J. et MacKenzie, I. S. (2014). Position vs. velocity control for tilt-based interaction. *In Proc. GI*, pages 51–58, Toronto. CIPS.

Teather, R. J. et Stuerzlinger, W. (2007). Guidelines for 3D positioning techniques. *In Proceedings of the 2007 conference on Future Play*, Future Play '07, page 61–68, New York, NY, USA. ACM.

Tsandilas, T., Dubois, E. et Raynal, M. (2013). Modeless pointing with low-precision wrist movements. *In Proc. INTERACT*, pages 494–511. Springer, Berlin/Heidelberg.

Tsang, M., Fitzmaurice, G. W., Kurtenbach, G., Khan, A. et Buxton, B. (2002). 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*, UIST '02, page 111–120, New York, NY, USA. ACM.

van Rhijn, A. et Mulder, J. D. (2006). Spatial input device structure and bimanual object manipulation in virtual environments. *In Proc. VRST*, pages 51–60, New York. ACM.

Wang, J., Zhai, S. et Canny, J. (2006). Camera phone based motion sensing : Interaction techniques, applications and performance study. *In Proc. UIST*, pages 101–110, New York. ACM.

Ware, C. et Arsenault, R. (2004). Frames of reference in virtual object rotation. *In Proc. APGV*, pages 135–141, New York. ACM.

Ware, C., Arthur, K. et Booth, K. S. (1993). Fish tank virtual reality. *In Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*, CHI '93, page 37–42, New York, NY, USA. ACM.

Ware, C. et Baxter, C. (1989). Bat Brushes : On the uses of six position and orientation parameters in a paint program. *In Proc. CHI*, pages 155–160, New York. ACM.

Warrick, M. J. (1947). Direction of movement in the use of control knobs to position visual indicators. *Psychological Research on Equipment Design*, pages 137–146.

Weberg, L., Brange, T. et Wendelbo-Hansson, Å. (2001). A piece of butter on the PDA display. *In CHI Extended Abstracts*, pages 435–436, New York. ACM.

Wiebe, J. et Vu, K.-P. L. (2009). Application of population stereotypes to computerized tasks. *In Proc. Human Interface and the Management of Information*, pages 718–725. Springer, Berlin/Heidelberg.

Wikipédia (2016). List of best-selling video games – Multi-platform. Version du 07/08/16.

Wilson, A. (2007). Depth-sensing video cameras for 3D tangible tabletop interaction. *In Proc. TABLETOP '07*, pages 201–204.

Yee, K.-P. (2003). Peephole displays : Pen interaction on spatially aware handheld computers. *In Proc. CHI*, pages 1–8, New York. ACM.

Yim, J., Qiu, E. et Graham, T. C. N. (2008). Experience in the design and development of a game based on head-tracking input. *In Proceedings of the 2008 Conference on Future Play : Research, Play, Share*, Future Play '08, page 236–239, New York, NY, USA. ACM.

Yokokohji, Y., Hollis, R. L. et Kanade, T. (1996). “What you can see is what you can feel” – Development of a visual/haptic interface to virtual environment. *In Virtual Reality Annual International Symposium, 1996., Proceedings of the IEEE 1996*, page 46–53. IEEE.

Zhai, S. (1995). *Human performance in six degree of freedom input control*. Thèse de doctorat, University of Toronto, Canada.

Zhai, S. (1998). User performance in relation to 3D input device design. *ACM SIGGRAPH Computer Graphics*, 32(4):50–54.

Zhai, S. et Milgram, P. (1998). 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*, CHI ’98, page 320–327, New York, NY, USA. ACM Press/Addison-Wesley Publishing Co.

Zhai, S., Milgram, P. et Buxton, W. (1996). The influence of muscle groups on performance of multiple degree-of-freedom input. *In Proceedings of the SIGCHI conference on Human factors in computing systems*, page 308–315. ACM.

université PARIS-SACLAY

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Nouvelles approches pour la portabilité, la non-intrusivité et l'accessibilité des interfaces de manipulation 3D

Mots clés : manipulation 3D, portabilité, non-intrusivité, accessibilité, appareils mobiles.

Résumé : Ce travail porte sur l'interaction avec des objets virtuels 3D, et plus particulièrement leur manipulation : sélection, translation et rotation. Les plates-formes les plus utilisées aujourd'hui pour ce type de tâche (ordinateur personnel et appareil mobile) ont été conçues avant tout pour l'interaction 2D, et sont donc peu adaptées à la manipulation 3D. Il existe pourtant une alternative plus efficace : les systèmes de réalité virtuelle. Cependant, les systèmes actuellement disponibles restent trop souvent encombrants, onéreux et/ou intrusifs pour l'utilisateur, et demeurent sous-utilisés pour cette raison.
Dans cette thèse, nous étudions de nouvelles approches pour rendre ce type d'interaction moins contraignant, tout en conservant une bonne efficacité de manipulation. Les principaux axes explorés sont la portabilité de l'interface, sa non-intrusivité, et l'utilisation de dispositifs plus largement accessibles au grand public.

Title: New approaches for the portability, non-intrusiveness and accessibility of 3D manipulation interfaces

Keywords: 3D manipulation, portability, non-intrusiveness, accessibility, mobile devices.

Abstract: This work focuses on 3D interaction with virtual objects, more specifically on 3D manipulation: selection, translation, and rotation. The most commonly-used interaction platforms (personal computer, mobile device) were designed for 2D interaction and thus are not well suited to 3D manipulation. There is a more efficient alternative for this task: virtual reality. However, current virtual reality systems are too often bulky, expensive and/or intrusive to the user, and for these reasons remain underused.
In this thesis, we propose and study new solutions to make this form of interaction more convenient without reducing its effectiveness. The main objectives we aim for are interface portability, non-intrusiveness, and the use of readily-available hardware.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France