



HAL
open science

Reconnaissance d'Objets Polyédriques à partir d'une image vidéo pour la téléopération

Mudar Shaheen

► **To cite this version:**

Mudar Shaheen. Reconnaissance d'Objets Polyédriques à partir d'une image vidéo pour la téléopération. Traitement du signal et de l'image [eess.SP]. Université d'Evry-Val d'Essonne, 1999. Français. NNT: . tel-00682206

HAL Id: tel-00682206

<https://theses.hal.science/tel-00682206>

Submitted on 23 Mar 2012

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.

N° d'ordre :

UNIVERSITE EVRY-VAL D'ESSONNE

THESE

Présentée en vue de l'obtention du

DOCTORAT DE L'UNIVERSITE EVRY-VAL D'ESSONNE

Spécialité:

SCIENCES DE L'INGENIEUR

Par

Mudar SHAHEEN

Titre:

**Reconnaissance d'Objets Polyédriques à partir d'une
image vidéo pour la téléopération**

Soutenue le 18 mars 1999 devant la commission d'examen :

M. Florent CHAVAND	Professeur Université d'Evry	Directeur de thèse
M. Mohamed BENJELLOUN	Professeur LASL-Univ. Littoral	Rapporteur
M. Etienne COLLE	Professeur Université d'Evry	Rapporteur
M. Michel DEVY	Professeur LAAS-Toulouse	Rapporteur
M. Raymond FOURNIER	Professeur CEA	Examineur
M. Wisama KHALIL	Professeur EC de Nantes	Examineur
M. Malik MALLEM	HDR, MCF à l'IIE-CNAM	Examineur

Table des matières

TABLE DES MATIERES	7
INTRODUCTION GENERALE	13
1. Etat de l'art sur les systèmes de reconnaissance d'objets (SRO)	17
Introduction	17
1.1 Systèmes de reconnaissance d'objets 3D (SRO)	18
1.1.1. Classification des SRO	18
1.1.1.1. Acquisition et Représentation Informatique du modèle 3D	18
• Méthodes basées sur la CAO	18
• Méthodes basées sur des mesures capteurs	19
1.1.1.2. Principales approches d'appariement	20
• Complexité de l'appariement	21
• Solutions	21
1.1.2. Quelques systèmes de reconnaissance d'objets	24
• ACRONYM	25
• SCERPO	25
• SRO de Ikeuchi et Kanade	26
• SRO de Flynn et Jain	28
• SRO de Siebert et Waxman	29
• SRO de Dickinson, Pentland et Rosenfeld	30
• PARVO	31
• SRO de Jain et Hoffman	32
• MORSE	33
• FCRG	33
1.2. Présentation du SRO développé pour MCIT	37
• Calibration de capteurs	37
• Modélisation 3D	38
• Traitement d'images 2D (TI)	39
• Appariement 2D/3D	39
• Vérification	40
Conclusion	41

2. Modélisation géométrique de l'environnement	43
2.1 Introduction	43
2.2 Modélisation et calibration de capteurs	43
2.2.1 Modèles de caméra.....	44
2.2.1.1 Modèle interne de la caméra.....	45
• modèle linéaire ne tenant pas compte des distorsions :.....	45
• modèle non-linéaire tenant compte des distorsions :.....	45
2.2.1.2 Modèle externe de la caméra.....	46
2.2.1.3 Modèle global de la caméra : cas du sténopé linéaire.....	47
2.2.1.4 Modèle géométrique direct de la caméra.....	48
2.2.1.5 Modèle géométrique inverse de la caméra.....	48
2.2.2 Calibration de la caméra.....	49
2.3 Modélisation géométrique de la scène	50
2.3.1 Représentation des objets 3D.....	51
2.3.1.1 Représentation filaire.....	51
2.3.1.2 Représentations surfaciques.....	51
2.3.1.3 Représentations volumiques.....	52
2.3.1.4 La représentation utilisée par MCIT.....	54
2.3.2 Acquisition des objets 3D.....	54
2.3.2.1 Plan d'interprétation.....	55
2.3.2.2 Acquisition de polyèdres.....	55
2.3.3 Construction des graphes d'aspects pour polyèdres.....	57
2.3.3.1 Méthode exhaustive.....	58
• Sphère de Gauss.....	58
• Projection du modèle.....	60
• Détermination des sommets cachés.....	61
• Discussion.....	63
2.3.3.2 Méthode analytique.....	64
• Cas convexe.....	64
• Cas concave.....	69
• Améliorations.....	73
2.4 Conclusion	74

3. Traitement et modélisation de l'image 2D	75
3.1. Introduction	75
3.2. Quelques définitions	75
• Voisinage.....	75
• Attributs et primitives.....	76
• Segmentation.....	76
3.3. Extraction de Contours	77
3.3.1. Introduction.....	77
3.3.2. Notions.....	78
• Modèles de contours.....	78
• Filtrage.....	78
• Approches de détection.....	79
3.3.3. Calcul du gradient.....	80
3.3.4. Amincissement de contours.....	82
3.3.5. Suivi de contours.....	83
3.4. Segmentation de contours	86
3.4.1. Introduction.....	86
3.4.2. La Transformée de Hough (TH).....	89
3.4.3. Amélioration de la Transformée de Hough.....	92
3.4.3.1. la méthode TH avec fusion.....	93
3.4.3.2. la méthode TH avec érosion.....	93
3.4.4. Passage aux paramètres réels des segments.....	97
3.5. Organisation perceptive	98
3.5.1. Introduction.....	98
3.5.2. Mise en équations.....	99
• Proximité.....	99
• Parallélisme.....	100
• Colinéarité.....	101
• Remarques.....	101
3.6. Représentation et correction du modèle 2D	102
3.6.1. Introduction.....	102
• Recherche des Chaînes de segments.....	103
• Correction des sommets.....	105
3.7. Conclusion	106

4. RECONNAISSANCE D'OBJETS POLYEDRIQUES.....	107
4.1. Introduction	107
4.2. Méthode du Graphe d'Appariement (GApp)	108
4.2.1. Formation des nœuds du GApp.....	111
• Critère de contour.....	112
• Critère de convexité.....	112
• Critère de colinéarité.....	113
• Critère de parallélisme.....	113
• Mise en correspondance segments-arêtes.....	114
• Exemple.....	115
4.2.2. Formation des arcs du GApp.....	116
• Critère logique.....	116
• Critère de visibilité.....	116
• Critère de connexité.....	117
• Critère d'invariance géométrique.....	118
• Exemple sur la formation des arcs.....	121
4.2.3. La recherche des cliques maximales.....	123
• Généraliser le critère d'invariance géométrique.....	123
• Procédure de recherche.....	125
4.2.4. Complexité de la méthode.....	126
4.2.5. Problèmes de l'approche graphe basé sur les chaînes.....	128
4.3. Méthode du Hachage Géométrique (HG)	130
4.3.1. Introduction.....	130
4.3.2. Propriétés d'invariances.....	130
4.3.3. Description de la méthode.....	132
4.3.3.1. Modélisation.....	133
• Extraction des aspects.....	134
• Représentation des aspects par des bases.....	134
• Construction de la Table de Hachage (TabH).....	135
4.3.3.2. Appariement.....	136
• Vote.....	136
• Mise en correspondance.....	137
4.3.4. Quantification de notre implantation.....	138
• Encombrement mémoire.....	139
• Complexité algorithmique.....	141
4.4. Mise en correspondance entre le modèle 2D et les modèles 3D	142
4.4.1. Méthodes de recalage.....	144
4.4.1.1. Le problème.....	144
4.4.1.2. Méthode géométrique des 3 segments.....	147
4.4.1.3. Méthode linéaire à petit mouvement.....	149
• Vecteur instantané de rotation.....	149
• Détermination de R et T.....	150
4.4.1.4. Méthode non-linéaire.....	151
4.4.1.5. Méthode employée (Mixte).....	153
4.4.2. Correction locale du traitement d'images.....	154
4.5. Conclusion.....	155

5. RESULTATS EXPERIMENTAUX.....	157
5.1 Modélisation de l'environnement	157
5.1.1 Calibration de la caméra.....	157
5.1.1.1 Conditions expérimentales.....	157
5.1.1.2 Calibration automatique de la caméra.....	159
5.1.1.3 Comparaison entre calibration manuelle et calibration automatique.....	161
5.1.1.4 Evaluation de l'apport du modèle non-linéaire.....	161
• Résultats avec la caméra CCD.....	162
• Résultats avec la caméra à tube.....	163
5.1.2 Graphe d'aspects.....	164
5.2 Reconnaissance d'objets polyédriques	167
5.2.1 Appariement.....	167
5.2.1.1 Appariement en utilisant les liaisons.....	167
• Introduction.....	167
• Algorithme.....	168
• Méthode de vérification.....	169
• Test d'appariement.....	170
5.1.1.2 Hachage géométrique.....	171
• Choix du type de bases.....	171
• Encombrement mémoire.....	173
• Temps d'exécution.....	174
5.1.2 Méthodes de recalage.....	175
• Conditions expérimentales.....	175
• Résultats et comparaisons.....	176
5.1.3 Validation et précision.....	181
5.3 Conclusion.....	184

CONCLUSION GENERALE	187
BIBLIOGRAPHIE	191
ANNEXES	199
Annexe 1 : Calibration du mini robot	199
• Résumé	199
• Notations	199
• Mise en équations	200
• Résultats de calibration	202
Annexe 2 : La méthode des moindres carrées	203
• Calcul des paramètres d'une droite	203
• Point à distance minimale d'un groupe de segments	204
Annexe 3 : Description de l'orientation	206
• Rappels	206
• Cosinus directeurs	207
• Angles de RTL (Roulis–Tangage–Lacet)	207
• Vecteur de rotation	208
• Quaternions	210
Annexe 4 : L'Algorithme de Levenberg-Marquardt	212
• Remarque	213
Annexe 5 : Méthode géométrique des 3 segments	214
• Détermination de la rotation R	214
• Détermination de la translation T	216

Introduction générale

La téléopération est la supervision et la commande à distance d'un système robotique employé sur les lieux de travail inaccessibles, hostiles à l'homme, tels que le milieu spatial, sous-marin, nucléaire ou chimique etc.

Notre laboratoire CEMIF (*Centre d'Etudes en Mécanique de l'Île de France*) travaille sur la mise au point d'un système d'interface pour la téléopération (MCIT : Multimedia Control Interface in Teleoperation). Le but de MCIT est de fournir à l'opérateur une aide visuelle pour la perception et pour le contrôle du site téléopéré. L'aide visuelle consiste en la mise à jour de la base de données géométriques de l'environnement (*monde virtuel*) et la superposition de cette base sur l'image vidéo. Cette amélioration de la perception visuelle de l'opérateur à partir d'une image vidéo est dite «*Réalité Augmentée*».

Le monde virtuel doit suivre les changements dans l'environnement : déplacement, apparition ou disparition des objets 3D. Cette mise à jour peut être effectuée manuellement par un opérateur ou d'une manière automatique à l'aide d'un *Système de Reconnaissance d'Objets* (SRO).

La reconnaissance d'objets tridimensionnels est un problème qui ne cesse pas de défier la communauté de vision par ordinateur. La preuve de l'existence de la solution (le système de vision humain), motive les chercheurs dans ce domaine. Ce problème peut être défini par la reconnaissance des objets prémodélisés dans une base de données à partir des données acquises de la scène. La reconnaissance signifie l'identification et la localisation des objets présents dans la scène.

Un système de vision «intelligent» doit être capable de reconnaître les objets existants dans son environnement (identification) et les situer (localisation). La majorité des systèmes de reconnaissance d'objets actuels tente de résoudre ce problème, en se basant sur un modèle (les objets sont connus a priori). La plupart de ces systèmes travaillent sur des objets dits *rigides* (non déformables) ou articulés. Un Système de Reconnaissance d'Objets (SRO) complet comprend différents modules : calibration de capteurs, modélisation 3D, traitement d'images de la scène et la reconnaissance proprement dite.

Selon le type de capteurs utilisé, la reconnaissance d'objets 3D peut être traitée en 3D/3D (capteurs télémétriques, stéréoscopiques ou séquences d'images) ou en 2D/3D basée sur une seule image de luminance. L'approche télémétrique pose des problèmes de mise en œuvre, de limite de portée et de lenteur. L'approche à plusieurs images (stéréo ou séquences) possède des difficultés d'appariement entre les différentes images d'entrée. Une seule image d'entrée ne fournit pas d'informations 3D (profondeur), elle donne lieu à un grand nombre d'hypothèses sur la reconnaissance. Peu de Systèmes de Reconnaissance d'Objets utilise la dernière approche.

Beaucoup de problèmes dans la reconnaissance d'objets n'ont pas été résolus et les systèmes actuels montrent leurs limites. Chaque SRO se concentre sur quelques aspects du problème en supposant que les autres aspects sont résolus ou le seront dans le futur.

Vu le temps limité d'une thèse nous nous sommes fixés un objectif, essayant de résoudre certains de ces problèmes sans trop compliquer le système : développer un système de reconnaissance de polyèdres à partir d'une seule image. Afin de réaliser avec précision une tâche de téléopération, le traitement d'images est une tâche indispensable et la reconnaissance d'objets est une autre. L'objet de la thèse est de développer ces deux points.

Nous avons développé un logiciel de traitement d'images basé sur la modélisation orientée objet. Deux méthodes d'amélioration de la méthode d'extraction de segments sont proposées. Les objets utilisés sont des polyèdres car il existait au préalable des méthodes pour leur modélisation. Cette hypothèse simplifie le traitement de l'image mais ne résout pas tous ses problèmes.

Concernant la reconnaissance d'objets, nous avons opté pour deux approches. L'approche graphe d'appariement nous permet de réduire efficacement le nombre d'hypothèses générées, notamment à l'aide des invariants projectifs. Ces invariants n'ont pas été appliqués dans un système de reconnaissance d'objets à notre connaissance. La méthode du graphe présente des problèmes lors de la perte des chaînes de segments données par le traitement d'images. Dans ce cas, nous appliquons l'approche du hachage géométrique qui donne toujours des hypothèses même en présence des défauts du traitement d'images.

L'utilisation de ces deux approches d'appariement nécessite la construction des graphes d'aspects à partir des modèles 3D. Une autre contribution est le développement de deux méthodes d'extraction de ces graphes. La première est destinée à l'appariement par graphes, la seconde est utilisée par le hachage géométrique.

Concernant la précision de la localisation d'objets, nous avons mis en œuvre des méthodes existantes de recalage et avons développé une méthode hybride qui donne une meilleure performance. Le développement de la calibration automatique de la caméra à l'aide d'un robot montrera également sa qualité en terme de précision.

Dans le chapitre 1, nous présentons les principales approches de résolution du problème de la reconnaissance d'objets 3D. Tout d'abord, une étude sur les SROs actuels est faite à l'aide d'une méthode de classification donnée. Cette étude se résume par un tableau comparatif de ces systèmes (domaine d'application, performances et limites). Ensuite nous exposons notre Système de Reconnaissance d'Objets (SRO) tel qu'il est intégré au système MCIT. Enfin, une conclusion vient justifier nos choix de développement.

Le chapitre 2 expose les méthodes de modélisation, de calibration de la caméra, de modélisation des objets utilisés par MCIT ainsi que le développement des deux méthodes d'extraction des graphes d'aspects intégrés à la base de données.

Le troisième chapitre concerne l'implémentation du module de traitement d'images basé sur la notion orientée objet. Des filtres successifs sont appliqués pour l'extraction des contours. La transformée de Hough est utilisée pour trouver les segments de droite. Une amélioration de la réponse de cette transformée est apportée. L'organisation perceptive est appliquée pour trouver les relations entre les segments de l'image. Un modèle 2D de l'image (représentation des primitives : sommets, segments et chaînes de segments) est fourni.

Dans le quatrième chapitre, nous développons les deux méthodes utilisées pour l'appariement entre modèle 2D et les modèles 3D de la BD3D. L'application des invariants projectifs constitue une originalité. Une méthode hybride de recalage améliorant la précision de celui-ci est décrite. La vérification des meilleurs hypothèses

d'appariement fournit une solution de la reconnaissance. Le monde virtuel (objets reconnus) est superposé à l'image vidéo.

Le dernier chapitre montre les tests effectués dans les différentes étapes de la thèse. Ils illustrent les intérêts des méthodes appliquées et fournissent une justification et des perspectives pour les travaux à venir.

1. Etat de l'art sur les systèmes de reconnaissance d'objets (SRO)

Introduction

La reconnaissance d'objets est une tâche majeure dans la vision par ordinateur. Un système de vision doit être capable de reconnaître certains objets dans la scène. Un système « intelligent » sait quels sont les objets existants dans son environnement (identification) et où sont-ils (localisation) à chaque instant. La majorité des systèmes de reconnaissance actuels tente de résoudre ce problème en se basant sur un modèle (les objets sont connus a priori). La plupart de ces systèmes travaillent sur des objets dits *rigides* (non déformables) ou articulés.

Selon le type de capteurs utilisé, la reconnaissance peut être traitée en 2D/2D, 3D/3D (capteurs télémétriques ou stéréoscopiques) ou en 2D/3D basée sur une seule image de luminance.

Nous présentons dans ce chapitre les principales approches de résolution de ce problème. Une étude sur les SRO actuels est faite à l'aide d'une méthode de classification donnée. Cette étude se résume par un tableau comparatif de ces systèmes (domaine d'application, performances et limites). Ensuite, une présentation de notre SRO actuel, tel qu'il est intégré au système de la téléopération MCIT (*Multimedia Control Interface in Teleoperation*), est introduite. Enfin, une conclusion vient justifier nos choix de développement et résume nos contributions et perspectives.

1.1 Systèmes de reconnaissance d'objets 3D (SRO)

Nous présentons une classification des SRO afin d'avoir une vue globale de leurs aspects. Ensuite et selon cette vue d'ensemble, nous décrirons les SRO les plus connus.

1.1.1. Classification des SRO

De nombreuses approches au problème de la reconnaissance d'objets 3D ont été développées dans la littérature. Elles peuvent être classifiées selon les méthodes et techniques mises en œuvre :

- Acquisition et représentation informatique du modèle 3D,
- Approche utilisée pour l'appariement

Dans les paragraphes suivants, ces méthodes et techniques sont décrites ainsi que leur influence sur la classification des SRO.

1.1.1.1. Acquisition et Représentation Informatique du modèle 3D

Les méthodes les plus connues pour l'acquisition du modèle 3D sont la CAO, les capteurs actifs (télémètres laser) et les capteurs passifs (caméra unique, stéréo ou plus). Elles impliquent des méthodes de représentation de données adaptées.

- Méthodes basées sur la CAO

L'approche *Conception Assistée par Ordinateur* (CAO) est la plus dominante car elle décrit un modèle 3D sans erreur de capture ; elle est applicable aux environnements industriels. En revanche, elle possède des inconvénients majeurs :

- L'implication d'un opérateur humain lors de la modélisation de la scène,
- la difficulté d'application pour des objets naturels.

Plusieurs types de représentation du modèle 3D adaptés à la CAO existent (cf. 2.3.1):

- volumique, la géométrie constructive des solides (terme anglo-saxon CSG : *Constructive Solid Geometry*) consiste en la représentation de l'objet à modéliser en différents éléments simples qui s'appellent *primitives* et à les lier par des opérateurs ensemblistes (union, intersection, etc.)
- Surfâcique, la représentation par les bords (BREP : *Boundary REPresentation*) permet de construire un modèle face/arête/sommet.

Une autre classification de la modélisation d'objets peut s'obtenir en considérant 2 types :

- la représentation par l'objet lui-même (centré objet),
- la représentation par les différentes vues 2D de l'objet (centré vue).

Le second type est plus proche de la représentation des scènes (qui est généralement en 2D). Ce type représente un objet par un graphe de différentes vues possibles avec des relations entre elles. Ceci simplifie le processus d'appariement, mais pose plusieurs problèmes : combien de vues faut-il prendre pour que l'objet soit représenté d'une manière optimale ? En effet, plus le nombre de vues est important et plus d'informations seront obtenues, mais plus l'appariement sera complexe. De nombreux travaux ont porté sur ce thème, aboutissant vers une même conclusion : l'idéal est d'avoir un ensemble minimal de vues d'un objet qui soient topologiquement différentes (aspects). Les *graphes d'aspects* (GA) peuvent être 2D ou 3D selon le type d'entrée et de pré-traitement dans le système. Dans notre application, c'est une représentation BREP qui sera utilisée, accompagnée d'un graphe d'aspects généré automatiquement hors-ligne.

- Méthodes basées sur des mesures capteurs

La méthode des capteurs télémétriques est plus facile à employer mais elle introduit des mesures imprécises et incertaines dans la phase de reconnaissance. Il faudra toujours répondre à certaines questions lors de l'acquisition telles que : «combien de vues faut-il pour couvrir toutes les informations nécessaires ?» ou «comment faut-il segmenter les images pour obtenir les descriptions surfâciques ou volumiques de l'objet ?». Ce sont des questions auxquelles il est difficile de répondre et qui constituent des problèmes ouverts en recherche.

Les méthodes d'acquisition des modèles 3D en utilisant des capteurs photométriques (une seule caméra statique ou en mouvement, avec ou sans modélisation des sources lumineuses, deux caméras ou plus), sont les plus difficiles à mettre en œuvre. En effet, elles posent les mêmes problèmes que les méthodes appliquées aux capteurs actifs en sus du problème de la projection perspective et de la perspective inverse. Plusieurs techniques ont été utilisées pour calculer la position 3D (techniques stéréos, séquences d'images, etc.) Ces méthodes sont rarement employées pour l'acquisition des modèles 3D dans les SRO. Mais, elles sont prometteuses car moins onéreuses, plus générales et adaptées à de nombreux environnements.

La représentation des scènes référencée capteurs peut se décomposer en deux types selon ceux-ci (actifs ou passifs). Le problème décisif de la représentation est la segmentation des données en primitives fiables pour la reconnaissance. Or, le processus de segmentation est déterminé par le type de représentation utilisé. En général les représentations 3D sont surfâciques ou volumiques.

La capture passive des scènes est la plus proche du système de vision humaine. L'utilisation des images de niveaux de gris introduit elle aussi le problème de segmentation. Les deux principales méthodes utilisées à ce niveau sont l'approche contour et l'approche région. Il existe d'ailleurs des systèmes qui mélangent les deux approches.

1.1.1.2. Principales approches d'appariement

L'appariement ou la mise en correspondance entre la scène et son modèle 3D est le point crucial d'un SRO. L'approche de l'appariement utilisée dans un SRO marque souvent ses performances. Nous présentons ci-dessous un aperçu de la complexité algorithmique de ce problème, puis les principales approches de résolution.

- Complexité de l'appariement

Soient :

O : l'ensemble des caractéristiques d'un objet (sommets et/ou arêtes et/ou faces ...),

D : l'ensemble des caractéristiques des données de son image,

n : le nombre des caractéristiques de l'objet,

m : le nombre des caractéristiques des données de son image,

$l = \min(n, m)$; $m \geq n$ car il y a souvent plusieurs objets dans la scène, dont le modèle 3D n'est pas connu.

Une solution optimale peut être trouvée en cherchant tous les appariement possible. Du fait d'occlusions et de caractéristiques du capteur utilisé, il faut comparer un sous-ensemble (de taille i) de D et de O [HORAUD 1993].

Le nombre de possibilités d'appariement entre O et D est :

$$N = \sum_{i=k}^l C_n^i \cdot C_m^i \cdot i! = \sum_{i=k}^l \frac{m!}{(m-i)!i!} \cdot \frac{n!}{(n-i)!i!} \cdot i! = m!n! \sum_{i=k}^l \frac{1}{(m-i)!(n-i)!i!}$$

k représente le nombre d'appariement minimal pour trouver une transformation objet/capteur.

N s'accroît rapidement en fonction du nombre d'objets dans la base.

Exemple : Soit un objet contenant 18 sommets, l'extraction de ces sommets dans une image 2D fournit 11 points. Le nombre de possibilités d'appariement est :

$$N = \sum_{i=3}^{11} C_{11}^i \cdot C_{18}^i \cdot i! = 4.34 \cdot 10^{12}$$

Pour trouver une solution optimale, il faut vérifier toutes ces possibilités.

- Solutions

Deux techniques d'appariement prédominent :

- l'approche graphe
- l'approche hachage géométrique

L'approche basée sur les règles d'évidence peut compléter les deux précédentes.

1. Travaux basés sur l'approche graphe :

Dans l'approche graphe, les nœuds représentent les primitives de l'objet et les arêtes représentent les relations entre elles. Le problème se ramène donc à un problème d'isomorphisme, puisque la scène est en général un sous-graphe du graphe de l'objet. Malheureusement ce problème NP-complet¹ est très difficile à résoudre et devient insoluble quand le nombre de modèles dans la base est important. Les essais pour réduire la taille du graphe utilisé pour l'appariement ont conduit à des différentes techniques

Bolles et Fischler [BOLLES 1981] ont proposé une méthode permettant l'extraction aléatoire d'un ensemble de primitives 2D qui sont appariées avec des primitives 3D. L'*alignement* obtenu (transformation rigide) est généralisé à toute l'image. Le processus est réitéré jusqu'à la convergence de l'algorithme.

Bolles et Cain [BOLLES 1982] procèdent en extrayant en premier dans une image une primitive caractéristique et discriminante telle que coin, trou ..., appelée "*link focus*". L'appariement est poursuivi avec les primitives 2D voisines de la primitive caractéristique. Cette technique fut également utilisée par Bolles et Horaud [HORAUD 1984], [BOLLES 1986] pour la reconnaissance d'objets 3D à partir des données de profondeur.

Ayache et Faugeras [AYACHE 1986] ont proposé l'utilisation d'un triplet de segments ayant un sommet commun. Cette technique permet de réduire considérablement le nombre d'appariements possibles.

Lowe [LOWE 1987] a proposé l'amélioration du traitement d'images. Il a établi des critères de regroupement des indices 2D afin d'obtenir des primitives consistantes qui facilitent l'appariement. Par ailleurs, un degré de probabilité est attribué à une hypothèse d'appariement afin de réduire l'espace des solutions et enfin, une reconstruction 3D est effectuée pour vérifier les hypothèses d'appariement.

Pampagnin et Devy [PAMPAGNIN 1990, 1991] ont introduit le *Graphe de Compatibilité* qui regroupe dans ses nœuds toutes les hypothèses d'appariement entre une chaîne de segments 2D et une face d'un aspect du modèle 3D dont les arcs représentent les compatibilités entre hypothèses d'appariement. L'utilisation du Graphe d'Aspects permet de limiter la taille du graphe de compatibilité.

Grimson et al [GRIMSON 1984, 1987] ont proposé une méthode de *recherche arborescente*. L'arbre est constitué de tous les appariements possibles. Un nœud de l'arbre contient une hypothèse de correspondance entre une primitive 2D et une primitive 3D. Le problème est de rechercher dans l'arbre, la suite d'hypothèses qui

¹ Ensemble des problèmes pouvant être résolu par des algorithmes Non déterministes en temps Polynomial. *Comple* : veut dire qu'il n'existe pas un algorithme P efficace pour le résoudre [SEDGEWICK 1990].

permettent une mise en correspondance correcte. La méthode est étendue en appliquant des contraintes pour limiter l'espace de recherche [GRIMSON 1991, 1993].

2. Travaux basés sur l'approche du hachage géométrique :

Lamdan et Wolfson [LAMDAN 1988] ont introduit le *Hachage Géométrique* (HG). Il consiste en la création hors ligne d'une *table de hachage* (TabH) et une reconnaissance en ligne. La table de hachage contient les différents aspects des objets du modèle 3D, exprimés dans des bases différentes (une base peut être représentée par 2, 3 points ou arêtes) [WOLFSON 1992]. Cette expression est basée sur des invariants ou des quasi invariants géométriques [BINFORD 1993] et [GROS 1995]. Pour chaque entrée dans cette table, il y a une liste des modèles susceptibles de satisfaire l'appariement. La reconnaissance en ligne est obtenue par mise en correspondance de points 2D, exprimés à leur tour dans différentes bases, avec le contenu de la table de hachage. Toutes les hypothèses sur l'appariement sont utilisées dans une étape de vérification. Cette approche est présentée en détail au quatrième chapitre (§4.3).

Cette approche a l'avantage de réduire l'espace de recherche des modèles possibles, spécialement dans les grandes bases de données (problème d'*indexage*). Son défaut est l'augmentation rapide de la taille de la table de hachage en fonction du nombre de modèles.

La *Transformé de Hough Généralisée* (THG) ou *Clustering* a beaucoup de similarité avec le hachage géométrique. Elle utilise l'espace de transformation objet/capteur au lieu de l'espace des coordonnées géométriques. Une table d'*indexage* est construite hors ligne. Chaque entrée de cette table (transformation donnée) est associée à une liste de triplets (objet, aspect, base de transformation). Plusieurs bases sont choisies en ligne. Les transformations objet/capteur correspondantes sont calculées et utilisées pour accéder à la table d'*indexage*. Elles fournissent des votes pour les triplets de la table (hypothèses). Cette méthode a l'avantage de fournir une transformation grossière avec chaque hypothèse mais elle nécessite une table à plusieurs dimensions. Wolfson [WOLFSON 1992] utilise cette méthode pour reconnaître un objet articulé. Gros [GROS 1995] l'utilise avec le hachage géométrique pour améliorer la robustesse de son algorithme de reconnaissance.

3. Travaux basés sur le raisonnement par évidence :

Jain et Hoffman [JAIN 1988] ont développé une méthode utilisant le *Raisonnement par Evidance* pour reconnaître un objet 3D dans une image de profondeur. Pour cela, ils utilisent un système à règles de production. La prémisse d'une règle est composée de conditions sur les différentes primitives de l'objet et la partie action attribue un degré de probabilité sur l'appartenance de l'objet à la base de données. Cette approche souffre d'un problème majeur, celui de l'incompatibilité d'*indexage* pouvant conduire à deux objets identiques, en termes d'attributs mais structurellement différents.

1.1.2. Quelques systèmes de reconnaissance d'objets

Les types d'objets ou images utilisées sont différents d'un SRO à un autre. La plupart des SRO utilisent une base de modèles réduite à deux ou trois objets voire un seul objet. Les images d'entrée sont souvent prises dans des conditions spéciales (fond noir, objets séparés...).

Différents systèmes de reconnaissance d'objets 3D sont décrits dans les paragraphes suivants :

- ACRONYM

C'est un système de reconnaissance d'objets, basé sur des images d'intensité et développé par Brooks [Brooks 81,83]. L'acquisition des modèles 3D s'appuyant complètement sur la CAO est présentée dans des primitives volumiques appelées les Cônes Généralisés. L'appariement est basé sur l'approche graphe.

Des segments sont extraits d'une image d'entrée ayant plusieurs objets. La base des modèles utilisés possède un ou deux objets seulement. Les tests montrent que le système est capable de trouver des avions à partir de la photo aérienne d'un aéroport, mais il est difficile d'évaluer ses capacités vis à vis d'autres domaines d'application. Les techniques de raisonnement géométrique sont utilisées pour déduire des invariants observables, qui sont typiquement formés par des ellipses et des rubans, pour prédire leur existence dans une image. L'appariement se fait en local, d'abord sur les primitives trouvés dans l'image, puis avec les contraintes, sur le modèle. On cherche des interprétations globales.

ACRONYM souffre de quelques limites : les modèles 3D sont entièrement acquis par CAO, la base de modèles 3D est très réduite et finalement, en utilisant un grand nombre d'objets la méthode d'appariement, pour fonctionner, nécessiterait des améliorations.

- SCERPO

(Spatial Correspondence, Evidential Reasoning and Perceptual Organization)

SCERPO est un SRO basé sur des images d'intensité, [Lowe 87]. Les modèles 3D sont acquis et présentés par le modèle filaire en utilisant la CAO. Les objets sont modélisés par des formes polyédriques. Les scènes d'entrée contiennent plusieurs instances d'un seul objet (ex. rasoir mécanique) et la base de données contient un seul modèle.

La procédure commence par la détection des contours avec la méthode de passage par zéro du laplacien (opérateur Sobel) puis, par le suivi des chaînes de contour et la segmentation par découpage récursif. La théorie de l'Organisation Perceptive s'applique sur les segments pour trouver des relations invariantes (indépendantes de la scène). Les relations recherchées sont la connexité, le parallélisme et la colinéarité ; des groupements dans un niveau plus élevé sont aussi visés (parallélogrammes ou autres formes). A chaque groupement perceptif est

assigné une valeur de probabilité, qui exprime la présence d'un invariant réel dans la scène ou d'un simple hasard.

Les groupements perceptifs sont pré-calculés, hors-ligne pour chaque objet 3D et en ligne pour la scène. L'appariement se fait donc, entre les groupements du modèle 3D et ceux de la scène selon leurs valeurs de probabilité. La méthode de vérification de l'appariement développée par Lowe lui-même est une méthode d'optimisation non-linéaire des distances entre les sommets du modèle et leurs segments trouvés dans l'image.

Les limites de SCERPO sont liées au fait, qu'un seul objet de la base est traité à la fois et qu'il doit être polyédrique. Si on augmente le nombre d'objets, il faudra chercher des groupements perceptifs plus complexes pour réduire l'espace de recherche car aucune contrainte topologique ni géométrique, n'est appliquée pour l'appariement. La méthode de vérification semble un peu lourde et souffre de problèmes de convergence.

- SRO de Ikeuchi et Kanade

Ikeuchi et Kanade présentent un système de génération automatique de programmes de reconnaissance, développé à l'Université de Carnegie-Mellon [Ikeuchi 86, 87A, 87B, 88A, 88B]. Ce système adopte des techniques pour compiler automatiquement un modèle d'un objet, dans une stratégie de reconnaissance visuelle, pour identifier et localiser l'objet dans l'espace tridimensionnelle.

Un modèle CAO d'un objet décrit l'ensemble des caractéristiques qui peuvent être mesurées; par ailleurs un modèle de capteur fournit des caractéristiques, détectées simultanément par le capteur. En regroupant les apparences de l'objet topologiquement équivalentes, la reconnaissance revient à résoudre deux problèmes :

- changement d'aspect : classifier l'image (apparence vue par le capteur) dans un des aspects.
- changement linéaire de forme : localiser l'objet (en position et en orientation).

Ainsi ils créent, à partir des étapes suivantes, un arbre d'interprétation constitué de deux parties, pour résoudre chacun des problèmes séparément :

1. Extraire les aspects possibles du modèle géométrique de l'objet.
2. Dédire les branches pour classifier le changement des aspects.
3. Déterminer les caractéristiques à utiliser pour brancher les nœuds de l'arbre.
4. Déterminer les caractéristiques à utiliser pour résoudre le problème de localisation.

Etant donnée la distance caméra/objet, le modèle de la caméra est considéré parallèle par rapport aux dimensions de l'objet. L'objet étant situé au centre de la sphère, pour exprimer les différentes directions de vue, le système extrait les différents aspects du modèle 3D en utilisant la sphère de Gauss. Ils introduisent la notion de détectabilité, se basant sur les caractéristiques du capteur utilisé, pour déterminer les faces visibles par le capteur dans chaque aspect.

Les aspects extraits sont classifiés ensuite dans un arbre d'interprétation en se basant sur certaines règles, ce sont les modèles de travail. Ces modèles sont ordonnés selon leur coût de calcul : l'aire d'une face, l'inertie d'une face, le nombre des faces voisines, etc. A chaque modèle de travail appliqué à une caractéristique donnée, dans une configuration quelconque, est associé un degré de confiance. En analysant ce degré de confiance pour chaque aspect, une stratégie de classification est définie qui donne les meilleurs résultats espérés.

Le système utilise un capteur stéréo photométrique dual (2 caméras avec 3 sources de lumière). Il est donc possible de recouvrir les informations 3D de la face observée en 2D, à l'aide d'une transformation calculée à partir des normales aux surfaces observées.

Une fois que la classification de l'objet dans l'arbre des aspects est complétée, la détermination de l'attitude précise de l'objet pour chaque aspect est nécessaire. Le modèle de la caméra utilisé étant parallèle, trois degrés de liberté sont à calculer : une rotation autour de la direction de vue Z et deux translations dans un plan perpendiculaire à l'axe de Z. Le système utilise l'image de Gauss étendue EGI. C'est une sphère entourant l'objet dont la représentation des faces se fait par des points sur la surface de la sphère. Chaque point est pondéré par une masse proportionnelle à l'aire de la face.

Une application avec succès de ces techniques a été menée sur deux objets, un seul à la fois. Mais on ne dispose pas d'évaluation claire sur le pourcentage de reconnaissance.

- Ce système utilise les notions de détectabilité et de plausibilité pour classifier et réduire le nombre d'aspects et par conséquent, le coût d'appariement. L'appariement se fait en utilisant les informations 3D extraites de l'image.
- Les caractéristiques des régions sont prises en compte pour faire l'appariement, donnant ainsi la possibilité de manipuler des objets surfaciques ou concaves.
- Ce système représente une restriction majeure de l'effet de la projection parallèle utilisée, ce qui restreint les programmes générés à des capteurs éloignés de la scène et fixés dans le site de travail.
- Le système reconnaît plusieurs instances d'un seul objet dans la base. Il semble très difficile de le développer pour traiter plusieurs objets à la fois.

- SRO de Flynn et Jain

Flynn et Jain [Flynn 92] décrivent un SRO basé sur des images télémétriques. Les modèles sont acquis par CAO et représentés dans des graphes basés sur des surfaces, dont 3 types sont utilisés : planes, cylindriques et sphériques. L'appariement est de type hachage géométrique. L'image d'entrée contient un seul objet et la base de données contient 23 modèles. Les tests consistent à générer 5 vues de chaque objet autour d'une sphère, chacune de ces vues est donnée au système, elles sont toutes parfaitement reconnues car se sont des objets assez simples et le nombre d'hypothèses générées est assez élevé.

Les tables de hachage sont générées automatiquement en utilisant des invariants extraits des images 3D (descriptions surfaciques : planaire, cylindrique et sphériques). Des groupes de 3 surfaces (plan-plan-plan, plan-cylindre-sphère, etc.) sont générés et utilisés comme entrée aux tables de hachage. Ensuite, l'image d'entrée est segmentée en surface et groupée en triplets. Ces triplets donnent toutes les hypothèses possibles sur les modèles 3D qui seront soumises à la procédure de vérification.

Le système est limité à des objets pré-modélisés et testé par des scènes contenant un seul objet donnant parfois une centaine d'hypothèses à vérifier. Le problème de scènes à plusieurs objets n'a pas été exploré, il risque d'engendrer une explosion d'hypothèses.

- SRO de Siebert et Waxman

Dans [Siebert 92] est décrit un système d'apprentissage et de reconnaissance d'objets à partir de plusieurs vues 2D (silhouettes ou aspects de l'objet). Un modèle 3D (avion) est acquis dans la base à partir d'un ensemble de vues 2D (séquences temporelles de l'avion dans un champ). Le modèle est représenté par un réseau (ou graphe) de ses aspects. L'approche d'appariement est une sorte de matrices de hachage. La base de données du système contient 3 objets et les images d'entrée doivent avoir un seul objet sur fond noir.

Pour chaque image, la silhouette de l'objet est trouvée, les coins étant situés sur les bords. Ils sont utilisés pour représenter l'objet par un vecteur d'attributs invariant dans un espace de transformation affine 2D (position, orientation et échelle). Les réseaux d'aspects sont construits en groupant les aspects trouvés, pour chaque nouvelle image, si l'aspect se situe à une distance assez grande d'un des groupes déjà enregistrés, un nouveau groupe est formé. Chaque objet est représenté par une matrice de transition d'aspects.

La reconnaissance d'une image commence par trouver son aspect puis, selon des critères d'évidence, par chercher les aspects les plus proches parmi toutes les matrices de transition. Les évidences sont accumulées au fur et à mesure que des nouvelles séquences sont entrées. L'objet présentant le maximum d'évidences est considéré comme étant le résultat de la reconnaissance.

Bien que le temps de reconnaissance puisse se réduire grâce à la nature parallèle du système, l'espace nécessaire de mémoire augmente rapidement pour chaque nouvel objet ajouté à la base. Les images d'entrée ne posent ni le problème de séparation du fond, ni d'occlusion.

- SRO de Dickinson, Pentland et Rosenfeld

Un SRO utilisant des vues 2D est apparu dans [Dickinson 92]. Les objets sont modélisés par CAO en utilisant des primitives volumiques. Les scènes sont des dessins 2D contenant des lignes parfaites représentant un ou

plusieurs objets. L'appariement est de type indexation (hachage) mélangé à des techniques de graphe. La base de données contient une dizaine d'objets.

Le système utilise une dizaine de primitives volumiques qui sont représentées d'une manière hiérarchique dans des graphes d'aspects : aspects, faces et frontières (segments groupés). Au premier niveau, se trouvent les primitives elles-mêmes. Au second niveau, il y a tous les aspects (probablement partagés entre les primitives). De même pour les faces (rectangulaires, cylindriques, etc.) qui sont dans le troisième niveau. Les frontières des faces (lignes parallèles, curvilignes, symétriques, etc.) sont dans le dernier niveau. A chaque élément dans la hiérarchie est assigné une probabilité de visibilité calculée à partir du modèle 3D.

La première phase de reconnaissance commence par une segmentation de type région de l'image (le système n'a pas encore employé le traitement d'images) et les contours régionaux sont segmentés selon leurs discontinuités de courbure. Une face se retrouve dans un graphe où les nœuds représentent les segments délimitant la face et les arcs, les relations entre les segments adjacents.

Les graphes de faces extraits ainsi sont appariés, dans la seconde phase, avec tous les graphes possibles de faces des objets 3D. Cet appariement donne des hypothèses de face auxquelles sont associées des probabilités. Si aucun appariement n'est trouvé pour une face, il faut descendre la hiérarchie pour trouver des appariements au niveau des segments de bords qui donnera ensuite, des hypothèses sur la face. On construit un graphe d'appariement de face où chaque nœud représente une face avec ses correspondants. Les arcs représentent les relations géométriques entre les faces adjacentes.

Ce dernier graphe est apparié, dans la troisième phase, avec tous les graphes d'aspects de la base, générant des hypothèses d'aspects qui sont vérifiées pour ne garder que les plus plausibles. Un graphe d'aspect de l'image d'entrée est ainsi produit et un processus similaire est appliqué pour générer des hypothèses de primitives (graphe de primitive décrivant la scène).

Les attributs des primitives qui sont fortement connectés dans la scène donnent des indices pour explorer une table de hachage de la base (cette table est construite hors ligne d'une façon similaire). Ceci donne des hypothèses finales sur les objets de la scène qui sont passées à une phase de vérification.

Ce système est capable de reconnaître des scènes à plusieurs objets, son approche hiérarchique réduit considérablement l'espace d'appariement. Il porte sur un schéma flexible, applicable sur des images d'intensité comme sur des images de profondeur. Mais, il ne traite pas le problème de bas niveau (extraction de régions, de contour), il reste donc à tester sa performance à des scènes réelles.

- PARVO

(Primal Access Recognition of Visual Objects)

PARVO est un système de reconnaissance d'objets génériques à partir des images de type filaire [Bergevin 93]. Les modèles sont acquis par CAO et représentés par leurs primitives volumiques. La stratégie d'appariement est

de type graphe. Des scènes de tests, contenant un seul objet à la fois parmi 23 objets assez complexes, sont reconnues à 74% avec une bonne performance.

Ce système procède à une reconnaissance par composants, les objets sont représentés dans des graphes contenant leurs primitives et les relations entre elles. L'organisation perceptive est appliquée à l'image d'entrée pour extraire des primitives volumiques, donc un graphe semblable à ceux de modèles 3D. Le problème d'appariement revient donc à un problème d'isomorphisme de sous-graphes.

PARVO souffre de problèmes fondamentaux : il lui faut une image déjà presque parfaitement segmentée contenant un seul objet. Les primitives choisies étant de type volumique, elles ne sont pas facilement extractibles, surtout en cas d'occlusion. Pourtant, le système ne pose pas le problème d'acquisition des modèles 3D puisqu'elle est faite entièrement par CAO.

- SRO de Jain et Hoffman

Jain et Hoffman [Jain 88] utilisent le raisonnement par évidence mentionné auparavant dans les approches de l'appariement. Caelli et Dreier [Caelli 94] plus tard développent une extension du même système. Les modèles 3D sont acquis par des images télémétriques ou synthétisées et représentées par un graphe basé sur une approche surfacique. Le système utilise différentes méthodes pour la génération des règles et pour le raisonnement. Les scènes sont des images de profondeur et contiennent un seul objet tandis que la base de données contient, quant à elle, entre 5 et 25 objets. La reconnaissance se fait dans 80% des tests reportés.

Les images de profondeur de chaque objet sont segmentées et les attributs de leurs parties sont utilisés pour calculer des intervalles. Les règles obtenues ainsi, sont utilisées pour estimer une classification d'une image donnée. Une image de profondeur est segmentée, les points sont groupés selon leurs coordonnées 3D et les coordonnées des normales à la surface sont calculées en chaque point. Chaque portion de surface est donc classifiée planaire, convexe ou concave ; les frontières entre les portions sont classifiées aussi selon les distances entre elles. Dans le cas de sur-segmentation les portions voisines sont fusionnées selon leurs classes. Trois types d'informations caractérisent donc chaque objet : information morphologique décrivant l'image de profondeur en général, description des portions de surface et description des relations (frontières) entre les portions. Ces informations sont utilisées pour donner une règle d'évidence pour chaque classe d'objets. Les règles d'évidence sont générées automatiquement à partir de la base de données.

Une image d'entrée sera associée à la classe qui donnera l'évidence maximale.

Les tests ont montré le succès de ce système pour des images de profondeur d'un seul objet isolé (la base contient environ 20 objets). On rencontre cependant quelques problèmes pour entraîner le système à reconnaître des scènes comportant plusieurs objets.

- MORSE

Mundy et al.[Mundy 94, 95] décrivent un système complexe utilisant des invariants géométriques pour la description et l'appariement. Les modèles 3D sont acquis à partir de plusieurs scènes 2D, ils sont représentés par des invariants géométriques basés sur différents types de symétries et des invariants projectifs de surfaces. La scène d'entrée est une image 2D d'intensité.

MORSE utilise des techniques standards : segmentation, groupement, génération d'hypothèses et vérification. Les caractéristiques niveau pixel sont groupées en contours continus qui sont, à leur tour, segmentés en courbes. Leurs concavités sont aussi calculées. Ensuite, un groupement générique est effectué pour trouver des incidences, colinéarité, etc. Chaque classe de groupement possède un moteur de recherche essayant de trouver des groupements génériques propres à la classe. L'indexation et la vérification d'hypothèses sont faites par une série de tables de hachage, une par classe. La vérification s'effectue sur différents niveaux de la formation d'hypothèses.

MORSE étant en cours de développement, il est difficile d'évaluer sa performance, surtout avec des objets plus complexes que les tests effectués.

- FCRG

(Fuzzy Conditional Rule Generation)

McCane [McCane, 1996] présente FCRG « *Fuzzy Conditional Rule Generation* ». C'est un système d'apprentissage et de reconnaissance à partir d'images stéréo d'intensité. Les paires d'images sont traitées, segmentées et des attributs (profondeur, longueur, etc.) sont calculés. Deux principaux modules sont implantés pour le traitement : traitement stéréo et segmentation.

Dans le premier, un algorithme regroupe les contours dans un espace d'attributs selon la proximité du plus proche voisinage. Ces groupes sont appariés dans l'espace image pour vérifier leur plausibilité. Si l'appariement échoue, le contour est subdivisé en deux et ce, plusieurs fois si nécessaire. Le second est une approche adaptative multi-échelle, basée sur des contours/régions pour segmenter l'image en régions fermées. Les attributs sont calculés à partir des deux modules.

Le système groupe les attributs unitaires de toutes les primitives de l'ensemble des vues, de tous les objets dans un espace U. Cet espace est subdivisé en sous-espaces U_i , selon des classes. Pour tout U_i contenant des descriptions de plus qu'une classe, ses relations binaires avec les autres groupes sont calculées pour former un espace UB_i . Cet espace est subdivisé, à son tour, en sous-espaces UB_{ij} et la procédure se répète jusqu'à ce que tous les groupes aient une classe unique. Un système classificateur de règles floues permet de classer les objets d'une manière flexible.

L'image stéréo d'entrée traitée de la même façon est donnée au classificateur flou. Le résultat est passé à l'étape finale pour trouver la meilleure hypothèse et la vérifier. Les tests sont effectués sur une base de 18 objets avec un pourcentage global de reconnaissance de 72%.

FCRG n'aborde pas le problème de séparation d'objets de la scène du fond (l'image d'entrée doit avoir un fond noir). En l'état actuel, il lui manque un module performant de vérification d'hypothèses.

Dans le tableau 1.1 sont résumés les principales caractéristiques des SRO étudiés ci-dessus. Il permet juger des progrès de la recherche en ce domaine et de les comparer en termes de performances par rapport à MCIT. On y retrouve :

- le nom du SRO et ses références bibliographiques,
- la méthode d'acquisition et de représentation de la base de connaissance des objets 3D et le nombre d'objets modélisés,
- type de la scène d'entrée et nombre d'objets à reconnaître,
- l'approche d'appariement,
- les performances et les limites du système.

SRO	Méthode de modélisation, nb d'objets	Type de la scène, nb d'objets	Méthode d'appariement	Performances et limites
ACRONYM [Brooks 81,83]	CAO (cônes généralisés) 2	Images de luminance >=1	Graphe + raisonnement de type géométrique pour trouver les invariants	Le système peut reconnaître un avion sur une photo aérienne d'aéroport.
SCERPO [Lowe 87]	CAO (polyèdres) 1	Images de luminance >=1	Alignement basé sur l'organisation perceptive	Le système permet de reconnaître plusieurs instances d'un même objet. Il subsiste des problèmes de convergence.

SRO de Ikeuchi et Kanade [Ikeuchi 86-88]	CAO (primitives surfaciques) 1	Images stéréos >=1	Recherche arborescente basée sur des règles d'évidence	Objets quelconques. Reconnaissance de plusieurs instances d'un même objet. Distance caméras-objet est fixe.
SRO de Flynn et Jain [Flynn 92]	CAO (primitives surfaciques, 5 vues différentes) 23	Images de luminance 1	Hachage géométrique	Nombre d'hypothèses élevé. Le système devient très lent pour reconnaître plus d'un objet.
SRO de Seibert et Waxman [Seibert 92]	Séquences de vues 2D (graphe) 3	Images de luminance 1	Hachage géométrique + règles d'évidence	Le principal inconvénient de ce système est l'espace mémoire nécessaire pour stocker chaque objet, ainsi que le temps de reconnaissance.
SRO de Dickinson, Pentland et Rosenfeld [Dickinson 92]	CAO (primitives volumiques) 10	Images prétraitées >=1	Graphes (3 niveaux hiérarchiques) + hachage géométrique	Structure hiérarchique souple. Reconnaissance des scènes à plusieurs objets avec un temps raisonnable. Manque de tests sur images réelles.
PARVO [Bergevin 93]	CAO (primitives volumiques) 23	Images de type filaire 1	Isomorphisme de graphes.	74% selon les tests reportés. Il faut une image parfaitement segmentée. Problème en cas d'occlusion.
SRO de Jain et Hoffman [Jain 88] [Caelli 94]	Images télémétriques (primitives surfaciques) 5 à 25	Images de profondeur 1	Raisonnement par évidence	80% sur les tests reportés. Difficultés si la scène contient plus qu'un objet.

MORSE [Mundy 94,95]	Scènes 2D représentées par des invariants géométriques ?	Images de luminance ?	Hachage géométrique basé sur organisation perceptive	Etant en phase de développement, il n'est pas encore possible d'évaluer ses performances.
FCRG [McCane 96]	Images stéréos (surfaces et contours) 18	Images stéréos de luminance ≥ 1	Raisonnement basé sur règles floues	Taux de reconnaissance 72%. Manque de module performant pour la vérification.
MCIT [Mallem 1993B] [Chavand 1997]	CAO (BREP) 6	Images de luminance ≥ 1	Graphe à l'aide de contraintes (topologiques, géométriques et liaisons) + hachage géométrique	Localisation < 1 mm pour une distance < 2 m. Objets polyédriques.

Tableau 1.1 : Comparaison entre les SRO actuels

1.2. Présentation du SRO développé pour MCIT

La Figure 1.1 représente la structure générale du système de reconnaissance automatique d'objets polyédriques :

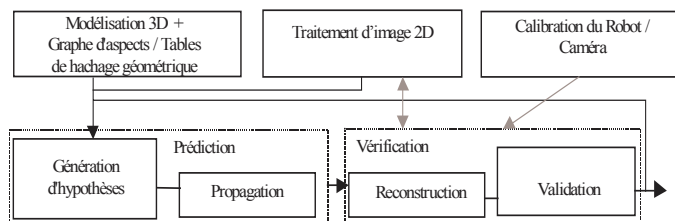


Figure 1.1 : Structure générale du système de reconnaissance d'objet du MCIT

Les chapitres suivants décrivent les parties principales du système de reconnaissance automatique que nous avons développé et qui sont :

- les calibrations du robot et de la caméra,
- la construction automatique des graphes d'aspects et des tables de hachage à partir du modèle 3D,
- le traitement de l'image, à l'issue duquel un modèle 2D de la scène est établi,
- l'appariement entre le modèle 2D et le modèle 3D qui permet la génération d'hypothèses classifiées par un score (prédiction),
- la vérification des meilleures hypothèses par les méthodes de recalage développées.

- Calibration de capteurs

Ce module est décrit dans le chapitre 2. Il fournit les calculs nécessaires pour les calibrations du robot et de la caméra, notamment la calibration automatique de la caméra que nous avons implémenté pour avoir une meilleure précision (cf. 5.1.1.2).

- Modélisation 3D

Les modèles 3D utilisés de la BD3D concernent, à l'état actuel, les objets polyédriques. Ils sont saisis hors ligne, soit par une méthode de type CAO, soit par des méthodes de modélisation utilisant la caméra et le télémètre laser (cf. 2.3.2). La présentation de ces modèles est de type BREP. Les primitives des objets 3D sont les faces, les arêtes et les sommets connus par leur position 3D dans un repère lié aux objets.

Nous avons développé deux méthodes d'extraction des graphes d'aspects à partir des objets de la base de données 3D (cf. 2.3.3). La première (exhaustive) se fait par projection perspective du modèle sur une sphère centrée sur l'objet. Cette sphère est discrétisée par un nombre fini de facettes représentant les différents points de vue. Les graphes d'aspects obtenus par cette méthode sont approximatifs et servent la méthode d'appariement basée sur le hachage géométrique. La seconde méthode (analytique) fournit les graphes d'aspects exacts. Elle est obtenue à l'aide d'une parcellisation de l'espace 3D, basée sur les plans porteurs des faces de l'objet. Les graphes d'aspects extraits par cette méthode sont utilisés par la méthode d'appariement basée sur les graphes. Ils représentent les contraintes topologiques de visibilité qui servent à réduire le nombre d'hypothèses de mise en correspondance.

Ce module contient également une partie concernant la construction des tables de hachage. Nous avons développé une méthode permettant d'extraire une table de hachage (TabH) à partir d'un groupe d'objets 3D. Des objets peuvent être ajoutés ou déduits de la table par la suite sans affecter le reste des informations mémorisées. Cette méthode donne la possibilité de sélectionner des bases de transformation à deux ou à trois points (cf. 4.3). Pour chaque groupe d'objets il est utile de construire une table de hachage au lieu d'une table englobant tous les modèles, pour éviter l'encombrement en mémoire. Ces tables de hachages sont utilisées lors de l'appariement avec la méthode du hachage géométrique.

Les graphes d'aspects et les tables de hachage sont calculés automatiquement hors ligne.

- Traitement d'images 2D (TI)

Les primitives de la scène sont des segments de droite, des sommets et des chaînes polygonales de segments, extraits de l'image à l'aide de ce module que nous avons développé pour les objets polyédriques. Décrit dans le chapitre 3, il fournit les algorithmes nécessaires pour l'extraction des points du contour des objets présents dans l'image vidéo. La *Transformée de Hough* (TH) est utilisée pour reconnaître les segments de droite. Deux améliorations importantes ont été apportées pour améliorer la réponse de la TH. L'organisation perceptive est utilisée pour exprimer les relations entre les segments. Les sommets 2D puis les chaînes de segments sont trouvés à la base de ces relations. Ce module a été conçu à l'aide d'une librairie de traitement d'images orientée objet. Des classes représentant les filtres de traitement permettent une évolution du système en profitant de la notion d'héritage. Un modèle 2D de l'image est produit en sortie.

- Appariement 2D/3D

Une première méthode d'appariement a été développée au CEMIF¹, elle est particulière car elle suppose une connaissance géométrique a priori. Cette méthode utilise la technique de la recherche arborescente et le raisonnement par évidence. Son originalité vient de la prise en compte des contraintes physiques (liaisons) entre les objets. Elle est brièvement présentée et testée dans les résultats (cf. 5.2.1.1).

Nous avons développé deux méthodes d'appariement d'objets polyédriques basées sur la connaissance de leur modèle 3D et de leur image caméra 2D. Elles sont plus générales car n'imposent pas de contraintes physiques. Les deux sont principalement basées sur les graphes d'aspects construits hors ligne.

- La première appelée *Grappe d'Appariement* (GApp) s'appuie sur l'approche graphe. Elle permet d'apparier des faces du modèle 3D avec des chaînes de segments d'un modèle 2D.
- La méthode du hachage géométrique permet de mettre en correspondance des sommets du modèle 3D avec des sommets du modèle 2D.

L'originalité réside dans l'utilisation de l'une ou l'autre de ces méthodes selon la qualité du traitement de l'image. Ceci permet une reconnaissance de l'objet, même si le traitement de l'image ne fournit pas suffisamment de chaînes cohérentes de segments.

La méthode du graphe d'appariement consiste à :

¹ Centre d'Etudes en Mécanique de l'Île de France

- chercher des correspondances entre primitives (chaînes polygonales de la scène avec les faces des modèles) *hypothèses élémentaires*. Ce sont les nœuds du graphe,
- créer les arcs du graphe en analysant les relations topologiques (*visibilité et connexité*) entre ces hypothèses élémentaires,
- chercher les *cliques maximales* (sous graphes cohérents) extraites de ce graphe,
- les invariants projectifs sont utilisés pour réduire le nombre d'hypothèses générées,
- classer les hypothèses résultantes (cliques maximales) en fonction d'un score.

La méthode du hachage géométrique consiste à :

- chercher dans l'image des sommets 2D pouvant établir une base de transformation (2 ou 3 points),
- calculer les coordonnées de tous les sommets de l'image dans cette base,
- accéder à la table de hachage à l'aide de ces coordonnées et voter pour les triplets (modèle, aspect, base) correspondants,
- classer les hypothèses résultantes (triplets) en fonction d'un score.

Le *score* utilisé dans les deux méthodes représente le nombre de points appariés. Chacune de ces méthodes est décrite dans le chapitre 4.

- Vérification

Ce sont les hypothèses ayant le plus haut score qui sont utilisées dans la procédure de vérification. Le score étant le nombre de points appariés, la vérification nécessite un minimum de ce score pour calculer la première transformation entre le repère/modèle et le repère/image. C'est une méthode hybride de recalage (*recalage mixte*) que nous avons développée pour calculer cette transformation (cf. 4.4.1.5). Elle est basée sur trois méthodes :

- géométrique (celle de Dhome [DHOME 1989]),
- linéaire basée sur le vecteur de rotation,
- non-linéaire utilisant l'algorithme de Levenberg-Marquardt et les quaternions unitaires.

La projection de l'objet après recalage sur son image fournit de nouveaux appariements possibles n'ayant pas été donnés par la méthode d'appariement (prédictions). Le traitement de l'image se refait dans ces zones pour obtenir les primitives manquantes (s'il y a lieu). Ces éléments viennent enrichir l'hypothèse en cours.

Une hypothèse est acceptée si elle répond à un critère d'évaluation de l'erreur de reconstruction. La superposition des modèles 3D sur leur image 2D est alors réalisée.

Conclusion

Après la définition du problème de la reconnaissance d'objets, nous avons proposé des critères de classification des SRO (Systèmes de Reconnaissance d'Objets). Ils sont basés sur les méthodes employées pour l'acquisition, la représentation du modèle 3D et l'appariement entre celui-ci avec les données capteur.

Nous avons présenté dans ce chapitre les principales approches de résolution du problème de reconnaissance, suivies par un tour d'horizon des SRO actuels les plus connus. Un tableau comparatif entre ces systèmes est présenté (méthodes utilisées, domaine d'application, performances et limites). Ensuite, une brève présentation de notre système de reconnaissance actuel, tel qu'il est intégré au système de la téléopération MCIT a été abordée.

Nous avons pu remarquer les limites des SRO tels qu'ils sont actuellement :

- la plupart des SRO utilisent une base de données 3D réduite à un ou deux objets simples ou peu complexes (peu utilisent une dizaine d'objets),
- peu de SRO manipulent des scènes à objets multiples, cas auquel les objets sont bien séparés sur un fond noir (la segmentation de l'image ne pose pas beaucoup de problèmes),
- le nombre d'hypothèses générées et la taille de mémoire occupée sont sources d'augmentation du temps d'exécution (ils n'ont pas été toujours reportés),
- le taux de réussite et la précision de la reconstruction n'ont pas été abordés clairement.

Pour ces raisons, nous nous sommes fixés un objectif, essayant de résoudre quelques uns de ces problèmes sans trop compliquer le système : un système de reconnaissance de polyèdres à partir d'une seule image pour la téléopération.

Adopter les polyèdres est sans doute une simplification de cette lourde tâche. Néanmoins, on disposait au préalable des méthodes de leur modélisation. Le traitement de l'image se trouve moins compliqué mais garde toujours ses difficultés qui n'ont pas été résolues dans le monde de la vision. Dans la suite de cette thèse, nous présentons nos différentes contributions à ce système de reconnaissance : calibration automatique d'une caméra, traitement d'images, construction de graphes d'aspects, méthodes d'appariement, méthodes de recalage et enfin les résultats et les perspectives.

2. Modélisation géométrique de l'environnement

2.1 Introduction

La modélisation de l'environnement se divise en deux parties :

- la modélisation et la calibration des capteurs (caméras, télémètre etc.)
- la modélisation de la scène (objets, robots etc.)

Dans la première partie, les méthodes existantes pour la modélisation et la calibration de la caméra sont présentées. Elles ont été développées dans des thèses antérieures [LOUKIL 1993], [TRIBOULET 1996]. Nous les avons utilisées pour le développement et pour l'évaluation de la calibration automatique présentés au chapitre des résultats.

Dans la seconde partie, un bref état de l'art des méthodes de modélisations d'objets, notamment celles utilisées dans MCIT, est exposé. Nous développons ensuite deux méthodes de construction de graphe d'aspects à partir d'un modèle 3D avec une comparaison.

2.2 Modélisation et calibration de capteurs

L'objectif de la modélisation des capteurs est la détermination des relations mathématiques, entre eux et l'environnement. Celle-ci permet de résoudre les problèmes de recalage 3D, ceux de la mise en coïncidence des points de vue réels et virtuels, problèmes tous liés à la Réalité Augmentée.

La calibration des capteurs consiste à identifier les paramètres représentant les modèles définis dans l'étape précédente. Nous avons utilisé une caméra comme capteur d'intensité lumineuse de la scène.

Les modèles et les méthodes de calibration utilisés sont rappelés ci-dessous.

2.2.1 Modèles de caméra

Un modèle de caméra représente le formalisme mathématique permettant d'établir une relation analytique entre les coordonnées d'un point dans le repère du monde et les coordonnées de son image dans le repère d'affichage. Le modèle est alors constitué d'un ensemble de paramètres associés aux transformations mathématiques, nécessaires au passage, entre le repère du monde et le repère d'affichage.

Le modèle géométrique de la caméra utilisé est celui du sténopé avec et sans distorsions. Dans le modèle du sténopé sans distorsions, un point 3D $P(X, Y, Z)$ et son image $p(u,v)$ sont supposés se trouver sur un rayon optique passant par le centre de la lentille et considéré non dévié (Figure 2.1). Deux types de modèles sont distingués, le modèle géométrique direct (MGD) et le modèle géométrique inverse (MGI). Le MGD est constitué par la relation $(u,v) = f(P)$, le MGI est représenté par : rayon visuel = $g(u,v)$.

Le MGD est formé par un :

- modèle interne (Mint), composé des transformations permettant d'exprimer les coordonnées pixel, dans le plan image (Ri), du point p connu dans le plan rétinien (Rc').
- modèle externe (Mext), composé des transformations permettant d'exprimer les coordonnées du point P, connu dans le repère du monde (Ro), dans le repère lié à la caméra (Rc).

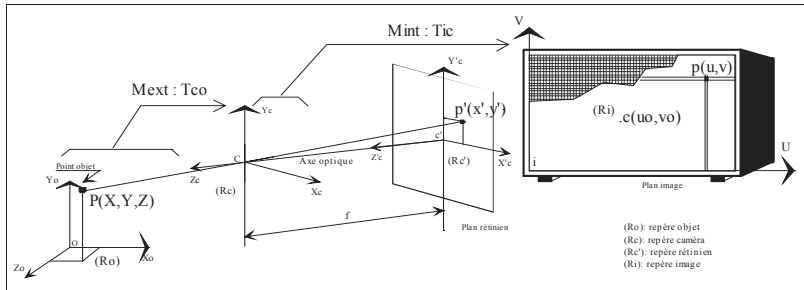


Figure 2.1 : Modèles géométriques de caméra

2.2.1.1 Modèle interne de la caméra

Le modèle interne de la caméra est représenté par les transformations permettant de passer du repère de la caméra au repère lié au plan rétinien puis, au plan image :

1. projection dans le plan rétinien : c'est la transformation qui permet de passer du repère de la caméra (Rc), au repère rétinien (Rc'). Soit P(Xc, Yc, Zc) un point dans (Rc), sa projection dans le plan rétinien donne p'(x', y') dans (Rc'). Cela donne :

$$Xc/Zc = x'/f, \quad Yc/Zc = y'/f$$

2. projection dans le plan image : c'est la transformation qui fait passer du repère (Rc') au repère (Ri). Cette transformation est une homothétie suivant les deux axes X'c et Y'c; puis un changement d'origine.

On distingue 2 modèles internes :

- modèle linéaire ne tenant pas compte des distorsions :

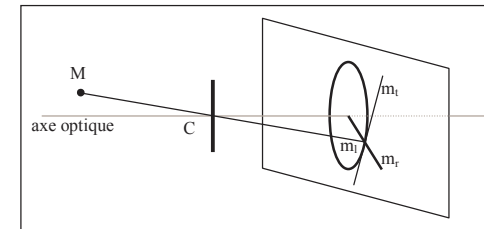
$$\begin{cases} u = ki \cdot x' + uo = ki \cdot \frac{Xc}{Zc} + uo \\ v = kj \cdot y' + vo = kj \cdot \frac{Yc}{Zc} + vo \end{cases} \quad \begin{matrix} ku = ki \cdot f \\ kv = kj \cdot f \end{matrix}$$

ce qui donne sous forme matricielle :

$$\begin{pmatrix} Zc \cdot u \\ Zc \cdot v \\ Zc \end{pmatrix} = \begin{pmatrix} ku & 0 & uo & 0 \\ 0 & kv & vo & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} Xc \\ Yc \\ Zc \\ 1 \end{pmatrix} = M \text{ int} (3 \times 4) \begin{pmatrix} Xc \\ Yc \\ Zc \\ 1 \end{pmatrix}$$

Mint(3x4) est le modèle interne, matrice de passage de (Rc) à (Ri) et les paramètres f, ki, kj, uo, vo s'appellent les paramètres internes de la caméra :

- uo, vo : représentent l'intersection entre l'axe optique et le plan image,
- ki, kj : les facteurs d'échelle exprimant l'homothétie entre plan caméra et plan image,
- f : distance focale de l'objectif.



- modèle non-linéaire tenant compte des distorsions :

Lorsque l'objectif ou le capteur présente des défauts non négligeables, le modèle précédent n'est plus pertinent. Il faut affiner le modèle du sténopé en tenant compte des distorsions. En présence d'une distorsion, l'image n'est pas exactement homologue à l'objet. Ce phénomène est causé par plusieurs défauts propres à l'objectif dont les plus importants sont des distorsions géométriques. Deux types de distorsions, radiales et tangentielles, sont distinguées.

Comme le montre la Figure suivante, l'image d'un point M devrait donner le point ml avec un modèle linéaire, mt avec un modèle à distorsion tangentielle et mr avec un modèle à distorsion radiale. Pour la distorsion radiale, la position du point mr est fonction du signe du facteur de distorsion (Kd) suivant lequel il se trouvera à l'intérieur ou à l'extérieur du cercle.

Figure 2.2 : Les distorsions radiales et tangentielles

La prise en compte de la distorsion tangentielle n'apporte pas beaucoup d'amélioration, [TSAI 1987], [CHAUMETTE 1989]. Il suffit de considérer le premier coefficient de distorsion radiale. Le modèle interne est alors modifié de la manière suivante :

$$\begin{cases} u = uo + ku \cdot \frac{Xc}{Zc} + kv \cdot kd \cdot \frac{Xc}{Zc} \cdot \frac{(Xc^2 + Yc^2)}{Zc^2} \\ v = vo + kv \cdot \frac{Yc}{Zc} + ku \cdot kd \cdot \frac{Yc}{Zc} \cdot \frac{(Xc^2 + Yc^2)}{Zc^2} \end{cases} \quad \begin{matrix} ku = ki \cdot f \\ kv = kj \cdot f \end{matrix}$$

kd est le facteur de distorsion radiale.

2.2.1.2 Modèle externe de la caméra

C'est la transformation qui permet de passer du repère (Ro) au repère (Rc). Un point de la scène ne peut être connu que par rapport à un repère lié à celle-ci, appelé *repère de travail* (Ro). Trois rotations sont possibles suivant les angles $(\theta_x, \theta_y, \theta_z)$ et trois translations sont possibles suivant le vecteur $(t_a, t_b, t_c)^t$ permettant de définir la situation de (Rc) par rapport à (Ro). Un point P, des coordonnées X,Y,Z connues dans le repère Ro, est représenté dans le repère Rc par :

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix} = \begin{pmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = M_{ext}(4 \times 4) \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

où $R_{3 \times 3} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$ et $T_{3 \times 1} = \begin{pmatrix} t_a \\ t_b \\ t_c \end{pmatrix}$

$M_{ext}(4 \times 4)$ est le modèle externe, exprimant la situation (position et orientation) de la caméra par rapport au repère objet (Ro).

Les paramètres $(t_a, t_b, t_c, \theta_x, \theta_y, \theta_z)$, qui sont déterminés à partir de $M_{ext}(4 \times 4)$, sont appelés les paramètres externes de la caméra.

2.2.1.3 Modèle global de la caméra : cas du sténopé linéaire

La matrice qui permet de passer directement du repère de travail (Ro) au repère (Ri) lié au plan d'affichage, permet de déterminer le modèle global de la caméra en composant les modèles interne et externe :

$$\begin{pmatrix} Zc.u \\ Zc.v \\ Zc \end{pmatrix} = \begin{pmatrix} ki.f & 0 & uo & 0 \\ 0 & kj.f & vo & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix} = \begin{pmatrix} ki.f & 0 & uo & 0 \\ 0 & kj.f & vo & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = M_{int} . M_{ext} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

qui s'écrit :

$$\begin{pmatrix} s.u \\ s.v \\ s \end{pmatrix} = C_{3 \times 4} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (2.1)$$

avec $s=Zc$ et $C_{3 \times 4} = \begin{pmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & c_{34} \end{pmatrix}$

$C_{3 \times 4}$ représente le modèle global de la caméra, qui exprime la relation entre le repère objet (Ro) et le repère image (Ri). Les paramètres c_{ij} sont appelés les *paramètres globaux* de la caméra.

2.2.1.4 Modèle géométrique direct de la caméra

Ce modèle permet d'exprimer les coordonnées (u,v) du point p image du point P .

La relation (2.1) est décrite par les équations :

$$\begin{cases} P^t . c_1 + c_{14} - u(P^t . c_3 + c_{34}) = 0 \\ P^t . c_2 + c_{24} - v(P^t . c_3 + c_{34}) = 0 \end{cases} \quad (2.2)$$

où :

- $(.)$: produit scalaire,
- P : vecteur de coordonnées X,Y,Z,
- c_1 : vecteur de coordonnées (c_{11}, c_{12}, c_{13}) ,
- c_2 : vecteur de coordonnées (c_{21}, c_{22}, c_{23}) ,
- c_3 : vecteur de coordonnées (c_{31}, c_{32}, c_{33}) .

La relation (2.2) permet d'exprimer les coordonnées pixel (u,v) de l'image p du point P .

$$\begin{cases} u = \frac{P^t . c_1 + c_{14}}{(P^t . c_3 + c_{34})} \\ v = \frac{P^t . c_2 + c_{24}}{(P^t . c_3 + c_{34})} \end{cases} \quad (2.3)$$

Il s'agit du modèle géométrique direct de la caméra.

2.2.1.5 Modèle géométrique inverse de la caméra

Ce modèle exprime l'équation du rayon optique passant par le point image p des coordonnées (u,v) . La relation (2.2) donne deux équations indépendantes caractérisant le modèle inverse de la caméra qui peut se mettre sous la forme :

$$\begin{cases} P^t . (c_1 - u.c_3) + c_{14} - uc_{34} = 0 \\ P^t . (c_2 - vc_3) + c_{24} - vc_{34} = 0 \end{cases} \quad (2.4)$$

et peut s'écrire :

$$\begin{cases} P^t . \overline{mu} + au = 0 \\ P^t . \overline{nv} + av = 0 \end{cases} \quad (2.5)$$

$$\text{ou } \vec{nu} = \begin{pmatrix} c_{11} - u.c_{31} \\ c_{21} - u.c_{32} \\ c_{13} - u.c_{33} \end{pmatrix}, \vec{nv} = \begin{pmatrix} c_{21} - v.c_{31} \\ c_{22} - v.c_{32} \\ c_{23} - v.c_{33} \end{pmatrix}, au = (c_{14} - u.c_{34}), av = (c_{24} - v.c_{34})$$

La relation (2.5) représente deux équations de plan où \vec{nu} et \vec{nv} sont les normales à ceux-ci.

Le rayon optique, contenant le point P et son image p , est fourni par l'intersection de ces deux plans. Le vecteur directeur du rayon optique, définissant le modèle géométrique inverse de la caméra, est :

$$\vec{r} = \vec{nu} \wedge \vec{nv} \quad (2.6)$$

2.2.2 Calibration de la caméra

La calibration de la caméra consiste à déterminer les paramètres du modèle de celle-ci. Le pré-requis est une acquisition d'un ensemble de points dans le repère objet et de leurs images. Afin de calibrer la caméra, nous avons expérimenté successivement les modèles linéaires et non-linéaire décrits dans le §2.2.1.

Pour le modèle linéaire, d'après la relation (2.2), un point donne deux équations. Afin d'identifier les paramètres globaux c_{ij} ($i=1..3 ; j=1..4$) de la caméra, il faut donc 6 points (non coplanaires, non optiquement alignés) et leurs images respectives. Pour avoir une meilleure précision lors de l'estimation de ces paramètres, par des méthodes d'optimisation mathématiques, plus de 6 points et leurs images sont nécessaires. Nous appliquons alors la méthode des moindres carrés.

Pour le modèle non-linéaire, c'est la méthode d'optimisation de Levenberg-Marquart qui est appliquée afin d'identifier les paramètres internes et externes. La rotation est représentée par le vecteur de rotation selon le formalisme de Rodrigues [CHAUMETTE 1989]. Les paramètres internes incluent le facteur de distorsion radiale [TSAI 1987]. L'estimé initial est fourni par la méthode des moindres carrés.

Dans une procédure de calibration habituelle, les coordonnées des points de la calibration sont fournies manuellement par un opérateur. Les coordonnées des points de l'image sont obtenues par pointage à l'écran. Devy et al. [DEVY 1997] utilisent plusieurs vues d'un objet de calibration 2D avec une extraction sub-pixellique des points d'intérêt afin d'augmenter la précision. Nous avons implémenté une procédure de calibration automatique, basée sur un robot à quatre degrés de liberté et sur un traitement des taches lumineuses. Les conditions expérimentales et les résultats d'évaluation de la calibration manuelle de la caméra et de celle basée sur un robot sont décrits dans le chapitre 5.

2.3 Modélisation géométrique de la scène

Un des objectifs du système MCIT est de créer des images synthétiques des objets de l'environnement et de les superposer à l'image caméra. C'est le rôle principale de la reconnaissance d'objets (cf. introduction générale et 1.2). Pour ce faire nous devons mettre en place une base de données. Cette base de données doit à tout moment représenter l'environnement de la tâche, c'est à dire doit contenir les informations nécessaires, pour répondre aux exigences et fonctions du système MCIT : reconnaissance, animation, visualisation, planification de trajectoire et de tâche.

La modélisation géométrique a pour objet de créer un modèle de la scène, ce modèle contient les données géométriques sur les entités de la scène (robots, objets) et topologiques définissant les relations entre ces entités et les constituants d'une même entité. Ces informations permettront de mettre en place une BD3D (Base de Données 3D) géométriques pour la téléopération.

Pour créer un modèle il faut définir un formalisme de représentation, puis employer une méthode pour l'acquisition des informations géométriques de ses éléments. Une autre représentation (graphe d'aspects) est exigée par les méthodes d'appariement entre ce modèle et son image.

Nous décrivons les représentations les plus connues, notamment celle utilisée par MCIT, suivies par la méthode d'acquisition de polyèdres. La dernière partie concerne le graphe d'aspects.

2.3.1 Représentation des objets 3D

Trois catégories de représentation existent : filaire, surfacique et volumique.

2.3.1.1 Représentation filaire

Le modèle *fil de fer* (*wire frame*) est historiquement la première représentation graphique à avoir été mise en œuvre. Il retient de l'objet, les coordonnées des sommets définissant ainsi les arêtes les joignant. Ne connaissant que les arêtes et les sommets, plusieurs interprétations d'un même modèle peuvent être faites.

Bien que rarement utilisée de nos jours, cette méthode a l'intérêt de permettre une création et une visualisation rapide du modèle car elle permet une modification aisée des points et des arêtes. La capacité mémoire utilisée est également très faible.

Cette représentation est aujourd'hui abandonnée, et le terme *fil de fer* est utilisé en visualisation lorsque les faces cachées de l'objet ne sont pas éliminées.

2.3.1.2 Représentations surfaciques

Ce sont les représentations référant uniquement aux surfaces de l'objet. On en distingue deux : La BREP et l'Image Gaussienne Etendue EGI.

La BREP (*Boundary REPresentation*) est une représentation du solide par ses limites (frontières) [QIANG 89]. Les surfaces envisageables peuvent être diverses: plans, surfaces de révolution, surfaces quelconques. L'objet est construit par assemblage interactif de primitives géométriques de base. Les primitives sont les éléments géométriques suivants : point, segment, polygone et polyèdre. Les liaisons entre ces primitives sont définies par des relations topologiques d'incidences, de contiguïté et d'inclusion. Ce type de modélisation permet implicitement de modéliser aussi les volumes dans la mesure où la position de la matière est connue. Dans le cas le plus fréquent des polyèdres, les limites de l'objet sont des plans. Il est alors défini par ses faces, ses arêtes et ses sommets; d'où l'appellation de modèle *Face-Arête-Sommet* (Figure 2.3). Deux façons permettent de construire le modèle:

- la première repose sur la définition d'un modèle fil de fer auquel sont associées les surfaces correspondantes.
- la seconde consiste à balayer un contour de base le long d'un parcours linéaire ou circulaire pour créer le solide correspondant.

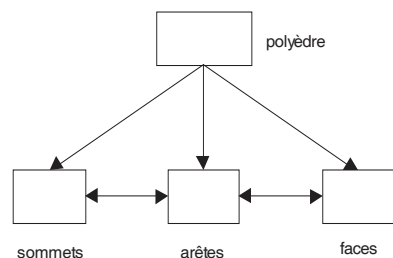


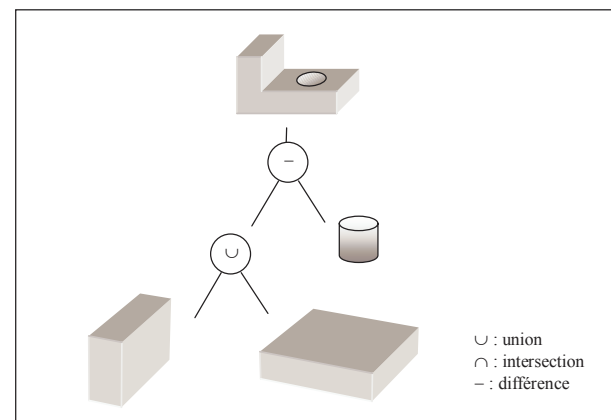
Figure 2.3 : Représentation informatique BREP d'un polyèdre

L'*Image Gaussienne Etendue* EGI proposée par Ikeuchi [IKEUCHI 1988B] est un modèle simple utilisé pour la reconnaissance d'objets. Il représente la distribution des normales aux surfaces de l'objet. L'objet est positionné au centre d'une sphère, à chaque point sur la sphère est associée l'aire de la surface dont la normale passe par ce point. Cette représentation a l'avantage d'être invariante par translation de l'objet et change d'une manière régulière par rotation. Il est noté qu'il est possible de trouver la même EGI pour deux objets.

2.3.1.3 Représentations volumiques

Ces représentations concernent le volume de l'objet. Différents modèles entrent dans cette catégorie, nous résumons quelques uns.

La représentation *constructive solid geometry* CSG : Il s'agit d'une méthode de construction d'objets par l'application d'opérations d'assemblage d'où l'appellation. La construction de l'objet se fait par combinaison de



volumes élémentaires. La démarche est semblable à celle de l'artisan, qui usine les pièces élémentaires, les assemble pour obtenir une plus complexe (Figure 2.4). Elle est décrite par un arbre de construction où les feuilles sont des solides de base et les nœuds sont des opérations comme: l'union, l'intersection, la différence, les transformations géométriques composées de translation ou de rotation. L'objet est construit à partir de primitives géométriques de base ou volumes élémentaires (cube, cylindre...) auxquels sont appliqués les opérateurs logiques afin de les assembler et de créer d'autres volumes. Les primitives géométriques de base sont paramétrées. Les données canoniques simples caractérisent chacune des primitives (centre, rayon et hauteur pour le cylindre par exemple).

Figure 2.4 : Représentation CSG simple

Le paramétrage des primitives et la variété d'opérateurs facilitent la création de nouveaux solides en quelques secondes. La simplicité des données canoniques des primitives rend la représentation des données concises, d'où gain en place mémoire. Utilisée souvent en CAO, elle ne peut être cependant exploitée directement pour la visualisation et nécessite une conversion en un modèle plus approprié (la BREP par exemple) au cours d'un processus de conversion lourd en temps de calcul.

Les volumes peuvent être représentés par *Enumération Spatiale*. Celle-ci contenant plusieurs sous catégories [PAMPAGNIN 90], nous présentons quelques unes :

- décomposition primaire de la scène en *Voxels*; dans ce cas les éléments de volume sont des cubes de taille fixée auxquels sont attribuées des valeurs de vérité concernant la présence ou non de la matière. La méthode est simple mais l'espace mémoire occupé est prohibitif.
- décomposition optimisée de la scène en *Octrees*; l'espace 3D est discrétisé de manière récursive. Si un cube n'est pas homogène (plein ou vide), il est décomposé en huit cubes plus petits et ainsi de suite. On obtient une description arborescente dont les feuilles sont des cubes "homogènes". Un exemple de l'utilisation de l'octree est la représentation de l'espace libre dans l'espace de configuration d'un robot manipulateur. Comme le voxel, l'octree demande trop de place mémoire.

2.3.1.4 La représentation utilisée par MCIT

Les plus utilisées des représentations de nos jours pour une modélisation 3D sont la CSG et la BREP. Les méthodes de représentation d'objets sont celles utilisées pour la conception de système CAO et développées à des fins de reconnaissance d'objets. La représentation CSG permet plus facilement de créer une représentation de la scène que la BREP. Cette simplicité d'utilisation se fait au détriment de la rapidité d'animation de l'image [MALLEM 90].

En effet, les données canoniques simples ayant servi à la représentation de la scène ne sont pas utilisées directement par le matériel graphique. Une conversion de ces données en points et en segments est nécessaire. De plus, le calcul des surfaces très utile pour les applications de robotique, est difficile avec la CSG, puisque les formules mathématiques de calcul de surface nécessitent la connaissance de points des frontières. Il est certain que la disponibilité des deux méthodes serait souhaitable. La CSG permettrait la saisie des données pour la création d'une représentation de la scène et la BREP l'animation et la visualisation.

La solution retenue pour le système MCIT est celle utilisant la BREP, elle est plus générale et plus proche des systèmes graphiques.

2.3.2 Acquisition des objets 3D

La coopération entre deux capteurs (caméra et télémètre 3D) est utilisée pour développer les méthodes d'acquisition [N'ZI 1995B], [TRIBOULET 1996]. Deux familles d'objets 3D sont considérées dans MCIT, les objets cylindriques et polyédriques. Nous présentons la méthode utilisée pour les polyèdres, car c'est le type d'objets qui intéresse la reconnaissance développée dans le chapitre 4. Auparavant un rappel est donné sur la signification des plans d'interprétation.

2.3.2.1 Plan d'interprétation

Le plan d'interprétation est défini à l'aide du modèle de la caméra (modèle du sténopé). Il exprime la relation entre un segment de droite réel, sa projection dans le plan image et le centre optique de la caméra (Figure 2.5).

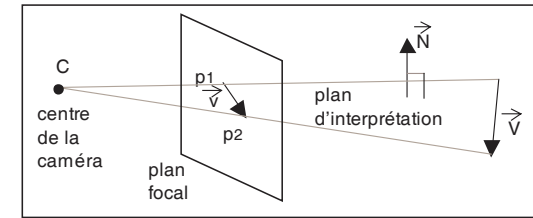


Figure 2.5 : Plan d'interprétation

Soit \vec{V} le vecteur associé à un segment de droite réel, \vec{v} son image et C le centre optique de la caméra déterminé lors de la calibration. Le plan d'interprétation est calculé à partir du vecteur \vec{v} et du point C, \vec{N} étant sa normale. Il contient également le vecteur réel \vec{V} .

2.3.2.2 Acquisition de polyèdres

La méthode d'acquisition est applicable aux polyèdres convexes (Figure 2.6). Un objet est de ce type lorsque l'angle (mesuré à l'intérieur de la masse de l'objet) entre deux faces voisines est toujours inférieur à 180° .

Un ensemble de points appartenant à la surface à modéliser est relevé avec le télémètre (Figure 2.7). On détermine l'équation du plan de cette face en appliquant les moindres carrés sur cet ensemble de points. Le coefficient directeur suivant z est toujours différent de zéro, donc une normalisation est possible. Pour une face qui n'est pas parallèle à l'axe des z, l'équation du plan est alors:

$$a \cdot x + b \cdot y + z + d = 0$$

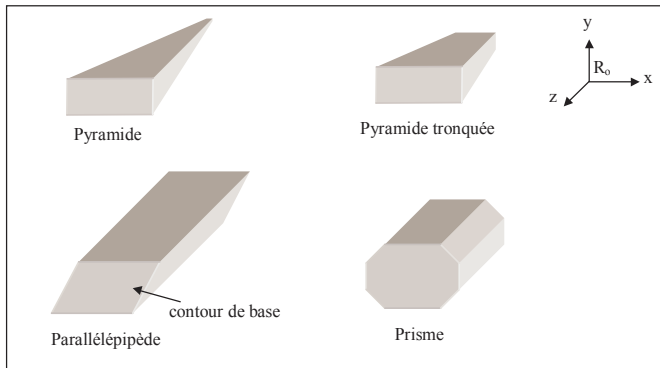


Figure 2.6 : Primitives polyédriques convexes simples

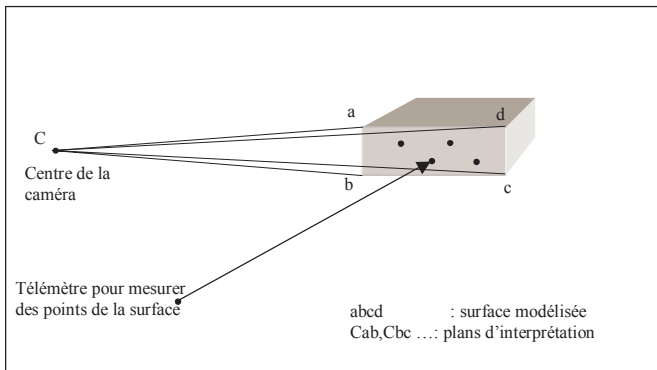


Figure 2.7 : Modélisation d'un polyèdre simple

Pour n points, on obtient n équations qui peuvent être mises sous la forme matricielle:

$$A \cdot X = B$$

avec:

$$A = \begin{bmatrix} x_1 & y_1 & 1 \\ \cdot & \cdot & \cdot \\ x_i & y_i & 1 \\ \cdot & \cdot & \cdot \\ x_n & y_n & 1 \end{bmatrix}, X = \begin{bmatrix} a \\ b \\ d \end{bmatrix} \text{ et } B = \begin{bmatrix} -z_1 \\ \cdot \\ -z_i \\ \cdot \\ -z_n \end{bmatrix}$$

Le calcul de la pseudo inverse permet alors d'obtenir les valeurs des coefficients a, b, d . L'utilisation de la caméra permet de déterminer les plans tangents (plans d'interprétation) aux contours de l'objet vus par la caméra. L'intersection du plan porteur de la surface à modéliser avec les plans d'interprétations permet de déterminer les coordonnées 3D des sommets et des arêtes de la face.

Le processus est réitéré pour toutes les faces de l'objet. Pour minimiser l'erreur du télémètre il faut orienter l'objet d'une manière que la face mesurée soit perpendiculaire à la direction de visée du télémètre [BARAT 1997].

Cette méthode peut être généralisée pour les polyèdres concaves. Le problème qui se pose est celui de trouver une configuration de l'objet devant la caméra d'une manière à voir le contour complet de chaque face. Or, cette configuration est difficile à trouver pour les faces formant une concavité.

2.3.3 Construction des graphes d'aspects pour polyèdres

Plusieurs chercheurs [IKEUCHI 1987], [STEWMAN 1988], [HANSEN 1989], [PLATINGA 1990], représentent un objet 3D par son graphe d'aspects. Le graphe d'aspects donne une description topologique du modèle 3D où chaque aspect représente des éléments géométriques (sommets, arête, face...) du modèle 3D, visibles simultanément par un observateur (caméra). Les avantages et inconvénients de l'utilisation du graphe d'aspects sont décrits par différents chercheurs dans [BOWYER 1992].

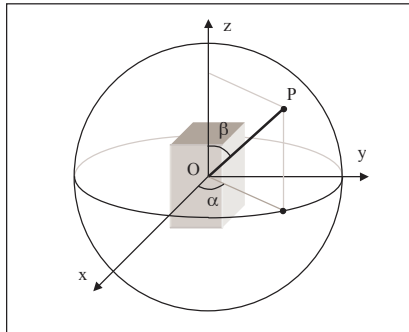
Un graphe d'aspects doit être généré pour chaque objet qui intègre la base de données des modèles 3D. Deux aspects d'un objet sont différents si l'une des primitives (sommets, arête ou face) au moins disparaît ou apparaît lorsqu'on passe de l'un à l'autre.

La notion de graphe d'aspects correspond à un ensemble de vues d'un objet topologiquement identiques. Pour chaque aspect de l'objet, il existe un ensemble de points de vues de l'espace pour lesquels l'image de l'objet observé change géométriquement, tout en ayant la même topologie. Cet ensemble de points de vue forme une région de l'espace appelé *cellule*. Lorsque l'œil de l'observateur se déplace d'une cellule à une autre, il voit la topologie de l'image de l'objet changer : des primitives disparaissent et d'autres deviennent visibles. Ce changement est appelé *événement visuel*.

Un graphe d'aspects est la représentation sous forme de graphe de la parcellisation de l'espace en cellules. Il a pour nœuds les aspects et pour arcs les événements visuels. Un résumé des méthodes existantes de calcul du graphe d'aspect et de leur complexité algorithmique, se trouve dans [PAMPAGNIN 1990].

Deux méthodes de calcul du graphe d'aspects sont développées. La première est basée sur l'utilisation de la sphère de Gauss et la seconde sur la parcellisation de l'espace par des plans pour la constitution des différentes

cellules. Chacune des deux est utilisée pour une méthode d'appariement compatible. Nous décrivons brièvement ces deux méthodes.



2.3.3.1 Méthode exhaustive

Cette méthode développée pour servir à l'appariement basé sur le hachage géométrique (§4.3.3.1), utilise les sommets du modèle 3D. Un aspect est alors représenté par un ensemble de sommets visibles simultanément.

Voici les étapes principales de l'algorithme :

- une sphère de Gauss est centrée sur l'objet et est discrétisée pour obtenir une série de points de vue repartis à sa surface,
- en chacun de ces points de vue, l'objet est projeté afin d'obtenir l'un de ses aspects,
- pour chaque aspect, les sommets cachés sont déterminés,
- la redondance est éliminée dans l'ensemble des aspects obtenus. Deux aspects sont reliés dans le graphe, si et seulement si, les points correspondants sur la sphère sont voisins.

Nous détaillons seulement les trois premières étapes, car la dernière est simple.

- Sphère de Gauss

Connaissant les coordonnées de tous les sommets du modèle 3D dans un repère absolu, le repère barycentrique R_o ayant pour origine la moyenne arithmétique des coordonnées de tous les points est établi (Figure 2.8). Le modèle 3D est replacé dans ce repère. Une sphère centrée sur le repère barycentrique est définie d'une manière qu'elle contienne le modèle.

Figure 2.8 : Sphère englobant un modèle de type boîte

Nous définissons deux angles représentant la *longitude* α et la *latitude* β permettant de parcourir la sphère. Ces angles varient :

- pour α entre 0 et 360 degrés,

- et pour β entre 0 et 180 degrés

On parvient à parcourir la sphère régulièrement et le point P associé à α et β aura pour coordonnées (en notant r le rayon de la sphère) :

$$P_x = r \cdot \sin(\beta) \cdot \cos(\alpha)$$

$$P_y = r \cdot \sin(\beta) \cdot \sin(\alpha)$$

$$P_z = r \cdot \cos(\beta)$$

Le point P représentera ainsi les différents points de vue adoptés pour projeter le modèle. Le pas de discrétisation de α et β est imposé empiriquement à 10°, il est suffisant pour obtenir quasiment tous les aspects représentatifs d'un polyèdre simple.

- Projection du modèle

Nous exprimons le modèle dans un repère orthonormé R_c d'origine P dont l'axe Z_c est normal à la sphère et dont les deux autres (X_c et Y_c) forment un plan tangent à la sphère (Figure 2.9).

Une caméra fictive est placée dans le repère R_c dont le plan image est confondu avec $X_c Y_c$. La distance entre le centre optique C de la caméra et le point P représente la *focale* f.

Considérons un sommet I du modèle, l'image de I que donnera la caméra est l'intersection entre le rayon optique CI et le plan (P). Le théorème de Thalès appliqué aux triangles COx_i et COy_i donne :

$$x_i/x_I = CP/CO \quad \text{et} \quad y_i/y_I = CP/CO$$

Ou encore :

$$x_i = x_I \cdot f/(f+r) \quad \text{et} \quad y_i = y_I \cdot f/(f+r)$$

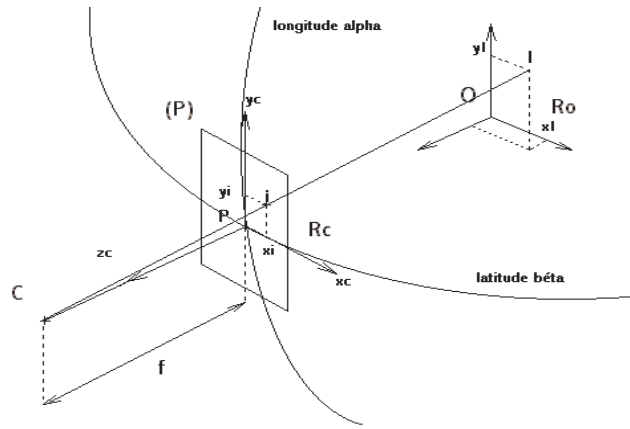


Figure 2.9 : Représentation de la caméra fictive

La projection du modèle est obtenue en projetant ainsi tous ses sommets, suivant un point de vue défini par une latitude et une longitude sur la sphère. Il reste maintenant afin d'obtenir un aspect concret du modèle à déterminer les sommets cachés lors de la projection.

- Détermination des sommets cachés

Ce problème se réduit à déterminer si un point projeté est caché par l'une des faces du modèle : le rayon visuel reliant le point au centre optique C est coupé par celle-ci.

Soit P un point du modèle, son rayon visuel CP coupant la face F_1 en un point A. Alors si $CA < CP$ le point P est caché sinon il est visible. Dans la Figure 2.10, le sommet P du modèle est caché par la face F_1 car $CP \cap F_1 = A$ et $CA < CP$. En revanche, le sommet Q n'est pas caché par la face F_2 car $CQ \cap F_2 = B$ et $CB > CQ$.

Le problème consiste maintenant à déterminer si le point A appartient réellement à la face F_1 et non pas seulement à son plan. Notons que P et A ont la même projection P' sur le plan image, car ils sont situés sur le même rayon visuel CP. Nous testons d'abord si P' appartient à la boîte englobante de F_1 (c'est le rectangle minimal contenant la projection de F_1). Cette vérification permet de réduire le nombre de faces testées. La dernière vérification consiste à tester si P' est à l'intérieur du contour projeté de F_1 .

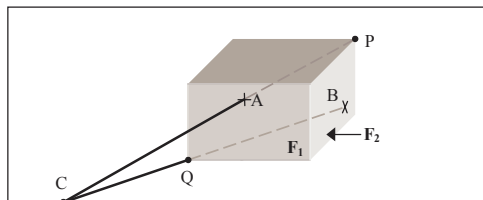
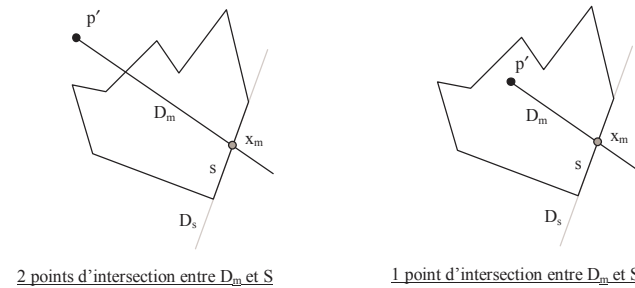


Figure 2.10 : Détermination des points cachés



Soit S l'ensemble des segments formant le contour d'une face 2D, $P'(x,y)$ est le point à tester (Figure 2.11). x_m définit le milieu du premier segment noté s de S tel que P' n'appartienne pas à D_s (la droite contenant le segment s), D_m est la demi droite issue de P' et qui passe par x_m . Notons nb , le nombre d'intersections de D_m avec les segments de S. Si nb est impaire alors le point P' est à l'intérieur de la face sinon il est à l'extérieur.

Figure 2.11 : Appartenance d'un point à une face

La Figure 2.12 nous donne un aperçu d'un modèle dans l'espace. La Figure 2.13 montre un des aspects de ce modèle obtenus en appliquant la méthode exhaustive.

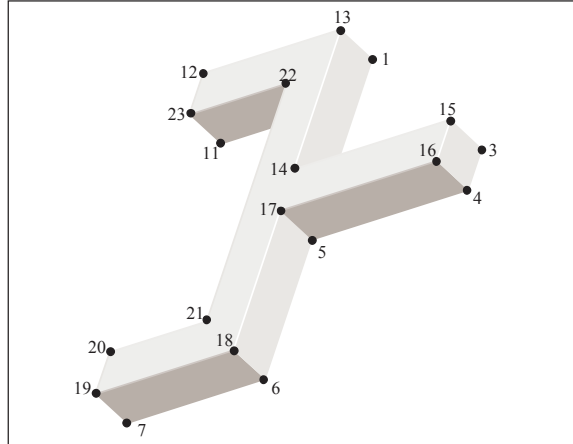


Figure 2.12 : Aperçu du modèle de l'objet Pinces dont l'indice des sommets et des faces a été noté.

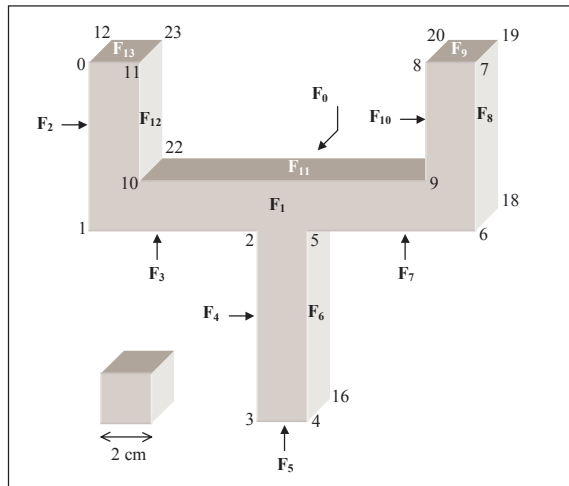


Figure 2.13 : Un aspect de l'objet Pinces
(l'aspect est représenté par les sommets visibles)

Un ensemble d'aspects est ainsi défini et rangé dans un graphe, dans lequel il suffit alors d'éliminer la redondance. Un arc relie deux aspects dans le graphe si et seulement si, leurs points de vue sont voisins sur la sphère.

- Discussion

Cette méthode a le mérite d'être rapide et de fournir un graphe correspondant uniquement à ce que l'observateur peut voir. De plus elle s'applique à n'importe quel type de polyèdre immédiatement, avec la même complexité algorithmique (le même nombre de projections qui dépend uniquement du pas de discrétisation). Cependant le graphe d'aspects est généralement incomplet et sa précision découle de celle de la discrétisation et du rayon de la sphère.

Une variante de cette méthode consiste à répéter la même procédure, pour plusieurs sphères de rayon différent centrées sur l'objet, permettant d'obtenir un graphe plus complet.

2.3.3.2 Méthode analytique

Cette méthode est développée pour servir à l'appariement basé sur un graphe (cf. 4.2). Elle fournit le graphe complet d'un objet polyédrique. Le problème est abordé en deux étapes de complexité croissante. Il s'agit tout d'abord de traiter les polyèdres convexes, puis de s'étendre au cas général, en incluant les polyèdres concaves.

- Cas convexe

Les notations concernant cette partie proviennent des travaux d'Edelsbrunner, O'Rourke et Seidel [EDELBRUNNER 1986]. Le problème mathématique consiste à déterminer la partition de l'espace (appelée *arrangement* et représentée par un *graphe d'incidence*) de dimension d , induite par n hyperplans.

Notion d'arrangement :

Soit H un ensemble d'hyperplans dans l'espace E^d , l'arrangement $A(H)$ est la partition de l'espace que forme H . $A(H)$ est formé de sous-espaces de dimensions diverses. Un sous-espace de dimension k est appelé *k-face* (k valant de 0 à d : 0, 1, 2 ou 3 dans l'espace 3D). Une 3-face représente une *cellule*, 2-face un plan ou une partie de plan (*patch*), 1-face une droite, semi droite ou arête, 0-face un point (ou *vertex*), (-1)-face l'ensemble vide.

Chaque hyperplan h divise l'espace E^d en trois régions : h^+ , h , h^- . La position d'un point p (0-face) est :

- soit au-dessus de h : $p \in h^+$,
- soit au-dessous de h : $p \in h^-$,

- soit dans $h : p \in h$.

En généralisant, chaque k -face possède un attribut la caractérisant, un *mot* ou *k-word* de n lettres décrivant sa position par rapport aux hyperplans de H (n est le nombre de ceux-ci). La $i^{\text{ème}}$ lettre de ce mot situe la k -face par rapport au $i^{\text{ème}}$ hyperplan et peut valoir :

- + si elle est au-dessus du plan,
- - si elle est au-dessous,
- 0 si elle est sur le hyperplan).

Dans le plan euclidien (E^2), les hyperplans sont des droites. La Figure 2.14 montre un arrangement créé par les deux hyperplans $h1$ et $h2$, leur visibilité est du côté des flèches. Cet arrangement contient des 2-faces (patches $p1$ à $p4$), des 1-faces (demi-droites $e1$ à $e4$) et un 0-face (vertex v). Chaque k -face est positionnée par son mot, "+-" signifie par exemple que $p1$ est du côté visible (au-dessus) de $h1$ et invisible de $h2$ (au-dessous).

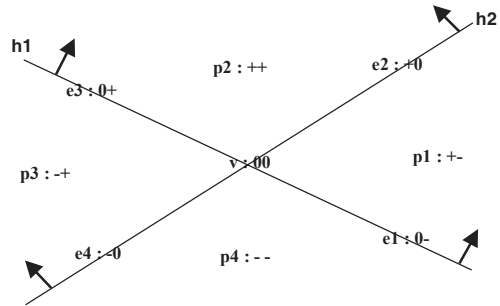


Figure 2.14 : Exemple d'arrangement $A(h1,h2)$ de deux droites en 2D

Graphe d'Incidence :

Soit $f1$ une k -face, $f2$ une $(k-1)$ -face, $f1$ et $f2$ sont *incidentes* si $f2$ est dans la fermeture de $f1$. $f1$ est aussi appelée *super-face* de $f2$, inversement $f2$ est une *sous-face* de $f1$. Par exemple, sur la figure 2.14, $p2$ et $e3$ sont incidentes de même que $e3$ et v , etc.

Le *graphe d'incidence* est un graphe dont les nœuds sont les k -faces de l'arrangement $A(H)$ correspondant et dont un arc relie toutes les deux k -faces incidentes. Dans le GI de la Figure 2.15, les 1-faces pointent également vers leur hyperplan support.

La détermination des arcs du graphe devient simple avec l'introduction du mot représentatif des k -faces. En effet, une i -face et une $(i+1)$ -face sont incidentes si leurs mots sont identiques à une lettre près qui sera 0 pour la i -face.

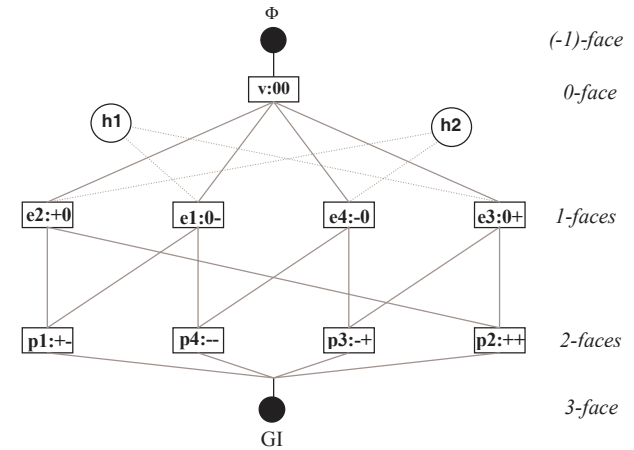


Figure 2.15 : Le graphe d'incidence de l'arrangement de la Figure 2.14

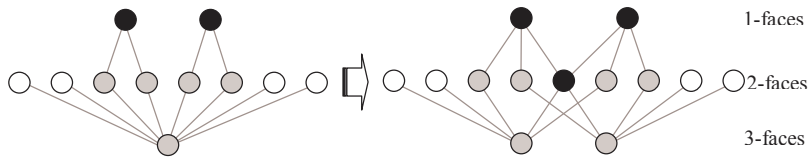
Pour des raisons pratiques, chaque k -face peut être complétée par d'autres attributs (selon l'usage voulu du graphe). Nous donnons à chaque k -face un point représentatif qui sera :

- le point lui même pour un point (0-face),
- son milieu pour un segment, ou un point à une distance fixée de son extrémité pour une demi-droite (1-face).
- le barycentre des points des sous-faces pour les patches et les cellules.

Il est utile de disposer d'un point à l'intérieur de chaque cellule, cet attribut sera utilisé comme point de vue représentatif de l'aspect concerné, nous l'appelons *p-point*.

Algorithmes :

Nous disposons à partir du modèle d'un polyèdre convexe, d'un ensemble de plans 'généralisant' l'espace (i.e. la famille de leur vecteurs normaux est une famille génératrice ou base). Cet ensemble contient donc au moins trois plans :



1. le graphe d'incidence de ces trois plans est construit. La structure générale du GI obtenu est toujours la même, seules les informations auxiliaires comme le p-point seront différentes d'un graphe à l'autre.
2. les autres plans sont insérés un par un de la même manière au graphe de départ, cette insertion constitue le cœur de l'algorithme.

Voici les différentes étapes d'insertion du $i^{\text{ème}}$ plan dans un GI déjà construit avec les $i-1$ premiers plans :

- 2.1. chaque arête (1-face) dans le graphe en intersection avec le plan est marquée.
- 2.2. toutes les super-faces (patches) d'une arête en intersection avec le plan sont marquées.
- 2.3. toutes les super-faces (cellules) d'un patch en intersection avec le plan sont marquées.
- 2.4. chaque k-faces (1,2,3-faces) marquée est coupée en deux, une nouvelle (k-1)-face (0,1,2-faces respectivement) est créée. La (k-1)-face créée est incluse dans le plan (c'est elle qui sépare la k-face coupée). Les arcs du graphe sont mis à jour à chaque fois.

Les Figures 2.16 et 2.17 montre deux exemples de mise à jour d'une arête et d'une cellule. Durant l'insertion d'un plan, toute k-face testée est marquée d'une couleur différente :

- rouge : si elle est en intersection directe avec le plan,
- blanche : si seule sa frontière est en intersection avec le plan ou si elle ne touche pas le plan,
- bleu : si elle vient d'être créée

Un système de 3 files d'attente mémorise les k-faces (arêtes, patches et cellules) intéressantes pour la dernière étape (2.4). Ceci permet de raccourcir la recherche dans le graphe.

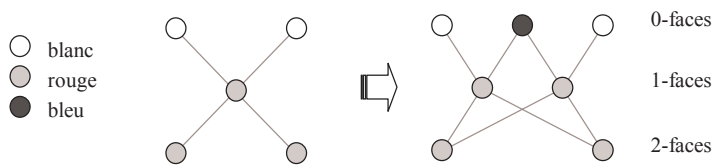


Figure 2.16 : Mise à jour du graphe pour une arête en intersection avec le plan inséré

Figure 2.17 : Mise à jour du graphe pour une cellule en intersection avec le plan inséré

Complexité :

Etudions d'abord la complexité de l'algorithme présenté ci-dessus dans l'espace 2D. Soit n droites situées dans un plan et $f(n)$ le nombre de régions créées par n . Supposant le pire des cas, la $n^{\text{ème}}$ droite se coupe avec les $n-1$ dernières en $n-1$ points et donc se divise en n segments. Ces n segments créent n nouvelles régions donc : $f(n) = f(n-1) + n$.



En développant, il vient : $f(n) = n^2/2 + n/2 + 1$ qui vaut la borne supérieure du nombre d'aspects dans l'espace E^2 .

Dans l'espace E^3 et pour n plans, nous cherchons le nombre de partitions $F(n)$ en suivant la même analyse. Le $n^{\text{ème}}$ plan se coupe avec les $n-1$ plans en $n-1$ droites créant sur lui $f(n-1)$ régions, et par conséquent $f(n-1)$ nouveaux volumes, donc : $F(n) = F(n-1) + f(n-1)$.

En remplaçant $f(n-1)$ dans la dernière équation et en développant nous obtenons :

$$F(n) = n^3/6 + 5n/6 + 1$$

Cette équation représente la borne supérieure du nombre de divisions de l'espace en cellules par tous les plans. En considérant que l'opération de division d'une cellule est d'une complexité constante, l'algorithme présenté possède une complexité de $O(n^3)$.

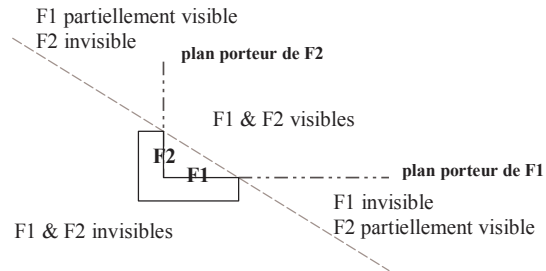
En général, l'algorithme décrit est optimal et de complexité $O(n^d)$, n étant le nombre d'hyperplans et d la dimension de l'espace.

Passage au graphe d'aspects :

Chaque cellule du GI représente une parcelle de l'espace, contenant tous les points de vue à partir desquels on voit toujours le même aspect du polyèdre. Le mot d'une cellule définit sa visibilité des faces du polyèdre et donc l'aspect recherché. Une face est visible à partir d'une cellule, lorsque la lettre concernant le plan support de la face vaut '+' (la cellule est au-dessus par rapport à la normale de la face). Les feuilles du GI (3-faces) représentent les différents aspects et donnent directement le graphe d'aspects pour un polyèdre convexe. Dans la Figure 2.15 les feuilles du GI sont des 2-faces, leurs mots correspondent à la visibilité des deux droites $h1$ et $h2$. Si l'objet se situe dans la zone $p4$ et si ses arêtes sont $e1$ et $e4$. Les mots des 2-faces fournissent le GA correspondant (Figure 2.18).

Figure 2.18 : Graphe d'aspects correspondant au graphe d'incidence de la Figure 2.15

Pour un polyèdre convexe la construction du GA se limite donc à la construction du GI de la liste des plans support des faces. Cette étape sera toujours essentielle dans le cas d'un polyèdre concave mais de nombreux



autres calculs seront nécessaires.

- Cas concave

Pour un aspect d'un polyèdre convexe, une face est soit visible soit invisible. Pour un aspect d'un polyèdre concave, il faut introduire la notion de *visibilité partielle*. La visibilité partielle pour les faces est liée à la notion de *qui-cache-qui* et à la notion de *qui-coupe-qui* pour les segments. Elle peut se résoudre à l'aide des événements visuels. Il est possible de fixer un nombre restreint d'événements visuels couvrant les phénomènes de visibilité partielle.

Evènements visuels :

Platinga et Dyer [PLATINGA 1990] classent les événements visuels que l'on va rencontrer d'une manière très accessible : événements point-segment ou segment-segment-segment. Sur la Figure 2.19 un simple plan (celui en diagonale) sépare une cellule simple dans un graphe d'incidence en deux :

- dans une cellule, une face est entièrement visible,
- dans l'autre, elle est partiellement cachée.

Ce plan est appelé *plan auxiliaire*, l'évènement visuel point-segment est le passage d'un côté à l'autre d'un tel plan. Sur la Figure 2.20 les points et les segments impliqués dans des plans auxiliaires sont en gras.

Figure 2.19 : Exemple d'un plan limite d'un événement visuel

Les autres événements possibles sont ceux impliquant trois segments (deux à deux non coplanaires). Cette fois la surface séparant les deux vues possibles (celle où l'on voit les trois segments se coupant en un même point) est une quadrique qui sera toujours un hyperboloïde à une nappe ou un parabololoïde hyperbolique.

Figure 2.20 : Les plans intervenant dans les événements point-segment d'un polyèdre concave simple

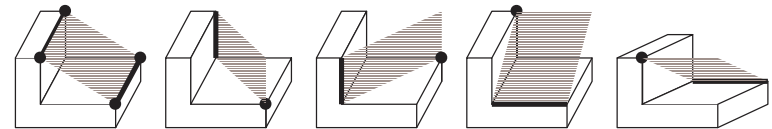
Dans notre implantation, les événements segment-segment-segment n'ont pas été pris en compte. En fait, la complexité de traitement de ce type est énorme par rapport au nombre d'aspects qu'il peut donner au niveau du graphe d'aspects pour un objet relativement simple. Dans de nombreux cas ce type d'évènement n'a pas lieu puisqu'il faut pouvoir voir trois segments se couper en un point.

La parcellisation de l'espace est complétée pour le cas concave en ajoutant celle obtenue par les plans auxiliaires. Il est intéressant de noter une particularité du graphe d'incidence pour un polyèdre concave : l'intérieur de l'objet n'est plus composé d'une seule cellule comme dans le cas convexe. En effet, certains plans coupent l'objet et il est important de localiser l'objet dans le graphe d'incidence. Cela empêche de 'subdiviser' l'intérieur de l'objet avec les plans auxiliaires.

Algorithme :

Cet algorithme est issu des travaux de Stewman et Bowyer [STEWMAN 1988] qui utilise celui de Edelsbrunner, [EDELBRUNNER 1986] pour la construction du graphe d'incidence. L'idée générale provient de leur algorithme mais une modification y a été apportée afin de limiter le nombre de plans auxiliaires pris en compte :

1. construire le graphe d'incidence (GI) avec les plans tirés de l'objet (comme dans le cas convexe)



2. repérer l'objet dans ce graphe
3. répertorier les événements visuels possibles
4. construire la liste des plans auxiliaires
5. insérer ces plans dans le GI
6. traduire le graphe d'incidence en graphe d'aspects

Complexité :

La complexité de construction du graphe d'incidence (et donc du graphe d'aspects) pour un polyèdre concave se déduit en considérant le nombre de plans (principaux n et auxiliaires n') divisant l'espace. En outre, le nombre de plans auxiliaires se déduit du nombre d'évènements visuels formés par les sommets S et les arêtes A du polyèdre. La borne supérieure du nombre des plans auxiliaire est donc : $n' = S \times A + A^3$. Etant donné que S et A sont

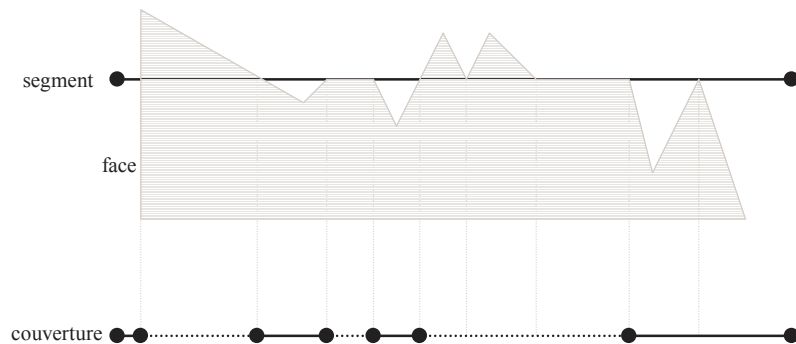
dépendants linéairement de n (de complexité $O(n)$). Nous trouvons que la complexité du cas concave pour une projection perspective vaut : $O(n^3)$.

Limiter le nombre de plans auxiliaires :

La méthode la plus simple pour construire les plans auxiliaires consiste à en construire un pour chaque couple point-segment de l'objet. Il n'y a aucune précaution à prendre mais on se retrouve instantanément avec un nombre impressionnant de plans dont la plupart sont inutiles. Le temps d'exécution s'accroît rapidement car cet algorithme fonctionne en $O(n^3)$ où n est le nombre de plans insérés.

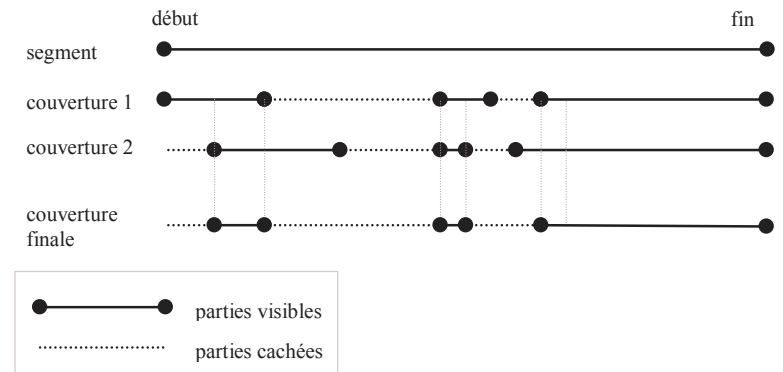
Notre développement consiste à vérifier si le segment est visible (au moins en partie) du point de vue donné (point à associer avec le segment par un événement visuel). Cette visibilité est évaluée en 'projetant' chaque face, dont le plan support passe entre le segment et le point, sur le segment et en cumulant ces projections. Les parties cachées du segment correspondent à une suite de bouts de segments alignés appelée *méta-segment*. Elle est représentée par une liste de points appelée *couverture* (du fait de la couverture du segment par une face). Pour obtenir une couverture les extrémités du segment sont projetées sur le plan support de la face, Figure 2.21, la direction de projection est toujours donnée par le point de vue. L'intersection du segment projeté avec chaque segment de la face est évalué. Lorsqu'il y a intersection, le point est calculé, projeté sur le segment et ajouté à sa couverture courante. La couverture est également utilisée lors de la construction des aspects. Elle sert à construire les méta-segments du contour d'une face partiellement cachée dans un aspect.

Figure 2.21 : Un segment, une face et la couverture correspondante



Cette opération est effectuée pour chaque face utile, la couverture obtenue est fusionnée à chaque fois avec la couverture de départ (Figure 2.22). Pour pouvoir fusionner toutes les couvertures concernant un même segment, nous prenons comme convention de toujours partir du début du segment vers son extrémité (direction prédéfinie). La couverture commence au début de la première partie visible du segment, chaque point marque ensuite une rupture entre une partie visible et une partie cachée. Chaque point est muni d'une variable indiquant si la partie le suivant est visible ou cachée, ce qui servira à fusionner les couvertures entre elles.

Figure 2.22 : Fusion de deux couvertures



La couverture de départ étant le segment complet (entièrement visible), la procédure s'arrête lorsque la couverture est nulle ou lorsqu'il n'y a plus de face à tester.

• Améliorations

Une amélioration non négligeable est d'incorporer le traitement des événements de type segment-segment puisque c'est le seul moyen d'avoir un graphe d'aspects complet pour un objet complexe.

Il est envisageable d'ajouter à ce programme un filtre limitant le graphe obtenu en sortie. Nous pouvons par exemple se limiter à des aspects vus 'd'assez loin' de l'objet ou définir les aspects uniquement à partir de la visibilité des faces. Il suffira en général de rajouter un test lors de la construction des aspects.

Nous travaillons sur un «visualiseur» d'aspects, ce dernier est très utile pour la consultation et la correction d'un graphe. Une telle extension nécessite un maximum d'informations pour l'affichage qu'il serait fastidieux de recalculer (les méta-segments par exemple). La visualisation d'un aspect pourra être utilisée en ligne pour la superposition du modèle 3D sur son image.

2.4 Conclusion

Le modèle géométrique de la caméra utilisé est celui du sténopé avec et sans distorsions. Plusieurs méthodes d'optimisations mathématiques sont utilisées pour la calibration de la caméra. Pour le modèle linéaire, la méthode des moindres carrés est appliquée. Pour le modèle non-linéaire, la méthode d'optimisation mathématique de Levenberg-Marquardt est appliquée en utilisant un estimé initial fourni par la méthode des moindres carrés. Ces travaux ont été effectués dans des thèses antérieures. Nous les avons utilisés pour le développement de la calibration automatique présenté au chapitre 5.

La modélisation géométrique de la scène passe par une représentation des objets 3D puis par l'acquisition de ceux-ci à l'aide des capteurs. Les objets modélisés étant des polyèdres, la représentation BREP a été retenue pour MCIT et l'acquisition de ce type d'objets se fait par une coopération caméra et télémètre.

Nous avons développé deux méthodes d'extraction de graphe d'aspects à partir d'un polyèdre. La première est exhaustive, simple à appliquer mais ne donne pas le graphe d'aspects complet. La seconde est analytique et plus difficile à mettre en œuvre, car elle se base sur la parcellisation de l'espace. Elle fournit en revanche, un graphe d'aspects exact du polyèdre. Les deux méthodes sont utilisées pour servir deux méthodes d'appariement.

3. Traitement et modélisation de l'image 2D

3.1. Introduction

Ce chapitre est consacré au traitement de l'image, à l'issue duquel un modèle 2D de la scène est établi. La scène est capturée à l'aide d'une caméra fixe, sous forme d'une image d'intensité (en 256 niveaux de gris), l'objectif étant d'extraire les contours des objets vus dans l'image. Le traitement d'images appliqué consiste à extraire des primitives 2D (sommets, segments, chaînes de segments) formant le modèle 2D. Ce traitement est constitué du filtrage du gradient, de l'amincissement et du suivi des contours par hystérésis, de l'extraction des segments et des chaînes de segments.

L'extraction de segments est réalisée à partir des points de contours en appliquant la Transformée de Hough (TH). Nous exposons, par la suite, deux nouvelles méthodes permettant d'améliorer l'extraction des segments obtenus à partir de la TH. L'organisation perceptive est appliquée afin d'obtenir les relations de proximité, de colinéarité et de parallélisme entre les segments. L'extraction de sommets et de chaînes de segments est basée sur ces relations. Nous apportons également une amélioration sur l'extraction du modèle 2D en adoptant des règles de correction.

Avant d'exposer chaque partie de notre logiciel de traitement d'images, la définition du vocabulaire utilisé est nécessaire.

3.2. Vocabulaire

- Voisinage

Un pixel f dans une image bidimensionnelle est représenté par sa position (x,y) et par son intensité $f(x,y)$ (ou niveau de gris). L'ensemble des voisins ayant les positions : $(x+1,y)$, $(x-1,y)$, $(x,y+1)$, $(x,y-1)$ est appelé *4-voisins* de f et noté $N_4(f)$, il est présenté en noir dans la Figure 3.1. Les pixels en gris forment l'ensemble des *voisins diagonaux* de f noté $N_d(f)$. L'union des 4-voisins et des voisins diagonaux est appelé *8-voisins* de f et noté $N_8(f)$, la numérotation de cet ensemble illustrée dans la Figure 3.1 est appelée *codage de Freeman*.

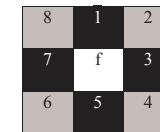


Figure 3.1 : Les voisins d'un pixel

Trois types de connectivité entre deux pixels f_1 et f_2 sont considérés :

- 4-connectivité si $f_2 \in N_4(f_1)$,
- 8-connectivité si $f_2 \in N_8(f_1)$,
- connectivité mixte si : $f_2 \in N_4(f_1)$ ou $(f_2 \in N_d(f_1) \text{ et } N_d(f_1) \cap N_4(f_2) \text{ est vide})$

- Attributs et primitives

On associe à tout pixel de position (x,y) un *attribut* $a(x,y)$ qui le caractérise, il peut être son intensité, sa couleur, son gradient, son angle, sa texture, etc. Deux pixels sont *liés* s'ils sont connexes et s'il existe une relation entre leurs attributs. Soit A un choix de valeurs d'attributs, une *primitive* est un ensemble de pixels (région) défini par : $R(A) = \{ (i,j) : a(i,j) = A \text{ et } \exists (k,l) \in R(A) \text{ tel que } (i,j) \text{ et } (k,l) \text{ sont liés} \}$.

Les primitives se divisent en deux types : surface (appelé par abus de langage région) et contour. Dans le premier, les pixels peuvent avoir plusieurs liaisons, dans le second chaque pixel est limité au maximum à deux liaisons.

- Segmentation

La *segmentation* est l'opération qui consiste à subdiviser une image en régions significatives ; primitives de surface et/ou de contour. La segmentation est la première procédure à réaliser pour effectuer une reconnaissance de formes 2D ou d'objets 3D, dont la qualité de celle-ci dépend. Cette étape permet donc d'engendrer une description compacte de l'image plus exploitable que l'ensemble des pixels. Le traitement d'images de polyèdres vise la recherche de contours, deux approches existe pour la détection de ces primitives : par régions (indirecte) ou par contours (directe).

La première représente la recherche des zones homogènes dans l'image dans le sens de leurs niveaux de gris. Dans la seconde, ce sont les variations des niveaux de gris séparant les régions (frontières) qui sont recherchées. Ces deux approches sont duales, car le résultat de l'une complète celui de l'autre.

L'application de l'approche région sur une image d'un polyèdre a l'avantage de donner les régions fermées mais la recherche des frontières exactes de ces régions (le même nombre de segments dans la réalité) est un problème difficile. En effet, les segments de droite sont une information importante sur laquelle nous nous basons pour effectuer la reconnaissance (cf. chapitre 4). Nous avons préféré l'approche directe car plus rapide, bien qu'il existe des méthodes basées sur la coopération des deux approches pour extraire au mieux les contours.

3.3. Extraction de Contours

3.3.1. Introduction

L'objectif est de développer un traitement d'images minimal d'objets polyédriques, pour fournir les éléments nécessaires à l'appariement. L'extraction de contours, se fait alors, par une application successive des filtres classiques :

- le filtre de Sobel dans deux axes, pour obtenir l'amplitude et l'angle du gradient,
- le filtre des maxima locaux, pour amincir les contours,
- le seuillage par hystérésis, pour obtenir un suivi simple des contours avec un bruit minimal.

La méthode de traitement utilisée se compose de plusieurs opérations décrites ci-dessous. Auparavant, quelques notions sur les signaux de contour et le filtrage sont brièvement rappelées.

3.3.2. Notions

- Modèles de contours

Le contour est une frontière de variation d'intensité (niveaux de gris). Plusieurs types de contours existent dans une image (Figure 3.2). Des bruits divers sont présents : bruit convolutif, bruit additif, bruit blanc, etc. La plupart des méthodes de détection de contours se focalisent sur les contours de type marche (discontinuité d'ordre 0) avec un bruit blanc additif [HORAUD 1993]. Dans le cas des polyèdres, un contour reflète le fait d'avoir deux surfaces : soit orientées différemment par rapport à une source de lumière et au point de vue soit de différentes couleurs.

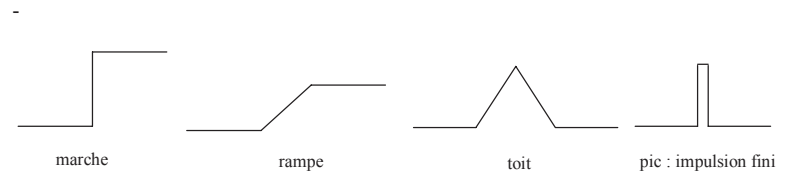


Figure 3.2 : Profils de contour

- Filtrage

Le *filtrage* d'une image $I(x,y)$ est la convolution de celle-ci avec une fonction $f(x,y)$ qui est la réponse impulsionnelle du *filtre*. Dans le cas discret I et f sont bornées, la convolution s'écrit :

$$I_f(i, j) = \sum_{i'=-k}^{+k} \sum_{j'=-k}^{+k} f(i', j') I(i+i', j+j') \quad (3.1)$$

le filtre est représenté donc par une matrice (généralement carrée) de dimension $2k+1$, celle-ci est appelée *masque* ou *opérateur* ou même « filtre ». Le filtrage linéaire consiste simplement à remplacer chaque niveau de gris par une combinaison linéaire des niveaux de gris des pixels voisins. Les coefficients de cette combinaison (le masque) sont définis par la réponse du filtre à une entrée constituée par la fonction impulsion.

- Le résultat d'une détection de contour est évalué à l'aide de certains critères de performances du filtre appliqué, ils sont essentiellement :
 - détection : le filtre doit fournir une réponse à un contour donné
 - localisation : c'est la précision de détection
 - réponse unique : un contour doit provoquer une seule réponse du filtre

- Approches de détection

Comme il est mentionné ci-dessus, un contour est une variation d'intensité. Il peut être détectée classiquement par deux manières (Figure 3.3) :

1. Calcul du gradient et extraction des maxima locaux de la norme du gradient
2. Calcul du laplacien et détermination des passages par zéro

Le *gradient* d'une image bidimensionnelle $I(x,y)$ est donné par :

$$G_x(x,y) = \partial I(x,y)/\partial x \quad G_y(x,y) = \partial I(x,y)/\partial y \quad (3.2)$$

En chaque pixel (x,y) de l'image, le module (amplitude) et l'angle du vecteur du gradient se calculent comme suit :

$$G = \sqrt{G_x^2 + G_y^2} \approx \max(G_x, G_y) \approx |G_x| + |G_y| \quad \text{et} \quad \alpha = \arctan \frac{G_y}{G_x} \in [0, \pi[\quad (3.3)$$

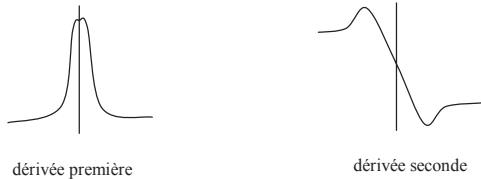


Figure 3.3 : Approches gradient et laplacien pour la détection de contours

Le gradient donne des contours larges, dont il ne faut garder que les maxima dans la direction du gradient.

Le *laplacien* est calculé par :

$$L(x,y) = \partial^2 I(x,y)/\partial x^2 + \partial^2 I(x,y)/\partial y^2 \quad (3.4)$$

qui est un scalaire, donc ne fournit pas l'angle de contour. La détection des maxima locaux du gradient revient à détecter le passage par zéro du laplacien. Les deux approches donnent alors des résultats très proches. Nous utilisons le gradient pour détecter les contours, car il fournit de plus l'angle de contour (celui-ci étant perpendiculaire à l'angle du gradient).

3.3.3. Calcul du gradient

Une manière commode de trouver le gradient est de chercher un filtre linéaire permettant l'approximation de celui-ci. Nous traitons le problème sous l'hypothèse d'un contour de type marche avec un bruit de moyenne nulle. Soit f la réponse impulsionnelle d'un filtre linéaire et I une image, la dérivée de l'image convoluée avec f est égal à l'image convoluée avec la dérivée de f : $(I \otimes f)' = I \otimes f'$ où \otimes le symbole de convolution

Un des filtres qui effectue une approximation du gradient est le filtre de Sobel, il s'agit de calculer le gradient du premier ordre en chaque pixel en fonction de ses voisins dans les deux directions principales. Ce filtre est représenté par le deux masques suivants :

$$\begin{bmatrix} -1 & 0 & +1 \\ -2 & * & +2 \\ -1 & 0 & +1 \end{bmatrix} \quad \begin{bmatrix} +1 & +2 & +1 \\ 0 & * & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Figure 3.4 : Masques horizontal et vertical de Sobel

L'étoile symbolise le pixel (i,j) à modifier. Chacun des deux masques s'applique en balayant l'image de départ pixel par pixel. Les deux gradients obtenus ainsi sont utilisés pour fournir le gradient total G_{ij} (amplitude) et son angle α_{ij} dans chaque pixel de l'image en utilisant la relation (3.3) ci-dessus. Des filtres directionnels existent également pour estimer le gradient dans une direction connue à priori, les masques de Kirsch en sont un exemple [HORAUD 1993].

3.3.4. Amincissement de contours

Le gradient obtenu par le filtre de Sobel conduit à des contours larges, qu'il faut amincir. Pour ce faire, il ne faut garder que les points au gradient maximal dans la direction de celui-ci.

La table ci-dessous renferme les amplitudes du gradient dans une portion d'image :

0	5	15	3	2	0
1	2	10	20	8	3
3	7	10	16	6	1

Tableau 3.1 : Les amplitudes du gradient sur une portion d'image

L'algorithme consiste à balayer l'image ligne par ligne, pour chaque élément (i,j) on applique les opérations suivantes :

- L'amplitude G_{ij} et l'angle α_{ij} du gradient dans (i,j) sont fournis par l'étape précédente,
- Les deux éléments voisins dans le sens de α_{ij} sont trouvés (une approximation se fait pour donner deux éléments opposés parmi les 8-voisins selon le codage de Freeman) :

$$(i-m,j-n) \text{ et } (i+m,j+n) \text{ où : } m,n \in \{0,1\} \text{ et } m+n > 0$$

- L'opérateur logique suivant est appliqué :

$$\text{si } G_{i-m,j-n} > G_{ij} \leq G_{i+m,j+n} \text{ poser } G_{ij} = 0$$

ceci revient à ne garder que les éléments continus ayant un gradient localement maximal sur les contours. Le fait que l'inégalité précédente soit stricte d'un seul côté signifie l'élimination d'un seul point (toujours du même côté) en cas d'égalité des deux gradients.

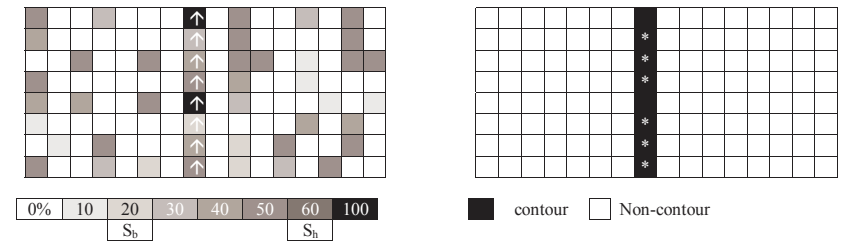
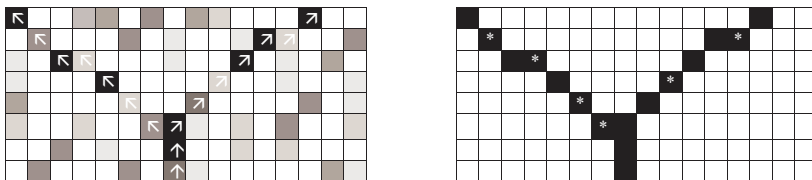
Sous l'hypothèse que le gradient sur l'image du Tableau 3.1 est horizontal, grâce à la connaissance de son angle α_{ij} , l'application de l'algorithme ci-dessus donne :

0	0	15	0	0	0
0	0	0	20	0	0
0	0	0	16	0	0

Tableau 3.2 : Les amplitudes du gradient sur le contour aminci

3.3.5. Suivi de contours

La méthode d'Hystérésis est choisie pour déterminer les points faisant partie d'un contour. Deux seuils sont fixés, un seuil haut et un seuil bas, correspondant à 2 valeurs de gradients. Les pixels de l'image dont le gradient est supérieur ou égal au seuil haut sont marqués. Le contour est suivi à partir de tous ces points, en retenant tous les points voisins dont le gradient est au moins égal au seuil bas. Une image binaire des contours détectés est obtenue. La première expérience amène à prendre conscience des limites du traitement d'images. Les seuils d'Hystérésis sont difficiles à choisir : on ne peut pas les fixer, car leurs valeurs optimales dépendent de la qualité de l'image (luminosité, contrastes, bruits). Un seuil trop haut ne donne aucun contour et un seuil trop bas génère des contours à l'intérieur d'une face, rendant le résultat inexploitable.



(a) : Image 16x16 de gradients simulés en 8 niveaux de gris, les flèches indiquent la direction du contour sur les pixels parcourus
 (b) : Résultat de l'hystérésis en appliquant un seuil bas à 20% et un seuil haut à 60% du gradient maximal. * indique les pixels qu'on aurait perdus en appliquant un seuillage simple à S_h

Figure 3.5 : Application de l'hystérésis sur une image simulée

les flèches indiquent la direction du contour

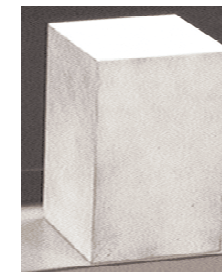
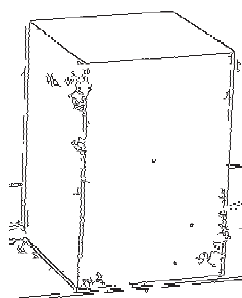


Figure 3.6 : Image d'intensité de l'objet cube

Des essais sur l'image en niveaux de gris d'un simple parallépipède rectangle illustrent cette procédure. Le traitement d'image est appliqué deux fois, avec des seuils différents.

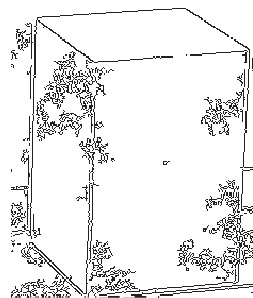
Dans le premier cas, une série de segments corrects est obtenue. Les segments de l'objet pourront être déduits après traitement.

En revanche, dans le deuxième cas, le traitement s'est révélé trop sensible et les segments obtenus sont beaucoup trop nombreux. Le résultat est inexploitable.



1^{er} essai :

Les seuils d'hystérésis ont été correctement choisis.



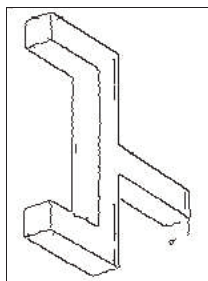
2^{ème} essai :

Les seuils d'hystérésis trop bas ou trop élevés conduisent à une image inexploitable. Soit il y a trop de segments (Figure ci-dessus), soit il n'y a pas assez pour reconstituer les segments du volume.

Figure 3.7 : Test de l'Hystérésis

En fait, il est difficile de choisir les mêmes seuils d'Hystérésis pour l'intégralité de l'image. Une solution serait de découper l'image en plusieurs zones et pour chacune d'elles choisir des seuils adaptés. Il faudrait trouver dans ce cas la stratégie à adopter pour diviser l'image en zones. Plusieurs manières d'opérer peuvent être envisagées. La première, arbitraire, consiste à utiliser des rectangles d'égales dimensions. Une autre serait de détecter les zones de l'image dont les caractéristiques sont voisines (par exemple, en termes de luminosité et de contraste). On pourrait également subdiviser l'image récursivement jusqu'à obtention de zones ayant un histogramme à 2 pics [ROSENFELD 1982], [HORAUD 1993].

La Figure 3.8 montre les points de contours obtenus (image binaire), à partir de l'image d'entrée (a).



(a) image de l'objet Pince (b) le résultat de l'extraction des contours

Figure 3.8 : Traitement de l'image de l'objet pince

3.4. Segmentation de contours

3.4.1. Introduction

Un problème fréquemment rencontré lors de l'extraction de primitives d'objets dans une image de contours est celui de la détection de courbes analytiques (comme les cercles, ellipses, etc.) et plus spécifiquement les droites appartenant aux contours rectilignes (parvenant de polyèdres). Chaque forme analytique constitue une famille C de l'espace ayant l'expression générale : $f(\vec{p}_{x,y}, \vec{v}) = 0$, où $\vec{p}_{x,y}$ est un vecteur représentant un point $p(x,y)$ de l'espace P et \vec{v} est un vecteur de paramètres caractéristiques de C. Etant donné un ensemble de points de contours P, le problème revient donc à déterminer les sous-ensembles de P faisant partie de C dont le vecteur - paramètre restent à définir. La segmentation d'un contour posera donc deux sous-problèmes [HORAUD 1993]:

1. trouver une parcellisation du contour en segments
 2. trouver pour chaque segment la meilleure approximation analytique
- Nous nous limitons à la détection des segments de droite que nous appelons simplement *segments*, le vecteur de paramètres est à deux dimensions. Après un aperçu des solutions existantes, nous présentons la méthode implantée pour résoudre le premier problème, les améliorations apportées à celle-ci et une méthode simple pour résoudre le second problème.

Une méthode qui explorerait tous les sous-ensembles possibles de n points de P aurait une complexité exponentielle, car le nombre de combinaisons serait : $\sum_{i=2}^n C_n^i$. En raison de cette complexité, les algorithmes existants ne garantissent pas en général une solution optimale. Deux approches existent : l'approximation polygonale et la transformée de Hough.

Les méthodes d'*approximation polygonale* reposent toutes sur une représentation des contours par une liste de chaînes de points, appelées simplement *chaînes*, une étape de chaînage est nécessaire pour appliquer cette approche [GARNESSON 1991], [HORAUD 1993]. Elles procèdent par parcellisation de chaque chaîne en plusieurs parties représentées par des segments. Ces méthodes peuvent être séparées en quatre classes :

1. Les méthodes basées sur la courbure de la chaîne ; la courbure mesure la variation de la tangente le long de la chaîne ce qui permet de classer chaque portion de celle-ci :

- segment de droite correspondant à une courbure nulle,
- arc de cercle correspondant à une courbure constante,
- angle correspondant à un maximum local de la courbure,
- point de flexion correspondant à un changement de signe de la courbure

Cette méthode est très sensible au bruit, un problème qui peut être résolu partiellement par une étape de lissage. Le lissage introduit à son tour un déplacement du signal d'entrée et doit être suivi par une étape de correction.

2. Les méthodes de fusion : leur principe consiste à parcourir la chaîne du premier point jusqu'au dernier. Les points rencontrés sont fusionnés au segment courant tant que l'ensemble de ces points vérifie un critère d'erreur. Ce processus est recommencé à partir du dernier point fusionné. Ces méthodes présentent un défaut considérable : les points de coupure sont généralement situés au delà des points de coupures souhaités, car le critère d'arrêt est un cumul d'erreurs. La localisation des angles dépend fortement du point de départ. Malgré l'efficacité de ces méthodes en temps de calcul, elles nécessitent l'intégration d'autres algorithmes pour corriger leurs défauts ce qui alourdit les calculs.

3. Les méthodes de découpage : Au départ la chaîne est représentée grossièrement par un seul segment. Un point caractéristique est recherché, tel que point à une distance du segment supérieure à un seuil. S'il existe, la chaîne est découpée en ce point et le processus recommence pour les deux portions, sinon le découpage prend fin. Ces méthodes sont sensibles au bruit, un petit changement des données peut provoquer des résultats complètement différents. Elles posent également un problème de localisation de points de coupures. Elles sont en revanche facile à mettre en œuvre et permettent plus de souplesse concernant le contrôle de leurs paramètres.

4. Les méthodes hybrides ; elles mélangent des algorithmes des trois premières classes. Un exemple de cette catégorie est la méthode de découpage – fusion, elle procède à un découpage de la chaîne puis à une évaluation des paires de segments consécutifs pour éventuellement les fusionner. Bien que cette approche soit plus lente, elle permet de résoudre certains problèmes tels que la forte sensibilité au bruit et l'erreur de localisation.

En résumé, l'approximation polygonale nécessite un chaînage de contours. Elle possède des défauts difficiles à contourner entièrement et font appel à plusieurs algorithmes. Malgré l'espace raisonnable qu'elle occupe en mémoire, sa complexité est fortement dépendante du nombre de points traités (n) et peut dépasser $O(n^2)$.

La méthode de la *Transformée de Hough* (TH) traite le problème de détection sous un différent angle, elle est basée sur l'exploration de l'espace des paramètres. La TH est considérée robuste vis à vis du bruit grâce à sa nature de recherche globale [ZHANG 1997]. En effet, elle ne procède pas par traitement local de chaque chaîne (pas besoin de chaînage), mais transforme la totalité de la zone d'intérêt d'une image dans l'espace des

paramètres recherchés. En reprenant les notations mentionnées ci-dessus, le principe est de calculer en chaque point $p(x,y)$ de contour, l'ensemble de paramètres \vec{v} qui vérifient $f(x, y, \vec{v}) = 0$ et de les accumuler dans leur espace ; ceci est équivalent à évaluer toutes les courbes passant par les points de contour.

Par ailleurs, la TH possède quelques limites :

- ses résultats sont sensibles à la quantification de l'espace des paramètres (tels que la difficulté de discerner deux segments parallèles et très voisins). Plus fine est la quantification, meilleure est la résolution ; mais le temps de calcul augmente
- sa complexité de calcul et son espace occupé en mémoire sont fortement dépendants de la dimension de l'espace des paramètres. Elle n'est pas conseillée pour un espace dépassant 2 paramètres.

La TH nous permet d'une détection robuste par sa globalité. De plus, elle ne nécessite pas de chaînage au préalable. Nous avons choisi la TH à deux paramètres (ainsi l'espace-mémoire et la complexité sont réduits au maximum) car nous ne l'utilisons que pour détecter des segments de droite. Les améliorations que nous avons apportées à la TH permettent, comme nous le verrons, de contourner le problème de quantification et de réduire la complexité.

3.4.2. La Transformée de Hough (TH)

Afin d'explorer l'espace de paramètres il faut transformer l'image de contours dans celui-ci. La représentation de cet espace se fait à l'aide d'une matrice appelée la *matrice de Hough* (MH). Dans le cas de détection des segments, c'est une matrice à deux dimensions représentant les deux paramètres d'une droite. Il est clair que cette représentation passe par une quantification de chaque paramètre (dimension) dans un intervalle de variation prévu pour celui-ci. La représentation cartésienne d'une droite prend la forme : $y = ax + b$ (Figure 3.9) où :

- x et y sont les coordonnées des points situés sur la droite
- a est le rapport de variation $\partial y / \partial x = \tan(\alpha)$ où : $\alpha \in [0, \pi[$
- b est la distance entre l'origine et l'intersection entre la droite et l'axe des ordonnées

Le paramètre b est difficile à quantifier car il varie dans $]-\infty, +\infty[$. Ceci mène à une représentation polaire :

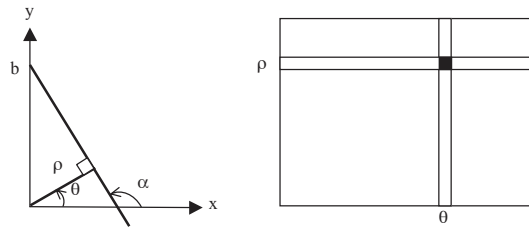
$$\rho = b \cdot \sin \theta = x \cdot \sin \theta + y \cdot \cos \theta \quad \text{où :}$$

- $\theta \in [-\pi/2, +\pi/2[$ est l'angle entre la droite et la verticale
- $\rho \in [-\rho_{\max}/2, +\rho_{\max}/2[$ est la distance entre la droite et l'origine, avec ρ_{\max} est la diagonale de l'image d'entrée

Chaque droite est repérée par sa distance ρ au centre de l'image et par son angle θ par rapport à la verticale.

L'espace de ces droites discrétisé se représente par la matrice MH dont la ligne correspond à la distance et la colonne à l'angle des droites. Chaque case de la matrice de Hough, contient un certain nombre de points,

détectés comme appartenant à la droite correspondante (Figure 3.9). Des accumulations de points vont se



produire dans certaines cases, qui correspondront à des segments contenus dans l'image.

Figure 3.9 : Angle θ et distance ρ d'une droite de la matrice de Hough

Nous prenons comme origine le centre de l'image (au lieu d'un coin), car l'erreur commise sur le calcul de ρ se réduit remarquablement. Nous analysons deux positionnements de l'origine, l'un au coin inférieur gauche, l'autre au centre de l'image. En effet, lorsque l'estimation de l'angle d'une droite subit une erreur de $\Delta\theta$ (la raison est expliquée plus tard), l'erreur $\Delta\rho$ se déduit :

- lorsque l'origine est le centre de l'image : $\Delta\rho_1 = (x \cdot \cos\theta - y \cdot \sin\theta) \cdot \Delta\theta$
- lorsque l'origine est le coin inférieur gauche :

$$\Delta\rho_2 = ((x + x_{\max}/2) \cdot \cos\theta - (y + y_{\max}/2) \cdot \sin\theta) \cdot \Delta\theta$$

avec : x_{\max} , y_{\max} respectivement la largeur et la hauteur de l'image et x, y exprimés toujours par rapport au centre de l'image. Le rapport entre ces deux erreurs est :

$$\frac{\Delta\rho_1}{\Delta\rho_2} = \frac{x - y \cdot \tan\theta}{(x + x_{\max}/2) - (y + y_{\max}/2) \tan\theta}$$

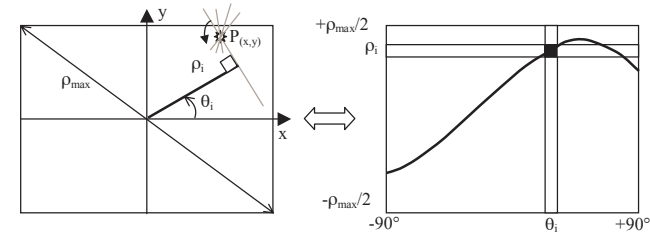
Ce qui donne, par exemple, 50% sur les points proche du coin supérieur droit d'une image carrée et ayant une droite passant par l'origine. De plus $\Delta\rho_1$ est une fonction plus uniforme par rapport à la zone d'intérêt (autour du centre de l'image). Hansen et Andersen [HANSEN 1997] arrivent à cette même conclusion par une autre analyse.

L'application de la TH classique, sans connaissance de l'angle de contour, est lente car il faut représenter chaque point du contour par une sinusoïde complète dans la matrice de Hough (MH). Chaque point $p(x, y)$ du contour peut correspondre à la cellule (θ_i, ρ_i) dans la MH (θ_i : angle de contour par rapport à la verticale, ρ_i : distance du centre de l'image à la droite support du point p), tel que :

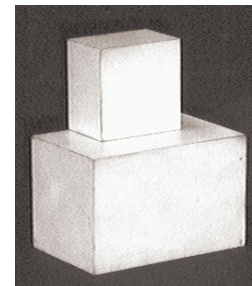
$$\rho_i = x \cdot \sin\theta_i + y \cdot \cos\theta_i \quad (3.5)$$

Il s'agit d'associer chaque point de l'image aux droites passant par lui, en fonction de sa position dans l'image (Figure 3.10).

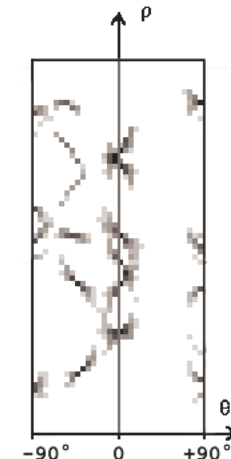
Figure 3.10 : Un point image est représenté dans une sinusoïde de cellules dans la matrice de Hough



La matrice dans la Figure 3.11-b est le résultat de l'application de la Transformée de Hough à l'objet L dans la Figure 3.11-a. Le nombre de points dans une case est représenté par le niveau de gris (noir = nombre maximum, blanc = aucun point). L'angle θ varie de -90° à $+90^\circ$, l'intervalle de la distance ρ dépend de la taille de l'image.



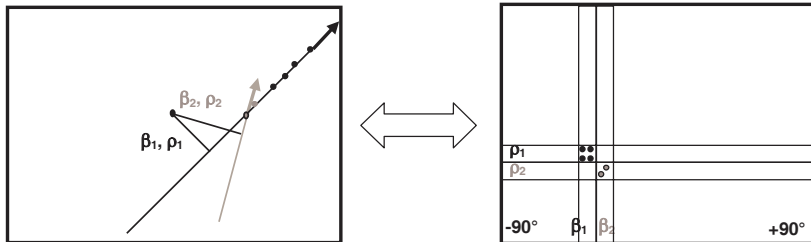
(a) Objet L



(b) Exemple de matrice de Hough

Figure 3.11 : Illustration de la Transformée de Hough

Nous utilisons la connaissance de l'angle du gradient dans chaque point de contour, cet angle n'est autre que l'angle θ de la droite support du contour en ce point. Appliqué à l'équation 3.5, il donne une seule correspondance dans la MH. La segmentation est devenue nettement plus rapide, mais l'erreur sur l'angle de gradient $\partial\theta$ a entraîné des segments incomplets. Pour pallier ceci, nous avons développé deux méthodes :



La première, appelée *TH avec fusion*, consiste à explorer les cellules voisines du segment à extraire. La seconde méthode, appelée *TH avec érosion*, consiste à définir une tolérance sur l'angle du gradient et à explorer uniquement la cellule correspondante à celui-ci. Chaque point du contour est alors représenté par une portion de sinuséide dans la MH.

3.4.3. Amélioration de la Transformée de Hough

Considérons une matrice de Hough de dimensions ($m \times n$). Un point d'une image 2D y apparaît n fois (pour chaque droite passant par lui). Afin d'améliorer le temps de réponse, l'angle de la droite à laquelle le point va appartenir peut être estimé en utilisant l'angle donné par les gradients en X et Y. Or, cet angle n'est qu'une approximation de l'angle réel du contour (les bruits dans l'image induisent des erreurs pouvant aller jusqu'à quelques dizaines de degrés). Cela conduit à des pertes d'informations (des segments ne sont pas complets, voire pas détectés du tout). Les points du contour, appartenant à une droite, sont dispersés dans différentes cellules voisines (Figure 3.12).

Figure 3.12 : L'effet de dispersion des points appartenant à la même droite dû à l'erreur sur l'angle

Pour contourner ce problème, Nous avons implanté les deux méthodes mentionnées ci-dessus.

3.4.3.1. la méthode TH avec fusion

Elle consiste à chercher, au voisinage d'une cellule, l'existence de points appartenant au segment en cours. Ils sont ensuite fusionnés avec la cellule considérée dans la MH. Deux règles sont adoptées pour cette fusion :

- la différence entre l'angle du contour en ce point et l'angle de la droite support du segment est inférieur à un seuil S_a
- la distance entre le point et le milieu du segment est inférieur à un seuil S_d

La réponse de cette méthode est nettement meilleure que la précédente (la TH utilisant l'angle du gradient mais sans fusion de cellules). Plus que 80% des segments sont retrouvés. En revanche, il est difficile de fixer les deux seuils optimaux S_a et S_d car ils changent selon l'erreur de l'angle du contour. Il est difficile de régler cette méthode pour annuler l'effet de l'erreur sur l'angle.

De ce fait, nous avons implémenté la seconde méthode.

3.4.3.2. la méthode TH avec érosion.

Elle procède de la manière suivante :

Tout point de l'image est testé avec les droites dont l'angle est proche de l'approximation calculée à l'aide du gradient en ce point. Chaque point apparaît dans toutes les droites dont l'angle est dans un intervalle centré sur la valeur approchée issue du gradient.

D'autre part, maintenant qu'un point de l'image apparaît plusieurs fois dans la matrice, il faut le retirer dès que l'on a trouvé à quel segment il appartient.

La valeur d'une case de la MH est définie par le nombre de points qui lui sont associés.

La Figure 3.13 illustre le fonctionnement de la seconde méthode, dont l'algorithme itératif est constitué des étapes suivantes :

- 1- calcul du vote (nombre de points) pour toutes les cellules de la MH,
- 2- détermination de la cellule à vote maximal : elle représente le segment courant,
- 3- retrait de la MH de tous les points appartenant au segment courant,
- 4- test du critère d'arrêt, s'il n'est pas satisfait retourner à 1.

Il y a cependant deux nouveaux seuils à fixer :

- le premier concerne le critère d'arrêt d'analyse de la matrice de Hough, qui dépend de la qualité de l'image, de la nature et de la position des objets à reconnaître.
- Le second est la tolérance sur l'angle du gradient. En effet, une grande tolérance permet d'obtenir de meilleurs résultats sur un image bruitée. Mais des segments d'inclinaisons trop proches risquent d'être confondus. Il faudra donc trouver un compromis.

Trois critères d'arrêt sont définis empiriquement. Ce sont des critères dynamiques, qui évoluent après chaque itération. Ils représentent des taux exprimés par rapport à l'état initial de la MH.

Le premier critère, appelé *maximum*, représente le vote maximal dans une itération donnée.

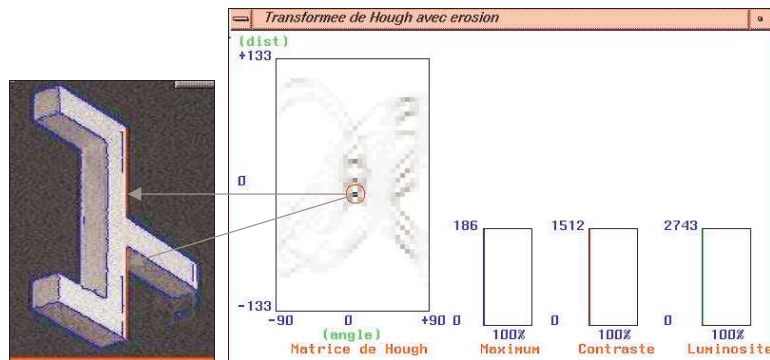
Le second critère, appelé *luminosité*, représente la moyenne des votes des différentes cellules restantes à explorer (v_i) :

$$\text{luminosité} = \text{moyenne} = \frac{\sum_{i=1}^N v_i}{N}$$

Le troisième critère, inspiré de la formule de l'écart type et appelé *contraste*, est donné par :

$$\text{contrast} = \sqrt{\frac{\sum_{i=1}^N (v_i - \text{moyenne})^2}{N}}$$

Après l'évaluation de ces trois critères, nous avons choisi un seuil basé sur le contraste de la matrice. Car celui-ci est faiblement dépendant de la taille et de la complexité (densité de segments) d'une image que les deux premiers. Les critères maximum et luminosité varient en fonction de la taille de l'image, car leur calcul dépend de la taille des segments. En revanche, le calcul du critère contraste dépend de la différence entre la longueur des segments et la moyenne, d'où sa faible variation. L'algorithme s'arrête lorsque le contraste atteint 10% de sa valeur initiale. A ce stade, tous les segments dans nos images de test auront été trouvés.

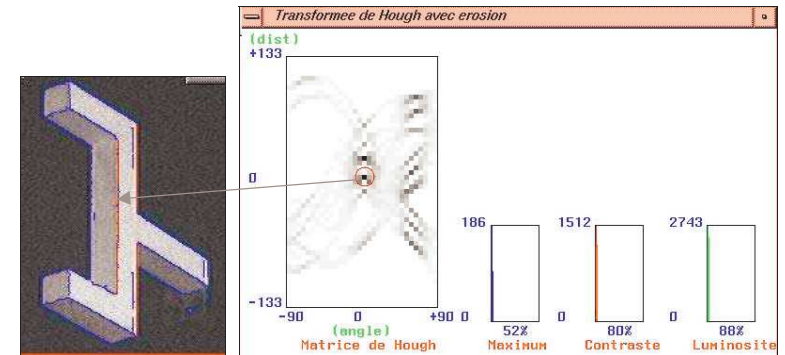


(a) première itération :

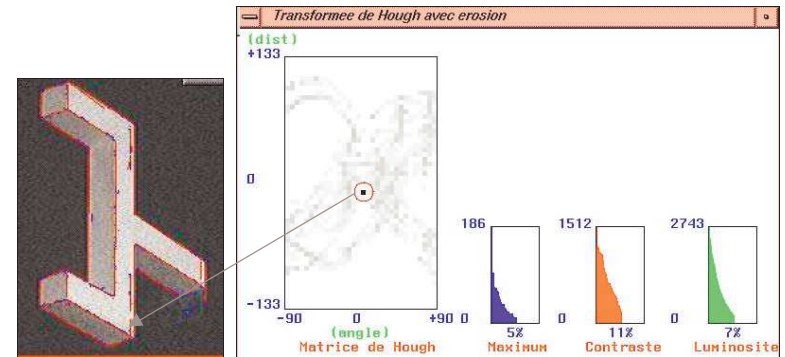
- le vote maximal (cellule de niveau de gris maximum) dans la MH donne le segment le plus long.
- le retrait de la MH de tous les points appartenant au segment courant .

Notes :

- un point est représenté par ses coordonnées polaires dans la matrice de Hough (l'angle de contour par rapport à la verticale et la distance de la droite support du point au centre de l'image).
- dans les rectangles représentant les critères (maximum, contraste et luminosité), l'axe des abscisses représente le nombre de segments traités (itérations). L'axe des ordonnées correspond à la valeur du critère en cours par rapport à celle de la première itération.



(b) seconde itération : le vote maximal s'est déplacé dans la MH, maximum, contraste et luminosité ont diminué.



(c) dernière itération : contraste est proche de 10% (limite fixée empiriquement pour l'arrêt du programme)

Figure 3.13 : La segmentation en utilisant la TH par érosion

La méthode de la TH avec érosion permet donc d'obtenir directement les segments de l'image, à partir des données fournies par Hystérésis. La méthode de la TH avec fusion imposait un regroupement de petits segments, pouvant provoquer de mauvaises fusions entre segments. De plus, elle est plus tolérante vis à vis des bruits contenus dans l'image puisque la tolérance sur l'angle, calculé par les gradients en X et Y, est plus large.

3.4.4. Passage aux paramètres réelles des segments

Le second sous-problème de la détection des segments (mentionné dans l'introduction §3.4.1), est la recherche de la meilleure approximation analytique à partir d'un sous-ensemble de points de contour. En effet, c'est une solution complémentaire et indispensable pour présenter le résultat de la segmentation. Concernant les méthodes de la TH, elles donnent les paramètres approximatifs des droites supports détectées dans l'image dont il faut calculer les paramètres réels.

Zhang [ZHANG 1997] a fait une étude comparative des techniques les plus célèbres pour l'estimation des paramètres. La méthode la plus connue est celle des moindres carrés (MMC), le développement initiale de cette méthode pour trouver les paramètres d'une droite passant par un nuage de points se trouve dans l'annexe 3. Cette méthode est basée sur l'optimisation d'un critère sous forme de somme des carrés des erreurs basées sur des distances algébriques. Elle est efficace lorsqu'une solution analytique est fournie (qui est notre cas), mais elle suppose l'existence d'aucun bruit ou un bruit faible à moyenne valant zéro. Il est prouvé qu'il suffit d'avoir une seule mauvaise donnée pour perturber complètement le résultat.

Parmi les techniques classifiées robustes, nous avons utilisé une variante de la technique de *régression* qui utilise la MMC. Cette technique fonctionne correctement si le bruit ne dépasse pas certaines limites. Elle est suffisante pour notre application, car la méthode de Hough fournie des données (liste de points) limitées grâce à la quantification des paramètres. L'algorithme décrivant cette technique est appliqué à chaque liste de points comme suit :

1. déterminer des paramètres initiaux de la droite, à partir de la liste des points, à l'aide de la MMC (cf. annexe 3),

2. calculer le critère d'erreur $C = \sqrt{\sum_{i=1}^n e_i^2} / n$ (à chaque point p_i est attribué une erreur e_i , variable) : s'il est

inférieur à un seuil S_e (2 pixels), les paramètres sont les bons, aller à 5

3. calculer l'erreur maximale e_{max} , éliminer tous les points ayant $e_i > S_e$ où $S_e = \gamma \cdot (e_{max} - C) / C$. γ est un pourcentage (valant 10% ou plus) qui représente la confiance à l'élimination de points. S_e représente un seuil dynamique qui donnera les points les plus éloignés du critère. Ceci permet d'éliminer plusieurs points de la liste dans une seule itération (sans perte de précision) contrairement à la méthode d'origine (Figure 3.14).

4. retourner à 1

5. projeter tous les points sur la droite support et trouver les extrémités de chaque segment, le résultat est une liste de segments alignés appelée *méta-segment*

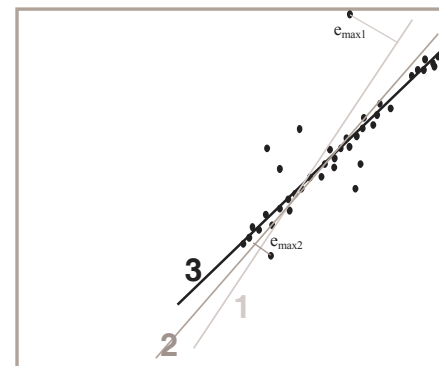


Figure 3.14 : Trois itérations de l'algorithme de régression par la MMC

3.5. Organisation perceptive

3.5.1. Introduction

Lowe [LOWE 1987] est le premier à avoir utilisé la notion de l'*Organisation Perceptive* (OP), dans son système SCERPO (décrit au chapitre 1). L'OP qu'il définit consiste à établir des relations de proximité, de parallélisme et de colinéarité entre les segments de la scène. Elle a été utilisée, depuis, par différents chercheurs pour décrire d'autres relations [SARKAR 1995] [MCCane 1996]. Sarkar l'utilise dans un graphe pour effectuer un suivi d'objets dans une séquence d'images 2D. Le principe est de constituer des groupements de primitives ayant des relations d'invariance ; un petit changement de point de vue ne doit pas influencer sur ces relations. Le système de vision humain semble chercher ce type de regroupement entre primitives de la scène en se basant sur la topologie.

Il est clair que les trois relations mentionnées ci-dessus constituent des invariances ; deux segments en proximité dans une scène réelle le sont toujours dans l'image quelque soit le point de vue. Il en est de même pour la colinéarité, en revanche il n'en est pas dans tous les cas pour le parallélisme. En fait, deux segments parallèles dans une image le sont forcément dans la réalité mais des segments 3D parallèles donnent généralement des

segments convergents vers un point de fuite dans une projection perspective sur l'image. L'OP est utilisée essentiellement pour réduire le nombre d'hypothèses sur l'appariement entre deux ensembles de primitives.

3.5.2. Mise en équations

- Proximité

Nous allons calculer une valeur de signification (probabilité) pour estimer ces relations. Prenons la proximité entre deux points, la valeur recherchée doit représenter la probabilité que cette proximité est réelle ou qu'elle est une simple coïncidence de projection de deux points éloignés. Nous allons nous baser sur les segments, puisque ce sont les primitives extraites des premières du traitement d'images, chaque segment ayant deux extrémités. Deux segments S_i et S_j ayant une proximité entre leurs extrémités E_{i1} et E_{j1} dans l'image, peut signifier qu'ils sont réellement contigus ou que c'est un simple hasard de projection. Pour chaque cas de proximité, nous devons donc calculer la probabilité qu'il soit une simple coïncidence. Puisqu'il n'existe pas de connaissance a priori sur l'image, les calculs doivent se faire sous l'hypothèse d'une distribution uniforme des segments dans la position, l'orientation et l'échelle.

Soit donc une image carrée I_{max} (cela ne constitue aucune simplification du problème), et soit d la densité de points (extrémités 2D) par unité de surface. Le nombre N de points se trouvant dans un disque centré de rayon r et de (Figure 3.15) : $N = d\pi r^2$

Lorsque le nombre N est $\ll 1$, il vaut la probabilité que cette proximité est hasardeuse. En fait, ce nombre étant calculé à la base de l'hypothèse d'une densité uniforme de points, il est imaginable naturellement qu'il peut valoir 1 pour tous les points de l'image. Une proximité entre deux points ayant $N=1$ est considérée un simple hasard. Nous avons introduit dans cette relation uniquement les points, il faut exprimés maintenant les segments.

En fait, la densité d n'est pas indépendante de la longueur des segments dans l'image. Sous l'hypothèse que d ne doit pas changer avec l'échelle de l'image : réduire la taille de l'image par un facteur 2 (cela revient à réduire la longueur des segments par 2) réduira sa surface par 4. Comme la densité de segments doit rester intacte par ce changement, elle doit être multiplier par 4 (elle est alors inversement proportionnelle au carré de la longueur des segments). D étant la densité des segments, la densité d'un segment court l_1 est dominante sur celle d'un segment long l_2 , on exprime alors le segment court dans la relation de N :

$$E_{\text{prox}} = 2D\pi r^2 / l_1^2 \quad (3.6)$$

le facteur 2 représente le fait d'avoir 2 points d'extrémité pour chaque segment. La valeur D est imposé empiriquement par Lowe : $D=1$, elle n'a pas beaucoup d'influence sur le calcul puisqu'une proximité est considérée quand $E_{\text{prox}} \ll 1$.

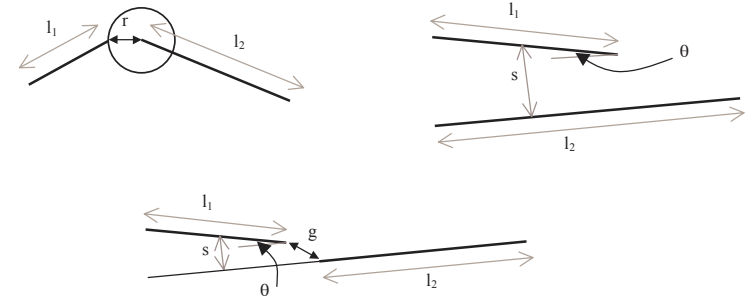


Figure 3.15 : Proximité, parallélisme et colinéarité entre segments

- Parallélisme

Une valeur représentant le parallélisme est déduite avec une analyse semblable. Soit dans la Figure (3.15) les deux segments l_1 et l_2 avec s la distance séparant l_2 du milieu de l_1 et θ et l'angle entre eux :

- l'indépendance de l'échelle donne un nombre de segments D/l_1^2 ,
- en considérant uniquement le rectangle formé par le segment long l_2 et la distance s qui les séparent (indépendance de la position) : $2sl_2$,
- il ne faut considérer que $2\theta/\pi$, l_1 est borné par cette variation de l'orientation par rapport à l_2 (indépendance de l'orientation)
-
- l'ensemble de ces facteurs donne :

$$E_{\text{par}} = (D/l_1^2)(2sl_2)(2\theta/\pi) = 4D\theta l_2 / \pi l_1^2 \quad (3.7)$$

- Colinéarité

- La différence entre la colinéarité et le parallélisme est que les deux segments colinéaires peuvent avoir une distance supplémentaire g qui les sépare. Le rectangle recherché dans ce cas est centré sur l_1+g car l_1 est censé être dans cette zone, on a alors :

$$E_{col} = (D/l_1^2)(2s(g+l_1))(2\theta/\pi) = 4D\theta s(g+l_1)/\pi l_1^2 \quad (3.8)$$

- Remarques

- Lowe n'a pas défini les seuils utilisés pour déterminer les relations mentionnées ci-dessus entre les segments, puisque deux segments sont soit reliés soit non par chacune de ces relations. Nous variions les seuils par expérience entre 0.2 et 0.6 selon le type d'images (densité approximative de segments)
- Les relations mentionnées par Lowe pour décrire le parallélisme et la colinéarité ne sont pas suffisantes. En fait, si θ ou s est proche de zéro, ces relations donnent systématiquement satisfaction. Or, on peut avoir deux segments ni parallèles ni colinéaires ayant $s=0$ mais θ est assez grande ou l'inverse. Nous ajoutons par conséquent un test supplémentaire sur ces valeurs, si l'une des deux est très petite l'autre doit être inférieure à un seuil.

3.6. Représentation et correction du modèle 2D

3.6.1. Introduction

L'élaboration du modèle 2D est basée sur les relations de l'OP entre les segments 2D extraits de l'image. Le modèle 2D est représenté dans une structure de données de type BREP (étudiée au chapitre 2, §2.3.1.2) dont :

- un *sommet 2D* est un ensemble de segments dont chacun se trouvant en proximité d'un seul côté avec tous les autres, il est caractérisé par un «*barycentre*» (point de rencontre de tous ses segments). Le barycentre est calculé par la méthode des moindres carrées minimisant la somme des carrées des distances entre celui-ci et tous les segments (cf. Annexe 3).
- une *chaîne de segments 2D* (appelée simplement *chaîne*) est un ensemble de segments, dont chacun se trouvant en proximité avec un autre et dont aucun ne possède plus qu'un lien par extrémité.

Cette représentation facilite l'accès aux informations nécessaires pour effectuer l'appariement avec un modèle 3D (cf. chapitre 4, §4.2). Le modèle 2D contient des erreurs dues au traitement de l'image (Figure 3.13-c). Des

règles de correction automatique sont adoptées, l'opérateur peut toujours intervenir pour changer ou tester les constituant de ce modèle manuellement.

Nous présentons la recherche des chaînes, ensuite une règle importante de correction automatique des sommets du modèle 2D.

- Recherche des Chaînes de segments

Une fois que les liaisons entre les segments sont trouvées, on peut déterminer les chaînes minimales fermées et les chaînes ouvertes présentes dans l'image. Comme le montre la Figure 3.16, la recherche des chaînes se fait dans un premier temps en tournant dans le sens trigonométrique : segment, sommet, segment etc. Si la chaîne est toujours ouverte le chaînage se poursuit dans le sens contraire à partir du premier segment entré dans la chaîne.

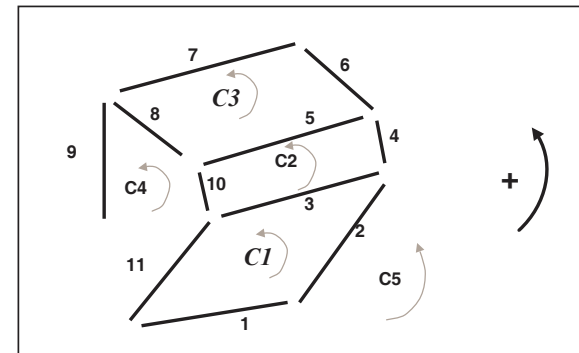


Figure 3.16 : Recherche de chaînes

Pour réaliser ce chaînage, nous utilisons une matrice dont les lignes pointent vers les segments ayant au minimum une relation de proximité et dont les colonnes pointent vers les sommets. Un segment possédant deux sens de parcours (à droite et à gauche), pour chaque passage le sens concerné est marqué pour éliminer toute redondance éventuelle. L'algorithme suivant est appliqué sur les segments $S_{i=0, n-1}$ de la matrice :

début : $i=0, k=0$

A. à gauche :

- Si les deux sens du S_i sont marqués, aller à C, sinon entrer S_i dans la chaîne C_k et choisir un sens non marqué du S_i
- chercher dans le sens actuel du S_i un sommet P : si trouvé, continuer, sinon aller à B
- à partir de P chercher le segment S_j ayant l'angle minimal avec S_i :

si trouvé reprendre A en partant de S_j , sinon aller à B

B. à droite :

- à partir du dernier segment considéré dans A et dans le sens contraire au dernier sens parcouru
- si ce sens n'est pas marqué continuer, sinon aller à C
- chercher dans le sens actuel un sommet P : si trouvé continuer sinon aller à C
- à partir de P chercher le segment S_j ayant l'angle maximal avec S_i : si trouvé entrer le dans C_k reprendre B en partant de S_j , sinon aller à C
- C. si elle n'est pas vide enregistrer la chaîne C_k et poser $k=k+1$,
poser $i=i+1$, si $i=n$ fin sinon aller à A

Cet algorithme est de $O(k)$ où k est le nombre de chaînes trouvées, car un seul parcours par chaîne est effectué.

Un autre avantage est qu'il ne possède pas de redondance de chaînes grâce au marquage des segments dans les deux sens. Certaines descriptions sont calculées pour chaque chaîne :

- chaîne ouverte ou fermée (le dernier segment est en lien avec le premier),
- nombre de segments dans la chaîne,
- chaîne convexe (n'ayant que des angles inférieur à 90°) ou concave
- listes des segments colinéaires dans la chaîne
- listes de segments parallèles (ou convergentes si plus de 2)

Les chaînes sont utilisées dans l'étape suivante (chapitre 4) et appariées avec des faces de l'objet à reconnaître selon leurs descriptions. Les sommets, segments et chaînes de segments sont organisés dans le modèle 2D, l'opérateur peut intervenir pour changer ou tester ce modèle. La Figure 3.17 montre un le résultat de chaînage appliqué à l'objet pince.

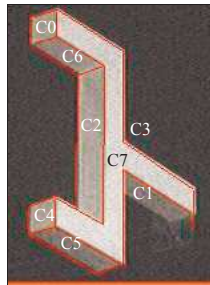


Figure 3.17 : Le modèle 2D de l'image de l'objet Pince

- Correction des sommets

Dans certains cas, la position des sommets est erronée. Cela se produit quand trois segments ou plus ont leurs extrémités proches correspondant en réalité à plusieurs sommets. Ou bien, quand un segment est partiellement caché par une face (Figure 3.18) Dans ce cas, le sommet est déplacé, il en résulte une erreur supplémentaire à l'appariement.

Pour éviter cette erreur, nous calculons la distance entre chaque extrémité de segment et leur barycentre. Tant que la somme de ces distances dépasse une valeur seuil S_v , on élimine du calcul du sommet le segment le plus éloigné.

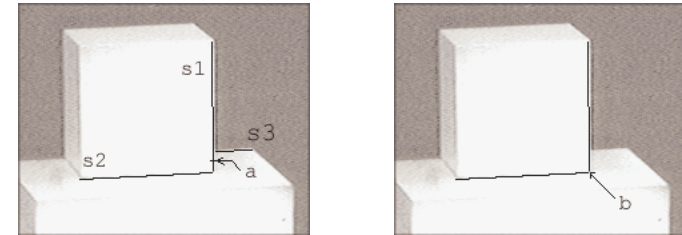


Figure 3.18 : Exemple d'erreur dans la recherche d'un sommet 2D et sa correction

Dans la Figure 3.18, le sommet (a) comporte une erreur. Il est déplacé car le segment (s3), partiellement caché, est intervenu dans le calcul.

Après élimination de (s3), le sommet retrouve une position normale (b), ce qui permet de gagner en précision au niveau de l'appariement.

L'expérience montre que lorsque tous les segments pris en compte pour un sommet sont effectivement concernés, la somme des distances entre les segments et le sommet est presque nulle et dans tous les cas, inférieure à 4 pixels. C'est cette valeur qui a été choisie comme seuil S_v .

3.7. Conclusion

Dans ce chapitre, nous avons développé les différents constituant du système de traitement d'images destiné aux objets polyédriques. Ce système remplit les tâches importantes d'extraction des primitives de l'image (segments, sommets et chaînes de segments) représentées dans un modèle 2D de la scène. Ce modèle a l'avantage de donner un accès rapide aux informations nécessaire pour la suite du système de reconnaissance de polyèdres.

Le traitement est constitué de détection et de segmentation de contours, de la recherche des relations de perception entre les segments, de représentation par un modèle et de correction de celui-ci. La détection de contours se fait par un filtrage de type gradient, de l'amincissement et du suivi des contours par hystérésis. La segmentation de contours détectés est réalisée en appliquant la Transformée de Hough. Nous avons apporté deux nouvelles méthodes permettant d'améliorer l'extraction des segments obtenus à partir de la TH. L'organisation perceptive est appliquée afin d'obtenir les relations de proximité, de colinéarité et de parallélisme entre les segments.

Loin d'être parfaite, l'implantation de ce logiciel a permis de prendre conscience des problèmes existants dans le traitement d'images et de donner des perspectives pour la suite. Les résultats obtenus permettant de fournir les éléments essentiels à la reconnaissance, nous nous sommes arrêtés à ce stade pour s'occuper de l'objectif principal.

-

4. RECONNAISSANCE D'OBJETS POLYEDRIQUES

4.1. Introduction

La reconnaissance est constituée de deux étapes, une de prédiction (appariement) et une autre de vérification. Une prédiction ou *hypothèse* confronte un aspect (vue) de l'objet, extrait d'un graphe de l'ensemble des aspects de l'objet, à l'image de celui-ci (§1.1.1.2). Deux méthodes de mise en correspondance 2D/3D ont été développées, l'une est basée sur le graphe d'appariement, l'autre sur le hachage géométrique. L'appariement effectué est applicable aux objets polyédriques. L'étape de vérification teste la véracité de cette hypothèse en utilisant les méthodes de recalage 3D. En effet, celles-ci permettent le calcul des transformations rigides (rotation et translation) subies par l'objet en connaissant son contour dans l'image et son modèle 3D. Ces transformations rigides, ainsi que le modèle de la caméra, sont appliquées au modèle 3D de l'objet. La projection obtenue est ensuite comparée au contour de l'objet.

L'objectif de ce chapitre est l'étude et la réalisation de la reconnaissance d'objets tridimensionnels (3D) à partir d'une seule image de luminance. A ce stade, on est confronté au problème de reconnaissance d'objets polyédriques, convexes ou concaves, à partir d'une représentation polygonale de la scène (modèle 2D, §3.6). Il s'agit d'identifier et de localiser des polyèdres, de la BD3D, présents dans la scène prise d'un point de vue quelconque. La procédure se divise en deux étapes : appariement de la BD3D avec le modèle 2D de la scène et la localisation des objets appariés.

Dans la première partie de ce chapitre, nous présentons la méthode basée sur le graphe d'appariement élaboré à partir des modèles 2D et 3D. Ce graphe est constitué d'hypothèses d'appariement entre entités des modèles 2D et 3D. Une recherche de sous graphes compatibles (appelés *cliques maximales*) est effectuée. A chaque sous graphe est attribué un degré de compatibilité représenté par un score qui prend en compte le nombre d'éléments géométriques correctement appariés.

La seconde partie est consacrée au développement de la méthode basée sur le hachage géométrique. Les aspects des modèles 3D de la base sont mémorisés dans une table de hachage dont les entrées constituent des «clés» pour accéder aux aspects. Les entités géométriques du modèle 2D (coordonnées relatives des sommets) sont utilisées afin d'accéder à cette table et voter pour les aspects correspondants.

La dernière partie du chapitre présente les méthodes utilisées pour la vérification des hypothèses ressortissant de l'étape d'appariement. Nous avons développé une méthode utilisant trois différentes approches de recalage d'objets.

4.2. Méthode du Graphe d'Appariement (GApp)

Les primitives de la scène sont des chaînes² polygonales de segments, des segments de droite ou des sommets 2D extraits de l'image, alors que les primitives du modèle sont des faces³, des arêtes et des sommets connus par leur position 3D dans un repère lié à l'objet. Le but est l'extraction des hypothèses maximales plausibles sur l'appariement, entre ces deux ensembles de primitives, à l'aide d'un graphe d'appariement

Le *graphe d'appariement* (GApp) est un graphe de compatibilité non-orienté dont les nœuds sont des hypothèses d'appariement chaîne-face (appelés *hypothèses élémentaires*) et dont les arcs représentent la compatibilité mutuelle entre les nœuds. La compatibilité entre deux nœuds est la cohérence vis-à-vis des contraintes employées. Dans notre cas, il s'agit de cohérences topologiques (forme, visibilité et connexité des faces) et géométriques (rapport de superficies). L'idée d'utiliser les chaînes dans le GApp vient des travaux de Pampagnin qui l'appelle *graphe de compatibilité* [PAMPAGNIN 1990, 1991]. La Figure 4.1 ci-dessous présente les différentes étapes d'élaboration du GApp.

Les nœuds sont directement issus de la topologie (forme) des faces et des chaînes. La topologie est représentée par la forme du contour : nombre et disposition des côtés (convexité, parallélisme et colinéarité).

² Une chaîne polygonale de segments de droite 2D est appelée simplement *chaîne*, Un segment de droite 2D est appelée également *segment*

³ Une face 3D est appelée simplement *face*, une arête 3D est appelée également arête

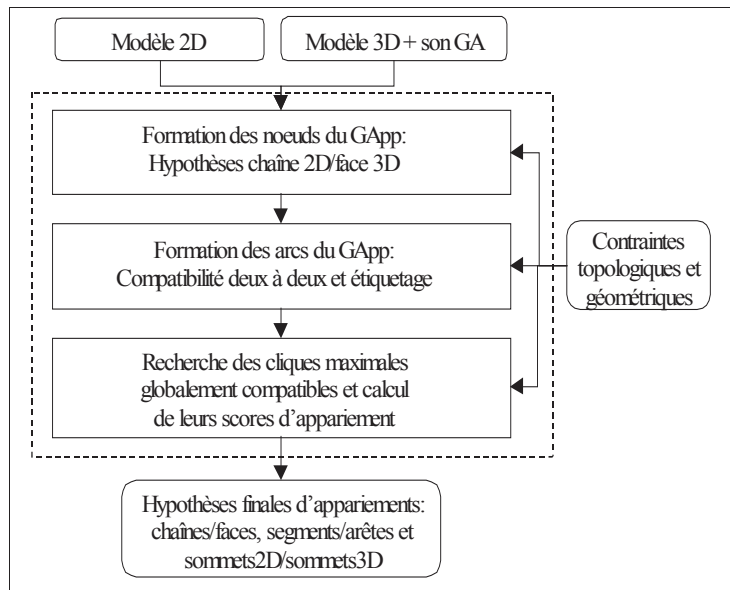


Figure 4.1 : Elaboration du graphe d'appariement

Un arc relie deux nœuds si les quatre critères suivants sont satisfaits:

- critère logique : les chaînes dans les deux nœuds sont distinctes, il est de même pour les faces,
- critère de visibilité : les deux faces apparaissent dans l'un des aspects du modèle 3D (la visibilité des deux faces est possible, information issue du graphe d'aspects),
- critère de connexité : les deux chaînes ont la même relation de connexité que les deux faces (sommets en commun, segment en commun ou rien),
- critère d'invariance : le test des invariants géométriques, basés sur des rapports de surfaces entre faces et chaînes est satisfait

Les arcs sont étiquetés par des appariements segment-arête lors de la construction du GApp. Ces appariements sont déduits de l'analyse de connexité (adjacence) entre les faces et les chaînes des deux nœuds reliés.

Nous cherchons ensuite des sous graphes complets du GApp, appelés *cliques maximales*. Une clique maximale doit être également soumise aux critères suivants :

- critère logique : tous les nœuds de la clique sont reliés deux à deux par des arcs,
- critère de visibilité : toutes les faces de la clique doivent apparaître dans un aspect du modèle 3D (information fournie par le graphe d'aspects GA),
- critère de connexité : tous les appariements, entre les segments 2D et les arêtes 3D de la clique, doivent être cohérents

Les cliques maximales forment les hypothèses plausibles sur l'appariement. Elles sont munies d'un score basé sur le nombre d'appariements sommet3D-sommet2D obtenus.

Afin de faciliter la description du GApp, nous allons nous baser sur des exemples. La Figure 4.2 montre un exemple de graphe d'appariement GApp entre un modèle 3D d'une pyramide simple à cinq faces et un modèle 2D, obtenu à partir de son image et contenant deux chaînes.

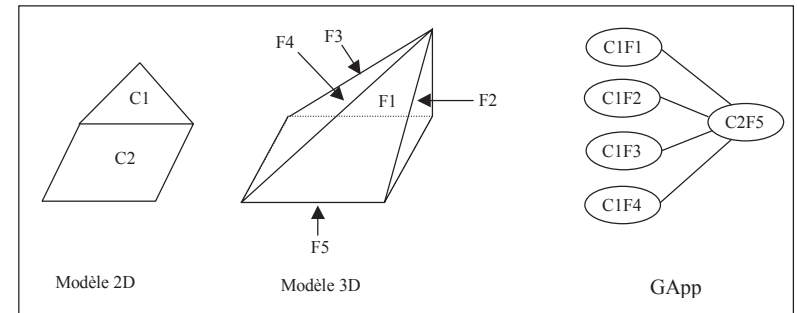


Figure 4.2 : Exemple d'un graphe d'appariement simple

Les nœuds du GApp représentent les hypothèses élémentaire chaîne-face. Ainsi, C1F1, C1F2, C1F3 et C1F4 représentent les différentes possibilités d'apparier la chaîne C1 avec des faces du modèle 3D, tandis que C2F5 représente la seule hypothèse existante d'appariement entre C2 et les faces du modèle 3D. Dans cet exemple, les hypothèses élémentaires sont basées uniquement sur le nombre de côtés des chaînes.

Les arcs du GApp représentent la compatibilité mutuelle entre l'hypothèse C2F5 et toutes les hypothèses de C1. Il faut noter que C1F1, par exemple, n'est pas compatible avec C1F2, car on ne peut apparier la même chaîne avec deux faces en même temps.

Les cliques maximales extraites de ce graphe, sont les sous graphes : $C_2F_5 C_{1F_{1..4}}$.

L'utilisation du GApp permet donc :

- d'intégrer explicitement les contraintes topologiques et géométriques des modèles 3D et du modèle 2D de la scène, ainsi nous pouvons ajouter ou modifier celles-ci sans changer l'implémentation,
- de limiter au maximum la combinatoire d'hypothèses générées en éliminant des sous-graphes avant d'aller plus loin dans la recherche.

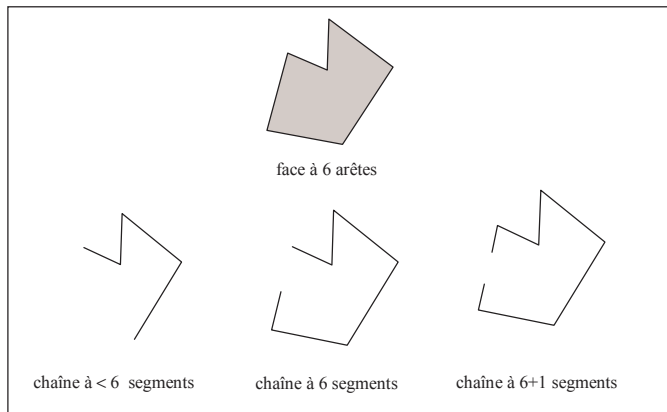
Le GApp est appliqué sur le modèle 2D et chacun des modèles 3D à reconnaître. Les hypothèses issues de tous les GApp sont ordonnées selon leur score et soumises à l'étape de vérification (§4.4). Nous présentons en détail dans les paragraphes suivants la méthode d'élaboration d'un GApp : formation des nœuds, des arcs et des cliques maximales.

4.2.1. Formation des nœuds du GApp

Rappelons nous que les chaînes possèdent les attributs topologiques (caractéristiques de forme) suivants :

- convexité : une chaîne est convexe si tous les angles (intérieur) entre ses segments sont inférieurs à π , sinon elle est concave
- fermeture : une chaîne (liste de segments) est fermée si son dernier segment est relié au premier, sinon elle est ouverte
- nombre de segments
- liste pointant vers les segments colinéaires dans la chaîne
- liste pointant vers les segments parallèles ou convergents vers un point de fuite, dans la chaîne

Ces attributs caractérisent également les faces, sauf la fermeture car une face 3D est toujours fermée. Les nœuds du GApp sont créés à la base de ces attributs, chaque couple chaîne-face doit vérifier les quatre critères suivants :



contour, convexité, colinéarité et parallélisme.

- Critère de contour

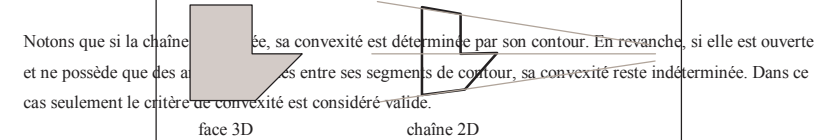
- une chaîne fermée est mise en correspondance avec toute face ayant le même nombre de côtés,
- une chaîne ouverte est mise en correspondance avec toute face ayant plus de côtés, autant ou un côté de moins (Figure 4.3). Une chaîne peut avoir un côté de plus que la face correspondante lorsque ses segments d'ouverture sont issus de la même arête.

Figure 4.3 : Critère de contour pour une hypothèse élémentaire

- Critère de convexité

La convexité est un invariant topologique qui découle de la colinéarité. Si un angle entre deux arêtes est inférieur à π , sa projection l'est toujours (de même s'il est supérieur à π) :

- une chaîne convexe est mise en correspondance avec toute face convexe
- une chaîne concave est mise en correspondance avec toute face concave



- Critère de colinéarité

Une face peut avoir deux arêtes ou plus colinéaires (liste d'arêtes non forcément liées sur le contour), cette caractéristique est invariante par projection perspective. Une telle face peut être appariée uniquement avec des chaînes ayant une liste de segments colinéaires situés sur le contour de la même manière que les arêtes. La Figure 4.4 montre que, même s'il y a un nombre de segments correspondant aux arêtes colinéaires, il faut qu'il soit situé de la même manière sur le contour (cas de la chaîne C_2).

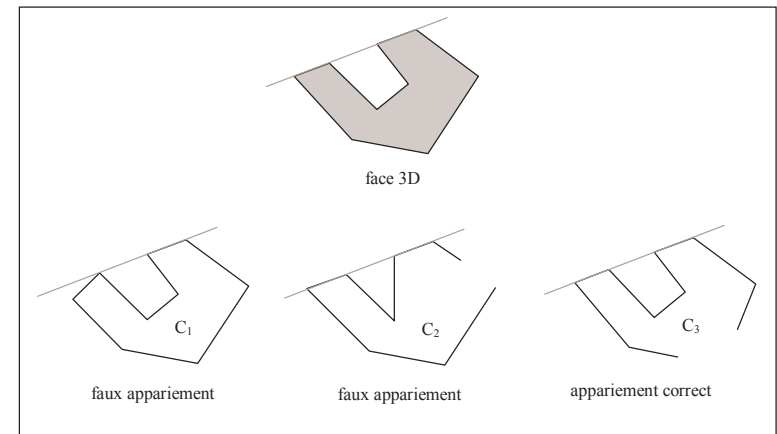


Figure 4.4 : Critère de colinéarité permettant d'éviter de faux appariements

- Critère de parallélisme

Ce critère est semblable au précédent, la seule différence est dans le parallélisme qui n'est pas entièrement conservé. En effet, des arêtes parallèles peuvent donner des segments convergents vers un même point de fuite par projection perspective, auquel cas il faut disposer de trois segments au moins pour évaluer ce critère (Figure 4.5).

Figure 4.5 : Les arêtes parallèles peuvent donner des segments qui convergent vers un point de fuite

- Mise en correspondance segments-arêtes

Les possibilités de mise en correspondance entre les segments de la chaîne et les arêtes de la face, se font en fonction des trois derniers critères. Une liste de ces possibilités est établie dès que possible afin d'alléger le graphe dès la formation des arcs, car un test de cohérence de mise en correspondance mutuelle entre les nœuds, permettra d'éliminer les arcs infaisables. Dans la Figure 4.6, les deux concavités symétriques de la face donnent deux possibilités de mise en correspondance segments-arêtes, soit (a_i, s_i) pour $i=1..10$ soit $(a_{s_{i+5}}$ pour $i<6$ et a_{s_i} ailleurs). Le critère de colinéarité est compatible avec ce résultat.

Figure 4.6 : Deux différentes possibilités de mise en correspondance segments-arêtes

- Exemple

Dans la Figure 4.7, on trouve un modèle CAO d'un polyèdre et une image de segments simulée. L'application des critères précédents sur l'ensemble des faces et des chaînes donne les 19 nœuds du GApp illustrés dans la Figure 4.8.

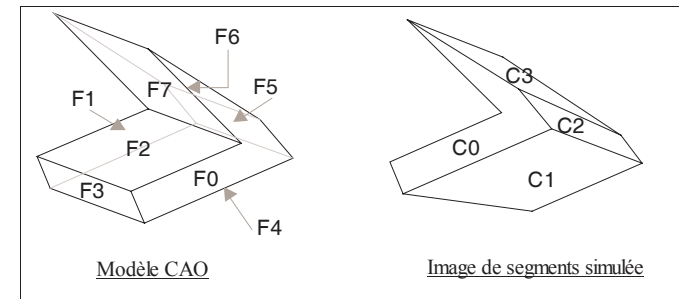
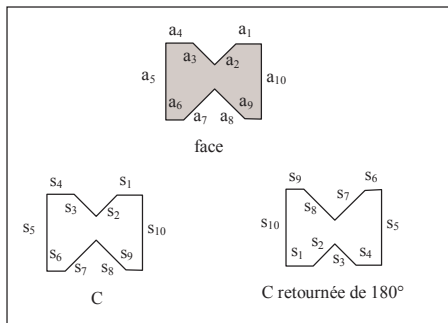


Figure 4.7 : Exemple d'appariement modèle 3D/modèle 2D simulé

		Faces							
		F ₀	F ₁	F ₂	F ₃	F ₄	F ₅	F ₆	F ₇
Chaînes	C ₀	N ₀	≠	□	□	□	□	□	□
	C ₁	□	□	N ₁	N ₂	N ₃	N ₄	N ₅	N ₆
	C ₂	□	□	N ₇	N ₈	N ₉	N ₁₀	N ₁₁	N ₁₂
	C ₃	□	□	N ₁₃	N ₁₄	N ₁₅	N ₁₆	N ₁₇	N ₁₈

Symboles	N _x	Nœud	□	Critère de contour ou de convexité non-satisfait	≠	Parallélisme non-satisfait
----------	----------------	------	---	--	---	----------------------------

Figure 4.8 : Les nœuds trouvés pour le GApp du prisme

Montrons un cas que nous a donné le programme, il s'agit du couple C_0F_1 . Ce nœud n'a pas eu lieu, car il ne vérifie pas le critère de parallélisme, même s'il existe une forte similitude (Figure 4.9). La raison de l'échec de cet appariement est dans la modélisation de l'objet, tenant compte de l'orientation des faces vers l'observateur, le parcours des arêtes étant toujours dans le sens trigonométrique. La chaîne extraite de l'image correspond bien à la face opposée F_0 . Elle a par conséquent une orientation différente de celle de F_1 .

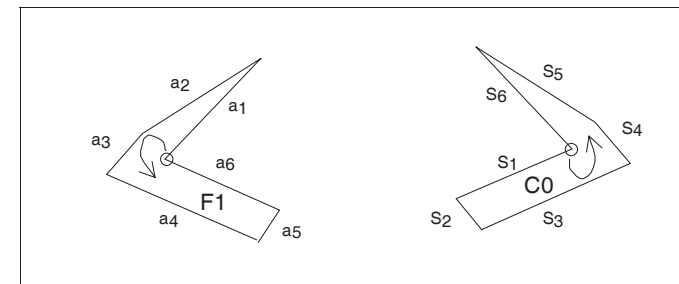


Figure 4.9 : Mise en correspondance entre les segments de C_0 et les arêtes de F_1
(S_2, S_4, S_6 sont quasi parallèles mais pas a_2, a_4, a_6)

4.2.2. Formation des arcs du GApp

Le graphe d'appariement se construit de façon incrémentale en examinant les nœuds du GApp deux par deux. Les arcs représentent la *compatibilité mutuelle* pour chaque paire de nœuds (hypothèses chaîne-face). Nous adoptons les critères suivants pour examiner la compatibilité : logique, visibilité, connexité et invariance.

- Critère logique

Un critère simple et efficace pour écarter plus que la moitié des liens entre les nœuds du GApp est celui de la cohérence logique (voir Figure 4.13 concernant le GApp du prisme). En fait, les chaînes dans les deux nœuds doivent être distinctes, car une chaîne ne peut pas s'apparier avec deux faces en même temps. La même face ne doit également pas apparaître dans les deux nœuds.

- Critère de visibilité

Les deux faces contenues dans les deux nœuds doivent apparaître dans l'un des aspects du modèle 3D (la visibilité des deux faces est possible). Cette information est issue du graphe d'aspects GA calculé hors-ligne à partir du modèle 3D (§2.3.3). Nous utilisons uniquement la visibilité des faces 3D, les informations du GA sont réduites au minimum dans la base de données afin de refléter strictement les besoins du GApp.

- Critère de connexité

Les deux chaînes doivent avoir la même relation de connexité que les deux faces (sommet en commun, segment en commun ou rien). Nous numérotions les cas possibles de satisfaction de ce critère (Figure 4.10) :

1. Les deux chaînes sont fermées :
 - a. les deux chaînes ont un segment en commun et les deux faces aussi,
 - b. les deux chaînes ont un sommet en commun et les deux faces aussi,
 - c. les deux chaînes ne sont pas adjacentes, les faces non plus,
2. L'une des deux chaînes est au moins ouverte. Ce dernier cas signifie toutes les possibilités d'adjacence qu'elles peuvent avoir.

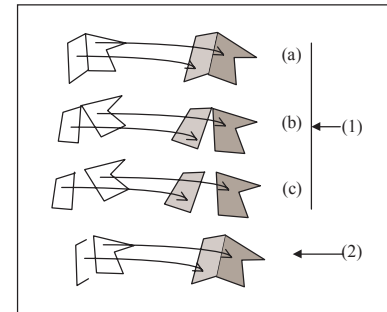
Figure 4.10 : Critère de connexité pour les arcs

Dans le cas a et b seulement, on peut établir par évidence une correspondance entre les segments et les arêtes. Les contours de chaque couple chaîne-face sont parcourus à partir du segment (ou sommet) caractérisant l'adjacence. A chaque arc est attribué alors, si possible, des listes de mise en correspondance segment-arête appelées *étiquetage*. Si cet étiquetage n'est pas cohérent avec ceux créés lors de la formation des nœuds correspondants (§4.2.1), l'arc échoue.

- Critère d'invariance géométrique

Les invariants topologiques (de forme) ont été étudiés au chapitre 3 (§3.5), les relations impliquées prennent en compte indirectement des mesures géométriques sur la longueur et l'orientation des segments. Un *invariant géométrique* se différencie par des mesures directes sur la géométrie des primitives impliquées. Le critère d'invariance géométrique, mis en œuvre par Zhu et al., [ZHU 1995], est utilisé pour tester le rapport des superficies entre une paire de chaînes et une paire de faces. Il est prouvé, sous certaines conditions, qu'une paire de faces possède une valeur invariante lors d'une projection perspective sur une image de n'importe quel point de vue dans l'espace. Nous allons présenter cet invariant et analyser son utilité pour l'appariement.

Soient deux faces non-coplanaires représentées chacune par quatre sommets ABCD et CDEF, dont deux en



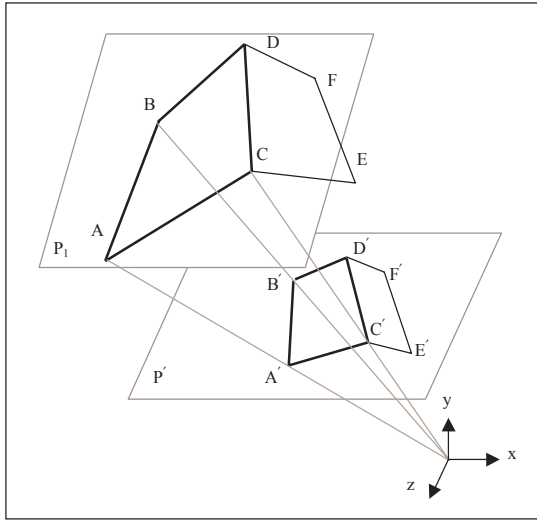
commun, chaque face ne contenant pas trois sommets colinéaires. Soient $A'B'C'D$ et $C'D'E'F'$ respectivement les projections des deux faces sur le plan image, comme le montre la Figure 4.11. Le sommet $A(x_A, y_A, z_A)$

s'exprime dans le plan $P_1(a, b, c, d)$ comme suit :

$$\frac{a}{d}x_A + \frac{b}{d}y_A + \frac{c}{d}z_A + 1 = 0$$

De la même manière, son image $A'(u_A, v_A, w_A)$ s'exprime dans le plan image $P'(e, f, g, h)$:

$$\frac{e}{h}u_A + \frac{f}{h}v_A + \frac{g}{h}w_A + 1 = 0$$



La projection perspective donne : $\frac{x_A}{z_A} = \frac{u_A}{w_A}$ et $\frac{y_A}{z_A} = \frac{v_A}{w_A}$

Les précédentes équations peuvent s'écrire dans la forme matricielle suivante, [ZHU 1995] :

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & a/d & b/d & c/d \end{bmatrix} \begin{bmatrix} 1 \\ x_A \\ y_A \\ z_A \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & e/h & f/h & g/h \end{bmatrix} \begin{bmatrix} 1 \\ u_A \\ v_A \\ w_A \end{bmatrix}$$

Figure 4.11 : Projection de deux faces à 4 points chacune

Cette équation se généralise aux points A,B,C et leurs images :

$$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & a/d & b/d & c/d \end{bmatrix} \begin{bmatrix} 1 \\ x_A \\ y_A \\ z_A \end{bmatrix} = \begin{bmatrix} 1 \\ x_B \\ y_B \\ z_B \end{bmatrix} \begin{bmatrix} 1 \\ x_C \\ y_C \\ z_C \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & e/h & f/h & g/h \end{bmatrix} \begin{bmatrix} 1 \\ u_A \\ v_A \\ w_A \end{bmatrix} \begin{bmatrix} 1 \\ u_B \\ v_B \\ w_B \end{bmatrix} \begin{bmatrix} 1 \\ u_C \\ v_C \\ w_C \end{bmatrix}$$

Cette équation est de la forme : $M \cdot Q_{ABC} = N \cdot Q_{A'B'C'}$

M et N représentent les paramètres des plans de l'objet et de l'image, les matrices Q représentent les coordonnées des points et de leurs images. Pour trouver un invariant, il ne faut laisser que des valeurs relatives à l'objet et à son image. Les déterminants de la dernière équation sont calculés et généralisés à quatre groupes de points ABD, ABC, FEC et FED, il vient :

$$\frac{\|M_1\| \cdot \|Q_{ABD}\| \frac{1}{z_A} \frac{1}{z_B} \frac{1}{z_D} \|M_2\| \cdot \|Q_{FEC}\| \frac{1}{z_F} \frac{1}{z_E} \frac{1}{z_C}}{\|M_1\| \cdot \|Q_{ABC}\| \frac{1}{z_A} \frac{1}{z_B} \frac{1}{z_C} \|M_2\| \cdot \|Q_{FED}\| \frac{1}{z_F} \frac{1}{z_E} \frac{1}{z_D}} = \frac{\|N\| \cdot \|Q_{A'B'D'}\| \frac{1}{w_A} \frac{1}{w_B} \frac{1}{w_D} \|N\| \cdot \|Q_{F'E'C'}\| \frac{1}{w_F} \frac{1}{w_E} \frac{1}{w_C}}{\|N\| \cdot \|Q_{A'B'C'}\| \frac{1}{w_A} \frac{1}{w_B} \frac{1}{w_C} \|N\| \cdot \|Q_{F'E'D'}\| \frac{1}{w_F} \frac{1}{w_E} \frac{1}{w_D}}$$

Cela donne :

$$\frac{\|Q_{ABD}\| \cdot \|Q_{FEC}\|}{\|Q_{ABC}\| \cdot \|Q_{FED}\|} = \frac{\|Q_{A'B'D'}\| \cdot \|Q_{F'E'C'}\|}{\|Q_{A'B'C'}\| \cdot \|Q_{F'E'D'}\|}$$

avec : $\|Q_{ABC}\| = \begin{vmatrix} 1 & 1 & 1 & 1 \\ 0 & x_C & x_B & x_A \\ 0 & y_C & y_B & y_A \\ 0 & z_C & z_B & z_A \end{vmatrix} = \begin{vmatrix} x_C & x_B & x_A \\ y_C & y_B & y_A \\ z_C & z_B & z_A \end{vmatrix}$ valant six fois le volume formé par

les sommets ABC et l'origine. L'équation s'écrit donc par rapport aux surfaces formées par les triplets de points, notées S et la distance entre leurs plans support et l'origine :

$$\frac{d_1 \cdot S_{ABD} \cdot d_2 \cdot S_{EFC}}{d_1 \cdot S_{ABC} \cdot d_2 \cdot S_{EFD}} = \frac{d \cdot S_{A'B'D'} \cdot d \cdot S_{E'F'C'}}{d \cdot S_{A'B'C'} \cdot d \cdot S_{E'F'D'}}$$

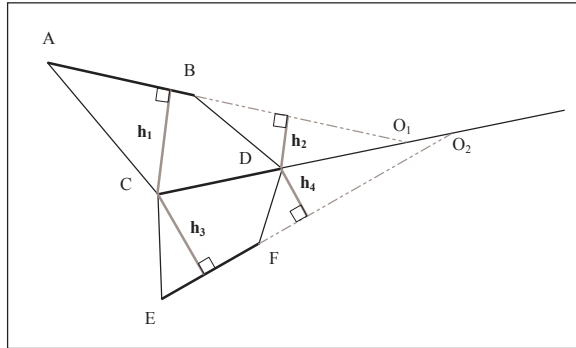
L'invariant exprimé uniquement par des rapports de surfaces est donc :

$$Inv_1 = \frac{S_{ABD} S_{EFC}}{S_{ABC} S_{EFD}} = \frac{S_{A'B'D'} S_{E'F'C'}}{S_{A'B'C'} S_{E'F'D'}} \quad (4.1)$$

Les surfaces de ces triangles sont indépendantes de tout repère. L'avantage de cet invariant est qu'il donne rapidement une validation ou un rejet d'un appariement entre un couple de faces 3D et son image 2D. L'inconvénient étant qu'avant son application, il faut vérifier les conditions suivantes :

- trouver quatre sommets dans chaque face dont chaque triplet est non-colinéaire,
- disposer de deux sommets communs entre les deux faces

- prouver qu'il n'existe pas un triplet de lignes formées par les six points dont les lignes se coupent en un seul point (point de fuite) ou sont parallèles (se coupant à l'infini).



Nous montrons la dernière condition en analysant le calcul des surfaces. Dans la Figure 4.12, les surfaces sont calculées à l'aide des bases et de hauteurs de chaque triangle :

$$Inv_1 = \frac{CD \cdot h_1 \cdot CD \cdot h_3}{CD \cdot h_2 \cdot CD \cdot h_4} = \frac{h_1}{h_2} \cdot \frac{h_3}{h_4} = \frac{CO_1 \cdot DO_2}{DO_1 \cdot CO_2} \quad (4.2)$$

Figure 4.12 : L'invariant est le rapport des distances $h_{i=1..4}$

Il est clair que lorsque O_1 et O_2 sont confondus, l'invariant vaut 1 quelque soit l'objet. L'équation 4.2 est utile lorsqu'on dispose uniquement des deux points C, D et des lignes porteurs des autres points, car il suffit de calculer $h_{i=1..4}$ par projection de C et D sur ces lignes.

C'est un critère efficace pour lever l'ambiguïté que les contraintes topologiques ne peuvent relever. En revanche, ce critère est inapplicable lorsque les quatre points d'extrémité convergent vers l'arête commune aux deux faces, d'où l'obligation de chercher d'autres sommets sur les deux faces vérifiant les conditions mentionnées ci-dessus. Pour mettre en œuvre ce critère, il faut établir un seuil que l'invariant ne doit pas dépasser, entre l'objet et son image. Ce seuil est important car le traitement d'images comporte toujours des erreurs de localisation (pouvant être considérées comme des erreurs de projection perspective). Un seuil de 5% constitue une marge suffisante pour couvrir ces erreurs.

• Exemple sur la formation des arcs

Reprenons l'exemple du prisme de la Figure 4.7, les nœuds de son GApp correspondant ont été calculés précédemment (Figure 4.8). L'application des critères de formation des arcs donne le résultat illustré dans la Figure 4.13. Le graphe d'aspect du prisme ne contient pas d'aspect ayant un des sous-ensembles de faces

suyvants : $\{F_2, F_4\}$, $\{F_3, F_5\}$, $\{F_5, F_7\}$. Il est clair, pour la Figure 4.7 du prisme, que ces sous-ensembles constituent des faces opposées ne pouvant pas être vues simultanément. Chaque arc reliant deux nœuds ayant l'un de ces sous-ensembles est écarté (critère de visibilité).

		Nœuds du GApp																	
Arcs du GApp	N_0																		
	C	N_1																	
	C	L	N_2																
	A_1	L	L	N_3															
	C	L	L	L	N_4														
	C	L	L	L	L	N_5													
	C	L	L	L	L	L	N_6												
	C	L	A_8	V	C	C	A_{24}	N_7											
	C	A_4	L	A_{12}	V	C	C	L	N_8										
	C	V	A_9	L	A_{16}	C	C	L	L	N_9									
	A_2	C	V	A_{13}	L	A_{19}	C	L	L	L	N_{10}								
	C	C	C	C	A_{17}	L	A_{25}	L	L	L	L	N_{11}							
	C	A_5	C	C	C	A_{20}	L	L	L	L	L	L	N_{12}						
	C	L	C	V	A_{18}	A_{21}	C	L	A_{30}	V	C	C	A_{38}	N_{13}					
	C	C	L	C	V	A_{22}	A_{26}	A_{28}	L	A_{32}	V	C	C	L	N_{14}				
	C	V	C	L	C	A_{23}	A_{27}	V	A_{31}	L	A_{34}	C	C	L	L	N_{15}			
	C	A_6	V	C	L	C	V	C	V	A_{33}	L	A_{36}	V	L	L	L	N_{16}		
	A_3	A_7	A_{10}	A_{14}	C	L	C	C	C	C	A_{35}	L	A_{39}	L	L	L	L	N_{17}	
C	C	A_{11}	A_{15}	V	C	L	A_{29}	C	C	V	A_{37}	L	L	L	L	L	L	N_{18}	

Figure 4.13 : Le GApp avec les arcs valides reliant les nœuds de la Figure 4.8

L'application du critère d'invariance géométrique en dernier n'a pas éliminé de nouveaux arcs. L'arc N_0N_5 par exemple, ne vérifie pas ce critère, mais il a été éliminé auparavant par le critère de connexion. Certains arcs n'ont pas pu appliquer l'invariance car leurs deux faces contiennent seulement 6 points et 3 segments parallèles, tels que l'arc A_8 aux faces F_2 et F_3 . Ceci met en évidence que le critère d'invariance géométrique est plus efficace pour les objets ne comportant pas de symétrie et ayant des faces assez complexes.

4.2.3. La recherche des cliques maximales

Pour réduire le nombre d'hypothèses finales générées par le GApp, nous cherchons des sous-graphes complets appelés *cliques maximales*, vérifiant les critères de compatibilité globale suivants : logique, visibilité et connexité.

1. critère logique : tous les nœuds de la clique sont reliés deux à deux par des arcs,
2. critère de visibilité : toutes les faces de la clique doivent apparaître dans un aspect du modèle 3D (information fournie par le graphe d'aspects GA). Ce test est nécessaire car la visibilité mutuelle entre toutes les deux faces n'entraîne pas forcément la visibilité globale de l'ensemble des faces.
3. critère de connexité : tous les appariements segment-arête de la clique (information fournie par l'étiquetage des arcs), doivent être cohérents.

La Figure 4.14 illustre une pyramide tronquée appariée avec son image. Dans la clique étudiée, les nœuds $C_i F_{i=1,2,3}$ sont compatibles deux à deux et vérifient le critère de visibilité. Dès la formation des arcs, un étiquetage est créé pour A_1 et A_2 . Il est clair qu'il y a une incohérence, car les segments $s_{i=1,4}$ se trouvent appariés d'une manière différente avec les arêtes $a_{i=1,4}$ selon chaque arc. Par conséquent, cette clique est globalement incompatible.

Nous allons détailler un quatrième critère pouvant être mis en œuvre, c'est l'invariance géométrique globale.

- Généraliser le critère d'invariance géométrique

Le même critère d'invariance géométrique, appliqué pour la compatibilité mutuelle, peut être généralisé et utilisé pour la compatibilité globale. Soient trois faces, Figure 4.15, formant avec leurs correspondantes 2D la clique courante dans le graphe d'appariement. L'application de l'équation 4.1 sur les trois couples de faces disponibles, et vérifiant les conditions nécessaires pour l'invariant, donne :

$$Inv_2 = \frac{S_{ABD} S_{EFC}}{S_{ABC} S_{EFD}} \frac{S_{KEA} S_{BDC}}{S_{KEC} S_{BDA}} \frac{S_{KAC} S_{DFE}}{S_{KAE} S_{DFC}} = \frac{S_{EFC}}{S_{ABC}} \frac{S_{BDC}}{S_{KEC}} \frac{S_{KAC}}{S_{DFC}} = \frac{S_{E'F'C'}}{S_{A'B'C'}} \frac{S_{B'D'C'}}{S_{K'E'C'}} \frac{S_{K'A'C'}}{S_{D'F'C'}} \quad (4.3)$$

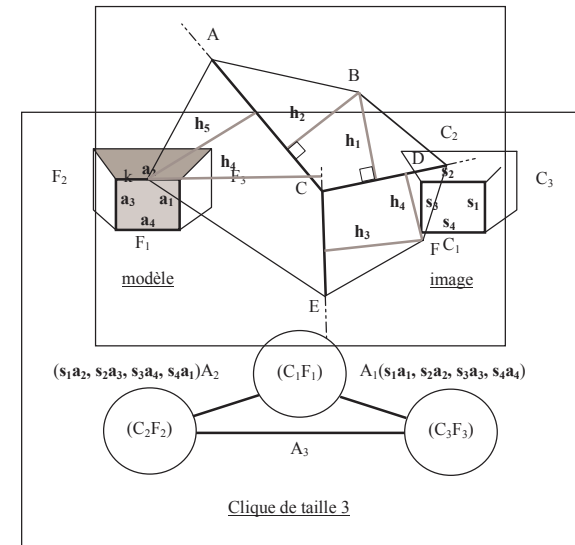


Figure 4.14 : Exemple d'incompatibilité globale pour le critère de connexité

La Figure 4.15 montre aussi qu'on peut calculer cet invariant à l'aide des distances simples mentionnées précédemment :

$$Inv_2 = \frac{h_1}{h_2} \cdot \frac{h_3}{h_4} \cdot \frac{h_5}{h_6} \quad (4.4)$$

qui fait intervenir uniquement les projections des points B, F et K sur les trois lignes commune des trois faces. Cet invariant «global» est issu directement d'une multiplication des invariants mutuels des arcs de la clique concernée, il ne fournit donc pas de nouvelle information. Pour cette raison nous ne l'avons pas utilisé.

En revanche, l'erreur sur l'invariant global représente un cumul d'erreurs sur les invariants entre l'ensemble des faces et leurs projections. Elle ne doit pas dépasser un seuil puisque l'image comporte des sommets de l'objet projetés de la même manière. Par conséquent, l'erreur sur l'invariant global doit rester proche de celle des invariants mutuels, ainsi on peut l'utiliser en tant que critère de cohérence de projection.

Figure 4.15 : L'invariant du triplet de faces est le rapport des distances $h_{i=1..6}$

- Procédure de recherche

Les cliques maximales forment les hypothèses plausibles sur l'appariement. La procédure de recherche est récursive, elle s'applique à tous les nœuds du GApp. Partant d'un nœud N_i , nous cherchons les nœuds N_j reliés avec N_i à l'aide des arcs mutuels déjà construits dans le graphe. N_i se trouvant dans une liste L_k (la clique courante), les nœuds N_j sont testés une par une et ajoutés à cette liste. Un nœud N_j est testé avec la clique courante, les critères mentionnés ci-dessus doivent être vérifiés pour ajouter N_j à la clique. Les passages de recherche d'un sous-graphe complet (une clique) sont mémorisés par un marquage des nœuds concernés, celui-ci évite la redondance. En fait, un nœud appartenant à une clique maximale ne peut pas apparaître une seconde fois dans une autre clique maximale.

A partir de la mise en correspondance segment-arête de chaque clique (l'étiquetage des arcs du GApp), nous pouvons extraire des appariements de points (sommets image / sommets modèle). La *score* d'une clique est la somme des appariements sommet2D-sommet3D de tous ses arcs. Ainsi, on cherche la clique de meilleur score pour appliquer le processus de prédiction-vérification qui nécessite un minimum de score (4 appariements de sommets non coplanaires) pour calculer la première transformation entre le repère-modèle et le repère-image.

Prenons l'exemple du prisme illustré dans les Figures 4.7, 4.8 et 4.13. Le Tableau 4.1 donne le nombre de cliques maximales trouvées à une taille donnée (nombre de nœuds). Parmi un total de 16 hypothèses, une seule (celle du meilleur score) donne la solution unique attendue. Cette hypothèse contient les nœuds N_0, N_3, N_{10} et N_{17} reliés par les arcs $A_1, A_2, A_3, A_{13}, A_{14}$ et A_{35} (voir également Figure 4.16).

		Taille de la clique				Nœuds	
		2	3	4			
Chaînes	4						
Faces	8						
Nœuds	8	10			N ₉ N ₁₆ , N ₁₀ N ₁₅ , etc.		
	16	5			N ₁ N ₁₂ N ₁₇	N ₂ N ₇ N ₁₈	
					N ₅ N ₁₂ N ₁₃	N ₅ N ₁₀ N ₁₅	
					N ₆ N ₇ N ₁₄		
Arcs	46			1	N ₀ N ₃ N ₁₀ N ₁₇		
Hypothèses	total	10	5	1	16		

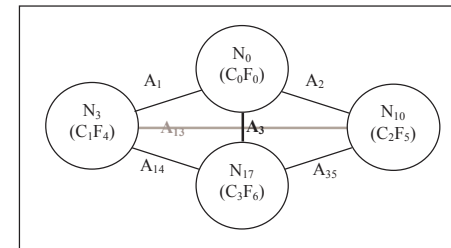
Tableau 4.1 : Résultats de l'identification du prisme

Figure 4.16 : La meilleure hypothèse d'identification du prisme

4.2.4. Complexité de la méthode

La complexité algorithmique de la méthode du GApp se divise en deux parties : hors-ligne et en-ligne. La première concerne l'extraction de graphes d'aspects étudié dans le second chapitre (§2.3.3). La complexité en-ligne est constituée de trois parties séquentielles concernant : la formation des nœuds, la formation des arcs et la recherche des cliques. Comme il n'existe pas de relations formelles entre la complexité et les critères topologique et géométriques, nous allons étudier le pire des cas en insérant seulement le critère logique. Soit N le nombre de modèles 3D, chacun possède un nombre de faces $F_{m=1..N}$ qui sont $f_{m,j}$ ($j=1..F_m$), et C le nombre de chaînes $c_{i=1..C}$ trouvées dans le modèle 2D. Un $GApp_{m=1..N}$ est construit entre le modèle 2D et chacun des modèles 3D.

1. **Formation des nœuds** : les combinaisons chaîne-face donne $C \cdot \sum_{m=1}^N F_m$. Supposons le même nombre de faces F pour les modèles 3D, il vient : $C \cdot N \cdot F$. La complexité de formation de nœuds vaut donc : $O(C \cdot N \cdot F)$.



2. **Formation des arcs** : Dans chaque $GApp_m$, un arc lie (à priori) chaque nœud $c_i f_{m,j}$ avec tous les autres $c_{1..C} f_{m,1..F}$. Le critère logique élimine un arc reliant $c_i f_{m,j}$ avec $c_{i f_{m,1..F}}$ ou avec $c_{1..C} f_{m,j}$. Le nombre d'arcs,

pour chaque $GApp_{m=1..N}$, est alors : $\frac{1}{2} (C F_m) \cdot ((C-1)(F_m-1))$ et pour tous les GApp :

$$\frac{1}{2} C(C-1) \cdot \sum_{m=1}^N F_m (F_m-1)$$

Le facteur $\frac{1}{2}$ vient du fait que les arcs sont non-orientés. En considérant le même nombre de faces F pour les N modèles, la complexité devient : $O(C^2 \cdot N \cdot F^2)$

3. **Recherche des cliques maximales** : La recherche, à partir d'un nœud, se fait en explorant tous les arcs le reliant. Dans le pire des cas une clique maximale se réduit à un arc simple (en partant du nœud c'est le premier arc qui réussit et les autres échouent). La complexité recherche en chaque nœud est proportionnelle

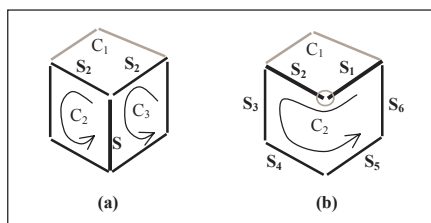
aux nombre d'arcs le reliant (qui à son tour égale le nombre des nœuds -1). En remarquant que chaque nœud déjà testé n'est plus parcouru, le nombre de parcours devient :

$$n_m + (n_m - 2) + (n_m - 4) + \dots + 2 \equiv n_m ((n_m / 2) - 1) / 2 \text{ où } n_m \text{ est le nombre de nœuds dans un}$$

GApp. La complexité devient : $O(n_m^2) = O(C^2 \times N \times F^2)$

La complexité algorithmique totale en-ligne s'obtient : $O(C^2 \times N \times F^2)$. Le seul moyen de la diminuer est de réduire le nombre de chaînes extraites par traitement en améliorant celui-ci. L'application de différentes contraintes en pratique réduit considérablement cette complexité.

4.2.5. Problèmes de l'approche graphe basé sur les chaînes



L'algorithme de recherche de chaînes donne de très bons résultats si tous les segments de l'image ont été détectés. Or, il arrive souvent qu'au moins un segment ne soit pas trouvé. Dans le meilleur des cas, nous trouverons une chaîne ouverte à la place d'une chaîne fermée. Mais il se peut aussi que deux chaînes se trouvent fusionnées, les rendant inutilisables. C'est ce qu'entraîne la perte du segment central S dans l'exemple de la Figure 4.17 La chaîne résultante C2 dans (b) ne correspond à aucune face dans la réalité. Ceci nous amène à penser que la méthode d'appariement n'est pas robuste et n'est pas adaptée aux images réelles, puisque le manque d'un seul segment entraîne la perte de deux chaînes.

Figure 4.17 : Fusion de 2 chaînes par manque d'un segment

Deux solutions existent pour résoudre ce problème :

- la première est de pousser l'analyse du chaînage pour le rendre plus cohérent,
- la seconde est d'abandonner les chaînes et travailler au niveau moins élevé (segments et sommets), car même dans le cas de la perte d'un segment, il est possible que tous les sommets soient conservés (c'est le cas de la Figure 4.17). Cette solution est le sujet de la seconde partie de ce chapitre.

Une incohérence de connexion peut arriver lorsque deux chaînes (C1 et C2 dans la Figure 4.17-b) comportent deux segments non-colinéaires ou plus en commun (S1 et S2). Ce problème peut être résolu par une procédure de *découpage* de la fausse chaîne C1 de (b) en sous-chaînes vérifiant la cohérence. Pour ce faire, il faut établir un degré de confiance aux chaînes. Ce degré de confiance sera plus importante pour les chaînes fermée et ayant une forte connectivité avec d'autres chaînes. Le découpage doit se faire sur la chaîne ayant le plus faible degré de confiance, dans le but d'avoir un seul segment partagé par sous-chaîne (Figure 4.17).

Le découpage d'une chaîne en sous-chaînes comporte un problème concernant le critère d'arrêt. Par exemple, le découpage de C2 dans la Figure 4.17-b donne plusieurs possibilités telles que les ensembles de sous-chaînes (S2S3S4, S1S6S5), (S2S3, S1S6, S4S5), (S2S3S4S5, S1S6), etc. Ce découpage peut rapidement aboutir à un grand nombre de possibilités et de chaînes ouvertes ayant chacune un petit nombre de segments (il revient dans le pire des cas à effectuer l'appariement au niveau de segments), une étape d'*union* est alors envisagée.

L'union prendra en charge de fusionner une petite chaîne faisant partie d'une autre plus grande. L'union peut se voir également comme un appariement entre une face et plusieurs sous-chaînes, d'où il faut étendre le GApp pour inclure ce type d'hypothèses. Cette analyse de chaînes nécessite une complication importante du système et dans le cas d'un mauvais traitement de l'image elle devient très lourde. Pour ces raisons, nous avons opté directement pour une méthode qui travaille sur les sommets (la seconde solution mentionnée ci-dessus).

En résumé : la méthode du GApp représente l'avantage d'utilisation du graphe d'aspects et des contraintes topologiques et géométriques de compatibilité pour réduire efficacement le nombre d'hypothèses de mise en correspondance pour les objets polyédriques. En revanche, il est difficile de la mettre en œuvre pour l'identification des objets quelconques, elle est également tributaire de la qualité de l'image caméra. Si celle-ci est dégradée, le résultat du traitement de l'image s'avère insuffisant - difficulté d'extraction des chaînes de segments - pour appliquer cette méthode. En outre, la complexité de cette méthode dépend proportionnellement du nombre d'objets à identifier et du carré du nombre de chaînes dans l'image. La méthode du hachage géométrique basée sur les sommets des objets (généralisable facilement aux segments) évite le problème du chaînage. Elle permet également de chercher des appariement directement sur l'ensemble des modèles 3D de la base.

4.3. Méthode du Hachage Géométrique (HG)

4.3.1. Introduction

L'approche du *hachage géométrique* (HG) fut introduit par Lamdan et Wolfson [LAMDAN 1988], [WOLFSON 1992]. Elle a été brièvement exposée au premier chapitre (§1.1.1.2), différents systèmes l'utilisant ont été également étudiés (cf. Tableau 1.1). Cette approche consiste en la création hors ligne d'une *table de hachage* (TabH), appelée aussi *table*, et un appariement en ligne. Une idée essentielle du HG est cette division de calculs en deux parties, dont la plupart sont concentrés dans la première (hors ligne) pour alléger la seconde.

La table de hachage contient les différents aspects des objets de la BD3D, exprimés dans des bases différentes (une base peut être représentée par 2, 3 points). Cette expression est fondée sur des invariants ou des quasi invariants géométriques. L'utilisation des caractéristiques locales (points ou arêtes) permet de trouver un appariement même avec une occlusion partielle de l'objet (absence d'une partie des primitives).

Pour chaque entrée dans la table, il existe une liste de modèles susceptibles de satisfaire la reconnaissance. L'appariement en ligne est obtenue en confrontant les points 2D, exprimés à leur tour dans différentes bases, avec le contenu de la table de hachage. Toutes les hypothèses sur l'appariement sont utilisées dans une étape de vérification.

Nous allons présenter les différentes parties de cette approche en commençant par la notion d'invariants ou de quasi invariants.

4.3.2. Propriétés d'invariances

Soit un objet 2D, représenté par ses caractéristiques locales (points d'intérêt), subissant une transformation affine dans son plan (translation, rotation et mise en échelle). Le but est de trouver une propriété, à partir de ces points d'intérêt, qui soit invariante par transformation. Cette propriété, si elle existe, permet d'identifier l'objet dans sa nouvelle situation. En plus, si cette propriété peut être déduite à partir d'un sous-ensemble des points d'intérêt de l'objet, celui-ci pourra être reconnu même en cas d'occlusion partielle (en disposant d'une partie de l'objet).

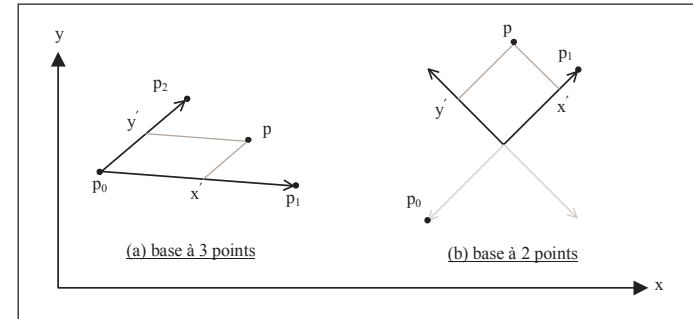
Dans cette optique nous cherchons des propriétés géométriques à une série de points demeurant invariante après une ou plusieurs de ces transformations. Considérons P_0 , P_1 et P_2 trois points non colinéaires du plan et considérons les deux vecteurs (P_0P_1) et (P_0P_2) formant une base du plan (Figure 4.18-a). Un point P se projettera dans cette base avec pour coordonnées x' et y' :

$$x' = a(p_1 - p_0) \quad \text{et} \quad y' = b(p_2 - p_0)$$

où : a , b sont les coordonnées relatives du point p dans la base $(P_0P_1P_2)$

L'observation à en tirer est que si tout le système subit une transformation affine dans le plan, les coordonnées du point P resteront a et b dans la base. Cette propriété s'appelle une *invariance* et est d'une utilité majeure dans la technique du hachage géométrique en particulier lors de l'appariement. De même en choisissant deux points pour former la base (voir Figure 4.18-b), cette propriété s'avérera encore exacte. Notons qu'ici l'origine a été choisie au milieu de P_0P_1 et que le second axe est perpendiculaire à P_0P_1 .

L'application de cet invariant pour l'appariement d'objets 3D (points d'intérêt 3D) à partir d'une image d'intensité (points d'intérêt 2D) est possible sous certaines hypothèses. Soit un objet 3d, projetés sur un plan image, les points d'intérêt de l'objet sont représentés en 2D dans une base de points. L'objet subissant une



transformation dans l'espace est re-projeté sur le plan image. La propriété étudiée précédemment ne garde plus son invariance. L'observation à faire est que la variation des coordonnées des points reste quasi indépendante (dans un petit intervalle) vis à vis de quelques degrés de liberté ; tels que le déplacement et la rotation autour de l'axe caméra-base. Cette propriété est alors *quasi invariante* sous réserve que la distance caméra-objet est assez grande et que la transformation que subit l'objet reste petite.

Figure 4.18 : Choix d'une base à trois points (a), ou à deux points (b).

4.3.3. Description de la méthode

Les caractéristiques locales utilisées, dans notre implémentation, pour reconnaître des polyèdres sont les sommets. La méthode du HG, illustrée dans la Figure 4.19, consiste en deux parties :

1. la première partie appelée *modélisation* où chaque polyèdre est décrit par ses sommets et par des relations géométriques les liant les uns aux autres dans une structure appelée *table de hachage* (TabH). Cette partie exécutée hors-ligne est décrite plus en détails au §4.3.3.1. Elle est constituée d'une extraction d'aspects de modèles de la BD3D, d'une représentation de ceux-ci dans des bases différentes et d'une construction de la TabH.
2. la seconde partie consiste à réaliser les relations qui lient les sommets d'un polyèdre avec une série de points extraits de son image. Cette partie exécutée en-ligne à l'aide de la table de hachage est décrite au §4.3.3.2. Elle est constituée d'une représentation des points de l'image dans une base, d'un vote dans la table de hachage et de l'appariement proprement dit selon les candidats (hypothèses) résultant du vote.

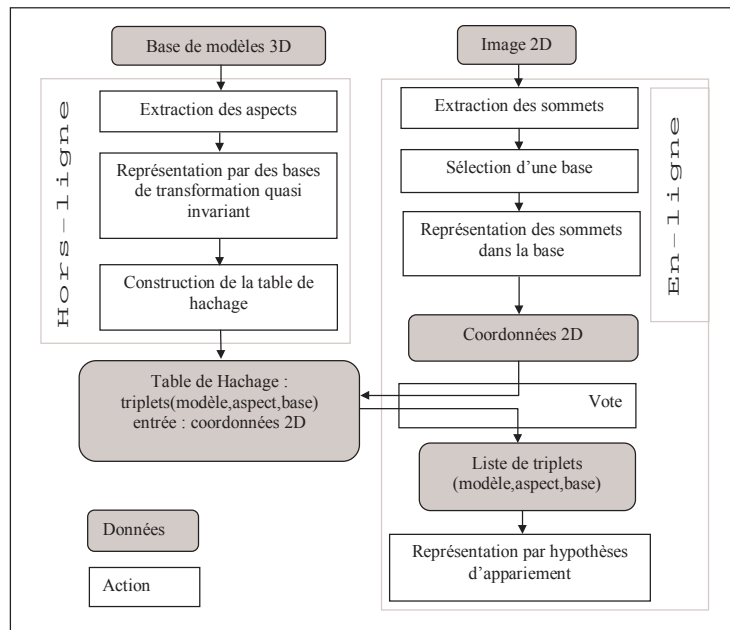


Figure 4.19 : Aperçu de la méthode du hachage géométrique

4.3.3.1. Modélisation

L'objectif de cette partie est de représenter les informations de la BD3D, nécessaires pour un appariement «rapide», dans une structure (TabH : Table de Hachage) accessible à partir des points clés extraits de la scène. Ces informations sont des aspects (projections de sommets visibles) représentatifs de chaque modèle 3D, chacun des aspects est représenté à son tour dans des bases différentes. Chaque sommet, caractérisé par un triplet (modèle, aspect, base)⁴ auquel il appartient, est rentré dans la TabH selon ses coordonnées exprimées dans sa base. Un élément (*case*) de la TabH est un point exprimé dans un repère absolu et modélisé par une liste de triplets (m,a,b) auxquels il appartient. Nous conserverons dans une autre structure, pour chaque modèle, les informations sur tous ses aspects et la trace des bases dans lesquelles les aspects ont été exprimés.

- Extraction des aspects

Le hachage géométrique en 2D/3D revient à étudier la similitude entre une image d'un objet 3D et un ensemble de projections de celui-ci (images 2D ou aspects). Un graphe d'aspects est alors nécessaire pour appliquer le HG. Pour réussir la reconnaissance de l'objet à partir de cet ensemble limité de vues, la propriété de quasi invariance

⁴ Un triplet (modèle, aspect, base) est noté désormais (m,a,b).

doit être conservée (§4.3.2). Les projections (aspects) doivent être prises avec un petit changement sur les coordonnées géométriques. C'est pour cette raison que nous utilisons la méthode d'extraction de graphes d'aspects par projection, étudiée dans le second chapitre (méthode exhaustive §2.3.3.1). Les projections se font alors, à l'aide d'une sphère centrée sur l'objet, à partir de points de vue peu espacés sur la surface de celle-ci.

Gavrila [GAVRILA 1992] propose une méthode dynamique de découpage non-uniforme de la surface par des triangles, cette méthode rend le HG plus robuste. La surface, au départ, est uniformément découpée par des triangles. Ensuite, la reconnaissance est testée sur chaque triangle ; l'erreur entre la projection au bords du celui-ci et celle au milieu doit être inférieure à un seuil E_e . E_e représente une borne supérieure de l'erreur sur la reconnaissance ; erreur des points d'entrée en dessous de laquelle la reconnaissance de l'objet est assurée. Si le test n'est pas satisfait, le triangle est découpé en quatre auxquels le processus s'applique récursivement.

- Représentation des aspects par des bases

Afin d'utiliser la propriété d'invariance vue précédemment, nous projetons chaque aspect dans différentes bases dont les vecteurs d'origine sont définis à partir des points de l'aspect. Ces vecteurs peuvent être définis de deux façons :

- à l'aide de trois points
- à l'aide de deux points

Dans le premier cas (cf. Figure 4.18-a) trois points non colinéaires de l'espace sont choisis de façon à respecter deux critères :

- l'angle formé par les deux vecteurs doit être contenu dans une certaine plage de valeurs afin que le modèle ne soit pas trop déformé. Nous fixons donc arbitrairement cet angle dans l'intervalle $[45^\circ, 135^\circ]$
- les deux vecteurs doivent avoir une longueur supérieure à un certain seuil. La raison vient du fait des erreurs dues à l'extraction des sommets par le traitement de l'image d'entrée. L'erreur sur la projection de ces sommets dans une base à petits vecteurs augmente sensiblement.

Ces deux critères reviennent en fait à maximiser l'aire de la surface formée par les deux vecteurs de base.

Dans le second cas (cf. Figure 4.18-b) deux points de l'aspect (formant un segment) sont choisis. L'origine de la base est positionnée au milieu de ce segment, le premier vecteur de base est celui reliant l'origine au second point. Le second vecteur de base est choisi directement perpendiculaire au premier dans le sens trigonométrique et de la même longueur. Pour le choix de cette base on ne conserve qu'un critère : la distance entre les deux points choisis doit dépasser un certain seuil fixé afin que les vecteurs de bases ne soient pas trop courts.

Ces deux types de bases ont été utilisés afin d'implémenter le hachage géométrique et seront mis en concurrence dans la partie résultat (§5.2.1.2). Un certain nombre de bases sont donc choisies et introduites dans une structure prévue à cet effet. La représentation de chaque aspect par plusieurs bases est nécessaire pour accélérer l'appariement en-ligne. Car, la chance de trouver, dès la première itération, une base dans l'image d'entrée correspondant à une base mémorisée augmente. En revanche, cette accélération se paie par une croissance de la taille de la TabH.

- Construction de la Table de Hachage (TabH)

La table de hachage se représente comme un tableau à deux entrées de dimension fixe. Chaque case de ce tableau constitue une liste dont les éléments sont des triplets (m,a,b).

Pour chaque modèle de la BD3D, les aspects qui le composent sont numérotés par latitude croissante puis par longitude croissante (point de vue sur la sphère de Gauss §2.3.3.1). Considérons un aspect **a** du modèle **m**, cet aspect possède un nombre défini de bases numérotées par leur ordre d'apparition. Choisissons la base d'indice **b**, un point P(x,y) de l'aspect se projète dans **b** suivant les coordonnées (x',y'). Ces deux coordonnées sont ensuite arrondies aux entiers qui leurs sont le plus proche (x'',y''). Dans la case de la TabH définie par les entrées x'' et y'', le triplet (**m,a,b**) est ajouté à la liste présente.

Ce processus est réitéré pour chaque point, chaque base et chaque aspect associé au modèle, celui-ci est alors entièrement introduit dans la TabH. La table de hachage constitue ainsi une base de donnée pour la reconnaissance d'objets. Différentes tables de hachage contenant un ou plusieurs modèles sont utilisées afin de reconnaître des ensembles d'objets spécifiques. Certaines parmi elles seront chargées à chaque utilisation pour servir la reconnaissance d'un groupe d'objets.

4.3.3.2. Appariement

Nous disposons afin de réaliser l'appariement d'une série de points d'une scène. L'extraction de ces points est réalisée par le traitement d'images (cf. chapitre 3, §3.6), ils sont pour la plupart des sommets des objets à reconnaître mais peuvent aussi être des points représentant des bruits. Le but de cette phase est de :

- fournir à partir de ces données et d'une table de hachage des triplets (m,a,b) susceptible de se superposer à la scène, c'est le rôle du vote,
- réaliser le maximum de correspondances entre points du modèle et points de la scène à partir de ces triplets

- Vote

Nous définissons pour les points $P_{i=1..n}$ extraits de l'image, une base **B** vérifiant les critères du §4.3.3.1. Le type de base choisi est le même que celui utilisé pour construire la table de hachage ; base à deux ou à trois points (k=2 ou 3). Les points $P_{i=1..n}$ sont projetés dans **B**, leurs coordonnées sont arrondies de la même façon qu'au §4.3.3.1. Chaque point va ainsi fournir une entrée dans la table de hachage (x_i,y_i) pointant vers une case de celui-ci. Dans cette case se trouve une liste de triplets (m,a,b), dont chacun possède un compteur. Le *vote* consiste à incrémenter les compteurs de ces triplets. En fixant un seuil de vote minimal à dépasser, il ressort une série triée de triplets (m,a,b) susceptible de satisfaire l'appariement modèle-scène.

- Mise en correspondance

Les triplets candidats pour satisfaire l'appariement ne comportent pas une mise en correspondance directe entre les points de l'image et les sommets du modèle. Car, les informations contenu dans un triplet (**m,a,b**) sont :

indice du modèle **m**, longitude et latitude sur la sphère de Gauss exprimant l'aspect **a** les indices des sommets formant la base **b**. Une solution peut se faire en ajoutant à chaque triplet :

- hors-ligne, les indices des sommets 3D ayant participé à son insertion dans la TabH
- en-ligne, les indices des points 2D ayant voté pour lui

Cette solution donne directement la mise en correspondance recherchée, mais nécessite une forte augmentation de la taille de la TabH. De plus, son gain en rapidité en-ligne n'est pas sûr car à chaque vote il faut ajouter plusieurs indices à chaque triplet.

Nous avons adopté une solution plus performante avec le même résultat : à partir d'un triplet (**m,a,b**) candidat à l'appariement nous re-projetons :

- d'une part les sommets du modèle **m** suivant la direction définie par l'aspect **a** (longitude et latitude sur la sphère de Gauss), les coordonnées sont exprimées dans la base **b** et arrondies comme dans l'étape de modélisation.
- d'autre part les points de la scène dans la base **B** en utilisant la même technique que précédemment.

Le nombre de correspondances dans un triplet ne vaut pas forcément le nombre de votes. En fait, un point 2d peut voter pour plusieurs sommets 3D et vice versa. Cela est dû à plusieurs erreurs : arrondissement des coordonnées, pas de hachage, bruit sur les points de l'image ou une simple coïncidence de sommets 3d sur le rayon visuel.

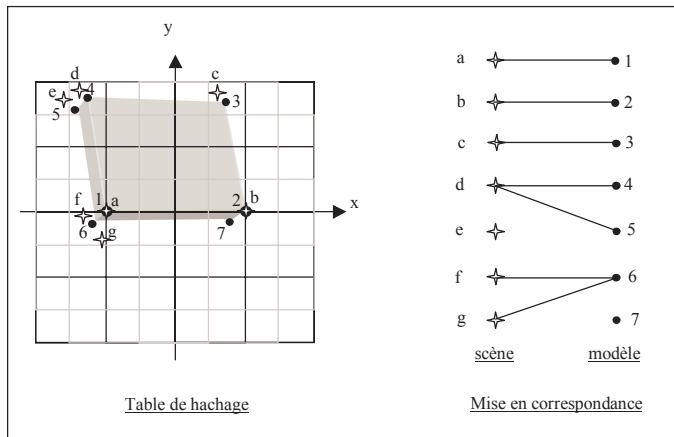
La Figure 4.20 montre cette ambiguïté de mise en correspondance. Les sommets 1 à 7 exprimés dans la base **b** (sommets 1 et 2) représentent un aspect **a** d'un parallélépipède **m**. Les coordonnées de ces sommets dans la base **b** sont arrondies et rentrées dans une table de hachage simple. Les points extraits de la scène a à g exprimés dans la base **B** (points a et b) votent pour le triplet (**m,a,b**). Les appariements (a,1) et (b,2) sont évidents car ce sont les bases choisies, le point d vote pour les deux sommets 4 et 5 présents dans la même case. f, mais aussi g un point de «bruit», votent pour 6. Il est clair que le nombre de votes vaut 6 (la base étant 2 votes), mais le nombre d'appariements évidents est seulement 3.

L'ambiguïté sur la mise en correspondance rend le nombre de votes moins significatif pour évaluer la plausibilité des hypothèses. Le HG est considérée alors comme une méthode d'*indexage* (recherche d'un ensemble comparable à une entrée à partir d'une grande base de données) plutôt que d'appariement. De ce fait, la méthode du HG ne doit pas être considérée comme une méthode conclusive et doit être suivie par une procédure plus précise. Pour chercher le maximum de correspondances possibles il faut appliquer la procédure itérative de prédiction–vérification détaillée dans §4.4.

Figure 4.20 : Ambiguïté sur la mise en correspondance 2D/3D

4.3.4. Quantification de notre implantation

Dans cette partie deux caractéristiques de notre implantation du HG sont étudiées : l'encombrement mémoire et la complexité algorithmique. Elles sont quantifiées en fonction des choix adoptés : structures de données et seuils



tels que le pas de hachage, le type de bases, le nombre d'aspects et de bases, etc.

• Encombrement mémoire

La procédure principale du HG est composée d'accès et de votes dans la TabH (données). Si la taille de celle-ci, résidant en mémoire vive, est trop importante elle peut générer une perte de temps à l'exécution. Les données stockées sont de deux types :

- les tableaux de type triplet (m,a,b), notés *TabT*,
- la table de hachage, un tableau à deux dimensions de listes de pointeurs vers les triplets

Nous allons examiner ces deux types en notant : *word* = 2 octets e s (entier court), *double* = 8 octets, *pointeur* = 4 octets et l'opérateur *t()* signifie «taille de».

Soit *m* le nombre de modèles 3D à reconnaître, *a* le nombre d'aspects, *b* le nombre de bases et *k* le nombre de points formant une base (*k*=2 ou 3). Un triplet est décrit par :

- un modèle 3D représenté par son numéro (word),
- un aspect de celui-ci représenté par 2 angles et le rayon de la sphère (3 doubles),
- une base représente *k* indices de sommets (*k* words)

Il vient : $t(TabT) = m \cdot t(word + a \cdot t(3doubles + b \cdot k(word)))$ octets

D'où :

$$t(TabT) = m \cdot (2 + a \cdot (24 + 2 \cdot b \cdot k)) \text{ octets} \quad (4.5)$$

Le nombre de modèles peut être diminué en construisant plusieurs TabH chacune spécialisée pour reconnaître un sous-ensemble d'objets (cf. tailles des TabH, §5.2.1.2). Dans l'application à la téléopération, l'opérateur choisit en-ligne la TabH minimale regroupant les objets à reconnaître.

L'expérience montre malheureusement qu'on ne peut jouer sur la valeur *a*, car une distance assez petite doit être conservé sur la sphère entre deux aspects (§4.3.3.1). Nous avons donc choisi une valeur de *a* de $36 \times 18 = 648$ représentant les aspects obtenus en parcourant les latitude et longitude de dix degrés en dix degrés. La valeur de *k* joue un rôle très important dans l'équation précédente, nous avons choisi *k*=2. De plus, le type de bases à 2 points est meilleur que celui de 3 vis à vis d'autres critères (cf. l'évaluation au §5.2.1.2).

Reste à déterminer *b*, un modèle à dix sommets possède un nombre total de bases allant de 72 (pour des bases à deux points) jusqu'à 504 (pour des bases à trois points). Le nombre de bases *b* choisi hors-ligne peut être diminué, mais au détriment de la rapidité de reconnaissance en-ligne. Prenons l'extrême : *b*=1, il faut alors dans le pire des cas choisir toutes les bases possibles en-ligne pour garantir la reconnaissance. Ce nombre peut être limité en choisissant, encore une, fois *k*=2, et en imposant les critères mentionnés au §4.3.3.1 pour la sélection des bases. Un nombre total de *b*=25 bases nous a paru raisonnable.

Regardons de près la taille d'une TabH : celle-ci comporte *n* cases dont chacune un pointeur vers une liste de pointeurs indiquant les triplets possibles. La taille totale vaut donc :

$$t(TabH) = \sum_{i=1}^n (t(\text{pointeur}) + nb_triplets_i \cdot t(\text{pointeur})) = t(\text{pointeur}) \cdot \left(n + \sum_{i=1}^n nb_triplets_i \right)$$

Le nombre de triplets total $\sum_{i=1}^n nb_triplets_i$ dans la TabH est le nombre de sommets visibles des modèles 3D ayant été projetés, lors de la construction de la table, selon tous les aspects et bases :

$$\sum_{i=1}^n nb_triplets_i = \sum_{j=1}^m a \cdot b \cdot nb_sommets_j$$

avec *nb_sommets_j* indique le nombre moyen des sommets visibles, dans tous les aspects du modèle *j*, moins *k*, donc :

$$t(\text{TabH}) = 4 \cdot \left(n + a \cdot b \cdot \sum_{j=1}^m \text{nb_sommets}_j \right) \text{octets} \quad (4.6)$$

Le nombre de cases n dépend de la longueur de la TabH et du pas de hachage : $n = (l/pas)^2$.

Une longueur trop importante de la TabH entraînerait une augmentation nécessaire du nombre d'entrées dans la table de hachage. De même une table de hachage trop petite conduirait à un trop grand nombre de solutions lors de la reconnaissance. Prenons un pas=1 et fixons la longueur l . La longueur de la TabH vaut deux fois la longueur du vecteur de base, car celui-ci se situe au milieu de la TabH et forme un repère absolu pour toutes les projections des sommets (Figure 4.20). Nous devons fixer une longueur minimale pour ce vecteur de base pour les raisons discutées dans §4.3.3.1 (critères de sélection des bases), une valeur de 50 est choisie. Le nombre de cases trouvé $n=10000$ est une valeur raisonnable, car $t(\text{TabH})$ dépend plutôt du second terme.

La somme des deux équations 4.5 et 4.6 donne la taille totale en mémoire :

$$\text{taille_mémoire} = m \cdot (2 + a \cdot (24 + 2 \cdot b \cdot k)) + 4 \cdot \left(n + a \cdot b \cdot \sum_{j=1}^m \text{nb_sommets}_j \right) \text{octets} \quad (4.7)$$

Remplaçons les valeurs fixes de a , b , k et n , il arrive :

$$\text{taille_mémoire} = 40000 + 80354 \cdot m + 64800 \cdot \sum_{j=1}^m \text{nb_sommets}_j \text{octets} \quad (4.8)$$

L'**encombrement mémoire** est alors linéairement dépendant du **nombre de modèles** 3D traités et de leur **complexité moyenne** (nb_sommets). Un tétraèdre simple à 5 points possède 18 aspects topologiquement différents (4 aspects à 3 sommets visibles, 5 à 4 et 9 à 5). La moyenne approximative du nb_sommets vaut 2.3, nous trouvons :

$$\text{taille} \approx 80354 + 40000 + 64800 \times 2.3 \approx 268 \text{ k octets} \quad (4.9)$$

La taille des tableaux de triplets valant 80354 octets est indépendante des modèles, elle reste très petite par rapport à la taille totale.

• Complexité algorithmique

Deux complexités existent, la première concerne la partie hors-ligne, la seconde plus importante est celle en-ligne. Prenons les même notations que ci-dessus :

1. complexité hors-ligne : dans le pire des cas, la sélection de toutes les bases possibles pour un modèle j et un aspect i donné est : $(\text{nb_s}_{ji})^k$ où nb_s est le nombre de tous les sommets visibles. Pour chaque sélection il faut accéder à la TabH $(\text{nb_s}_{ji} - k)$ fois. La

complexité est donc : $\sum_{j=1}^m \sum_{i=1}^{a_j} (op_proj + op_rep \cdot (\text{nb_s}_{ji})^k \cdot op_ins \cdot (\text{nb_s}_{ji} - k))$

- op_proj est l'opération de projection d'un modèle pour fournir un aspect
- op_rep est l'opération de représentation d'un aspect dans une base
- op_ins est l'opération d'insertion d'un triplet dans l'entrée correspondant de la TabH

ces opérations dépendent de la complexité du modèle j (nb_s_j). Il est clair que les calculs dépendent fortement de k (nous avons choisi $k=2$ minimal). En considérant la complexité des modèles constante et en remplaçant a et k , la complexité hors-ligne devient :

$$O(m \times \text{nb_s}^3) \text{ qui vaut } O(m) \text{ lorsque } m \text{ est très grand}$$

2. complexité en-ligne : Soit p le nombre de points dans l'image d'entrée, avec la même analyse que précédemment, la complexité vaut :

$$op_rep \cdot p^k \cdot op_ins \cdot (p - k)$$

L'opération d'insertion veut dire un vote dans la case correspondante au point courant et dépendante de la taille des liste de triplets dans la TabH. La complexité dépend donc en général du nombre de triplets nb_trip dans la TabH. Nous trouvons dans le pire des cas (sélection de toutes les bases possibles) : $O(p^3 \times \text{nb_trip})$ nb_trip étant dépendant, comme vu précédemment, de m , de a et de b , nous espérons reconnaître un objet dès la première sélection d'une base ($O(\text{nb_trip})$). Ceci est possible lorsqu'on mémorise toutes les bases possibles hors-ligne et lorsque l'objet existe effectivement dans l'image ! Sinon, nous sommes obligé d'explorer toutes les bases en-ligne.

4.4. Mise en correspondance entre le modèle 2D et les modèles 3D

Les différents constituants du système de reconnaissance de polyèdres sont représentés dans la Figure 4.21. Ce système est constitué des éléments suivants :

- une saisie et traitement hors ligne des modèles 3D : ce module a été étudié au chapitre 2 et contient la modélisation des polyèdres et la construction des graphes d'aspects. La construction des tables de hachage a été évoqué dans ce chapitre (§4.3.3.1)
- un traitement de la scène (image des objets à reconnaître) à l'aide duquel nous obtenons un modèle 2D, ceci a fait l'objet du chapitre 3,
- génération des hypothèses valides par mise en correspondance entre des primitives de la scène et primitives disponibles dans les modèles 3D,
- propagation de chaque hypothèse par une procédure itérative de prédiction-vérification pour la localisation de l'objet et enfin,
- la reconstruction des objets reconnus sur l'image du monde réel.

Dans la méthode du graphe d'appariement (§4.2), la mise en correspondance du modèle 2D est effectué successivement avec tous les modèles 3D à reconnaître. Nous obtenons autant de GApp que de modèles 3D. Dans la méthode du hachage géométrique (§4.3), l'appariement se fait directement avec tous les modèles 3D stockés dans la table de hachage. La sélection de l'une des deux méthodes se fait par l'opérateur selon le résultat

du traitement de l'image et le nombre de modèles à reconnaître. Les hypothèses finales générées par la méthode utilisée sont ordonnées selon leur score et soumises à l'étape de vérification.

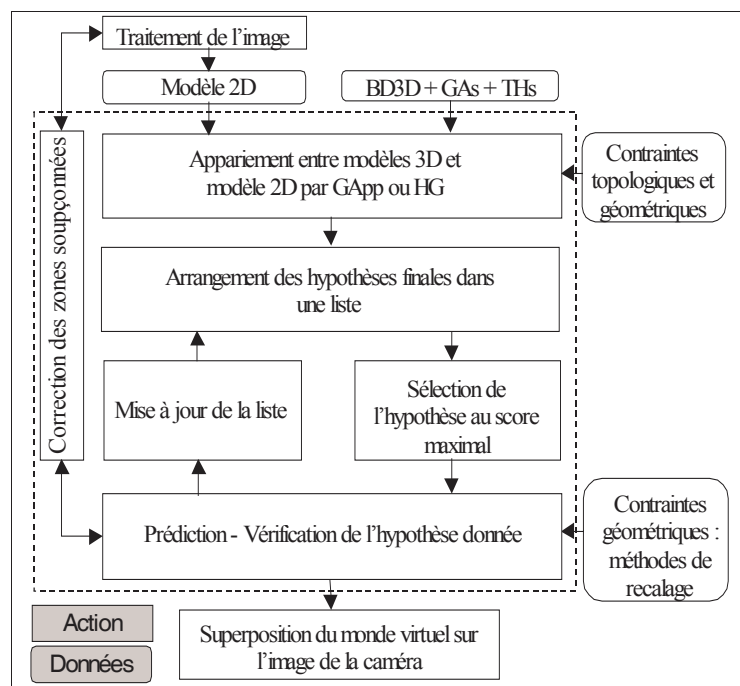


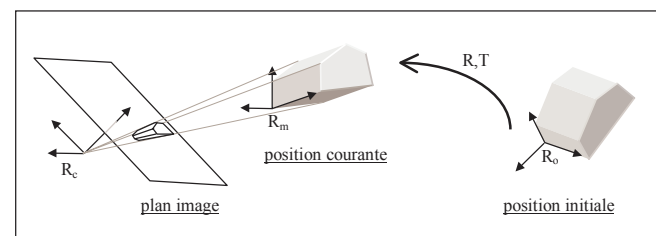
Figure 4.21 : Mise en correspondance entre modèle 2D et modèles 3D.

Une méthode de prédiction-vérification classique est appliquée, aux hypothèses une par une. Si une hypothèse réussit, ses éléments sont retirés du modèle 2D de l'image (mise à jour des listes de primitives). Toute hypothèse restante dans la liste est retirée si elle dépend des primitives déjà utilisées. Nous continuons cette opération jusqu'à l'exploration de tout le modèle 2D ou de toutes les hypothèses. Il faut remarquer que dans le cas du HG, la première hypothèse réussie signifie que la base de points, sélectionnée dans l'image, appartient à l'instance de l'objet désigné par cette hypothèse. Les hypothèses restantes dans la liste n'ont alors aucune signification, il faut sélectionner une base différente, n'appartenant pas à l'instance trouvé, et relancer la procédure du HG.

La transformation rigide, entre le modèle 3D et le repère de l'affichage, est calculée dans la phase de vérification, pour chaque hypothèse testée à l'aide des méthodes de recalages. Une correction du résultat du traitement de l'image est faite localement, à l'aide des informations parvenant de l'hypothèse en cours, pour

renforcer la vérification de celle-ci et améliorer son recalage. Enfin, tous les modèles reconnus sont superposés sur l'image.

Nous avons développé trois méthodes de recalage et amélioré leur performance par une procédure mixte. Les paragraphes suivants font l'objet de ce développement. De plus, l'étape de correction mentionnée ci-dessus est illustrée à la fin de ce sous-chapitre.



4.4.1. Méthodes de recalage

Dans le cas d'une assistance à la perception, un monde virtuel est superposé au monde réel grâce à la base de données géométriques 3D des objets de l'environnement. Le changement de la configuration d'un objet dans l'environnement nécessite une mise à jour de sa situation dans l'espace. La détermination des transformations géométriques qui permettent d'obtenir la nouvelle situation de l'objet s'appelle *recalage* ou *reconstruction 3D*. Le recalage est un élément essentiel pour la vérification des hypothèses de l'appariement (entre image et BD3D) et pour la superposition du monde virtuel à l'image. Deux types de recalages qui dépendent de l'ampleur du mouvement sont distingués : recalage à grand mouvement et recalage à petit mouvement. Nous présentons le formalisme mathématique du recalage ainsi que les méthodes utilisées pour le résoudre. L'évaluation de ces méthodes est donnée dans le chapitre 5.

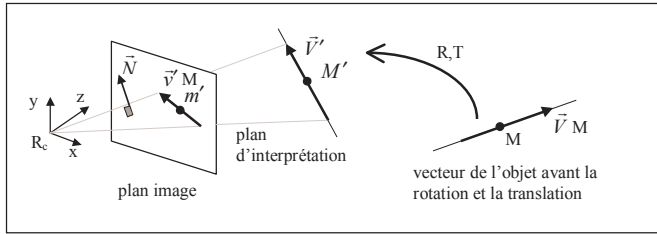
4.4.1.1. Le problème

L'objectif de la reconstruction polyédrique 3D est de retrouver la rotation et la translation subies par un objet (localisation) dans l'espace. La localisation fait intervenir trois repères principaux (Figure 4.22) :

- ① (R_c) repère lié à la caméra.
- ② (R_m) repère lié à l'objet (modèle) représentant la position courante.
- ③ (R_o) repère de travail, représente (R_m) à l'instant initial.

La situation initiale de l'objet est exprimée dans (R_o). Il s'agit de calculer R et T , c'est à dire la situation de (R_m)/(R_o) et donc la nouvelle situation de l'objet.

Figure 4.22 : La rotation et la translation à déterminer



Toutes les méthodes de localisation tridimensionnelle reposent sur des bases mathématiques communes exploitant des indices visuels 2D provenant de la caméra. La caméra est considérée déjà calibrée, la correspondance entre les points dans l'espace et leurs projections sur le plan rétinien est donc connue. Les repères (Rc) et (Ro) sont connus l'un par rapport à l'autre (par la calibration). On suppose par la suite pour simplifier les expressions, mais sans rien enlever à la généralité du problème, que (Ro) est confondu avec (Rc).

Pour tout point M et vecteur \vec{V} de l'objet exprimés dans (Rm), leurs expressions respectives M/Rc et \vec{V}/Rc sont :

$$M/Rc = T + RM/Rm \quad (4.10)$$

$$\vec{V}/Rc = R\vec{V}/Rm \quad (4.11)$$

Le vecteur 3D, \vec{V}' (Figure 4.23) donne le vecteur v' par projection perspective de centre C sur le plan rétinien.

A l'aide des coordonnées 2D, on définit le plan d'interprétation qui passe par le vecteur v' et le centre optique C.

Ce plan contient également le vecteur affine \vec{V}' de l'espace. \vec{N} étant la normale à ce plan, nous obtenons :

Figure 4.23 : Transformation d'un vecteur dans un plan d'interprétation

$$\vec{N} \cdot \vec{V}' = 0 \quad (4.12)$$

$$\vec{N} \cdot \vec{CM}' = 0 \quad (4.13)$$

Les vecteurs présents dans ces deux relations sont exprimés dans un même repère (Rc). En tenant compte des équations (4.10) et (4.11), on obtient :

$$\vec{N} \cdot (R\vec{V}) = 0 \quad (4.14)$$

$$\vec{N} \cdot (T + RM) = 0 \quad (4.15)$$

\vec{V} est un vecteur reliant deux points de l'objet et exprimé dans (Rm) (connu)

M est un point de l'objet exprimé dans (Rm) (connu)

Etant donné :

- un ensemble de points 3D décrits dans un référentiel 3D "objet",
- leurs projections 2D décrites dans un référentiel "caméra",
- les paramètres de la caméra,

Le problème de recalage revient à déterminer la transformation rigide (rotation et translation) entre le référentiel objet à un instant donné et le référentiel caméra.

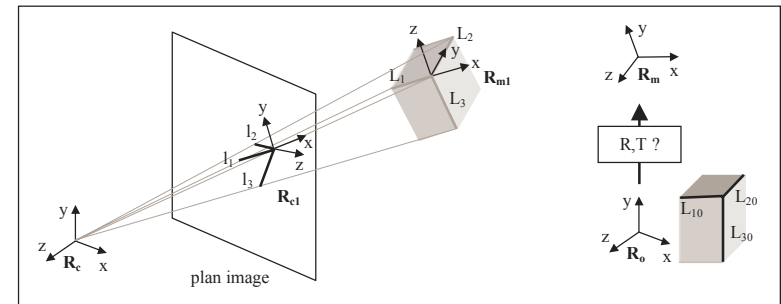
Les solutions proposées peuvent se regrouper en deux catégories (voir [N'ZI 1995] et [MOREAU 1997] pour plus de détails) :

- ① Une solution numérique : le système d'équations à résoudre est non-linéaire. Les méthodes utilisées font appel à des algorithmes d'optimisation qui entraînent souvent des problèmes d'initialisation, de convergence et de temps de calcul.
- ② Une solution analytique par détermination directe des paramètres : les équations à résoudre sont compliquées et donnent plusieurs solutions.

Nous présentons deux méthodes analytiques, une d'entre elles est appliquée pour les petits mouvements, et une méthode numérique basée sur les quaternions et utilise l'algorithme de Levenberg - Marquardt.

4.4.1.2. Méthode géométrique des 3 segments

La détermination directe de l'orientation de (Rm)/(Ro) est complexe car elle fait intervenir trois angles. Afin de simplifier les calculs nous avons utilisé l'approche présentée dans [DHOMÉ 1989] qui consiste à introduire des repères intermédiaires permettant de ne faire intervenir que deux angles.



Soient trois arêtes $L_{i=1,2,3}$ de l'objet définis dans (Rm), et leurs images $l_{i=1,2,3}$ définies dans (Rc). L'objectif est de déterminer la rotation R et la translation T à appliquer à L_i afin que leurs images se projettent sur les droites supports de l_i (Figure 4.24). Le principe est de résoudre les deux équations (4.14) et (4.15).

Figure 4.24 : Les différents repères mis en jeu

Les différents repères mis en jeu sont :

- (Ro) repère de travail (repère objet).
- (Rc) repère de la caméra.
- (Rm) repère lié à l'objet et qui subit donc le même mouvement que l'objet. Dans la position initiale de l'objet, le repère (Rm) est le même que le repère (Ro).
- (Rm1) repère défini à partir des arêtes 3D de l'objet courant.
- (Rc1) repère défini à partir des indices visuels 2D de l'objet courant.

Les matrices de passage suivantes sont définies :

- Roc1 matrice de passage de (Ro) à (Rc1).
- Rmm1 matrice de passage de (Rm) à (Rm1).
- Rc1m1 matrice de passage du repère (Rc1) à (Rm1).
- R, T matrices de rotation et de translation qui permettent de passer de (Ro) à (Rm).

La rotation R cherchée devient une composition de matrices de passage :

$$R = \text{Roc1 Rc1m1 Rm1m} \quad (4.16)$$

La rotation R contient trois variables (les trois angles d'Euler), donc en principe elle peut être trouvée avec trois vecteurs 3D différents et leurs images. L'idée de la méthode des trois segments est de choisir les deux repères Rc1 et Rm1 de telle façon qu'ils aient chacun un axe dans le même plan qui est le plan d'interprétation de l'une des arêtes. Ainsi le passage de Rc1 à Rm1 se fait seulement par deux rotations.

Le développement des équations (4.14) et (4.16) aboutit à un polynôme de degré 8 (cf. annexe 5), auquel est appliqué une méthode numérique basée sur le calcul de valeurs propres de la matrice "compagnon" construite à partir du polynôme [N'ZI 1995]. Les angles α et β qui caractérisent la matrice Rc1m1 sont obtenus. La rotation R subie par l'objet est ensuite calculée par l'équation (4.16). Une fois la rotation obtenue, la translation est déduite avec également trois points appartenant aux segments 3D et leur images. Nous utilisons 3 points formant trois segments pour appliquer la méthode.

Afin d'améliorer les résultats obtenues par cette première résolution valable dans le cas général, nous appliquons une seconde résolution valable pour les faibles mouvements, ce qui permet de linéariser le problème et d'introduire des informations redondantes.

4.4.1.3. Méthode linéaire à petit mouvement

L'objectif est d'affiner le recalage à grande amplitude, à l'aide d'une méthode de reconstruction appliquée aux petites rotations, pour réduire les erreurs [N'ZI 1995]. Le vecteur instantané de rotation utilisé, ensuite la méthode de calcul de R et T sont définis.

- Vecteur instantané de rotation

Une rotation quelconque peut être représentée par trois angles $\theta_x, \theta_y, \theta_z$ dénommés *angles d'Euler*. Chacun de ces angles quantifie la rotation autour d'un axe d'un repère donné. L'avantage de ce formalisme est qu'il représente directement la commande de l'articulation d'un robot. L'inconvénient est que le produit de ces trois rotations n'est pas commutatif, l'ordre dans lequel s'effectuent ces rotations conditionne la rotation finale. Egalement, il présente pour le problème inverse plusieurs solutions et quelques positions singulières (cf. annexe 3).

Une autre façon d'exprimer une rotation est d'utiliser le *vecteur instantané de rotation* dont :

- la direction est celle de la rotation,
- la norme est l'angle de cette rotation

Ce vecteur s'exprime par : $\vec{R} = \theta \cdot \vec{r}$ où \vec{r} est un vecteur unitaire. La matrice de rotation $R(\vec{r}, \theta)$ associée au vecteur \vec{r} est la suivante (cf. annexe 3):

$$R(\vec{r}, \theta) = I_3 + \sin(\theta)X(\vec{r}) + (1 - \cos(\theta))(X(\vec{r}))^2$$

où : $X(\vec{r}) = \begin{bmatrix} 0 & -rz & ry \\ rz & 0 & -rx \\ -ry & rx & 0 \end{bmatrix}$ est la matrice du pré-produit vectoriel, avec $\vec{r} = \begin{bmatrix} rx \\ ry \\ rz \end{bmatrix}$

Cette représentation possède l'avantage de l'interprétation physique que l'on peut en faire : le mouvement d'un solide se décompose en une translation de son centre de gravité et en une rotation autour d'un axe de vecteur directeur \vec{r} .

- Détermination de R et T

Soit \vec{V} le vecteur directeur d'une arête d'un polyèdre subissant une légère déplacement. Le même vecteur après déplacement \vec{V}' peut s'écrire :

$$\vec{V}' = \vec{V} + d\vec{V} \quad \text{avec :} \quad d\vec{V} = \vec{R} \wedge \vec{V}$$

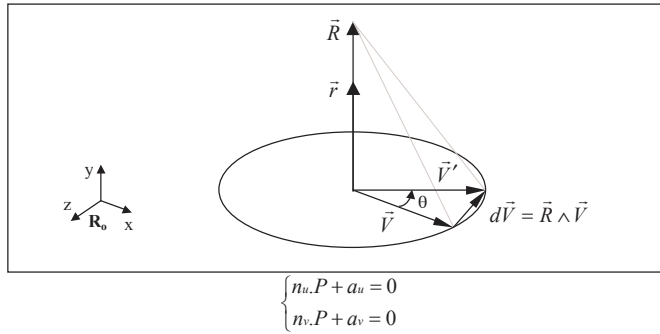
\vec{R} étant le vecteur instantané de rotation, l'écriture de cette équation revient à assimiler l'arc issu de la rotation à la tangente (Figure 4.25).

Figure 4.25 : Approximation des petits angles

Pour un point P, ces expressions deviennent :

$$dP = T + \vec{R} \wedge AP \quad (4.21)$$

Un point P et son image (u,v) dans la caméra sont liés par le rayon visuel (§2.2.1.5) :



qui donnent après dérivation (cas de petits déplacements : en rotation et translation) :

$$\begin{cases} dn_u.P + n_u.dP + da_u = 0 \\ dn_v.P + n_v.dP + da_v = 0 \end{cases}$$

dP est donné par l'équation (4.21) et en utilisant une propriété du produit mixte de trois vecteurs i, j, k quelconques : $i.(jAk) = j.(kAi)$ ces deux équations sont écrites de la manière suivante :

$$\begin{cases} n_u.T + (PAN_u).R + dn_u.P + da_u = 0 \\ n_v.T + (PAN_v).R + dn_v.P + da_v = 0 \end{cases} \quad (4.22)$$

qui peut se mettre sous la forme: $AX = B$

avec : $X = (a, b, c, Rx, Ry, Rz)^t$ où $R = \begin{pmatrix} Rx \\ Ry \\ Rz \end{pmatrix}$ et $T = \begin{pmatrix} a \\ b \\ c \end{pmatrix}$ sont les inconnus

La résolution de ce système linéaire permet de déterminer R et T. Pour obtenir les 6 inconnues un minimum de $n=3$ points est nécessaire. Ces trois points doivent avoir des images distinctes et non alignées. Le système peut être résolu par une méthode de type moindres carrés lorsque $n>3$. Ce recalage dit de petit déplacement permet

ainsi de réduire les erreurs obtenues sur la méthode de recalage pour les grands mouvements (rotation de 0 à 2π et translation quelconque) et peut être itéré.

4.4.1.4. Méthode non-linéaire

La méthode précédente a résolu le problème de recalage au voisinage de l'optimum. En supposant un petit mouvement, les équations non-linéaires du problème ont abouti à un système linéaire grâce au vecteur de rotation. L'évaluation de cette méthode (cf. chapitre 5) montre qu'elle ne converge pas dans tous les cas. Pour évaluer la convergence et la précision de cette méthode, nous la comparons avec une méthode non-linéaire ne faisant aucune simplification au problème.

La méthode non-linéaire résout la rotation et la translation en minimisant un critère d'erreur, celui-ci est exprimé par la formule suivante :

$$\text{Critère} = \text{Min} \left[\sum_{i=1}^N ((\hat{u}_i - u_i)^2 + (\hat{v}_i - v_i)^2) \right] \quad (4.23)$$

Où : N est le nombre de points d'appariement,

(u_i, v_i) les coordonnées d'un point image,

(\hat{u}_i, \hat{v}_i) estimés des coordonnées du point de l'objet après projection sur le plan image

Chaque point 3D $P(x,y,z)$ ayant subi une rotation R et une translation T, est projeté sur l'image pour obtenir ses estimés (cf. §2.2.1.3) :

$$\begin{bmatrix} s.\hat{u} \\ s.\hat{v} \\ s \end{bmatrix} = C_{3 \times 4} \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Le vecteur de paramètres à estimer \vec{p} est constitué des paramètres de rotation et de translation. La translation est exprimée simplement par trois paramètres :

$$T_{3 \times 1} = [a \quad b \quad c]^t$$

Le formalisme d'Olinde-Rodrigues ou *Quaternions* est utilisé pour exprimer la rotation de l'objet par rapport au repère de travail (cf. annexe 3) :

$$\vec{Q} = [\lambda_1 \quad \lambda_2 \quad \lambda_3 \quad \lambda_4]^t$$

Ils donnent une solution unique au problème inverse (passage de la matrice de rotation aux quaternions) et ne présentent aucune position singulière. Les quaternions représentent trois degrés de liberté, nous utilisons donc trois de ces paramètres avec une équation supplémentaire :

$$\lambda_1^2 + \lambda_2^2 + \lambda_3^2 + \lambda_4^2 = 1$$

Le vecteur de paramètres de recalage devient :

$$\vec{p} = [\lambda_2 \quad \lambda_3 \quad \lambda_4 \quad a \quad b \quad c]^T$$

L'application d'une méthode numérique au critère (4.23) permet d'identifier les paramètres recherchés à partir des points 3D et de leurs images 2D. Nous avons choisi la méthode d'optimisation de Levenberg-Marquardt, car son coefficient de réglage λ permet d'éviter les minima locaux. L'algorithme de mise en œuvre de cette méthode est décrit dans l'annexe 4.

Une bonne initialisation du vecteur de paramètres est nécessaire pour assurer la convergence de l'algorithme. La méthode des trois segments est utilisée, là encore, pour donner une bonne initialisation à cette méthode. A chaque itération de l'algorithme, les quaternions sont calculés à partir de la matrice de rotation courante (problème inverse, annexe 3 : A3.15) et par conséquent le vecteur \vec{p} courant. Une nouvelle estimation de \vec{p} est fourni par l'algorithme, le calcul de la matrice de rotation à partir des quaternions se fait par l'équation (annexe 3 : A3.14). L'algorithme s'arrête lorsque le critère d'erreur devient suffisamment petit.

4.4.1.5. Méthode employée (Mixte)

L'évaluation des trois méthodes de recalage étudiées ci-dessus (les résultats sont illustrés au chapitre 5), a permis d'adopter la méthode *mixte*. Celle-ci mélange les trois méthodes pour assurer la convergence vers l'optimum quasiment dans tous les cas. En fait, les méthodes linéaire et non-linéaire nécessitent une bonne initialisation contrairement à la méthode des 3 segments. En revanche, Elles donnent toutes les deux une meilleure précision que cette dernière. Nous utilisons alors la méthode des trois segments qui donne une solution approximative pour initialiser les deux autres méthodes. En tant que précision, la méthode linéaire est meilleure que la non-linéaire, car elle donne une solution analytique optimale dans le sens du critère, tandis que l'autre converge souvent vers des minima locaux. Par ailleurs la méthode linéaire représente des positions singulières (problème inverse du vecteur de rotation, cf. annexe 3) que la méthode non-linéaire ne subit pas.

Voici l'algorithme qui décrit la méthode mixte :

1. Appliquer la méthode des trois segments pour obtenir la matrice de recalage (les trois segments en triangle doivent être non-colinéaires : la superficie du triangle > seuil)
2. Utiliser la méthode linéaire pour affiner le recalage,
 - si le critère d'erreur est suffisamment petit : le résultat est considéré bon, fin
 - sinon : position singulière, continuer
3. Chercher la convergence à l'aide de la méthode non-linéaire,

- si elle ne converge pas : pas de solution, fin
- sinon : sauver le résultat, continuer

4. Appliquer à nouveau la méthode linéaire,

- si le critère d'erreur est suffisamment petit : le résultat est considéré bon, fin
- sinon : pas de solution, fin

4.4.2. Correction locale du traitement d'images

Les méthodes d'appariement ont montré, par expérience, leur dépendance au résultat du traitement d'images. En effet, la perte de primitives existantes dans l'image ou la mauvaise localisation de celles-ci entraînent des conséquences concernant l'appariement. Il existe deux axes de recherche pour améliorer le résultat final de la reconnaissance :

- l'amélioration du traitement d'images lui-même,
- l'amélioration de la méthode d'appariement en utilisant plusieurs approches et approfondir les possibilités de recherche de meilleurs hypothèses

Beaucoup de recherches se font dans le premier axe, dont le problème essentiel est de savoir : quoi chercher dans l'image ? En fait, chercher quelque chose doit correspondre à un phénomène physique existant dans la réalité et pouvant être extractible de l'image. Un exemple est l'extraction de segments, est-ce qu'un segment existe dans la réalité ? Est-ce un invariant quelque soit la position de l'objet et celle de la source de lumière ? Le meilleur traitement se fait donc si on connaissait l'objet recherché dans l'image et sa position ! Il est facile de constater l'interdépendance entre les deux axes mentionnés ci-dessus.

Pour améliorer la reconnaissance, il faut remonter dans le contre sens ; de l'appariement vers le traitement d'images. L'idée est d'utiliser donc les connaissances dans l'hypothèse courante pour refaire le traitement, si nécessaire, dans des zones susceptibles de *renforcer* cette hypothèse (l'améliorer ou la rejeter). Le traitement peut se refaire dans un ou plusieurs niveaux dans la hiérarchie du système. Un exemple d'une telle action est donné lorsque la superposition de l'objet, recalé à la base d'une hypothèse, sur son image, donne des primitives qui n'ont pas été détectées par le traitement (tel qu'un segment ou sommet). Cas auquel il faut remonter dans la hiérarchie du traitement pour en savoir la cause, le traitement se refait d'une manière locale et précise dans un niveau donné (détection de contour, segmentation, chaînage, etc.)

Une telle analyse nécessite l'implantation d'un système expert pouvant prendre les décisions dans les différents niveaux hiérarchiques du système de reconnaissance. Car compliquée, nous n'avons pas traité cette solution générale. Une seule partie a été traitée, celle de la localisation des sommets 2D. En fait, une information importante provenant d'une hypothèse est celle de la liste de segments ayant participé au calcul du barycentre de chaque sommet 2D utilisé dans l'hypothèse. Nous avons vu au chapitre 3 (§3.6.1) qu'un sommet 2D est calculé à partir d'une liste de segments se trouvant en proximité. L'extraction de cette liste, ne dépendant que de l'organisation perceptive, peut être erronée. Si l'hypothèse courante est bonne, elle donnera la bonne liste pour

chaque sommet 2D apparié dans cette hypothèse. Une bonne localisation des sommets se reflète par une bonne vérification de cette hypothèse. Dans le cas contraire, cette «correction» renforcera le rejet de l'hypothèse.

4.5. Conclusion

La méthode du graphe d'appariement permet d'incorporer les contraintes topologiques des objets, en partie grâce au graphe d'aspects. Cela mène à une réduction importante du nombre d'hypothèses pour la mise en correspondance des objets polyédriques avec leur image. Des contraintes supplémentaires de type géométriques (invariants projectifs) ont été abordées pour diminuer la combinatoire de création d'arcs dans le graphe d'appariement. En revanche, il est difficile de mettre cette méthode en œuvre pour l'identification des objets quelconques.

Vu les nombreuses possibilités d'appariement qu'elles offrent, les chaînes ouvertes sont source d'explosion du nombre d'hypothèses. Il faut donc améliorer la qualité de l'image et de son traitement pour réduire le nombre d'hypothèses. Le problème du mauvais chaînage a également été étudié, il nécessite une solution pour trouver une hypothèse valide. L'amélioration du traitement de l'image ne suffit pas pour contourner l'incomplétude de son résultat. D'où l'obligation d'approfondir la recherche dans les méthodes d'appariement.

Nous avons opté pour une autre méthode d'appariement qui prend le relais dans le cas de dégradations du traitement de l'image. En effet, la méthode du hachage géométrique s'est révélée efficace dans ce cas, puisqu'elle est toujours capable de fournir des hypothèses lorsque la première échoue. Elle offre la possibilité de réussir l'appariement à la base des sommets uniquement, dont l'extraction est plus robuste que les chaînes. En outre, le hachage géométrique ne doit pas être considéré comme une méthode conclusive et nécessite d'approfondir ses résultats par vérification.

La phase de vérification des hypothèses est nécessaire pour donner la bonne solution à l'identification d'objets, elle est basée sur le recalage d'objets et sur une correction de la localisation de sommets 2D. Trois méthodes de recalage ont été mises en œuvre. Une validation et comparaison de précisions entre ces méthodes sont données dans le prochain chapitre.

Conclusion générale

Un système de reconnaissance de polyèdres à partir d'une image de luminance a été développé et intégré au MCIT (Multimedia Control Interface in Teleoperation). Il sert à la mise à jour de la base de données des objets pré-modélisés de l'environnement. La superposition des objets reconnus sur l'image de la scène apporte une aide

visuelle à la perception pour la téléopération. Ce système automatise des tâches fastidieuses imposées à l'opérateur, celles d'appairer les primitives des objets vus par une caméra avec celles des modèles de la BD3D pour les identifier puis les localiser.

Les contributions apportées par cette thèse au système MCIT sont les suivantes :

- le développement de modules nécessaires pour effectuer la reconnaissance : traitement d'images et appariement,
- l'amélioration des modules de modélisation, calibration et recalage,
- la mise en œuvre de certaines méthodes : construction numérique et analytique du graphe d'aspects, graphe d'appariement et hachage géométrique,
- l'amélioration d'autres méthodes telles que la transformée de Hough et la procédure globale de la reconnaissance.

Deux méthodes d'appariement sont appliquées :

1. La méthode des graphes d'appariement pour une image de bonne qualité. Elle donne un nombre d'hypothèses très réduit aboutissant à une solution en un temps inférieur à une seconde (reconnaissance de 2 objets parmi 6 de la base). Ce résultat vient en partie de l'utilisation des invariants projectifs, mais la reconnaissance échoue pour une image dégradée.
2. La méthode de hachage géométrique pour contourner les défauts du traitement d'images causés par la dégradation de celles-ci.

La précision du recalage a été améliorée par une méthode mixte, composée d'une méthode géométrique et de méthodes numériques linéaire pour les faibles mouvements et non-linéaire dans le cas contraire. Elle donne une erreur de reconstruction moyenne de 0.55 mm pour une distance inférieure à 2 m entre l'objet et la caméra. Cette précision est obtenue notamment grâce à la calibration automatique de la caméra à l'aide d'un robot à 4ddl.

Dans le chapitre 1, nous avons présenté les principales approches de résolution du problème de la reconnaissance d'objets 3D, une étude comparative entre les SRO (Systèmes de Reconnaissance d'Objets) actuels et une description de notre SRO développé et intégré à MCIT.

Dans le chapitre 2 sont exposés, les méthodes de modélisation, de calibration de la caméra, de modélisation des objets utilisés par MCIT et le développement des deux méthodes d'extraction de graphes d'aspects. Les deux méthodes d'extraction de graphe d'aspects sont applicables aux polyèdres. La première est exhaustive, simple à appliquer et d'une complexité constante (nombre d'aspects fixe) mais ne donne pas le graphe d'aspects complet. La seconde est analytique, elle fournit un graphe d'aspects exact du polyèdre. En revanche, elle est

difficile à mettre en œuvre et d'une complexité polynomiale par rapport au nombre de faces du polyèdre. Les deux méthodes sont utilisées pour servir deux méthodes d'appariement.

Le troisième chapitre concerne l'implémentation du module de traitement d'images basé sur la notion orientée objet. Des filtres successifs sont appliqués pour l'extraction des contours. La transformée de Hough est utilisée pour extraire les segments de droite de l'image. Une amélioration du temps de réponse de cette transformée est apportée. L'organisation perceptive est appliquée pour trouver les relations entre les segments de l'image aboutissant au modèle 2D.

Dans le quatrième chapitre, nous avons présenté les deux méthodes utilisées pour l'appariement entre modèle 2D et les modèles 3D de la BD3D. L'application des invariants projectifs lors de l'élaboration des hypothèses dans le graphe d'appariement constitue une originalité. Ces invariants sont basés sur les rapports entre les superficies des faces du modèle 3D et celles de son image. Pour la construction de la table de hachage, nous appliquons des quasi-invariants obtenus par des bases à deux points. Une méthode hybride de recalage améliorant la précision de celui-ci est décrite. La vérification des meilleures hypothèses d'appariement fournit une solution de la reconnaissance. Le monde virtuel (objets reconnus) est superposé à l'image vidéo.

Le dernier chapitre a montré les tests effectués dans les différentes étapes de la thèse. Ces tests illustrent les intérêts des méthodes appliquées. Grâce à la calibration automatique nous avons pu effectuer une évaluation de l'apport du modèle non-linéaire de la caméra à l'aide d'un grand nombre de tests. Les résultats montrent la véracité de l'étude théorique sur la complexité des méthodes utilisées telles que le graphe d'aspects et le hachage géométrique. Le système a été évalué sur un banc de mesures. Les résultats sont obtenus avec une précision de l'ordre d'un millimètre pour une distance caméra/objet valant 2 mètres. Ces résultats sont encourageants et nous pensons pouvoir les améliorer.

Un résumé de nos perspectives :

- nous sommes conscients des travaux existants dans le traitement d'images, qui sont à explorer ; à savoir : le filtrage optimal (Deriche), la méthode des lignes des partages des eaux (WaterSheds) et l'extraction des segments curvilignes (cercles, ellipses, etc.). Il faut également approfondir la recherche sur les critères d'arrêt de la méthode de la transformée de Hough avec érosion (§3.3.3.2).
- l'extension du système de reconnaissance aux objets quelconques (éventuellement déformables) nécessite une évolution quant à la modélisation, au traitement d'images et à l'intégration d'autres capteurs (télémètre, stéréo, caméra embarquée sur un robot mobile ou avec zoom, sources de lumière contrôlées, etc.).
- nous avons remarqué l'importance du retour d'information de l'appariement pour orienter le traitement d'images (le refaire localement), la nécessité de décision pour le choix des méthodes appliquées en fonction de l'entrée du système (notamment les méthodes d'appariement). La souplesse d'un tel système peut être

réalisée par une implémentation hiérarchique : un système expert sera destiné à l'échange d'informations entre les différents niveaux. Ainsi, une décision est prise à chaque étape : appliquer un filtrage directionnel, changer les seuils d'hystérésis ou choisir une méthode d'appariement donnée etc.

- il serait intéressant d'étendre les contraintes de liaisons aux objets articulés (robot manipulateur). Deux développements seront nécessaires : l'utilisation du raisonnement par évidence et un recalage simplifié par ces contraintes.
- un travail abordable à l'état actuel du système est le suivi des polyèdres dans une séquence d'images. Une application temps réel est possible grâce au résultat de la reconnaissance qui peut être amélioré (quelques secondes sans optimisation).
- les travaux de Gros [GROS 1995] consistent à intégrer le quasi invariant formé par deux segments (rapport de longueurs et angle entre deux segments) dans le hachage géométrique. Nous pouvons généraliser cette méthode en utilisant les relations perceptives (§3.5) de la même manière. En fait, deux segments ayant une forte probabilité de parallélisme ou de colinéarité forment un quasi invariant. Cette probabilité peut être estimée et utilisée pour voter à partir d'une table de hachage, les triplets auront des accumulateurs de probabilités qui rendent l'appariement plus robuste.
- Des travaux sont en cours pour intégrer notre système de reconnaissance à un système de télé-programmation des tâches robotiques.

Bibliographie

- [AYACHE 1986] N. AYACHE ET O.D. FAUGERAS, «HYPER : A NEW APPROACH FOR THE RECOGNITION POSITIONING OF TWO-DIMENSIONAL OBJECTS», **IEEE TRANS. PAMI**, 8(1) , P. 44-54, JANVIER, 1986.
- [AYACHE 1989] N. AYACHE, «VISION STEREOSCOPIQUE ET PERCEPTION MULTISENSORIELLE : APPLICATION A LA ROBOTIQUE MOBILE», **INTEREDITIONS**, 1989.
- [AYACHE 1991] N. AYACHE ET F. LUSTMAN, «TRINOCULAR STEREO VISION FOR ROBOTICS», **IEEE TRANS. ON PAMI**, 13(1), P. 73-85, JANVIER, 1991.
- [BARAT 1997] CH. BARAT, «MODELISATION D'UN SYSTEME DE RELEVÉ TELEMETRIQUE LASER 3D : APPLICATION A LA ROBOTIQUE» , **THESE DE DOCTORAT, UNIVERSITE EVRY VAL D'ESSONNE, FRANCE, SEPTEMBRE, 1997.**
- [BERGEVIN 1993] R. BERGEVIN ET M.D. LEVINE, «GENERIC OBJECT RECOGNITION : BUILDING AND MATCHING COARSE DESCRIPTIONS FROM LINE DRAWINGS», **IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE**, 15(1), P. 19-36, JANVIER, 1993.
- [BESANÇON 1988] J.E. BESANÇON, «VISION PAR ORDINATEUR EN DEUX ET TROIS DIMENSIONS», **EDITIONS EYROLLES, PARIS, FRANCE, 1988.**

[BINFORD 1993] T.O. BINFORD ET T.S. LEVITT, «QUASI INVARIANTS : THEORY AND EXPLOITATION», IN PROCEEDINGS OF DARPA, IMAGE UNDERSTANDING WORKSHOP, P. 819-829, 1993.

[BOLLES 1981] R.C. BOLLES ET M.A. FISCHLER, «A RANSAC-BASED APPROACH TO MODEL FITTING AND ITS APPLICATION TO FINDING CYLINDERS IN RANGE DATA», INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE, P. 637-643, VANCOUVER, CANADA, 1981.

[BOLLES 1982] R.C. BOLLES ET R.A. CAIN, «RECOGNIZING AND LOCATING PARTIALLY VISIBLE OBJECTS : THE LOCAL FEATURE -FOCUS METHOD», INTERNATIONAL JOURNAL OF ROBOTICS RESEARCH , 1(3) , P. 57-82, 1982.

[BOLLES 1986] R.C. BOLLES ET R. HORAUD, «3DPO : A THREE DIMENSIONAL PART ORIENTATION SYSTEM», INTERNATIONAL JOURNAL OF ROBOTICS RESEARCH, 5(3) , P. 3-26, 1986.

[BONNEAU 1993] P. BONNEAU ET PH. EVEN, « MAN MACHINE COOPERATION FOR 3D OBJECTS POSE ESTIMATION », IEEE/SMC'93, SYSTEMS, MAN AND CYBERNETICS, P. 294-299, LE TOUQUET, FRANCE, 1993.

[BOWYER 1992] K.W. BOWYER, «WORKSHOP PANEL REPORT - WHY ASPECT GRAPHS ARE NOT (YET) PRACTICAL FOR COMPUTER VISION», CVGIP : IMAGE UNDERSTANDING, VOL. 55, N°2, P. 212-218, MARS, 1992.

[BROOKS 1981] R.A. BROOKS, «SYMBOLIC REASONING AMONG 3D MODELS AND 2D IMAGES», ARTIFICIAL INTELLIGENCE, 17, P. 285-348, 1981

[BROOKS 1983] R.A. BROOKS, «MODEL-BASED 3D INTERPRETATION OF 2D IMAGES», IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 5(2), P. 140-150, MARS, 1983.

[BURDEA 1993] G. BURDEA ET PH. COIFFET, «LA REALITE VIRTUELLE», EDITIONS HERMES, 1993.

[CAELLI 1994] T. CAELLI ET A. DREIER, «VARIATION ON THE EVIDENCE BASED OBJECT RECOGNITION THEME», PATTERN RECOGNITION, 27(2), P. 185-204, 1994.

[CHAUMETTE 1990] F. CHAUMETTE, «LA RELATION VISION-COMMANDE : THEORIE ET APPLICATION A DES TACHES ROBOTIQUES», THESE DE DOCTORAT, UNIVERSITE DE RENNES I, IFSIC, JUILLET, 1990.

[CHAVAND 1991] F. CHAVAND, E. COLLE, M. MALLEM ET A. LOUKIL, «SYSTEME D'AIDE AU CONTROLE-COMMANDE A DISTANCE DE DISPOSITIFS ROBOTISES», ORIA'91, 4TH INTERNATIONAL SYMPOSIUM ON OFFSHORE, ROBOTICS AND ARTIFICIAL INTELLIGENCE, P. 199-205, MARSEILLE, 11-12 DECEMBRE, 1991.

[CHAVAND 1992] F.CHAVAND, E. COLLE, M. MALLEM ET A. LOUKIL, «SYSTEM FOR THE REMOTE MONITORING AND CONTROL OF ROBOTIZED DEVICES», PROCEEDINGS OF THE IMACS/SICE INTERNATIONAL SYMPOSIUM ON ROBOTICS, MECHATRONICS AND MANUFACTURING SYSTEM '92 KOBE, P. 1077-1082, JAPON, 16-20 SEPTEMBRE, 1992.

[CHAVAND 1994] F. CHAVAND, J.M. OLLIER ET S. LELANDAIS, «UPDATING THE ENVIRONMENT'S GEOMETRIC DATA BASE IN TELEROBOTICS : DETERMINATION OF POSITION/ORIENTATION OF KNOWN OBJECTS», IEEE SMC'94, SAN ANTONIO, USA, 2-5 OCTOBRE, 1994.

[CHAVAND 1997] F. CHAVAND, M. SHAHEEN ET M. MALLEM, «MATCHING BETWEEN A 2D-IMAGE AND ITS 3D-MODEL : APPLICATION TO UPDATING 3D INFORMATION OF WORKING SPACES», INTERNATIONAL SYMPOSIUM ON ARTIFICIAL INTELLIGENCE, ROBOTICS, AND INTELLECTUAL HUMAN ACTIVITY SUPPORT FOR NUCLEAR APPLICATIONS, WAKO-SHI, SAITAMA, JAPON, 19-21 NOVEMBRE, 1997.

[CHEKHAR 1994A] Y. CHEKHAR ET E. COLLE, «CONSTRUCTION D'IMAGES DE PROFONDEUR PAR TELEMETRIE LASER», ICEEE-94, INTERNATIONAL CONFERENCE ON ELECTRICAL & ELECTRONIC ENGINEERING, PP104-111, ORAN, ALGERIE, 1-3 OCTOBRE, 1994.

[CHEKHAR 1994B] Y. CHEKHAR, A. LOUKIL, E. COLLE ET F. CHAVAND, «CALIBRATION MULTICAPTEUR D'UN SYSTEME D'AIDE A LA TELEOPERATION», REVUE APII, VOL. 6, DECEMBRE, 1994.

[CHEKHAR 1994C] Y. CHEKHAR, «SAISIE ET TRAITEMENT D'IMAGES TELEMETRIQUES -APPLICATION A LA TELEOPERATION», THESE DE DOCTORAT, UNIVERSITE D'EVRY VAL D'ESSONE, 13 DECEMBRE, 1994.

[CHEN 1992] C. CHEN, M. TRIVEDI ET C.R. BIDLACK, «SIMULATION AND GRAPHICAL INTERFACE FOR PROGRAMMING AND VISUALIZATION OF SENSOR BASED ROBOT OPERATION», IEEE PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION, NICE, MAI, 1992.

[COLLE 1994A] COLLE E., ET Y. CHEKHAR, «PROCESSUS DE CONSTRUCTION D'IMAGES 3D», REVUE RAPA, VOL. 7, N°2, EDITIONS HERMES, 1994.

[COLLE 1994B] E. COLLE, Y. CHEKHAR ET CH. BARAT, «3D IMAGE BUILDING PROCESS FROM LASER RANGE FINDER DATA», ORIA'1994, 5TH INTERNATIONAL SYMPOSIUM & BUSINESS CONVENTION, MARSEILLE, DECEMBRE, 1994.

[COLY 1991] J. COLLY ET M. DEVY, «RECONNAISSANCE D'UN OBJET 3D PAR UN SYSTEME MULTISENSORIEL, LASER/CAMERA», AFCET 8EME CONGRES

DES FORMES ET INTELLIGENCE ARTIFICIELLE, LYON-VILLEURBANNE, FRANCE, 25-29 NOVEMBRE, 1991.

[DERICHE 1987] R. DERICHE, «USING CANNY'S CRITERIA TO DERIVE A RECURSIVELY IMPLEMENTED OPTIMAL EDGE DETECTOR», INTERNATIONAL JOURNAL OF COMPUTER VISION, 1(2), P. 167-187, 1987.

[DEVY 1997] M. DEVY, V. GARRIC ET J.J. ORTEU, «CAMERA CALIBRATION FROM MULTIPLE VIEWS OF A 2D OBJECT, USING A GLOBAL NON LINEAR MINIMIZATION METHOD», IEEE IROS'97, INTERNATIONAL CONFERENCE ON INTELLIGENT ROBOTS AND SYSTEMS, VOL. 3, P. 1583-1589, GRENOBLE, SEPTEMBRE, 1997.

[DHOME 1989] M. DHOME, M. RICHTIN, J.T. LAPRESTE ET G. RIVES, «DETERMINATION OF THE ATTITUDE OF 3D OBJECTS FROM A SINGLE PERSPECTIVE VIEW», IEEE TRANS. ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 11, N°12, P. 1265-1278, DECEMBRE, 1989.

[DICKINSON 1992] S.J. DICKINSON, A.P. PENTLAND ET A. ROSENFELD, «FROM VOLUMES TO VIEWS : AN APPROACH TO 3D OBJECT RECOGNITION», COMPUTER VISION, GRAPHICS AND IMAGE PROCESSING : IMAGE UNDERSTANDING, 55(2), P. 130-154, MARS, 1992.

[DOURILLE 1995] X. DOURILLE, «APPARIEMENT ENTRE UNE IMAGE ET SON MODELE EN UTILISANT LES LIAISONS ENTRE OBJETS», RAPPORT DE DEA CMPI, PARIS XII, 1995.

[EDELSBRUNNER 1986] H. EDELSBRUNNER, J. OROURKE ET R. SEIDEL, «CONSTRUCTING ARRANGEMENTS OF LINES AND HYPERPLANES WITH APPLICATIONS», SIAM JOURNAL ON COMPUTING, 15(2), P. 341-363, MAI, 1986.

[ESPIAU 1986] B. ESPIAU, «TELEOPERATION AVANCEE : LE PROJET DE TELEOPERATION AVANCEE», JOURNEES ,CEN – SACLAY, 3-5 DECEMBRE, 1986.

[EVEN 1993] PH. EVEN ET R. FOURNIER, «TELEROBOTICS TASKS EXECUTION BASED ON 3D GEOMETRIC MODELLING AND GRAPHICAL PROGRAMMING», IEEE/SMC'93, SYSTEMS, MAN AND CYBERNETICS, P. 132-137, LE TOUQUET, FRANCE, 1993.

[FAUGERAS 1993] O.G. FAUGERAS, «THREE DIMENSIONAL COMPUTER VISION : A GEOMETRIC VIEWPOINT», MIT PRESS, BOSTON, 1993.

[FLYNN 1992] J. FLYNN ET K. JAIN, «3D OBJECT RECOGNITION USING INVARIANT FEATURE INDEXING OF INTERPRETATION TABLES», COMPUTER VISION, GRAPHICS AND IMAGE PROCESSING : IMAGE UNDERSTANDING, 55(2), P. 119-129, MARS, 1992.

[GARNESON 1991] PH. GARNESON ET G. GIRAUDON, «POLYGONAL APPROXIMATION : OVERVIEW AND PERSPECTIVES», RAPPORT DE RECHERCHE N° 1621, PROGRAMME 4 : ROBOTIQUE, IMAGE ET VISION, INRIA, SOPHIA ANTIPOLIS, JUIN, 1991.

[GAVRILA 1992] D.M. GAVRILA ET F.C.A. GROEN, «3D OBJECT RECOGNITION FROM 2D IMAGES USING GEOMETRIC HASHING», PATTERN RECOGNITION LETTER, (13)263-278, 1992.

[GRIMSON 1984] W.E.L. GRIMSON ET T. LOZANO-PEREZ, «MODEL-BASED RECOGNITION AND LOCALIZATION FROM SPARSE RANGE OR TACTILE DATA», INTERNATIONAL JOURNAL OF ROBOTICS RESEARCH, 5(3), P. 3-34, AUTOMNE, 1984.

[GRIMSON 1987] W.E.L. GRIMSON ET T. LOZANO-PEREZ, «LOCALIZING OVERLAPPING PARTS BY SEARCHING THE INTERPRETATION TREE», IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. PAMI-9, N°4, P. 469-482, JUILLET, 1987.

[GRIMSON 1991] W.E.L. GRIMSON, «LOCALIZING OVERLAPPING PARTS BY SEARCHING THE INTERPRETATION TREE», IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 13, N°9, P. 920-935, SEPTEMBRE, 1991.

[GRIMSON 1993] W.E.L. GRIMSON, T. LOZANO-PEREZ, S.J. WHITE AND N. NOBLE, «RECOGNIZING 3D OBJECTS USING CONSTRAINED SEARCH», THREE-DIMENSIONAL OBJECT RECOGNITION SYSTEMS, ELSEVIER SCIENCE PUBLISHERS B.V., P. 259-284, 1993.

[GROS 1995] P. GROS, «MATCHING AND CLUSTERING : TWO STEPS TOWARDS OBJECTS MODELLING IN COMPUTER VISION», THE INTERNATIONAL JOURNAL OF ROBOTICS RESEARCH, 14(5), OCTOBRE, 1995.

[HANSEMANN 1994] J. HANSEMANN, «RECOGNITION AND LOCALIZATION OF TECHNICAL OBJECTS. RESULTS OF THE TELEMAN PROGRAMM VISYS», IMAC'S INTERNATIONAL SYMPOSIUM ON SIGNAL PROCESSING, ROBOTICS AND NEURAL NETWORKS, VILLENEUVE D'ASCQ, FRANCE, P. 266-269, AVRIL, 1994.

[HANSEN 1989] C. HANSEN ET T.C. HENDERSON, «CAGD-BASED COMPUTER VISION», IEEE TRANSACTIONS ON PAMI, 11(11), P. 1181-1193, NOVEMBRE, 1989.

[HANSEN 1997] K. HANSEN ET J.D. ANDERSEN, «UNDERSATANDING THE HOUGH TRANSFORM : HOUGH CELL SUPPORT AND ITS UTILISATION», IMAGE AND VISION CPMUTING, VOL. 15, P. 205-218, ELSEVIER SCIENCE PUBLISHERS B.V., 1997.

[HASEGAWA 1991] T. HASEGAWA, T. SUEHIRO, T. OGASAWARA, T. MATSUI, K. KITAGARI ET K. TAGASE, «A MODEL-BASED TELE-ROBOT SYSTEM WITH MANIPULATION SKILLS», ISART TOKYO, JAPON, P. 499-506, MARS, 1991.

[HEATH 1998] M. HEATH, S. SARKAR, T. SANOCKI ET K. BOWYER, «COMPARISON OF EDGE DETECTORS : A METHODOLOGY AND INITIAL STUDY», COMPUTER VISION AND IMAGE UNDERSTANDING, VOL. 69, N°1, P. 38-54, JANVIER, 1998.

[HIRAI 1990] S. HIRAI, T. SATO, T. MATSUI ET M. KAKIKURA, «INTEGRATION OF A TASK KNOWLEDGE BASE AND COOPERATIVE MANEUVERING SYSTEM FOR THE TELEROBOT MEISTER», IROS'90, TSUCHIURA, JUILLET, 1990.

[HORAUD 1984] R. HORAUD ET R.C. BOLLES, «3DPO'S STRATEGY FOR MATCHING THREE-DIMENSIONAL OBJECTS IN RANGE DATA», IEEE CONFERENCE ON ROBOTICS & AUTOMATION, P. 78-85, ATLANTA, 13-15 MARS, 1984.

[HORAUD 1993] R. HORAUD ET O. MONGA, «VISION PAR ORDINATEUR, OUTILS FONDAMENTAUX», HERMES, PARIS, FRANCE, 1993.

[IKEUCHI 1986] K. IKEUCHI, «PRECOMPILING A GEOMETRICAL MODEL INTO AN INTERPRETATION TREE FOR OBJECT RECOGNITION IN BIN-PICKING TASKS», IN PROCEEDINGS OF DARPA IMAGE UNDERSTANDING WORKSHOP, P. 321-339, LOS ANGELES, CALIFORNIE, FEVRIER, 1986.

[IKEUCHI 1987A] K. IKEUCHI, «GENERATING AN INTERPRETATION TREE FROM A CAD MODEL FOR 3D-OBJECT RECOGNITION IN BIN-PICKING TASKS», INTERNATIONAL JOURNAL OF COMPUTER VISION, 1(2), P. 145-165, 1987.

[IKEUCHI 1987B] K. IKEUCHI, «DETERMINING A DEPTH MAP USING A DUAL PHOTOMETRIC STEREO», INTERNATIONAL JOURNAL ROBOTICS RESEARCH, VOL. 6, N°1, 1987.

[IKEUCHI 1988A] K. IKEUCHI ET T. KANADE, «MODELING SENSORS AND APPLYING SENSOR MODEL TO AUTOMATIC GENERATION OF OBJECT RECOGNITION PROGRAMS», IN PROCEEDINGS OF DARPA IMAGE UNDERSTANDING WORKSHOP, AVRIL, 1988.

[IKEUCHI 1988B] K. IKEUCHI ET T. KANADE, «AUTOMATIC GENERATION OF OBJECT RECOGNITION PROGRAMS», PROCEEDINGS IEEE, 76(8), P. 1016-1035, 1988.

[JAIN 1988] A.K. JAIN ET R. HOFFMAN, «EVIDENCE-BASED RECOGNITION OF 3-D OBJECTS», IEEE TRANS. ON PAMI, 10(6), P. 783-802, NOVEMBRE, 1988.

[KAOULA 1994] K. KAOULA, «CONTRIBUTION DE L'APPROCHE REGION A L'ANALYSE DE SCENES DYNAMIQUES EN VISION BINOCULAIRE», THESE DE

DOCTORAT, UNIVERSITE DE TECHNOLOGIES DE COMPIEGNE, 28 MARS, 1994.

[KHALIL 1993] W. KHALIL, «MODELISATION DES ROBOTS I», POLYCOPIE, CHAPITRE 2, P. 1-25, ECOLE CENTRALE DE NANTES, FRANCE, 1993.

[KIM 1993] W. KIM, «GRAPHICAL OPERATOR INTERFACE FOR SPACE TELEROBOTICS», IN I3E, INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION, P. 761-768, ATLANTA, GEORGIE, 2-6 MAI, 1993.

[KITCHEN 1988] L.J. KITCHEN ET J.B. BURNS, «RAPID OBJECT RECOGNITION FROM LARG MODEL BASE USING PREDICTION HIERARCHIES», IN PROCEEDINGS OF DARPA IMAGE UNDERSTANDING WORKSHOP, VOL. II, P. 711-716, 1988.

[KOENDERINK 1979] J.J. KOENDERINK ET A.J. VAN DOORN, «THE INTERNAL REPRESENTATION OF SOLID SHAPE WITH RESPECT TO VISION», BIOLOGICAL CYBERNETICS, (32), P. 211-216, NOVEMBRE, 1979.

[KOIZUMI 1988] M. KOIZUMI ET F. TOMITA, «QUALITATIVE AND QUANTITATIVE MATCHING OF SOLID MODELS AND IMAGES OF 3D OBJECTS», PROCEEDINGS 9TH INTERNATIONAL CONFERENCE ON PATTERN RECOGNITION, P. 681-684, NOVEMBRE, 1988.

[LAM DAN 1988] Y. LAM DAN ET H.J. WOLFSON, «GEOMETRIC HASHING : A GENERAL AND EFFICIENT MODEL-BASED RECOGNITION SCHEME», SECOND IEEE INTERNATIONAL CONFERENCE COMPUTER VISION, P. 238-249, TAMPA, FLORIDE, DECEMBRE, 1988.

[LIMSAKOUNE 1991] K. LIMSAKOUNE, «ETUDE DE LA MODELISATION DES ROBOTS MOBILES ET DE LEUR ENVIRONNEMENT : APPLICATION AU DEVELOPPEMENT D'UN LOGICIEL CAO OPTIMISANT UNE MISSION», THESE DE DOCTORAT, UNIVERSITE PARIS VI, 14 MAI, 1991.

[LIU 1990] Y. LIU, T.S. HUANG ET O.D. FAUGERAS, «DETERMINATION OF CAMERA LOCATION FROM 2D TO 3D LINE AND POINT CORRESPONDANCES», IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 12(1), P. 28-37, JANVIER, 1990.

[LOUKIL 1993] A. LOUKIL, «INTERFACE HOMME-MACHINE DE CONTROLE-COMMANDE EN ROBOTIQUE TELEOPEREE», THESE DE DOCTORAT, UNIVERSITE D'EVRY VAL D'ESSONE, 09 DECEMBRE, 1993.

[LOUKIL 1994] A. LOUKIL, Y. CHEKHAR, F. CHAVAND, E. COLLE ET J. TRIBOULET, «MODELISATION ET CALIBRATION D'UN SYSTEME MULTICAPTEUR. APPLICATION A LA TELEOPERATION», ICEEE-94, INTERNATIONAL CONFERENCE ON ELECTRICAL & ELECTRONIC ENGINEERING, P. 117-124, ORAN, ALGERIE, 1-3 OCTOBRE, 1994.

[LOWE 1987] D.G. LOWE, «THREE DIMENSIONAL OBJECT RECOGNITION FROM SINGLE TWO-DIMENSIONAL IMAGES», ARTIFICIAL INTELLIGENCE, VOL. 31, P. 355-395, ELSEVIER SCIENCE PUBLISHERS B.V., NORTH-HOLLAND, 1987.

[MALLEM 1990] M. MALLEM, «AIDE A LA PERCEPTION EN TELEOPERATION : SUPERPOSITION A UNE IMAGE CAMERA D'UNE IMAGE SYNTHETIQUE ANIMEE EN 'TEMPS REEL' A PARTIR D'INFORMATIONS CAPTEURS», THESE DE DOCTORAT, UNIVERSITE PARIS XII, AVRIL, 1990.

[MALLEM 1991] M. MALLEM, E. COLLE ET F. CHAVAND, «CREATION AND ANIMATION OF COMPUTER-GENERATED IMAGES COMBINED WITH A CAMERA AND SMART GRAPHICS CARD», MICROPROCESSORS AND MICROSYSTEMS, VOL. 15, N°8, OCTOBRE, 1991.

[MALLEM 1992] M. MALLEM, F. CHAVAND ET E. COLLE, «COMPUTER-ASSISTED VISUAL PERCEPTION IN TELEOPERATED ROBOTICS», ROBOTICA, INTERNATIONAL JOURNAL OF INFORMATION, EDUCATION AND RESEARCH IN ROBOTICS AND ARTIFICIAL INTELLIGENCE, EDITED BY J. ROSE, CAMBRIDGE UNIVERSITY PRESS, ANGLETERRE, VOL.10, P. 93-103, 1992.

[MALLEM 1993A] M. MALLEM, S. ROUGEAUX, H. MELLANGER ET F. CHAVAND, «A TRAJECTORY GENERATION MODULE FOR 2D AND 2D 1/2 ENVIRONMENT», PROCEEDINGS IEEE COMPUTERS IN DESIGN, MANUFACTURING AND PRODUCTION, 7TH ANNUAL EUROPEAN COMPUTER CONFERENCE, 24-27 MAI, 1993.

[MALLEM 1993B] M. MALLEM, E. COLLE ET F. CHAVAND, «MULTIMEDIA CONTROL INTERFACE IN TELEOPERATION», PROCEEDINGS OF IEEE INTERNATIONAL CONFERENCE ON SYSTEM, MAN AND CYBERNETICS, P. 126-131, LE TOUQUET, FRANCE, 17-20 OCTOBRE, 1993.

[MALLEM 1996] M. MALLEM, M. SHAHEEN, X. DOURILLE ET F. CHAVAND, «A MATCHING METHOD BETWEEN AN IMAGE AND ITS 3D-MODEL USING A GEOMETRIC CONSTRAINT APPROACH BASED ON CONTACT», CESA'96 IMACS MULTICONFERENCE, P. 565-569, LILLE, FRANCE, JUILLET, 1996.

[MCCANE 1996] B.J. MC CANE, «LEARNING TO RECOGNIZE 3D OBJECTS FROM 2D INTENSITY IMAGES», DEPARTEMENT OF COMPUTER SCIENCE, JAMES COOK UNIVERSITY OF NORTH QUEENSLAND, THESE PHD., MARS, 1996.

[MOREAU 1997] G. MOREAU, M. MALLEM, F. CHAVAND ET E.C. N'ZI, «TWO 3D RECOVERING METHODS FOR ROBOT CONTROL», IFAC'97, SYROCO, P. 531-537, NANTES, FRANCE, 1997.

[MUNDY 1994] J.L. MUNDY, C. HUANG, D. FORSYTH, C. ROTHWELL, A. ZISSRMAN, S. UTCCKE ET O. BOURNES, «MORSE : A 3D OBJECT RECOGNITION SYSTEM BASED ON GEOMETRIC INVARIANTS», IN 23RD IMAGE UNDERSTANDING WORKSHOP, P. 1393-1402, 1994.

[MUNDY 1995] J.L. MUNDY, R. CURWEN, J. LIU, C. ROTHWELL, A. ZISSRMAN ET D. FORSYTH, «MORSE : AN ARCHITECTURE FOR 3D OBJECT RECOGNITION BASED ON INVARIANTS», IN SECOND ASIAN CONFERENCE ON COMPUTER VISION, VOLUME II, PAGES 16-21. SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING, NANYANG TECHNOLOGICAL UNIVERSITY, SINGAPORE, DECEMBRE, 1995.

[NASHASHIBI 1992] F. NASHASHIBI, M. DEVY ET PH. FILLATREAU, «INDOOR SCENE MODELING USING MULTIPLE IMAGES FOR AUTONOMOUS MOBILE ROBOTS», INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION, IEEE, P. 40-46, NICE, FRANCE, MAI, 1992.

[NASHASHIBI 1993] F. NASHASHIBI, «PERCEPTION ET MODELISATION D'UN ENVIRONNEMENT TRIDIMENSIONNEL POUR LA NAVIGATION AUTONOME D'UN ROBOT MOBILE». THESE DE DOCTORAT, UNIVERSITE PAUL SABATIER, TOULOUSE, 19 JANVIER, 1993.

[N'ZI 1994] C. N'ZI, F. CHAVAND, M. MALLEM ET J. TRIBOULET, «METHODS FOR UPDATING THE ENVIRONMENT'S GEOMETRIC DATABASE IN TELEROBOTICS», IMACS SPRANN'94, SIGNAL PROCESSING ROBOTICS AND NEURAL NETWORKS, P. 628, LILLE, FRANCE, 1994.

[N'ZI 1995A] E.C. N'ZI, M. MALLEM ET F. CHAVAND, «METHOD FOR UPDATING THE ENVIRONMENT 3D GEOMETRICAL DATABASE USING SINGLE CAMERA VIEW», IFIP WG 7.6 WORKING CONFERENCE, P. 30.1-30.7, NOISY LE GRAND, FRANCE, 28-30 MAI, 1996.

[N'ZI 1995B] E.C. N'ZI, «MODELISATION ET RECONSTRUCTION 3D INTERACTIVE D'ENVIRONNEMENT : APPLICATION A LA TELEOPERATION», THESE DE DOCTORAT, UNIVERSITE EVRY VAL D'ESSONNE, FRANCE, 1995.

[OLLIER 1994] J.M. OLLIER, «RECONSTRUCTION D'OBJET EN 3D A PARTIR D'INDICES VISUELS 2D», DEA MODELISATION ET CONTROLE DE SYSTEMES INDUSTRIELS, PARIS XII, JUILLET, 1994.

[OPDERBECKE 1994] J. OPDERBECKE, «LOCALISATION D'UN ROBOT MOBILE PAR FUSION DE DONNEES MULTISENSORIELLES DANS UN ENVIRONNEMENT STRUCTURE », THESE DE DOCTORAT, UNIVERSITE PARIS VI, 21 JANVIER, 1994.

[PAMPAGNIN 1990] L.H. PAMPAGNIN, «RECONNAISSANCE D'OBJETS TRIDIMENSIONNELS EN PERCEPTION MONOCULAIRE ET

MULTISENSORIELLE - APPLICATION A LA ROBOTIQUE SPATIALE», THESE DE DOCTORAT, UNIVERSITE PAUL SABATIER, TOULOUSE, 31 OCTOBRE, 1990.

[PAMPAGNIN 1991] L.H. PAMPAGNIN ET M. DEVY, «3D OBJECT IDENTIFICATION BASED ON MATCHING BETWEEN A SINGLE IMAGE AND A MODEL», IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION, SACRAMENTO, USA, P. 1580-1587, 9-11 AVRIL, 1991.

[PLATINGA 1990] H. PLATINGA ET C. DYER, «VISIBILITY, OCCLUSION AND THE ASPECT GRAPH», INTERNATIONAL JOURNAL OF COMPUTER VISION, 5(2), P. 137-160, 1990.

[PRESS 1992] W. H. PRESS, S. A. TEUKOLSKY, W. T. VETTERLING ET B. P. FLANNERY, «NUMERICAL RECIPES IN C : THE ART OF SCIENTIFIC COMPUTING», CAMBRIDGE UNIVERSITY PRESS, 1992.

[QIANG 89] SH. QIANG, «STRATEGIE DE LOCALISATION ET IDENTIFICATION D'OBJET A PARTIR DE QUELQUES MESURES TRIDIMENSIONNELLES», THESE DE DOCTORAT DE L'INSTITUT NATIONAL POLYTECHNIQUE DE LORRAINE, 10 JUILLET, 1989.

[REIS 1991] PH. REIS, «VISION MONOCULAIRE POUR LA NAVIGATION D'UN ROBOT MOBILE DANS UN UNIVERS PARTIELLEMENT MODELISE», THESE DE DOCTORAT, UNIVERSITE DE CLERMONT-FERRAND, 6 MARS, 1991.

[ROSENFELD 1982] A. ROSENFELD ET A.C. KAK, «DIGITAL PICTURE PROCESSING», ACADEMIC PRESS, ORLANDO, FLORIDE, 1982.

[SARKAR 1995] S. SARKAR, «TRACKING 2D STRUCTURES USING PERCEPTUAL ORGANIZATION PRINCIPLES», RAPPORT INTERNE N°11-94-01, P. 1-30, DEPARTEMENT OF COMPUTER SCIENCE AND ENGINEERING, UNIVERSITY OF SOUTH FLORIDA, 1995.

[SEDGEWICK 1990] R. SEDGEWICK, «ALGORITHMS IN C», ADDISON-WESLEY PUBLISHING COMPANY, INC., READING, MASS., USA, 1990.

[SEIBERT 1992] M. SEIBERT ET A. WAXMAN, «ADAPTIVE 3D OBJECT RECOGNITION FROM MULTIPLE VIEWS», IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, 14(2), P. 107-123, FEVRIER, 1992.

[SHERIDAN 1989] T.B. SHERIDAN, «TELEROBOTICS», AUTOMATICA, VOL. 25, N°4, P. 487-507, 1989.

[SOLINA 1994] F. SOLINA, A. LEONARDIS ET A. MARCEL, «A DIRECT PART-LEVEL SEGMENTATION OF RANGE IMAGES USING VOLUMETRIC MODELS», IEEE, ROBOTICS AND AUTOMATION, P. 2254-2259, SAN DIEGO, CALIFORNIE, 8-13 MAI, 1994.

[STEWMAN 1988] J.H. STEWMAN ET K.W. BOWYER, «CREATING THE PERSPECTIVE PROJECTION ASPECT GRAPH OF POLYHEDRAL OBJECTS», IN IEEE INTERNATIONAL CONFERENCE ON COMPUTER VISION, P. 494-500, 1988.

[SUGIMOTO 1988] K. SUGIMOTO, H. TAKAHASHI ET F. TOMITA, «SCENE INTERPRETATION BASED ON BOUNDARY REPRESENTATION OF STEREO IMAGES», PROCEEDINGS 9TH INTERNATIONAL CONFERENCE ON PATTERN RECOGNITION, P. 155-159, NOVEMBRE, 1988.

[SWAIN 1988] M. SWAIN, «OBJECT RECOGNITION FROM A LARGE DATA BASE USING A DECISION TREE», IN PROCEEDINGS OF DARPA IMAGE UNDERSTANDING WORKSHOP, VOL. II, P. 690-696, 1988.

[TACHI 1991] S. TACHI, H. ARAI, T. MAEDA, E. OYAMA, N. TSUNEMOTO ET Y. INOUE, «TELE-EXISTANCE EXPERIMENTAL SYSTEM FOR REMOTE OPERATION WITH A SENSATION OF PRESENCE», ISART, P. 451-457, TOKYO, JAPON, MARS, 1991.

[TAKAHASHI 1988A] H. TAKAHASHI ET F. TOMITA, «PLANARITY CONSTRAINT IN STEREO MATCHING», PROCEEDINGS 9TH INTERNATIONAL CONFERENCE ON PATTERN RECOGNITION, NOVEMBRE, 1988.

[TAKAHASHI 1988B] H. TAKAHASHI ET F. TOMITA, «SELF CALIBRATION OF STEREO CAMERAS», PROCEEDINGS 9TH INTERNATIONAL CONFERENCE ON PATTERN RECOGNITION, NOVEMBRE, 1988.

[TRIBOULET 1996] J. TRIBOULET, «CARACTERISATION D'UN SYSTEME MULTICAPTEUR POUR LA MODELISATION D'ENVIRONNEMENT : APPLICATION A LA TELEOPERATION», THESE DE DOCTORAT, UNIVERSITE EVRY VAL D'ESSONNE, FRANCE, 1996.

[TSAI 1987] R.Y. TSAI, «A VERSATILE CAMERA CALIBRATION TECHNIQUE FOR HIGH-ACCURACY 3D MACHINE VISION METROLOGY USING OFF-THE-SHELF TV CAMERAS AND LENSES», IEEE JOURNAL OF ROBOTICS AND AUTOMATION, VOL. RA-3, N°4, P. 323-344, AOÛT, 1987.

[TSUKIYAMA 1988] T. TSUKIYAMA ET J. SATO, «INCREMENTAL RECONSTRUCTION OF A SCENE MODEL FOR MOBIL ROBOTS FROM A SEQUENCE OF MONOCULAR IMAGES», IEEE INTERNATIONAL WORKSHOP ON INTELLIGENT ROBOTS AND SYSTEMS, IROS 88, P. 665-670, 1988.

[WOLFSON 1992] H.J. WOLFSON ET Y. LAMDAN, «TRANSFORMATION INVARIANT INDEXING», GEOMETRIC INVARIANCE IN COMPUTER VISION, EDITED BY J.L. MUNDY AND A. ZISSERMAN, THE MIT PRESS, P. 335-353, CAMBRIDGE, MASSACHUSETTS, 1992.

[ZHANG 1997] Z. ZHANG, «PARAMETER ESTIMATION TECHNIQUES : A TUTORIAL WITH APPLICATION TO CONIC FITTING», IMAGE AND VISION COMPUTING, VOL. 15, P. 59-76, ELSEVIER SCIENCE PUBLISHERS B.V., 1997.

[ZHOU 1992] J. ZHOU, «CONTRIBUTION AUX METHODES D'ETALONNAGE DES CAPTEURS D'IMAGES», THESE DE DOCTORAT, ECOLE CENTRALE DE PARIS, ISMCM, OCTOBRE, 1992.

[ZHU 1995] Y. ZHU, L.D. SENEVIRATNE ET S.W.E. EARLES, «A NEW STRUCTURE OF INVARIANTS FOR 3D POINT SETS FROM SINGLE VIEW», IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION, P. 1726-1731, NAGAYA, JAPON, 21-27 MAI, 1995.

ANNEXES

Annexe 1 : Calibration du mini robot

• Résumé

La calibration de la tige montée sur la tourelle du banc a été réalisée afin de valider le processus de calibration automatique de la caméra. Elle sert également à tester la précision de la localisation des objets lors de la reconnaissance. La tige montée sur le banc dispose de quatre degrés de liberté : deux translations et deux rotations. La méthode employée pour la calibration est la méthode d'optimisation non linéaire de Levenberg-Marquardt.

Afin de valider l'ensemble du système, une tâche de robotique est défini. Une fois l'objet modélisé reconnu, l'opérateur pointe une de ses faces à l'aide d'un réticule. Les données 3D sont alors définies. Celles-ci sont communiquées au module de commande de la tige pour que son extrémité touche le point visé.

Au vu de la configuration du système (4 ddl), un même point peut être atteint en utilisant plusieurs combinaisons des différents moteurs. Il faut imposer une valeur à l'un des moteurs pour obtenir une solution unique. La rotation θ_y est fixée à zéro par défaut, sinon elle est optimisée pour réaliser une tâche supplémentaire. Pour la calibration automatique de la caméra, cette tâche supplémentaire est de chercher un point 3d maximisant le champs visuel de la caméra.

Pour une démonstration réalisée au CEMIF, le choix retenu est de définir à partir de la surface connue d'un objet à désigner, un plan perpendiculaire à cette face passant par un point désigné par l'opérateur. Les données 3D du point sont connues, connaissant un vecteur normal à cette face, la direction de visée est déterminée. Il reste alors à inverser le modèle pour calculer les autres consignes moteurs.

• Notations

- R_o : repère objet

- R_t : repère associé à la position de départ (origine des moteurs) du banc de mesure
- R_m : repère mécanique de la tourelle où s'effectue la rotation
- (x_o, y_o, z_o) : coordonnées du point dans le repère objet
- (dx, dz, θ_x) : consignes des moteurs du banc de mesure
- ρ : longueur de la tige (suivant z)
- b : bras de levier (suivant y)
- a : bras de levier (suivant x)
- (t_{ox}, t_{oy}, t_{oz}) : paramètres de la translation de R_o vers R_t

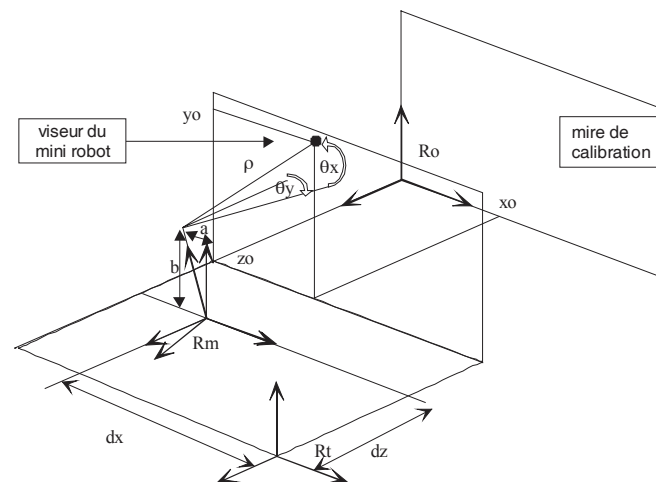


Figure A1.1 : les différents repères du système de mesures

• Mise en équations

Le passage de R_t vers R_o se fait par une translation pure, il s'exprime par :

$$\begin{cases} x_o = t_{ox} + x_t \\ y_o = t_{oy} + y_t \\ z_o = t_{oz} + z_t \end{cases}$$

Les coordonnées du repère mécanique R_m s'expriment dans le repère R_t par :

$$\begin{cases} x_i = d_x + x_m \\ y_i = y_m \\ z_i = d_z + z_m \end{cases}$$

Les coordonnées de l'extrémité de la tige s'expriment dans le repère mécanique par :

$$\begin{pmatrix} x_m \\ y_m \\ z_m \\ 1 \end{pmatrix} = \begin{pmatrix} c\theta_y & s\theta_x \cdot s\theta_y & c\theta_x \cdot s\theta_y & 0 \\ 0 & c\theta_x & -s\theta_x & 0 \\ -s\theta_y & s\theta_x \cdot c\theta_y & c\theta_x \cdot c\theta_y & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & -\rho \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Soit finalement les coordonnées dans le repère objet en fonction des autres paramètres :

$$\begin{cases} x_o = t_{ox} + d_x + a \cdot c\theta_y + b \cdot s\theta_x \cdot s\theta_y - \rho \cdot c\theta_x \cdot s\theta_y \\ y_o = t_{oy} + \rho \cdot s\theta_x + b \cdot c\theta_x \\ z_o = t_{oz} + d_z - \rho \cdot c\theta_x \cdot c\theta_y + b \cdot s\theta_x \cdot c\theta_y - a \cdot s\theta_y \end{cases}$$

Cette relation permet de définir le critère quadratique suivant :

$$C = \sum_{i=1}^n (\hat{x}_{oi} - x_{oi})^2 + (\hat{y}_{oi} - y_{oi})^2 + (\hat{z}_{oi} - z_{oi})^2$$

où n est le nombre de points de calibration.

La méthode d'optimisation Levenberg-Marquardt minimise ce critère, avec le vecteur de paramètres suivant :

$$p = (\rho, a, b, t_{ox}, t_{oy}, t_{oz})^T$$

Le vecteur de mesure correspond aux coordonnées 3D des points dans le repère objet ainsi que les consignes moteurs correspondantes :

$$m = (x_o, y_o, z_o, d_x, d_z, \theta_x, \theta_y)^T$$

Les paramètres sont déterminés, puis une inversion du modèle (2 solutions) permet de définir les consignes à appliquer aux moteurs lorsque les coordonnées 3D d'un point lui sont fournies dans le repère objet, il vient :

$$\begin{cases} d_x = x_o - t_{ox} - a \cdot c\theta_y + b \cdot s(2r_1) \cdot s\theta_y + \rho \cdot s\theta_y \cdot c(2r_1) \\ d_z = z_o - t_{oz} + \rho \cdot c\theta_y \cdot c(2r_1) + a \cdot s\theta_y + b \cdot c\theta_y \cdot s(2r_1) \\ \theta_x = -2r_1 \end{cases}$$

Avec :

$$r_1 = \arctan \left(\frac{-\rho \cdot \sqrt{\rho^2 - y_o^2 + 2 \cdot y_o \cdot t_{oy} - t_{oy}^2 + b^2}}{y_o - t_{oy} + b} \right)$$

• Résultats de calibration

Les données expérimentales suivantes sont obtenues sur le banc en plaçant la tige à calibrer à la limite du contact avec les points de la mire. Les données 3D du point ainsi que les consignes moteurs sont stockées. Les coordonnées et les translations sont exprimées en mètres, les rotations en degrés. Les valeurs du banc sont directement employées et converties pour la calibration.

Paramètres de calibration
rho = 7.287348e-001
a = -2.456087e-002
b = 4.340690e-002
tox = 3.940609e-001
toy = 2.365562e-001
toz = 1.671921e+000
Nombre d'itérations : 61
Critère final = 1.493131e-005

Annexe 2 : La méthode des moindres carrées

Cette annexe concerne le traitement d'images détaillé dans le chapitre 3. Nous montrons l'utilisation de la méthode des moindres carrées pour quelques problèmes de recherches de paramètres. Le critère à minimiser est constitué d'une somme des carrées d'erreurs. Nous présentons le cas de la recherche des paramètres d'une droite passant par un nuage de points. Cette étape est nécessaire lors du passage d'un ensemble de point alignés trouvé par la méthode de Hough vers les paramètres du segment recherché. Un autre cas est la recherche d'un point minimisant l'écart entre celui-ci et un groupe de segments (sommet dans le modèle 2D).

• Calcul des paramètres d'une droite

Soient (x_i, y_i) un ensemble de n points (nuage), on souhaite obtenir les paramètres de la droite passant par ces points. Le critère à optimiser est la somme des carrées des distances entre la droite et l'ensemble des points (Figure A2.1) :

Figure A2.1: Droite passant par un nuage de points

$$Crit = \sum_{i=1}^n r_i^2 = \sum_{i=1}^n s_i^2 \cos(\alpha)^2 = \sum_{i=1}^n s_i^2 \frac{1}{1 + \tan^2(\alpha)}$$

En considérant l'équation de la droite : $y = ax + b$, on obtient le critère à minimiser :

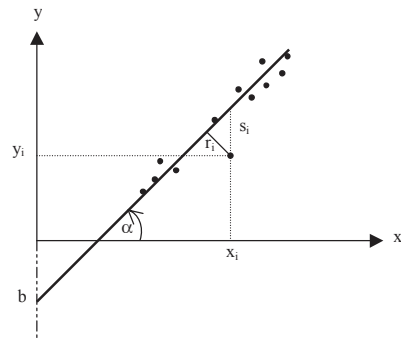
$$Crit = \sum_{i=1}^n \frac{(ax_i + b - y_i)^2}{1 + a^2} \quad (A2.1)$$

avec les dérivées nulles :

$$\frac{\partial Crit}{\partial a} = 0 \text{ et } \frac{\partial Crit}{\partial b} = 0 \quad (A2.2)$$

Le développement des équations (A2.2) donne :

$$b = \bar{y} - a \cdot \bar{x} \quad (A2.3)$$



où : $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$, $\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$ sont les moyennes des abscisses et des ordonnées, et

$$A \cdot a^2 + B \cdot a - A = 0 \quad (A2.4)$$

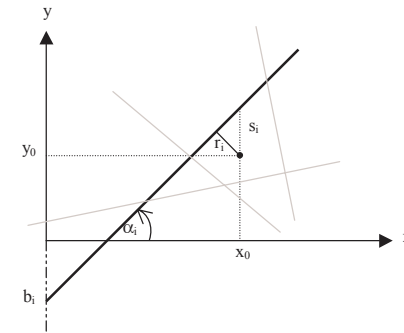
où : $A = \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) = n \cdot \text{Covariance}_{x,y}$ et

$$B = \sum_{i=1}^n (x_i - \bar{x})^2 - \sum_{i=1}^n (y_i - \bar{y})^2 = n \cdot (\text{Variance}_x - \text{Variance}_y)$$

La résolution des équations (A2.3) et (A2.4) donnent les paramètres de la droite optimale.

- Point à distance minimale d'un groupe de segments

Soient (a_i, b_i) les paramètres d'un ensemble de n droites, on souhaite obtenir le point (x, y) ayant la distance minimale de ces droites. Le critère à optimiser, comme le cas précédent, est la somme des carrées des distances



entre le point et l'ensemble des droites (Figure A2.2) :

Figure A2.2 : Un point à une distance minimale avec un ensemble de droites

$$Crit = \sum_{i=1}^n \frac{(a_i x + b_i - y)^2}{1 + a_i^2} \quad (A2.5)$$

La dérivation de ce critère et la résolution directe des équations obtenues est semblable au cas précédent.

Annexe 3 : Description de l'orientation

[KHALIL 1993]

Cette annexe concerne les méthodes de recalage étudiées dans le chapitre 5. Parmi les méthodes les plus utilisées pour décrire l'orientation d'un solide dans l'espace nous présentons : les cosinus directeurs, les trois angles RTL, le vecteur de rotation et les quaternions.

- Rappels

1. Multiplication des transformations homogènes :

Soit R_i un repère ayant subi des transformations consécutives connues $T_{k=1..n}$ qui l'amènent au repère R_j . La transformation finale ${}^i T_j$ exprimant le repère R_j dans le repère R_i est calculée selon deux cas de figures :

- Si les transformations $T_{k=1..n}$ sont exprimées par rapport au repère courant $R_{k=1..n}$, ${}^i T_j$ s'obtient par multiplications à droite : ${}^i T_j = {}^i T_1 \cdot {}^1 T_2 \cdot \dots \cdot {}^{n-1} T_n$
- Si les transformations $T_{k=1..n}$ sont exprimées par rapport au repère d'origine R_n , ${}^i T_j$ s'obtient par multiplications à gauche : ${}^i T_j = {}^i T_n \cdot {}^n T_2 \cdot {}^2 T_1$

2. Pré-produit vectoriel :

Soient \vec{u} et \vec{v} deux vecteurs, leur produit vectoriel s'écrit :

$$\vec{u} \wedge \vec{v} = \begin{bmatrix} u_y v_z - u_z v_y \\ u_z v_x - u_x v_z \\ u_x v_y - u_y v_x \end{bmatrix}$$

Il peut s'écrire, pour faciliter le développement, sous forme matricielle :

$$\vec{u} \wedge \vec{v} = \begin{bmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = X(\vec{u}) \cdot \vec{v}$$

$X(\vec{u})$ est la matrice du pré-produit vectoriel de \vec{u} .

- Cosinus directeurs

Soit R une matrice orthogonale dans l'espace E^3 telle que :

$$R = \begin{bmatrix} \vec{i} & \vec{j} & \vec{k} \end{bmatrix} = \begin{bmatrix} i_x & j_x & k_x \\ i_y & j_y & k_y \\ i_z & j_z & k_z \end{bmatrix} \quad (A3.1)$$

Ses éléments représentent les cosinus directeurs d'orientation de trois vecteurs. Elle ne contient que trois paramètres indépendants, car le vecteur \vec{k} se déduit par un produit vectoriel des deux autres dont la norme est égale à 1 et le produit scalaire $\vec{i} \cdot \vec{j} = 0$. Cette représentation redondante n'est pas optimale d'où l'intérêt de chercher d'autres formules pour exprimer la rotation.

- Angles de RTL (Roulis-Tangage-Lacet)

Les angles de *Roulis - Tangage - Lacet* expriment l'orientation par trois rotations successives d'un repère autour de ses trois axes principaux (z, y puis x). Les trois rotations étant exprimées par rapport au repère courant, la rotation finale s'obtient par multiplications à droite :

$$R(\theta_z, \theta_y, \theta_x) = \begin{pmatrix} \cos(\theta_z) & -\sin(\theta_z) & 0 \\ \sin(\theta_z) & \cos(\theta_z) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos(\theta_y) & 0 & \sin(\theta_y) \\ 0 & 1 & 0 \\ -\sin(\theta_y) & 0 & \cos(\theta_y) \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_x) & -\sin(\theta_x) \\ 0 & \sin(\theta_x) & \cos(\theta_x) \end{pmatrix} \quad (A3.2)$$

θ_z , θ_y et θ_x sont les trois angles de rotations

d'où :

$$R = \begin{pmatrix} \cos(\theta_y)\cos(\theta_z) & \sin(\theta_x)\sin(\theta_y)\cos(\theta_z) - \cos(\theta_x)\sin(\theta_z) & \cos(\theta_x)\sin(\theta_y)\cos(\theta_z) + \sin(\theta_x)\sin(\theta_z) \\ \cos(\theta_y)\sin(\theta_z) & \sin(\theta_x)\sin(\theta_y)\sin(\theta_z) + \cos(\theta_x)\cos(\theta_z) & \cos(\theta_x)\sin(\theta_y)\sin(\theta_z) - \sin(\theta_x)\cos(\theta_z) \\ -\sin(\theta_y) & \sin(\theta_x)\cos(\theta_y) & \cos(\theta_x)\cos(\theta_y) \end{pmatrix}$$

L'avantage de ce formalisme est qu'il présente directement la commande d'un robot (la rotation autour de chaque axe signifie l'envoi direct d'une commande au moteur spécifique). L'inconvénient est que le produit des trois matrices dans l'équation (A3.2) n'est pas commutatif. L'ordre dans laquelle s'effectuent les trois rotations conditionne donc la valeur finale de la rotation. Un autre inconvénient majeur est que le problème inverse (trouver les trois angles à partir de R) présente plusieurs solutions et des positions singulières.

- Vecteur de rotation

Considérons une rotation $R(\vec{u}, \theta)$ autour d'un axe portant un vecteur unitaire $\vec{u} = [u_x \ u_y \ u_z]^T$ passant par l'origine d'un repère $R_i ([.]^T$ signifie transposé). Supposons que \vec{u} soit le vecteur unitaire suivant l'axe z d'un repère R_j dont l'origine est confondu avec R_i (Figure A3.1). R_j peut être obtenu à partir de R_i par deux rotations successives exprimées par la transformation suivante :

$${}^i T_j = R(\vec{z}, \alpha) R(\vec{x}, \beta) \quad (A3.3)$$

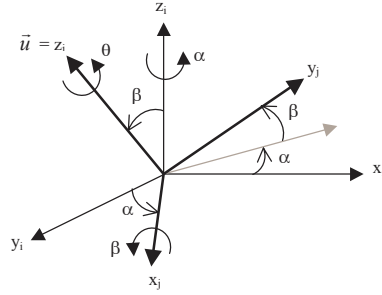


Figure A3.1 : R_j est obtenu de R_i par 2 rotations α autour \vec{z} et β autour \vec{x}

En développant la relation (A3.3), on obtient :

$$\vec{u} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} = \begin{bmatrix} \sin \alpha \sin \beta \\ -\cos \alpha \sin \beta \\ \cos \beta \end{bmatrix} \quad (A3.4)$$

Tourner autour de \vec{u} est équivalent à tourner autour de l'axe \vec{z} du repère R_j . Ceci revient à faire une transformation de R_i vers R_j , une rotation $R(\vec{z}, \theta)$ puis une transformation inverse vers R_i , on en déduit que :

$$R(\vec{u}, \theta) = R(\vec{z}, \alpha) R(\vec{x}, \beta) R(\vec{z}, \theta) R(\vec{x}, -\beta) R(\vec{z}, -\alpha)$$

Nous développons la formule précédente en tenant compte de la relation (A3.4) :

$$R(\vec{u}, \theta) = \begin{bmatrix} u_x^2(1-C\theta) + C\theta & u_x u_y(1-C\theta) - u_z S\theta & u_x u_z(1-C\theta) + u_y S\theta \\ u_x u_y(1-C\theta) + u_z S\theta & u_y^2(1-C\theta) + C\theta & u_y u_z(1-C\theta) - u_x S\theta \\ u_x u_z(1-C\theta) - u_y S\theta & u_y u_z(1-C\theta) + u_x S\theta & u_z^2(1-C\theta) + C\theta \end{bmatrix} \quad (A3.5)$$

avec : $C\theta = \cos(\theta)$ et $S\theta = \sin(\theta)$. En utilisant la matrice du pré-produit vectoriel de \vec{u} , nous obtenons la formule de Rodrigues :

$$R(\vec{u}, \theta) = I_3 C\theta + S\theta X(\vec{u}) + (1 - C\theta) \vec{u} \vec{u}^T \quad (A3.6 a)$$

$$R(\vec{u}, \theta) = I_3 + S\theta X(\vec{u}) + (1 - C\theta) X(\vec{u})^2 \quad (A3.6 b)$$

Ses deux équations sont équivalentes, où I_3 représente la matrice unité d'ordre trois.

Problème inverse :

Le problème inverse est celui de trouver le vecteur et l'angle de rotation correspondant à une matrice de rotation donnée. En faisant la somme des termes diagonaux dans les équations (A3.1) et (A3.5), on trouve :

$$C\theta = \frac{1}{2}(i_x + j_y + k_z - 1) \quad (A3.7)$$

A partir des termes restants, il vient :

$$\left. \begin{aligned} 2u_x S\theta &= j_z - k_y \\ 2u_y S\theta &= k_x - i_z \\ 2u_z S\theta &= i_y - j_x \end{aligned} \right\} \quad (A3.8)$$

d'où :

$$S\theta = \frac{1}{2} \sqrt{(j_z - k_y)^2 + (k_x - i_z)^2 + (i_y - j_x)^2} \quad (A3.9)$$

L'angle est déduit :

$$\theta = \text{Arctan}(S\theta / C\theta) \quad \text{avec } 0 \leq \theta \leq \pi \quad (A3.10)$$

En analysant les signes dans l'équation (A3.8) et en utilisant les expressions de (A3.1) et de (A3.5), on obtient :

$$\left. \begin{aligned} u_x &= \text{Signe}(j_z - k_y) \sqrt{(i_x - C\theta)/(1 - C\theta)} \\ u_y &= \text{Signe}(k_x - i_z) \sqrt{(j_y - C\theta)/(1 - C\theta)} \\ u_z &= \text{Signe}(i_y - j_x) \sqrt{(k_z - C\theta)/(1 - C\theta)} \end{aligned} \right\} \quad (A3.11)$$

Il existe donc deux solutions pour le problème inverse : $R(\vec{u}, \theta)$ et $R(-\vec{u}, -\theta)$. Un inconvénient remarqué est que la solution représente une singularité au voisinage de $\theta = 0$.

• Quaternions

Les *quaternions* (ou *paramètres d'Euler*, ou *paramètres d'Olindes-Rodrigues*) sont introduit essentiellement pour éliminer la singularité déjà évoqué. L'orientation est exprimée par quatre paramètres qui décrivent une rotation unique θ ($-\pi \leq \theta \leq +\pi$) autour d'un axe de vecteur unitaire \vec{u} . Ces paramètres sont définis par :

$$\left. \begin{aligned} \lambda_1 &= \cos(\theta/2) \\ \lambda_2 &= u_x \sin(\theta/2) \\ \lambda_3 &= u_y \sin(\theta/2) \\ \lambda_4 &= u_z \sin(\theta/2) \end{aligned} \right\} \quad (\text{A3.12})$$

La somme au carré de ces termes donne : $\lambda_1^2 + \lambda_2^2 + \lambda_3^2 + \lambda_4^2 = 1$ (A3.13)

En remplaçant \vec{u} et θ par les quaternions dans l'équation (A3.5), la matrice de rotation s'écrit :

$$R = \begin{bmatrix} 2(\lambda_1^2 + \lambda_2^2) - 1 & 2(\lambda_2\lambda_3 - \lambda_1\lambda_4) & 2(\lambda_2\lambda_4 + \lambda_1\lambda_3) \\ 2(\lambda_2\lambda_3 + \lambda_1\lambda_4) & 2(\lambda_1^2 + \lambda_3^2) - 1 & 2(\lambda_3\lambda_4 - \lambda_1\lambda_2) \\ 2(\lambda_2\lambda_4 - \lambda_1\lambda_3) & 2(\lambda_3\lambda_4 + \lambda_1\lambda_2) & 2(\lambda_1^2 + \lambda_4^2) - 1 \end{bmatrix} \quad (\text{A3.14})$$

Par analyse, semblable au cas du vecteur de rotation, des équations (A3.1) et (A3.14), la solution au problème inverse s'obtient par les formules suivants :

$$\left. \begin{aligned} \lambda_1 &= \frac{1}{2} \sqrt{i_x + j_y + k_z + 1} \\ \lambda_2 &= \frac{1}{2} \text{Signe}(j_z - k_y) \sqrt{+i_x - j_y - k_z + 1} \\ \lambda_3 &= \frac{1}{2} \text{Signe}(k_x - i_z) \sqrt{-i_x + j_y - k_z + 1} \\ \lambda_4 &= \frac{1}{2} \text{Signe}(i_y - j_x) \sqrt{-i_x - j_y + k_z + 1} \end{aligned} \right\} \quad (\text{A3.15})$$

L'intéressant est que les quaternions donnent une solution unique et ne présentent aucune position singulière.

Annexe 1 : L'Algorithme de Levenberg-Marquardt

L'Algorithme de Levenberg-Marquardt est une combinaison de l'algorithme de Gauss-Newton et la méthode du gradient [PRESS 1992]. Le critère à minimiser est généralement choisi comme une somme quadratique d'erreur e représentant la relation entre le vecteur de paramètres \mathbf{p} et les données d'entrée $\mathbf{d}_{i=1..N}$:

$$\text{Critère}(p) = \sum_{i=1}^N [e(d_i, p)]^2 \quad (\text{A4.1})$$

Le gradient de ce critère est donné par :

$$\text{Gradient}(p) = 2 \sum_{i=1}^N \left[\frac{\partial e(d_i, p)}{\partial p} \cdot e(d_i, p) \right] \quad (\text{A4.2})$$

et son hessien s'exprime par :

$$\text{Hessien}(p) = 2 \left\{ \sum_{i=1}^N \left[\frac{\partial e(d_i, p)}{\partial p} \cdot \frac{\partial e(d_i, p)}{\partial p'} \right] + \sum_{i=1}^N \left[\frac{\partial^2 e(d_i, p)}{\partial p \partial p'} \cdot e(d_i, p) \right] \right\}$$

Cette relation lourde à calculer à chaque itération, conduit à une approximation en négligeant le terme du seconde ordre. Ceci définit le hessien approché qui est le jacobien fois son transposé :

$$\text{Hessien}(p) \cong 2 \sum_{i=1}^N \left[\frac{\partial e(d_i, p)}{\partial p} \cdot \frac{\partial e(d_i, p)}{\partial p'} \right] \quad (\text{A4.3})$$

Pour converger vers l'optimum, une correction à effectuer sur les paramètres est calculée à l'aide de l'équation suivante :

$$\Delta p = -\text{Gradient}(p) \cdot (\text{Hessien}(p) + \lambda \cdot I)^{-1} \quad (\text{A4.4})$$

avec :

- \mathbf{p} le vecteur de paramètres à l'itération courante
- λ est un paramètre unidimensionnel permettant de régler le pas de convergence
- I est la matrice unité de dimension égale à celui du hessien

L'Algorithme, de mise en œuvre de la méthode de Levenberg-Marquardt, est le suivant :

1. Initialiser \mathbf{p} , calculer le critère initial et initialiser λ à une valeur assez faible (10^{-3})
2. Déterminer la correction $\Delta \mathbf{p}$ à faire à l'aide de l'équation (A4.4)
3. Avec le nouveau vecteur de paramètre, calculer le nouveau critère d'erreur :
 - s'il est inférieur au critère précédent, conserver \mathbf{p} et le critère puis, diviser λ par τ ($\tau=10$ ou 2 selon les auteurs)
 - sinon, reprendre le résultat précédent et multiplier λ par τ
4. Tester le seuil d'arrêt de l'algorithme : s'il n'est pas satisfait recommencer l'étape 2

L'algorithme s'arrête quand le critère d'erreur est suffisamment petit, ou lorsque la norme de $\Delta \mathbf{p}$ devient trop faible (c'est-à-dire qu'à chaque cycle, le vecteur de paramètres varie très peu).

La méthode d'optimisation de Levenberg-Marquardt combine celle de descente du gradient, quand on est éloigné de la solution, et celle de Newton, quand on est près du minimum du critère à minimiser. Elle est appliquée afin de permettre l'identification des paramètres du recalage (chapitre 4).

- Remarque

Si l'estimation des paramètres est trop éloignée de la solution, la méthode de Levenberg-Marquardt augmente le paramètre λ (pas de calcul) ce qui change de manière significative les valeurs des paramètres. En revanche, si l'estimation des paramètres se trouve au voisinage de la solution, λ est diminué.

Annexe 2 : Méthode géométrique des 3 segments

Cette annexe concerne la méthode de recalage à trois segments abordée au chapitre 4. Nous allons décrire les différentes étapes pour le calcul de la rotation R, la translation T est obtenue en fonction de R.

- Détermination de la rotation R

Afin de déterminer R, des repères intermédiaires ont été définis (cf. §4.4.1.2). La matrice de rotation R, devient une composition de passage entre ces repères (Figure 4.24 et équation 4.16) :

$$R = R_{oc1} R_{c1m1} R_{m1m}$$

Détermination de R_{oc1} :

Le modèle géométrique inverse de la caméra permet d'obtenir le rayon visuel passant par un point p sur l'image (cf. 2.2.1.5). Les deux points extrêmes $p_{j=1,2}$ de L_1 permettent de définir deux rayons visuels \vec{r}_1 et \vec{r}_2 à l'aide de l'équation (2.6). Le rayon visuel \vec{r}_1 définit l'axe \vec{x} du repère (Rc1). La normale \vec{N}_1 au plan d'interprétation contenant L_1 mais aussi L_1 est obtenue par :

$$\vec{N}_1 = \vec{r}_1 \wedge \vec{r}_2$$

Cette normale définit l'axe \vec{y} du repère (Rc1). L'axe \vec{z} de (Rc1) est défini par le produit vectoriel de \vec{x} et \vec{y} . Les coordonnées de $(\vec{x}, \vec{y}, \vec{z})$ forment les colonnes de la matrices R_{oc1} ou les lignes de la matrice R_{co1} :

$$R_{oc1} = \begin{pmatrix} (\vec{r}_1)^* & (\vec{N}_1)^* & (\vec{r}_1 \wedge \vec{N}_1)^* \end{pmatrix} \quad \text{où } (\cdot)^* \text{ signifie vecteur normé}$$

Pour la détermination de R_{oc1} , il suffit d'utiliser l'image de L_1 .

Détermination de R_{mm1} :

Le repère (Rm) étant lié à l'objet et étant le même que (Ro) à l'étape initiale, les coordonnées de L_1 par rapport à (Ro) à l'étape initiale restent les mêmes par rapport à (Rm) à chaque instant. Ainsi les segments L_1 sont connus par rapport à (Rm). On définit (Rm1) tel que son axe \vec{x} soit porté par L_1 , son axe \vec{z} est donné par le produit vectoriel de \vec{V}_1 et \vec{V}_2 . L'axe \vec{y} est donné par le produit vectoriel de \vec{z} et \vec{x} . Les coordonnées de $(\vec{x}, \vec{y}, \vec{z})$ forment les colonnes de la matrices R_{mm1} ou les lignes de la matrice R_{m1m} :

$$R_{mm1} = \left(\vec{v}_1^* \left((\vec{v}_1 \wedge \vec{v}_2) \wedge \vec{v}_1^* \right) \left(\vec{v}_1 \wedge \vec{v}_2 \right)^* \right) \text{ où } (\cdot)^* \text{ signifie vecteur normé}$$

Pour déterminer Rmm1, il suffit donc d'utiliser L₁ et L₂ exprimés dans (Rm).

Détermination de Rc1m1 :

L'axe \vec{x} du repère (Rm1) est porté par L₁, donc se trouve dans le plan d'interprétation de I₁ et l'axe \vec{x} du repère (Rc1) est également dans ce même plan. De même l'axe \vec{y} du repère (Rc1) est perpendiculaire à ce plan (Figure 4.24). Donc la matrice Rc1m1 ne dépend plus que de deux rotations, une rotation (\vec{y}, β) pour que les deux axes des \vec{x} soient parallèles et ensuite une rotation (\vec{x}, α) pour que les deux axes \vec{z} soient colinéaires. Une fois ces deux correspondances faites, les deux axes des \vec{y} deviennent automatiquement parallèles. Le principe étant acquis, l'objectif est de trouver α et β s'ils existent.

Pour un segment 3D quelconque L_i, auquel on associe son projeté l_i et la normale \vec{N}_i au plan d'interprétation, on a :

$$\vec{N}_{i(Rc1)} \cdot \vec{V}_{i(Rc1)} = 0 \quad (i = 1 \text{ à } 3)$$

Connaissant \vec{N}_i par rapport à (Ro) et les \vec{L}_i par rapport à (Rm), nous appliquons les différentes matrices de passage :

$$[Rc1o \vec{N}_{i(Ro)}] \cdot [Rc1m1 Rm1m \vec{V}_{i(Rm)}] = 0 \quad (A5.1)$$

Le développement de l'équation (A5.1) aboutit à un polynôme de degré 8, auquel est appliqué une méthode numérique basée sur le calcul de valeurs propres de la matrice "compagnon" construite à partir du polynôme [N'ZI 1995]. Les angles α et β qui caractérisent la matrice Rc1m1 sont obtenus. La rotation R subie par l'objet est ensuite calculée par l'équation (4.16).

• Détermination de la translation T

Nous allons exprimer les équations faisant intervenir T dans (Rc) avant de passer à (Ro). L'expression de tout point M_{i(Rm)}, dans (Rc) est donnée par :

$$M_{iRc} = T + RcmM_{iRm} \quad (A5.2)$$

Comme le vecteur \vec{CM} est dans le plan d'interprétation :

$$\vec{N}_{iRc} \cdot \vec{CM}_{iRc} = 0 \quad (A5.3)$$

En remplaçant M_{iRc} donné par (A5.2), l'équation (A5.3) devient :

$$\vec{N}_{iRc} \cdot (T + RcmM_{iRc}) = 0 \quad (A5.4)$$

Pour exprimer tous les vecteurs dans (Ro), il faut tenir compte des coordonnées de C par rapport à (Ro). Le modèle externe de calibration de la caméra Toc donne cette information, (A5.4) devient :

$$\vec{N}_{iRo} \cdot (T + RomM_{iRo}) = 0$$

Trois vecteurs permettent de trouver les trois équations nécessaires pour calculer (a,b,c) les paramètres de translation.

Dhome [DHOMÉ 1989] travaille avec les modèles interne et externe de la caméra séparément. Or, le fait de séparer les deux modèles peut présenter un problème de stabilité lorsque la méthode de calibration donne directement les paramètres globaux [LOUKIL 1993]. La méthode que nous utilisons consiste à travailler avec le modèle global pour nous libérer de ce problème d'instabilité.

Résumé

Notre laboratoire travaille sur la conception et le développement de Modules de Contrôle et d'Interface pour la Téléopération (MCIT). Le but de MCIT est de fournir à l'opérateur une aide pour la perception et pour la commande du site téléopéré. L'aide visuelle consiste en la mise à jour et la superposition de la BD3D sur l'image vidéo. Afin d'automatiser cette aide, un système de reconnaissance de polyèdres à partir d'une image de luminance a été développé et intégré à MCIT dans le cadre de cette thèse. Ce système est constitué d'un module de traitement d'images et d'un module d'appariement 2D/3D.

Le 1^{er} module est basé sur la modélisation orientée objet. La transformée de Hough, dont une amélioration est apportée, est utilisée pour extraire les segments de droite de l'image. L'organisation perceptive est appliquée pour trouver un modèle 2D de l'image.

Le 2nd module est constitué de deux étapes. La 1^{ère} étape concerne la prédiction d'hypothèses, elle utilise 2 méthodes d'appariement : la méthode des graphes qui donne un nombre d'hypothèses très réduit grâce à l'utilisation des invariants topologiques et projectifs mais, elle échoue en présence de défauts du traitement d'images. Dans ce cas, nous appliquons la méthode du hachage géométrique qui donne toujours une solution. Deux méthodes d'extraction de graphes d'aspects applicables aux polyèdres ont été également développées. La première est destinée à l'appariement par graphes, la seconde est utilisée par le hachage géométrique.

La 2^{ème} étape concerne la vérification de l'appariement, nous avons mis en œuvre des méthodes existantes de recalage et avons développé une méthode hybride qui donne une meilleure précision. Le développement de la calibration automatique de la caméra à l'aide d'un robot a permis également d'augmenter la précision et l'autonomie du système.

Mots clés : reconnaissance d'objets, polyèdres, traitement d'images, transformée de Hough, organisation perceptive, calibration automatique, graphe d'aspects, appariement, hachage géométrique, recalage.

Thesis Title: Polyhedral Object Recognition from a Single Video Image for Telerobotics

Abstract

Our laboratory is developing an interface system for Telerobotics MCIT. The aim is to provide visual aid and control of the site for the operator. The visual aid consists in updating and superimposing a 3DDB to a video image. In this thesis, a polyhedra recognition system from a single 2D image has been developed into MCIT to solve the last problem.

Image processing software based on object oriented library has been developed, within which an improvement of the Hough transform was achieved to better extract image line segments. Perceptual organization is also used to provide a 2D model of the image.

Two matching methods have been applied: the graph method which gives a minimal number of hypotheses using projective invariants but, it fails when the image processing is of poor quality. In this case, we apply the geometric hashing method, which always provides a solution.

We have developed two aspect graph extraction methods applicable on polyhedra. Each one is used by one of the above-mentioned matching methods.

For object localization, we apply a hybrid technique that makes use of three well-known reconstruction methods, which gives a better precision. The automatic camera calibration using a 4dof robot has been developed to reduce system errors.

Key words: object recognition, polyhedra, image processing, Hough transform, perceptual organization, automatic camera calibration, aspect graph, matching, geometric hashing, reconstruction.

Discipline : Sciences de l'Ingénieur

Centre d'Etudes en Mécanique de l'Ile de France (CEMIF) – Systèmes Complexes
40, rue de Pelvoux – CE1455 Courcouronnes – 91020 Evry cedex – France