

N° d'ordre :

**UNIVERSITE EVRY VAL D'ESSONNE**

# **THÈSE**

*pour obtenir le titre de*

**Docteur de l'Université Evry Val d'Essonne**

**Spécialité:**

**Sciences de l'Ingénieur**

*Présenté par*

**Fakhr-Eddine ABABSA**

**Recalage automatique d'objets 3D de forme libre en  
utilisant la clinométrie**

*Soutenue le 16 décembre 2002 devant le jury composé de*

M. P. Fuchs	MA, HDR, ENSMP Paris	Président
M. P. Even	Professeur, UHP de Nancy	Rapporteur
M. M. Devy	IR, HDR, LAAS Toulouse	Rapporteur
M. E. Colle	Professeur, Université d'Evry	Rapporteur
M. M. Mallem	Professeur, Université d'Evry	Examineur
M. D. Roussel	MCF, IIE CNAM	Examineur



# Remerciements

Je tiens à exprimer mes remerciements et ma vive gratitude :

A Messieurs David Roussel et Malik Mallem qui m'ont permis, grâce à leur confiance et leur soutien précieux, de réaliser et surtout de mener à terme ce travail.

A Monsieur Florent Chavand, directeur du Laboratoire Systèmes Complexes, pour m'avoir accueilli dans son laboratoire dans des conditions plus que favorables à une recherche fructueuse.

A Monsieur Philippe Fuchs qui me fait l'honneur de présider ce jury de thèse.

A Messieurs Philippe Even, Michel Devy et Etienne Colle qui ont accepté d'être les rapporteurs de ce mémoire, je les remercie tout particulièrement pour l'attention et le temps qu'ils y ont consacrés.

J'exprime également toute ma sympathie et ma gratitude à tous les doctorants du LSC pour l'ambiance agréable qu'ils ont su créer.

Enfin, je voudrais remercier ma famille et en particulier ma mère qui m'a donné l'éducation sans laquelle je n'en serais pas là aujourd'hui.

# Table des matières

<b>Introduction Générale</b>	<b>1</b>
<b>1 Etat de l'art sur les systèmes de reconnaissance d'objets de forme libre</b>	<b>5</b>
1.1 Définitions et propriétés d'un objet de forme libre.....	6
1.2 Représentations géométriques d'objets 3D de forme libre.....	7
1.2.1 Surfaces paramétriques.....	7
1.2.2 Surfaces implicites.....	8
1.2.3 Superquadriques.....	10
1.2.4 Cylindres généralisés.....	11
1.2.5 Polygones maillés.....	12
1.3 Systèmes de reconnaissance d'objets de forme libre.....	13
1.3.1 Méthodes basées sur les images d'intensité.....	13
1.3.2 Méthodes basées sur les images de profondeur.....	20
1.4 Conclusion.....	29
<b>2. Outils et techniques pour la reconnaissance et le recalage d'objets 3D de forme libre</b>	<b>31</b>
2.1 Problème de la reconnaissance d'objets.....	32
2.1.1 Définition.....	32
2.1.2 Difficultés.....	32
2.1.3 Approches.....	33
2.1.4 Critères de performances.....	34
2.2 Acquisition de la scène 3D.....	34
2.2.1 Vision active.....	37
2.2.2 Vision passive.....	39
2.3 Primitives.....	42
2.3.1 Primitives de segmentation.....	43
2.3.2 Primitives de modélisation.....	49
2.4 Invariance géométrique dans la reconnaissance d'objets 3D.....	50
2.4.1 Définition.....	50
2.4.2 Types d'invariants.....	51
2.5 Modes de représentation des objets 3D.....	54
2.5.1 Représentation centrée sur l'objet.....	54
2.5.2 Représentation centrée sur l'observateur.....	55
2.6 Techniques de Mise en correspondance pour la reconnaissance d'objets 3D.....	55
2.6.1 Arbres de recherche ou d'interprétation.....	56
2.6.2 Plus proche voisin itéré (ICP : "Iterative Closest Point").....	58
2.6.3 Transformée de Hough généralisée.....	59
2.6.4 Prédiction/vérification d'hypothèses.....	60
2.6.5 Hachage géométrique (HG).....	61

2.6.6	Indexation d'invariants géométriques .....	68
2.6.7	Isomorphisme de graphes .....	69
2.6.8	Graphe d'aspects .....	76
2.7	Recalage .....	76
2.8	Description du système de reconnaissance élaboré .....	77
2.9	Conclusion .....	79

### **3 Approche photoclinométrique** **81**

3.1	Description générale du système .....	82
3.1.1	Hypothèses de base .....	82
3.1.2	Modèle de la réflexion diffuse .....	83
3.1.3	Modèle géométrique de l'objet de forme libre .....	84
3.1.4	Principe de la méthode photoclinométrique .....	85
3.2	Construction de la base de données .....	87
3.2.1	Suréchantillonnage du maillage .....	87
3.2.2	Echantillonnage de la sphère de vue .....	89
3.2.3	Construction de la table d'aspects photométriques .....	90
3.2.4	Indexation des sommets du maillage .....	92
3.3	Appariement automatique image/modèle .....	98
3.4	Recalage 3D et vérification d'hypothèses .....	105
3.4.1	Définition du problème .....	106
3.4.2	Algorithme d'itération orthogonale (IO) .....	108
3.5	Evaluation des performances .....	114
3.5.1	Validité du modèle photométrique .....	115
3.5.2	Images de test .....	115
3.5.3	Base de données .....	116
3.5.4	Système de vision Appariement .....	117
3.5.5	Recalage 3D .....	120
3.5.6	Vérification d'hypothèses .....	121
3.6	Conclusion .....	121

### **4 Approche Clinométrique** **123**

4.1	Principe de la méthode .....	124
4.2	Système de vision .....	126
4.2.1	Calibration du système de vision .....	127
4.2.2	Extraction de la grille image .....	130
4.2.3	Reconstruction des points 3D .....	137
4.2.4	Estimation de la précision de la reconstruction 3D .....	138
4.3	Construction de la base de données .....	139
4.3.1	Numérisation de l'objet de forme libre .....	139
4.3.2	Construction des patches de surface .....	140
4.3.3	Table d'indexation .....	141
4.3.4	Résultats expérimentaux .....	145
4.4	Mise en correspondance .....	146
4.5	Localisation 3D .....	149

4.5.1	Recalage 3D .....	149
4.5.2	Recalage 3D fin .....	154
4.5.3	Résultats expérimentaux .....	155
4.6	Vérification d'hypothèses .....	157
4.7	Evaluation des performances .....	157
4.7.1	Système de vision .....	157
4.7.2	Construction de la base de données .....	158
4.7.3	Indexation .....	158
4.7.4	Vérification d'hypothèses .....	159
4.8	Conclusion .....	159
<b>Conclusion Générale</b>		<b>161</b>
<b>Bibliographie</b>		<b>165</b>
<b>Annexe A</b>	<b>Calibration de la caméra</b>	
<b>Annexe B</b>	<b>Algorithme de l'Itération Orthogonale (IO)</b>	
<b>Annexe C</b>	<b>Géométrie différentielle</b>	
<b>Annexe D</b>	<b>Représentation de la matrice de rotation</b>	
<b>Annexe E</b>	<b>Méthode d'optimisation de Levenberg-Marquardt</b>	

# Introduction Générale

Au sein de l'équipe Réalité Virtuelle et Vision Artificielle du LSC (Laboratoire Systèmes Complexes), l'un des thèmes fédérateurs est celui de la reconnaissance d'objets 3D pour la téléopération. Ainsi au cours des travaux précédents, un système de reconnaissance d'objets polyédriques à partir d'une image vidéo fut développé [Shaheen, 1999]. Nos travaux s'inscrivent dans le cadre général de la reconnaissance automatique d'objets 3D de forme libre. Nous nous sommes intéressés en particulier au problème du recalage 3D de ce type d'objets dans le contexte de la vision monoculaire associée à une lumière non structurée ainsi que celui de la stéréovision active.

La reconnaissance d'objets tridimensionnels de forme libre est un problème qui est loin d'être résolu en vision par ordinateur. Plusieurs méthodes ont été développées ces dernières années, elles se basent généralement sur la connaissance à priori du modèle de l'objet à identifier. Les modèles utilisés sont donc construits à partir de primitives géométriques (points, segments de droite, courbes, etc.). L'identification d'un objet dans la scène à partir d'une image d'intensité et/ou de profondeur est obtenue grâce à des appariements effectués entre des primitives 2D ou 3D extraites de ces images et des primitives 3D du modèle de l'objet. Pour cela, plusieurs techniques de mise en correspondance peuvent être utilisées, on peut citer par exemple les méthodes de prédiction-vérification, les méthodes du hachage géométrique, ou bien les méthodes d'indexation par invariant géométrique.

Le but recherché dans le cadre de notre étude est de démontrer que l'utilisation de la clinométrie<sup>1</sup> de l'objet de forme libre, sous certaines conditions, est suffisante pour réaliser un système de recalage fiable et précis. La finalité de ce système est de déterminer la pose (i.e. la position + l'orientation) des objets dans la scène pour des applications postérieures. Parmi ces applications, nous pouvons citer : la localisation, la poursuite, l'évitement d'obstacles, la préhension pour la robotique et l'assistance visuelle de l'opérateur pour la téléopération et la réalité augmentée.

Notre but étant de développer un système de recalage simple. Il est donc nécessaire de travailler à tous les niveaux du système (extraction/mise en correspondance/recalage) en apportant des contributions particulières sur plusieurs points de la chaîne de traitement global. Ainsi notre intérêt s'est particulièrement porté sur la construction de la base de données qui décrit les différentes configurations de l'objet de forme libre afin de permettre une reconnaissance rapide et correcte car l'amélioration du temps de reconnaissance reste la préoccupation majeure de ce type de système.

---

<sup>1</sup> Par clinomètre, nous entendons la variation de la normale à la surface de l'objet de forme libre.

Ce mémoire s'articule autour des différentes étapes qui caractérisent le recalage 3D d'objets de forme libre par clinométrie. Ainsi, pour chacun des niveaux de traitement, une explication détaillée des méthodes élaborées ainsi qu'une interprétation des résultats expérimentaux obtenus seront proposées. De même, pour chacun des choix effectués durant cette étude, nous donnerons une justification en fonction de l'objectif que nous nous sommes fixé mais également en comparaison des travaux déjà existants. Ainsi, nous dégagerons les diverses contributions que nous avons pu apporter à certains des niveaux du traitement global.

Le premier chapitre est consacré à la présentation et l'analyse des différentes techniques qui ont été développées ces dernières années dans les domaines de la représentation et la reconnaissance d'objets de forme libre. Notre objectif est de classer ces différentes méthodes et de mettre en évidence les particularités ainsi que les avantages et les inconvénients de chacune d'elles par rapport aux autres. Ceci nous a permis, par la suite, d'expliquer et de mettre en avant les facteurs qui ont guidé nos choix.

Le second chapitre est dédié aux outils et techniques de traitements utilisés dans les différents niveaux d'un système de reconnaissance d'objets 3D. Ainsi, nous détaillons les systèmes de vision utilisés pour acquérir la scène, le choix des primitives de base et des méthodes permettant de les extraire. Nous présentons également les méthodes de mises en correspondance les plus utilisées en insistant sur celles qui exploitent l'invariance géométrique. Enfin, nous donnons l'architecture globale de notre système de recalage 3D qui résume les deux solutions que nous avons élaborées, en l'occurrence l'approche photoclinométrique<sup>2</sup> et l'approche clinométrique. Nous situons, à chaque fois, nos contributions par rapport aux solutions existantes.

Dans le troisième chapitre, la solution photoclinométrique est présentée en détail. Cette solution exploite une seule image de luminance de l'objet à partir de laquelle elle extrait un invariant projectif de type photométrique (non géométrique) qu'elle utilise pour apparier des zones de l'image de l'objet avec celles du modèle. Elle est séparée en deux parties distinctes, qui sont la construction de la base de donnée (hors-ligne) et la mise en correspondance (en ligne). Nous proposons une méthode originale basée sur ce que nous avons appelé les aspects photométriques pour construire une base de données discriminante du modèle 3D de l'objet. La seconde partie concerne la méthode de mise en correspondance. Nous avons élaboré une méthode qui s'inspire de la technique de prédiction/vérification d'hypothèses. Notre méthode nous permet d'effectuer des appariements très rapidement grâce à des calculs simples. Une fois les appariements image/modèle effectués, nous formulons le problème de recalage 3D au sens des moindres carrés avec contraintes que nous résolvons en utilisant l'algorithme de l'itération orthogonale (IO). Nous présentons les résultats expérimentaux obtenus par l'utilisation de l'approche photoclinométrique sur des images de synthèse et terminons par une évaluation qui permet d'examiner les performances et les limites de cette approche.

---

<sup>2</sup> La photoclinométrie est l'analyse de la variation de la normale à la surface de l'objet d'après la photométrie.

Le quatrième chapitre décrit la solution clinométrique que nous avons élaborée pour remédier aux défauts de l'approche photoclométrique. Cette solution utilise un capteur stéréo actif (caméra CCD + projecteur d'une grille laser) pour extraire des points 3D d'une région de la surface de l'objet de forme libre. A partir de ces points 3D, nous construisons un invariant géométrique pour identifier, dans une table d'indexation du modèle, la région de l'objet ainsi reconstruite. L'indexation permet d'établir des appariements entre des zones de l'objet détecté dans la scène avec celles du modèle de la base de données. Nous formulons d'une manière différente le problème du recalage 3D car dans ce cas nous utilisons des appariements de type 3D/3D. Nous présentons les avantages que nous obtenons par l'utilisation de la méthode clinométrique ainsi que les résultats expérimentaux obtenus sur des images réelles.

Nous résumons, dans la conclusion générale, pour les deux approches élaborées, la démarche suivie, les méthodes retenues et la contribution apportée. Enfin, une analyse critique sur le comportement des deux systèmes nous permet de conclure sur les améliorations possibles et sur les développements futurs de ces systèmes.



# Chapitre 1

## Etat de l'art sur les systèmes de reconnaissance d'objets de forme libre

La reconnaissance d'objets tridimensionnels de forme libre est un problème qui est loin d'être résolu en vision par ordinateur. Cependant la progression spectaculaire des capacités de calculs et de stockage ont rendu possible l'émergence de nouvelles techniques de modélisation et de visualisation d'objets 3D très complexes composés de milliers de polygones. Ces nouveaux systèmes très performants qui permettent d'élaborer des modèles photo-réalistes fiables d'objets 3D constitués de surfaces de forme quelconque ont ouvert à la communauté de la vision par ordinateur de nouvelles voies très prometteuses dans le domaine de la reconnaissance et la localisation des objets de formes libres. En effet les techniques de reconnaissance classiques qui utilisent des représentations géométriques simples pour modéliser les objets 3D (tel que la représentation polygonale) sont limitées et n'apportent pas une solution satisfaisante à ce type de problème.

Selon le type de données qu'elles exploitent, les techniques de reconnaissance d'objets de forme libre sont classées en deux grandes catégories, d'une part les techniques qui exploitent les images d'intensité lumineuse (luminance) de l'objet et d'autre part les techniques qui exploitent les images de profondeur dites «range images».

Ce chapitre a pour but de présenter les travaux de recherches les plus récents effectués dans les domaines de la représentation et la reconnaissance d'objets de forme libre. Nous définissons dans un premier temps les concepts de base et les propriétés fréquemment utilisées pour caractériser ce type d'objets, ensuite nous décrivons les méthodes les plus utilisées pour les représenter géométriquement. Dans la deuxième partie de ce chapitre, nous présentons les différentes techniques élaborées pour reconnaître un objet de forme libre à partir d'images d'intensité et/ou de profondeur. Enfin, nous terminons par une conclusion dans laquelle nous donnons les grandes lignes des systèmes de reconnaissance que nous avons développés en précisant les contributions apportées.

## 1.1 Définitions et propriétés d'un objet de forme libre

La définition des objets et/ou surfaces de forme libre reste souvent subjective et intuitive plutôt que formelle et explicite. Généralement l'adjectif « forme libre » est utilisé pour caractériser les objets ayant des surfaces complexes et/ou non facilement identifiables tel que les surfaces planes et les quadriques. Une caractérisation approximativement équivalente mais plus concrète est fournie par [Besl, 1990] : Un objet de forme libre possède une normale à sa surface bien définie qui est continue presque partout excepté sur les sommets, les bords et les pics de l'objet». Cette définition a été adoptée par de nombreux chercheurs du domaine ([Stein et Medioni, 1992] , [Dorai et Jain, 1997]). On peut citer comme exemples typiques d'objets de forme libre les sculptures, le visage et les organes humains, les cartes topographiques, les carrosseries de voitures.

Divers *critères* pour la représentation des objets de forme libre ont été proposés. Ces critères permettent d'évaluer la pertinence d'une représentation donnée pour le problème de la reconnaissance des objets. Comme nous ne sommes pas conscients des représentations que nous utilisons pour reconnaître les objets autour de nous et que les études en biologie et en psychologie nous informent encore relativement peu sur ces sujets, les critères que nous définissons doivent être basés sur des considérations *géométriques* et *algorithmiques* reliés à notre compréhension de la tâche à accomplir, en l'occurrence la reconnaissance (et donc la discrimination) des objets. [Marr & Nishihara, 1976] ont proposé trois critères de base. Le premier est l'*accessibilité* qui stipule qu'une représentation doit être calculable efficacement (i.e rapidement à partir de ressources limitées) à partir des informations des images. Le second critère est la *portée/unicité*. La représentation doit s'appliquer au plus grand nombre d'objets possibles, être unique pour chaque objet (ou simplement *ambiguë*, un critère moins sévère qui permet qu'un objet ait plus d'une représentation mais elles doivent toutes être différentes de celles des autres objets) et être invariante aux paramètres d'acquisitions des images. En pratique, l'unicité parfaite est impossible car tout modèle implique une *approximation*. On doit donc parler du degré de respect de l'unicité. Celui-ci est généralement moins grand lorsque la représentation est plus abstraite, grossière ou qualitative. Enfin, le troisième critère est la *sensibilité/stabilité*. La représentation doit être robuste aux bruits de mesures et rendre explicite, les informations ou caractéristiques qui sont importantes pour la discrimination de la forme des objets.

De nombreux autres critères ont été proposés par différents auteurs. Par exemple l'*efficacité*, une représentation doit permettre une reconnaissance efficace, c'est à dire rendre explicite à un niveau d'abstraction suffisamment élevé les paramètres de forme de l'objet. Ce critère fait référence entre autres, à la possibilité d'indexage dans une base de modèles (appariement) et à la reconnaissance générique au moyen de modèles qualitatifs. De même, le critère de *localité* de la représentation peut s'avérer primordial pour les applications de reconnaissance d'objets en présence d'occultations.

Quoique ces critères soient très intuitifs, il s'avère très difficile de les respecter tous intégralement. On peut affirmer qu'à ce jour il n'existe pas de représentation générale

connue et surtout pas de caractéristiques que l'on peut extraire d'une représentation qui soient tout à fait générales et qui respectent parfaitement tous les critères énoncés. Les types de représentation et les caractéristiques proposés sont des compromis limités par notre compréhension encore imparfaite des objets 3D ainsi que du processus de la reconnaissance visuelle.

L'histoire des techniques de reconnaissance d'objets 3D montre une progression dans la complexité des modèles utilisés, ainsi des polyèdres on est passé aux quadriques pour finir aux formes libres. Le choix d'une représentation doit être accompagné de méthodes robustes permettant d'extraire à partir des modèles et de leurs images des primitives compatibles afin que la reconnaissance puisse être effectuée.

## 1.2 Représentations géométriques d'objets 3D de forme libre

Cette section décrit les modèles les plus couramment employés pour représenter la surface d'un objet de forme libre. Ces modèles sont dit *complets* car ils décrivent explicitement la totalité de la géométrie des objets. Ceci permet, connaissant les spécifications de la prise de vue, de générer des images de synthèse de l'objet quelque soit sa position dans la scène. Ces images de synthèse peuvent être ensuite utilisées pour tester des algorithmes de reconnaissance d'objets.

### 1.2.1 Surfaces paramétriques

Les surfaces paramétriques ont été largement utilisées dans le domaine de la Conception et de la Fabrication Assistée par Ordinateur (CFAO). Une surface paramétrique peut être décrite par l'expression suivante :

$$S(u,v) = \begin{cases} x=f(u,v) \\ y=g(u,v) \\ z=h(u,v) \end{cases} \quad (1.1)$$

Les fonctions  $f(u,v)$ ,  $g(u,v)$  et  $h(u,v)$  représentent les coordonnées de la surface, elles peuvent être des expressions *polynomiales* ou *rationnelles*. De plus, elles dépendent des deux paramètres  $u$  et  $v$  qui varient dans un espace paramétrique de dimension 2. On restreint les paramètres  $(u,v)$  à l'intervalle  $[0,1] \times [0,1]$ , sans perdre en généralité.

Les surfaces NURBS (Non-Uniform Rational B-Splines) représentent la formulation paramétrique la plus commune, elles sont définies par l'équation :

$$S(u,v) = \sum_i \sum_j B_{i,j}^h N_{i,k}(u) M_{i,l}(v) \quad (1.2)$$

ou  $N_{i,k}(u)$  et  $M_{i,l}(v)$  sont les fonctions de base B-splines d'ordre  $k$  et  $l$ , et  $B_{i,j}^h$  sont les coordonnées homogènes des points de contrôle de la surface.

Les surfaces NURBS ont la propriété de ne pas avoir leurs points de contrôles (les points sur lesquels on agit pour modifier la surface) situés sur la surface. De plus, chaque point de contrôle possède un "poids" qui permet de définir sa force d'attraction sur la surface. Les objets NURBS sont généralement représentés comme dans la figure (1.1a) où nous pouvons distinguer, d'une part les courbes qui mettent en évidence la surface et d'autre part, les lignes qui relient les points de contrôle. La figure (1.1b) illustre le rendu de la surface NURBS de la figure(1.1a).

Toutefois, la complexité de la mise en œuvre des surfaces paramétriques a limité leur utilisation en vision par ordinateur. En effet, il est généralement très difficile de modéliser une région quelconque de la surface de l'objet par une surface paramétrique. De plus, les points homogènes de contrôle ne sont effectivement pas facilement détectables en imagerie (problème de la projection). Cependant, les formes paramétriques peuvent être utilisées comme une spécification initiale du modèle de l'objet à partir de laquelle d'autres représentations (telle que les polygones maillés) sont générées pour être employées dans les systèmes de vision.

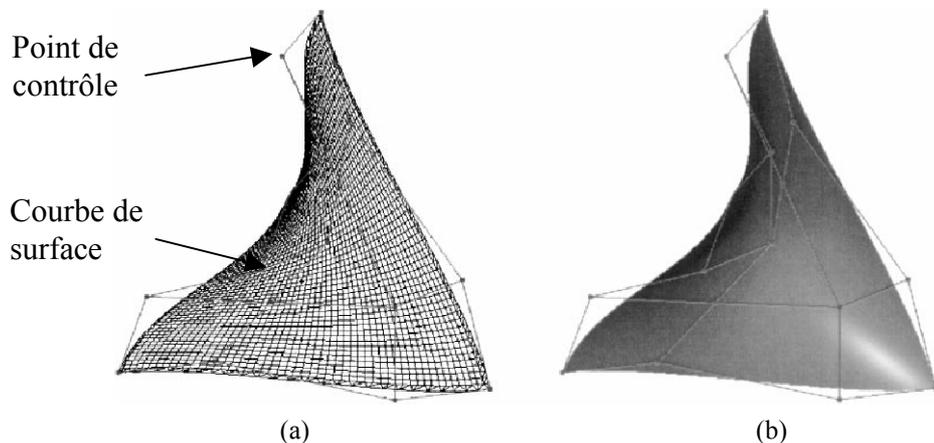


Figure 1.1 : Exemple d'une surface NURBS

### 1.2.2 Surfaces implicites

Bien que les surfaces implicites soient un modèle connu et étudié depuis longtemps par les mathématiciens, la modélisation d'objets de forme libre par surfaces implicites est un domaine très récent [Taubin, 1991].

De manière formelle, une surface implicite  $S$  est définie par une fonction implicite  $F: \mathcal{R}^3 \rightarrow \mathcal{R}$  qui associe une valeur scalaire à tout point de l'espace. La surface  $S$  est alors définie comme l'ensemble des points de valeur nulle, i.e :

$$S = \{M \in \mathcal{R}^3 / F(M) = 0\} \quad (1.3)$$

$S$  est appelée alors *surface zéro* ou *ensemble zéro*. La fonction  $F$  permet de séparer l'espace en deux composantes, l'intérieur et l'extérieur de la surface  $S$ . On obtient ainsi une classification d'un point  $M$  par rapport à  $S$  :

- si  $F(M) < 0$ ,  $M$  est à l'intérieur de la surface.
- si  $F(M) = 0$ ,  $M$  est sur la surface.
- si  $F(M) > 0$ ,  $M$  est à l'extérieur de la surface.

L'exemple le plus simple de surface implicite est la sphère de centre  $Q:(x_Q, y_Q, z_Q)$  et de rayon  $R$ , dont la fonction implicite est définie par :

$$F(M) = (x - x_Q)^2 + (y - y_Q)^2 + (z - z_Q)^2 - R^2 \quad (1.4)$$

Une propriété remarquable des surfaces implicites est qu'il est possible de combiner plusieurs fonctions implicites *indépendantes* pour générer une fonction implicite globale en utilisant les opérations ensemblistes classiques telle que l'union, l'intersection, la soustraction, etc.

Bien évidemment, la modélisation d'objets de forme libre ne peut se contenter de primitives sphériques uniquement. En effet, les recherches dans le domaine ont porté sur la définition de primitives plus complexes, en l'occurrence les *surfaces algébriques*. Ce sont une famille particulière de surfaces implicites pour laquelle la fonction  $F$  est un polynôme. L'intérêt pour la modélisation d'objets de forme libre est que ces fonctions fournissent des degrés supplémentaires pour contrôler la forme du modèle. La figure (1.2) illustre un objet dont la surface est définie par l'équation algébrique suivante :

$$F(x, y, z) = 2x^4 - 3x^2y^2 + 3y^4 - 3y^2z^2 - 2x^3z + 6z^4 - 1 = 0 \quad (1.5)$$

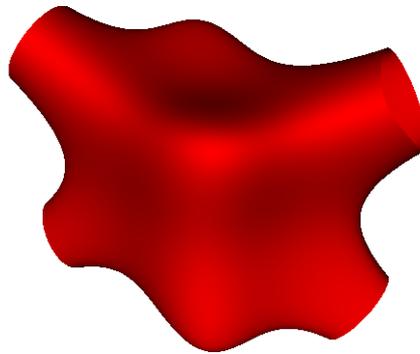


Figure 1.2 : Exemple d'une surface algébrique

Malgré ces degrés de liberté, la classe des formes représentables est limitée par la complexité des équations. C'est pourquoi d'autres surfaces algébriques se basent sur des polynômes par morceaux : le modèle des carreaux implicites de [Bajaj et al., 1995] par exemple permet de définir un morceau de fonction implicite à l'intérieur d'un polyèdre de contrôle. Mais cette formulation implique une continuité des fonctions entre deux polyèdres adjacents difficiles à contrôler ainsi qu'une topologie des formes résultantes contrainte par le réseau de polyèdres. De plus elle est coûteuse en temps de calcul.

### 1.2.3 Superquadriques

Les superquadriques sont une famille de primitives volumétriques qui a d'abord été développé pour la *synthèse d'objets* en infographie puis introduit en vision numérique par [Pentland, 1986]. Elles sont subdivisées en trois classes : les *superellipsoïdes*, les *superhyperboloïdes* et les *supertoroïdes*. Seul le premier de ces trois types a été utilisé en vision par ordinateur.

La forme paramétrique de la surface de la superellipsoïde est donnée par la fonction  $S(\eta, \omega)$  suivante :

$$S(\eta, \omega) = \begin{bmatrix} x(\eta, \omega) \\ y(\eta, \omega) \\ z(\eta, \omega) \end{bmatrix} = \begin{bmatrix} a_1 \cos^{\varepsilon_1}(\eta) \cos^{\varepsilon_2}(\omega) \\ a_2 \cos^{\varepsilon_1}(\eta) \sin^{\varepsilon_2}(\omega) \\ a_3 \sin^{\varepsilon_1}(\eta) \end{bmatrix} \quad -\frac{\pi}{2} \leq \eta \leq \frac{\pi}{2}, -\pi \leq \omega \leq \pi \quad (1.6)$$

Tant dis que sa forme explicite est définie comme suit :

$$S(x, y, z) = \left[ \left( \frac{x}{a_1} \right)^{\frac{2}{\varepsilon_2}} + \left( \frac{y}{a_2} \right)^{\frac{2}{\varepsilon_2}} \right]^{\frac{\varepsilon_2}{\varepsilon_1}} + \left( \frac{z}{a_3} \right)^{\frac{2}{\varepsilon_1}} - 1 = 0 \quad (1.7)$$

Généralement, la forme implicite est utilisée pour extraire les paramètres de la superellipsoïde à partir de données réelles mesurées, c'est à dire à ajuster un modèle de superellipsoïdes aux données.

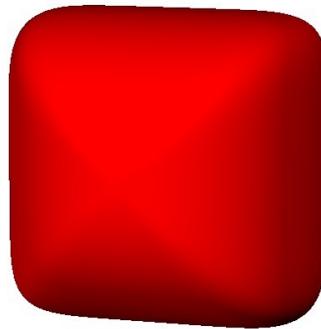


Figure 1.3 : Superellipsoïde ( $a_1=a_2=a_3=1$  et  $\varepsilon_1=1$   $\varepsilon_2=0.3$ )

Le processus d'ajustement consiste à déterminer la valeur des paramètres de la superellipsoïde qui minimiseront la somme du carré des distances entre les points des données et la surface de la superellipsoïde.

Dans le but de représenter une variété encore plus grande de formes libres, [Pentland, 1986] et [Solina et Bajcsy, 1990] ont montré qu'il suffisait pour cela d'introduire des

*déformations globales* sur les superellipsoïdes. Deux exemples de déformations souvent utilisées sont :

- l'amincissement linéaire « linear tapering » qui consiste à appliquer une transformation linéaire aux points de la surface de la superellipsoïde initiale afin de la déformer et ainsi générer une nouvelle superellipsoïde aminci (le taux d'amincissement dépend des paramètres de la transformation).
- le recourbement plan ou fléchissement « bending » : celui-ci consiste à transformer l'axe droit de la superellipsoïde aligné avec  $z$  par un arc de cercle de rayon de courbure  $R$ . Ceci aura pour effet de comprimer la partie intérieure de la superellipsoïde initiale et d'en allonger la partie extérieure tout en gardant fixe la longueur de l'axe  $z$ .

Il est à noter que des méthodes utilisant plutôt des *déformations locales* (i.e. potentiellement différentes à chaque point de la surface) ont été proposées récemment par plusieurs groupes de recherches en vision numérique ([Delingette et al., 1992] , [Shen et Yang, 1998]) en particulier pour la modélisation de visages humains. On parle alors de *modèles déformables*.

## 1.2.4 Cylindres généralisés

Un cylindre généralisé (CG) est un autre type de primitive volumétrique, il est défini par une forme plane appelée *section* « cross-section » qui est *balayée* le long d'une courbe 3D appelée *axe*. La forme de la section peut changer le long de l'axe et la fonction décrivant ces changements s'appelle règle de balayage « sweep rule ». Par exemple, un cylindre est un cylindre généralisé d'axe droit et de section circulaire constante. La figure (1.4) montre un exemple d'un objet de forme libre représenté par un cylindre généralisé.

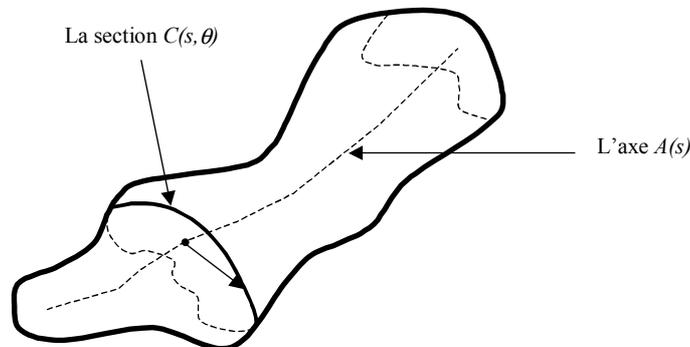


Figure 1.4 : Cylindre généralisé

La représentation paramétrique des points sur la surface d'un cylindre généralisé peut s'exprimer de façon simple, en fonction de la section, de l'axe et de la règle de balayage, par l'utilisation de repères appropriés. Ainsi, l'axe peut être décrit par la courbe 3D paramétrique  $A(s)=[x(s), y(s), z(s)]$ . Le repère local doit être défini pour représenter la section  $C(s, \theta)$  en chaque point le long de l'axe, son origine sera donnée par la position du

point sur l'axe, alors que son orientation sera déterminée par les vecteurs normal et binormal dans le repère de Frenet à ce point (figure 1.4).

Les CG sont particulièrement attractifs pour modéliser les objets complexes. En effet, [Marr et Nishihara, 1976] ont proposé une représentation *hiérarchique* basée sur les CG pour modéliser les formes articulées réelles comme les animaux ou les personnes où chaque partie de l'objet peut, à son tour être décomposée en un ensemble de sous-parties, de façon hiérarchique.

D'un autre côté, les objets pour lesquels il est difficile (voire impossible) de définir un axe dont les sections contiennent un seul contour fermé, sont difficilement décrits par les cylindres généralisés. De plus, la représentation générée est généralement loin d'être la meilleure en terme de design et de discrimination [Nevatia et Binford, 1977].

### 1.2.5 Polygones maillés

Un polygone maillé est un ensemble d'arêtes, de sommets et de polygones reliés de manière que les arêtes ne soient communes qu'à deux polygones au plus. Différentes méthodes ont été élaborées pour décrire un polygone maillé, la plus courante est celle qui utilise les pointeurs sur une liste de sommets. Ainsi, un objet  $O$  sera défini par deux listes ordonnées :

$$O = \langle P, V \rangle \quad (1.8)$$

où  $V = \{v_1, \dots, v_{N_v}\}$  est la liste des  $N_v$  sommets 3D  $v_i = (x_i, y_i, z_i)^T$ , chaque sommet dans le polygone maillé est stocké une seule fois dans la liste des sommets  $V$ .  $P = \{p_1, \dots, p_{N_p}\}$  est la liste des polygones où chaque polygone est défini par une liste d'indices (ou de pointeurs) dans la liste des sommets, ainsi un polygone utilisant les sommets numérotés 3, 5, 7 et 10 dans la liste des sommets pourrait alors être représenté par  $p = \{3, 5, 7, 10\}$ .

Cette représentation a plusieurs avantages : du fait que chaque sommet est stocké une seule fois, une place mémoire considérable est économisée. En outre, les coordonnées du sommet peuvent être changées facilement.

Un ensemble de techniques, communément appelées les méthodes de *polygonisation*, ont été élaborées afin de générer une représentation en polygones maillés à partir d'autres primitives géométriques telle que les surfaces implicites [Ning et Bloomenthal, 1993] et les surfaces paramétriques [Krishnamurty et Levoy, 1996].

La représentation d'objets 3D par polygones maillés est de plus en plus répandue en vision par ordinateur surtout depuis l'avènement de nouvelles machines de plus en plus puissantes (grande capacité de stockage et puissance de traitement), en plus de l'immense popularité que connaît l'utilisation des capteurs de profondeurs « dense range sensors ». En effet, ces derniers produisent des nuages de points 3D à partir desquels un maillage triangulaire peu facilement être généré. De plus, les maillages générés peuvent approximer fidèlement n'importe quel objet de forme libre aussi complexe soit il et avec la précision que l'on veut du moment que l'espace mémoire alloué pour stocker la représentation soit suffisamment

important. La figure (1.5) montre une représentation en polygones maillés (maillage triangulaire) d'une surface de forme libre ainsi que le rendu généré à partir de ce maillage. Généralement la résolution exigée pour un maillage (densité du maillage) varie en fonction de l'application. De nombreuses recherches sont récemment menées dans plusieurs laboratoires de vision et qui ont pour but d'élaborer des méthodes qui permettent, à partir d'un maillage grossier, de générer un maillage plus fin dans le but d'avoir une représentation plus précise de l'objet [Xia et Varshney, 1996].

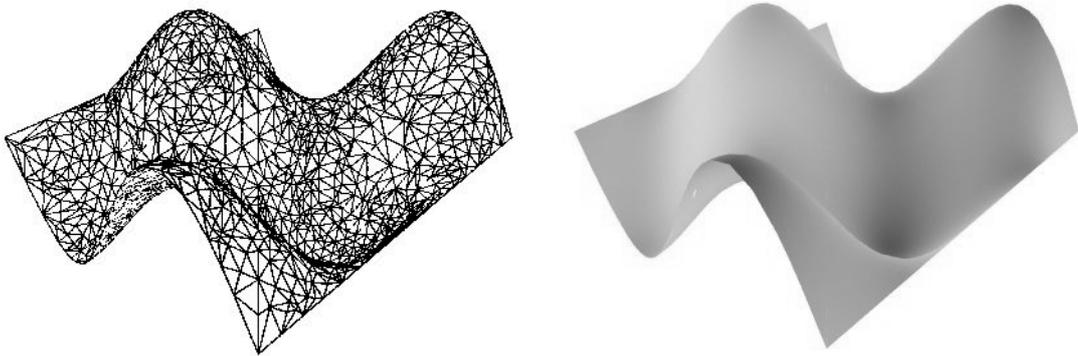


Figure 1.5 : Polygone maillé

### 1.3 Systèmes de reconnaissance d'objets de forme libre

Dans cette partie, nous présentons les différentes approches développées durant ces dernières années dans le domaine de la reconnaissance d'objets de forme libre en présence d'*occlusion* ou non et ce, à partir de leurs images d'intensité ou de profondeur.

#### 1.3.1 Méthodes basées sur les images d'intensité

Les méthodes de reconnaissance d'objets de forme libre qui utilisent exclusivement des images de luminance peuvent être classées comme suit :

##### 1.3.1.1 Méthodes basées sur l'apparence

Parmi les méthodes de reconnaissance qui utilisent les images d'intensité, les méthodes basées sur l'apparence « appearance-based approaches » sont de loin les plus populaires. Leur principe consiste à représenter chaque objet de la scène par un ensemble de ses images d'intensité prises à partir de plusieurs points de vue et avec toutes les directions d'illumination possibles. Chaque vue de l'objet est représentée par un point dans un espace multidimensionnel appelé *espace propre* « eigenspace ». Les fonctions de base qui engendrent cet espace sont déterminées à partir d'un ensemble d'images de références appelées images d'apprentissage. Pour reconnaître une vue quelconque de l'objet on la

projète dans l'espace propre et on détermine, parmi les images d'apprentissage, celle qui s'en rapproche le plus.

Cette idée simple mais efficace a été largement exploitée par les chercheurs sous des formes différentes. Les premiers travaux dans le domaine de la reconnaissance basée sur l'apparence étaient orientés vers la reconnaissance faciale. [Kirby et Sirovich, 1990] ont utilisé l'analyse en composantes principales (ACP) afin de déterminer un système de coordonnées qui prend le mieux en compte (en terme de compression) leurs images d'apprentissage. Les vecteurs de base du nouveau système de coordonnées ont été appelé *images propres* « *eigenpictures* ».

Soit l'ensemble  $X$  des images d'apprentissage  $x_i$  représentant toutes les apparences possibles de l'objet :

$$X = \{x_1, x_2, \dots, x_M\} \quad (1.9)$$

Chaque image d'apprentissage  $x_i$  est un vecteur colonne dont les éléments représentent les valeurs des  $N$  pixels de l'image d'intensité correspondante. Pour identifier correctement une apparence quelconque de l'objet, ce dernier doit être vu dans un ensemble de poses et sous un ensemble de conditions d'illumination qui sont supposées être identiques lors de la phase de reconnaissance.

L'ACP est utilisée pour déterminer un ensemble de vecteurs orthogonaux qui vont former la base de l'espace propre engendré par  $X$ . Pour cela le vecteur moyen :

$$m = \frac{1}{M} \cdot \sum_{i=1}^M x_i \quad (1.10)$$

est soustrait de chaque image  $x_i$  de  $X$  pour produire l'ensemble centré  $X_c$ .  $X_c$  est alors utilisé pour former la matrice de covariance  $N \times N$  :

$$Q = X_c X_c^T \quad (1.11)$$

Les valeurs et vecteurs propres  $\{(\lambda_i, v_i), i=1 \dots N\}$  de la matrice  $Q$  sont ensuite déterminés. L'ensemble des  $N$  vecteurs propres forme la base orthogonale recherchée. La représentation en composantes principales est aussi utilisée pour réduire la dimension du problème. En effet, les images d'apprentissage peuvent être approximées dans un espace de dimension inférieure qui encode grossièrement mais efficacement l'apparence des objets. Pour cela on utilise les seuls vecteurs propres qui correspondent aux valeurs propres les plus importantes afin de caractériser les objets. [Murase et Nayar, 1995] ont démontré qu'un sous espace de dimension 20 est suffisant pour caractériser l'apparence d'un objet pour le problème de la reconnaissance d'objets.

Soit  $\{e_1, \dots, e_k\}$  l'ensemble des vecteurs propres qui correspondent aux  $k$  plus grandes valeurs propres. Les vecteurs centrés  $(x_i - m)$  sont projetés dans le sous espace représenté par les fonctions de base  $\{e_1, \dots, e_k\}$  pour générer des vecteurs réduits de dimensions  $k$  notés  $g_i$ . Une approximation de l'image  $x_j$  peut être construite en utilisant les fonctions de base  $e_i$  et le point projeté  $g_j$ , elle est donnée par :

$$\tilde{x}_j = (e_1, \dots, e_k) g_j + m \quad (1.12)$$

Malgré le fait que cette approximation utilise uniquement  $k$  vecteurs propres au lieu de  $\min(N, M)$  (le rang de la matrice de covariance  $Q$ ), les détails importants dans l'apparence de l'objet sont généralement préservés.

[Turk et Pentland, 1991] ont noté que l'utilisation de la reconnaissance basée sur l'apparence avec un petit nombre d'images d'apprentissage permettait de décrire et de reconstruire les visages d'une large tranche de la population. Leur algorithme a été testé sur une base de données de 16 sujets dont les visages ont été capturés en variant à chaque fois les conditions d'illumination, de pose et d'échelle (2500 images en tout). Ils ont constaté que leur système est plus sensible aux variations de la dimension des visages dans les images qu'aux variations de l'illumination et de la pose. Ils ont montré que cette méthode peut être utilisée pour reconnaître des surfaces complexes et sculptées (visages) en moins d'une seconde.

[Murase et Nayar, 1995] ont étendu l'utilisation de la reconnaissance basée sur l'apparence pour des objets quelconques. En utilisant deux espaces propres, ils sont capables de reconnaître et de localiser les objets sous des conditions d'éclairage variables. L'espace propre universel  $U$  est construit à partir de l'ensemble  $X$  de toutes les images d'apprentissage, il est utilisé pour identifier un objet qui se trouve dans l'image de la scène, par contre l'espace propre de l'objet  $O_j$  est construit pour chaque objet  $j$  à partir de toutes les images d'apprentissage  $x_i$  qui contiennent cet objet, il est utilisé pour déterminer la pose de l'objet ainsi que les conditions d'éclairage. Pour chaque objet ils ont créé une variété<sup>1</sup> *manifold* qui permet d'interpoler les points  $g_i$  dans les espaces propres. Ces variétés sont construites grâce à un algorithme d'interpolation spline-cubique. Pour identifier un objet dans une image  $s$ , celle-ci est premièrement projetée dans l'espace propre universel:

$$g = \begin{pmatrix} e_1^T \\ \vdots \\ e_k^T \end{pmatrix} \cdot (s - m) \quad (1.13)$$

puis appariée avec l'objet ayant la variété la plus proche. Dans des travaux parallèles, [Murase et Nayar, 1994] ont étudié plus en profondeur l'influence de l'éclairage sur leur méthode de reconnaissance. Le but de leur étude était de déterminer les paramètres de l'illumination qui permettaient de maximiser la différence en apparence entre les objets.

[Munday et al., 1996] ont comparé la méthode de [Murase et Nayar, 1995] avec deux approches référencées modèles. Ils ont conclu que le système référencé apparence possédait la plus grande vitesse de reconnaissance, mais d'un autre côté il fournissait un plus grand nombre de faux appariements. Ceci est dû à l'absence de la procédure de

<sup>1</sup> Une variété est un concept mathématique utilisé en géométrie différentielle et qui permet de généraliser la notion d'espace pour aborder d'une manière rigoureuse des géométries différentes de la géométrie euclidienne.

vérification dans les méthodes basées sur l'apparence ainsi qu'à leur sensibilité aux occultations et aux erreurs de segmentation. A l'inverse, les approches référencées modèles sont basées sur le principe même de la génération/vérification d'hypothèses, et de ce fait minimisent le nombre de faux appariements.

[Black et Jepson, 1998] ont montré que la projection  $g$  est le résultat d'une estimation au moindre carrés de l'image originelle  $x$ . Pour compenser la sensibilité bien connue des techniques des moindres carrés aux bruits, les auteurs ont utilisé des méthodes statistiques robustes dans le calcul de la projection  $g$ . Le but de leur étude était de développer un système de poursuite et de reconnaissance des gestes de la main.

Toutefois, les méthodes basées sur l'apparence souffrent de leur manque d'efficacité pour traiter les cas où la scène contient plusieurs objets avec une possibilité d'occultation. [Huang et al., 1997] ont proposé de subdiviser les images en plusieurs régions. Ils exploitent les relations entre ces régions ainsi que leur apparences afin d'identifier aussi bien les objets de la scène que leur pose.

[Camps et al., 1998] se sont basés sur les travaux de [Huang et al., 1997] pour développer un système de reconnaissance qui intègre des bases de données hiérarchiques et une méthode d'appariement Bayésienne. L'inconvénient des approches « régions » est qu'elles doivent impérativement segmenter les images d'entrée en temps réel, ce qui limite la catégorie d'objets reconnaissables aux seuls objets pouvant être segmentés d'une manière fiable, ce qui est loin d'être le cas des objets de forme libre. [Ohba et Ikeuchi, 1997] et [Krumm, 1996] ont surmonté ce problème en développant une nouvelle technique basée sur l'apparence qui utilise des fenêtres locales définies à partir de l'apparence des objets. [Ohba et Ikeuchi, 1997] sont capables de traiter les cas de translation et d'occultation d'un objet de la scène en utilisant ce qu'ils ont appelé les *fenêtres propres* « eigenwindows ». Les fenêtres propres encodent les informations sur l'apparence de l'objet dans de petites parties de l'image. Des mesures de la *déteçtabilité*, l'*unicité* ainsi que la *précision* ont été aussi développées pour les fenêtres propres. Ces mesures permettent d'écarter de l'ensemble d'apprentissage, lors de la phase de reconnaissance, les fenêtres qui sont difficiles à détecter, ne sont pas discriminantes ou bien sont sensibles aux bruits. Grâce à cette technique, les auteurs sont capables de reconnaître plusieurs objets qui ont subi des occultations dans la scène.

[Leonardis et Bischof, 1996] se sont aussi intéressés au problème de l'occlusion, leur méthode consistait à sélectionner d'une manière aléatoire un ensemble de points de l'image de la scène et leur correspondants dans l'espace engendré par la base des vecteurs propres  $[e_1, \dots, e_k]$ . Ils ont utilisé le paradigme de la génération/vérification d'hypothèses pour reconnaître les objets. Cette méthode a démontré qu'il était possible de reconstruire des portions non visibles des objets de la scène. [Bischof et Leonardis, 1998] ont étendu leurs travaux pour traiter les problèmes du changement d'échelle « scaling » et de la translation des objets dans la scène.

[Rao, 1997] a appliqué l'apprentissage adaptatif aux vecteurs de base  $[e_1, \dots, e_k]$  de l'espace propre dans les méthodes basées sur l'apparence. L'approche multi-vues dynamique est utilisée pour prédire les changements spatio-temporels dans l'apparence d'une séquence d'images. Cette prédiction permet de modifier les vecteurs propres en minimisant une fonction de coût basée sur le principe de la longueur minimale de la description « MDL : Minimum Description Length ». Ceci permettait d'apprendre les données d'entrées sans pour autant avoir à mémoriser leur détails, ce qui rendait le système de reconnaissance plus flexible.

Le tableau (1.1) résume les principales caractéristiques des techniques de reconnaissance référencées apparence étudiées ci-dessus. Pour chaque technique, Il indique son aptitude ou non à traiter de multiples objets de la scène avec possibilité d'occlusion et changement d'échelle. Il donne aussi les tailles des bases de données utilisées ainsi que le taux de reconnaissance de chaque technique défini par  $\frac{N_c}{N_{essais}}$  où  $N_c$  correspond au nombre de fois où l'objet est correctement identifié et  $N_{essais}$  est le nombre total d'essais.

Technique	Occlusion	Changement d'échelle	Taille de la BD	Taux de reconnaissance
[Kirby et al, 1990].	Non	Oui	N.P <sup>2</sup>	N.P
[Turk et Pentland, 1991]	Non	Non	16	1.0
[Murase et Nayar, 1995].	Non	Oui	100	1.0
Black et Jepson, 1998]	Non	Non	N.P	N.P
[Camps et al., 1998]	Oui	Non	24	1.0
[Ohba et Ikeushi, 1997]	Oui	Non	N.P	N.P
[Krumm , 1996]	Oui	Non	2	0.8
[Leonardis et Bischof, 1996]	Oui	Oui	20	N.P
[Rao, 1997]	Oui	Non	N.P	N.P

Tableau 1.1 : Résumé des techniques de reconnaissance basées sur l'apparence

### 1.3.1.2 Méthodes basées sur l'extraction de silhouettes

Les silhouettes d'objets ont été largement utilisées ces dernières années par la communauté de vision par ordinateurs pour caractériser les objets de formes libres. En effet, dans un environnement à éclairage contrôlé, la silhouette d'un objet peut être tout à fait utile pour déterminer son identité et sa pose.

[Mokhtarian, 1995] a développé un système de reconnaissance basé sur l'utilisation des silhouettes fermées des objets. Le système est élaboré pour reconnaître des objets de forme libre en ayant seulement quelques vues de disponibles dans un environnement où un seul objet est présent. Pour cela, une source lumineuse contrôlable est utilisée pour illuminer l'objet de sorte à ce que la frontière entre le fond et l'objet soit facile à détecter, ce qui permet d'isoler l'objet par un simple seuillage. Les courbes correspondantes aux frontières

<sup>2</sup> N.P : non précisé par les auteurs

de l'objet (extraites par suivi de contour) sont alors représentés par leur courbures calculées dans l'espace échelle «CSS : Curvature Scale Space representation). L'appariement entre les courbes CSS est effectué en se basant sur la position des maximums de courbure. Pour les courbes de frontière convexes, la représentation CSS est lissée jusqu'à ce qu'il ne reste que quatre maximums de courbure, par contre pour les courbes concaves, elle utilise la totalité des maximums de courbure obtenus pour une échelle donnée. Durant la phase de reconnaissance, le rapport d'aspect de la silhouette de l'objet est utilisé en premier afin de préfiltrer les appariements possibles entre la scène et le modèle avant de procéder à l'appariement proprement dit des silhouettes, ceci permet de réduire l'espace de recherche. Cette technique représente un moyen rapide et efficace pour appairer des caractéristiques grossières de la scène avec celles du modèle. Les appariements plausibles obtenus sont alors vérifiés par une méthode de reconstruction qui permet de mesurer l'erreur entre chacune des deux courbes appariées. Ce système est capable d'identifier les 19 objets stockés dans la base de données.

[Ponce et Kriegmann, 1989] ont utilisé les contours générés à partir des normales à la surface discontinues ainsi que les contours d'occlusion pour reconnaître les objets 3D. Ils ont modélisé les objets par des surfaces paramétriques afin de générer les modèles de contours 3D à reconnaître dans les images d'intensité. Leur méthode consiste à déterminer les équations implicites des patches paramétriques, ensuite utiliser les intersections entre ces patches pour construire une équation implicite du contour. Un modèle de projection perspective faible "*weak*" est utilisé pour corrélérer les contours 3D avec les contours 2D extraits des images d'intensité. Les contours 2D sont découpés manuellement puis regroupés en plusieurs « clusters » représentant un seul objet. Ce système a été utilisé pour modéliser les surfaces de révolution. Leur résultats expérimentaux ont donné des temps de reconnaissance rapides ( $\approx 30$  secondes) et une localisation assez précise (erreur moyenne de l'ordre de 0.4-1.4 pixels).

[Joshi et al., 1994] ont développé une méthode de reconnaissance utilisant les courbes HOT ("*higher order tangent*") pour modéliser un objet lisse ("*smooth*"). Ces courbes sont identifiées à partir d'un ensemble de points sur la silhouette de l'objet dans l'image d'intensité. Les angles entre les droites tangentes aux points et le rapport des distances entre les points du contour sont utilisés pour construire un invariant géométrique à la pose de l'objet ainsi qu'au facteur d'échelle. Ce dernier est ensuite utilisé pour indexer une table de reconnaissance. Ils ont testé leur système de reconnaissance sur quatre objets (une courge, une poire, une banane et un leurre de canard) et ont démontré que la représentation par des courbes HOT permettait de retrouver dans tous les cas, l'identité et la pose de l'objet, après que la phase de vérification ait rejetée les faux appariements.

[Chen et Stockman, 1998] ont utilisé les silhouettes avec les bords internes ("*internal edges*") pour retrouver l'identité et la pose de 20 modèles d'objets de forme libre à partir d'images d'intensité. La courbe de la silhouette est combinée avec un invariant géométrique (dont le rôle est d'indexer la base de données) pour générer des hypothèses sur la pose et l'identité du modèle. Afin de tolérer les occlusions, ils ont utilisé la représentation partielle de contour. Les attributs invariants des segments de courbe extraits

à partir de cette représentation sont ensuite utilisés pour indexer les modèles dans la table de hachage. Cette méthode permet d'apparier des parties du modèle avec les parties observées dans l'image. Les appariements ainsi obtenus sont ensuite groupés dans des hypothèses consistantes basées sur l'identité de l'objet (associé à chaque partie) et sur une estimation approximative de la pose. Chaque hypothèse est ensuite vérifiée en comparant la carte des bords internes du modèle « model edges map » avec celle des bords observés. La phase de vérification produit aussi une estimation plus raffinée de la pose de l'objet. L'inclusion des bords internes dans la phase de vérification améliore les performances du système en permettant de rejeter les fausses hypothèses sur l'identité et la pose de l'objet.

### 1.3.1.3 Méthodes basées sur le « shape from shading »

Comme son nom l'indique, il s'agit d'extraire une forme uniquement à partir des variations de lumières observées sur la surface d'un objet. En dépit de l'intérêt que portent les chercheurs au "shape from shading" (SFS) depuis plusieurs années, on trouve très peu de publications sur son utilisation dans les systèmes de reconnaissance d'objets 3D [Worthington et Hancock, 2001]. Une des raisons principales de cet état de fait (omission) est l'absence d'algorithmes robustes capables de retrouver les détails fins et discriminants des surfaces des objets. Ceci dit, nous pensons que cette approche peut apporter une solution efficace au problème de la reconnaissance d'objets de forme libre, et mérite d'être explorée plus en détails, surtout lorsqu'on sait la capacité du SFS à fournir directement à partir d'images d'intensité, des informations sur la topographie de la surface de l'objet.

Dernièrement [Worthington et Hancock, 2001] ont étudié la possibilité d'utiliser le SFS dans la reconnaissance d'objets 3D. Ils ont démontré que les informations sur l'orientation locale de la surface obtenues en utilisant le SFS représentent des données d'entrées très utiles pour les systèmes de reconnaissance basés sur l'apparence. En effet, à partir des informations photométriques, ils extraient deux représentations de l'objet à savoir, la "needle-map" et l'index de la courbure locale de la forme ("local curvature shape-index"), puis ils examinent leur performances relatives dans la reconnaissance d'objets 3D. Concrètement, ils ont développé deux stratégies de reconnaissance basées sur les histogrammes, la première utilise directement les normales à la surface, tant dis que la seconde est basée sur l'index de la forme. Ils ont expérimenté ces deux stratégies de reconnaissance sur 20 objets de formes quelconques, et ont constaté que les deux représentations donnaient de bon résultats. Cependant, Ils ont trouvé que l'histogramme des normales à la surface était plus efficace que l'histogramme de l'index de la forme. Une étude de la sensibilité a révélé que la méthode "needle-map" offre une discrimination significative de la topologie différentielle de l'apparence de l'objet. Elle permet d'extraire des caractéristiques discriminantes de l'objets 3D à partir de ses images 2D.

### 1.3.2 Méthodes basées sur les images de profondeur

L'utilisation des images de profondeur dans la reconnaissance d'objets 3D ne cesse d'augmenter ces dernières années. La diversité des systèmes développés dans ce domaine démontre la variété des approches élaborées. Le but de cette section est de décrire, d'une manière non exhaustive, les systèmes de reconnaissance d'objets de forme libre les plus récents basés sur les images de profondeur.

Comparés aux images d'intensité, les images de profondeur offrent une information plus complète sur la géométrie de l'objet. En effet, leur capacité de capturer directement des points 3D appartenant à la surface de l'objet permet de comparer géométriquement les images de profondeur avec les modèles 3D.

[Nevatia et Binford, 1977] ont élaboré une technique qui utilise une description symbolique obtenue à partir d'une segmentation partielle en cônes généralisés des images de profondeur afin de reconnaître des objets articulés de forme libre (une poupée, un cheval, un gant) en présence d'occlusion. Ces parties (cônes généralisés) permettent de construire une description symbolique de la scène à partir de leur propriétés (e.g. longueur de l'axe, largeur moyenne de la section), des relations de connectivité (e.g. le nombre de parties connectées de chaque objet) et des propriétés globales (e.g. nombre total de parties). Les parties les plus distinctives sont utilisées pour indexer les objets possibles dans la base de données. Ces parties distinctives correspondent aux plus larges cônes généralisés de la base de données, elles sont intéressantes car elles permettent une meilleure segmentation en présence d'occultation. Les descriptions de la scène et du modèle sont ensuite comparées, en commençant par appairier les parties distinctives. Ces appariements sont ensuite enrichis par d'autres appariements des parties dont les propriétés et les connections sont consistantes. Ceci est équivalent à un appariement de graphes. Le graphe représentant les parties connectés de la scène est plus petit que le graphe du modèle à cause de la présence d'occlusion et des erreurs de segmentation. La qualité de l'appariement des graphes est évaluée en fonction de la qualité des appariements individuels des différentes parties. Cette représentation fonctionne très bien pour les objets qui ont des descriptions symboliques très différentes.

[Raja et Jain, 1992] ont développé un système qui permet d'ajuster des modèles de superquadriques déformables aux données de profondeur, où les déformations considérées sont l'amincissement linéaire et le recourbement plan. Les paramètres qui définissent la superquadrique ajustée sont utilisés pour classifier la forme parmi 12 classes de géons. Les classes de géons sont utilisées pour discriminer les formes qui sont « qualitativement différentes ». Ces différences qualitatives concernent la forme des axes et des sections (droits ou courbes), l'augmentation, la diminution, ou bien l'augmentation puis la diminution de la surface de la section selon l'axe principal de la superquadrique. Avec des images de profondeurs réelles, leur méthode de classification a pu identifier la classe de la geon dans 77% des cas, par contre avec des images de synthèse le taux d'identification correcte était de 87%.

[Thirion, 1996] a utilisé la courbure de la surface pour définir une représentation globale des objets de forme libre. Sa méthode exploite les extremums de courbure sur la surface pour trouver les points singuliers et les contours. Les points/contours extremums sont définis par les passages par zéro entre les maximums et les minimums de la courbure gaussienne  $K$  ( $K=K_1*K_2$ , où  $K_1$  et  $K_2$  sont les courbures principales). Cette représentation a été appliquée sur des données de profondeur réelles ainsi que sur des modèles synthétiques pour déterminer des maillages extrêmes ("extremal meshes") de l'objet. L'auteur a noté que cette représentation restait encore très complexe pour être appliquée à la reconnaissance d'objets car elle ne s'adapte pas facilement aux techniques rapides d'appariement.

La courbure de la surface a été aussi utilisée pour classifier les formes des surfaces locales en un ensemble de formes représentatives. [Besl et Jain, 1985] ont utilisé la courbure gaussienne  $K$  et la courbure moyenne ( $H=(K_1+K_2)/2$ ) pour classifier la forme de la surface locale en 8 catégories de bases. [Dorai et Jain, 1997] ont étendu cette approche par la définition de deux nouvelles mesures de courbures, en l'occurrence l'index de la forme donné par :

$$S = \frac{1}{2} - \frac{1}{\pi} \arctan\left(\frac{K_1 + K_2}{K_1 - K_2}\right) \quad (1.14)$$

et le taux de courbure ("curvedness") :

$$R = \sqrt{\frac{K_1^2 + K_2^2}{2}} \quad (1.15)$$

où l'index de la forme  $S$  permet de classifier la forme locale de la surface en 9 types différents et le taux de courbure  $R$  mesure l'amplitude des variations de la courbure.

Les auteurs ont utilisé ces nouvelles mesures pour construire une représentation des objets de forme libre qui dépend du point de vue. Leur système (appelé COSMOS, pour "Curvedness-Orientation-Shape Map On Sphere") utilise un histogramme des valeurs de l'index de la forme  $S$  pour caractériser les surfaces présentes dans chaque vue de l'objet. Ainsi, pour une vue donnée, l'histogramme permet de stocker la distribution des zones ("patch") en fonction des valeurs de l'index de la forme. Ces histogrammes (appelés aussi spectres de la forme) peuvent être appariés rapidement en utilisant leur moments, de plus ils sont invariants à la rotation. Le système construit une base de données constituée de plusieurs vues de chaque objets à reconnaître. Pour réduire la complexité de la procédure de recherche, les vues d'un objet sont groupées en "clusters" en se basant sur leur similarité. Ensuite, pour chaque cluster ainsi défini, un histogramme prototype est calculé en faisant la moyenne de tous les histogrammes du "cluster". Le processus qui permet d'apparier l'histogramme de la scène avec la base de données commence premièrement par le comparer avec chaque prototype de cluster. Ensuite, les  $n$  clusters qui coïncident le mieux avec la scène sont examinés pour trouver quelles vues dans ces clusters correspondent le mieux à la scène. La vue qui offre le meilleur appariement permet d'identifier l'objet ainsi que sa pose. En moyenne, seulement 20% des vues de la base de

données avec un total de 20 objets sont utilisées pour obtenir une identification. Ce système de reconnaissance fonctionne très bien dans le cas de scènes contenant un seul objet non polyédrique.

[Stein et Medioni, 1992] ont utilisé les variations de l'orientation de la surface afin d'apparier localement des patches de surfaces. La nature locale de leur technique d'appariement leur permet d'identifier plusieurs objets de formes libres dans des scènes occluses. Pour réaliser de bons appariements entre les points de la scène et les points d'un modèle connu à priori ils ont imaginé une nouvelle méthode qui permet de mesurer la différence entre les distributions relatives des normales à la surface. Pour un point donné  $P$  de la surface, les normales à une distance  $d$  de  $P$  contiennent une information structurale sur la surface autour du point  $P$  (figure 1.6). La distribution de toutes les normales  $N_d(\theta)$  à la surface autour du point  $P$  est appelée "splash", car son apparence est remarquablement similaire à celle des éclaboussures d'eau d'eau.

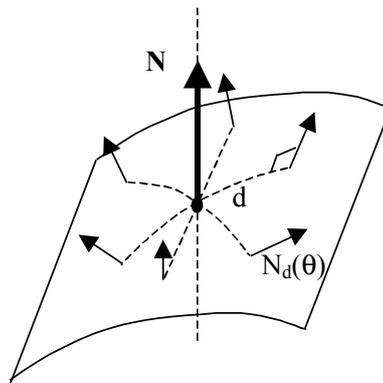


Figure 1.6 : Définition d'un "splash"

Pour coder les informations relatives aux normales  $N_d(\theta)$ , un système de coordonnées sphériques est utilisé (figure 1.7), où les angles  $\phi(\theta)$  et  $\psi(\theta)$  donnent l'orientation relative de  $N_d(\theta)$  par rapport à la normale au point  $P$  et l'axe  $X(\theta)$ .  $X(\theta)$  est perpendiculaire au vecteur  $\vec{N}$  et se trouve dans le plan qui contient  $\vec{N}$ , le point  $P$  est le point dont la position est définie par la distance  $d$  de  $P$  et l'angle  $\theta$ .

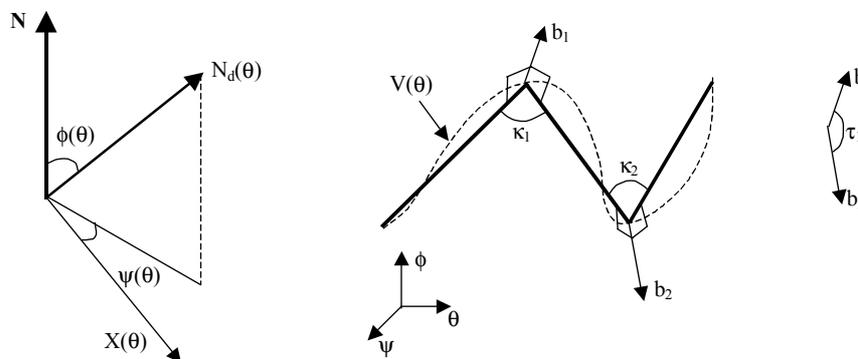


Figure 1.7 : Les coordonnées d'orientation des paramètres du "splash"

Puisque  $\theta$  varie entre 0 et  $2\pi$ , les valeurs de  $\phi(\theta)$  et  $\psi(\theta)$  forment une courbe 3D :

$$v(\theta) = \begin{pmatrix} \phi(\theta) \\ \psi(\theta) \end{pmatrix} \quad (1.16)$$

Pour pouvoir apparier rapidement les paires de courbes  $v_i(\theta)$  et  $v_j(\theta)$ , les courbes sont représentées par des polygones en commençant par le point dont la valeur de  $\theta$  maximise  $\phi(\theta)$ . Ce choix du point de départ offre une méthode qui assure une invariance à la rotation dans le codage des courbes. Finalement, la courbe polygonale est codée par une représentation appelée super segment 3D (figure 1.7). Le super segment 3D stocke les angles de courbures  $\kappa_i$  entre les points de liaison (les points d'intersection entre les segments du polygone) et les angles de torsion  $\tau_i$  illustrés dans la figure (1.7).

Pour les besoins de la reconnaissance, les "splash" sont choisis dans les zones de forte courbure de l'objet. En effet, dans ces zones les variations entre la normale au point  $P$  et les normales  $N_d(\theta)$  donnent une description structurelle très riche sur la forme de la surface locale de l'objet. Les auteurs ont employé l'indexation structurelle pour l'appariement. Cette méthode est une variante du hachage géométrique (voir chapitre 2 § 2.6.5) où les indices de la table de hachage sont liés aux structures formées à partir des données. Dans ce cas les structures considérées sont les super segments 3D. En effet, les angles de courbure  $\kappa_i$  et de torsion  $\tau_i$  entre les segments de la courbe polygonale avec le nombre de segments, la différence maximale de l'orientation entre le point d'intérêt et le contour maximal  $\phi(\theta)$ , et le rayon  $d$  sont utilisés pour indexer la table de hachage. Pour un point d'intérêt donné, l'entrée de la table est codée pour différentes valeurs du rayon  $d$  et du nombre de segments de l'approximation polygonale du "splash". Durant la phase de reconnaissance, les points d'intérêt de la scène sont déterminés et utilisés pour indexer la table de hachage. Les appariements sont utilisés pour générer des hypothèses, ces dernières sont groupées en fonction du modèle auxquelles elles correspondent. Ensuite, pour chaque modèle un ensemble d'hypothèses est construit. Ces ensembles sont ensuite utilisés pour trouver la pose de l'objet. Finalement chaque modèle supposé dans la pose estimée est vérifié grâce aux données de la scène. Les auteurs ont démontré que dans le meilleur des cas, quand un seul objet est présent dans la scène, la complexité de la méthode est donnée par  $O(n)$ , alors que dans le pire des cas quand plusieurs instances de l'objet sont présentes dans la scène avec une occultation partielle, la complexité est égale à  $O(n^2 m^3)$ , où  $n$  est le nombre de points 3D de la scène et  $m$  le nombre de modèles. Ils ont montré par l'expérimentation que leur système fonctionne assez rapidement et peut s'appliquer aux scènes complexes modérément bruitées.

[Chua et Jarvis, 1997] ont élaboré une nouvelle représentation qu'ils ont appelé « signature du point » et qui est inspirée des travaux de [Stein et Medioni, 1992]. La signature du point est différente du "splash" dans le sens où elle ne code pas l'information sur les normales autour des points d'intérêt, mais code plutôt les distances minimales des points d'un contour 3D à un plan de référence (figure 1.8). Le contour 3D est construit à partir de l'intersection de la surface de l'objet avec une sphère de rayon  $r$  centrée sur  $P$ . L'analyse en composantes principales du contour 3D permet de définir un plan pour lequel les distances des points du contour à ce plan sont minimales. Le plan est ensuite translaté

jusqu'à ce qu'il contienne le point  $P$ . La distance signée au contour forme une courbe paramétrique 1D  $d(\theta)$  comme c'est illustré dans la figure (1.8). La représentation finale est donnée par la courbe discrète  $d[n]=d(n*\Delta\theta)$ , où  $\Delta\theta$  est égal à 15 degrés. Ceci fournit un moyen compact pour stocker les informations sur la structure de la surface autour du point d'intérêt, de plus cette représentation est invariante à la pose. Pour améliorer la sélectivité de l'appariement, ils ont choisi de coder deux signatures pour chaque point d'intérêt sur l'objet, ces deux signatures sont définies par le choix de sphères ayant des rayons différents.

Pour discriminer suffisamment l'objet, la courbe paramétrique discrète  $d[n]$  est définie à chaque point du modèle. Ces signatures sont ensuite placées dans une table d'indexation. Chaque cellule de la table contient la liste des signatures des points du modèle dont les valeurs de  $\min d[n]$  et  $\max d[n]$  sont similaires. Durant la phase de reconnaissance, les signatures des points sont calculées sur une grille également espacée dans la scène. Ces signatures sont comparées avec celles du modèle stockées dans la cellule appropriée de la table d'indexation. Les signatures similaires génèrent des correspondances plausibles entre le modèle et la scène. Les hypothèses ainsi générées sont groupées par modèle, les modèles sont ensuite ordonnés en fonction du nombre d'hypothèses qu'ils reçoivent. Les modèles ayant le plus de correspondances sont vérifiés. Finalement, la transformation rigide (rotation + translation) entre les points de la scène et les points du modèle est calculée.

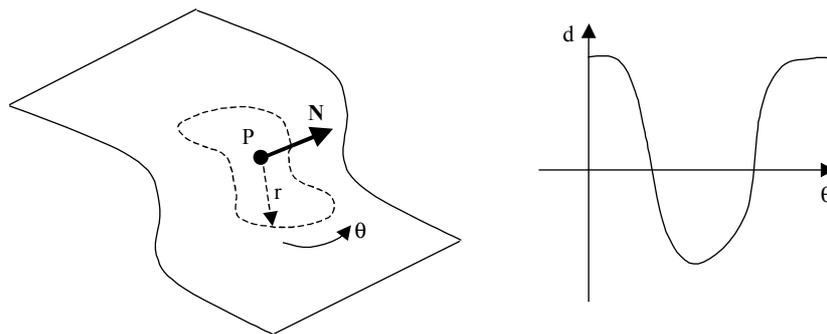


Figure 1.8 : Signature d'un point

Une fois implémenté, ce système a donné des temps de reconnaissance très courts pour des scènes contenant plusieurs objets d'une base de données de 15 modèles.

[Jonhson et Hebert, 1999] ont eux aussi utilisé des caractéristiques définies à partir de points pour résoudre le problème de la reconnaissance d'objets de forme libre. En effet, les images de spin ("spin images") qu'ils ont élaborées sont des histogrammes 2D qui codent les emplacements des points de la surface de l'objet autour d'un point donné. L'image de spin est générée en utilisant la normale au point d'intérêt et un plan coupant tournant autour de ce point dont la normale constitue l'axe de rotation (figure 1.9).

Les intersections entre le plan tournant et la surfaces sont utilisées pour indexer l'histogramme 2D. Ainsi, les cellules de l'histogramme vont contenir le nombre de fois qu'un patch donné du plan tournant coupe la surface de l'objet. Les images de spin sont

généérées à partir des modèles en utilisant une approximation polygonale de leurs surfaces. Les sommets dans un modèle polygonal sont approximativement uniformément distribués le long de l'objet. Lorsque les images de spin sont générées à partir des données réelles issues d'un scanner, un critère similaire est exigé. Cette uniformité est nécessaire afin que les cellules dans les images de spin approximent au mieux les zones de la surface coupées par le plan tournant.

Durant la phase de reconnaissance, la scène est échantillonnée d'une manière aléatoire afin de déterminer les points où seront générées les images de spin. Ceci est réalisé avant même que 20% des points de la scène ne soient tous échantillonnés. Chaque image de spin de la scène est comparée avec toutes les images de spin des modèles. Le résultat de ces comparaisons permet de construire un histogramme de similarité.

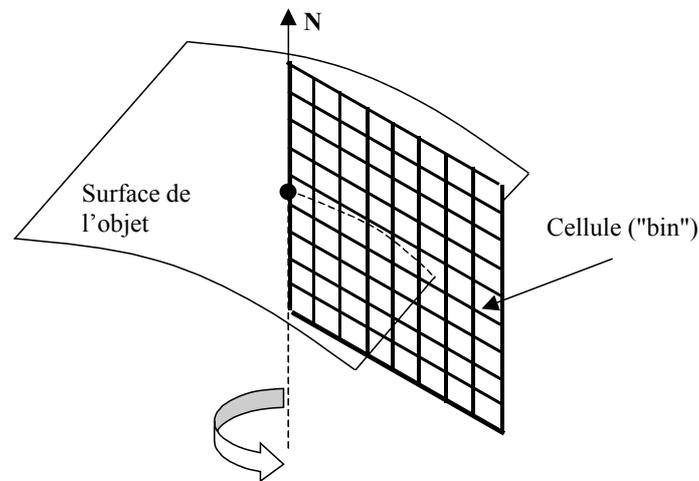


Figure 1.9 : Image de spin

Les meilleures similitudes trouvées par une analyse de l'histogramme précédent sont utilisées pour générer des appariements hypothétiques entre un point de la scène et peut être plus d'un point du modèle. Une fois que tous les points de la scène ont été utilisés pour générer de possibles correspondances, ils sont à nouveau filtrés pour supprimer 50% des appariements les plus mauvais. Les correspondances restantes sont alors groupées par modèle et leur consistance géométrique est mesurée. Les groupes d'appariements, géométriquement consistants, entre les points de la scène et du modèle sont utilisés pour calculer la transformation rigide modèle/scène. Un algorithme ICP (voir § 2.6.2) modifié [Besl et McKay, 1992] est ensuite utilisé pour affiner la transformation avant que l'étape finale de vérification confirme ou rejette l'appariement modèle/scène. Ce système de reconnaissance a été utilisé pour identifier des objets de forme libre partiellement occlus. En effet, les auteurs ont montré que leur système fonctionne très bien sur un objet où seulement 20% de sa surface est visible. La complexité du calcul pour établir un appariement est  $O(SMI + \log(M))$ , où  $S$  est le nombre de points de la scène,  $M$  est le nombre de points du modèle, et  $I$  est le nombre de pixels (cellules) dans l'image de spin.

[Delingette et al., 1993] ont développé une nouvelle représentation sphérique pour la reconnaissance d'objets qu'ils ont appelé attribut sphérique de l'image (SAI : spherical attribute image). La représentation SAI permet de représenter les points de la surface de l'objet par des sommets sur une sphère maillée « tessellated » quasi régulière. La régularité implique que les facettes triangulaires de la sphère ont des surfaces similaires et sont équilatérales. Les caractéristiques de la surface locale sont ainsi stockées dans les sommets de la sphère maillée qui correspondent aux points de la surface. La correspondance entre un point de la surface et un sommet de la sphère maillée est déterminée par l'application d'une déformation/rétrécissement à une version ellipsoïdale de cette même sphère pour qu'elle coïncide avec la surface de l'objet. La déformation est contrôlée par des forces qui permettent de préserver la régularité en rétrécissant le maillage à la surface. Cette condition de régularité offre à la représentation SAI une propriété d'invariance à la rotation ainsi qu'une capacité d'appariement des objets occlus. Toutefois, les auteurs ont signalé que l'invariance à la rotation est vraie seulement lorsque le nombre de nœuds du maillage devient très grand, ceci est dû au fait que la représentation SAI est une représentation discrète (approximative) de la surface de l'objet. Ils ont résolu ce problème en calculant la moyenne aux sommets de la SAI afin d'approximer les valeurs en tout point entre les nœuds du maillage. L'appariement entre un objet de la scène et le modèle se réduit à déterminer la meilleure transformation entre leur deux représentations SAI. La stratégie la plus simple pour réaliser cela est d'échantillonner l'espace de toutes les rotations possibles que peut subir la sphère puis d'évaluer la mesure de distance entre les SAI. La rotation qui minimise l'erreur est considérée comme une candidate d'appariement. Pour les bases de données qui contiennent plus d'un objet, la représentation SAI de chaque objet doit être appariée avec la scène. L'appariement optimal peut être déterminé en quelques secondes. Cette approche a été étendue pour traiter les objets occlus. Le maillage résultant à partir d'une vue partielle de l'objet est mis à l'échelle, et les nœuds interpolés ne sont pas pris en compte dans les mesures de distances entre les SAI. Une mesure de similarité entre les formes des SAI a été aussi introduite, ainsi une classification des objets similaires (bases de données hiérarchiques) peut être utilisée pour les regrouper dans des ensembles (ou bien classes) sous un seul prototype de classe. La mesure de similarité est effectuée pour différentes valeurs d'échelles de la représentation SAI. Ainsi, la version multi-échelle de la représentation SAI de l'objet peut être comparée en premier avec le prototype pour déterminer si les objets de la classe doivent être appariés ou non avec la représentation SAI de la scène. Ceci permet de réduire énormément le temps d'appariement entre la scène et la base de données.

[Zhang et Herbert, 1997] ont aussi utilisé la mesure de similarité pour classifier la représentation des objets par SAI pour différentes valeurs du facteur d'échelle. Ils ont introduit la notion de lissage de la surface d'un objet à travers sa représentation SAI sans pour autant utiliser une opération d'amincissement. Ceci a permis l'appariement d'objets à des niveaux différents du détail, améliorant ainsi l'efficacité de leur système de reconnaissance.

[Barequet et Sharir, 1997] ont élaboré une nouvelle approche d'appariement entre des portions de surfaces en déterminant les paramètres d'une reconstruction. La technique de

reconstruction utilisée a été inspirée par une méthode basée sur les images d'intensité pour la reconnaissance et la localisation d'objets. La méthode utilise une primitive de type empreinte ("footprint"), invariante à la rotation et la translation, pour différencier les objets. Un exemple d'empreinte utilisée par les auteurs est la courbure en un point de la surface. L'utilisation des empreintes permet d'établir des correspondances entre les points de la scène et du modèle. Une série de rotations est alors appliquée à la liste de correspondances pour déterminer la qualité de chaque appariement. Ceci produit une table de votes dont les entrées enregistrent la qualité de chaque rotation (i.e on attribue des notes aux rotations en fonction de la qualité de l'appariement qu'elles engendrent). L'entrée de la table ayant le meilleur score est utilisée comme une estimation initiale de la rotation de l'objet dans l'algorithme itératif de raffinement. Une fois la rotation optimale est déterminée, elle est utilisée pour calculer la translation. La complexité de la méthode pour déterminer un appariement est  $O(n+k+s^3)$ , où  $n$  est le nombre de points dans l'ensemble d'entrée,  $k$  est le nombre de correspondances et  $s$  est la taille maximale de la table de vote.

[Greenspan, 1998] a utilisé le paradigme de la génération/vérification d'hypothèses pour éviter le problème typique d'extraction de primitives associé aux techniques de reconnaissance 3D. Le but de sa méthode était de déterminer toutes les occurrences du modèle dans l'image de profondeur. Pour réaliser cela il a utilisé des points de l'image, qu'il a appelé *graine* ("seed points"), comme points de départ d'un algorithme séquentiel de type hypothèse-et-test. Une *graine* est un point de l'image supposé appartenir à la surface du modèle, l'ensemble des poses possibles pour lesquelles cette hypothèse reste vraie est très grand puisque une graine peut correspondre à n'importe quel point sur l'objet. Pour réduire le nombre de poses possibles, des points successifs de l'image sont considérés et les hypothèses qu'ils génèrent sont vérifiées jusqu'à ce qu'il n'en reste qu'un petit nombre d'entre elles. Le choix des points dans l'image autour d'une *graine* est guidé par le parcours d'un *arbre échantillon* généré à partir du modèle. Lors de l'implémentation du système, l'image de profondeur est utilisée pour générer une représentation des données par voxels (volume elements), ceci pour deux raisons : premièrement il est plus facile de déterminer si un point de l'espace appartient ou non à la surface de l'objet (les voxels sont étiquetés comme les surfaces, occultés, et de forme libre) et deuxièmement, cela limite considérablement le nombre de poses possibles du modèle. Le temps de calcul nécessaire pour parcourir un arbre échantillon ayant  $n$  nœuds généré à partir de  $s$  points de "graine" est  $O(s \cdot \log^2 n)$ , il correspond au temps nécessaire pour générer l'ensemble initial des hypothèses sur la pose de l'objet dans la scène. Les hypothèses sont ensuite vérifiées par corrélation entre le modèle et les données de la scène pour une pose donnée. Les hypothèses non plausibles restantes sont éliminées en observant la zone de chevauchement ainsi que le volume entre les surfaces de l'objet de la scène et du modèle dans la région appropriée. Finalement, le peu d'hypothèses qui reste est traité par un algorithme ICP afin de raffiner la pose de l'objet puis testé de nouveau en déterminant une autre fois la zone de chevauchement et le volume entre les surfaces. Le système a montré une grande efficacité pour la reconnaissance et le recalage d'objet 3D de forme libre dans des scènes complexes en présence d'occultations.

La reconnaissance d'objets complexes ayant des formes globalement similaires mais dont les détails sont différents (tel que le cerveau humain) représente un nouveau problème pour la communauté de vision par ordinateur. [Naf et al., 1997] ont utilisé des squelettes de Voronoi 3D pour représenter les données médicales volumétriques. Un squelette de Voronoi est construit en déterminant en premier le diagramme 3D de Voronoi des données volumétriques puis en enlevant les faces les plus proches de la frontière de l'objet et dont la suppression ne va pas changer la topologie du diagramme. Le squelette de Voronoi 3D de l'objet est obtenu lorsque plus aucune face ne peut être enlevée. Comparé au diagramme de Voronoi originel de l'objet qui code les caractéristiques individuelles, le squelette donne une idée sur la forme générale de l'objet. Les squelettes de Voronoi ont été utilisés pour analyser les anomalies dans des IRM du lobe temporel du cerveau humain.

Le tableau (1.2) résume les différentes techniques citées ci-dessus. Il indique pour chacune d'elle le type de primitives géométriques qu'elle utilise (normale à la surface, courbure, géons, cône généralisés), sa capacité de traiter ou non les objets occlus, la taille de la base de données, le taux de reconnaissance et aussi la complexité de l'algorithme. Il est à noter que pour certaines techniques la taille de la base de données est définie par '1' pour indiquer que ces techniques ne sont pas conçues pour discriminer les objets entre eux, mais plutôt pour recouvrir la pose d'un objet connu dans la scène.

Méthode	Primitives	Occlusion locale	Nombre d'objets	Taux de reconnaissance	complexité
[Nevatia et Binford, 1977]	Cones généralisés	Oui	5	N.P	N.P
[Raja et Jain, 1992]	Géons	Non	36 synthétiques et 12 réels	0.77	N.P
Maillage extrême [Thirion, 1996]	courbure	Non	1	N.P	N.P
COSMOS [Dorai et Jain, 1997]	courbure	Non	22 synthétiques et 10 réels	0.97	N.P
Splash [Stein et Medioni, 1992]	Normales	Oui	9	1.0	$O(n)-O(n^2m^3)$
Signature du point [Chua et Jarvis, 1997]	Distance	Oui	15	1.0	N.P
Image de spin [Jonhson et Hebert, 1999]	Histogramme de surface	Oui	4	1.0	$O(k.log_2(n))$
SAI [Delingette et al., 1993]	Angle	Oui	1	N.P	N.P
[Barequet et Sharir, 1997]	Courbure	Oui	1	N.P	$O(n+k+s^3)$
[Greenspan, 1998]	Arbre échantillon	Oui	5	N.P	$O(s.log^2(n))$
[Naf et al., 1997]	Skeleton de Voronoi	Non	1	N.P	N.P

Tableau 1.2: Résumé des techniques de reconnaissance basées sur les images de profondeur

## 1.4 Conclusion

Après avoir donné la définition ainsi que les propriétés d'un objet de forme libre, nous avons dans ce chapitre présenté, d'une part les différentes représentations géométriques utilisées actuellement pour modéliser ce type d'objets, et d'autre part, nous avons détaillé les principales approches élaborées récemment pour résoudre le problème de la reconnaissance d'objets de formes libre. Nous avons constaté que la qualité des systèmes développés dépendait énormément des primitives de haut niveaux qu'ils utilisent (les silhouettes, needle-map, splash, image de spin, skeleton, etc.) afin d'apparier les objets dans les images (d'intensité et/ou de profondeur) de la scène avec les modèles de la base de données. D'après le tableau (1.2) nous remarquons que les primitives construites à partir des normales à la surface de l'objet permettent un taux de reconnaissance nettement supérieur et s'adaptent très bien aux occultations, ce qui nous laisse penser que les normales à la surface offrent un moyen efficace pour discriminer les objets de forme libre en présence ou non d'occultations.

De plus, nous avons noté que l'approche basée sur le « shape from shading » n'a pas été pleinement exploré dans la reconnaissance d'objets de forme libre malgré l'intérêt qu'elle présente. D'un autre côté, nous avons remarqué que les techniques qui possédaient une connaissance a priori du modèle de l'objet "model-based approaches" sont actuellement les plus performantes (en terme de complexité et temps de calcul), et que la représentation du modèle par polygones maillés est de loin celle qui est la plus utilisée dans les systèmes de reconnaissance.

Cette étude de l'état de l'art nous a permis d'analyser de plus près l'architecture des systèmes de reconnaissance d'objet de forme libre les plus récents, les problèmes ardues auxquels ils sont confrontés et qui n'ont pas encore été résolus d'une manière satisfaisante (tel que la complexité, l'occultation, l'efficacité,...). Notre objectif, est de concevoir un système de reconnaissance inspiré des techniques citées dans ce chapitre et qui intègre des solutions que nous avons élaborées pour résoudre quelques uns de ces problèmes sans pour autant compliquer le système. Dans la suite de cette thèse, nous présentons nos contributions à des niveaux différents du système: calibration des capteurs (source de lumière, caméra CCD et grille laser), construction d'invariants géométriques photoclino-métrique et clinométrique, construction de tables d'aspects, et les méthodes d'appariement et de recalage.



# Chapitre 2

## Outils et techniques pour la reconnaissance et le recalage d'objets 3D de forme libre

La mise en œuvre d'un système de reconnaissance d'objets 3D efficace et performant passe nécessairement par une bonne compréhension du problème, des difficultés qu'il engendre, des critères auxquels il doit satisfaire, ainsi que des divers outils et techniques disponibles dans la littérature pour sa résolution.

Ce chapitre a pour objectif de détailler les différents aspects liés à la problématique de la reconnaissance d'objets de forme libre. Ainsi, après une brève description des besoins et contraintes d'un tel système, nous présenterons les différentes méthodes mises au point pour acquérir la scène. Ensuite, nous donnerons les types de primitives discriminantes pour la reconnaissance que l'on peut utiliser dans une image (d'intensité et/ou de profondeur) de l'objet. Nous étudierons les techniques les plus utilisées pour établir des appariements modèle/données. Et enfin, nous donnerons une description détaillée du système de recalage que nous avons élaboré.

## 2.1 Problème de la reconnaissance d'objets

### 2.1.1 Définition

Ayant une scène contenant un ou plusieurs objets, et une image de cette scène prise par un capteur de vision (une caméra CCD pour une image de luminance et un scanner pour une image de profondeur), la reconnaissance d'objets implique la résolution des deux sous problèmes suivants:

**Identification** : quels sont les objets présents dans la scène ?

**Localisation** : quelle est la position et l'orientation (la pose) de chaque objet dans le repère des données (du capteur) ou bien dans le repère de la scène, l'opération qui permet de déterminer la pose de l'objet est appelée **recalage**.

### 2.1.2 Difficultés

La reconnaissance d'objets par des systèmes de vision artificielle est particulièrement difficile pour plusieurs raisons:

#### 2.1.2.1 Encombrement et occultation

Généralement, les scènes réelles contiennent rarement un objet seul et isolé. En effet, si nous jetons un coup d'œil autour de nous, nous allons rapidement nous rendre compte que l'encombrement est une chose courante plutôt qu'exceptionnelle dans une scène réelle. L'occultation partielle d'un objet d'intérêt peut être causée lorsqu'une partie de lui est cachée par d'autres objets de la scène. D'un autre côté, les propriétés de réflexion de la surface de l'objet (dans le cas d'une image d'intensité) peuvent être à l'origine d'une invisibilité dans l'image d'une région de cet objet, ce cas est considéré aussi comme une occultation. La première difficulté à laquelle doit faire face un système de reconnaissance est de pouvoir extraire des primitives de la scène qui doivent impérativement appartenir au même objet (segmentation) occlus ou non, et de les regrouper de façon pertinente pour permettre l'identification de l'objet.

#### 2.1.2.2 Variabilité de l'apparence

Cette difficulté concerne exclusivement les systèmes de reconnaissance qui utilisent une caméra. En effet, l'apparence d'un objet dans une image d'intensité varie en fonction du point de vue de la caméra: l'objet apparaîtrait dans l'image comme ayant subi une translation ou bien une rotation en fonction de la position et de l'orientation de la caméra. D'un autre côté, la forme de l'objet subit les effets de la projection perspective qui se produit lors du processus de capture de l'image par une caméra. Un autre défi pour les systèmes de reconnaissance serait alors de trouver des primitives qui restent *invariantes* au changement de point de vue, en l'occurrence ce qui nous importe ici est l'invariance projective. En effet, l'utilisation de tels invariants permet d'identifier ces primitives

(apparié) dans les différentes images d'un même objet prises à partir de différents points de vue sans aucune connaissance à priori sur le déplacement effectué. La description de ce type d'invariants sera détaillée dans le paragraphe (2.3).

### 2.1.2.3 Efficacité

La reconnaissance d'objet est un processus très coûteux en temps de calcul et en espace mémoire. Des algorithmes rapides sont requis dans tous les étages du système de reconnaissance y compris pour l'extraction de primitives, le calcul des invariants et l'appariement entre les images de la scène et les modèles de la base de données, afin d'obtenir un système de reconnaissance temps réel.

## 2.1.3 Approches

Les approches proposées pour résoudre le problème de la reconnaissance d'objets sont nombreuses, cependant aucune solution universelle n'a pu être développée jusqu'à maintenant et seules des réponses partielles avec des réussites limitées ont été appliquées. Généralement, ces approches diffèrent selon:

- a) les connaissances qu'elles utilisent: ainsi, on peut distinguer les approches basées sur la connaissance à priori d'un modèle explicite de la forme de l'objet ("*Model-based approach*"), d'autres sont basées sur le contexte dans lequel les objets peuvent être trouvés ("*Context-based approach*") ou bien basées sur la fonction pour laquelle les objets peuvent servir ("*Function-based approach*").
- b) le type d'objets utilisés : il s'agit dans ce cas des approches de reconnaissance d'objets 2D ou 3D, d'objets simples ou complexes, rigides ou déformables.
- c) le mode de représentation : on trouve généralement deux catégories : les approches basées sur une représentation centrée sur l'objet ("*Object-centered*") et les approches basées sur une représentation centrée sur l'observateur ("*Viewer-centered*") (voir § 2.5).
- d) le mode d'appariement : les techniques d'appariement utilisées dans les systèmes de reconnaissance sont classées en deux grandes classes : les techniques basées sur la géométrie des objets ("*geometry-based*") et les techniques basées sur l'apparence des objets dans l'image ("*appearance-based*").
- e) le modèle de formation de l'image : c'est à dire le type de transformation utilisée pour projeter le modèle tridimensionnel sur un plan image afin de relever ses caractéristiques. Nous distinguons principalement deux catégories de projection : la projection orthographique et la projection perspective.
- f) le type d'image : les images employées dans les systèmes de reconnaissance dépendent du système de vision utilisé, ainsi nous pouvons distinguer deux classes d'images : les images de luminance acquises à partir de caméras CCD, où chaque pixel correspond à la luminance en un point de l'espace, projeté sur le plan image de la caméra. Et les images de profondeur qui collectent une grande quantité de données en coordonnées tridimensionnelles à partir des surfaces visibles dans une scène. Chaque pixel dans l'image de profondeur représente une distance par rapport au capteur.

### 2.1.4 Critères de performances

Malgré la diversité des systèmes de reconnaissance d'objets, ils respectent tous un certain nombre de critères de performances définis pour assurer fiabilité et rapidité du processus de reconnaissance dans des applications souvent temps réel.

- 1) **Invariance** : les systèmes de reconnaissance doivent impérativement respecter la propriété d'invariance d'une manière générale, et plus particulièrement :
  - l'invariance au changement du point de vue : ils doivent parvenir à reconnaître l'objet quelque soit l'endroit à partir duquel il est observé, même occlus par lui même ou par d'autres objets se trouvant dans la scène.
  - l'invariance aux propriétés des capteurs : les systèmes de reconnaissance doivent prendre en compte dans leur architecture les erreurs dus aux bruits capteurs, aux conditions d'illumination de la scène et aux traitements bas niveaux (tel que la détection de contour) et qui peuvent induire une mauvaise reconnaissance.
- 2) **Etendue** : ce critère permet de définir le type d'objets que le système peut reconnaître ainsi que le type de scène considérée.
- 3) **Robustesse** : elle permet de définir la quantité de bruit et d'occultation dans la scène tolérée par la méthode de reconnaissance.
- 4) **Efficacité**: comme nous l'avons mentionné précédemment, ce critère définit le temps de calcul et l'espace mémoire nécessaires pour trouver l'espace des solutions.
- 5) **Précision** : enfin ce dernier critère définit la précision du système de reconnaissance en donnant :
  - le taux de reconnaissance correcte.
  - le taux des faux positifs (la reconnaissance d'objets qui ne sont pas présents dans la scène, appelée aussi reconnaissance fantôme).
  - le taux des faux négatifs (la non reconnaissance d'objets qui sont bel et bien présents dans la scène).

## 2.2 Acquisition de la scène 3D

Plusieurs méthodes d'acquisition d'objets 3D ont été élaborées ces dernières années. En fonction des capteurs de vision qu'ils utilisent, ces systèmes sont subdivisés en deux grandes classes : les systèmes utilisant des dispositifs à contact et sans contact. La figure (2.1) illustre une taxonomie tirée de [Curless et Seitz, 2000] et qui permet de présenter les différentes méthodes disponibles dans la littérature.

Les dispositifs à contact sont généralement constitués d'un bras articulé sur lequel est fixé un palpeur point par point. Les coordonnées d'un point quelconque de l'objet se déterminent aisément par la géométrie du palpeur et les codeurs des axes de mouvement du bras articulé.

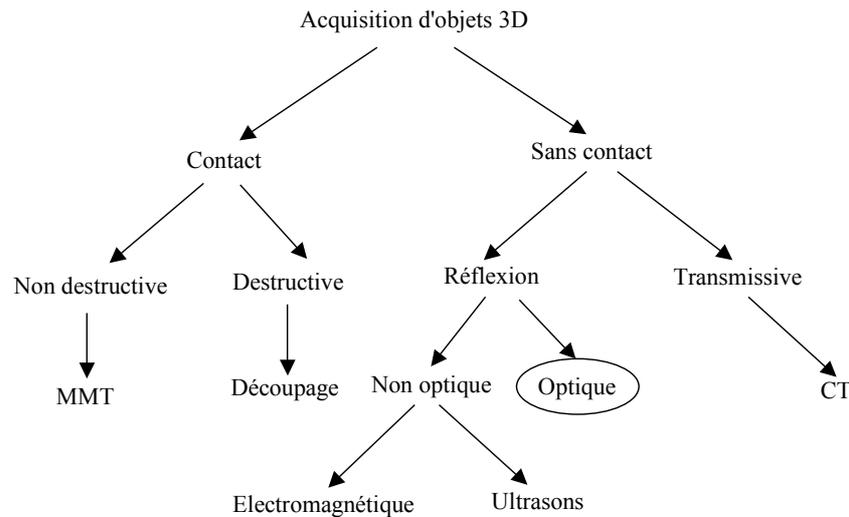


Figure 2.1 : Taxonomie des méthodes d'acquisition d'objets 3D

En effet, lorsque le palpeur touche la surface de l'objet, cet événement est signalé et la position du palpeur est enregistrée. Dans cette catégorie on trouve par exemple les machines de mesure tridimensionnelle (MMT), elles sont extrêmement précises mais restent très chères, elles sont actuellement le standard pour la digitalisation des objets dans le monde industriel. Toutefois, elles souffrent des inconvénients suivants :

- elles sont lentes (quelques points par secondes).
- elles nécessitent généralement un opérateur humain.
- elles doivent toucher la surface de l'objet, ce qui est indésirable pour les objets fragiles.

D'une façon générale, cette technologie s'applique aux cas où le nombre de points nécessaire reste faible et où le contact avec l'objet est possible.

D'un autre côté, les dispositifs sans contact opèrent de la manière suivante: ils projettent des ondes d'énergie sur l'objet puis enregistrent à chaque fois l'énergie transmise et l'énergie réfléchie. Une méthode *transmissive* très puissante souvent utilisée pour la capture de formes est la *tomographie* (CT : Computed Tomography). Elle consiste à bombarder un objet avec des rayons X très puissants puis à mesurer la quantité de radiation qui a traversé l'objet le long de diverses lignes de vue (incidences). L'image volumétrique de l'objet est ensuite obtenue par reconstruction algorithmique à partir de ces différentes mesures. Les avantages principaux de cette méthode par rapport aux méthodes réfléchives sont :

- elle est en général insensible aux propriétés de réflexion de la surface de l'objet.
- elle peut capturer les cavités internes de l'objet qui ne sont pas visibles de l'extérieur.

Toutefois, ces scanners présentent des inconvénients majeurs :

- ils sont très chers.

- des variations importantes dans la densité des matériaux (tel que du bois collé à du métal) peuvent dégrader énormément la précision.
- leur fonctionnement est potentiellement dangereux du fait qu'ils utilisent des matériaux radioactifs.

De même, les méthodes réflectives peuvent être subdivisées en deux catégories: les approches optiques et non optiques. Les approches non optiques incluent les sonars et les radars électromagnétiques. Ces systèmes sont constitués d'un émetteur et d'un récepteur. La forme de l'objet et ses dimensions sont déterminées par le temps de vol des ondes ultrasoniques ou électromagnétiques entre l'émetteur et le récepteur. Des dispositifs de modulation d'amplitudes et/ou de fréquence sont généralement utilisés dans ces systèmes. Les capteurs à ultrasons sont peu coûteux mais en revanche, ils ne sont pas très précis et présentent des temps d'acquisition très lents. D'un autre côté, les objets métalliques ne peuvent pas être saisis par des ondes électromagnétiques. Les méthodes réflectives sont souvent utilisées en imagerie médicale. L'échographie par exemple utilise la réflexion d'ultrasons sur les organes pour en donner une image tridimensionnelle.

Les dispositifs optiques permettent quant à eux de subdiviser les systèmes de vision en deux grandes classes: la vision dite active et la vision dite passive (cf. figure 2.2) [Roussel, 1999].

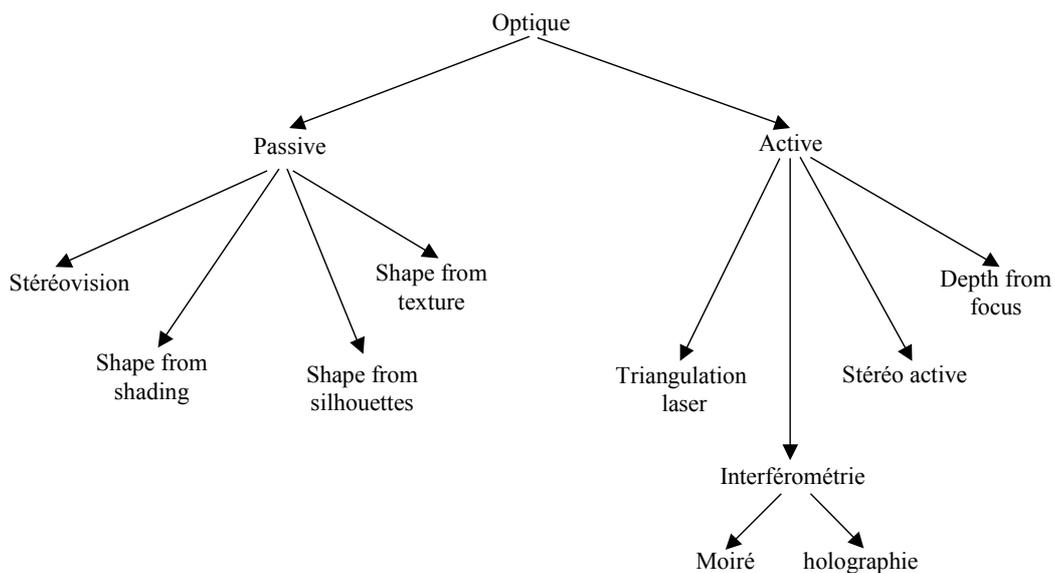


Figure 2.2: Taxonomie des systèmes de vision optiques

## 2.2.1 Vision active

Nous appelons vision active tout système de vision utilisant un capteur actif qui permet d'illuminer les objets de la scène d'une manière particulière en projetant sur leur surface des motifs structurés (un spot, un plan lumineux, une grille, etc.), ceci permet d'en extraire des informations 3D indépendamment des conditions propres d'éclairage de la scène. Ainsi, l'objectif de la vision active est de réduire la complexité des traitements à effectuer pour reconstruire la scène. En effet, les motifs projetés sur la scène sont facilement identifiables [Hu et Stockman, 1989] à l'inverse des primitives « naturelles » tel que les contours.

### 2.2.1.1 Interférométrie

Parmi ce type de méthodes, on peut citer la technique du Moiré qui consiste à projeter une lumière, blanche en général, à travers des réseaux de lignes parallèles espacées par des distances variables afin de créer une trame sur l'objet à acquérir. Le relief de l'objet est issu d'une comparaison entre deux images d'un même réseau de traits projeté une première fois sur un plan de référence et ensuite sur l'objet à analyser. La numérisation de l'objet est instantanée (environ 100 000 points par seconde) et la précision varie du dixième au centième de millimètre, en revanche, la reconstruction 3D nécessite beaucoup de calculs. Les méthodes de Moiré peuvent avoir des problèmes pour discriminer la phase lorsque la surface ne présente pas des variations lisses. Cette difficulté impose généralement une limite supérieure à l'inclinaison que peut avoir la surface de l'objet pour éviter les erreurs d'alignement "*ranging errors*".

### 2.2.1.2 Triangulation Laser

Il s'agit de projeter soit un point, soit un plan (bande étroite), soit une structure (une grille par exemple) laser de faible puissance sur la surface de l'objet. L'élément projeté est visualisé par une ou deux caméras CCD et le système calcule les coordonnées du ou des points par triangulation.

La figure (2.3a) illustre l'exemple d'une source laser qui traverse une lentille cylindrique pour créer un plan lumineux qui va éclairer une région de l'objet de la scène. La partie illuminée de l'objet est ensuite capturée par une caméra. La connaissance de l'équation du plan lumineux dans le référentiel de la scène et ses coordonnées dans le plan image permettent de calculer la position dans le référentiel de la scène de la partie éclairée de l'objet. Pour avoir un système avec une bonne résolution, on doit calibrer avec précision aussi bien la source laser que la caméra utilisée pour acquérir la scène.

Le déplacement de la source laser en translation et/ou en rotation autour de l'objet permet une reconstruction complète de ce dernier (cf. figure 2.3b). La reconstruction 3D par triangulation laser est une méthode très précise (une précision inférieure au millimètre pour des scènes d'intérieur). Cependant, elle souffre d'un problème principal qui est la structuration des données 3D reconstruites. En effet, ce système permet généralement de

disposer de plusieurs milliers de points de mesure pour une petite surface de l'objet, qu'il doit nécessairement traiter pour en extraire le maillage minimal [Curless et Levoy, 1996], [Turk et Levoy, 1994].

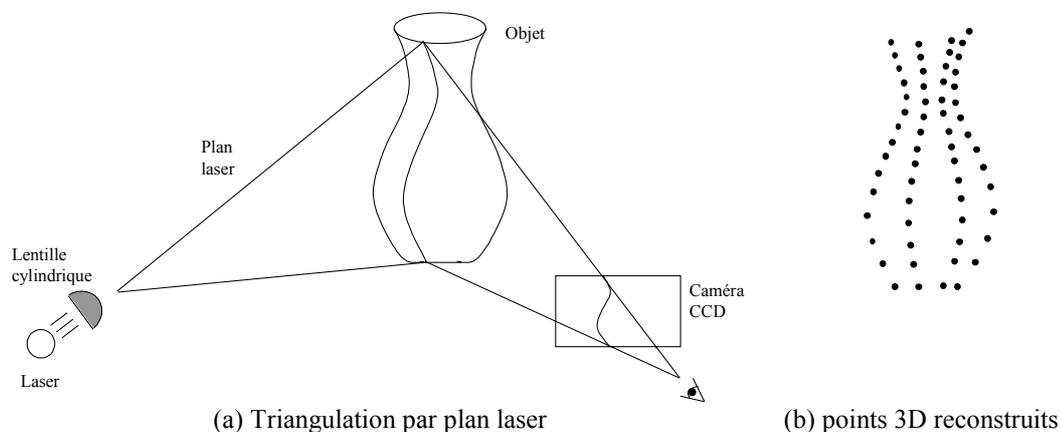


Figure 2.3 : Triangulation laser

Une autre technique de triangulation laser largement utilisée, consiste à projeter sur l'objet un motif structurant (souvent une grille). Le processus de triangulation est dans ce cas un peu plus complexe, car il s'agit d'apparier la grille projetée sur l'objet de la scène avec la grille d'origine du projecteur. En pratique, on suppose que le projecteur de la grille laser se comporte comme une caméra, ce qui permet de considérer les méthodes qui utilisent la projection d'une grille laser comme un homologue de la stéréovision dans lequel on aurait remplacé une des deux caméras par le projecteur.

### 2.2.1.3 Depth from focus

Cette technique utilise la focale de la caméra pour rechercher des informations tridimensionnelles dans une image. Pour cela, elle exploite le fait que les objets de la scène vont paraître plus ou moins nets dans l'image en fonction de leur proximité du point focal de l'objectif [Subbarao et Choi, 1995]. Il est clair que la mise en œuvre de cette technique nécessite un moyen qui permet de mesurer le degré de netteté, ce dernier peut être basé sur l'amplitude des transitions dans l'image de luminance. Ainsi, les objets qui se trouvent près du point focal (objets nets) seront caractérisés par des transitions rapides de niveaux de gris. Par contre, les objets situés en dehors du champs (objets flous) seront caractérisés par des transitions lentes. Un moyen simple pour mesurer le degré de netteté serait donc l'observation de la magnitude du gradient des niveaux de gris dans l'image de l'objet.

## 2.2.2 Vision passive

La vision passive utilise les images de luminance de la scène pour en extraire la géométrie. Pour cela, elle doit définir quelles sont les informations pertinentes dans l'image à extraire, ce qui n'est pas une tâche facile lorsqu'on connaît la complexité que présentent ces images. Les caractéristiques pertinentes pour extraire des informations dans une image de luminance sont nombreuses, les plus utilisées sont les discontinuités photométriques (les transitions clair/sombre ou inversement) qui permettent l'extraction de points, de contours ou de régions.

En vision passive, nous pouvons distinguer deux grandes familles abondamment étudiées dans la littérature : la vision monoculaire (utilisation d'une caméra fixe) et la vision binoculaire ou vision stéréoscopique (utilisation de deux caméras fixes). La vision monoculaire ne permet pas, d'un point de vue mathématique, de recouvrer l'information tridimensionnelle de la scène perçue. En effet, une caméra est caractérisée par son image bidimensionnelle formée par projection géométrique de l'espace 3D. Toutefois, la même image 2D peut s'avérer être la représentation de plusieurs scènes différentes, car la projection inverse n'est pas bijective. Afin de pallier à ce manque d'informations des hypothèses de diverses natures sont émises. Ce sont les techniques couramment appelées "*Shape From X*". On trouve alors des approches basées sur les variations globales de luminosité qui sont censées décrire les variations des normales à la surface observée, ce que l'on appelle le "*Shape From Shading*" ou SFS. On extrait alors la forme des objets en fonction des variations photométriques observées. D'autres approches utilisent des informations comme la texture dont les caractéristiques sont conservées dans l'image grâce au processus de formation d'une image, on parlera alors de "*Shape from Texture*" [Aloimonos, 1986], [Blake et Marinos, 1990]. Et enfin des approches qui extraient les silhouettes en observant les contours de l'objets sous différents angles, on parlera alors de "*shape from silhouettes*" [Mokhtarian, 1995]. Dans la section suivante, nous allons détailler un peu plus le principe du "*shape from shading*" qui nous a inspiré dans une partie de nos travaux.

### 2.2.2.1 Shape from shading

Le Shape from shading consiste à extraire une forme à partir d'une analyse des variations de la lumière observée sur la surface d'un objet. Cette méthode semble relativement naturelle pour l'être humain puisqu'elle doit faire partie du processus d'interprétation des formes [Marr, 1982].

L'hypothèse de base du "*shape from shading*" aussi appelé "*photoclinométrie*" est que l'intensité lumineuse perçue sur la surface d'un objet dépend de l'orientation de cette surface par rapport à la source de lumière éclairant la scène, en supposant que cette surface possède des propriétés de réflexion diffuse (ou Lambertienne) :

$$\begin{aligned}
 I(x,y) &= aR[p(x,y),q(x,y)] \\
 p(x,y) &= \frac{\delta z(x,y)}{\delta x} \\
 q(x,y) &= \frac{\delta z(x,y)}{\delta y}
 \end{aligned} \tag{2.1}$$

où  $a$  est l'albedo de la surface  $z(x,y)$ .  $R[p,q]$  définit donc la relation entre la photométrie  $I(x,y)$  de l'image et les variations de la normale à la surface  $(p,q)$ .  $R$  définit sur l'ensemble de l'image une carte de "reflectance" qu'il faut intégrer pour obtenir une représentation de la surface  $z(x,y)$ .

L'analyse photométrique exige d'inférer quelques hypothèses fortes sur les propriétés photométriques de la scène.

- *Source lumineuse*: puisque les normales à la surface des objets sont déterminées à partir de la direction de la source de lumière, cette dernière doit être estimée avec précision. Généralement, une source de lumière directionnelle est modélisée en plaçant une source ponctuelle loin de la scène, ce qui permet de considérer les rayons lumineux incidents comme étant parallèles.
- *Unicité de la source*: on doit utiliser une seule source de lumière à la fois pour éviter une mauvaise interprétation des variations photométriques. Si l'on a besoin d'utiliser plusieurs sources de lumière en même temps, on doit être capable de différencier l'effet de chacune dans les variations photométriques. Une autre méthode consiste à utiliser plusieurs sources lumineuses mais à des instants différents, on obtient ainsi une série d'images ; chacune d'elles correspond alors à une source particulière. Cette méthode est souvent appelée "stereo shape from shading" [Christensen et Shapiro, 1993].
- *Objets non texturés*: les objets observés ne doivent pas être texturés pour éviter que les variations de luminosité soit dûes à la texture, ce qui entraîne une erreur dans l'estimation des normales à la surface de l'objet. On parlera alors d'objets ayant une surface d'albédo constant. L'albédo est défini par le rapport de la quantité de lumière (d'énergie) réfléchiée par une surface sur la quantité de lumière (d'énergie) absorbée par cette surface [Horn, 1986].
- *Réflexion diffuse*: on suppose que la lumière incidente sur la surface de l'objet se réfléchit de manière isotrope, Cette hypothèse est vérifiée par les objets ayant des surfaces granuleuses ou mates [Foley et al., 1990].
- *Conditions aux limites*: l'analyse photométrique permet de déterminer les normales de la surface observée en chaque pixel de l'image de luminance. L'intégration de ces normales permet de reconstruire une profondeur relative de cette surface. La reconstruction de la surface réelle de l'objet nécessite la connaissance à priori des conditions aux limites.

### 2.2.2.2 Stéréovision

L'apport d'une seconde caméra permet de recouvrer l'information tridimensionnelle d'une scène sans émettre d'hypothèses quant à la nature de la scène. La vision stéréoscopique se veut la plus proche possible de la stratégie perceptive humaine, plaçant les caméras sur le même axe de vision, dans une configuration convergente (cf. figure 2.4). Chaque image droite et gauche représentent la même scène vue sous un angle différent. Par appariement (mise en correspondance) de primitives homologues dans la paire d'images, cette méthode

permet de recouvrir les trois dimensions de l'espace, par une simple triangulation, en mesurant la disparité des deux projections d'un point  $P$  de la scène.

Pour mettre en œuvre un système de vision stéréoscopique, on doit résoudre deux problèmes fondamentaux :

- L'extraction des primitives dans l'image qui serviront à la mise en correspondance, tel que les points, les contours ou bien les régions.
- La définition des critères de mise en correspondance. Ces critères dépendent étroitement de la nature des primitives choisies. Ainsi, pour des primitives ponctuelles, on utilise le critère de disparité. Par contre, pour des primitives de haut niveau tel que les contours ou les régions, on définit des critères heuristiques relatifs à la structure de ces primitives (forme, aire) [Gherbi, 1992].

Ainsi, pour permettre une mise en correspondance plus aisée, on utilise des primitives plus symboliques comme les segments de droites [Lim et Binford, 1987] ou plus généralement des invariants géométriques [Gros et al., 1995].

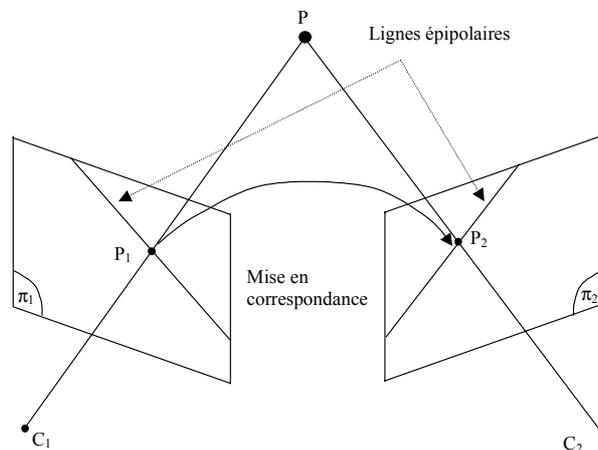


Figure 2.4 : Vision stéréoscopique

Une fois les primitives détectées dans chaque image, il faut pouvoir affirmer que la primitive  $i$  d'une image correspond à la primitive  $j$  de la seconde image, sans ambiguïté. Pour cela, plusieurs contraintes ont été développées :

- La contrainte épipolaire contraint chaque point projeté dans une image à être apparié avec un point de l'autre image appartenant à la ligne épipolaire définie par la projection dans l'autre image de la ligne de vue du premier point.
- à cette contrainte s'adjoint la contrainte d'orientation entre le vecteur directeur associé à la primitive et l'un des axes du repère image.

Ces deux contraintes permettent de sélectionner les candidats possibles à l'appariement avec une primitive, tout en minimisant leur nombre. Pour ne retenir qu'un candidat unique, d'autres contraintes sont utilisées :

- l'ordre des primitives doit être respecté dans les deux images. Par exemple, si la projection d'un point  $P_2$  se situe à gauche de la projection d'un point  $P_1$  dans l'image gauche, il doit être de même dans l'image de droite (contrainte d'ordre).
- la contrainte d'unicité des primitives doit être respectée. Dans le cas le plus général, chaque point d'une image ne possède qu'un et un seul correspondant dans l'autre image. Un cas particulier peut toutefois se présenter, si deux points se projettent en un même point sur une image, et en deux points distincts dans la seconde image.

## 2.3 Primitives

Toutes les approches de reconnaissance d'objets 3D sont basées sur l'extraction, à partir d'une ou de plusieurs images (2D ou 3D), d'entités ou de primitives géométriques qui permettront, une fois traitées, l'identification et la localisation de l'objet de la base de données vu dans l'image. L'efficacité d'une primitive peut être classifiée par :

- son pouvoir descriptif à résoudre les ambiguïtés.
- son invariance par rapport au point de vue, que l'on appelle aussi sa stabilité.

On peut ainsi affecter aux diverses primitives les caractéristiques suivantes:

- **Dimensionnalité** : Ces primitives peuvent être de différentes dimensions géométriques. Les primitives de dimension 0 ou 0D sont le point, dans le domaine analytique, et le pixel ou voxel, dans le domaine discret. Ces primitives sont insensibles aux déformations perspectives mais possèdent de faibles caractéristiques discriminatoires contrairement aux primitives de dimension supérieure. Les primitives de dimension 1 ou 1D sont la courbe (plane en 2D et spatiale en 3D) et le contour discret (ensemble mince de pixels ou voxels connexes dans l'espace). Les primitives de dimension 2 ou 2D sont la surface (plane en 2D, courbe en 3D) et la région discrète (ensemble non mince de pixels connexes en 2D ou ensemble mince de voxels connexes en 3D). Et enfin, les primitives de dimension 3 ou 3D sont le volume (spatial) et l'espace discret (ensemble non mince de voxels en 3D).
- **Taille**: Les primitives de grande taille possèdent plus de caractéristiques distinctives mais sont souvent sensibles aux variations perspectives.
- **Contraste**: les primitives qui ne se situent pas sur des variations photométriques notables sont plus difficiles à distinguer entre elles.
- **Contenu sémantique**: plus le contenu sémantique d'une primitive est grand plus il sera aisé de différencier entre eux les différents candidats à l'appariement.
- **Distinguabilité / Unicité**: Il est évident qu'une primitive qui permettrait d'éliminer tout faux appariement de part son unicité faciliterait grandement l'appariement. C'est le but recherché par les invariants projectifs, si ce n'est que ceux-ci ne sont pas des primitives à part entière et qu'ils doivent être construits à partir d'autres primitives tels des points.

On distingue généralement les primitives selon qu'elles sont locales (comme les contours ou les points) ou globales (comme les régions). On peut classifier les primitives

généralement utilisées dans les systèmes de reconnaissance en deux grandes classes : les primitives de segmentation extraites d'une image (2D ou 3D) et les primitives de modélisation définies par des expressions mathématiques [Roussel, 1999].

### 2.3.1 Primitives de segmentation

Les primitives que l'on peut extraire d'une image de luminances sont nombreuses, mais elles sont toutes définies à partir des discontinuités et/ou de continuité photométriques dans l'image de l'objet. Les primitives de segmentation les plus utilisées dans les systèmes de reconnaissance sont :

#### 2.3.1.1 Régions

Une région dans une image peut être définie comme un ensemble de pixels connexes possédant des caractéristiques semblables : par exemple leur luminosité [Gagalowicz et Monga, 1986]. La connexité est alors définie en fonction du type de voisinage et de son étendue.

L'obtention d'une description de la forme des régions dans l'image suppose l'extraction préalable de ces régions dans l'image. On parle alors de segmentation de l'image, chaque région correspond préférentiellement à un objet ou une partie de l'objet.

La définition formelle de la segmentation comporte quatre conditions que doivent respecter les sous-ensembles de pixels (régions):

1. l'union des sous-ensembles correspond à l'image complète.
2. l'intersection de deux sous-ensembles quelconques est l'ensemble vide.
3. chaque sous ensemble respecte une *condition d'uniformité*.
4. l'union de deux régions quelconques ne respecte pas la condition d'uniformité.

Il existe plusieurs méthodes pour réaliser la segmentation d'une image en régions, cependant il demeure très difficile d'arriver à une segmentation correspondant aux objets et aux parties d'objets, sauf pour les images les plus simples, comme par exemple, les images binaires d'objets isolés.

Une représentation basée sur les régions pourra donc être utile pour la reconnaissance seulement si les régions sont significatives i.e. qu'elles correspondent aux objets ou aux parties d'objets présents dans la scène. En effet, les caractéristiques de formes de ces régions pourront alors permettre de comprendre la scène et de reconnaître les objets présents. Lorsque l'image ne comporte que peu d'objets et qui sont sans occultations, des caractéristiques ou attributs globaux des régions peuvent être extraits et utilisés par le processus de reconnaissance. Par exemple, l'excentricité ou élongation (rapport du grand axe sur le petit axe), la compacité (carré du périmètre divisé par l'aire), les moments invariants (aire, centre de gravité, axes principaux d'inertie, etc.) sont tous des attributs globaux.

On peut classifier grossièrement les algorithmes de segmentation en régions suivant deux catégories :

- Les algorithmes qui réalisent une partition de l'image en utilisant par exemple une décomposition en quadtree.
- Et les algorithmes qui procèdent par une aggrégation itérative de pixels connexes possédant des caractéristiques similaires, appelés couramment les algorithmes de croissance de régions.

Dans les deux cas, le résultat de base de la segmentation est une image dont les pixels sont étiquetés en fonction de la région à laquelle ils appartiennent.

### 2.3.1.2 Contours

Les contours des objets dans des images correspondent le plus souvent aux extremas locaux du gradient (dérivée première, eq. 2.2) ou aux zéros du Laplacien (dérivée seconde, eq. 2.3) de la fonction des niveaux de gris.

$$\begin{aligned}
 \text{dérivée en } x : \quad & I_x(x,y) = \frac{\delta I(x,y)}{\delta x} \\
 \text{dérivée en } y : \quad & I_y(x,y) = \frac{\delta I(x,y)}{\delta y} \\
 \text{magnitude du gradient :} \quad & G(I(x,y)) = \sqrt{I_x^2 + I_y^2} \\
 \text{direction du gradient :} \quad & \theta(I(x,y)) = \arctan\left(\frac{I_y}{I_x}\right)
 \end{aligned} \tag{2.2}$$

$$\text{laplacien :} \quad L(x,y) = \frac{\delta^2 I}{\delta x^2} + \frac{\delta^2 I}{\delta y^2} \tag{2.3}$$

Les difficultés de la détection des contours proviennent du bruit important présent dans les images de luminance. Il est donc nécessaire de définir des critères de performance d'un opérateur incluant une modélisation des contours recherchés et la notion de bruit. La combinaison du filtrage et de la dérivation définit alors l'opération comme une opération de filtrage linéaire discret. Etant donnée une image  $I(x,y)$  l'opération de filtrage consiste à convoluer cette image par une fonction  $f(x,y)$  représentant la réponse impulsionnelle du filtre dérivateur :

$$I_f(x,y) = (f * I) = \sum_{x'=-k/2}^{k/2} \sum_{y'=-k/2}^{k/2} f(x-x', y-y') I(x', y') \tag{2.4}$$

$I$  : image de taille  $(m \times n)$

$I_f$  : image filtrée de taille  $(m \times n)$

$f$  : réponse impulsionnelle du filtre de taille  $(k \times k)$

On notera que le filtrage linéaire dans ce cas consiste à remplacer chaque niveau de gris par une combinaison linéaire des niveaux de gris des points voisins ; les coefficients de cette combinaison linéaire sont définis par la réponse impulsionnelle du filtre.

Le contour discret peut être décrit de façon exhaustive par une liste ordonnée de coordonnées des points formant le contour. Une information comprimée (sans perte) de cette information est le code en chaîne ("*chain-code*") [Haralick et Shapiro, 1993]. Celui-ci donne, à chaque point du contour, un code (un nombre entre 0 et 7) correspondant à la direction à suivre pour atteindre le prochain point du contour. Les coordonnées du premier point du contour (choisi arbitrairement) sont conservées. On peut calculer le chain-code d'un contour discret en utilisant la connexité 8-voisins de l'image (ou 4-voisin selon le cas). Il est possible de calculer certaines caractéristiques globales d'un contour directement à partir de son chain-code (tel que l'aire de la surface délimitée par le contour).

Pour arriver à reconnaître un objet, il faut comparer un modèle qui est souvent de nature analytique avec une image qui est de nature numérique et discrète. Une solution pour la reconnaissance serait ainsi d'approximer le contour 2D discret de l'image au moyen de primitives ou fonctions analytiques ajustées ("*fitted*") sur ses points. Pour ce faire, on peut utiliser la courbe *psi-s* [Ballard, 1982] (cf. figure 2.5). Celle-ci donne l'orientation de la tangente en chaque point (par rapport à un axe fixe) en fonction de sa position sur le contour, c'est à dire la longueur d'arc. Dans cette représentation, un segment de droite est représenté par un segment horizontal tandis qu'un arc de cercle est représenté par un segment de droite non horizontal. Lorsque le contour est fermé, la courbe *psi-s* est périodique avec une discontinuité entre 0 et  $2\pi$ .

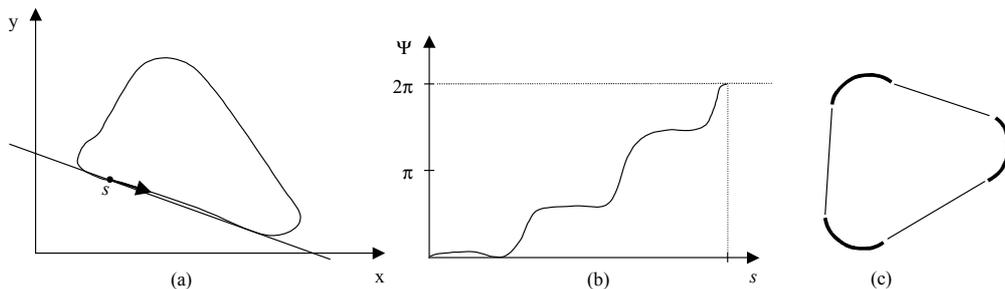


Figure 2.5 : Segmentation par courbe *psi-s*. (a) courbe triangulaire. (b) la courbe  $\Psi$ -s montrant les régions de forte courbure. (c) Résultat de la segmentation.

Dans le même ordre d'idée, la transformée de Fourier (FFT) d'une fonction complexe périodique obtenue par les deux coordonnées des points d'un contour fermé peut être tronquée à un certain nombre de termes que l'on appelle *descripteurs de Fourier* [Persoon et Fu, 1977]. Cette représentation est de type global. Elle n'est donc valide que pour les contours fermés et complets.

Une représentation basée sur le découpage des contours en morceaux de formes simples est préférable car plus généralement applicable. Cependant, il peut être difficile de représenter la forme complexe d'un morceau au moyen d'une seule fonction analytique ayant un nombre limité de paramètres, surtout si le morceau comporte des points de discontinuité. Souvent, on choisira des primitives 1D de forme simple. Par exemple, on pourra segmenter un contour en primitives de courbure constante, segments de droites et arcs de cercle, sur la

base de sa courbe  $\psi$ -s ou par une autre méthode. La représentation la plus simple d'un contour par morceaux reste cependant celle formée seulement d'éléments linéaires connexes. Le contour est donc approximé par un arc polygonal, c'est à dire une séquence de segments de droites parfois appelées "*polyline*".

Dans le choix d'une représentation d'une primitive, il faut toujours se rappeler que le but visé est la reconnaissance et non nécessairement une fidélité visuelle maximale de l'approximation.

### 2.3.1.3 Points d'intérêts

Les points d'intérêts sont définis comme des points singuliers dans une image. On peut donner comme exemple les points de jonction entre plusieurs régions ou encore les points anguleux d'une région homogène. Par ailleurs, des points particuliers le long d'une courbe comme les points de convexité pointue ou de concavité profonde (coins), de courbure élevée, de changement du signe de la courbure (points d'inflexions), peuvent aussi être considérés comme points d'intérêts.

En général, on distingue principalement deux types de points d'intérêt :

- les points extraits directement à partir des images.
- les points extraits de primitives plus évoluées comme les contours par exemple.

Ainsi, lorsqu'il s'agit d'extraire des primitives ponctuelles, la caractéristique que l'on cherche souvent à améliorer est leur position dans l'image (subpixel ou pas). En effet, Cette exigence de précision sera fondamentale pour les systèmes de reconnaissance qui utilisent les points isolés dans leur processus d'appariement. Les points d'intérêt sont souvent caractérisés par des coins ou des jonctions dans le niveaux de gris.

Les points d'intérêt extraits d'autres primitives correspondent principalement à des points particuliers que l'on peut extraire d'une courbe discrète et donc en particulier d'un contour. Ces points servent surtout à caractériser une première segmentation pour faciliter la reconnaissance. En effet, ces points particuliers seront caractérisés par, d'une part, leur appartenance à une primitive de haut niveau (courbe, contour), et d'autre part, leur faible occurrence dans ces primitives. Pour détecter ce type de points, on utilise principalement la variation de la courbure le long d'une courbe discrète. Ainsi, les coins seront isolés par les maximums de courbure et les inflexions par un passage par zéro de la courbure.

Dans le cas d'un contour discret, l'opération de dérivation est remplacée par une opération de calcul des différences. Les points particuliers correspondent alors aux points ayant une différence d'orientation élevée avec leur voisins précédents ou suivants. On appelle souvent *coins* ces points particuliers sur un contour discret. Dans ce processus, il ne faut pas oublier qu'un contour discrétisé présente souvent dans sa suite de pixels connexes plusieurs changements de direction "en escalier" qui ne correspondent pas toujours à des discontinuités de la tangente sur le contour continu de l'objet. Ainsi, on filtrera souvent le signal discret pour adoucir les fausses discontinuités résultant du processus de quantification. Ce filtrage produit un contour discret dont les points se retrouvent à des positions intermédiaires ou interpolées entre les centres des pixels. C'est un contour dit

*sub-pixel*. En pratique, on détecte sur les contours préalablement filtrés les points où il y a des *extrema locaux significatifs* de la courbure discrète. Ces points représentent les coins extraits du contour. Ce sont les points où on estime qu'il y a une discontinuité dans l'orientation de la tangente au contour discret.

[Richard et Hoffman, 1985] ont étudié les contraintes sur la formation de contours courbes et lisses ouverts ou fermés. Ces contraintes permettent de générer des modèles qualitatifs des objets 2D, c'est à dire des modèles qui assignent à chaque objet une parmi un nombre fini de catégories ou de classes possibles. Dans la représentation proposée le contour 2D d'un objet donnée est découpé de façon unique en différents morceaux, chacun étant délimité par deux extrema de courbure. Au total, il y a cinq types de morceaux, appelés *codons*, en fonction du nombre de points d'inflexion sur le morceau et du signe de la courbure le long du morceau. Un objet est donc décrit par un code donnant la liste des codons rencontrés en balayant le contour au complet. Chaque code correspond à une catégorie ou classe différente d'objets. Dans cette approche, les contours considérés sont lisses c'est à dire sans discontinuité. Ils proviennent donc d'objets naturellement lisses ou dont les contours sont lissés par filtrage.

L'extraction des points d'intérêts sur un contour discret dépend du calcul robuste des tangentes et courbures locales. Ce dernier problème n'est pas facile car il est souvent relié au choix de modèles mathématiques appropriés pour l'approximation des données et à l'estimation robuste des paramètres de ces modèles en présence de bruit.

#### 2.3.1.4 Approche hybride

Cette approche permet de construire, à partir de primitives 2D simples, une description moins abstraite de l'image que de simples chaînes ou ensemble de points difficilement manipulables. La majorité des techniques hybrides font appel à la notion de *groupement perceptuel* ou *organisation perceptuelle* dont [Marr, 1982] et [Lowe, 1985] sont les précurseurs. L'objectif recherché est la description de l'image par association de primitives liées entre elles par des propriétés géométriques et/ou topologiques.

#### 2.3.1.5 Squelette

Lorsque des objets complexes font partie de l'image, il peut être intéressant de les décomposer en parties simples, dans le but de diminuer la sensibilité aux occultations d'un algorithme de reconnaissance. Une façon de décomposer un objet isolé dans une image est de passer par une description intermédiaire, le squelette. Le squelette d'une région est défini par l'ensemble des centres des disques maximaux qui sont complètement inclus dans la région et qui touchent à son contour en deux points ou plus. En d'autres termes, le squelette est formé des centres de disques inclus dans la région et qui ne sont pas inclus dans aucun autre disque du squelette. Cette définition est connue sous le nom de SAT ("*symetric axis transform*") ou MAT ("*medial axis transform*") [Blum, 1967]. En vue de permettre la reconstruction de la région, la représentation par squelette comporte généralement deux éléments:

- la position du centre de chaque disque maximal.

- le rayon de chacun de ces disques.

Dans le domaine continu, le squelette d'une région bornée par une courbe 2D fermée sera formé d'un réseau de courbes continues par morceaux. La segmentation en différentes parties de la région peut être basée sur les points de jonction de ce réseaux de courbes. Dans le domaine discret, le squelette d'une région bornée par un contour discret sera formé d'un nombre fini de centres de disques maximaux. Un exemple trivial est le MAT d'une région circulaire qui est formé d'un seul point (son centre) avec le rayon associé. Lorsque la région est un polygone convexe, les branches du squelette sont des segments linéaires.

Le principe de base pour le calcul du squelette est de trouver les points de la région dont la distance minimale au contour est obtenue en deux endroits différents. En pratique, on calcule la *transformée de distance* de l'objet et le squelette comprend les maxima locaux de cette fonction. La transformé de distance assigne à chaque point de la région la distance minimale entre ce point et un point du contour. Un maximum local est défini comme un point dont la distance au contour est supérieure ou égale à celle de tous ses voisins immédiats. Le choix d'un voisinage à 4 ou à 8 peut influencer la forme du squelette obtenu.

Une définition différente a été proposée par [Brady et Assada, 1984] pour pallier à l'inconvénient de la sensibilité au bruit. C'est l'axe des symétries locales ou SLS ("smooth local symetries"). Une symétrie locale est définie comme le point milieu d'un segment  $BA$  joignant une paire de points  $A$  et  $B$  sur le contour tel que l'angle  $\alpha$  entre  $BA$  et la normale en  $A$  est le même que l'angle  $\beta$  entre  $BA$  et la normale en  $B$  (cf. figure 2.6).

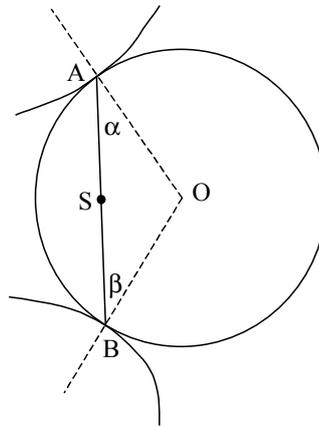


Figure 2.6 : Définition d'un point du squelette SLS

Le calcul du SLS peut se faire point par point. Dans ce cas, chaque point du contour est évalué avec tous les autres points du contour pour déterminer s'il forment une symétrie locale. Cette approche est assez coûteuse en temps de calcul. Une autre approche consiste à approximer d'abord le contour aux moyens de primitives simples, par exemple des primitives de courbure constante. La symétrie locale de tous les points d'un arc est ensuite évaluée simultanément, limitant ainsi la complexité algorithmique.

Finalement, un dernier groupe de méthodes, définies seulement dans le domaine discret, utilise plutôt le principe de l'*amincissement* de la région. Dans cette approche, on efface les points du contour de façon itérative tant que ceux-ci ont plus d'un voisin sur l'objet et que leur élimination ne rend pas l'objet non connecté localement.

## 2.3.2 Primitives de modélisation

Ce type de primitives est construit à partir de modèles prédéfinis possédant suffisamment de paramètres d'une part, pour représenter les caractéristiques que l'on recherche dans une image, et d'autre part, pour pouvoir distinguer ces primitives entre elles lors de l'appariement. Ces modèles apportent un niveau d'abstraction supplémentaire par rapport aux primitives classiques telles que les contours ou les régions qui n'ont d'existence que dans les images. Parmi les primitives de modélisation les plus utilisés, on peut citer:

### 2.3.2.1 Segments de droites

Le segment de droite représente la plus simple des primitives modélisées, il est caractérisé par la position de son centre (ou de ses extrémités), son orientation et sa longueur. Les segments de droite sont souvent extraits d'une approximation polygonale d'un contour. [Thirion et Ronse, 1996] utilisent des segments appariés dans le cadre d'un système où la caméra est en translation pour, d'une part, réaliser l'autocalibration d'un tel système, et d'autre part, reconstruire les segments de droite dans l'espace.

### 2.3.2.2 Arcs paramétriques

Les arcs paramétriques sont un modèle intéressant, ils représentent des contours (et non plus seulement leur approximation polygonale) dont la paramétrisation reste relativement concise. Ils permettent d'envisager une reconstruction qui tout en restant structurée comme une ligne brisée (par les points de contrôle) n'est déjà plus polygonale.

### 2.3.2.3 Contours actifs

Les contours actifs permettent, d'une part, de construire des primitives modélisées de haut niveau, et d'autre part, de résoudre le problème de la segmentation. Les contours actifs ont pour capacité de venir se positionner sur les caractéristiques que l'on recherche dans une image (maximum de gradient ou passage par zéro du laplacien) en minimisant une énergie potentielle (le champ de gradient par exemple) par la dissipation de leur énergie cinétique (déplacement du contour actif). Le principe général des contours actifs introduit par [Kass et al., 1988] réside donc dans la résolution d'une équation de mouvement et peut ensuite être appliqué à toutes sortes de courbes qui apportent alors le pouvoir d'abstraction nécessaire aux modèles évolués.

### 2.3.2.4 Primitives composites

Une primitive composite combine de façon hiérarchique un ou plusieurs exemplaires de différentes primitives géométriques en un seul modèle comprenant plusieurs parties constituantes. Pour créer une telle primitive, il faut spécifier les paramètres de composition c'est à dire les orientations, positions, échelles, connexions et déformations des primitives constituantes. Différentes règles de composition sont possibles.

**Composition booléenne:** les primitives sont combinées par des opérations d'union, d'intersection et de différence entre leurs ensembles de points. On peut tracer un arbre binaire où les *feuilles* sont des exemplaires des primitives et les autres *nœuds* des opérations de composition. La racine de l'arbre est la forme composite.

**Composition de frontières:** Dans ce cas, une primitive n-D est obtenue par la composition de primitives (n-1)-D. Par exemple la surface continue par morceaux d'un solide (tel que les surfaces planes et courbe d'un cylindre) est une représentation par composition de frontière ou B-REP ("boundary representation"). Un autre exemple est la formation d'un contour 2D au moyen de morceaux simples comme des segments de droites et des arcs de cercles.

**Composition balayée ("sweep"):** dans ce type de composition, une primitive (n+1)-D est définie en terme de trois éléments, d'abord la section qui est une primitive n-D, ensuite l'axe qui est une courbe 1-D le long de laquelle la section est balayée, et enfin la fonction de transformation du balayage, qui sert souvent à faire une rotation, une expansion ou une contraction de la section le long de l'axe. Lorsque la section est 2D, le volume obtenu est le cylindre généralisé.

## 2.4 Invariance géométrique dans la reconnaissance d'objets 3D

### 2.4.1 Définition

Un invariant projectif est une quantité calculée à partir de primitives extraites à chaque fois des images de l'objet et qui a la même valeur quelque soit le point de vue à partir duquel ces images ont été prises. L'utilisation des invariants a deux avantages essentiels : premièrement, l'acquisition des modèles devient plus facile dans la mesure où ils peuvent être acquis directement à partir des images sans avoir besoin d'effectuer des mesures sur les objets eux mêmes. Deuxièmement, les invariants peuvent être utilisés comme des fonctions d'indexation "*index functions*" de telle sorte que la valeur mesurée dans l'image indexe directement un modèle de la base des modèles, ce qui rend le temps de reconnaissance indépendant du nombre de modèles.

Les invariants peuvent être utilisés pour l'appariement d'un modèle avec une image, le modèle pouvant être lui aussi une image. On extrait du modèle des caractéristiques auxquelles on associe un invariant, puis on recherche des parties de l'image présentant le même invariant. Les invariants peuvent aussi servir à compléter un appariement incomplet.

Par exemple, lorsqu'on observe quatre points alignés avec une caméra, si on connaît le birapport de ces quatre points, il suffit d'avoir identifié trois des points sur l'image de la caméra pour pouvoir situer le quatrième.

Par ailleurs, la description de modèles basée sur l'invariance est plus compacte et offre une certaine robustesse contre les occultations partielles (quand les primitives utilisées sont locales). Une limitation, cependant, est que les invariants sont connus seulement pour certaines classes de formes d'objets (polyédriques, de révolution), et que la définition d'invariants pour des objets quelconque n'est pas du tout évidente.

### 2.4.2 Types d'invariants

Les invariants fournissent un outil efficace pour résoudre l'une des difficultés centrales de la reconnaissance d'objets, à savoir la variabilité de l'apparence (cf. § 2.1.2.2).

**Définition :** d'une manière plus formelle, les invariants peuvent être définis comme des propriétés de configurations géométriques qui restent inchangées pour certaines classes de transformations. Dans le contexte de la vision par ordinateur nous nous intéressons particulièrement à la détermination des invariants à la transformation perspective. Soit par exemple  $I = I(f_g)$  une propriété (fonction) d'un groupe de primitive  $f_g$ , et  $T$  une transformation projective, alors  $I$  est un invariant sous la transformation  $T$  si et seulement si :

$$I(f_g) = I(T(f_g)) \quad (2.6)$$

Par exemple, la distance (longueur) entre deux points est invariante pour une transformation euclidienne (translation + rotation), par contre elle peut prendre une valeur quelconque sous une transformation perspective.

Le groupe de primitives utilisé dans le processus de reconnaissance peut être un ensemble de l'objet, des points d'intérêts, ou simplement la silhouette entière de l'objet. Les invariants qui sont définis à partir de points seuls de l'objet sont dits invariants *locaux*, par contre ceux qui nécessitent un nombre de points à proximité l'un de l'autre sont appelés invariants *semi-locaux*. Les invariants *globaux*, quant à eux, sont définis à partir de la forme entière de l'objet. Généralement, les invariants locaux et semi-locaux sont préférés à cause de leur tolérance aux occultations partielles. En général, un nombre suffisant d'invariants locaux (semi-locaux) doit être utilisé pour décrire un objet; ainsi dans le cas où plusieurs primitives seraient occultées, la détermination d'un seul invariant suffirait pour reconnaître l'objet.

Les invariants globaux conviennent mieux aux applications où les objets à traiter ne sont pas occultés. Ils sont particulièrement intéressants lorsque les primitives nécessaires pour calculer les invariants locaux/semi-locaux sont difficilement voire impossible à extraire des objets. Ils sont aussi très utiles pour donner une description cohérente de l'objet.

Plusieurs formes d'invariants peuvent être construites en fonction des différents groupements de primitives utilisés. Le choix du type d'invariant à utiliser dans un système

de reconnaissance dépend largement de deux facteurs : les caractéristiques géométriques de la classe d'objets utilisée, et les conditions sous lesquelles ces objets sont vus. Malheureusement, les invariants sont connus seulement pour une catégorie limitée d'objet (par exemple un ensemble de points 3D coplanaires). En général, la projection de l'objet 3D dans une seule image 2D ne possède pas d'invariants géométriques projectifs ([Burns et al., 1990], [Huttenlocher et Kleinberg, 1994]); en effet il n'y a pas assez d'information dans une image 2D pour compenser l'information manquante sur la profondeur et ce en utilisant uniquement des méthodes géométriques. Des invariants peuvent être cependant construits pour quelques classes d'objets 3D du moment que l'on dispose de connaissances additionnelles sur leurs contraintes de forme. Quatre grandes classes d'invariants géométriques sont généralement utilisées dans les problèmes de reconnaissance :

#### 2.4.2.1 Invariants algébriques

Ces invariants sont calculés à partir de primitives algébriques telle que les points d'intérêts, les droites ou les coniques. Un exemple simple d'un invariant algébrique est le birapport ("*cross ratio*") défini par 4 points colinéaires (cf. figure 2.7), ou par 4 droites concourantes. [Rothwell et al., 1995] ont utilisé avec succès les invariants algébriques comme fonctions d'indexation pour identifier des objets plans caractérisés par des droites et des coniques coplanaires. Les invariants algébriques sont semi-locaux dans le sens où ils extraient l'information invariante à partir de plus d'un point mais ne nécessitent pas pour autant que la forme entière de l'objet soit visible.

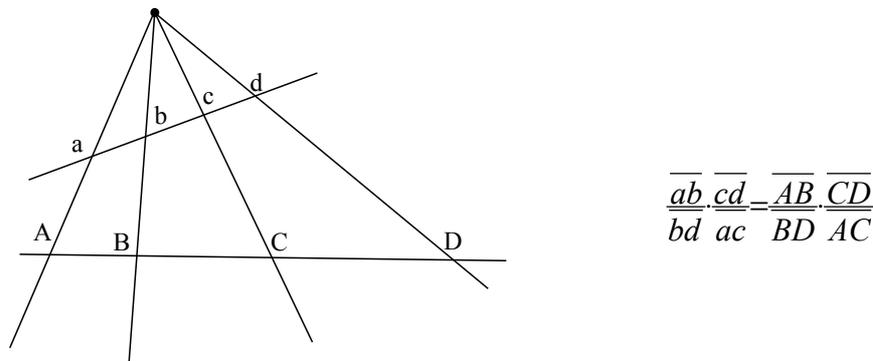


Figure 2.7 : Définition du birapport

#### 2.4.2.2 Invariants différentiels et semi-différentiels

Ces invariants sont généralement utilisés dans les méthodes de reconnaissance basées sur les contours de l'objets. Les invariants différentiels sont complètement locaux. Ils sont uniquement fonction de la position ainsi que des dérivées de la courbe en un point donné. Ces invariants surviennent naturellement lorsque les points de la courbe,  $x$ , sont exprimés par une fonction d'un paramètre  $t$ ,  $x=x(t)$ , plutôt que par une fonction implicite. Les descripteurs de forme les plus naturels pour ce type de représentation sont donnés par

$\frac{d^n x_i}{dt^n}$ . Ainsi, un invariant différentiel sera défini en fonction de  $\frac{d^n x_i}{dt^n}$ . Les invariants différentiels sont complets : un ensemble réduit d'invariants contient l'essentiel de l'information sur la courbe. De plus leur caractère local les rend insensibles aux occultations. Malheureusement, leur calcul nécessite la connaissance des dérivées d'ordre supérieur, des dérivées d'ordre 7, par exemple, pour les transformations projectives de plans, et des dérivées d'ordre 5 pour une transformation affine [Zisserman et al., 1995], ce qui les rend inapplicables pour des images réelles généralement bruitées et sur lesquelles les dérivées d'ordre supérieures sont alors inexploitable en raison du bruit.

D'un autre côté, les invariants semi-différentiels nécessitent des dérivées d'ordre inférieur. Ils sont semi-locaux, basés sur un compromis entre le processus de recherche des points d'intérêt et le calcul des dérivées d'ordre supérieur des contours extraits ([Brill et al., 1992]; [Moons et al., 1995]; [Pajdla et Van Gool, 1995]; [Weiss, 1993]).

#### 2.4.2.2 Invariants de moments

Les invariants de moments fournissent une caractérisation (classification) de la forme de l'objet qui permet de le reconnaître. [Hu, 1962] fût le premier à établir le fondement mathématique des invariants de moments bidimensionnels et à démontrer leur application pour la reconnaissance de formes. [Dudani et al., 1977] ont utilisé les invariants de moments pour reconnaître les formes d'avions, ils ont démontré que cela rendait le processus de reconnaissance plus rapide et plus fiable. Les valeurs de ces invariants de moments restaient inchangées pour une translation, une rotation ou un changement d'échelle de la forme de l'objet.

Les invariants de moments sont globaux et donc sensibles aux occultations. Toutefois, Ils peuvent être considérés comme des invariants locaux lorsqu'une segmentation de l'image en régions représentant des formes partielles de l'objet est possible [Aström, 1993]. Bien que les invariants de moments ont été principalement utilisés dans la reconnaissance d'objets 2D et dans la description de formes, leur utilisation a été étendue aux images 3D [Yang et al., 1997]. Les avantages des invariants de moments sont dus essentiellement à la facilité avec laquelle ils peuvent être utilisés pour représenter un objet, et aussi au fait qu'ils permettent d'obtenir un recalage robuste de l'objet représenté dans l'image, à partir de certains types de moments *orthogonaux*, tel que les moments de *Zernike* [Khotanzad et Lu, 1990]. L'existence de cette propriété de recalage est très importante car elle suggère un appariement non-arbitraire dans l'espace des invariants, et de ce fait permet une interprétation significative des similarités et des différences dans cet espace. Malgré tout, le calcul d'un ensemble de ces moments reste très coûteux.

#### 2.4.2.3 Repères canoniques

Un repère canonique est un moyen de normalisation de courbes. Cette technique procède par la détermination d'une transformation  $T$  qui projète les caractéristiques distinctives d'une courbe, tel que des points ou des droites, dans un repère dans lequel ces

caractéristiques possèdent un emplacement particulier. Pour une transformation projective par exemple, le nombre minimal de points distinctifs nécessaires est 4, ces points peuvent être projetés en quatre coins d'un carré [Rothwell et al., 1992]. Dans le cas affine, trois points seulement sont nécessaires. La transformation  $T$  est ensuite appliquée à la courbe qui contient les points distinctifs pour produire une courbe dans le repère canonique; toute les courbes équivalentes (qui correspondent à la même courbe de l'objet) auxquelles on applique la transformation  $T$  vont être projetées en une seule courbe qui représentera donc une signature invariante. Toute mesure, telle que la longueur d'un arc, le maximum, le minimum, l'inflexion ou même une ordonnée devient une caractéristique invariante. Les repères canoniques sont des invariants semi-locaux car ils dépendent de plus d'un point.

[Rothwell et al., 1995] ont utilisé les repères canoniques projectifs pour reconnaître une classe d'objets plans 2D à partir desquels ils pouvaient extraire des courbes lisses non-convexes. Les repères canoniques affines ont aussi été utilisés par [Pillow et al., 1995] pour reconnaître une classe particulière d'objets 3D, telle que les pipes, connue sous le nom de surface de canal ("*canal surfaces*").

Par ailleurs il existe aussi d'autres invariants projectifs à caractère non géométriques mais qui peuvent être utilisés pour faciliter la mise en correspondance : Ainsi, la réflexion diffuse d'un objet et la quantité de lumière réfléchi isotropiquement qui en découle, est elle aussi un invariant projectif puisqu'elle ne varie pas suivant le point de vue (contrairement à la réflexion spéculaire, voir chapitre 3).

## 2.5 Modes de représentation des objets 3D

La mise en correspondance entre l'image et un modèle implique que leurs descriptions respectives par les primitives citées précédemment soient effectuées dans des repères homogènes. Ce paragraphe est ainsi dédié à la description de ces repères et des contraintes, avantages et difficultés qu'ils présentent.

### 2.5.1 Représentation centrée sur l'objet

Cette représentation consiste à associer un repère fixe à un objet, ou pour un objet décomposé en partie élémentaires, plusieurs repères associés à chacune des parties et référencés dans le repère principal. Un objet représenté dans un repère centré sur l'objet propose une description précise et concise. Cette description est unique pour un objet ce qui présente l'avantage d'un stockage en mémoire peu coûteux mais apporte une difficulté essentielle : lors de la mise en correspondance, il devient indispensable de reconstruire, à partir des données image, l'objet dans son repère 3D afin de pouvoir le comparer aux différents modèles. Toutefois, afin d'éviter cette reconstruction, une solution alternative consiste à déduire une description bidimensionnelle de l'objet dans l'image et de projeter le modèle 3D sur un plan 2D afin d'établir la mise en correspondance durant le processus de reconnaissance. Cette projection impose la connaissance du modèle de projection de la caméra et fait intervenir la notion de point de vue et d'auto occultations. Malgré les difficultés apportées par ce mode de représentation, de nombreux travaux ont été développés avec des descriptions centrées sur l'objet. Ainsi [Lowe, 1987] propose dans son

système SCERPO d'établir directement une mise en correspondance entre les modèles 3D et les caractéristiques 2D de l'image.

### 2.5.2 Représentation centrée sur l'observateur

Cette représentation reprend l'idée de la projection 2D du modèle 3D d'un objet. Elle décrit ainsi un objet 3D par des vues caractéristiques 2D que l'on nomme communément des aspects (voir § 2.1.3). Chaque aspect ou vue caractéristique décrit comment l'objet apparaît à partir d'un simple point de vue ou d'une gamme de plusieurs points de vue donnant des configurations visuelles similaires. Cette représentation a pour avantage d'être simple et rapide lors de la mise en correspondance grâce à l'homogénéité des dimensions du modèle et de l'image. Cependant, cette technique présente la difficulté suivante : même pour un objet simple ayant une structure géométrique peu complexe, de nombreuses vues caractéristiques sont nécessaires afin de déterminer l'ensemble des descriptions différentes selon les points de vues possibles. Ceci implique un stockage en mémoire beaucoup plus important que pour la représentation centrée sur l'objet. Ainsi, des stratégies de regroupement de vues similaires ou partageant des structures communes doivent être appliquées afin de réduire cet espace [Basri, 1993], [Horaud et Sossa, 1995]).

## 2.6 Techniques de Mise en correspondance pour la reconnaissance d'objets 3D

Le problème de la mise en correspondance ou d'appariement entre le modèle et la scène est sans nul doute le problème le plus difficile à mettre en œuvre dans les systèmes de reconnaissance. Connaissant le modèle de l'objet, il s'agit de trouver dans l'image (de luminance ou de profondeur) les primitives qui font partie de ce modèle afin d'établir la reconnaissance de l'objet dans la scène.

Dans la pratique, même si le modèle de l'objet recherché est connu, l'appariement est loin d'être trivial, c'est un problème fortement combinatoire surtout dès que les modèles des objets traités sont complexes.

D'une manière générale, on retrouve trois classes de méthodes d'appariement:

1. appariement de primitives 3D avec un modèle géométrique 3D.
2. appariement de primitives 2D entre elles
3. appariement de primitives 2D avec un modèle géométrique 3D.

Dans un contexte applicatif, si on considère le problème de reconnaissance d'objets 3D décrits par un modèle géométrique, les approches envisageables sont l'appariement de primitives 3D-3D ou l'appariement 2D-3D.

Dans la littérature scientifique, de nombreuses méthodes sont proposées. Nous n'avons pas la prétention d'établir une liste exhaustive de toutes les méthodes existantes. Nous

souhaitons juste décrire, à travers des exemples variés, les principales approches envisageables et donner les critères de choix relatifs à chaque technique.

### 2.6.1 Arbres de recherche ou d'interprétation

Les algorithmes de recherche dans les arbres sont basés sur la construction d'un arbre représentant l'ensemble des appariements possibles entre les données images et les primitives du modèle [Grimson, 1990]. On parle aussi d'arbre d'interprétation (cf. figure 2.8). Ces algorithmes sont très pénalisant de part le coût de recherche élevé engendré par le balayage de toutes les hypothèses d'appariement référencées dans l'arbre. La complexité du problème croît exponentiellement avec la quantité de données à traiter.

Afin d'éviter ce problème, une solution consiste à limiter au maximum le domaine de recherche. Il s'agit d'opérations d'élagage ("*pruning*" dans le cas des arbres) réalisées par des algorithmes basés sur l'élimination des couples appariés indésirables. Ces méthodes agissent par propagation de contraintes géométriques ou topologiques dans l'arbre. Ce type d'algorithmes a été introduit par [Montanari 1974] puis ensuite repris et généralisé par [Freuder, 1978] et [Haralick, 1979].

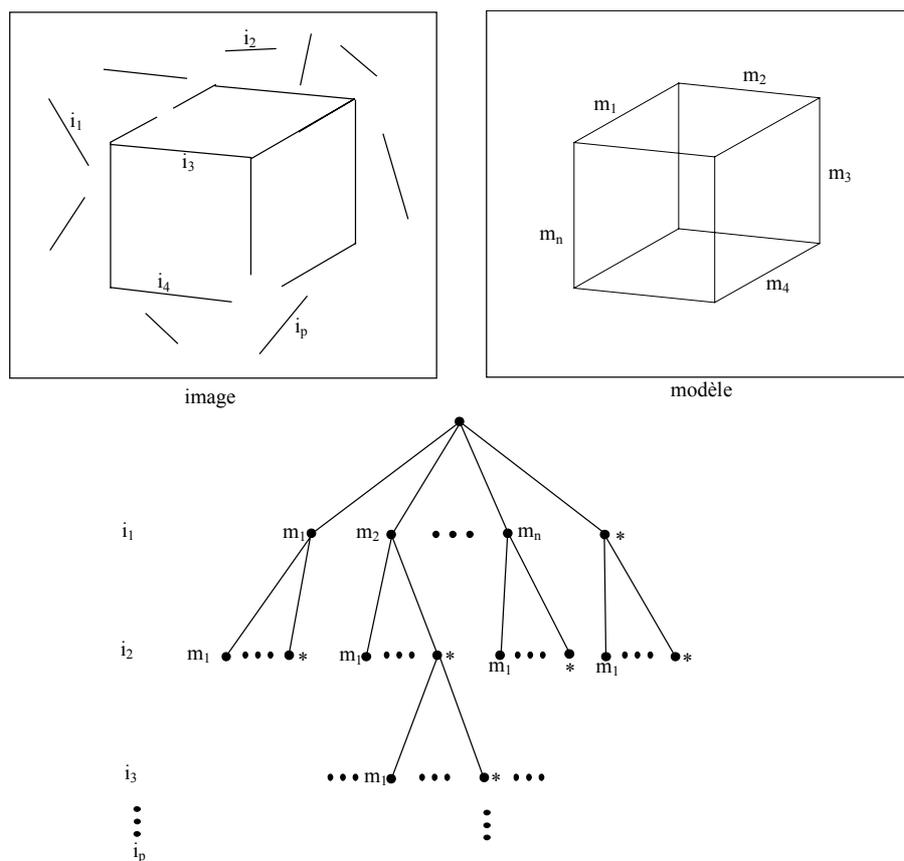


Figure 2.7 : Construction d'un arbre de recherche

[Grimson et Lozano-Pérez, 1984] traitent le problème de l'appariement de primitives 3D entre elles. Il cherchent à reconnaître et à localiser des objets polyédriques à partir de données sensorielles de profondeur (points 3D). Ils supposent que le modèle géométrique est connu. La recherche d'appariements corrects se fait par l'intermédiaire d'un arbre d'interprétation répertoriant tous les appariements possibles entre les informations de profondeur et les faces du modèle. L'arbre construit est composé d'autant de niveaux qu'il y'a de points. La racine de l'arbre possède autant de descendants que l'objet polyédrique a de faces. A chaque niveau est associée une donnée de profondeur différente. L'élagage de l'arbre ("*pruning*") se fait par propagation de quatre contraintes géométriques (la distance, l'angle, la direction et le produit des normales en trois points). Ceci est équivalent à trouver le chemin optimal en profondeur dans cet arbre en déterminant à chaque niveau un appariement entre primitives en utilisant ces contraintes. A la suite de cette opération, chaque feuille de l'arbre atteinte représente une interprétation possible. L'auteur détermine alors la transformation correspondante à chaque interprétation possible et vérifie l'adéquation du reste du modèle.

Diverses méthodes classiques de l'intelligence artificielle permettent d'élaguer la recherche dans l'arbre d'interprétation par contraintes, il est commun d'en distinguer deux catégories. Les contraintes *unaires* imposent que les primitives du modèle et de l'image possèdent des caractéristiques similaires alors que les contraintes *binaires* assurent la consistance géométrique et topologique. Supposons par exemple qu'on utilise des primitives de type segments : la longueur du segment vu dans la scène 3D ne peut être qu'inférieure à celle du segment modèle, puisque les seules modifications possibles sont les erreurs de mesure et l'occultation, c'est une contrainte *unaire* sur l'appariement. Supposons maintenant l'appariement de deux paires de segments vérifiant les contraintes *unaires*; les angles entre les deux droites support dans la scène et dans le modèle doivent être quasiment identiques, c'est une contrainte binaire. On peut ainsi développer pour chaque type de primitives un ensemble de contraintes géométriques unaires, binaires et d'ordres supérieurs qui permettent de conserver lors de la descente dans l'arbre une consistance locale de l'interprétation, la recherche de la transformation fournit en fin de compte la preuve de la consistance globale.

[Grimson, 1990] complète l'approche proposée dans [Grimson et Lozano-Pérez, 1984] et [Grimson et Lozano-Pérez, 1987] pour la reconnaissance d'objets bruités. Il propose une méthode, comme la précédente, efficace sur des objets isolés et bruités, mais dont les performances sont supérieures lorsque des données erronées sont introduites. C'est le cas notamment lorsque l'on est en présence d'occultations. Il traite non plus le problème d'appariement de primitives 3D entre elles mais celui de l'appariement 2D. Les objets sont modélisés par des polygones ou des polyèdres, de telle manière que chaque primitive du modèle soit un segment. Il suppose que les objets sont rigides. Il construit un arbre d'interprétation où à chaque niveau est associé une donnée différente. La profondeur de l'arbre d'interprétation est donc égale au nombre de données dans l'image. En partant de la racine il cherche à atteindre une feuille de l'arbre. L'évolution du processus est guidée dans ce cas par quatre contraintes aussi (taille, angle, distance et contrainte sur les normales). L'utilisation d'un caractère spécial (*null character "\*"*) permet de distinguer les données erronées des bonnes (cf. figure 2.7). Le processus de recherche comporte deux modes

d'évolution appelés "*seek mode*" et "*backward mode*". Le premier est utilisé pour descendre dans l'arbre tant que cela est possible (contraintes satisfaites ou "*null character*"). Il permet de poursuivre l'appariement entre les données et les primitives du modèle. Le second est, quant à lui, utilisé pour une recherche ascendante. L'évolution dans l'arbre progresse logiquement tant que l'affectation d'une primitive modèle à une donnée image est possible. Lorsque toutes les possibilités d'affectation ont été testées sans succès, on affecte le caractère *null* (\*) à la donnée et on poursuit l'évolution descendante (*seek mode*). Quand une feuille de l'arbre est atteinte, toutes les données ont alors été prises en compte et donc un caractère "*null*" ou une primitive du modèle lui est attribuée. L'auteur teste alors la validité globale de cet appariement en calculant la transformation rigide associée. La poursuite de la recherche se fait naturellement en basculant en mode *backward*. Cette opération entraîne la recherche du dernier nœud de la branche tel que celui-ci ne corresponde pas au caractère "*null*". La complexité de la méthode est donnée par  $O((n+1)^m)$  dans le cas le pire, où  $n$  est le nombre de primitives de la scène et  $m$  celui du modèle.

Devant la complexité combinatoire des arbres d'interprétation, cette technique est envisageable lorsqu'une étape préalable d'isolement des objets détectés dans la scène a été effectuée. De cette façon le nombre de combinaisons décroît alors que l'efficacité croît considérablement. [Grimson, 1990] utilise ainsi la transformé de Hough pour isoler les ensembles de données susceptibles d'appartenir à l'objet. De même, dans [Grimson et al., 1992], un critère qualitatif permet de stopper prématurément le processus de recherche. Il s'agit d'une valeur mesurant la qualité du processus d'appariement, dès que cette valeur excède un seuil donné la recherche est arrêtée.

Cette approche propose des solutions résistantes au bruit, aux données erronées et aux problèmes d'occultations dans les images. Le principe utilisé reste général et facilement adaptable quelque soit le type de primitives à mettre en correspondance (points, segments, coins,...). Les heuristiques mises en place pour évoluer dans chaque arbre sont dépendante de la nature de l'application.

### 2.6.2 Plus proche voisin itéré (ICP : "Iterative Closest Point")

L'algorithme de plus proche voisin itéré, ou ICP, a été introduit par [Besl et McKay, 1992], et utilisé intensivement en imagerie médicale par [Feldmar, 1995]. Cela consiste en une optimisation alternée d'appariements et de transformations. Ainsi, à partir d'une transformation initiale, on réalise les deux étapes suivantes:

- *mise en correspondance* (plus proche voisin) : on apparie chaque primitive du modèle transformé avec la primitive la plus proche dans la scène.
- *recalage* : la transformation est généralement calculée aux sens des moindres carrés, surtout si l'on travaille avec des points. Si l'on possède une information d'incertitude, on peut l'utiliser dans les étapes terminales pour affiner la solution.

Un problème générique pour ce type d'algorithme itératif est le critère de terminaison. Puisque les primitives utilisées sont physiquement séparées dans l'espace (par opposition à

des points répartis "continûment" sur une courbe ou une surface), l'ensemble des appariements possibles est discret et chaque hypothèse d'appariement donne lieu à une transformation unique. Lorsque l'algorithme a convergé, non seulement les appariements restent les mêmes lors de l'itération suivante, mais la transformation calculée est strictement identique. Il suffit donc d'arrêter l'algorithme lorsque les transformations sont identiques d'une itération sur l'autre ou lorsqu'on a atteint un nombre maximal d'itérations.

Le principal problème de cet algorithme est sa sensibilité aux conditions initiales, particulièrement lorsque la partie commune au modèle et à la scène est faible par rapport au nombre total de primitives présentes. Il est alors impératif de démarrer avec une transformation ou un ensemble d'appariements qui est très proche de la solution espérée. C'est pour cette raison que cet algorithme est souvent utilisé comme un algorithme de vérification à la fin du processus de reconnaissance, pour affiner les appariements et la transformation obtenue par l'une des autres techniques présentées dans cette section.

### 2.6.3 Transformée de Hough généralisée

La transformée de Hough a été introduite par [Hough, 1962] pour détecter des primitives géométriques simples comme des droites à partir de points en accumulant des évidences dans leur espace paramétrique. La méthode a été généralisée pour la mise en correspondance de la manière suivante [Dhome et al., 1986]: soit  $k$  le nombre minimum d'appariements nécessaire pour calculer une transformation unique (ou au moins un nombre fini de transformations) entre le modèle et la scène. Pour chacun de ces  $k$ -uplets d'appariements, on calcule cette transformation (si elle existe) et on accumule dans l'espace des transformations les évidences pour cerner la transformation qui permet d'apparier le maximum de primitives. L'accumulation des évidences peut se faire par une discrétisation de l'espace des transformations. Ainsi, un tableau indexé par les paramètres de la transformation est initialisé à zéro. Ensuite, pour chaque ensemble d'appariements possible entre le modèle et la scène, les paramètres de la transformation ainsi que la pose de l'objet sont estimés et sont utilisés comme vote dans les cases associées du tableau. Une fois que toutes les possibilités de mise en correspondance ont été analysées, le tableau est balayé afin de détecter les cases qui ont reçu le plus de suffrage. De cette façon, la transformation de l'objet est estimée par les index du tableau qui sont alors directement accessibles. Généralement une erreur bornée est utilisée afin d'établir l'échantillonnage des index du tableau recevant les votes.

Cet algorithme a donné lieu à beaucoup d'études. Ainsi, [Wolfson et Lamdan, 1992] l'utilisent pour reconnaître un objet articulé, alors que [Gros, 1995] le combine avec le hachage géométrique (voir § 2.6.5) pour améliorer la robustesse de son algorithme de reconnaissance.

Cette méthode possède l'avantage essentiel d'éviter la recherche exponentielle des primitives, cependant l'algorithme nécessite le stockage en mémoire de l'espace des transformations et son parcours en fin d'algorithme pour trouver le maximum. Si les transformations rigides en 2D n'occupent qu'un espace de dimension 3, on passe à une

dimension 6 pour les transformations rigides 3D et à 9 pour les transformations affines 3D, ce qui peut donner lieu à des temps de calculs rédhibitoires si l'on ne gère pas efficacement cette recherche. De plus, cette méthode est très sensible aux bruits et si l'échantillonnage du tableau est mal conçu, certaines cases risquent de comptabiliser de nombreux votes correspondant à des poses multiples d'objets différents. Ainsi, il est possible que des solutions correctes soient masquées par des solutions ayant reçu plus de votes mais totalement erronées.

## 2.6.4 Prédiction/vérification d'hypothèses

### 2.6.4.1 Principe

Il s'agit d'un principe utilisé par la majorité des méthodes de reconnaissance (exposées au chapitre 1). Il est très simple et particulièrement bien adapté à l'utilisation de modèles géométriques formés de coins, de segments, surfaces, etc. Il s'intègre classiquement dans les processus de reconnaissance d'objets 3D de la façon suivante

1. *Segmentation* et *structuration* de l'image de manière à obtenir des primitives 2D susceptibles d'être les projections des primitives 3D. Par exemple, si le modèle de l'objet est un cube, il est judicieux de chercher dans l'image des contours fermés formés de quatre segments qui forment les bords d'une face. Ainsi, la probabilité qu'un tel contour détecté dans l'image soit la projection d'une face du modèle 3D du cube (1/6) est beaucoup plus forte que s'il s'agissait de simples segments (1/12).
2. *Appariement* hypothétique entre  $n$  primitives du modèle 3D et  $n$  primitives de l'image, susceptibles d'être les projections des  $n$  primitives 3D sélectionnées.
3. *Calcul des paramètres de la projection perspective* en utilisant l'une des méthodes exposées dans le paragraphe § 2.7.
4. *Vérification* par projection perspective de l'adéquation du reste des primitives 3D du modèle (au sens d'un critère mesurant la qualité de l'appariement). Le score de qualité de cette hypothèse peut être simplement le nombre d'appariement trouvés. Si la mise en correspondance est *suffisamment bonne*, on arrête le processus.

Cette technique a été principalement développée par [Ayache et Faugeras., 1986] et [Huttenlocher et Ullman, 1987].

[Bak, 1991] propose une méthode de reconnaissance 3D utilisant le principe de prédiction/vérification d'hypothèses. L'algorithme est appliqué à la reconnaissance et la classification du zoo-ploncton à partir d'images obtenues par microscopie. Le processus de reconnaissance est basé sur la mise en correspondance de groupes de segments avec les projections perspectives des vues caractéristiques des modèles 3D des objets. Après extraction des points de contours et chaînage de ceux-ci, les chaînes les plus significatives (longueur supérieure à un seuil) sont approximées par des segments de droite. Les segments sont ensuite regroupés en fonction de différents critères propres à des groupements perceptuels que sont la proximité, le parallélisme et la colinéarité. Tous les appariements possibles entre les groupements perceptuels et les vues caractéristiques de chaque modèle

sont répertoriés et la transformation associée calculée. Quatre points (les extrémités des deux segments composant un groupement perceptuel) sont utilisés pour retrouver les trois rotations et la translation dans un plan permettant d'amener le modèle de l'objet dans le repère lié à la caméra. La vérification est effectuée par projection 3D du modèle géométrique complet avec les paramètres trouvés. Si la mise en correspondance est suffisamment bonne (calcul d'une distance en pixel) le processus est arrêté. Dans le cas contraire, l'auteur sélectionne les points 2D extrémités proches du modèle 3D projeté et les utilise pour affiner les paramètres de la transformation initiale. Le processus est itéré plusieurs fois jusqu'à obtenir une mise en correspondance satisfaisante. La méthode nécessite la création, au préalable, d'une base de modèles 3D des objets, l'auteur utilise un modèle fil de fer.

### 2.6.4.2 Complexité

Prenons l'exemple d'un modèle constitué de  $m$  primitives de type point 3D [Pennec, 1998].

Il faut trois appariements pour spécifier une transformation rigide. Or il y a  $C_m^3 = \frac{m!}{3!(m-3)!}$

façons de choisir 3 points parmi  $m$  du modèle,  $C_n^3$  pour la scène et  $3!$  manières d'apparier les deux triplets, ce qui donne en définitive  $O(m^3 n^3)$  alignements ou prédictions à vérifier. On peut aussi utiliser les trois invariants du triplet (distances inter-points) pour indexer le triplet dans une table de hachage, ce qui permettra de retrouver en temps constant les triplets compatibles dans l'image. La complexité est alors réduite à  $O(m^3 + n^3)$  pour l'étape de prédiction. L'étape de vérification est particulièrement importante puisqu'elle doit rejeter les mauvaises hypothèses mais conserver les bonnes. Il s'agit en général d'une (ou plusieurs) étapes d'ICP (voir §2.6.2).

### 2.6.4.3 Discussions

Le principal inconvénient relatif à l'utilisation des techniques de prédiction/vérification d'hypothèses est la quantité d'hypothèses engendrées par les appariements possibles entre les primitives de l'image et les primitives du modèle. L'utilisation des groupements perceptuels permet de réduire la nature combinatoire du problème, mais ne la supprime pas. L'avantage de telles méthodes vient du fait que l'on peut stopper le processus de reconnaissance dès que l'on a trouvé une mise en correspondance suffisamment correcte (qui satisfait un critère d'erreur).

D'un autre côté l'utilisation du principe de prédiction/vérification d'hypothèses pour la reconnaissance d'objets définis par un modèle 3D permet de résoudre simultanément les problèmes d'appariements et de localisation.

## 2.6.5 Hachage géométrique (HG)

### 2.6.5.1 Principe

Cette technique a été introduite par [Lamdan et Wolfson, 1988]. Leur idée consiste à déterminer, durant une phase de prétraitement, les informations pertinentes du modèle et de

les stocker dans une zone mémoire qu'ils ont appelé table de hachage. Cette table est indépendante de la scène et peut donc être calculée hors ligne (off-line) pour ne pas affecter le temps de reconnaissance. L'accès à la table de hachage est basé sur une information géométrique qui est, d'une part *invariante* à la *pose* de l'objet, et d'autre part extraite directement à partir de l'image de la scène.

Durant la phase de reconnaissance, la table de hachage est utilisée pour indexer les propriétés géométriques des primitives de la scène afin de déterminer des appariements possibles avec les modèles de la base de données. L'un des avantages du hachage géométrique est qu'il réduit énormément l'espace de recherche en évitant de parcourir tout les modèles.

Les primitives de la scène généralement utilisées sont les points, les segments de droites, les segments curvilignes, les coins, etc. ; ils sont accumulés dans la table tout au long du processus d'extraction de primitives. Une collection de telles primitives peut être représentée par un ensemble de points dans la table : chaque point symbolise l'emplacement d'une primitive, il lui est associé une liste d'attributs qui dépendent du type de la primitive correspondante.

Dans ce qui suit, nous allons détailler les différentes parties de cette approche. Sans perte de généralité, nous allons examiner le cas simple du problème de la reconnaissance basée sur le regroupement de points.

Supposons que nous souhaitons réaliser la reconnaissance de modèles définis par des primitives de type points qui peuvent subir des translations, des rotations ou bien des changements d'échelle (une similitude). La figure (2.8) montre un modèle  $M_1$  constitué de 5 points définis par leurs vecteurs de position  $P_1$ ,  $P_2$ ,  $P_3$ ,  $P_4$  et  $P_5$  respectivement. Nous voulons encoder convenablement l'information sur ces points et la stocker dans une table. Cela permet, si on arrive à détecter ce groupement de points dans la scène, de conclure que ces points appartiennent bien au modèle  $M_1$ . Un codage idéal peut être obtenu par une *indexation des couleurs*. En effet, si tous les points (de tout les modèles et dans toute scène) avaient des couleurs distinctes, on pourrait construire une table de hachage où la couleur d'un point servirait à l'indexer et l'information codée puis stockée serait le numéro du modèle. Durant la phase de reconnaissance, il suffit de parcourir les points, puis accéder à la table en utilisant les couleurs trouvées et compter combien de fois le nom de chaque modèle a été atteint.

Toutefois, dans le cas d'un manque d'informations, où les points appartenant aux modèle n'ont pour attributs que leur configuration géométrique, une *couleur géométrique* naturelle d'un point est l'ensemble de ses coordonnées. Or, les coordonnées d'un point dépendent du repère de référence (référentiel) considéré. Une question s'impose alors : existe-t-il un référentiel naturel pour le modèle qui reste visible dans le cas d'occultations partielles de l'objet ? Pour répondre à cette question on choisi une paire de points appartenant au modèle et on défini un référentiel *non équivoque* qui reste *inchangé* même si le modèle subit une rotation, une translation ou bien un changement d'échelle. Considérons la paire de points  $P_4$  et  $P_1$  comme étant une *base ordonnée* d'un tel référentiel. On met à l'échelle

le modèle  $M_I$  de telle sorte que l'amplitude du vecteur  $\vec{p_4 p_1}$  dans le système de coordonnées  $Oxy$  soit égale à 1. Supposons maintenant que le point milieu entre les points  $P_4$  et  $P_1$  soit placé à l'origine du référentiel  $Oxy$  et que le vecteur  $\vec{p_4 p_1}$  soit orienté vers les  $x$  positifs. Les trois points restants du modèle  $M_I$  vont se retrouver dans trois emplacements par rapport au repère  $Oxy$ . On enregistre dans une table de hachage *quantifiée*, dans chacune des trois cellules où les points restants se retrouvent, le fait que le modèle  $M_I$  avec la base (4,1) présente une entrée dans ces cellules, ceci est illustré graphiquement dans la figure (2.8).

Puisque notre but est de réaliser une reconnaissance en présence d'occultation, on ne peut pas garantir que les deux points  $P_4$  et  $P_1$  de la base vont apparaître tout les deux dans chaque scène où le modèle  $M_I$  sera présent. Par conséquent, nous encodons les informations sur les points du modèle dans toutes les bases ordonnées possibles. Ainsi, la table de hachage contiendra trois entrées de la forme  $(M_I, (4,2))$ , trois autres entrées  $(M_I, (4,3))$ , etc. Chaque triplet d'entrées est généré par la mise à l'échelle du modèle  $M_I$  de telle sorte que la base correspondante ait une longueur unité dans le système de coordonnées  $Oxy$ , puis par le placement du point milieu de la base sur l'origine de la table de hachage de telle manière à ce que le vecteur de base soit orienté vers les  $x$  positifs. Le même processus est répété pour chaque modèle dans la base de données. Il est tout à fait clair que des cellules de la table de hachage peuvent recevoir plus d'une entrée. Finalement, dans chaque cellule de la table de hachage sera stocké une liste des entrées de la forme (modèle, base).

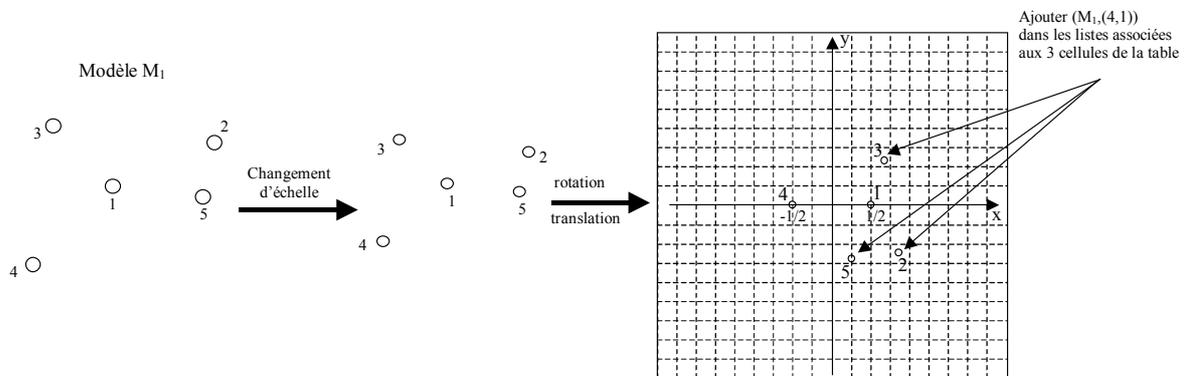


Figure 2.8 : Détermination des entrées de la table de hachage dans le cas où les points 1 et 4 sont utilisés pour définir la base

D'une manière plus formelle, on définit des bases orthonormales du référentiel  $Oxy$ , en utilisant à chaque fois une paire de vecteurs :

$$\mathbf{P}_x = \mathbf{P}_{\mu_2} - \mathbf{P}_{\mu_1} \quad \text{et} \quad \mathbf{P}_y = \text{Rot}(\mathbf{P}_x) \quad (2.7)$$

où  $\mu_1, \mu_2$  sont deux points distincts du modèle  $M_l$ . Pour chaque choix de la base, les points  $P$  restants du modèle  $M_l$  sont définis dans cette base par l'équation suivante :

$$P - P_0 = \mu P_x + \nu P_y \quad (2.8)$$

où  $P_0 = \frac{P_{\mu_1} + P_{\mu_2}}{2}$  est le point milieu entre  $P_{\mu_1}$  et  $P_{\mu_2}$ .

La quantification des scalaires  $\mu$  et  $\nu$  qui sont invariants pour une similitude de  $M_l$ , nous permet de déterminer un index  $(\mu_q, \nu_q)$  en deux dimensions pour la table de hachage. Dans la cellule de la table à laquelle on accède via l'index  $(\mu_q, \nu_q)$  on inscrit l'information  $(m, (P_{\mu_1}, P_{\mu_2}))$ .

Durant la phase de reconnaissance, une paire de points  $(P_{\mu_1}, P_{\mu_2})$  de l'image est choisie comme une base candidate. Comme précédemment, cette base ordonnée définit un système de coordonnées  $Oxy$  dont l'origine coïncide avec le milieu de la paire de points et la direction du vecteur de base  $P_{\mu_2} - P_{\mu_1}$  coïncide avec celle de l'axe des  $x$  positifs. L'amplitude du vecteur de base définit la longueur « unité » dans le référentiel  $Oxy$ . Pour chaque base sélectionnée et pour chacun des points restants dans la scène, l'équation (2.8) est utilisée afin de déterminer l'index  $(\mu_q, \nu_q)$  qui permet d'accéder à une cellule de la table de hachage. Chaque information (modèle, base) trouvée dans la cellule obtient alors un vote supplémentaire. La figure (2.9) illustre ceci graphiquement.

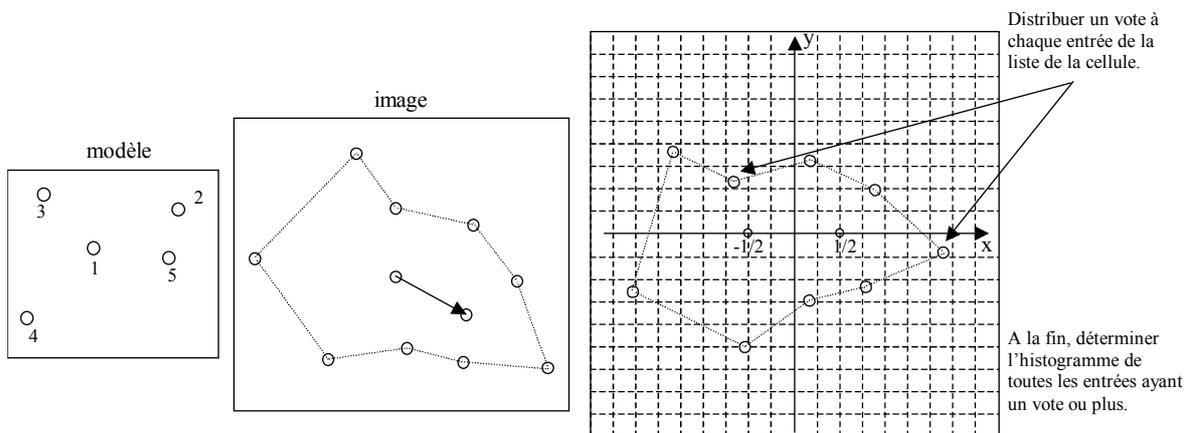


Figure 2.9 : Détermination des cellules de la table de hachage à prendre en compte lorsque deux points arbitraires de l'image sont choisis comme base

Si nous avons sélectionné par hasard une paire de points de la scène qui correspond à une base sur un des modèles, nous nous attendons à ce que son score accumule les votes de tous les autres points non occultés appartenant à ce même modèle. S'il y a suffisamment de votes pour une ou plusieurs combinaisons (modèle, base), alors une étape ultérieure tente de vérifier la présence du modèle avec la base désignée par l'appariement avec les points de la base choisie. Dans le cas où des points du modèle manquent dans l'image à cause d'une occultation, la reconnaissance reste toujours possible à partir du moment où il y a

suffisamment de points se projetant dans les bonnes cellules de la table de hachage qui correspondent au modèle. La liste des entrées dans chaque cellule de la table peut être très grande, mais comme il y a beaucoup de modèles et de bases possibles, la probabilité qu'un seul modèle et une seule base reçoivent plusieurs votes est assez faible, à moins que la configuration des points transformés coïncide avec celle du modèle. En général, on attend pas de ce système qu'il ne donne qu'une seule solution candidate. En effet, le but du vote est d'agir comme un filtre afin de réduire considérablement le nombre d'hypothèses plausibles pour l'étape de vérification. Pour que l'algorithme réussisse il suffit de choisir dans l'image n'importe quel ensemble de points à condition que ces points appartiennent au même modèle. Des méthodes de classification ou bien de regroupement de primitives peuvent être utilisées afin d'améliorer l'efficacité de la procédure de recherche de primitives dans la scène.

Enfin pour récapituler, nous résumons ci-dessous les deux phases principales du hachage géométrique :

### Phase de pré traitement

Pour chaque modèle  $m$  faire :

- (2) Extraire les primitives de type points du modèle. Soit  $n$  le nombre de primitives trouvées.
- (3) Pour chaque paire ordonnée de points (i.e. base) faire :
  - (a) Calculer les coordonnées  $(\mu, \nu)$  des points restants dans le système de coordonnées défini par la base.
  - (b) Après une quantification adéquate, utiliser le couple  $(\mu_q, \nu_q)$  comme un index pour une table de hachage à deux dimensions, et insérer dans la cellule de la table correspondante à cet index l'information  $(m, (base))$ , à savoir, le numéro du modèle et le couple de points constituant la base qui a été utilisée pour déterminer  $(\mu_q, \nu_q)$ .

### Phase de reconnaissance

Lorsqu'une image d'entrée se présente :

- (1) Extraire les différents points d'intérêt. Soit  $S$  l'ensemble des points d'intérêt trouvés.
- (2) Choisir arbitrairement une paire ordonnée (i.e. base) de points d'intérêt dans l'image.
- (3) Calculer les coordonnées des points d'intérêt restants dans le système de coordonnées  $Oxy$  défini par la base choisie.
- (4) Quantifier convenablement chaque coordonnée ainsi calculée et accéder à la cellule appropriée de la table de hachage ; pour chaque entrée trouvée dans cette cellule, donner un vote au modèle et à la base.

- (5) Calculer l'histogramme de toutes les entrées de la table de hachage ayant reçues un ou plusieurs votes durant l'étape (4). Déterminer celles qui ont reçu plus d'un certain nombre (seuil) de votes. Chaque entrée ainsi trouvée correspond à un appariement potentiel.
- (6) Pour chaque appariement potentiel trouvé dans l'étape (5), recouvrir la transformation  $T$  qui donne la meilleure correspondance, au sens des moindres carrés, entre les paires de points du modèle et de l'image.
- (7) Transformer toutes les primitives du modèle selon la transformation recouverte  $T$  et vérifier-les avec les primitives de l'image d'entrée. Si la vérification échoue, revenir à l'étape (2) et répéter la procédure en utilisant une autre paire de points image (base).

Le hachage géométrique représente une approche *unifiée* qui peut être appliquée à d'autres types de transformations rencontrées dans les problèmes de reconnaissance. La seule différence d'une application à une autre réside dans le nombre et le type de primitives à utiliser pour former une base dans le repère de référence. Ceci, bien évidemment, affecte la complexité de l'algorithme dans les différents cas.

La liste suivante donne des exemples typiques de transformations pour lesquelles ce paradigme général peut s'appliquer. Nous examinons en premier la reconnaissance d'objets 2D à partir de données 2D.

- 1) Translation 2D : la technique est applicable avec une base constituée d'un seul point. Ce point est pris comme étant l'origine du système de coordonnées.
- 2) Translation et rotation 2D : une base à deux points peut être utilisée, cependant un seul point orienté (obtenu à partir d'un segment d'une arête par exemple) suffit pour la définition unique de la base.
- 3) Transformation affine 2D : une base à trois points permet de définir un repère de référence non équivoque.
- 4) Transformation perspective 2D : une base à quatre points est nécessaire pour recouvrir une transformation perspective entre deux plans.

Lorsque les données 3D sont utilisées on parle alors de la reconnaissance d'objets 3D à partir d'images 3D. Des techniques de hachage géométrique basées sur les données 3D sont appliquées dans la CAO, l'imagerie médicale, et la comparaison de protéines dans la biologie moléculaire [Pennec, 1998].

- 5) Translation 3D : identique au cas 2D, une base à un point est suffisante.
- 6) Transformation rigide (translation et rotation 3D) : une base constituée de deux lignes non colinéaires est suffisante. Une autre alternative serait d'utiliser trois points pour définir la base, avec une information additionnelle sur les longueurs des côtés du triangle construit par ces points.

- 7) Translation, rotation et changement d'échelle 3D : une base constituée de deux droites non coplanaires et non colinéaires est suffisante. Un point et une droite peuvent aussi être utilisés.

### 2.6.5.2 Complexité

En général, si la base de données contient  $m$  modèles ayant chacun  $n$  primitives, et si durant la phase de reconnaissance, la scène contient  $S$  primitives, et que  $c$  primitives sont nécessaires pour former une base, alors la complexité sera donnée par  $O(mn^{c+1})$  pour la phase de pré traitement, et par  $O(S^{c+1}H)$  pour la phase de reconnaissance, où  $H$  représente la complexité pour accéder à une cellule de la table de hachage, il dépend du taux d'occupation de la table de hachage. Il est important de noter que la structure de la table de hachage est connue à l'avance avant la phase de reconnaissance. Ainsi, on peut évaluer cette structure et décider si elle nécessite ou non l'application d'une procédure de re-hachage, la subdivision de la table en plusieurs sous tables, ou bien le changement de la structure de l'index par une autre de dimension supérieure. On devrait noter aussi que les cellules ayant un taux d'occupation élevé, et qui entraînent des temps de calculs importants, peuvent tout simplement être ignorées puisque l'information qu'elles contiennent n'est pas assez pertinente pour aider à la reconnaissance car elles ne sont alors pas assez discriminantes. Ainsi, on peut décider a priori de la borne supérieure de la taille des cellules de la table de hachage. Une extension de cette idée serait de pondérer les votes en supposant que l'information contenue dans une cellule de la table de hachage soit inversement proportionnelle au nombre de modèle que référence cette cellule.

### 2.6.5.3 Méthodes et discussions

L'idée du hachage géométrique, dans sa version moderne, trouve son origine dans les travaux de [Schwartz et Sharir, 1986]. Ces premiers efforts étaient concentrés sur la reconnaissance d'objets 2D pouvant subir une rotation, une translation ou une occultations à partir de leur silhouettes en utilisant des techniques d'appariement de courbes de frontière. [Schwartz et al., 1988] ont étendu ce concept pour permettre la reconnaissance à partir, non seulement de courbes 2D, mais dans le cas général de primitives 2D ou 3D. Ils ont développé une technique de hachage géométrique applicable à un ensemble arbitraire de points (constellation) pouvant subir des transformations géométriques de différents types. Ils ont développé des algorithmes efficaces pour la reconnaissance d'objets rigides et plats représentés soit par un ensemble de points soit par des courbes, sous des transformations affines. Cette technique a été étendue à son tour pour la reconnaissance d'un ensemble de points sous des transformations quelconques et aussi à la reconnaissance d'objets 3D rigides à partir d'une seule image 2D [Lamdan et Wolfson, 1988].

L'un des avantages majeurs du hachage géométrique est qu'il est fondamentalement parallèle, sa structure peut être facilement décomposée et partagée entre un certain nombre de processeurs coopératifs avec un coût de communication minimal. Des implémentations parallèles du hachage géométrique sont décrites dans [Rigoutsos et Hummel, 1992].

Juste après la conception de la technique du hachage géométrique, [Flynn et Jain, 1992] ont réalisé que la méthode présentait la propriété suivante : la distribution des indices dans l'espace des invariants était non-uniforme. Cette non uniformité n'est pas en fait spécifique aux techniques de hachage géométriques mais elle apparaît d'une manière récurrente dans tous les schémas d'indexation ([Costa et al., 1992], [Mohan et al., 1993] et [Rigoutsos et al., 1996]). D'un point de vue pratique, la non uniformité de la distribution permet de réduire énormément le temps d'exécution de l'algorithme et aussi de réaliser une structure efficace de stockage des données de la table de hachage.

Récemment [Rigoutsos et Hummel, 1995] a développé une extension Bayésienne du hachage géométrique pour la reconnaissance d'objets. Il utilise des notions de probabilité dans les appariements modèles/scènes ce qui confère à sa technique de reconnaissance une robustesse aux bruits de la scène.

### 2.6.6 Indexation d'invariants géométriques

La généralisation du hachage géométrique à des appariements de primitives autres que des points n'est pas complètement évidente, contrairement à la prédiction/vérification d'hypothèses ou à la transformé de Hough généralisée. En effet, la notion de base et de coordonnées des autres points dans cette base est liée à la structure d'espace vectoriel. On peut toutefois concevoir un algorithme similaire mais faisant intervenir les invariants de plusieurs primitives à la place des coordonnées, l'accumulation des évidences se faisant toujours dans l'espaces des appariements.

Prenant l'exemple des points 3D [Pennec, 1998] : l'équivalent du hachage géométrique présenté ci-dessus sera d'utiliser dans la table de hachage les invariants caractéristiques de la forme constituée des quatre points ordonnés  $(P_i, P_j, P_k, P_l)$  au lieu des coordonnées de  $P_l$  dans une base construite à partir des points  $(P_i, P_j, P_k)$  (voir § 2.4). Au niveau de l'accumulation des évidences, il paraît alors naturel d'incrémenter les appariements possibles individuellement au lieu de le faire pour la base : si  $(P'_i, P'_j, P'_k, P'_l)$  est un quadruplet de points compatible dans la scène, on votera alors pour chaque appariement  $(P_i, P'_i)$  du quadruplet. Cette modification faite, il n'est plus réellement utile de considérer des quadruplets de points : on peut tout à fait réduire le nombre de point à trois, voire simplement à deux, ce qui permet de réduire énormément la complexité : on passe pour la reconnaissance de  $O(n^4)$  à  $O(n^3)$  voire  $O(n^2)$  respectivement.

Il faut cependant noter que, dans le cas où l'on ne considère que les invariants d'un seul couple de points, la probabilité de faux positifs devient vite très importante et les résultats ont de grandes chances d'être aberrants. Si l'on considère un triplet de points, on retrouve un algorithme très proche de la transformée de Hough généralisée, sauf que l'accumulation n'a plus lieu dans l'espace des transformations mais dans celui des appariements. Il serait sans doute intéressant de combiner les deux approches, par exemple en regroupant les transformations compatibles associés à chacun des votes pour un appariement.

Ce type d'algorithme a été utilisé par [Fischer et al., 1992] pour réduire la complexité du recalage basé sur les points 3D de  $O(n^4)$  à  $O(n^3)$ , l'accumulation se faisant ici sur

l'appariement d'une pseudo-base des deux premiers points de chaque triplet. Un autre exemple d'algorithme de ce type utilisant des primitives complexes (points munis de descripteurs de forme locaux) a été développé par [Califano et Mohan, 1991].

Toutefois, l'utilisation de cette méthode avec des primitives autres que les points pose un problème de taille : celui du calcul des invariants.

### 2.6.7 Isomorphisme de graphes

Le problème de la reconnaissance d'objets a souvent été interprété comme un problème d'isomorphisme de graphes. Ainsi, au problème de la mise en correspondance de primitives se substitue alors la question suivante: quelle est la correspondance entre le graphe du modèle et le graphe de la scène?

On peut distinguer deux approches pour répondre à cette question : La première se rapproche de l'utilisation des arbres d'interprétation et se nomme la recherche de cliques [Horaud et Skordas, 1989]. Son objectif est d'une part de construire un graphe relationnel qui recense tous les appariements possibles entre les primitives de l'image et celles du modèle, puis d'autre part de rechercher dans ce graphe une clique maximale, c'est à dire un sous graphe totalement connecté.

[Horaud et Skordas, 1989] proposent une méthode de mise en correspondance de primitives 2D par recherche de cliques maximales. Cette méthode est appliquée au problème d'appariement de primitives segments issues de deux images stéréoscopiques. Un graphe relationnel (jonction, colinéarité,...) est construit pour chaque image. Les auteurs cherchent, pour chaque segment d'une image, les segments de l'autre image susceptibles d'être appariés avec celui-ci. Les critères d'appariement tiennent compte de la géométrie des deux caméras (contrainte épipolaire, de position et d'orientation). Ils construisent alors un graphe de correspondance où les nœuds représentent tous les appariements possibles répertoriés auparavant. Un arc est construit lorsque deux nœuds du graphe sont compatibles, c'est à dire lorsque la description donnée par le graphe structurel de l'image de gauche est similaire à celle donnée par le graphe structurel de l'image de droite. Les meilleurs appariements sont données par les cliques maximales du graphe de correspondance.

Compte tenu du fait que la recherche de cliques maximales est un problème NP-complet dont la complexité croît avec le nombre de nœuds, il est évident que cette technique est limitée aux scènes simples.

La seconde approche repose sur l'appariement direct entre deux graphes représentant respectivement le modèle et l'objet détecté dans la scène. Ainsi, les nœuds correspondent aux caractéristiques utilisées alors que les arcs déterminent les relations entre les caractéristiques. L'isomorphisme de graphes n'est pas le terme le plus juste pour symboliser cette opération. En effet, du fait des occultations et défauts de traitements bas-niveau, certaines caractéristiques peuvent être manquantes et dans ce cas on recherche un sous isomorphisme de graphes [Shapiro et Haralick, 1981]. L'association entre les deux graphes

s'effectue sur la similarité des caractéristiques et de leur relation. Ainsi, la notion de distance entre graphes est utilisée afin de quantifier cette similitude [Cinque et al., 1996]. De même que pour la technique de la recherche de cliques, l'appariement direct est également confronté à la complexité croissante en fonction du nombre de nœuds dans les graphes. Des méthodes permettant de réduire cette complexité en partitionnant la mise en correspondance de graphes en sous-problèmes à l'aide de contraintes [Dargenton et al., 1994] ont été explorées.

L'avantage de l'isomorphisme de graphes est son pouvoir de modéliser une information plus importante que la simple géométrie des primitives, on pourrait ajouter par exemple le type de ligne ou de surface reliant deux primitives dans l'étiquetage de l'arête ou toute autre information que l'on voudrait conserver dans la modélisation d'un objet et qui permet de simplifier grandement le problème de la reconnaissance. En fait, il apparaît que cette technique est adaptée à une structuration haut-niveau de l'information, et donc à la reconnaissance des objets déjà modélisés ou d'objets articulés [Pennec, 1998].

## 2.6.8 Graphe d'aspects

### 2.6.8.1 Définitions

Une approche intéressante pour reconnaître des objets 3D est d'obtenir une collection de vues 2D d'un objet connu a priori (cf. figure 2.10), et de les représenter d'une manière efficace pour simplifier leur stockage, et puis d'apparier une ou plusieurs vues 2D d'un objet inconnu avec les vues stockées auparavant, de cette façon le problème d'appariement 2D/3D, dont la solution est généralement difficile à mettre en œuvre, se réduit à un problème d'appariement 2D/2D plus facile à traiter.

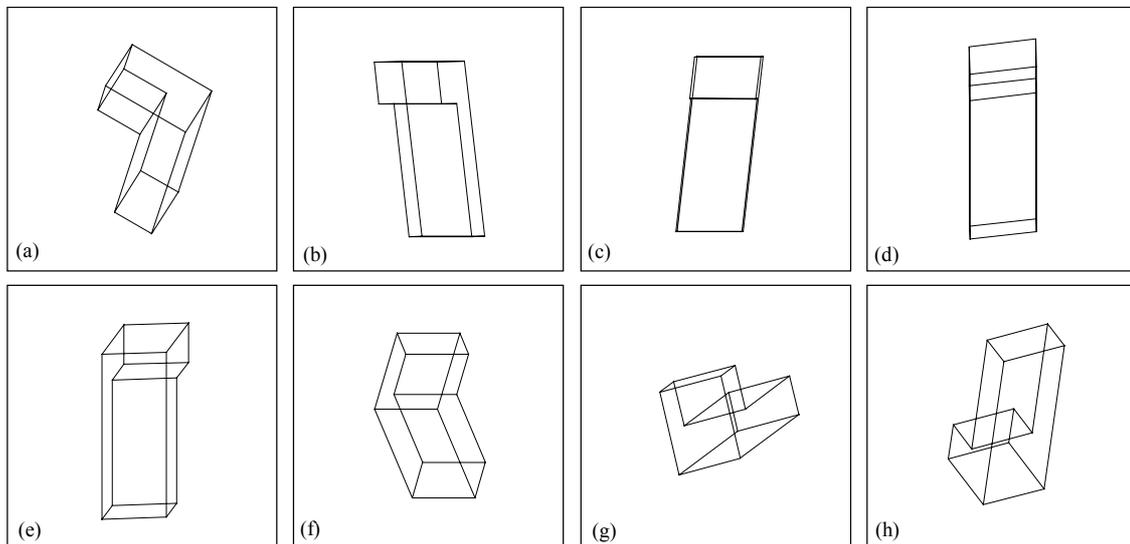


Figure 2.10 : Un ensemble de vues 2D d'un objet connu en forme de L

Le graphe d'aspects (ou graphe de vues) est une représentation qualitative d'un objet qui énumère toutes ses apparences topologiquement distinctes. Le modèle de caméra étant

fixé, le choix d'un point de vue détermine la structure de l'aspect de l'objet. L'espace d'observation considéré est alors partitionné en un ensemble de régions maximales, où tous les points de vue voient les mêmes indices de l'objet. Le changement d'aspect à la frontière entre deux régions s'appelle un *événement visuel*. Deux aspects d'un objet sont différents si l'une des primitives au moins disparaît ou apparaît lorsqu'on passe de l'un à l'autre.

Un graphe d'aspects est la représentation sous forme de graphe de la parcellisation de l'espace d'observation en cellules. Il a pour nœuds les aspects et pour arcs les événements visuels. De plus, il conserve l'information d'adjacence entre les aspects considérés. Deux aspects sont dits adjacents s'il est possible, pour un observateur en mouvement, de passer d'un aspect à un autre sans pour autant passer par d'autres aspects intermédiaires.

L'approche haut niveau de base qu'utilisent la plupart des algorithmes de calcul des graphes d'aspects est la suivante [Eggert et al., 1993]:

1. Utiliser la définition géométrique de l'objet pour énumérer l'ensemble des événements visuels pouvant se produire.
2. Trouver une parcellisation de l'espace d'observation défini par l'ensemble des surfaces associées aux événements visuels.
3. Traverser la parcellisation de l'espace d'observation pour construire le graphe d'aspects et stocker dans les nœuds les descriptions des aspects représentatifs.

Les algorithmes ainsi développés peuvent être classifiés selon trois propriétés; le **type d'objets** utilisés; la **représentation de la vue** (i.e. la définition avec précision des éléments qui constituent une vue de la scène à partir d'un point de vue donné) et la **modélisation de l'espace d'observation** (i.e. l'ensemble des points de vues disponibles à partir desquels la scène peut être vue). Plusieurs travaux de recherche concernant la détermination des graphes d'aspects pour différents types d'objets ont été publiés ces dernières années, par exemple pour les objets polygonaux [Gualtieri et al., 1989], les objets polyédriques ([Wang et Freeman, 1990], [Platinga et Dyer, 1990] et [Gigus et Malik, 1990] ), les solides de révolution [Kriegman et Ponce, 1990], les objets lisses par morceaux "piecewise-smooth" [Rieger, 1990] et les objets articulés [Sallam et al., 1990].

Deux modèles de l'espace d'observation sont communément utilisés par la plupart des chercheurs, le *modèle orthographique (orthogonal)* et le *modèle perspective*. Dans le *modèle orthographique*, l'espace d'observation est constitué de tous les points situés sur une sphère ayant un rayon infini. Chaque point de vue représente une direction particulière d'observation, il est défini par deux paramètres, en l'occurrence sa *longitude*  $\varphi$  et sa *latitude*  $\theta$  (cf. figure 2.11). On peut noter qu'il existe une correspondance naturelle entre l'ensemble de ces points de vue et des points se trouvant sur une sphère unité  $S$  autour de l'origine.

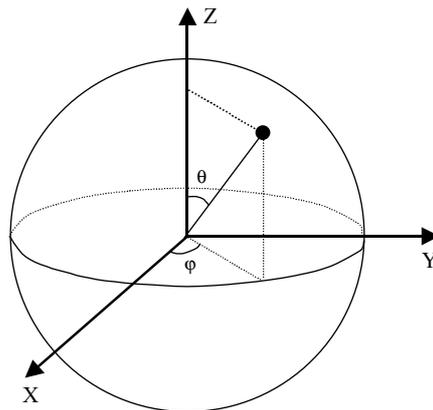


Figure 2.11: Un point  $(\theta, \varphi)$  sur la sphère  $S$  représentant un point de vue dans le modèle orthographique.

Dans ce modèle, toutes les lignes de vue (rayons visuels) émanant à partir d'un point de vue donné sont parallèles et orientées dans une direction opposée à celle définie par ce point de vue. Chaque ligne de vue qui intersecte l'objet s'arrête au point d'intersection correspondant de l'objet, définissant ainsi un rayon. Les primitives visibles des objets (sommets, arêtes ou portions d'arêtes, faces ou portions de faces) par rapport à un point de vue donné sont toutes celles rencontrées par l'ensemble de ces rayons. Une primitive de l'objet qui n'est pas visible à partir d'un point de vue donné et pour laquelle le volume immédiatement devant elle (par rapport au point de vue) se trouve à l'extérieur de l'objet lui-même est dite être *occultée* depuis ce point de vue. Le modèle de projection orthographique est adapté aux cas où la distance entre l'objet et le capteur est grande devant la taille de l'objet. En ce cas, la détermination des aspects pour un objet quelconque peut se faire de manière approchée par simulation de la projection [Ikeuchi et Kanade, 1988].

Dans le modèle de projection perspective, l'espace d'observation est constitué de tous les points dans  $\mathfrak{R}^3$  à l'exception des points occupés par les objets eux-mêmes. Ainsi, chaque point de vue est défini par ses coordonnées  $(x, y, z)$  par rapport à un repère de référence. Dans ce cas, les lignes de vue émanent de toutes les directions à partir d'un point de vue donné. Chaque ligne de vue qui intersecte un objet définit un segment de droite dont le point de début correspond au point de vue et le point de fin au point d'intersection avec l'objet. Les primitives visibles des objets par rapport à un point de vue donné sont toutes celles rencontrées par l'ensemble de ces segments de droites.

Concernant la représentation d'une vue, nous donnons ici sa définition la plus commune, nous allons nous référer à un objet seul dans la scène, mais la méthode reste valable pour des scènes contenant de multiples objets.

Avec le modèle de projection orthographique, nous choisissons tout d'abord un point de vue puis nous projetons les sommets, les arêtes et les morceaux d'arêtes dans un plan  $P$  orthogonal aux lignes de vue. Un sommet ou bien un point d'une arête intersecté par une

ligne de vue est projeté sur le plan  $P$  en un point qui correspond à l'intersection de cette ligne de vue avec le plan  $P$ . Puisque toutes les lignes de vue sont parallèles, la projection obtenue est sans distorsion; ainsi les primitives le long de deux lignes de vue  $l$  et  $l'$  séparées par une distance  $d$  auront des projections qui seront elles aussi séparées de la même distance  $d$  dans le plan de projection. Avec le modèle de projection perspective, une fois le point de vue choisi, on projette ces mêmes primitives sur une sphère  $S$  centrée sur le point de vue. Pour chaque ligne de vue  $l$ , le sommet de l'objet ou bien le point de l'arête intersecté par celle-ci est projeté sur la sphère  $S$  en le point où  $l$  intersecte  $S$ . Des distorsions perspectives peuvent se produire avec le modèle de projection perspective, en effet les arêtes parallèles de l'objet par exemple, qui sont loin du point de vue vont paraître, une fois projetés, se croiser à un point communément appelé le point de fuite.

La projection de l'objet sur le plan  $P$  ou bien sur la sphère  $S$  définis à partir d'un point de vue donnée est appelée *image* de l'objet de ce point de vue. Cette image est généralement constituée de trois types de primitives: *sommets*, *arêtes* et *jonctions en T*. Ainsi, les sommets et les morceaux des arêtes de l'objet se projettent respectivement sur les sommets et les arêtes de l'image. Les sommets supplémentaires, appelés jonctions en T, proviennent à partir des intersections entre les projections des arêtes non-adjacentes de l'objet.

Un graphe étiqueté de la structure de l'image (LISG: *labeled image structure graph*) est construit en faisant correspondre ("mapping") les sommets de l'image (y compris les jonctions en T) avec les nœuds, et les arêtes de l'image avec les arcs, et en étiquetant, d'une manière consistante, ces nœuds et arcs ainsi que les faces incidentes. Un schéma d'étiquetage logique consiste en l'attribution d'une étiquette différente à chaque sommet, arête et face de l'objet originel. Les sommets de l'image qui ne sont pas des jonctions en T et les arêtes de l'image reçoivent les étiquettes des primitives correspondantes de l'objet qui se projettent en eux. A chaque jonction en T est attribué l'ensemble des étiquettes des arêtes de l'image qui l'ont engendrée. Finalement, on attribue à chaque face de l'image l'étiquette de la face de l'objet vu à cet endroit de ce point de vue. Ainsi, avec ce schéma d'étiquetage, le LISG définit un graphe non-orienté, étiqueté et complet (dans le sens où il fournit la totalité des informations qui caractérisent les différents aspects). La description d'une vue de l'objet à partir d'un point de vue sera donnée par le LISG de l'image telle qu'elle est vue de ce point de vue.

De cette façon le problème de reconnaissance se ramène à un problème d'isomorphisme de graphes. Deux graphes LISG noté  $A$  et  $B$  sont isomorphes lorsqu'ils possèdent la même structure topologique (qualitative), c'est à dire lorsqu'il existe une correspondance un-à-un  $M$ , appelée isomorphisme, qui permet de faire correspondre aux nœuds, arcs et faces de  $A$ , les nœuds, arcs et faces de  $B$ , tel que pour un nœud donné  $n$  de  $A$ , si les arcs et les faces adjacents à  $n$  (pris dans le sens horaire) sont  $a_1, f_1, a_2, f_2, \dots, a_k, f_k$  et si les arcs sont adjacents aux nœuds  $n_1, n_2, \dots, n_k$ , respectivement, alors les arcs et les faces adjacents à  $M(n)$  dans  $B$  seront  $M(a_1), M(f_1), M(a_2), M(f_2), \dots, M(a_k), M(f_k)$  (pris aussi dans le sens horaire) et les arcs seront adjacents aux nœuds  $M(n_1), M(n_2), \dots, M(n_k)$ , respectivement. Les nœuds  $n$  et  $M(n)$ , les arcs  $a$  et  $M(a)$  ainsi que les faces  $f$  et  $M(f)$  doivent être étiquetés d'une manière identique. Deux vues sont dites *distinctes* si leurs graphes LISG ne sont pas isomorphes.

Pour calculer le graphe d'aspects, il est nécessaire tout d'abord de partitionner l'espace d'observation en un ensemble de régions connexes maximales appelées aussi régions non critiques, et pour lesquelles les vues correspondantes sont isomorphes. Un point de vue représentatif de chaque région de la partition de l'espace d'observation (*VSP: viewpoint space partition*) est alors choisi et son graphe LISG construit. Les régions maximales et leurs représentants forment les nœuds du graphes d'aspects alors que les frontières entres régions adjacentes forment ses arcs.

Les points de vue des régions de la VSP sont appelés points de vue généraux "*general viewpoints*". Ce sont des points de vue pour lesquels tout mouvement infinitésimal dans l'espace des points de vue conduit à un autre point de vue ayant une vue isomorphe. Les vues associées aux régions non critiques sont dites *stables*. Avec le modèle de projection orthographique les régions non critiques sont des cellules 2D bornées par des courbes 1D (cf. figure 2.12). Par contre, avec le modèle de projection perspective les régions non critiques sont des cellules 3D bornées par des surfaces 2D. Ces courbes et surfaces, appelées régions critiques concrètes "*actual critical regions*", contiennent les points de vues pour lesquels il existe un mouvement infinitésimal dans l'espace des points de vue conduisant à des points de vue ayant des vues non isomorphes. Les points de vue dans ces courbes et surfaces critiques sont appelés points de vue accidentels "*accidental viewpoints*". Les vues associées aux régions critiques concrètes sont dites *instables*. L'information qu'elles apportent pour l'appariement d'images est insignifiante car les régions dans l'espace des points de vue auxquelles elles sont associées sont infiniment petites, et de ce fait, ces vues ne sont pas stockées dans le graphe d'aspects.

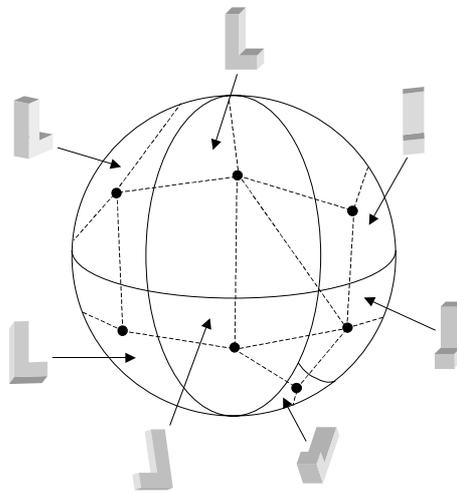


Figure 2.12: Partition de l'espace d'observation (lignes continues) et le graphe d'aspects (lignes discontinues) d'un objet en forme de L pour le modèle orthographique

### 2.6.8.2 Complexité

La complexité des graphes d'aspects a limité énormément leur utilisation dans la reconnaissance d'objets pour deux raisons principales : leur construction est coûteuse et leur comparaison peut engendrer une explosion combinatoire. A titre d'exemple, la complexité de construction d'un graphe d'aspects pour un polyèdre concave ayant  $n$  sommets, arêtes et faces est donnée par  $O(n^6)$  pour le modèle de projection orthographique, et  $O(n^9)$  pour le modèle de projection perspective [Schiffenbauer, 2001].

Les algorithmes de construction des graphes d'aspects géométriques pour des objets quelconques (de forme libre) sont très coûteux en temps de calculs : les études exhaustives se sont en général consacrées à quelques objets simples révélant les difficultés d'une automatisation de la procédure de reconnaissance.

Pour remédier à la complexité relativement prohibitive des graphes d'aspects, des méthodes d'approximation de graphes d'aspects ont été développées. Elles permettent de construire des structures de données simples, peu complexes et qui restent suffisamment discriminantes pour la reconnaissance. On peut citer, à titre d'exemples de ce type de structures, les graphes d'aspects à résolution finie "*finite-resolution aspects graphs*" [Shimshoni et Ponce, 1997], les graphes d'aspects dans les espaces d'échelles "*scale-space aspect graphs*" [Eggert et al., 1993] et les graphes d'aspects pondérés "*weighted aspect graphs*" [Ben-Arie, 1990].

### 2.6.8.3 Méthodes et Discussions

[Koenderink et Van Doorn, 1979] sont les premiers à avoir introduit la notion de graphe d'aspects en étudiant la modélisation de la vision humaine. leurs travaux ont d'abord été prolongés, sous un aspect mathématique, par [Plantinga et Dyer, 1990] puis utilisés en vision par ordinateur pour la reconnaissance d'objets polyédriques par [Ikeuchi et Kanade, 1988], [Stewman et Bowyer, 1990] et [Gigus et al., 1991].

D'un autre côté, de nombreuses nouvelles techniques ont été proposées et implantées pour construire le graphe d'aspects d'objets de formes courbes. L'exemple standard de ce type de graphes est fourni par la variation du contour d'occultation d'un objet lisse par morceaux [Rieger, 1993]. Le contour apparent, correspondant aux points de l'objet pour lesquels la direction de projection sur la rétine de la caméra est tangente à la surface extérieure ou rencontre une arête, se projette en une courbe différentiable, sauf en un nombre fini de points. Pour certaines directions d'observations, le contour d'occultation subit des variations brusques dans la structure de ses singularités, manifestant ainsi une discontinuité des apparences. Deux contours d'occultation seront déclarés semblables ou de même type qualitatif s'ils peuvent être transformés l'un dans l'autre par un *difféomorphisme*. La théorie des singularités (appelée aussi théorie des catastrophes) offre un catalogue complet d'événements visuels qui peuvent être utilisés pour comprendre la structure et la formation des graphes d'aspects permettant ainsi une approche systématique de leur détermination pour certaines classes d'objets (solides de révolution, surfaces algébriques) [Petitjean et al., 1992].

En conclusion, on peut dire que l'intérêt des graphes d'aspect tient à trois propriétés principales:

1. Ils permettent de réduire l'indétermination du point de vue selon deux directions : par partitionnement de l'espace d'observation en classes d'équivalences définies par les propriétés de stabilité structurelle du contour d'occultation, et par structuration différentielle globale de ces classes d'apparence. La définition d'une catégorisation intrinsèque des apparences est produite essentiellement par des propriétés locales de la variation du contour d'occultation. L'ensemble des catégories d'apparences, dans leur globalité, peut également être caractérisé plus spécifiquement : c'est ici qu'intervient la structure de graphe.
2. Ils décrivent la structure différentielle qui régit cette catégorisation intrinsèque des données sensorielles. En effet, les différences entre catégories de contours d'occultation, à cause de la typologie finie des catastrophes élémentaires, ne sont pas quelconques : un graphe d'aspects est en général de faible complexité. Sa structure est donc caractéristique de l'objet et sera porteuse d'information potentiellement discriminante.
3. En plus d'organiser les catégories, le graphe d'aspects fournit une unité à la diversité des apparences de l'objet. Ainsi, la donnée de l'ensemble des aspects et de leur organisation structurée par un graphe détermine qualitativement l'objet observé.

Ceci dit, l'information utilisée pour construire un graphe d'aspects dans les méthodes de reconnaissance citées ci-dessus est réduite généralement au seul contour d'occultation de l'objet, et néglige d'autres indices visuels, en particulier photométriques, potentiellement caractéristiques. Ce problème reste d'actualité et n'a pas été résolu dans sa globalité. L'approche que nous proposons dans le chapitre (3) est une approche originale dans le sens où elle combine l'information photométrique avec l'information géométrique (silhouette de l'objet) pour construire le graphe d'aspects d'un objet de forme libre.

## 2.7 Recalage

Par recalage, nous entendons détermination de la pose de l'objet par rapport à un référentiel connu. Il s'agit de calculer les trois paramètres de translation ainsi que les trois paramètres de rotation permettant d'amener l'objet de sa position initiale à sa position courante telle que perçue dans l'image. Classiquement il s'agit de calculer la position de l'objet soit dans le repère associé au capteur soit dans le repère associé à la scène.

Le recalage d'objets suppose la connaissance d'un nombre suffisant d'appariements entre les primitives de la scène ainsi que du modèle 3D géométrique associé à l'objet. De plus, dans le cas où l'on utilise des images d'intensité, on a besoin aussi de connaître le modèle de projection perspective de la caméra. Il est déterminé par le processus de calibration de la caméra. Ce modèle rend possible, connaissant les coordonnées d'un point 3D de la scène, le calcul des coordonnées 2D associées dans le plan image de la caméra.

D'un point de vue purement mathématique, si on considère un ensemble de  $n$  primitives de l'image (d'intensité et/ou de profondeur) appariées avec le même nombre de primitives du modèle 3D de l'objet à localiser, le problème s'exprime alors sous la forme d'un système d'équations non linéaires à six inconnues (3 pour la rotation et 3 pour la translation). En pratique, à cause du bruit, on a jamais des données exactes, et l'on doit donc prendre en compte les erreurs de mesure. Ainsi, la solution usuelle est alors de minimiser un critère quadratique qui exprime la distance entre les points transformés du modèle et les points correspondants de la scène. La minimisation de ce critère est conditionnée par [Horaud et Monga, 1993]:

- les caractéristiques visuelles utilisées ne sont pas toujours des points. On aura à traiter aussi bien des points que des segments de droite, des facettes planes ou des primitives surfaciques.
- la représentation adoptée pour décrire la rotation.

Dans les chapitres suivants, nous allons détailler à chaque fois le critère à minimiser pour déterminer les paramètres de la transformation optimale ainsi que le formalisme mathématique adopté pour représenter la rotation.

## 2.8 Description du système de reconnaissance élaboré

Les systèmes que nous avons développés dans cette thèse traitent exclusivement le sous problème de la localisation et du recalage 3D automatique d'objets de forme libre inhérent au problème de la reconnaissance d'objets d'une manière générale. En effet, étant donné que le système est destiné à des tâches de téléopération, le problème de l'identification de l'objet dans la scène ne se pose pas vu que cette tâche est réalisée par l'opérateur. Par contre, l'opérateur doit savoir, avec une grande précision, la pose de l'objet dans la scène pour mettre à jour son environnement de travail. De plus, le recalage est un élément essentiel pour la superposition du monde virtuel avec l'image dans les applications de réalité augmentée (RA). L'architecture du système élaboré est inspirée des systèmes utilisés par [Mundy et al., 1996], [Rothwell et al.; 1995] et [Zisserman et al., 1995], elle est illustrée globalement dans la figure (2.13).

Nous avons élaboré deux approches qui respectent cette architecture mais dont les modules diffèrent en fonction du type d'image d'entrée utilisé. La première approche exploite une seule image d'intensité, nous l'avons appelé approche photoclométrique, elle utilise un invariant global pour appairer l'image avec le modèle, et de ce fait suppose que l'objet soit isolé dans la scène. Elle est décrite en détail dans le chapitre (3). La deuxième approche utilise des points 3D acquis de la surface visible de l'objet grâce à un capteur stéréo actif que nous avons conçu. Elle utilise un invariant local pour appairer le modèle avec la scène, et tolère donc des occultations partielles de l'objet. Nous l'avons appelée approche clinométrique et elle est présentée dans le chapitre (4).

**Base de données :** Les approches élaborées sont des approches "model-based" et donc supposent la connaissance à priori du modèle de l'objet de la scène. La représentation que nous avons adopté pour modéliser l'objet de forme libre est la représentation en polygones

maillés (voir chapitre 1). Cette représentation nous permet de construire hors ligne, la base de données dans laquelle on va stocker les informations invariantes sous forme de graphes pour l'approche photométrique et sous forme d'une table pour l'approche clinométrique. Ces informations nous permettront d'effectuer les appariements image/modèle.

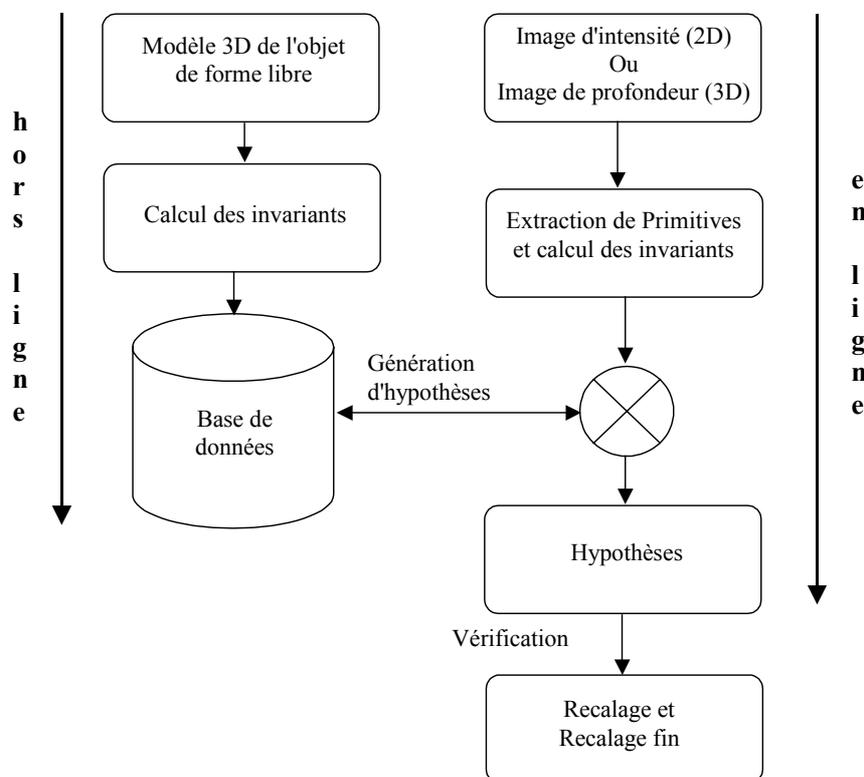


Figure 2.13 : Architecture globale du système de recalage élaboré

**Extraction de primitives et calcul des invariants :** Cette étape prend en entrée l'image de la scène (intensité ou profondeur) et produit les primitives nécessaires pour calculer les invariants. Généralement, plusieurs types d'invariants sont nécessaires pour discriminer les différentes régions de l'objet. Ces invariants forment ce que l'on appelle un vecteur d'invariants qui représente un point dans l'espace multidimensionnel des invariants.

**Génération d'hypothèses :** Après le calcul des invariants à partir de l'image d'entrée, leurs valeurs sont comparées à ceux qui sont stockés dans la base de données, si la mise en correspondance est établie alors il y aura génération d'une hypothèse sur l'identification des points ou des régions de l'objet. Cette hypothèse d'appariement permet de calculer la transformation rigide adéquate entre le modèle et l'image.

**Vérification d'hypothèses :** Pour vérifier une hypothèse générée, on re-projète les primitives du modèle 3D de l'objet sur le plan image selon la transformation trouvée, si ces primitives re-projetées ne coïncident pas (selon un seuil de tolérance fixé) avec les primitives images qui leur correspondent alors cette hypothèse est rejetée. De plus, dans le

cas où l'hypothèse serait retenue, l'étape de vérification permet de l'enrichir avec de nouveaux appariements ce qui permet d'améliorer le calcul de la transformation et ainsi d'avoir un recalage plus fin.

Toutes ces étapes ainsi que les différents modules de traitements qui les constituent seront détaillés dans les deux chapitres suivants.

## 2.9 Conclusion

Nous avons décrit dans ce chapitre, les différentes étapes du processus de reconnaissance d'un objet 3D en décrivant à chaque fois les techniques et les outils les plus utilisés, leurs avantages, leurs inconvénients et le type d'application pour lequel ils sont le mieux adaptés. Il paraît clair à travers les multiples approches étudiées que le choix des primitives à exploiter est primordial dans les systèmes de reconnaissance. Or, vu l'objectif que l'on s'est fixé, à savoir recalculer automatiquement un objet de forme libre dans son environnement de travail, le choix de primitives simples de type points ou segments n'est pas du tout envisageable car il est difficile (voire impossible) de les détecter dans une image. Ainsi, nous avons choisi d'exploiter les normales à la surface de l'objet qui donne une information sur la variation de la surface de l'objet. Nous verrons dans le prochain chapitre que cette information est discriminante et permet d'identifier des régions particulières de l'objet à partir de son image d'intensité. A partir de ce choix, nous avons élaboré deux approches d'appariements de l'objet avec son modèle, basées toutes les deux sur l'extraction d'invariants clinométriques à partir soit d'une image d'intensité ou de profondeur. Nous avons choisi la technique des invariants pour tous les avantages qu'elle présente (voir § 2.4).

Notre souci est de développer un système de recalage le plus fiable possible. Il est donc nécessaire de travailler à tous les niveaux du système en apportant des contributions particulières sur plusieurs points de la chaîne de traitement global.

Ainsi notre intérêt s'est particulièrement porté sur la construction de la base de données qui code le mieux possible l'objet de forme libre afin de permettre une reconnaissance rapide et correcte car comme nous l'avons déjà mentionné (chapitre 1) l'amélioration du temps de reconnaissance reste la préoccupation majeure de ce type de système. D'un autre côté, nous nous sommes intéressés, pour la deuxième approche, à la mise en œuvre du capteur stéréo actif qui permet une reconstruction 3D très précise car basée sur une méthode d'extraction de primitives subpixelique.

Nous testerons les deux approches élaborées sur des images synthétiques pour la première et réelles pour la deuxième, ce qui nous permettra d'évaluer les performances de chacune d'entre elles et donnera également leurs limites d'utilisation. Finalement, nous essayerons de mener une étude critique et constructive sur les méthodes implantées et les résultats obtenus afin d'exposer les perspectives et améliorations à développer dans le futur.



# Chapitre 3

## Approche Photoclinométrique

Ce chapitre présente la solution photoclinométrique<sup>1</sup> que nous avons élaborée pour recalculer automatiquement l'objet de forme libre à partir d'une image d'intensité. En effet, nous allons démontrer que l'utilisation de la photoclinométrie, sous certaines conditions, permet d'extraire à partir de l'image des primitives pouvant être appariées sans équivoque avec des primitives du modèle de l'objet grâce à un invariant photométrique global que nous avons défini. Pour cela, nous allons tout d'abord donner une présentation générale de l'ensemble du procédé en explicitant les hypothèses de base que nous avons considérées. Ensuite nous allons décrire comment nous avons construit, hors ligne, la base de données du modèle qui caractérise les régions de ce dernier, d'une manière appropriée en utilisant cet invariant. Dans une troisième phase, nous allons détailler la procédure d'appariement automatique image/modèle, en ligne, étape par étape en précisant les différents traitements effectués sur l'image. Nous présentons ensuite la technique de recalage 3D qui utilise les résultats du module d'appariement pour déterminer la position courante de l'objet. Nous concluons enfin sur les contributions que nous avons apportées ainsi que sur les résultats obtenus.

---

<sup>1</sup> La photoclinométrie est l'analyse de la variation de la normale à la surface de l'objet d'après la photométrie.

### 3.1 Description générale du système

Ce paragraphe décrit les hypothèses de base considérées, le modèle de réflexion diffuse utilisée ainsi que l'architecture de l'approche photoclinométrique.

#### 3.1.1 Hypothèses de base

Dans la mise en œuvre de l'approche photoclinométrique du recalage automatique nous avons considéré les hypothèses de base (contraintes) suivantes :

- 1) les objets 3D de forme libre utilisés sont placés dans une scène éclairée par une source de lumière directionnelle dont la position par rapport au repère monde ( $R_0$ ) est connue à priori.
- 2) la prise de vue de la scène est effectuée par une seule caméra CCD, calibrée au préalable (voir annexe A) et dont l'axe optique est colinéaire avec la source de lumière (cf. figure 3.1).
- 3) nous considérons que les objets utilisés ont des surfaces mates (surface rugueuses au niveau microscopique). Ce type d'objets possède une réflexion diffuse, appelée aussi réflexion Lambertienne [Foley et al., 1990] (la lumière est réfléchi de manière isotrope<sup>2</sup>), leurs surfaces ont donc une luminosité constante de tous les points de vue, et qui ne dépend que de l'angle entre la direction de la source de lumière et de la normale à la surface (i.e. la quantité de lumière vue par la caméra est indépendante de sa position par rapport à la source de lumière).
- 4) le modèle géométrique de l'objet 3D de forme libre est connu à priori et est défini par sa représentation en polygones maillés (cf. chapitre 1).

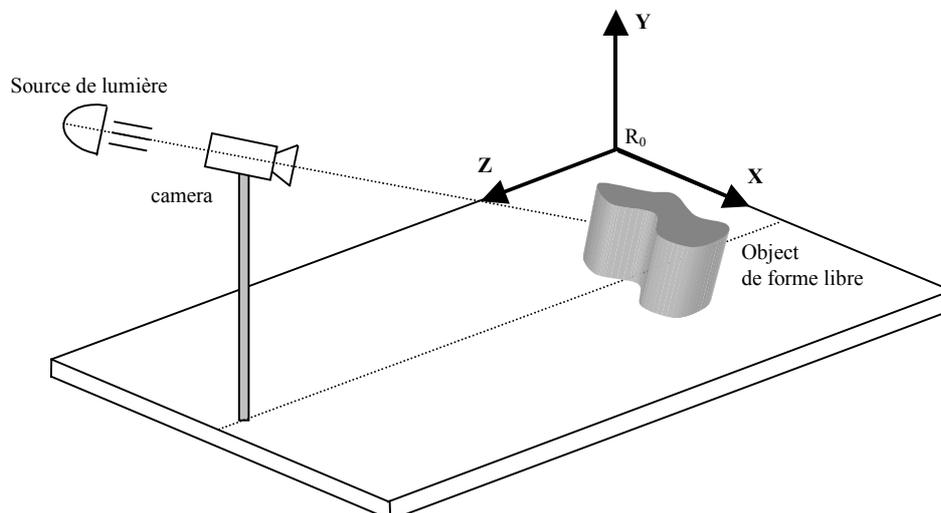


Figure 3.1 : Banc d'essai

<sup>2</sup> Dans toute les directions

### 3.1.2 Modèle de la réflexion diffuse

Comme nous l'avons mentionné ci-dessus, la luminosité apparente d'une surface lambertienne ne dépendra que de l'angle  $\theta$  entre la direction  $\vec{L}$  vers la source lumineuse et la normale  $\vec{N}$  à la surface (cf. figure 3.2).

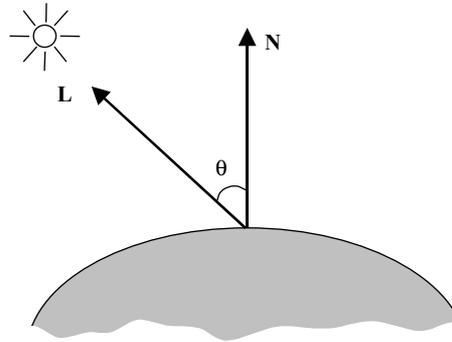


Figure 3.2 : Réflexion diffuse

La figure (3.3) montre comment un faisceau lumineux qui frappe la surface intercepte une aire inversement proportionnelle au cosinus de l'angle entre le faisceau et  $\vec{N}$ . Si le faisceau à une section infinitésimale  $dA$ , le faisceau intercepte une aire  $dA/\cos(\theta)$  sur la surface. Pour un rayon lumineux incident, la quantité d'énergie reçue par  $dA$  est inversement proportionnelle au  $\cos(\theta)$ . Ceci est vrai pour toute surface, indépendamment du matériau.

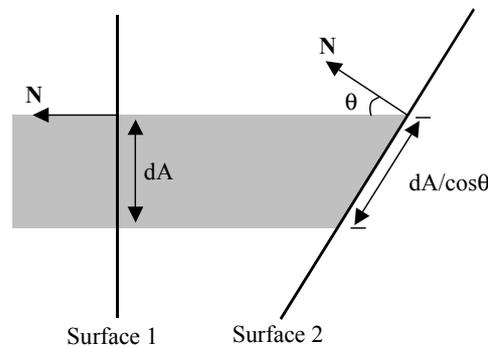


Figure 3.3 : Aire interceptée par les faisceaux élémentaires de section  $dA$

D'un autre côté, les surfaces Lambertiennes partagent la propriété connue sous le nom de loi de Lambert, selon laquelle la quantité de lumière réfléchie par une unité de surface  $dA$  vers l'observateur est proportionnelle au cosinus de l'angle entre l'axe de visée et la normale  $\vec{N}$  à la surface. Comme la surface vue est proportionnelle au cosinus du même angle, les deux facteurs s'annulent réciproquement. Par exemple, lorsque l'angle d'incidence du regard augmente, l'observateur intercepte une plus grande aire, pendant que la quantité de lumière réfléchie diminue en conséquence. Ainsi, pour les surfaces

Lambertiennes, la quantité de lumière vue par l'observateur est indépendante de la position de l'observateur et est proportionnelle à  $\cos(\theta)$ , angle d'incidence du faisceau lumineux.

Ainsi l'équation d'illumination diffuse est donnée par :

$$I = I_p K_d \cos(\theta) \quad (3.1)$$

$I_p$  est l'intensité de la source lumineuse;  $K_d$  est le coefficient de réflexion diffuse; c'est une constante comprise entre 0 et 1 et qui varie d'un matériau à un autre. L'angle  $\theta$  doit être compris entre 0 et  $\pi/2$  si la source lumineuse doit avoir un effet sur la surface éclairée. En d'autres termes, nous considérons les surfaces auto-masquantes, ainsi une lumière placée derrière la surface ne peut l'illuminer. Le modèle de la réflexion diffuse est une loi empirique qui s'applique sur des surfaces mates comme le papier ou le plâtre.

En supposant que les vecteurs  $\vec{N}$  et  $\vec{L}$  sont normalisés, on peut réécrire l'équation (3.1) en utilisant le produit scalaire :

$$I = I_p K_d (\vec{N} \cdot \vec{L}) \quad (3.2)$$

où  $\vec{N} \cdot \vec{L}$  représente le produit scalaire des vecteurs  $\vec{N}$  et  $\vec{L}$ .

C'est ce modèle que nous avons choisi d'adopter pour mener nos analyses photométriques sur les images d'intensité. Nous avons fixé, sans perte de généralité, l'intensité de la lumière incidente à 1. Par ailleurs, le coefficient de réflexion diffuse  $K_d$  peut être fixé à la valeur photométrique maximum observée sur une portion de l'objet dont on sait qu'elle est orientée vers la source de lumière.

De plus nous considérons que la source lumineuse est directionnelle, ce qui implique que l'angle d'incidence de ses rayons par rapport à des surfaces de même normale peut être considéré comme constant. En d'autres termes le vecteur  $\vec{L}$  est constant pour cette source lumineuse.

### 3.1.3 Modèle géométrique de l'objet de forme libre

Afin d'illustrer l'ensemble de l'algorithme et les résultats obtenus au fur et à mesure de son déroulement nous prenons l'exemple particulier de la Vénus définie par sa représentation en polygones maillés et dont le maillage est illustré dans la figure (3.4a).

Ainsi, le modèle géométrique 3D initial de la Vénus est donné par (cf. figure 3.4b):

- les sommets du maillage définis par leurs coordonnées 3D dans un référentiel lié à l'objet et centré sur son centre de gravité. Chaque sommet est identifié grâce à son index qui donne sa position dans le tableau des sommets.
- les normales aux sommets définies par leurs coordonnées normalisées.
- les faces du maillage qui sont dans notre cas des faces triangulaires, elles sont définies par une liste. Chaque ligne de cette liste définit une face particulière du maillage en donnant les index des sommets qui la constituent.

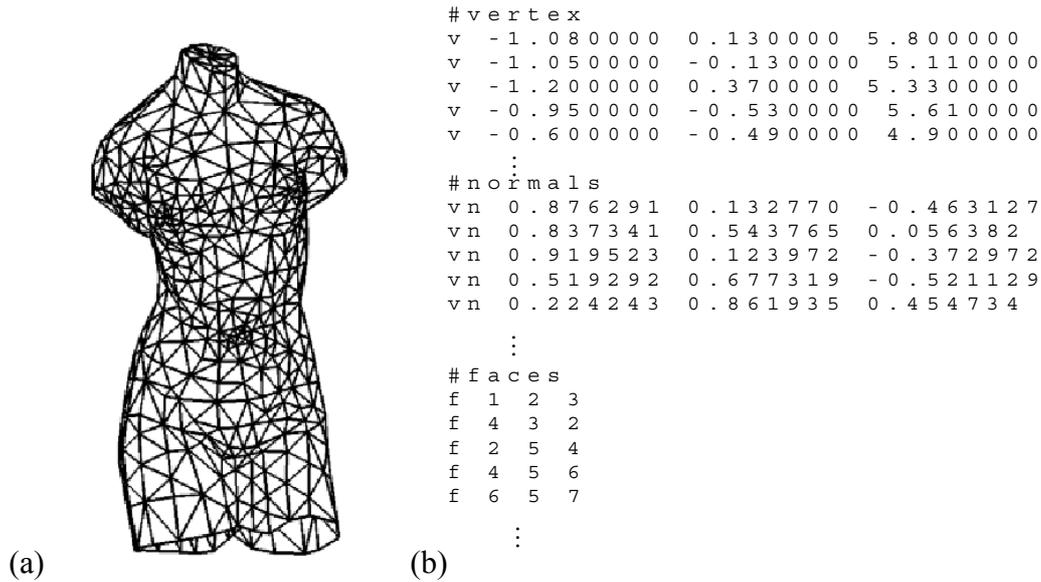


Figure 3.4 : Modèle géométrique 3D de la Vénus

De plus, pour des besoins de la méthode, nous avons enrichi ce modèle en lui rajoutant les normales aux faces. Chaque normale à la face est déterminée par le produit vectoriel de deux vecteurs directeurs de deux de ses trois arêtes, pris dans le sens trigonométriques (cf. figure 3.5):

$$\vec{V}_f = \vec{V}_{s_1s_2} \times \vec{V}_{s_1s_3} \quad (3.3)$$

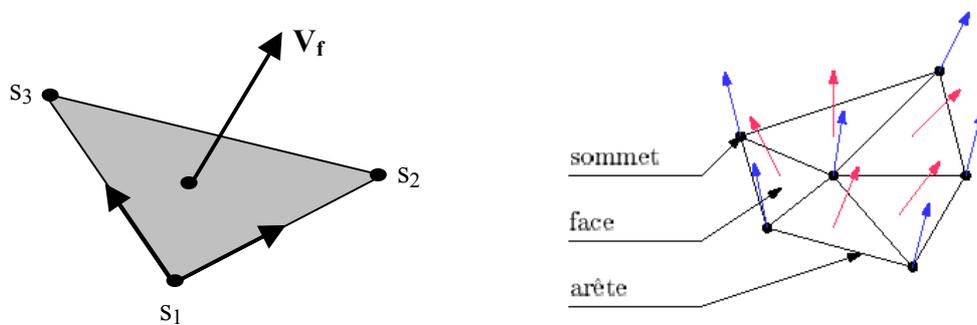


Figure 3.5 : Définition des normales aux surfaces du maillage

### 3.1.4 Principe de la méthode photoclinométrique

Le fait de considérer le modèle de réflexion diffuse nous permet d'affirmer que chaque niveau de gris dans l'image de luminance de l'objet dépend uniquement de l'angle entre la normale à la surface de l'objet au point considéré et la direction de la source de lumière. De plus, la détermination de la valeur de cet angle se fera, à chaque fois que cela est

nécessaire, par une simple inversion du modèle de réflexion diffuse donné par l'équation (3.1). Pour exploiter à bon escient cette propriété, nous avons choisi de déterminer les zones de l'image les plus brillantes que nous avons appelé taches et qui vont constituer les primitives images à partir desquelles nous allons réaliser les appariements image/modèle. Ces taches correspondent à des portions de surfaces de l'objet orientées dans la direction de la source de lumière ( $\theta=0$ ), or comme cette dernière est connue à priori dans le repère de la scène (hypothèse 1), les normales 3D des taches exprimées dans le repère monde sont donc déterminées. Pour chaque tache nous calculons les angles entre la normale extraite et les normales aux points qui sont tout autour de la tache. La répartition des angles ainsi trouvés est un invariant projectif pour le modèle de réflexion diffuse, et donc ne dépend pas du point de vue. De plus, cette répartition est discriminante pour les objets de forme libre. Ainsi, chaque tache extraite de l'image aura une répartition angulaire spécifique qui permettra de l'identifier dans le modèle 3D.

Afin de pouvoir réaliser des appariements automatiques image/modèle, on construit, hors ligne, une base de données du modèle 3D sous forme de tables d'aspects, en se basant sur le principe de ces répartitions angulaires [Ababsa et al., 2002a]. Chaque aspect dans cette table sera défini par tous les sommets du modèle 3D ayant une normale identique, que nous appelons normale de référence (définie avec une tolérance angulaire près). Une des difficultés consiste à identifier les différents aspects discriminants de l'objet et déterminer une représentation simple et pertinente. L'utilisation des méthodes classiques des graphes d'aspects (voir chapitre 2) nécessite de répertorier les différents aspects visuels en cherchant les frontières qui les séparent les uns des autres. Ces frontières correspondent à des événements visuels particuliers désignant des variations significatifs de l'aspect visuel. Ces variations sont généralement faciles à détecter lorsqu'il s'agit de manipuler des objets polyédriques car elles correspondent à l'apparition ou la disparition d'une face de l'objet. Par contre, lorsqu'il s'agit d'objets de forme libre, la discrimination entre les aspects visuels en utilisant des événements visuels basés sur les primitives géométriques (tel que les segments de droite) n'est pas du tout évidente voire impossible. En effet, un objet de forme libre vu de deux points de vue différents peut avoir la même apparence dans les deux images de luminance respectives. L'originalité de la méthode que nous proposons est qu'elle définit des aspects de type photométrique pour discriminer les différentes vues de l'objet de forme libre. Elle associe à chaque aspect photométrique un point de la sphère de vue, qui permet d'englober l'objet. Cette solution simple n'entraîne pas de calculs trop complexes et évite de devoir rechercher les aspects analytiquement. Un point important concerne l'information qui doit représenter les aspects photométriques retenus. Pour chaque aspect, nous stockons les répartitions angulaires de tous les sommets qui le constituent.

Enfin, le dernier choix à établir concerne la méthode de mise en correspondance. Comme nous l'avons mentionné précédemment (chapitre 2), plusieurs paradigmes existent pour répondre à ce problème. Le défaut majeur de ces méthodes est la complexité du temps de calcul qu'elles engendrent. Nous avons choisi la méthode de génération/vérification d'hypothèses, cette technique permet d'effectuer des appariements très rapidement grâce à des calculs simples. Nous remédions à son principal handicap, en l'occurrence son caractère combinatoire trop important, en fournissant à l'entrée du système un nombre restreint de taches (juste le nombre qu'il faut pour déterminer la transformation rigide).

La figure (3.6) reprend les deux grandes parties du système en ligne et hors ligne qui permettent de construire la base de données du modèle et d'établir la mise en correspondance entre les primitives extraites de l'image et les primitives du modèle. On peut constater l'importance de l'invariant photométrique qui conditionne la réussite du processus d'appariement.

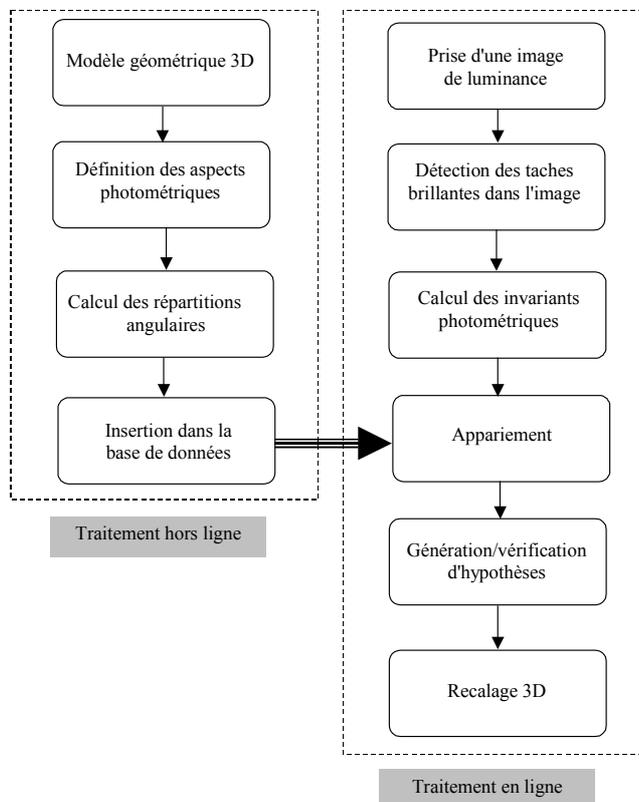


Figure 3.6 : Synoptique global de l'approche photoclinométrique

La suite de ce chapitre est dédiée à la description des étapes de construction de la base de données, de l'appariement image/modèle et de recalage 3D.

## 3.2 Construction de la base de données

Dans cette partie, nous examinons le traitement hors ligne qui est effectué afin d'établir les principaux aspects photométriques de la Vénus à intégrer dans la base de données afin de minimiser le temps de reconnaissance une fois "en ligne".

### 3.2.1 Suréchantillonnage du maillage

Afin d'avoir la meilleure adéquation possible entre l'image et le modèle, celui-ci doit être discrétisé d'une manière "fine". En effet, pour réaliser des mises en correspondance entre des zones de l'image et du maillage, nous utilisons les sommets et les faces du maillage. Si

ces derniers sont plus nombreux, ils permettront de déterminer plus précisément la zone du maillage incriminée. Les résultats fournis par l'algorithme de recalage n'en seront que meilleurs. Nous avons donc mis au point un algorithme qui permet de générer un maillage quatre fois plus fin et donc plus approprié, à partir du maillage initial.

La figure (3.7) présente la subdivision réalisée au niveau d'une face triangulaire ainsi que le maillage suréchantillonné obtenu.

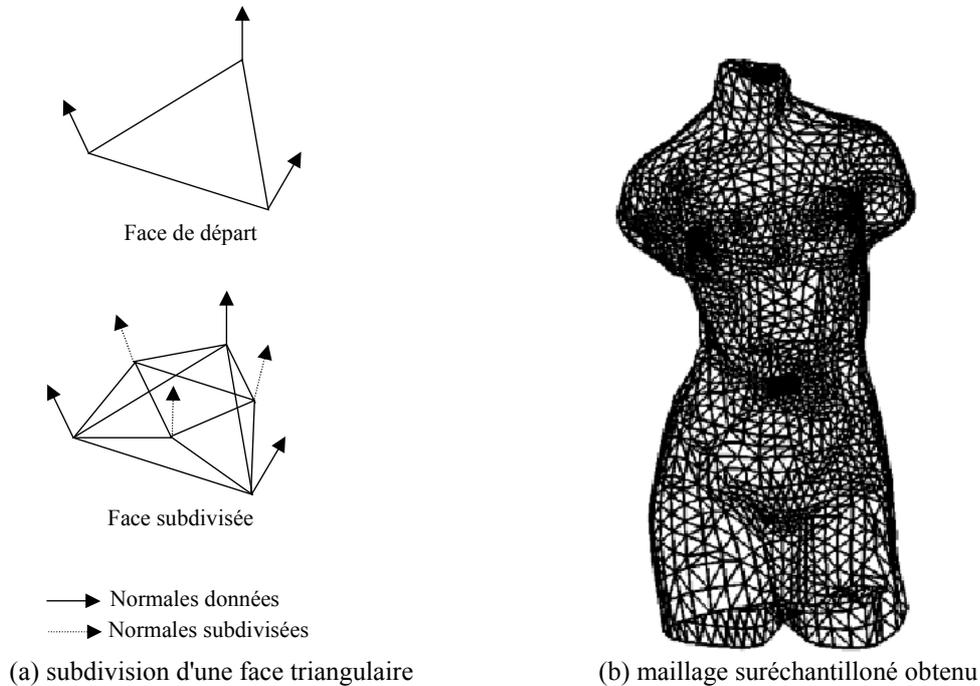


Figure (3.7) : Suréchantillonnage du maillage

Cet algorithme fonctionne de la manière suivante :

1. lister toutes les arêtes constituant le maillage initial.
2. calculer le milieu de chacune de ces arêtes. On appellera l'ensemble de ces milieux ensemble 'M'.
3. calculer la normale en tout point de l'ensemble 'M'. Chaque normale est égale à la moyenne des normales aux deux sommets, extrémités de l'arête considérée. La normale ainsi calculée est ensuite rendue unitaire. L'ensemble de toutes ces normales est noté 'N'.
4. pour chaque point de l'ensemble 'M', calculer la moyenne des distances aux deux plans formés chacun par le couple (sommets, normale) des deux sommets extrémités de l'arête porteuse de ce point (cf. figure 3.8).
5. décaler chaque point de 'M' de la distance calculée en (4). Ce décalage s'effectue sur les droites passant par les points de 'M' et orientées selon les normales 'N'. Le nouvel ensemble de points obtenus est appelé ensemble 'P'.

6. Incorporer les ensembles 'P' et 'N' dans le maillage initial.

Cette méthode permet de multiplier par 2 le nombre de sommets du maillage et par 4 le nombre de faces triangulaires. Si on veut obtenir un suréchantillonnage encore plus fin, il suffira alors de réitérer le processus.

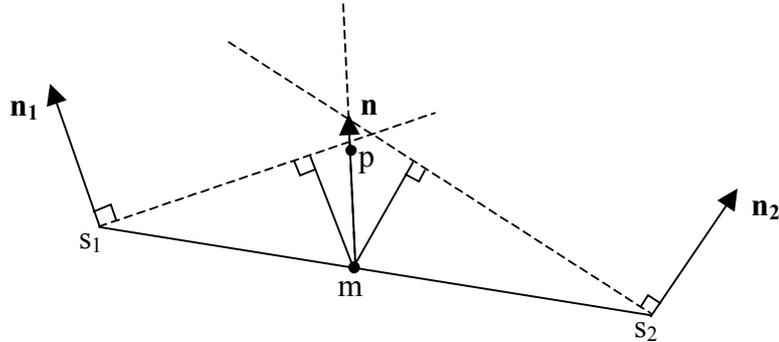


Figure 3.8 : Construction d'un point supplémentaire du maillage

### 3.2.2 Echantillonnage de la sphère de vue

Puisque dans notre méthode, chaque aspect photométrique est défini par une normale de référence qui est associée à un point de vue particulier sur la sphère de vue, il est important de développer le parcours de la sphère de vue pour établir un échantillonnage adéquat des aspects de l'objets. Afin de générer une distribution uniforme des différentes lignes de vue, Nous avons utilisé une grille ISEA ("Icosahedral Snyder Equal Area"). Il s'agit d'un système de grille qui permet de subdiviser une sphère en des surfaces de même forme et d'aires à peu près égales. Cette méthode repose sur l'approximation de la sphère par un icosaèdre. Chaque face est un triangle équilatéral, nous subdivisons chacune de ces faces en triangle plus petits par subdivision de leurs arêtes. En général, le nombre de facettes est donné par  $N=20 \times n^2$  où  $n$  représente la division de l'arête du triangle (cf. figure 3.9). Chacun des points ainsi obtenus est reprojété sur la sphère (après normalisation du vecteur qui lui correspond). Nous obtenons finalement un jeu de normales uniformément réparties sur l'ensemble de la sphère.

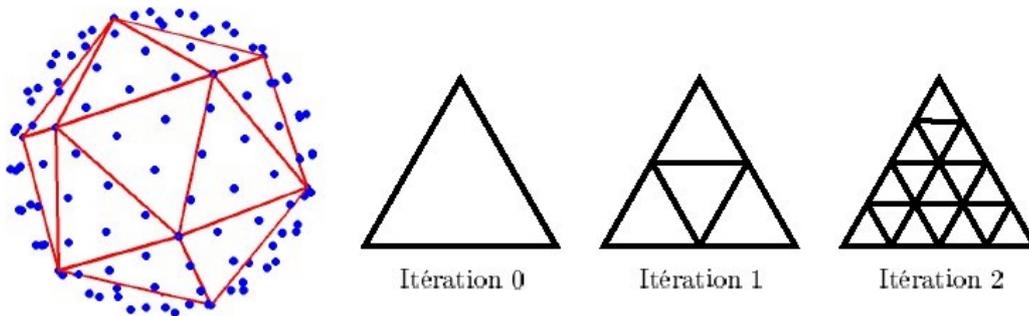


Figure 3.9 : Grille ISEA

L'ordre de la subdivision dépend, d'une part, de la précision que l'on s'est fixée pour discriminer les aspects photométriques, et d'autre part, du temps d'exécution engendré et qui doit rester acceptable. En fait nous devons choisir un ordre de subdivision qui constitue un compromis entre ces deux contraintes. Le tableau (3.1) donne une idée sur le nombre de normales générées en fonction du nombre d'itérations de subdivision.

Itération	1	2	3	4	5
Nombre de normales	42	162	642	2562	10242
Angle min entre normales	31,72°	14,55°	6,85°	3,32°	1,63°
Angle max entre normales	35,63°	17,17°	9,20°	4,70°	2,36°

Tableau 3.1 : Normales générées par la grille ISEA

Les différents tests effectués (voir § 3.5.2) démontrent qu'une subdivision d'ordre 4 de la grille ISEA suffit pour générer un ensemble de normales représentatives des aspects photométriques (2562 normales) et pour lesquels le temps d'exécution reste raisonnable. La figure (3.10) montre la sphère de vue séparée en 2562 facettes.

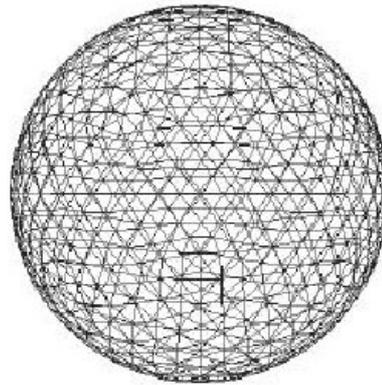


Figure 3.10 : Sphère de vue composée de 2562 facettes

### 3.2.3 Construction de la table d'aspects photométriques

Comme nous l'avons mentionné précédemment, un aspect photométrique est défini par une normale de référence associée à chaque fois à une direction de la sphère de vue. Et comme nous avons subdivisé la sphère de vue en 2562 facettes, nous aurons à stocker 2562 aspects photométriques potentiels. Afin de réaliser l'ensemble des calculs, il est nécessaire que l'objet soit placé au centre de la sphère de vue (figure 3.11), et ensuite d'effectuer un changement de repère. Ce changement de repère consiste à prendre le centre de gravité de l'objet, défini par le point de coordonnées (0,0,0) dans le repère de la Vénus, comme centre de la sphère. Ainsi l'objet localisé au centre de la sphère est vu à partir du centre de chacune des facettes constituant la sphère de vue.

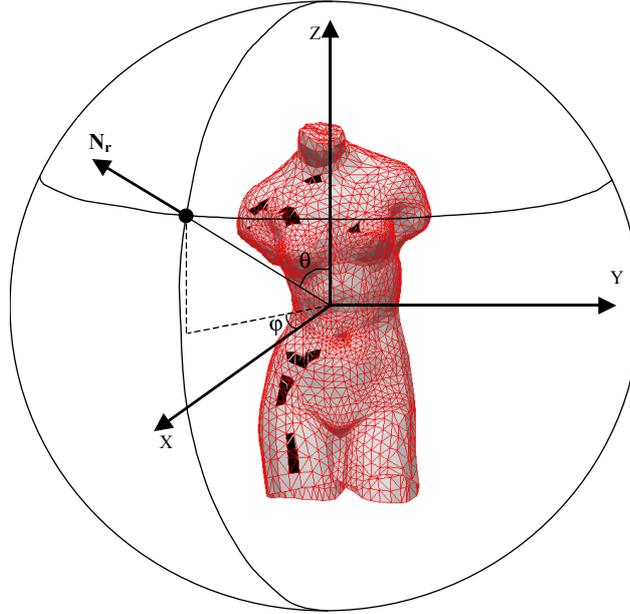


Figure 3.11 : Un exemple d'un aspect photométrique de la Vénus. Les régions du maillage appartenant à cet aspect (i.e. ayant une normale égale à  $N_r$ ) sont colorées en noir sur le modèle

La direction de la ligne de vue est définie à chaque fois par les deux paramètres  $\varphi$  et  $\theta$  (cf. figure 3.11). Pour une normale de référence normalisée  $N_r(x,y,z)$  donnée,  $\varphi$  et  $\theta$  peuvent être données par les relations suivantes:

$$\varphi = \arctan\left(\frac{y}{x}\right) \quad \text{où } 0 \leq \varphi \leq 2\pi \quad (3.4)$$

$$\theta = \arccos(z) \quad \text{où } 0 \leq \theta \leq \pi \quad (3.5)$$

Chaque aspect photométrique sera constitué par les sommets du maillage dont les normales sont égales à la normale de référence considérée. Or, dans le maillage de l'objet; il est généralement rare de trouver des sommets dont les normales soient rigoureusement identiques, de plus, les zones dans l'image qui ont le même niveau de gris correspondent en réalité à des zones du maillage dont les normales diffèrent quelque peu entre elles. Ainsi nous avons fixé une tolérance angulaire de  $5^\circ$  (cette valeur est déterminée expérimentalement, voir § 3.5.2) pour chaque normale de référence. Pour chaque aspect, nous prenons donc, tout les sommets du maillage dont les normales font un angle avec la normale de référence inférieur ou égale à  $5^\circ$ .

Pour cela, nous calculons le produit scalaire de chacune des normales aux sommets du maillage avec la normale de référence, le produit scalaire étant égal au cosinus de l'angle entre les deux vecteurs (en s'assurant au préalable que chacun des deux vecteurs est unitaire), on sélectionne donc les sommets pour lesquels la valeur de ce produit scalaire est supérieure à celle du cosinus de l'angle de tolérance :

$$\vec{N}_r \cdot \vec{N}_s \geq \cos(5^\circ) \quad (3.6)$$

Nous stockons les différents aspects ainsi générés sous forme d'une table multidimensionnelle. Chaque entrée de cette table correspond à un aspect donné et contient d'une part, les coordonnées 3D de la normale de référence correspondante, et d'autre part, la liste des index de tous les sommets du maillage appartenant à cet aspect (cf. figure 3.12).

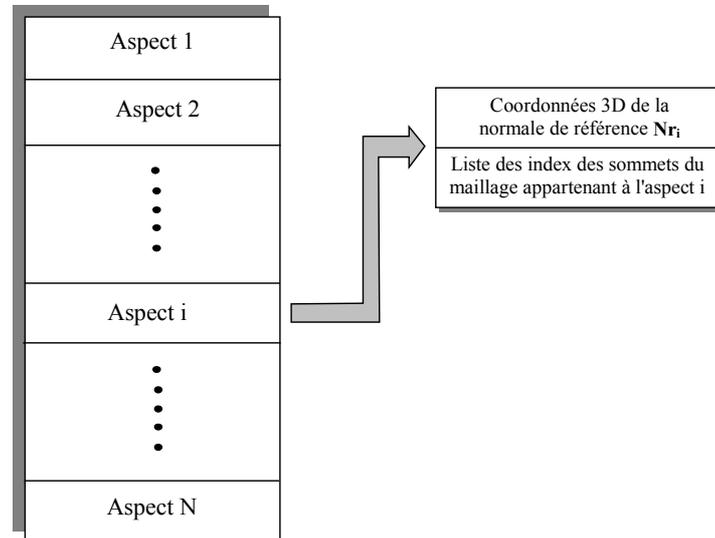


Figure 3.12 : Structure de la table d'aspects photométriques

Notons toutefois que ces listes peuvent être vides, c'est le cas lorsque le modèle considéré ne possède pas des normales aux sommets qui soient orientées dans la direction de la normale de référence associée à un aspect photométrique donné.

### 3.2.4 Indexation des sommets du maillage

L'étape suivante consiste à déterminer puis stocker, pour chaque sommet du maillage, l'invariant projectif qui permettra de l'identifier dans l'image de luminance lors de la phase en ligne. Il s'agit en l'occurrence de déterminer, pour chaque sommet, la répartition des angles que fait la normale à ce sommet avec les normales aux sommets situés dans un voisinage. Ce voisinage doit être défini d'une manière très précise car c'est de lui que dépendra la précision des appariements image/modèle. Cette section a pour but, d'une part, de détailler la procédure que nous avons élaborée pour définir ce voisinage dans le modèle, et d'autre part, d'illustrer la construction de la table d'indexation des sommets du maillage.

Etant donné que la reconnaissance est basée sur une image 2D de luminance, il est impératif de ramener la représentation 3D du maillage à une représentation 2D. Comme nous l'avons mentionné dans le chapitre (2), il existe de nombreuses possibilités de projections permettant de modéliser la prise de vue par une caméra. Dans notre approche, nous utilisons la projection orthographique pour créer les aspects photométriques du modèle en considérant que leurs équivalents image seront similaires.

L'opération suivante consiste à projeter orthographiquement, pour chaque normale de référence, les sommets du maillage sur le plan tangent à la sphère de vue défini par cette même normale de référence et dont l'équation est donnée par (cf. figure 3.13):

$$(x_{vue}-x).cos(\varphi).sin(\theta)+(y_{vue}-y).sin(\varphi).sin(\theta)+(z_{vue}-z).cos(\theta)=0 \quad (3.7)$$

Ainsi, à partir de la connaissance du plan tangent à la sphère de vue, des coordonnées des sommets du maillage de l'objet et de l'équation de la droite de vue, nous pouvons calculer l'ensemble des projetés orthographiques à l'aide de l'équation suivante:

$$\begin{cases} x_p = x_s + x_{vue} \cdot t \\ y_p = y_s + y_{vue} \cdot t \\ z_p = z_s + z_{vue} \cdot t \end{cases} \quad (3.8)$$

$$t = \frac{cos(\varphi).sin(\theta)(x_{vue}-x_s)+sin(\varphi).sin(\theta)(y_{vue}-y_s)+cos(\theta)(z_{vue}-z_s)}{x_{vue}.cos(\varphi).sin(\theta)+y_{vue}.sin(\varphi).sin(\theta)+z_{vue}.cos(\theta)}$$

les triplets  $(x_s, y_s, z_s)$  et  $(x_p, y_p, z_p)$  représentent respectivement les coordonnées d'un sommet du maillage et de son projeté sur le plan tangent à la sphère et passant par le point de coordonnées  $(x_{vue}, y_{vue}, z_{vue})$ .

Lors de la projection du maillage sur le plan tangent, nous devons prendre en compte la cohérence visuelle, c'est à dire ne considérer que les sommets visibles du maillage. Tout les sommets du maillage sont définis avec une normale pointant vers l'extérieur de la Vénus. Ainsi, par rapport à une ligne de vue donnée, un sommet caché est détecté par un produit scalaire négatif ou nul, entre le vecteur normal à ce sommet et la normale de référence qui correspond aussi à la normale au plan tangent de la facette de la sphère d'observation.

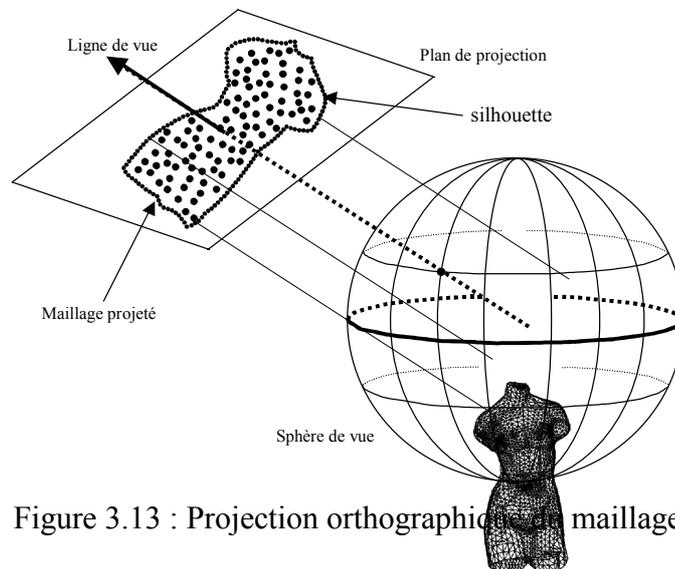


Figure 3.13 : Projection orthographique d'un maillage

Pour déterminer la répartition angulaire associée à chaque sommet de référence projeté, en l'occurrence le point  $P_1$  dans la figure (3.14a), nous devons échantillonner les valeurs des angles associés aux sommets se trouvant autour de lui et projetés sur son plan tangent.

Nous choisissons un deuxième point de référence (nommé  $P_2$ ) de telle sorte que le voisinage soit défini de manière similaire pour le modèle et pour l'image. Le point considéré est le point de la silhouette du maillage projeté le plus proche du sommet de référence.

Trois méthodes pour construire un voisinage et calculer une répartition angulaire ont été envisagées :

1. tracer un cercle ayant pour centre le point  $P_1$  et pour rayon la distance  $[P_1P_2]$ . Tracer des rayons espacés entre eux d'un angle  $\theta$ . L'échantillonnage des angles se fera sur chacun des rayons ainsi définis (cf. figure 3.14b).
2. Dans certains cas, la méthode précédente ne fonctionne pas correctement, notamment lorsque les projetés des sommets considérés sont très proches du bord de la silhouette car le cercle devient très petit. Pour remédier à ce problème une autre solution consiste à déterminer le point  $P_3$  situé à la deuxième intersection entre la silhouette et la droite  $(P_1P_2)$  (la première intersection étant  $P_2$ ), puis nous définissons deux cercles ayant pour centres les milieux des deux segments de droite  $[P_1P_2]$  et  $[P_1P_3]$  respectivement, auxquels nous appliquons la première méthode (cf. figure 3.14c).

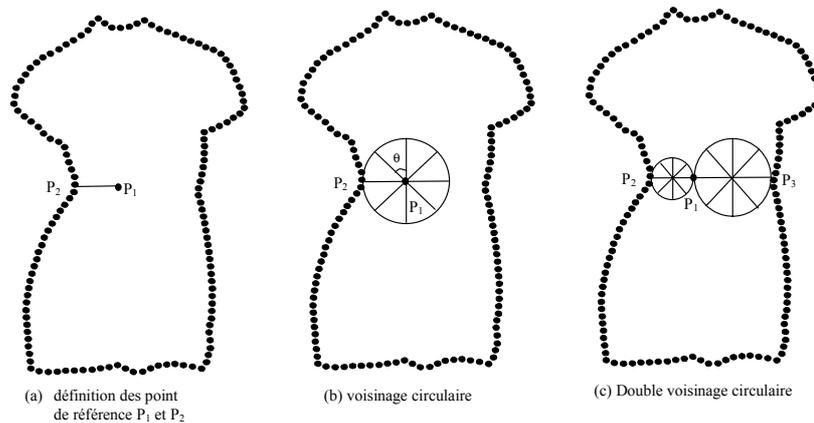


Figure 3.14 : Définition du voisinage

3. La deuxième méthode présente toujours les mêmes inconvénients que la première méthode, à savoir lorsqu'on se rapproche du bord de la silhouette, les mesures angulaires ne sont pas assez fiables. D'un autre côté, cette méthode ne donne pas une idée de la position relative du sommet projeté considéré par rapport à la silhouette de l'objet. La solution finale que nous avons adoptée est celle d'un profil en croix (cf. figure 3.15).

On étudie l'évolution des angles suivant la droite  $[P_2P_3]$  ainsi que sur une droite perpendiculaire passant par le point  $P_1$  (les intersections de cette droite avec le contour de la silhouette sont nommées  $P_4$  et  $P_5$ ). Chaque segment  $[P_1P_i]$   $i=2,3,4,5$  du profil est échantillonné en  $P$  points avec un pas d'échantillonnage donné par :

$$\Delta d_i = \frac{\|P_1P_i\|}{P} \quad i=2,3,4,5 \quad (3.9)$$

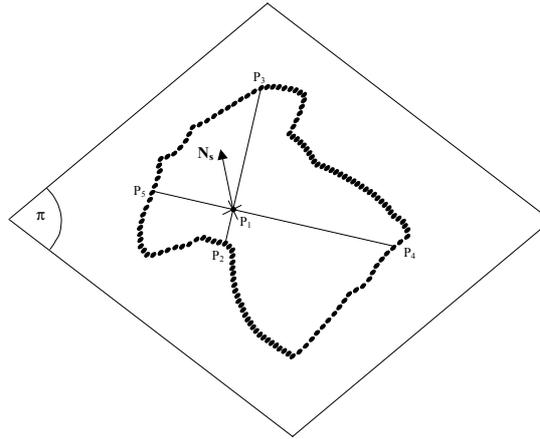


Figure 3.15 : Profil en croix

Le nombre de point  $P$  est déterminé expérimentalement de sorte à avoir une répartition angulaire discriminante (voir § 3.5.2). D'autre part, nous calculons la position relative de chaque sommet projeté considéré du maillage par rapport à la silhouette, elle est définie par le rapport  $R$  des longueurs des axes du profil en croix, tel que:

$$R = \frac{\|P_2P_3\|}{\|P_4P_5\|} \quad (3.10)$$

Le rapport  $R$  est utilisé pour affiner l'appariement entre les taches de l'image et les sommets du maillage.

Enfin, le calcul de la répartition angulaire le long du profil de croix se fait de la manière suivante : pour chaque point échantillonné du profil en croix, nous considérons le sommet du maillage dont le projeté sur le plan est le plus proche de ce point échantillonné (cf. figure 3.16). Le calcul de l'angle pour ce point du profil est déterminé par le produit scalaire entre le vecteur normal à ce sommet et la normale du sommet de référence

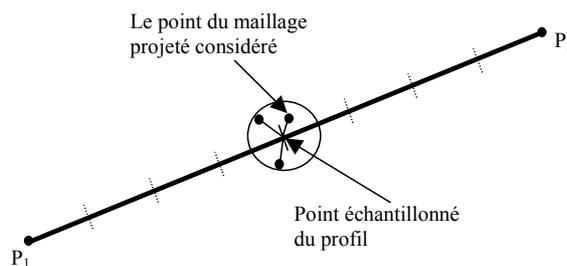
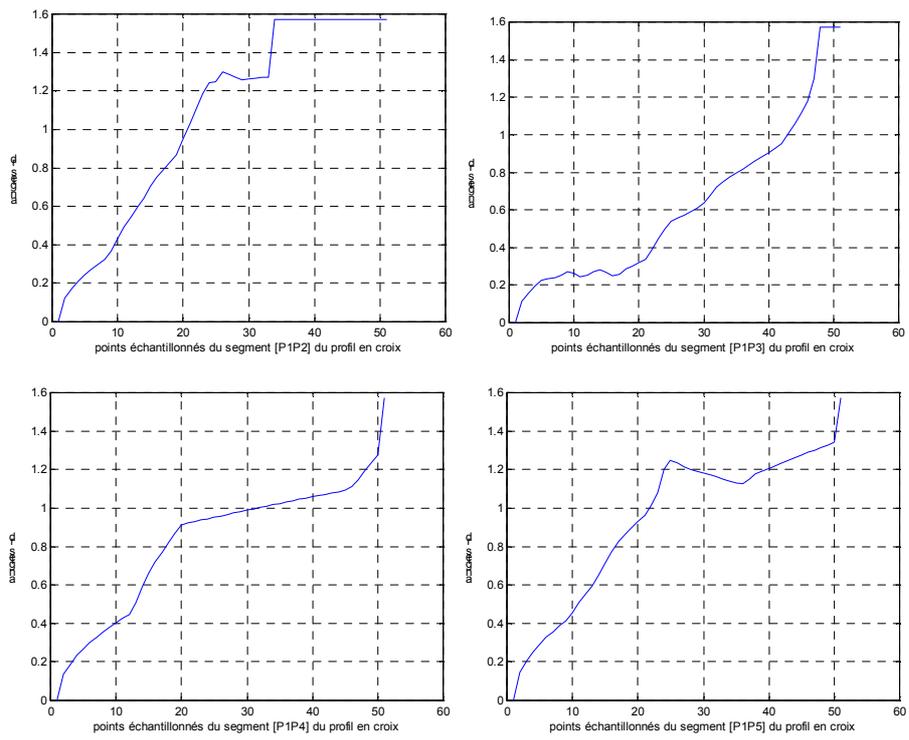
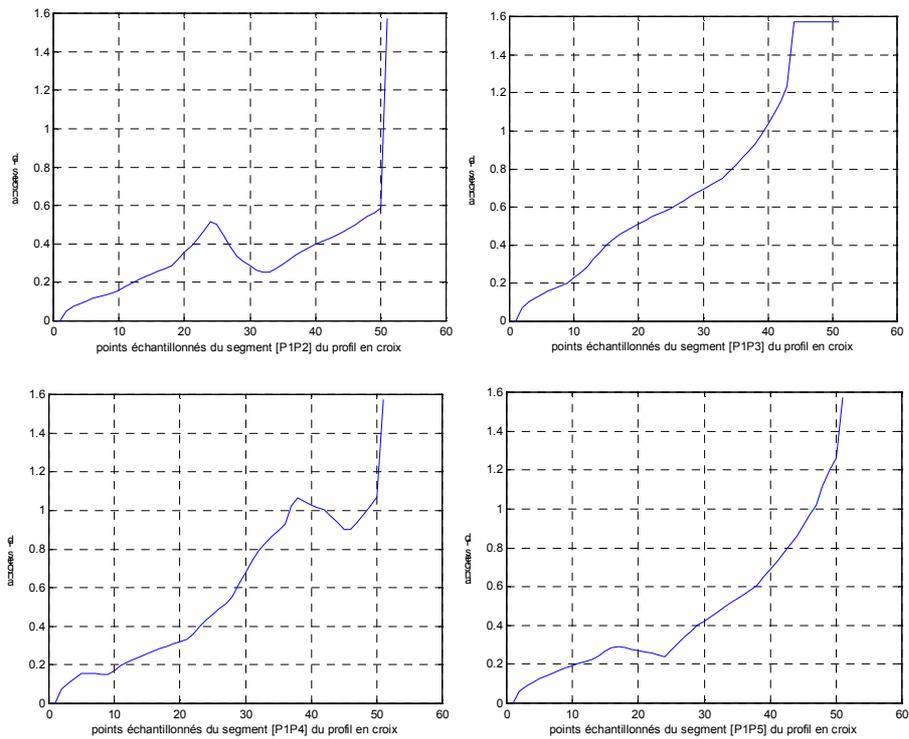


Figure 3.16 : échantillonnage du profil en croix

La figure (3.17) représente les répartitions angulaires associées respectivement aux deux sommets du maillage 1 et 1200, selon les quatre segments qui constituent le profil en croix.



(a) sommet N° 1



(b) sommet N° 1200

Figure 3.17 : Exemples de répartitions angulaires

Nous remarquons que ces répartitions sont discriminantes, et permettent donc de caractériser fidèlement chaque sommet du maillage.

D'un autre côté, comme le calcul du profil en croix est basé sur la détermination de la silhouette de l'objet à partir des projetés des sommets du maillage dans le plan de projection. Nous avons donc élaboré un algorithme qui permet de calculer le contour extérieur du maillage projeté (cf. figure 3.18). Ce dernier fonctionne de la façon suivante: Tout d'abord, nous recensons toutes les arêtes qui constituent le modèle géométrique 3D, une arête relie deux sommets adjacents du maillage. La silhouette sera obtenue en passant d'un sommet projeté à un autre suivant les arêtes projetées. Les arêtes sélectionnées à chaque fois sont celles qui forment le plus petit angle orienté dans le sens trigonométrique entre elles. L'algorithme doit être initialisé en lui précisant un point de départ. Le point de départ choisi est le point projeté du maillage le plus éloigné du sommet de référence projeté dans le plan. Le calcul des angles entre les arêtes se fait par rapport à la droite définie à chaque fois, par le point courant de la silhouette et le point précédemment déterminé.

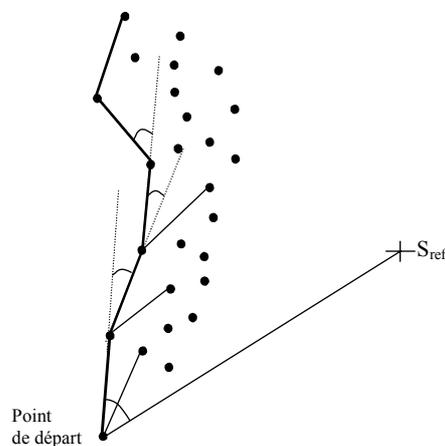


Figure 3.18 : Détection de la silhouette du maillage projeté

Une fois les répartitions angulaires du profil en croix calculées, on détermine pour chaque sommet du maillage la liste des aspects photométriques auxquels il appartient. Ces listes vont servir, comme nous le verrons ultérieurement, à générer d'une manière efficace les hypothèses de reconnaissance des aspects photométriques.

Finalement, nous stockons toutes les informations calculées précédemment dans une table multidimensionnelle que nous appelons table des sommets. Chaque entrée de cette table correspond à un sommet donné du maillage, elle contient la répartition angulaire du sommet, son rapport  $R$  et la liste des aspects photométriques auxquels il appartient (cf. figure 3.18).

Ainsi, la base de données est constituée des deux tables, la table d'aspects photométriques et la table des sommets. A titre indicatif, avec  $M=2719$  sommets,  $N=2562$  aspects photométriques (normales de référence) et  $P=51$  échantillons sur le profil en croix, il faut environ 6 heures pour générer la base de données (sous Matlab).

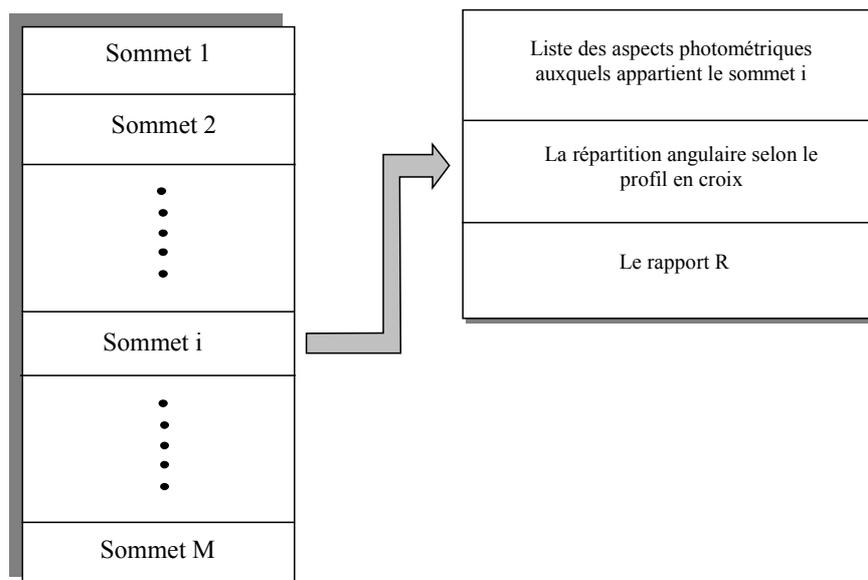


Figure 3.18 : Table des sommets

### 3.3 Appariement automatique image/modèle

Contrairement à l'étape hors ligne décrite précédemment où le temps d'exécution n'est pas un paramètre essentiel, ce dernier devient très important dans l'étape en ligne. En effet, non seulement le procédé doit être rapide, mais de plus la sélection d'un ou plusieurs sommets du modèle doit être précise. Pour cela l'algorithme d'appariement doit avoir une base de comparaison commune entre l'image et le modèle. Des traitements similaires à ceux utilisés pour la génération de la base de données devront être appliqués sur l'image de luminance. Toutefois, deux différences importantes sont à noter, d'une part, l'image est déjà dans un plan, ce qui nous affranchit du problème de la projection, et d'autre part, l'image représente l'objet dans un aspect photométrique, la normale de référence dans ce cas n'est autre que la direction de la source de lumière incidente sur l'objet. Nous avons utilisé des images de synthèse pour tester et valider notre méthode d'appariement automatique.

La figure (3.19) illustre une image de luminance prise de la Vénus dans une pose inconnue et sur laquelle nous allons appliquer les algorithmes élaborés à chaque étape du processus d'appariement en commentant à chaque fois, les résultats obtenus.

Les traitements que nous allons appliquer sur l'image de luminance sont :

- déterminer les zones (taches) les plus brillantes de l'image. Chaque tache brillante, est définie par une agglomération de pixels connexes ayant le niveau de gris le plus élevé dans l'image de luminance.
- compter les taches ainsi trouvées et calculer leurs barycentres.
- déterminer les répartitions angulaires dans le voisinage de chaque barycentre à partir des niveaux de gris observés dans l'image.

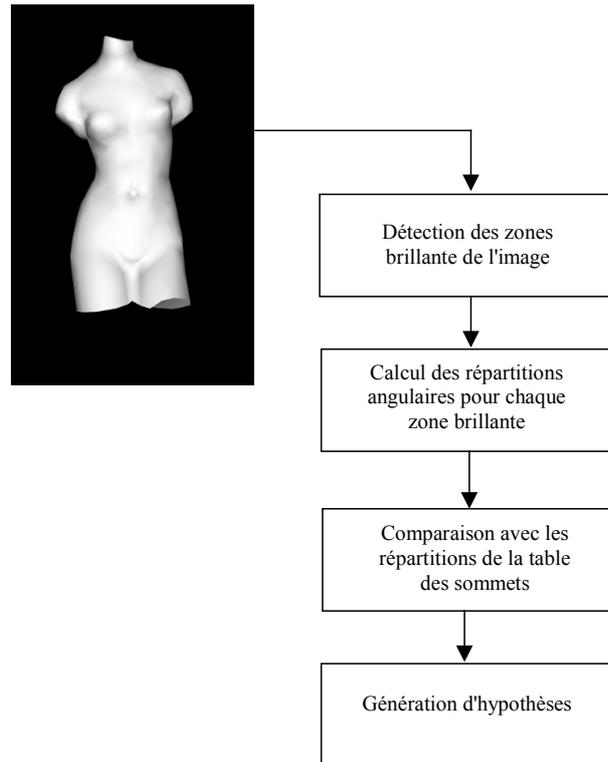


Figure 3.19 : Appariement automatique

Les coordonnées image de chaque barycentre d'une tache sont données par :

$$u_b = \frac{1}{N_b} \sum_{i=1}^{N_b} u_i \quad v_b = \frac{1}{N_b} \sum_{i=1}^{N_b} v_i \quad (3.11)$$

où  $N_b$  et  $(u_i, v_i)$  sont respectivement, le nombre de points appartenant à la tache et leurs coordonnées image.

L'image (3.20a) illustre les taches les plus brillantes détectées dans l'image initiale, elles sont numérotées de 1 à 13, leurs barycentres respectifs sont représentés par des croix.

L'étape suivante consiste à déterminer les répartitions angulaires de chaque barycentre de tache selon le profil en croix. Pour que l'appariement taches/sommets soit correct, le profil en croix sur l'image doit être déterminé de la même manière que celle utilisée sur le modèle à la différence de projection près. Il est donc nécessaire d'extraire tout d'abord la silhouette de l'objet dans l'image de luminance. Comme l'objet est isolé dans l'image, nous avons fixé un seuil pour le niveau de gris qui permet de différencier l'objet du fond (cf. figure 3.20b). Le point du contour le plus proche de la tache va correspondre au point de l'image le plus proche du barycentre de la tache dont le niveau de gris est inférieur au seuil. Ce point nous permet de construire le premier segment du profil en croix image, en l'occurrence celui qui correspond au segment  $[P_1P_2]$ . Les trois segments restant du profil en croix image sont déterminés de la même manière décrite dans la section précédente.

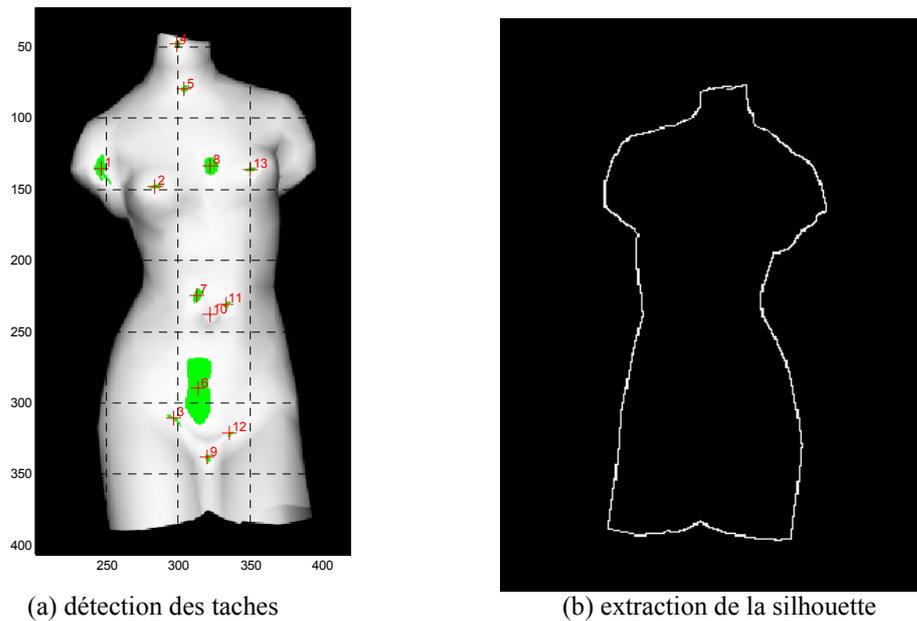


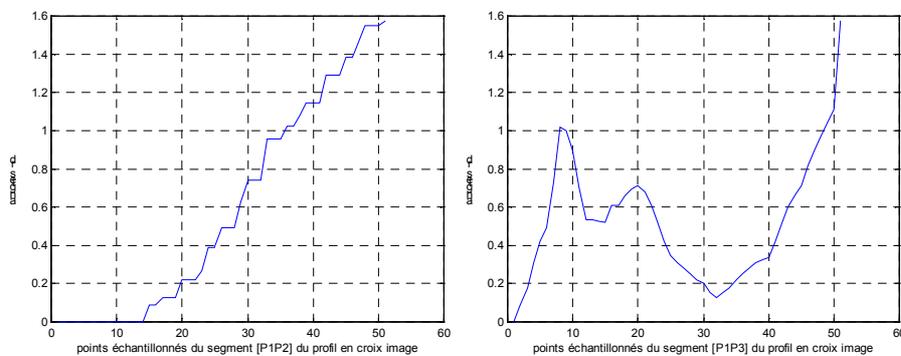
Figure 3.20 : Résultats de segmentation de l'image

Pour chaque barycentre de tache, nous calculons, d'une part, la répartition angulaire suivant le profil en croix qui lui est associé, et d'autre part, le rapport  $R$  défini par l'équation (3.10). La valeur de l'angle entre la normale au barycentre et la normale à un point du profil est donnée par inversion du modèle de réflexion diffuse:

$$\theta = \arccos\left(\frac{c}{c_m}\right) \quad (3.12)$$

où  $c_m$  est le niveau de gris maximal dans l'image et  $c$  le niveau de gris du point du profil en croix considéré.

L'échantillonnage du profil en croix image est effectué de manière similaire à celle décrite pour le profil en croix du modèle. La figure (3.21) illustre la répartition angulaire obtenue pour la tache N° 1 (épaule droite).



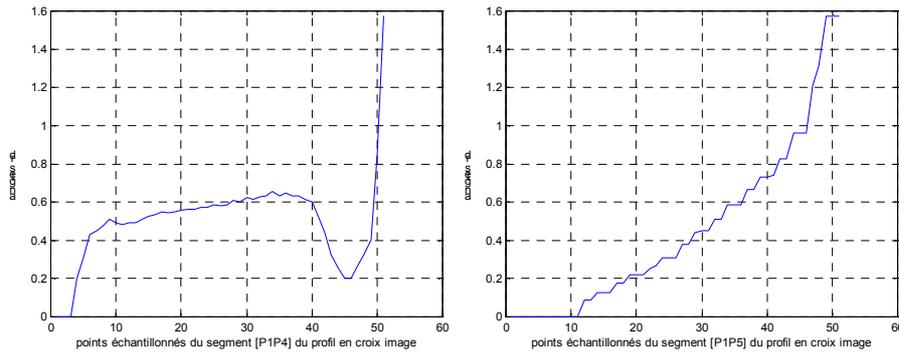


Figure 3.21 : Répartition angulaire de la tache N° 1

Arrivé à ce stade du traitement, nous pouvons commencer la procédure d'appariement proprement dite entre l'image et la base de données. Celle-ci est réalisée en deux phases : la première phase permet de reconnaître l'aspect photométrique dans lequel se présente le modèle 3D dans l'image alors que la deuxième phase permet de mettre en correspondance les barycentres des taches avec les sommets du maillage.

### Etape 1: Reconnaissance de l'aspect photométrique

Nous comparons la répartition angulaire et le rapport  $R$  de chaque barycentre de tache avec les répartitions angulaires et les rapports  $R$  des sommets du maillage stockés dans la table de sommets. Pour cela, nous fixons deux seuils d'erreurs,  $E_a$  pour la répartition angulaire et  $E_r$  pour le rapport  $R$ . Nous retenons les sommets pour lesquels les écarts obtenus sont inférieurs aux deux seuils d'erreurs. Ces critères ( $E_a$ ,  $E_r$ ) sont très importants car ils permettent de sélectionner les sommets du maillage qui correspondent le mieux aux taches de l'image tout en limitant leur nombre, ce qui, d'une part, donne à notre algorithme une robustesse contre les bruits et les erreurs de traitements, et d'autre part, diminue énormément le temps d'appariement. Les valeurs de ces critères ont été déterminées expérimentalement après plusieurs tests (voir § 3.5.3), elles constituent un compromis entre la qualité et le temps de reconnaissance pour avoir un système le plus performant possible. L'algorithme développé pour réaliser cette tâche est le suivant:

#### Procédure *Aspects\_retenus*

**Entrée:** *Sommet.Aspects*( $M \times 1$ ), *Sommet.Distrib*( $M \times 4 \times P$ ), *Tache.Disrib*( $L \times 4 \times P$ ), *Sommet.Ratio*( $M \times 1$ ), *Tache.Ratio*( $L \times 1$ ).

$M$  : le nombre de sommets.

$P$  : le nombre d'échantillons par segment du profil en croix.

$L$  : le nombre de taches.

*Sommet.Aspects*( $M \times 1$ ): les listes des aspects photométriques du modèle auxquels appartient chaque sommets du maillage.

*Sommet.Distrib*( $M \times 4 \times P$ ) : les répartitions angulaires des sommets du maillage.

*Tache.Distrib*( $L \times 4 \times P$ ) : les répartitions angulaires des barycentres des taches.

*Sommet.Ratio*( $M \times 1$ ) : les rapports  $R$  des  $M$  sommets du maillage.

*Tache.Ratio*( $L \times 1$ ) : les rapports  $R$  des barycentres des  $L$  taches extraites de l'images.

**Sortie:** l'ensemble des aspects possibles, noté  $P_{set}$

### Début

1.  $P_{set} \leftarrow \emptyset$
2. **pour** chaque tache  $i$  **faire**
3.     **pour** chaque sommet du maillage  $j$  **faire**
4.          $Err\_dist(j) \leftarrow Sommet.Distrib(j \times 4 \times P) - Tache.Distrib(i \times 4 \times P)$
5.          $Err\_rat(j) \leftarrow Sommet.Ratio(j \times 1) - Tache.Ratio(i \times 1)$
6.         **si** ( $|Err\_dist(j)| \leq E_a$ ) **et** ( $|Err\_rat(j)| \leq E_r$ )
7.             **alors**  $P_{set} \leftarrow P_{set} \cup Sommet.Aspects(j)$
8.         **fin**
9.     **fin**pour
10. **fin**pour

### Fin

Lorsque nous traçons les normales de référence des aspects photométriques associés aux sommets retenus et stockées dans la table d'aspects photométriques nous remarquons que celles-ci s'organisent suivant des cônes (cf. figure 3.22). Ceci s'explique par le fait que chaque sommet retenu appartient généralement à plusieurs aspects photométriques adjacents à cause de la discrétisation de la sphère d'observation. Cette adjacence se manifeste par des normales de référence qui sont orientées presque dans la même direction (à un angle près). Ainsi nous pouvons dire que la direction de chaque cône représente une hypothèse de reconnaissance de l'aspect photométrique du modèle 3D. De plus, le cône qui contient le plus de normales de référence représente l'hypothèse la plus probable. En effet, cette caractéristique est très intéressante, car non seulement elle permet à chaque fois de regrouper des hypothèses semblables en une seule hypothèse ce qui réduit énormément le nombre total d'hypothèses générées, mais elle classe implicitement aussi, ces hypothèses de la plus probable à la moins probable.

L'algorithme qui permet, à partir des normales de référence retenues, de séparer les différents cônes est le suivant:

### Procédure *Separation\_Distributions\_Coniques*

**Entrée:** ensemble  $P_{set}$

**Sortie:** les distributions coniques  $C_i$

### Début

1.  $i \leftarrow 1$
2.  $temp \leftarrow P_{set}$
3. **tant que**  $temp \neq \emptyset$  **faire**
4.      $k \leftarrow 1$
5.      $C_i \leftarrow \{temp(1)\}$

```

6.   pour chaque élément  $k$  de  $C_i$  faire
7.     pour chaque élément  $j$  de  $temp \neq C_i(k)$  faire
8.       if  $\langle C_i(k), temp(j) \rangle \leq \theta_{seuil}$  then
9.          $C_i \leftarrow C_i \cup \{temp(j)\}$ 
10.         $k \leftarrow k+1$ 
11.     finsi
12.   finpour
13.    $temp \leftarrow temp - C_i$ 
14. finpour
15.    $i \leftarrow i+1$ 
16. fintant que
Fin

```

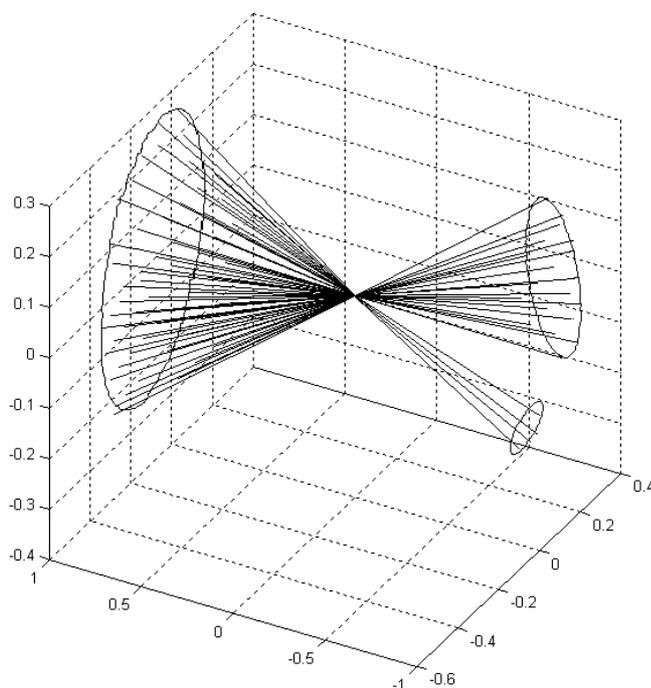


Figure 3.22 : Normales de références des aspects retenus.

La figure (3.22) montre que dans notre exemple, les normales de référence retenues se répartissent selon trois cônes, ce qui génère trois hypothèses de reconnaissance d'aspects photométriques. De plus nous remarquons qu'un des trois cônes possèdent plus de vecteurs que les deux autres ce qui implique que l'hypothèse de reconnaissance qui lui est associée est la plus probable.

Pour déterminer l'aspect photométrique représentatif de chaque hypothèse, nous calculons tout d'abord la direction de chaque cône ainsi généré. Cette dernière est donnée à chaque fois, par le vecteur moyen des normales de référence appartenant au cône considéré.

Ensuite, nous déterminons pour chaque cône la normale de référence qui fait un angle minimal avec le vecteur moyen, l'aspect représentatif correspondra donc à l'aspect photométrique associée à cette normale de référence.

Ainsi, pour notre exemple, nous obtenons les hypothèses suivantes (classées de la plus probable à la moins probable):

N° de l'Aspect	Nombre de population	Orientation (degrés)	
		Elevation ( $\theta$ )	Azimet( $\varphi$ )
77	53	86.3447	2.2689
720	19	86.3619	196.7722
802	4	101.4078	192.3777

Tableau 3.2 : Hypothèses générées

Pour chaque hypothèse générée, la première colonne de ce tableau donne le numéro de son aspect photométrique représentatif, la deuxième colonne nous informe sur le nombre de sa population, et enfin la troisième colonne donne l'orientation de sa normale de référence. L'aspect numéro 77 est donc l'aspect qui correspond à l'hypothèse la plus probable, la figure (3.23) représente la Vénus dans cet aspect, on voit bien que les zones mises en évidence dans le maillage sont disposées de la même manière que les taches sur l'image de luminance (figure 3.20). Ce résultat sera confirmé par la suite grâce au module de vérification d'hypothèses.

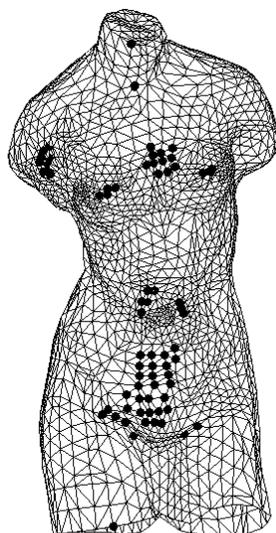


Figure 3.23 : Représentation de la Vénus dans l'aspect 77

Toutefois, il n'est pas exclus que dans certains cas de symétrie notamment, l'hypothèse la plus probable ne soit pas la bonne [Ababsa et al.,2002b]. Ceci dit, la méthode proposée

arrive toujours à déterminer la meilleure hypothèse parmi celles générées grâce au module de vérification d'hypothèses (voir §. 3.4.2.3).

## Etape 2 : reconnaissance des taches

Arrivé à ce stade, nous nous retrouvons avec  $m$  hypothèses générées pour la reconnaissance de l'aspect photométrique qui sont classées de la plus probable à la moins probable. L'étape suivante consiste donc à considérer l'hypothèse la plus probable, puis appairer les sommets qui la constituent avec les barycentres des taches. Une fois l'appariement effectué, nous vérifions cette hypothèse par re-projection. Si la vérification est bonne nous retenons l'hypothèse, sinon nous considérons la deuxième hypothèse et nous recommençons la même démarche, et ainsi de suite.

Il est clair que l'étape de vérification fait appel au module de localisation qui permet, à partir des appariements trouvés, de déterminer la transformation rigide correspondante.

Pour une hypothèse donnée, l'appariement taches/sommets est déterminé en prenant à chaque fois, le sommet de l'aspect représentatif qui minimise l'erreur quadratique moyenne entre sa répartition angulaire et celle du barycentre de la tache, définie par:

$$E = \frac{1}{P} \sum_{i=1}^P \|f_b(i) - f_s(i)\|^2 \quad (3.13)$$

où  $f_b(i)$  est la répartition angulaire du barycentre, et  $f_s(i)$  est celle du sommet de l'aspect.  $P$  étant le nombre d'échantillons utilisés dans le profil en croix.

L'identification de l'aspect photométrique de l'objet ainsi que la mise en correspondance des barycentres des taches de l'image avec des sommets du maillage constituent le résultat de l'appariement automatique 2D/3D recherché. Ces informations seront utilisées par l'algorithme de recalage afin de déterminer la position courante de l'objet dans la scène.

## 3.4 Recalage 3D et vérification d'hypothèses

L'objectif du recalage 3D est de déterminer la transformation géométrique rigide (rotation + translation) permettant d'amener l'objet de sa position initiale à sa position courante dans le repère de la scène. De plus, le recalage suppose la connaissance d'un nombre suffisant d'appariements entre les primitives de la scène ainsi que du modèle 3D associé à l'objet (voir § 2.7).

La méthode développée ci-dessus nous permet d'appairer des points 2D de l'image (les barycentres des taches) avec des points 3D du modèle (sommets du maillage). Nous allons donc formuler le problème de recalage 3D de telle manière à ce que la transformation recherchée soit exprimée en fonction de ces données.

### 3.4.1 Définition du problème

Soit un ensemble de points de référence 3D non colinéaires  $p_i=(x_i, y_i, z_i)$ ,  $i=1, \dots, n$   $n \geq 3$  exprimés dans le référentiel propre de l'objet ( $R_O$ ), les points 3D qui leur correspondent dans le référentiel de la caméra  $q_i=(x'_i, y'_i, z'_i)$  sont définis par une transformation rigide:

$$q_i = R \cdot p_i + T \quad (3.14)$$

où

$$R = \begin{pmatrix} r'_1 \\ r'_2 \\ r'_3 \end{pmatrix} \text{ tel que } R^t R = I \quad \text{et} \quad T = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \quad (3.15)$$

sont la matrice de rotation et le vecteur de translation, respectivement. Dans notre cas, les points  $p_i$  correspondent au sommets du maillage de l'aspect photométrique retenu. La condition  $R^t R = I$  est la condition d'orthonormalité de la matrice de rotation  $R$ .

La méthode de recalage 3D que nous avons élaborée fait intervenir trois repères principaux (cf. figure 3.24) :

- ( $R_C$ ) le repère lié à la caméra.
- ( $R_O$ ) le repère lié à l'objet représentant la position courante. Dans notre cas ( $R_O$ ) correspond au repère centré sur la Vénus.
- ( $R_M$ ) le repère monde (de la scène).

Le référentiel de la caméra est choisi de telle sorte que le centre de projection de la caméra soit à l'origine et que l'axe optique soit orienté dans la direction des  $z$  positifs. Les points de référence  $p_i$  sont projetés dans le plan  $z=1$ , appelé plan image normalisé, et décrits dans le repère caméra<sup>3</sup>. Soit  $m_i=(u_i, v_i, 1)^t$  les points image qui correspondent aux projections de  $p_i$  dans le plan image normalisé (cf. figure 3.24). Dans notre cas, les points  $m_i$  correspondent aux barycentres des taches appariés avec les point  $p_i$ .

Sous l'hypothèse du modèle pinhole idéal, les points  $m_i$ ,  $q_i$  et le centre de projection sont colinéaires. Cette propriété est exprimée par l'équation suivante:

$$u_i = \frac{r'_1 p_i + t_x}{r'_2 p_i + t_z} \quad (3.16)$$

$$v_i = \frac{r'_1 p_i + t_y}{r'_2 p_i + t_z} \quad (3.17)$$

ou bien

$$m_i = \frac{1}{r'_3 p_i + t_z} (R \cdot p_i + T) \quad (3.18)$$

<sup>3</sup> Nous supposons que la caméra a été préalablement calibrée, et de ce fait, ses paramètres intrinsèques et extrinsèques sont connus.

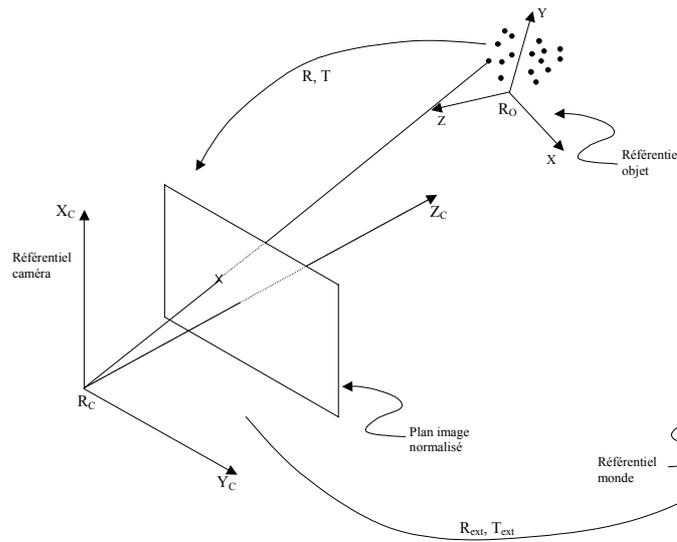


Figure 3.24 : Les différents repères mis en jeu dans le problème de recalage 3D

l'équation (3.18) est appelée "équation de colinéarité". Une autre manière d'exprimer cette colinéarité est d'utiliser le fait que la projection orthogonale de  $q_i$  sur la ligne de vue définie par le point  $m_i$  doit être égale au point  $q_i$  lui-même. Ce fait est exprimé par l'équation :

$$R \cdot p_i + T = V_i \cdot (R \cdot p_i + T) \quad (3.19)$$

où

$$V_i = \frac{m_i \cdot m_i^t}{m_i^t \cdot m_i} \quad (3.20)$$

représente la matrice de projection de la ligne de vue qui, lorsqu'elle est appliquée à un point de la scène, le projette orthogonalement sur la ligne de vue définie par le point image  $m_i$ . Ainsi, l'équation (3.18) représente l'équation de colinéarité dans l'espace image tandis que l'équation (3.19) représente l'équation de colinéarité mais dans l'espace objet. Le recalage 3D consiste à développer une méthode qui permet de trouver la transformation rigide  $(R, T)$  minimisant une certaine forme d'accumulation d'erreurs (tel que la sommation des carrés des erreurs) de l'une des deux équations de colinéarité (cf. figure 3.25).

Dans les méthodes classiques de recalage 3D, la fonction objective à minimiser est donnée par:

$$Q = \sum_{i=1}^n \left[ \left( \hat{u}_i - \frac{r_1^t \cdot p_i + t_x}{r_3^t \cdot p_i + t_z} \right)^2 + \left( \hat{v}_i - \frac{r_2^t \cdot p_i + t_y}{r_3^t \cdot p_i + t_z} \right)^2 \right] \quad (3.21)$$

où  $\hat{m}_i = (\hat{u}_i, \hat{v}_i, 1)^t$  sont les points image observés. La matrice de rotation  $R$  est généralement représentée par les angles d'Euler. Notons au passage que la minimisation dans ce cas est basée sur la colinéarité dans l'espace image. Deux méthodes sont généralement utilisées

pour résoudre ce type de problème, en l'occurrence la méthode de Gauss-Newton [Haralick et Shapiro, 1993] et la méthode de Levenberg-Marquardt [Lowe, 1991].

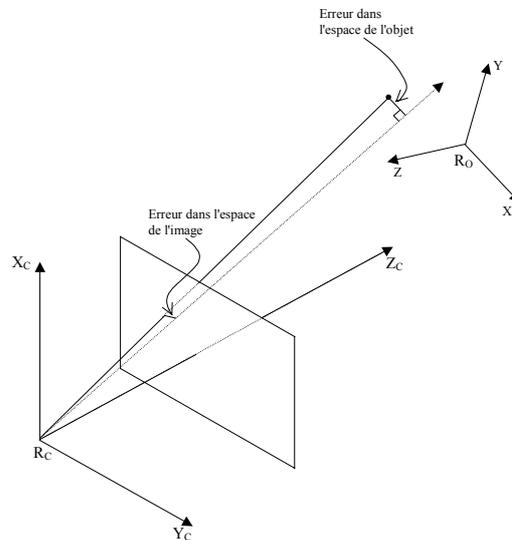


Figure 3.25 : Les erreurs de colinéarité dans l'espace image et l'espace objet

En ce qui nous concerne, nous avons adopté la méthode élaborée par [Lu et Hager, 2000] qu'ils ont appelée algorithme d'itération orthogonale ("*Orthogonal Iteration -OI-algorithm*"). A l'inverse des méthodes classiques qui sont utilisées pour résoudre les problèmes d'optimisation d'une manière générale, l'algorithme d'itération orthogonale exploite judicieusement la structure spécifique du problème de recalage 3D. De plus, il donne des résultats très précis et converge globalement d'une manière assez rapide ce qui fait de lui un outil très intéressant pour les applications temps réel. Cet algorithme est détaillé dans la section suivante.

### 3.4.2 Algorithme d'itération orthogonale (IO)

Pour estimer la pose de l'objet, cet algorithme utilise une fonction d'erreur appropriée définie dans l'espace de l'objet qu'il réécrit sous une forme particulière afin qu'elle admette une itération basée sur la solution du problème classique de l'estimation de la pose 3D/3D appelé problème de l'orientation absolue ("*absolute orientation problem*"). Comme l'algorithme de l'IO dépend entièrement de l'orientation absolue, nous allons tout d'abord décrire ce problème ainsi que la solution proposée pour le résoudre.

#### 3.4.2.1 Problème de l'orientation absolue

Le problème de l'orientation absolue peut être posé comme suit : Supposons que les coordonnées 3D des points  $q_i$  définis dans l'espace caméra puissent être reconstruits

physiquement (par exemple, par un scanner) ou par calcul (en utilisant la stéréovision), alors pour chaque point observé, nous pouvons écrire:

$$q_i = R \cdot p_i + T \quad (3.22)$$

Le calcul de l'orientation absolue revient à déterminer  $R$  et  $T$  à partir des paires d'appariement  $(q_i, p_i)$ . Avec au moins 3 points de référence non colinéaires,  $R$  et  $T$  peuvent être obtenues comme une solution du problème des moindres carrés suivant :

$$\min_{R,T} \sum_{i=1}^n \|R \cdot p_i + T - q_i\|^2 \quad \text{tel que } R \cdot R^t = I \quad (3.23)$$

Ce problème des moindres carrés avec contrainte (la contrainte d'orthogonalité de la matrice de rotation) peut être résolu en utilisant les quaternions [Horn, 1987] ou bien la décomposition en valeurs singulières (SVD) [Arun et al., 1987]. Nous allons détailler ci-dessous, la solution par SVD que nous allons utiliser par la suite dans l'algorithme IO.

Soient  $\{p_i\}$  et  $\{q_i\}$  les deux ensembles de points mis en correspondance, et soit:

$$\bar{p} = \frac{1}{n} \sum_{i=1}^n p_i \quad \bar{q} = \frac{1}{n} \sum_{i=1}^n q_i \quad (3.24)$$

leurs centres de gravité respectivement. On définit alors les points:

$$p'_i = p_i - \bar{p} \quad q'_i = q_i - \bar{q} \quad (3.25)$$

et la matrice

$$M = \sum_{i=1}^n q'_i p_i'^t \quad (3.26)$$

d'inter covariance entre les points  $\{p_i\}$  et les points  $\{q_i\}$ . On démontre que [Horn et al., 1988] si  $R^*$  et  $T^*$  minimisent l'équation (3.23) alors ils satisferont:

$$R^* = \arg \max_R \text{trace}(R^t M) \quad (3.27)$$

$$t^* = \bar{q} - R^* \cdot \bar{p} \quad (3.28)$$

Soit  $(U, \Sigma, V)$  la décomposition en valeurs singulières de la matrice  $M$ , tel que  $U^t M V = \Sigma$ . Alors, la solution de l'équation (3.23) est donnée par:

$$R^* = V U^t \quad (3.29)$$

Notons que la translation optimale est entièrement définie par la rotation optimale et que les informations nécessaires pour trouver cette rotation sont contenues dans la matrice  $M$  définie par l'équation (3.26). Ainsi, seule la position relative des points 3D par rapport à leurs centres de gravité est significative dans la détermination de la matrice de rotation optimale.

### 3.4.2.2 Algorithme

L'algorithme IO traite le problème du recalage 3D en utilisant le vecteur erreur de colinéarité défini dans l'espace de l'objet (cf. figure 3.24):

$$e_i = (I - \hat{V}_i)(R.p_i + T) \quad (3.30)$$

où  $\hat{V}_i$  est la matrice de projection de la ligne de vue observée correspondant au point  $\hat{m}_i$ , elle est définie par:

$$\hat{V}_i = \frac{\hat{m}_i \cdot \hat{m}_i^t}{\hat{m}_i^t \cdot \hat{m}_i} \quad (3.31)$$

Nous cherchons à déterminer  $R$  et  $T$  qui minimisent la somme des erreurs quadratiques suivantes:

$$E(R, T) = \sum_{i=1}^n \|e_i\|^2 = \sum_{i=1}^n \|(I - \hat{V}_i)(R.p_i + T)\|^2 \quad (3.32)$$

Notons que toutes les informations fournies par les points image observés  $\{\hat{m}_i\}$  sont incluses dans l'ensemble des matrices de projection  $\{\hat{V}_i\}$ . Puisque la fonction objective  $E$  est quadratique en  $T$ , pour une rotation  $R$  donnée, la valeur optimale de  $T$  peut être calculée d'une manière approchée par :

$$T(R) = \frac{1}{n} \left( I - \frac{1}{n} \sum_{j=1}^n \hat{V}_j \right)^{-1} \sum_{j=1}^n (\hat{V}_j - I) R.p_j \quad (3.33)$$

Ayant la translation optimale en fonction de  $R$  et en définissant:

$$q_i(R) = \hat{V}_i(R.p_i + T(R)) \quad \text{et} \quad \bar{q}(R) = \frac{1}{n} \sum_{i=1}^n q_i(R) \quad (3.34)$$

l'équation (3.32) peut être réécrite comme suit:

$$E(R) = \sum_{i=1}^n \|R.p_i + T(R) - q_i(R)\|^2 \quad (3.35)$$

cette équation ressemble à l'équation (3.23) qui définit le problème de l'orientation absolue. Malheureusement, dans ce cas on ne peut pas déterminer la matrice  $R$  directement à partir d'une simple estimation de la matrice d'inter-covariance entre  $\{p_i\}$  et  $\{q_i(R)\}$ , car la matrice:

$$M(R) = \sum_{i=1}^n q_i'(R) \cdot p_i^t \quad \text{où} \quad p_i' = p_i - \bar{p} \quad \text{et} \quad q_i'(R) = q_i(R) - \bar{q}(R) \quad (3.36)$$

dépend de  $R$ .

Cependant, la matrice  $R$  peut être calculée itérativement comme suit : premièrement, nous supposons que l'estimée d'ordre  $k$  de la matrice  $R$  est  $R^{(k)}$ ,  $T^{(k)} = T(R^{(k)})$ , et  $q_i^{(k)} = R^{(k)} \cdot p_i + T^{(k)}$ .

L'estimée suivante,  $R^{(k+1)}$ , est déterminée en résolvant le problème de l'orientation absolue suivant:

$$R^{(k+1)} = \arg \min_R \sum_{i=1}^n \|R \cdot p_i + T - \hat{V}_i \cdot q_i^{(k)}\|^2 = \arg \max_R \text{tr}(R^t M(R^{(k)})) \quad (3.37)$$

où l'ensemble de points  $\hat{V}_i q_i^{(k)}$  est considéré comme une hypothèse de l'ensemble de points  $q_i$  de la scène dans l'équation (3.23). Sous cette forme, la solution pour  $R^{(k+1)}$  est donnée par l'équation (3.29). Nous calculons ensuite l'estimée suivante ( $k+1$ ) de la translation en utilisant l'équation (3.33), tel que:

$$T^{(k+1)} = T(R^{(k+1)}) \quad (3.38)$$

et nous répétons le processus. La solution optimale  $R^*$  est obtenue lorsque l'équation (3.37) reste inchangée, en d'autres termes, lorsque  $R^*$  satisfait:

$$R^* = \arg \min_R \sum_{i=1}^n \|R \cdot p_i + T - \hat{V}_i (R^* \cdot p_i + T(R^*))\|^2 \quad (3.39)$$

L'avantage de cet algorithme est qu'il converge globalement vers l'optimum de l'équation (3.37) pour n'importe quel ensemble de points observés et pour tout point de départ  $R^{(0)}$ . La seule contrainte sur  $R^{(0)}$  est qu'elle ne doit pas placer l'objet observé derrière la caméra (voir annexe B).

Finalement, la transformation rigide qu'a subi l'objet pour passer de sa position initiale à sa position courante dans le référentiel monde est donnée par:

$$A = A_{ext}^{-1} \cdot A_{opt} \quad (3.40)$$

où

$$A_{ext} = \begin{pmatrix} R_{ext} & T_{ext} \\ 0 & 1 \end{pmatrix} \quad \text{et} \quad A_{opt} = \begin{pmatrix} R^* & T^* \\ 0 & 1 \end{pmatrix} \quad (3.41)$$

La matrice  $A_{ext}$  représente les paramètres extrinsèques de la caméra qui correspondent à la transformation rigide entre le repère monde et le repère caméra.

En pratique, pour minimiser le temps de reconnaissance des taches, nous avons choisi d'arrêter ce processus dès qu'il arrive à apparier 4 taches avec 4 sommets de l'hypothèse en cours. Le reste des taches sera utilisé, d'une part, pour la vérification de cette hypothèse, et d'autre part, pour affiner la transformation rigide trouvée.

Pour l'exemple considéré, l'algorithme de reconnaissance de taches a pu générer les 4 appariements nécessaires pour déterminer la transformation rigide entre les barycentres des taches et les sommets de l'aspect photométrique le plus probable, en l'occurrence l'aspect N° 77. La figure (3.26) illustre, à titre d'exemple, deux résultats de la reconnaissance de tache, en l'occurrence, la mise en correspondance de la tache N°5 (voir figure 3.20 page 19) avec le sommet du maillage N° 512 et de la tache N°7 avec le sommet N°1882. On voit bien que, pour chaque cas d'appariement, les profil en croix des taches et des sommets sont similaires.

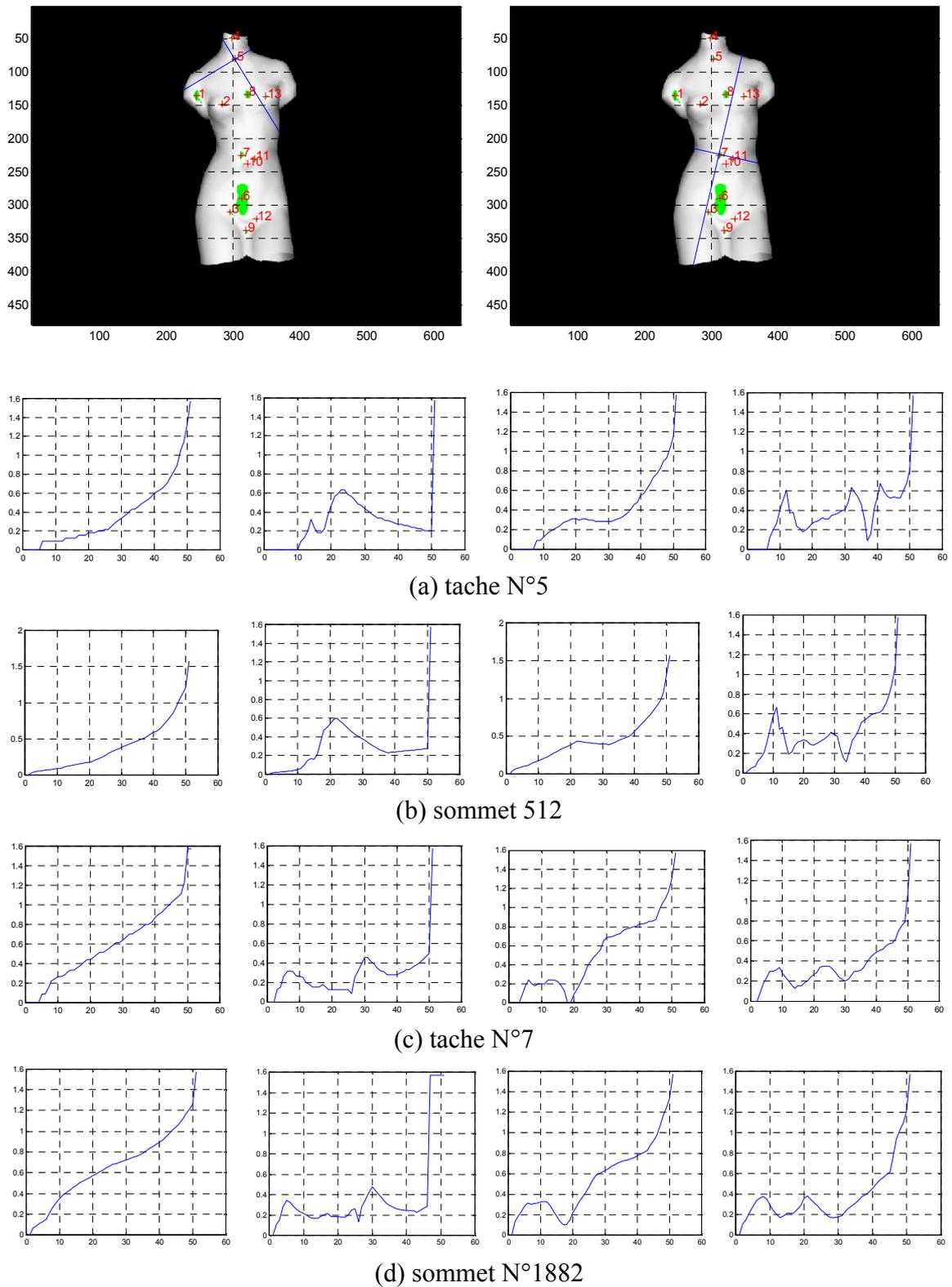


Figure 3.26 : Les profils en croix des taches N°5 et N°7 et des sommets de maillage N°512 et N°1882 qui leur correspondent

La transformation rigide trouvée est la suivante:

Vecteur de translation (m) $T$	-0.2554	0.1084	-1.3427
Matrice de rotation $R$	0.9995	0.0214	-0.0214
	0.0214	0.0012	0.9998
	0.0214	-0.9998	0.0007

Tableau 3.3 : Transformation optimale

La figure (3.27a) montre la projection du maillage dans l'image de luminance après qu'on lui ait appliqué la transformation optimale trouvée, on remarque que le modèle est recalé dans l'ensemble sur son image, toutefois on constate une erreur de recalage qui est due essentiellement:

1. à la nature même de l'image de luminance, en effet les profils de croix des barycentres de taches sont calculés à partir des niveaux de gris de l'image, et comme cette information est discrète, ceci se manifeste par le fait que les points qui se trouvent dans un voisinage proche d'un barycentre donné possèdent la même répartition angulaire. Ainsi, le sommet apparié avec le barycentre de la tache peut en réalité correspondre à un autre point de l'image se trouvant dans le voisinage de ce barycentre.
2. au nombre d'appariements considérés pour calculer la transformation rigide. Nous avons considéré seulement 4 appariements pour déterminer cette transformation rigide. Ceci se répercute sur la qualité de l'estimation de la transformation optimale.

### 3.4.2.3 Vérification d'hypothèses et recalage fin

La re-projection du modèle dans l'image en utilisant la transformation optimale trouvée nous permet de vérifier la validité de l'hypothèse en cours. Pour cela nous vérifions que chaque barycentre de tache qui n'a pas été utilisé dans l'estimation de la transformation rigide coïncide bien avec au moins un sommet parmi les sommets non appariés de l'aspect photométrique. Un barycentre est identifié si la distance entre lui et un sommet projeté dans l'image est inférieur à un seuil donné. Si plusieurs sommets vérifient cette conditions nous considérons alors le sommet le plus proche. L'hypothèse en cours sera retenue si tous les barycentres des taches ont été vérifiés.

D'un autre côté, pour affiner la transformations rigide, nous injectons les nouveaux appariements taches/sommets générés par le module de vérification dans le module de recalage 3D et nous réestimons la transformation rigide.

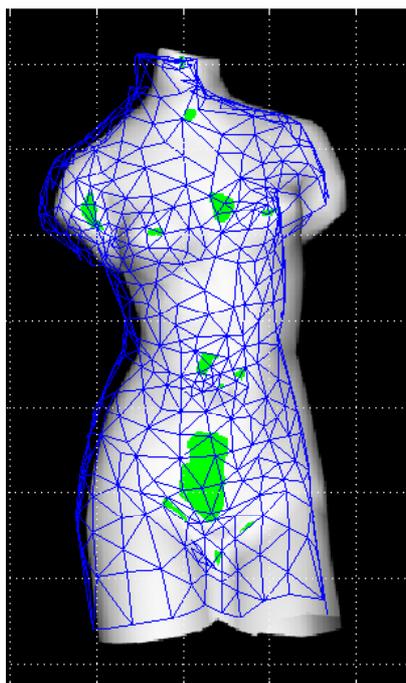
Dans notre exemple, la vérification d'hypothèses permet de confirmer l'hypothèse de reconnaissance de l'aspect N°77. La transformation rigide finale estimée est:

Vecteur de translation $T$ (m)	0.00471	0.0202	-1.4561
Matrice de rotation $R$	0.9997	0.0174	-0.0173
	0.0174	-0.0062	0.9998
	0.0173	-0.9998	0.0065

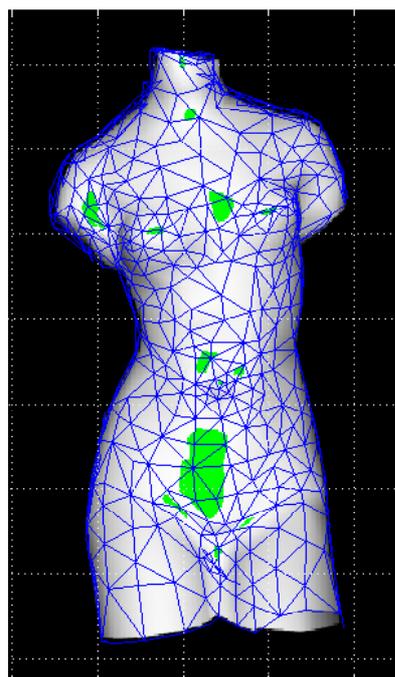
Tableau 3.4 : Transformation optimale affinée

La figure (3.27b) illustre la re-projection du maillage dans le plan image en utilisant cette fois-ci la transformation affinée, nous remarquons que le modèle projeté se superpose mieux sur l'image de la "Vénus", ce qui implique une meilleure qualité du recalage 3D.

Une méthode pour mesurer la qualité du recalage 3D est présentée dans le paragraphe (3.5.5), elle permet de calculer l'erreur entre la transformation estimée et la transformation attendue, pour les deux méthodes de recalage. Ainsi, pour l'exemple que nous traitons ici, nous nous référons à la première ligne du tableau (3.5), page 117, nous remarquons que l'erreur due au recalage fin est nettement inférieure à celle due au premier recalage.



(a) re-projection du maillage en utilisant la transformation optimale



(b) re-projection du maillage en utilisant la transformation affinée

Figure 3.27 : Re-projection du maillage dans le plan image

### 3.5 Evaluation des performances

Dans les paragraphes précédents, nous avons présenté le principe de l'approche photoclinométrique ainsi que les différents algorithmes élaborés pour son implémentation réelle. Les résultats ont été présentés sur une scène représentant l'objet dans une pose quelconque. Cependant, il paraît difficile de conclure à partir de ces résultats, que le système proposé est entièrement satisfaisant. Ainsi, le but de cette section est de tenter d'évaluer notre approche de façon à dégager ses points forts ses faiblesses et ses limites.

L'évaluation des performances d'un système de reconnaissance est une étape essentielle mais extrêmement complexe à mettre en œuvre. Cette complexité s'explique par le manque

de formalisme et de normes dans les protocoles de tests pour établir les performances d'un algorithme de reconnaissance.

Nous allons présenter dans cette section une méthode d'évaluation de notre approche photométrique en proposant quelques critères qui nous semblent importants et en les appliquant à notre système.

La première question essentielle à laquelle nous devons répondre concerne ce que l'on veut tester. En effet, le processus de recalage que nous avons élaboré est basé sur une succession d'étapes et de traitements différents mais étroitement liés, faut-il tenter d'évaluer l'ensemble de l'algorithme ou chaque phase séparément? Nous avons choisi la deuxième option, en l'occurrence de tester séparément chacune des différentes phases afin de pouvoir optimiser leurs résultats dans l'objectif final d'avoir un recalage le plus fiable possible. La méthode de test consiste à évaluer, sur un jeu d'images de référence prises de l'objet dans des poses différentes, des critères d'erreurs définis en fonction des paramètres de l'algorithme de recalage.

### 3.5.1 validité du modèle photométrique

L'hypothèse du modèle de réflexion diffuse que nous avons considérée dans notre approche est tout à fait plausible. En effet [Roussel, 1999] a démontré par une expérimentation (sur des images réelles) que des objets ayant des surfaces mates (tel que du polystyrène peint ou du papier) présentent une réponse photométrique très proche du modèle de réflexion Lambertien (cf. figure 3.28).



Figure 3.28 : comparaison des réponses photométriques pour un objet en papier tirée de [Roussel, 1999]

### 3.5.2 Images de test

L'obtention d'un ensemble d'images de référence est la première étape dans le processus d'évaluation de performance. A cause de la difficulté de mise en œuvre de l'hypothèse de lumière directionnelle (manque de directionnalité des sources de lumière utilisées en plus de l'effet indésirable de la lumière ambiante), nous n'avons pas pu générer des images réelles satisfaisant le modèle de réflexion diffuse, ceci nous a amené à utiliser des images de synthèse qui satisfont les hypothèses énoncés afin de tester l'approche élaborée. Malgré

leur caractère idéal, les images de synthèse sont intéressantes car elle permettent d'extraire les informations pertinentes sans aucune erreur et ainsi donnent une référence idéale permettant d'évaluer les différents critères.

Enfin, le dernier point à préciser concerne le nombre d'images de référence à considérer pour permettre une évaluation correcte. [Salotti et al., 1996] estiment que deux images permettent des tests significatifs. En ce qui nous concerne, nous avons choisi de considérer cinq images de la Vénus prises dans des positions différentes (cf. figure 3.29) afin de mener nos tests de validation.

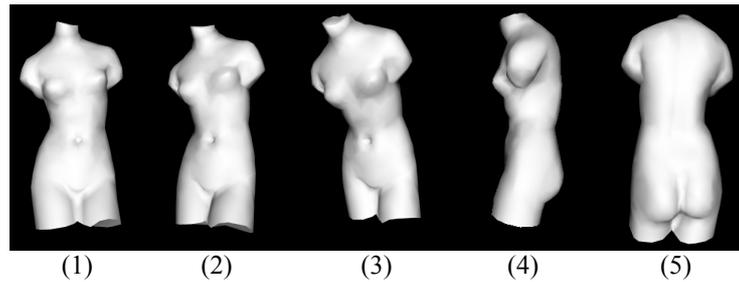


Figure 3.29 : Images de référence

### 3.5.3 Base de données

La construction de la base de données est une étape primordiale, elle permet d'encoder les informations pertinentes du modèle afin de réaliser des appariements image/modèle pour le recalage 3D final. De nombreux paramètres dans cette étape sont à optimiser. Ainsi, on distingue :

- 1) le nombre d'aspects photométriques représentatifs de la Vénus. En effet, les aspects doivent être, d'une part, discriminants pour minimiser le risque d'erreurs, et d'autre part non redondants afin de réduire le temps de reconnaissance. Nous avons donc discrétisé la sphère d'observation de telle manière à satisfaire ces deux exigences. Concrètement, nous avons constaté que les zones du maillage présentent le plus de similarité avec les zones de l'image qui leur correspondent lorsque l'on règle la tolérance angulaire pour les normales de référence des aspects photométriques entre  $5^\circ$  et  $10^\circ$ . Ce résultat est tout à fait logique. En effet, une zone brillante dans l'image possède un niveau de gris au maximum égal à 255 (sur une échelle allant de 0 à 255, 0 pour noir et 255 pour blanc). La différence d'angle entre cette zone et une zone moins éclairée (qui fait au moins un niveau de gris de moins) est donnée par l'équation  $\Delta\theta = \arccos(254/255)$ , ce qui donne une différence d'angle d'environ  $5,08^\circ$ . En d'autres termes, tous les sommets du maillage dont les normales font un angle inférieur à  $\Delta\theta$  entre elles vont avoir le même niveau de gris dans l'image et donc appartiennent au même aspect photométrique. Nous avons, donc choisi de discrétiser la sphère d'observation de telle sorte que l'angle entre deux directions de vue adjacentes soit égal à  $5^\circ$ , ce qui correspond à une itération d'ordre 4 de la grille ISEA permettant de générer 2562 normales de référence.

- 2) Echantillonnage du profil en croix. La discrétisation du profil en croix doit, d'une part, permettre de récolter le maximum d'information sur la répartition angulaire le long de ce profil, et d'autre part, fournir un nombre d'échantillons raisonnable pour ne pas alourdir le processus de reconnaissance. Nous avons défini un critère qui permet d'estimer le nombre moyen d'échantillons à prendre sur le profil de croix pour avoir une répartition angulaire discriminante. Comme le calcul des répartitions angulaires est basé sur les niveaux de gris extraits de l'image de luminance, nous avons déterminé, le long de plusieurs profil en croix, le nombre de niveaux de gris différents en fonction du nombre d'échantillons (cf. figure 3.30), nous avons constaté qu'en moyenne à partir de 50 points par segment de profil, le nombre de niveaux de gris différents restait pratiquement le même. Ceci veut dire qu'au delà de cette limite, un suréchantillonnage n'apporterait pas un plus d'informations discriminantes.

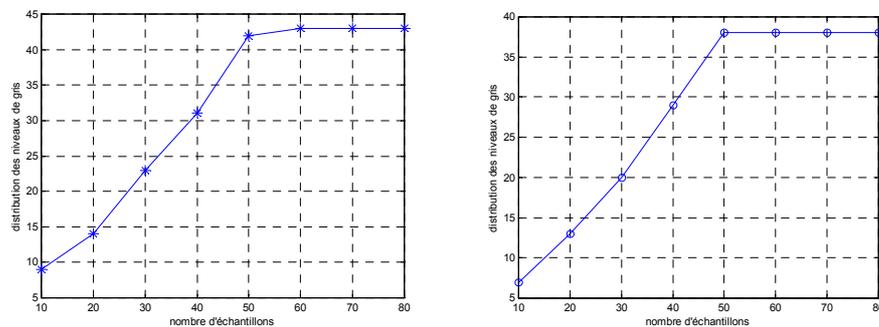


Figure 3.30 : Exemples de distributions des niveaux de gris en fonction du nombre d'échantillons pris sur deux segments d'un profil en croix

Ainsi, ces tests nous ont permis d'optimiser les paramètres essentiels qui rentre dans la construction de la base de donner et qui ont une influence directe sur le processus de reconnaissance durant la phase en ligne.

### 3.5.4 Appariement

L'appariement image/modèle est basé sur une comparaison des invariants extraits de l'image avec ceux du modèle, en l'occurrence les répartitions angulaires et les rapports  $R$  des taches et des sommets du maillage respectivement. Ainsi nous avons définis deux critères pour avoir des appariements taches/sommets les plus fiables possibles, le premier critère permet de mesurer la similarité entre les répartitions angulaires, il consiste en un seuil d'erreur tolérée que nous avons appelé  $E_a$  tel que, si la différence entre deux répartitions angulaires définie par l'erreur quadratique moyenne de l'équation (3.13) est inférieure à  $E_a$  alors on conclura à un appariement entre la tache et le sommet en question ( $E_a$  représente donc l'énergie maximale de la fonction d'erreur entre les répartitions angulaires). Nous avons estimé le seuil  $E_a$  empiriquement. Pour cela nous avons procédé de la manière suivante : nous avons considéré dix aspects photométriques, ensuite, pour chaque aspect, nous avons mis en évidence ses sommets et les taches qui leur correspondent dans l'image de luminance. Nous avons considéré plusieurs valeurs pour le seuil  $E_a$ , puis pour chacune d'elles, nous avons comptabilisé le nombre total d'appariements

exacts, les résultats obtenus sont représentés dans la figure (3.31a). On remarque que pour un seuil  $E_a \geq 10^{-3}(rd)$ , les appariements taches/sommets sont tous identifiés. D'un autre côté, nous définissons un autre critère, noté  $E_r$ , basé sur les rapports  $R$  des profils en croix pour affiner l'appariement. En effet, ce deuxième critère nous permet de ne garder que les sommets du maillage qui, d'une part, satisfont le critère  $E_a$ , et d'autre part, possèdent un rapport  $R$  très proche de celui de la tache, ce qui réduit le nombre d'hypothèses d'appariement généré. Le seuil d'erreur  $E_r$  est défini comme étant l'erreur relative maximale entre les rapports  $R$  d'un sommet et d'une tache respectivement, c'est à dire:  $(R_{\text{sommet}} - R_{\text{tache}}) / R_{\text{sommet}} \leq E_r$ . Pour estimer la valeur du seuil  $E_r$ , nous avons suivi la même méthode que précédemment. Ainsi, nous avons considéré plusieurs valeurs du seuil  $E_r$ . Ensuite, nous avons comptabilisé le nombre d'appariements exacts (connus à priori) taches/sommets qui satisfont chaque critère. Les résultats obtenus sont illustrés dans la figure (3.31b). Nous remarquons que pour un seuil  $E_r \geq 6\%$ , nous arrivons à retrouver la totalité des appariements taches/sommets pré définis.

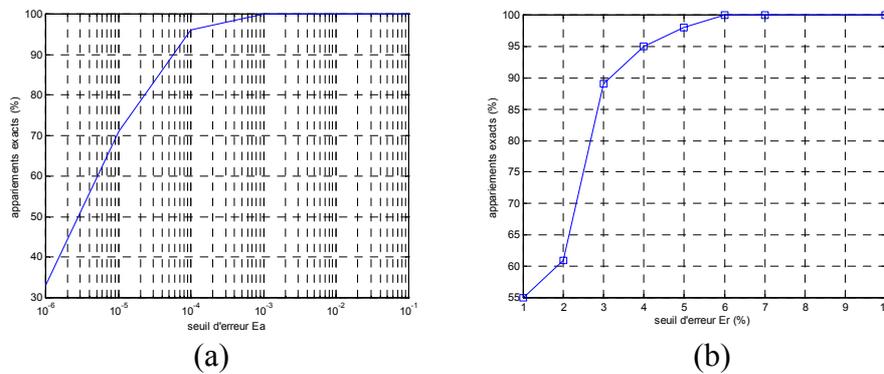


Figure 3.31 : Détermination des seuils  $E_a$  et  $E_r$

La figure (3.32) illustre les différents résultats de la mise en correspondance des images de tests avec les aspects photométriques du modèle en utilisant les paramètres trouvés ci-dessus. Nous donnons pour chaque image, d'une part, les distributions coniques qui définissent le nombre d'hypothèses d'appariement généré, et d'autre part, la représentation de l'aspect photométrique retenu.

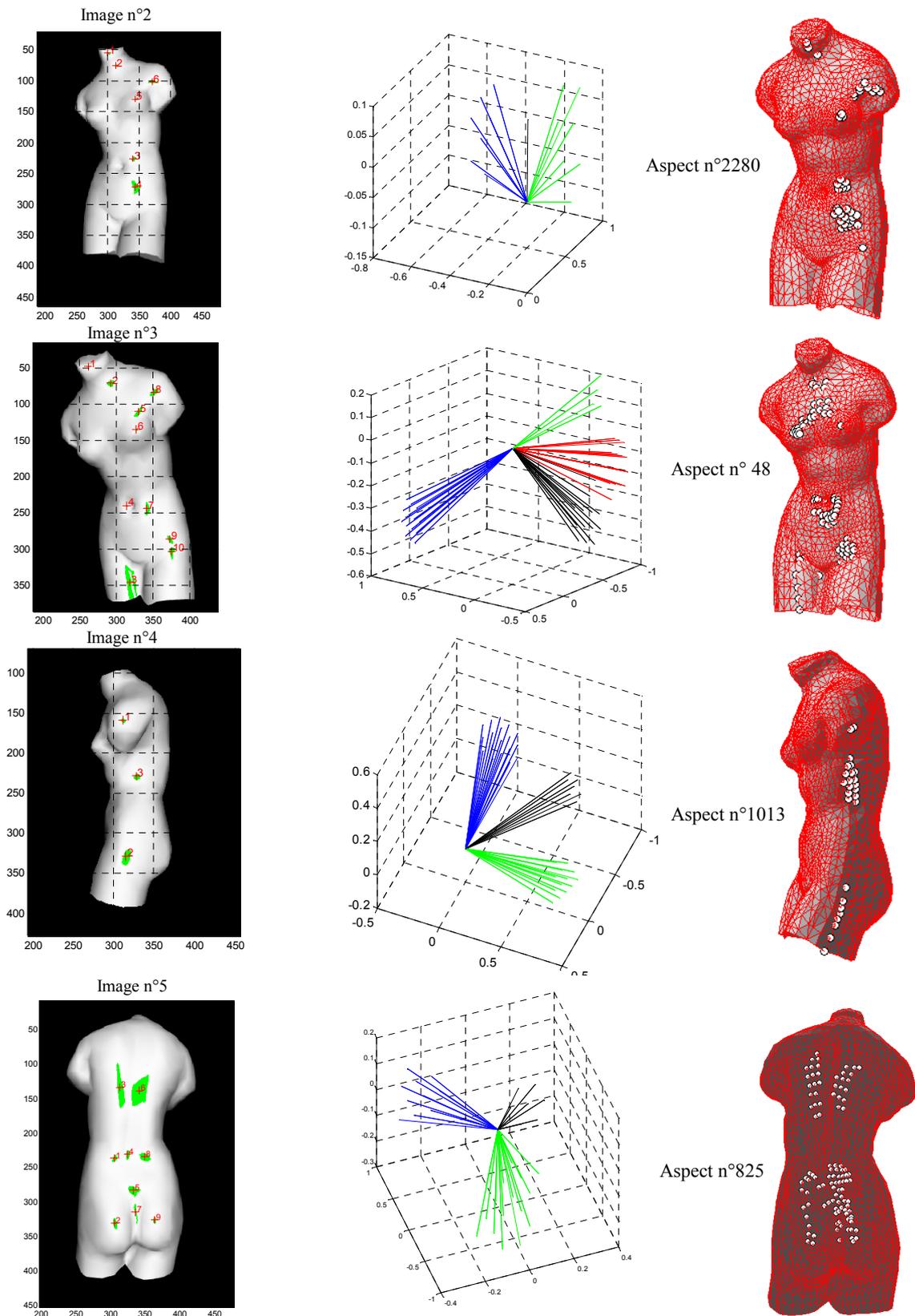


Figure 3.32 : Résultats de la mise en correspondance

### 3.5.5 Recalage 3D

Pour évaluer la méthode de recalage 3D élaborée, nous avons donc considéré cinq images de référence prises de la Vénus dans des poses connues a priori. Chaque pose de la Vénus est générée par l'application d'une transformation rigide à son modèle maillé. Les paramètres de chaque transformation sont définis de telles sorte que la Vénus paraisse dans les images de luminance dans des aspects photométriques les plus variés possibles. Les critères d'évaluation sont définis par les erreurs de reconstruction de la translation et de la rotation. Ainsi pour la translation, l'erreur est définie par la distance entre les vecteur de translation théorique et reconstruit, elle est donnée par:

$$E_T(m) = \sqrt{(t_x - t'_x)^2 + (t_y - t'_y)^2 + (t_z - t'_z)^2} \quad (3.42)$$

où  $(t_x, t_y, t_z)$  et  $(t'_x, t'_y, t'_z)$  sont les vecteurs translation théorique et reconstruite respectivement.

De même, nous définissons pour la rotation trois erreurs d'angles qui correspondent aux erreurs entre les angles d'Euler de la rotation théorique et ceux de la rotation reconstruite selon les axes  $X$ ,  $Y$  et  $Z$ , nous aurons ainsi:

$$E_{\theta_x}(\text{degrés}) = |\theta_x - \theta'_x| \quad E_{\theta_y}(\text{degrés}) = |\theta_y - \theta'_y| \quad E_{\theta_z}(\text{degrés}) = |\theta_z - \theta'_z| \quad (3.43)$$

où  $(\theta_x, \theta_y, \theta_z)$  et  $(\theta'_x, \theta'_y, \theta'_z)$  correspondent aux angles d'Euler des rotations théorique et reconstruite, respectivement.

D'un autre côté, nous définissons un critère qui permet de valider la faisabilité du recalage 3D, il s'agit du nombre de taches brillantes extraites de l'image, ce nombre doit nécessairement être supérieur ou égal à trois pour que notre algorithme de recalage 3D puisse générer une transformation rigide. Les résultats des différents calculs effectués sur les images de référence sont présentés dans le tableau (3.5).

	Nb de taches	Recalage 3D				Recalage 3D fin			
		$E_T(\times 10^{-3}m)$	$E_{\theta_x}(\text{°})$	$E_{\theta_y}(\text{°})$	$E_{\theta_z}(\text{°})$	$E_T(\times 10^{-3}m)$	$E_{\theta_x}(\text{°})$	$E_{\theta_y}(\text{°})$	$E_{\theta_z}(\text{°})$
Image 1	13	1.235	0.097	0.165	1.712	0.165	0.087	0.143	0.166
Image 2	6	3.493	2.456	1.034	1.168	0.477	0.824	0.647	0.891
Image 3	10	2.712	1.955	1.121	2.03	0.235	0.7513	0.399	0.584
Image 4	3	4.951	1.113	2.833	1.873	4.951	1.113	2.833	1.873
Image 5	9	1.766	1.881	2.735	1.426	0.363	0.591	0.762	0.244

Tableau 3.5 : résultats des tests d'évaluation du recalage 3D

La première remarque que nous pouvons faire est que le nombre de taches brillantes dans les images de luminance est à chaque fois supérieur à 3, ce qui veut dire qu'il existe une solution pour le problème de recalage 3D. En effet, comme nous l'avons déjà mentionné, les taches brillantes dans les images de luminance correspondent en réalité à des surfaces de l'objet qui sont orientées dans la même direction, en l'occurrence dans ce cas, dans la direction de la source de lumière, or comme nous travaillons sur des objets de forme libre, nous espérons avoir au moins trois régions voire plus de sa surface qui seraient orientées

dans la même direction. De plus, Nous remarquons que le recalage fin améliore nettement la précision, puisqu'il permet de diminuer les erreurs sur la translation et la rotation dans tous les cas, et que sa qualité dépend du nombre total d'appariements tache/sommet. En effet plus ce nombre est grand plus le recalage fin est meilleur. En l'occurrence, dans l'image 4 (cf. figure 3.32), le recalage fin n'apporte pas d'amélioration de la précision par manque de taches supplémentaires sur lesquelles réitérer le processus de recalage. D'un autre côté, nous avons noté que les erreurs en translation sont fortement influencées par les erreurs en rotations, ceci est tout à fait logique car le vecteur de translation dépend de la rotation.

### 3.5.6 Vérification d'hypothèses

Le dernier critère à définir concerne le module de vérification d'hypothèses. En effet ce module est basé sur la mesure de la distance dans l'image (en pixel) entre, à chaque fois, un barycentre d'une tache et un sommet du maillage projeté dans l'image. Ainsi, le critère à définir doit déterminer à partir de quelle distance (que nous avons appelé rayon de vérification) on peut dire que le barycentre de la tache correspond bien au sommet projeté afin de générer un appariement tache/sommet. Suite à une série de résultats obtenus avec différentes valeurs de rayons, nous avons pu constater que l'utilisation d'un rayon trop important engendrait des fausses reconnaissances. En effet, nos essais ont été effectués avec des rayons de 5,10,15 et 20 pixels. Pour un rayon égal à 5 pixels, nous avons constaté que le nombre de sommets projetés mis en correspondance avec les barycentres des taches est restreint mais suffisant pour vérifier une hypothèse. Par contre, au fur et à mesure que ce seuil augmente, on s'aperçoit que de plus en plus de sommets sont appariés mais de façon inopinée. Ainsi, nous avons retenu le rayon de 5 pixels qui nous semble être la meilleure solution pour avoir une bonne précision de la vérification tout en évitant des temps de calculs importants.

## 3.6 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle méthode de recalage automatique 3D pour des objets de forme libre qui utilise le concept de la photoclinométrie pour établir des appariements image/modèle. En effet, nous avons démontré que sous certaines conditions (voir § 3.1.1), la photoclinométrie nous permet d'extraire à partir d'une seule image de luminance d'un objet de forme libre des invariants projectifs de type photométrique permettant ainsi de reconnaître les points du modèle vus dans l'image, il s'agit en l'occurrence des répartitions angulaires selon les profils en croix. Nous avons montré comment exploiter cet invariant pour construire, hors ligne, une base de données qui répertorie d'une manière simple et efficace les informations pertinentes sur le modèle permettant ainsi de l'apparier avec les primitives image (les barycentres des taches brillantes). D'un autre côté, nous avons utilisé l'algorithme de l'itération orthogonale afin de calculer la pose courante de l'objet dans la scène, il prend en entrée les appariements image/modèle générés par le module d'appariement et donne en sortie la transformation rigide recherchée. Enfin, afin d'établir une évaluation de l'efficacité de ce système de recalage, nous l'avons testé sur une série d'images représentant le modèle dans des poses

différentes. Ces tests n'ont pas pour objectif de fournir une mesure exacte de qualité par rapport à une échelle, mais plutôt d'évaluer la qualité des résultats par rapport à nos besoins et de trouver les limites du système. De plus, ces tests sont destinés à obtenir les réglages les plus appropriés lorsque certaines variables sont à paramétrer. A la suite de l'analyse des résultats obtenus, nous avons relevé les défauts suivants:

- La difficulté de mettre en pratique l'hypothèse de base concernant la source de lumière directionnelle. En effet, nous n'avons pas réussi à trouver un spot qui soit assez directionnel pour générer des images réelles satisfaisantes, ceci nous a amené à travailler sur des images de synthèse. Ainsi, les conclusions établies à partir de ces images possèdent une portée plus ou moins limitée.
- L'invariant projectif que nous avons construit est un invariant global, ce qui veut dire que l'objet doit être isolé dans la scène (non occlus) pour que l'on puisse extraire cet invariant d'une manière discriminante.

En contrepartie, de nombreux points positifs ont été relevés:

- La qualité de la discrimination des aspects photométriques, qui permet de définir chaque point de vue d'une manière non équivoque.
- La robustesse et la précision de la mise en correspondance, qui génère un nombre limité d'hypothèses classées par ordre, de la plus probable à la moins probable.
- La pertinence de la vérification basée sur la mesure de distance dans l'image permettant ainsi de rejeter les mauvaises hypothèses et de confirmer la reconnaissance de l'aspect photométrique de l'objet.
- La qualité du recalage 3D, qui est dû à la qualité du système et à sa capacité d'apparier avec précision, un nombre important ( $\geq 3$ ) de points image avec des points 3D du modèle.

# Chapitre 4

## Approche Clinométrique

Dans ce chapitre, nous allons présenter une autre solution pour le recalage automatique d'objets de forme libre. A la différence de l'approche photoclinométrique exposée dans le chapitre précédent, la solution clinométrique ne dépend pas des conditions propres d'éclairage de la scène, elle utilise la projection d'une grille laser sur la surface de l'objet afin d'en extraire des informations 3D à partir desquelles elle construit un invariant géométrique à la rotation et à la translation. Ce dernier est ensuite utilisé pour indexer, dans une table d'indexation du modèle, la région de l'objet reconstruite. De plus, cet invariant est de type local, ce qui fait de la méthode clinométrique une méthode robuste aux occlusions. Nous allons décrire, en détails, les outils et techniques élaborés pour mettre en œuvre cette solution. Nous allons ainsi commencer par décrire le système de vision réalisé pour reconstruire les données 3D (calibration, détection subpixel, appariement grille laser/grille projetée,...), nous démontrons la précision (indispensable pour le recalage) de ce système grâce à des mesures effectuées sur un objet étalon. Dans la deuxième partie de ce chapitre, nous allons décrire, comme pour la méthode photoclinométrique, le processus de recalage 3D proprement-dit étape par étape. Nous commencerons par décrire la phase hors ligne qui, dans ce cas, consiste à construire une table d'indexation du modèle indexer par l'invariant clinométrique, ensuite nous exposerons la méthode d'appariement objet/modèle (en ligne), et enfin nous décrirons le formalisme mathématique de la localisation 3D. Nous concluons sur les contributions que nous avons amenées ainsi que sur les résultats obtenus.

## 4.1 Principe de la méthode

La méthode clinométrique est une approche de recalage qui exploite, comme la méthode photoclinométrique, les informations sur les normales à la surface de l'objet de forme libre. La différence entre les deux approches réside, d'une part, dans la manière d'extraire ces informations, et d'autre part, dans leur utilisation pour définir l'invariant géométrique. En effet, dans la méthode clinométrique nous reconstruisons les points 3D appartenant à chaque fois à une région locale de la surface visible de l'objet grâce à un capteur stéréo actif constitué d'une caméra CCD combinée avec un projecteur d'une grille laser [Ababsa et al., 2002c]. A partir de ces points, nous construisons des patches de surface qui sont des maillages partiels caractérisés, en plus des coordonnées 3D des points, par les coordonnées 3D des normales en ces mêmes points. Pour chaque patch ainsi défini, nous déterminons sa distribution angulaire ainsi que ses courbures moyenne et gaussienne [Dorai et Jain, 1997]. Pour cela, nous calculons les angles entre la normale au point central du patch et les normales aux autres points appartenant à ce même patch, puis nous déterminons leur distribution. Cette distribution angulaire ainsi que les fonctions de courbure (moyenne et gaussienne) sont invariantes à la rotation et à la translation [Zisserman et al., 1995] (cf. figure 4.1), de plus, elles sont discriminantes pour les objets de forme libre, et de ce fait, caractérisent les régions locales de l'objet, ce qui permettra de les identifier dans la base de données.

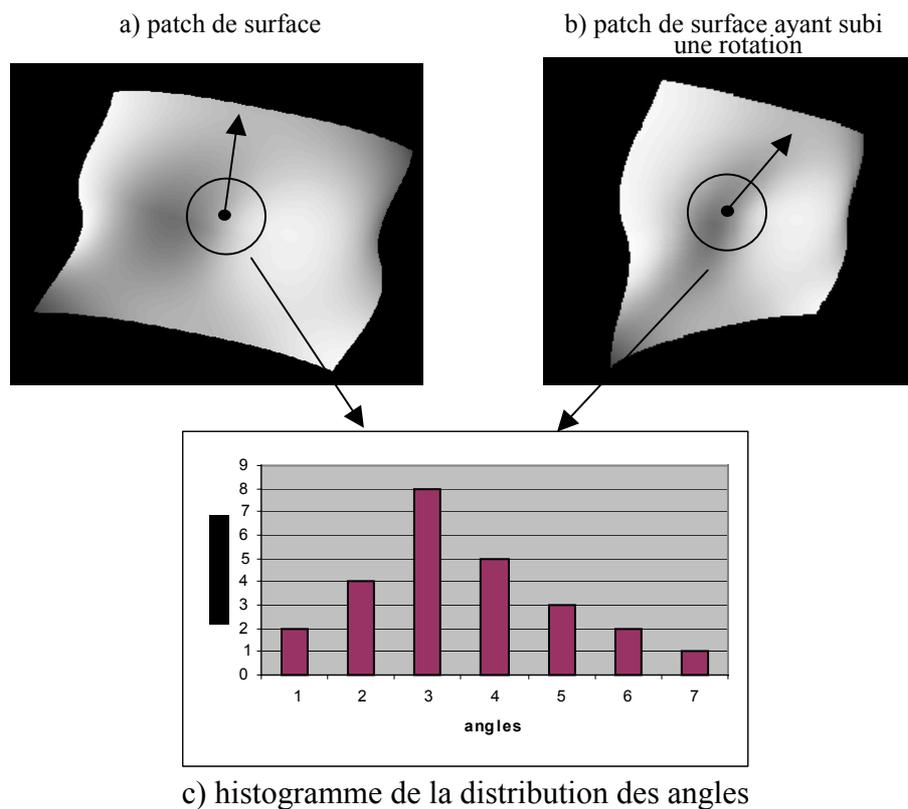


Figure 4.1: Principe de l'invariance de la distribution angulaire associée à un patch de surface

Afin de pouvoir exploiter ces invariants d'une manière efficace pour effectuer des appariements objet/modèle, nous avons créé, hors ligne, une table d'indexation qui organise les différents patches<sup>1</sup> de surface de l'objet d'une manière ordonnée grâce à des index calculées à partir de la distributions angulaire de chaque patch de surface. Il s'agit en l'occurrence des moments d'ordre 1 et 2 (la moyenne et l'écart type) de la distribution angulaire. Ainsi, chaque cellule de la table d'indexation contient la liste de tous les patches de surface du modèle possédant cet index, et dans chaque entrée de cette liste nous stockons l'indice du patch ainsi que ses courbures moyenne et gaussienne. Ensuite le même procédé calculatoire est appliqué en ligne à chaque patch de surface reconstruit pour accéder à un ou plusieurs patches de surface du modèle possédant un index et des courbures similaires dans la base de données. De cette façon, une majorité de patch du modèle se trouve écartée de la mise en correspondance, ce qui réduit d'une manière significative la complexité du processus d'appariement.

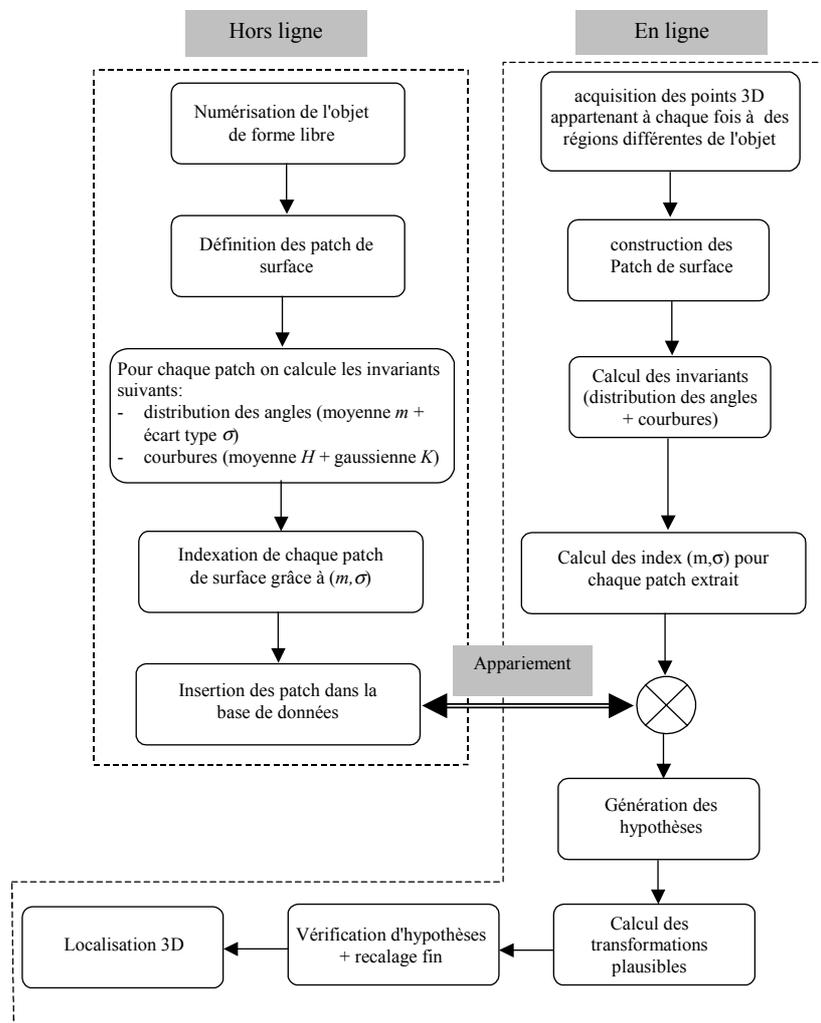


Figure 4.2 : Architecture du système de recalage clinométrique

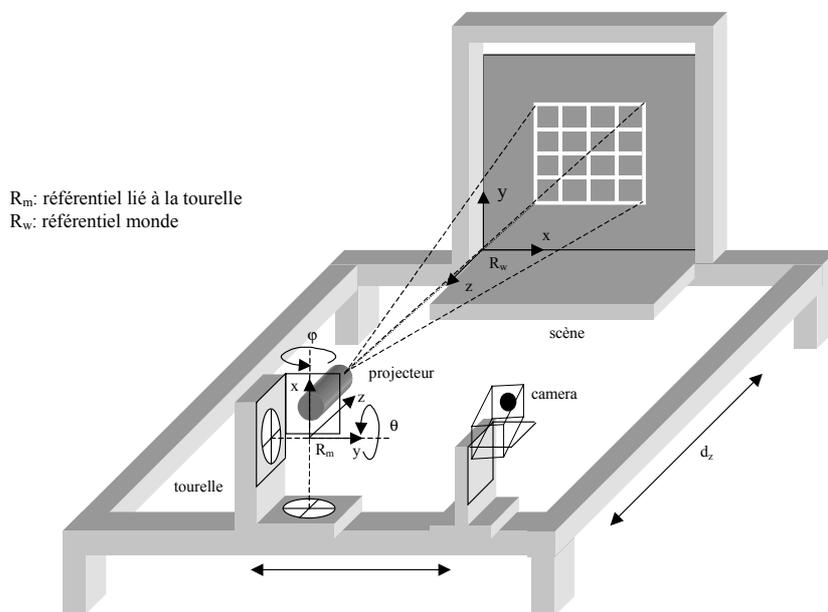
<sup>1</sup> Les patches de surface sont construits à partir d'un ensemble structuré de points 3D obtenu lors de la phase de numérisation de l'objet de forme libre.

Enfin, comme pour l'approche photoclinométrique, nous avons choisi le paradigme de génération/vérification d'hypothèses pour appairer l'ensemble des patches de surface extraits de l'objet dans sa position courante avec ceux de la base de données [Huttenlocher et Ullman, 1987]. La vérification d'une hypothèse étant effectuée par re-projection dans l'image de luminance des patches du modèle transformés par la transformation rigide déterminée par le recalage.

La figure (4.2) illustre l'architecture globale du système clinométrique, subdivisée en deux grandes parties : en ligne et hors ligne, qui permettent de former la base de données du modèle et d'établir la mise en correspondance [Ababsa et al., 2002d].

## 4.2 Système de vision

Le capteur de vision retenu afin de reconstruire les points 3D de la surface de l'objet de forme libre se compose d'un projecteur de grille laser et d'une caméra CCD. Le projecteur projette une grille laser carrée composée de cinq lignes horizontales et cinq lignes verticales ([Jarvis, 1983], [Valkenburg et McIvor, 1998], [Rocchini et al., 2001]). Le principe général de cette approche consiste à reconstruire, dans l'espace, les nœuds<sup>2</sup> de la grille déformée projetée sur l'objet, à partir de l'image de cette grille acquise par la caméra (cf. figure 4.3). Cette reconstruction 3D s'effectue par triangulation en utilisant les nœuds de la grille d'origine avec les nœuds homologues de la grille image. La précision de la reconstruction est alors liée à la précision du calibrage de l'ensemble {caméra, projecteur}, et de la qualité de l'appariement des nœuds homologues (grille/image) qui est fortement liée à la détection des nœuds dans l'image de la grille projetée, d'où la détection subpixel de croix.



<sup>2</sup> Un nœud est défini comme étant le point d'intersection entre une ligne horizontale et une ligne verticale de la grille laser.

Figure 4.3 : Capteur stéréo actif 3D

De plus, afin de pouvoir reconstruire plusieurs points d'une région quelconque de l'objet, nous avons monté le projecteur laser sur une tourelle à deux degrés de liberté (cf. figure 4.3) lui permettant ainsi d'avoir deux mouvements de rotation dans le référentiel ( $R_m$ ), le premier selon l'axe ( $ox$ ) et le second selon l'axe ( $oy$ ) qui permettent de balayer l'objet. De cette façon, nous pouvons acquérir n'importe quelle région de l'objet avec la résolution que l'on souhaite. En effet, la résolution étant définie par l'angle de balayage que nous pouvons contrôler avec une grande précision.

Enfin, Nous pouvons aussi appliquer une translation ( $d_z$ ) au capteur 3D (projecteur+caméra) selon l'axe ( $oz$ ) du référentiel de la scène.

### 4.2.1 Calibration du système de vision

La méthode de calibration développée est basée sur l'utilisation du modèle sténopé pour la caméra mais aussi le projecteur ([Tsai, 1987], [Faugeras, 1993]). En effet, d'un point de vue mathématique, les principes de projection centrale sont exactement les mêmes pour le projecteur, et de ce fait ce, dernier peut être modélisé de manière similaire. Les systèmes de coordonnées utilisés dans la procédure de calibration du système de vision sont illustrés dans la figure (4.4).

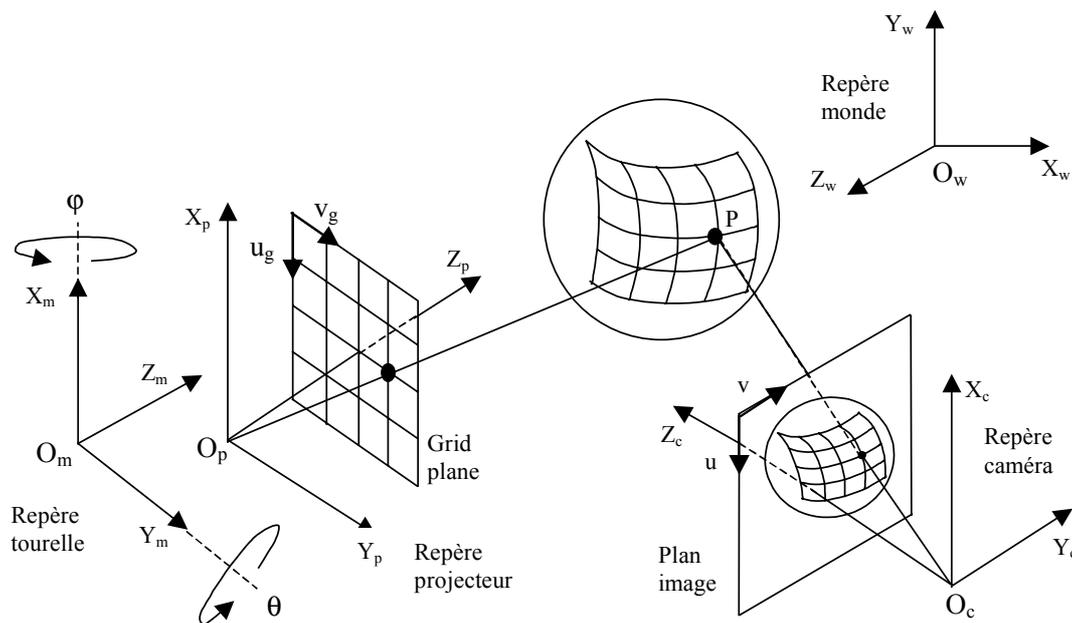


Figure 4.4 : Systèmes de coordonnées

#### 4.2.1.1 Calibration de la caméra

Soit  $P(x,y,z)$  un point 3D représenté dans le repère monde (cf. figure 4.4), la relation entre le point  $P$  et son correspondant  $p(u,v)$  dans le plan image est donnée par :

$$\begin{bmatrix} s.u \\ s.v \\ s \end{bmatrix} = M_c \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (4.1)$$

où  $s$  est un facteur d'échelle, et  $M_c$  la matrice (3×4) de projection perspective (voir annexe A).

Calibrer une caméra consiste donc à déterminer les éléments de la matrice  $M_c$ . Pour cela, on utilise un ensemble de  $n$  points dont on connaît la position dans le repère monde. Pour chacun de ces points, on mesure la position  $(u,v)$  de leur projection dans le plan image. Un ensemble de quintuplets  $\{(X_i, Y_i, Z_i, u_i, v_i)\}_{i=1, \dots, n}$  est alors obtenu. Les éléments de la matrice  $M_c$  sont alors la solution au sens des moindres carrés d'un système d'équations linéaires où le nombre d'équations est supérieur ou égal au nombre d'inconnues. En pratique, nous avons utilisé la grille projetée comme mire de calibration, pour cela nous avons tout d'abord placé notre capteur de vision à une distance  $d_1$  du plan  $(oxy)$  du référentiel de la scène pour simuler un plan vertical ( $z=d_1=100\text{mm}$ ) dans le repère monde, ensuite nous avons projeté la grille laser sur ce plan pour différentes valeurs de  $\varphi$  et  $\theta$ . Ensuite, nous avons mesuré les coordonnées 3D dans le repère monde des nœuds de la grille projetée. Les angles  $\varphi$  et  $\theta$  ont été choisis de telle manière à ce que les points 3D générés soient répartis de façon homogène dans l'espace de travail afin d'augmenter la précision de la reconstruction 3D et donc du recalage. Ce processus nous permet, d'extraire un certain nombre de points 3D coplanaires. Il doit être effectué au moins à deux profondeurs différentes du plan virtuel (nous avons choisi  $d_2=300\text{ mm}$  et  $d_3=600\text{ mm}$ ) car la calibration impose d'avoir des points qui ne soient pas tous coplanaires. La précision de la matrice  $M_c$  dépend directement du nombre de points, dans notre cas nous avons utilisé en tout 126 points. Par ailleurs, pour estimer les coordonnées image des nœuds. Nous avons élaboré un algorithme (voir § 4.1.2) qui, à partir d'une image de luminance de la grille projetée prise par la caméra, détermine d'une part les coordonnées image des nœuds de cette grille avec une précision subpixel, et d'autre part, apparie chaque nœud dans l'image avec le nœud qui lui correspond dans la grille d'origine, ce qui permet ainsi de générer l'ensemble des quintuplets nécessaires pour déterminer  $M_c$ .

#### 4.2.1.2 Calibration du projecteur

Comme nous l'avons déjà mentionné, notre projecteur est orientable en azimut d'un angle  $\varphi$  et en site d'un angle  $\theta$ , selon les deux axes  $(O_m X_m)$  et  $(O_m Y_m)$  du repère de la tourelle respectivement. Les angles  $\varphi$  et  $\theta$  sont contrôlables avec une grande précision (0.01 degré), grâce à des moteurs pas à pas. Nous avons donc développé une méthode qui permet de calibrer le projecteur laser quelque soit les valeurs des angles  $\varphi$  et  $\theta$ .

Dans un premier temps, nous avons calibré le projecteur dans sa position initiale, en l'occurrence pour  $\varphi = \theta = 0$ . La relation entre les nœuds 3D de la grille exprimés dans le repère monde et les nœuds de la grille d'origine est donnée par:

$$\begin{bmatrix} t \cdot u_g \\ t \cdot v_g \\ t \end{bmatrix} = M_p \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (4.2)$$

où  $t$  est un facteur d'échelle et  $M_p$  la matrice de projection perspective du projecteur.

De la même manière que pour la calibration de la caméra, nous avons généré plusieurs points 3D en utilisant la procédure expliquée dans la section précédente. Et comme nous connaissons, pour chaque point 3D, le nœud qui lui correspond dans la grille d'origine, nous construisons ainsi l'ensemble des quintuplets  $\{(X_i, Y_i, Z_i, u_{gi}, v_{gi})\}_{i=1, \dots, n}$ . Les éléments de la matrice  $M_p$  sont alors la solution au sens des moindres carrés du système d'équations linéaires construit à partir de l'équation (4.2).

Par ailleurs, nous remarquons que lorsque le projecteur est orienté d'un angle quelconque autour de l'axe  $(O_m X_m)$  ou bien  $(O_m Y_m)$ , ce sont uniquement ses paramètres extrinsèques qui changent (en d'autres termes ce sont la matrice de rotation et le vecteur de translation entre le repère monde et le repère projecteur qui changent), les paramètres intrinsèques restent quant à eux inchangés. Ainsi, à partir de la matrice de projection perspective  $M_p$ , on extrait la matrice intrinsèque (i.e. les paramètres internes du projecteur) et la matrice extrinsèque notées  $I_p(3 \times 4)$  et  $A_p(4 \times 4)$  respectivement, tel que :

$$M_p = I_p \cdot A_p \quad (4.3)$$

et

$$A_p = \begin{pmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0_{1 \times 3} & 1 \end{pmatrix} \quad (4.4)$$

où  $R_{3 \times 3}$  et  $T_{3 \times 1}$  sont respectivement, la matrice de rotation et le vecteur de translation entre le repère monde et le repère projecteur respectivement.

Lorsque la tourelle effectue deux rotations simultanées, la première d'un angle  $\varphi$  autour de l'axe  $(O_m X_m)$  et la deuxième d'un angle  $\theta$  autour de l'axe  $(O_m Y_m)$ , la matrice de rotation devient alors :

$$R' = (R_\varphi \cdot R_\theta) R \quad (4.5)$$

De même, le vecteur de translation devient:

$$T' = C_{m/w} \cdot ([R_\varphi \cdot R_\theta] \cdot O_{p/m}) \quad (4.6)$$

où  $C_{m/w}$  exprime la transformation rigide du repère de la tourelle vers le repère monde. L'équation (4.6) implique que nous calculons tout d'abord les coordonnées du centre de projection du projecteur laser dans le repère de la tourelle après qu'il ait subi la rotation, ensuite nous multiplions le résultat par la matrice  $C_{m/w}$  pour obtenir le nouveau vecteur de translation  $T'$  exprimé dans le repère monde.

Ainsi, la nouvelle matrice extrinsèque  $A'_p$  peut être écrite comme suit:

$$A'_p = \begin{pmatrix} R'_{3 \times 3} & T'_{3 \times 1} \\ \mathbf{0}_{1 \times 3} & 1 \end{pmatrix} \quad (4.7)$$

Finalement, la matrice de projection perspective  $M'_p$  du projecteur laser dans sa position courante (après rotation) est donnée par :

$$M'_p = I_p \cdot (A'_p)^{-1} \quad (4.8)$$

L'équation (4.8) est très intéressante car elle nous permet de recalibrer le projecteur laser en ligne lorsque ce dernier subi des rotations quelconques selon les axes  $(O_m X_m)$  et  $(O_m Y_m)$  du repère de la tourelle sans avoir pour autant à refaire toute la procédure de calibration.

## 4.2.2 Extraction de la grille image

La reconstruction tridimensionnelle de points à partir de leurs projections dans les plans images implique la capacité de détecter ces projections de nœuds avec une grande précision. Ce paragraphe décrit les algorithmes mis en œuvre afin d'arriver à une détection subpixelique [Izquierdo et al., 1999] des nœuds de la grille image. En effet, nous avons vu dans la section précédente que la détection des nœuds est nécessaire non seulement pour la calibration de la caméra mais aussi pour la reconstruction 3D des points de l'objet de forme libre qu'on veut recalibrer.

La procédure d'extraction de la grille se subdivise en deux étapes :

1. détection des nœuds.
2. reconstitution de la grille image.

### 4.2.2.1 Détection des nœuds

Chaque nœud dans l'image de la grille projetée est considéré, localement, comme le centre d'une croix. Plusieurs algorithmes permettant de détecter le centre d'une croix existent dans la littérature [Dérêche et Giraudon, 1993], cependant ils prennent comme point de départ une image binaire (noir et blanc) ce qui induit des imprécisions importantes dans la détection. Nous proposons une méthode qui détecte le centre d'une croix dans une image de luminance (en niveau de gris) en se basant sur une modélisation mathématique de la luminance d'une croix. Le modèle proposé permet de modéliser des croix de forme quelconque (des croix obliques et/ou droites, cf. figure 4.5), il est défini par l'équation suivante :

$$L(x,y) = a + b \cdot \left( e^{c \cdot (x-d \cdot y - e)^2} + e^{-f \cdot (y-g \cdot x - h)^2} - e^{-c \cdot (x-d \cdot y - h)^2 - f \cdot (y-g \cdot x - h)^2} \right) \quad (4.9)$$

tel que:

- a est la valeur minimale de la luminance  $L$ .
- b est la hauteur de la croix.

- c et f représentent la largeur des deux axes de la croix.
- d et g définissent l'inclinaison des deux axes de la croix.
- e et h définissent la position de la croix.

Il est clair que le modèle de l'équation (4.9) s'applique uniquement à des images de luminance contenant une et une seule croix, ce qui n'est pas le cas ici où nous avons à manipuler des images contenant la totalité de la grille projetée. Ainsi, pour pouvoir utiliser ce modèle, nous devons tout d'abord extraire, à partir de l'image originelle, de petites images que nous avons appelées "imassettes" qui ne contiennent qu'une seule croix à la fois.

Le processus de détection précise des centres de croix est le suivant:

- prédétection approximative des centres des croix.
- extraction d'une imasette centrée sur chaque point détecté ci-dessus.
- détermination des paramètres du modèle de l'équation (4.9) pour chaque imasette afin qu'il soit le plus proche possible de la luminance des imassettes réelles.
- Calcul de la position exacte dans l'image globale du centre de la croix à partir des paramètres du modèle.

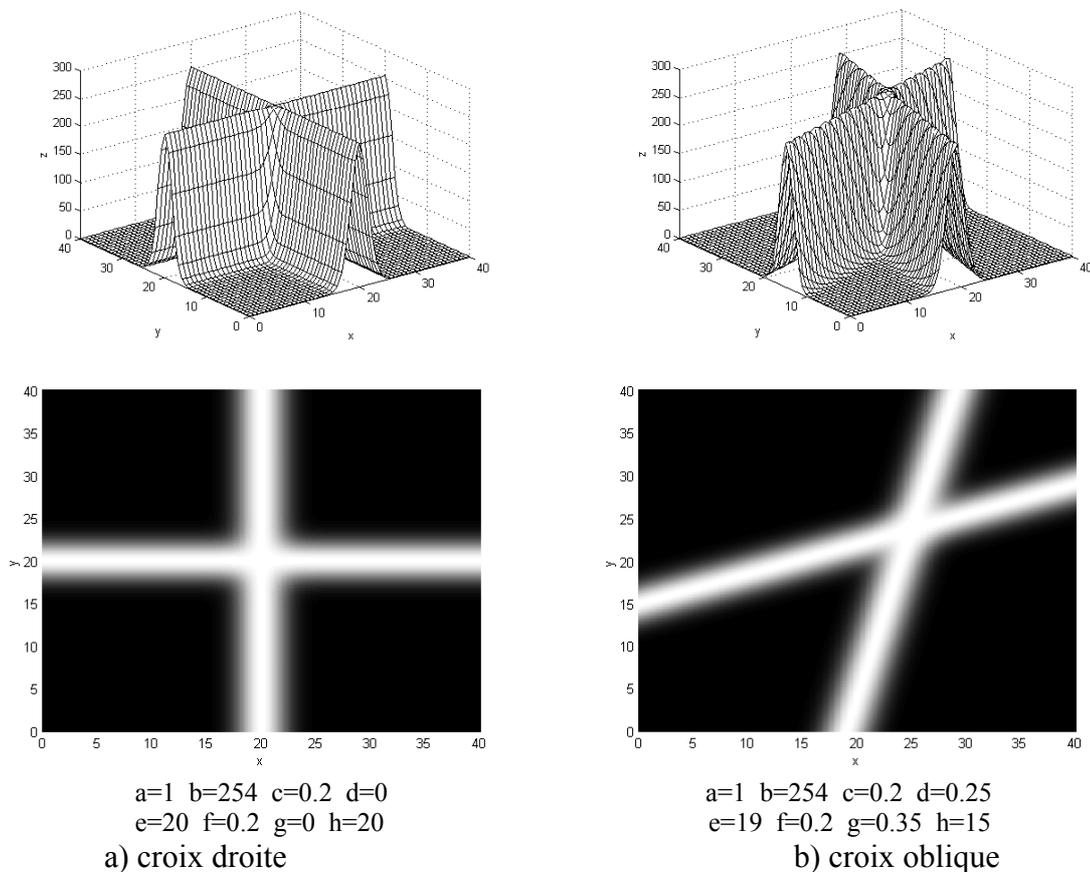


Figure 4.5: Exemples de croix

La détection approximative des centres des croix est obtenue suite à un ensemble de traitements que nous effectuons sur l'image de luminance initiale et que nous détaillons ci-dessous :

**Binarisation de l'image** : la binarisation consiste à transformer une image en niveaux de gris en une image binaire. Pour cela nous fixons un seuil (un entier compris entre 0 et 255) puis nous parcourons l'image en mettant les pixels à 0 (noir) si leur valeur est inférieure au seuil et à 255 (blanc) si leur valeur lui est supérieure. Cette procédure construit, en même temps que l'image binarisée, une liste des points blancs de l'image. Cette liste permet d'améliorer le temps des traitements sur l'image seuillée. En effet, tous ces traitements s'appliquent aux points blancs et le fait de les repérer dès la binarisation évite de parcourir l'image inutilement lors des procédures suivantes.

**Squelettisation** : cette opération a pour but de décrire l'objet de départ par un ensemble de lignes infiniment fines (analogues à une ossature de l'objet), réduisant sensiblement le volume d'information à manipuler. Le squelette est généralement défini comme étant l'ensemble des lignes médianes, c'est-à-dire l'ensemble des points équidistants de deux points de la frontière (voir chapitre 2). La méthode de squelettisation que nous avons utilisée est la méthode d'amincissement : l'objet est assimilé à un oignon épluché couche par couche de façon itérative, ou à une prairie sur le pourtour de laquelle on allume un feu ; le squelette est le point de rencontre des fronts de feu. Pratiquement, pour obtenir le squelette nous balayons l'image binarisée avec les 8 masques 3×3 de la figure (4.6) [Coster et Chermant, 1989]. Si l'une des configurations est vérifiée, le pixel central est remplacé par 0 (noir). L'opération est réitérée jusqu'à stabilité. Si les frontières de l'objet sont bruitées le squelette peut comporter beaucoup de barbules. Il convient d'appliquer alors des post-traitements (cf. figure 4.7 partie droite).

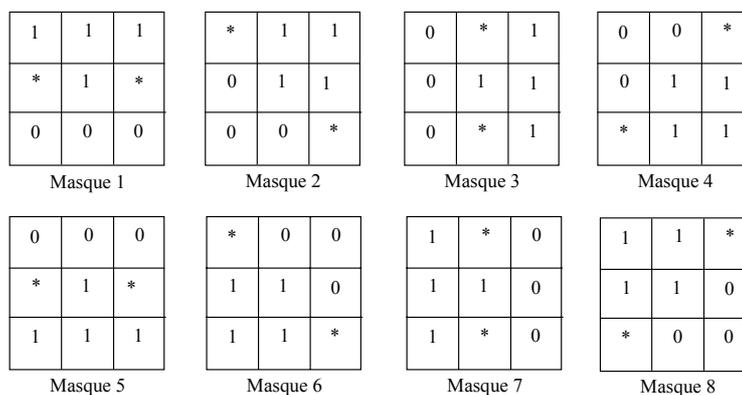


Figure 4.6 : Masques de squelettisation (les "\*" correspondent aux pixels dont la valeur est indifférente).

Comme les points noirs ne peuvent correspondre avec aucun des huit masques présentés ci-dessus, il est donc inutile de les tester. Ainsi, l'algorithme de squelettisation teste uniquement les points blancs de la liste construite lors de la phase de binarisation. De cette

façon le temps imparti à la squelettisation se retrouve énormément réduit. La procédure de squelettisation modifie l'image et met à jour la liste des points blancs.

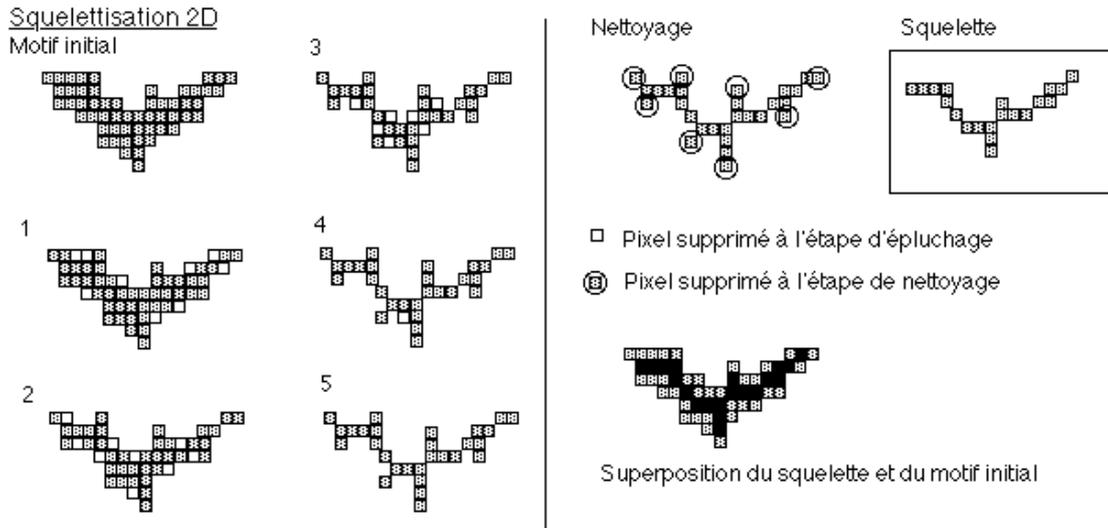


Figure 4.7 : Exemple de squelettisation par amincissement

**Prédétection des centres des croix :** l'image squelettisée permet d'avoir une idée sur la forme et la localisation de la grille. Un nœud est, par définition, un point situé à l'intersection de deux courbes. Ainsi, un point sera considéré comme le centre approximatif d'une croix s'il satisfait les critères suivants:

1. c'est un point blanc.
2. il a au moins quatre voisins blancs.
3. il a quatre blocs de points sur la frontière à une distance donnée (cf. figure 4.8).
4. les quatre points sur la frontière sont opposés deux à deux.

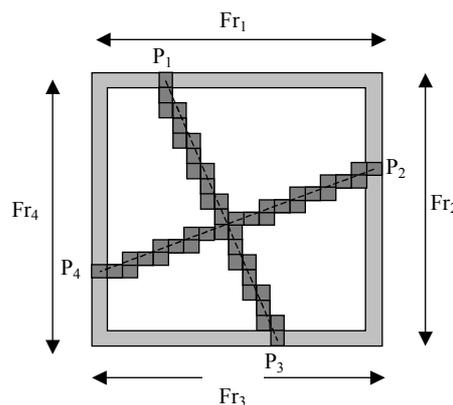


Figure 4.8 : Définition d'un centre de croix approximatif

Soit  $p(i,j)$  un point de l'image squelettisée qui satisfait les critères (1) et (2), et  $F$  une fenêtre  $(2n+1) \times (2n+1)$  centrée sur  $p$ . les quatre blocs  $Fr_1$ ,  $Fr_2$ ,  $Fr_3$  et  $Fr_4$  sont définis de la façon suivante (cf. figure 4.6):

$$\begin{aligned}
 Fr_1 &= \text{val}(i-n, (j-n)+(k-n)) && \text{pour } k=1, 2n+1 \\
 Fr_2 &= \text{val}((i-n)+(k-(2n+1)), j+n) && \text{pour } k=2n+2, 4n+1 \\
 Fr_3 &= \text{val}(i+n, (j+n)+((4n+1)-k)) && \text{pour } k=4n+2, 6n+1 \\
 Fr_4 &= \text{val}((i+n)+((6n+1)-k), j-n) && \text{pour } k=6n+2, 8n
 \end{aligned} \tag{4.10}$$

avec  $\text{val}(i,j)$  = la valeur du pixel  $(i,j)$ .

Quelquefois il arrive que plusieurs points très proches puissent correspondre au centre de la même croix. Dans ce cas, puisqu'une détection approximative suffit, un seul point est gardé.

**Détection précise des centres des croix** : la prédétection des nœuds sur une image squelettisée est imprécise, elle sert surtout à isoler les zones d'intérêt dans l'image originelle. Ainsi, pour chaque centre de croix prédétesté, on extrait une imagerie de l'image de luminance contenant une seule croix. Pour chaque imagerie  $I(i,j)$ , nous utilisons la méthode d'optimisation non linéaire de Levenberg-Marquardt (voir annexe E) pour déterminer les paramètres optimaux du modèle qui minimisent l'erreur quadratique de luminance entre l'imagerie et le modèle, définie par:

$$C = \sum_{i=1}^{N_i} \sum_{j=1}^{N_c} [L(i,j) - I(i,j)]^2 \tag{4.11}$$

Le vecteur de paramètres est défini par:

$$V = [a \ b \ c \ d \ e \ f \ g \ h] \tag{4.12}$$

Dans cette méthode le vecteur de paramètres  $V$  doit nécessairement être initialisé. Nous avons noté que certains paramètres varient peu d'une imagerie à l'autre et ont pu être déterminés de manière expérimentale, c'est le cas notamment des paramètres  $c$  et  $f$  que nous avons fixé à:

$$c=f=0.2 \tag{4.13}$$

D'autres paramètres dépendent des valeurs minimale et maximale du niveau de gris dans l'imagerie courante,  $I_{min}$  et  $I_{max}$  respectivement:

$$a = I_{min} \ , \ b = I_{max} - a \tag{4.14}$$

Les paramètres restants sont initialisés en utilisant des méthodes d'approximation appliquées à l'imagerie squelettisée. Ainsi, nous déterminons les points  $P_1, P_2, P_3, P_4$  (cf. figure 4.8) qui sont les points d'intersection de la croix squelettisée avec les quatre blocs  $Fr_1, Fr_2, Fr_3$  et  $Fr_4$  respectivement, ensuite nous les utilisons pour initialiser les paramètres restants tel que :

$$e = P \quad h = P_4$$

$$d = \frac{\| \vec{P_1 P_2} \|}{N_b} \quad g = \frac{\| \vec{P_4 P_3} \|}{N_b} \quad (4.15)$$

où  $N_b$  est la largeur des blocs (nous avons pris  $N_b=12$ ).

Une fois les paramètres optimaux déterminés, les coordonnées du centre de la croix dans l'imagette courante sont données par :

$$\begin{bmatrix} u_c \\ v_c \end{bmatrix} = \begin{bmatrix} 1 & -g \\ -d & 1 \end{bmatrix} \begin{bmatrix} h \\ e \end{bmatrix} \quad (4.16)$$

Ils correspondent en fait à l'intersection des deux droites qui définissent le maximum de luminance dans le modèle de la croix.

Finalement nous déterminons les coordonnées du centre exact de la croix dans l'image globale.

Nous avons appliqué notre algorithme sur une image de la grille laser projetée sur un plan vertical placé à une distance  $d_z=1.5m$  du projecteur avec  $\varphi=-3^\circ$  et  $\theta=2.2^\circ$  (cf. figure 4.9a).

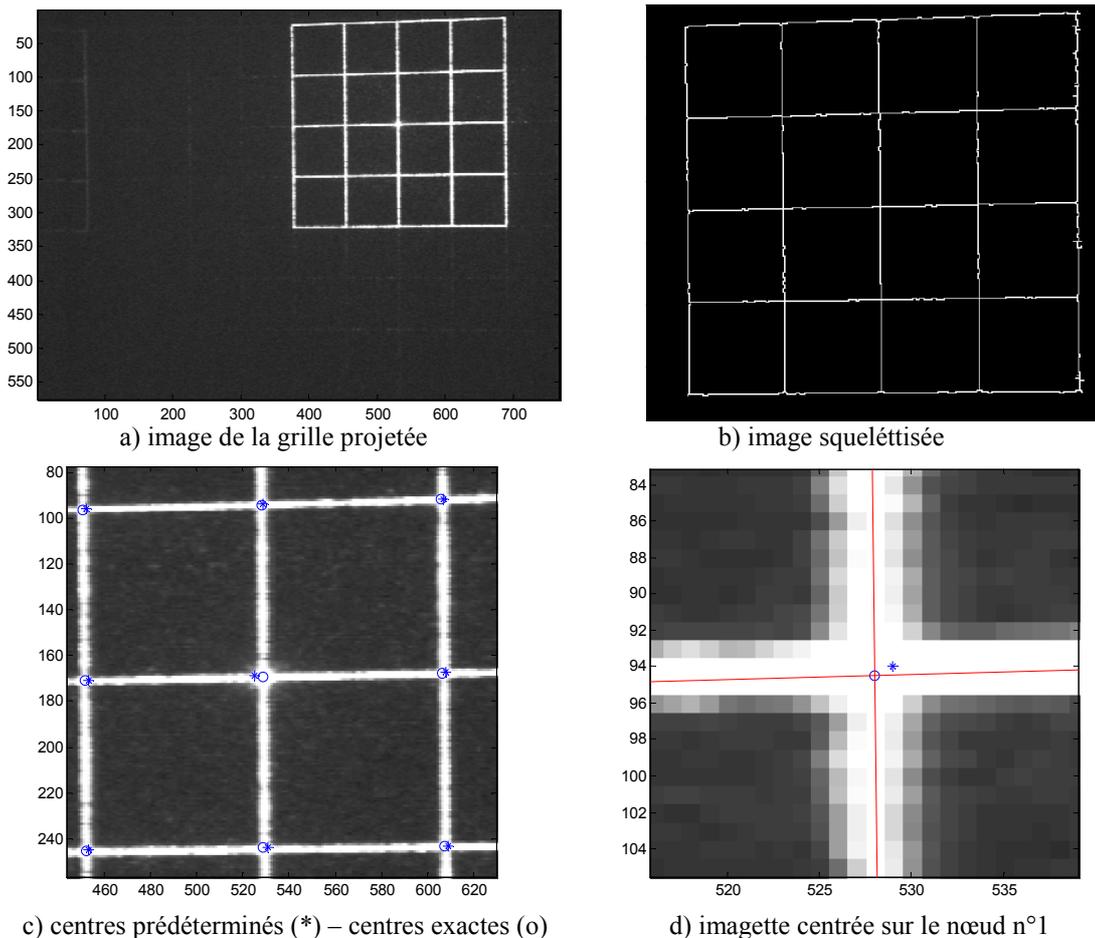


Figure 4.9 : Détection subpixel des nœuds

La figure (4.9b) illustre l'image squelettisée de la grille projetée. Par ailleurs, la figure (4.9d) illustre l'imagette d'une croix que nous avons isolée en utilisant le centre prédéecté N° 1 (en haut à gauche) dans l'image de la grille projetée de la figure (4.9a) et sur laquelle nous avons précisé le centre exact (intersection des deux axes de la croix) ainsi que le centre prédéecté. Les coordonnées dans l'image globale des centres de croix prédéectés et précis déterminés par notre algorithme pour cette exemple sont données dans le tableau suivant :

N° du nœud	Nœuds approximés	Nœuds précis
1	(96, 452)	(96.48, 450.56)
2	(94, 529)	(94.61, 528.06)
3	(92, 607)	(92.17, 606.03)
4	(171, 453)	(171.11, 451.67)
5	(169, 529)	(169.53, 529.07)
6	(167, 608)	(168.09, 606.55)
7	(245, 453)	(245.58, 452.14)
8	(244, 531)	(244.04, 529.02)
9	(243, 609)	(243.05, 607.51)

Tableau 4.1 : Comparaison entre les nœuds prédéectés et les nœuds exacts

Lorsqu'on visualise les centres exacts des croix dans l'image globale de luminance, on voit bien qu'ils correspondent nettement mieux aux centres réels de ces croix que les centres prédéectés (cf. figure 4.9c).

#### 4.2.2.2 Reconstitution de la grille image

Dans cette étape, la grille image sera reconstituée à partir des nœuds déterminés précédemment. Pour cela, nous utilisons le fait que ces nœuds ne sont pas des points quelconques, sans connexion les uns avec les autres : ils forment un motif (en l'occurrence une grille). Il convient donc de les ordonner afin de reconstituer la grille projetée dans l'image ce qui va faciliter son appariement avec la grille d'origine.

La méthode d'ordonnement des nœuds élaborée est inspirée de [Hu et Stockman, 1989] et se déroule comme suit : D'abord, nous parcourons l'image squelettisée de la grille pour extraire, point par point, les courbes horizontale et verticale. Ensuite, nous sélectionnons uniquement les courbes qui contiennent les nœuds prédéectés, ces courbes sont réorganisées de haut en bas et de droite à gauche (cf. figure 4.10). Les nœuds de la grille reconstituée vont correspondre aux points d'intersection entre les courbes horizontales et verticale ordonnées. Ainsi, par exemple, le nœud numéro 1 de la grille reconstituée va correspondre au point d'intersection des courbes ( $H_1, V_1$ ) et ainsi de suite.

La méthode élaborée est très fiable, elle permet de reconstituer la grille image même si les nœuds ne sont pas tous détectés, il suffit de trouver au moins un nœud par ligne et par colonne. Dans tous les cas, on arrive à former la grille image, éventuellement incomplète. Les nœuds manquants n'auront aucune influence sur la suite du traitement, il suffit simplement de les ignorer lors de la reconstruction des points 3D.

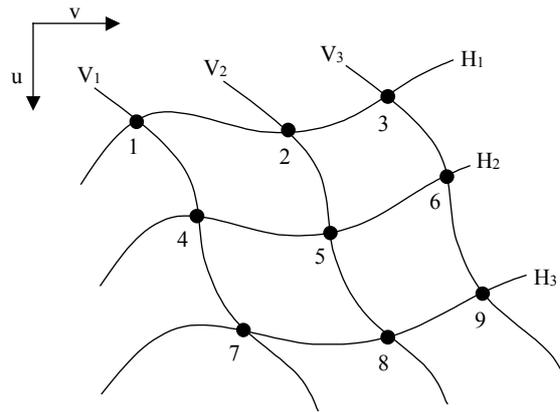


Figure 4.10 : Reconstitution de la grille image

### 4.2.3 Reconstruction des points 3D

Une fois le capteur stéréo actif calibré, l'acquisition via la caméra d'une seule image de la grille projetée sur l'objet de forme libre doit permettre d'en estimer un certain nombre de points 3D. La détection des points de la grille image a déjà été traitée dans la section précédente. Mais pour pouvoir remonter aux points 3D, il est important de pouvoir appairer les points détectés dans le plan image de la caméra et leur correspondants dans le plan image du projecteur. La méthode d'appariement que nous avons retenue est basée sur le fait que la grille détectée dans l'image caméra possède un ordonnancement des nœuds, les uns par rapport aux autres, relativement proche de celui de la grille d'origine dessinée sur l'optique du projecteur (peu de rotation sur elle même) et de ce fait les nœuds image doivent respecter la contrainte d'ordre, c'est à dire qu'ils doivent apparaître dans le même ordre que les nœuds de la grille d'origine. Ainsi, le point situé en haut à gauche de la grille image est apparié au point en haut à gauche de la grille projecteur et ainsi de suite.

Ainsi, pour chaque mise en correspondance entre un nœud de la grille image et un nœud de la grille projetée, nous avons deux ensembles d'équations établies à partir des équations (4.1) et (4.2):

$$\begin{aligned}
 u &= \frac{M_{c11} \cdot x + M_{c12} \cdot y + M_{c13} \cdot z + M_{c14}}{M_{c31} \cdot x + M_{c32} \cdot y + M_{c33} \cdot z + M_{c34}} \\
 v &= \frac{M_{c21} \cdot X + M_{c22} \cdot Y + M_{c23} \cdot Z + M_{c24}}{M_{c31} \cdot X + M_{c32} \cdot Y + M_{c33} \cdot Z + M_{c34}} \\
 u_g &= \frac{M_{p11} \cdot x + M_{p12} \cdot y + M_{p13} \cdot z + M_{p14}}{M_{p31} \cdot x + M_{p32} \cdot y + M_{p33} \cdot z + M_{p34}} \\
 v_g &= \frac{M_{p21} \cdot x + M_{p22} \cdot y + M_{p23} \cdot z + M_{p24}}{M_{p31} \cdot x + M_{p32} \cdot y + M_{p33} \cdot z + M_{p34}}
 \end{aligned} \tag{4.17}$$

Les coordonnées  $(x,y,z)$  du nœud projeté  $P$ , dans le repère monde, se calculent en résolvant ce système de 4 équations linéaires. Nous avons utilisé les moindres carrés pour résoudre ce système d'équations.

#### 4.2.4 Estimation de la précision de la reconstruction 3D

Etant donné que le système de vision est dédié au recalage 3D, nous devons nous assurer de la qualité des mesures 3D qu'il fournit. Pour cela, nous avons estimé sa précision sur un objet étalon (cf. figure 4.11a) pour 2 orientations différentes (cf. figure 4.11b). La procédure d'estimation de la précision est la suivante [Guisser et al., 2000]:

1. projeter la grille sur l'objet étalon de telle sorte que les nœuds de la grille viennent sur les plans de références  $(P_1, P_3)$  pour le test 1 et  $(P_1, P_2)$  pour le test 2.
2. acquérir une image de la grille projetée pour les deux tests.
3. reconstruire, par triangulation, les coordonnées 3D des nœuds projetés dans les plans considérés.
4. Calculer les équations des plans  $P_1, P_3$  pour le test 1 et  $P_1, P_2$  pour le test 2 par approximation linéaire des points des plans.
5. calculer la longueur des segments  $[A,B]$  et  $[B,C]$ , tel que:  
 $[A,B]$  représente la distance entre les deux plans  $(P_1, P_3)$  déterminés à partir du test 1 et  $[B,C]$  représente la distance entre les deux plans  $(P_1, P_2)$  estimés à partir du test 2.
6. Estimer la précision de la reconstruction 3D en comparant les longueurs calculées en (5) avec les longueurs réelles de l'objet étalon.

Le tableau (4.2) donne les mesures obtenues par notre système de vision pour les deux tests tout en les comparant avec les dimensions réelles de l'objet étalon. Nous remarquons que l'erreur moyenne de reconstruction est de 0.9 mm pour une distance entre le projecteur et l'objet de 2 m environ ce qui donne une précision de 0.85% pour le segment  $[A,B]$  et de 0.66% pour le segment  $[B,C]$ . Ainsi, nous pouvons conclure que le système de vision conçu est suffisamment précis pour la tâche à laquelle il sera dédié, en l'occurrence le recalage 3D.

	Objet étalon (mm)	Mesures du système de vision (mm)	Erreur résiduelle (mm)	Erreur %
<i>Test 1</i>				
Segment $[A,B]$	140	138.8	1.2	0.85
<i>Test 2</i>				
Segment $[B,C]$	90	89.4	0.6	0.66

Tableau 4.2 : Résultats de mesure du système de vision élaboré

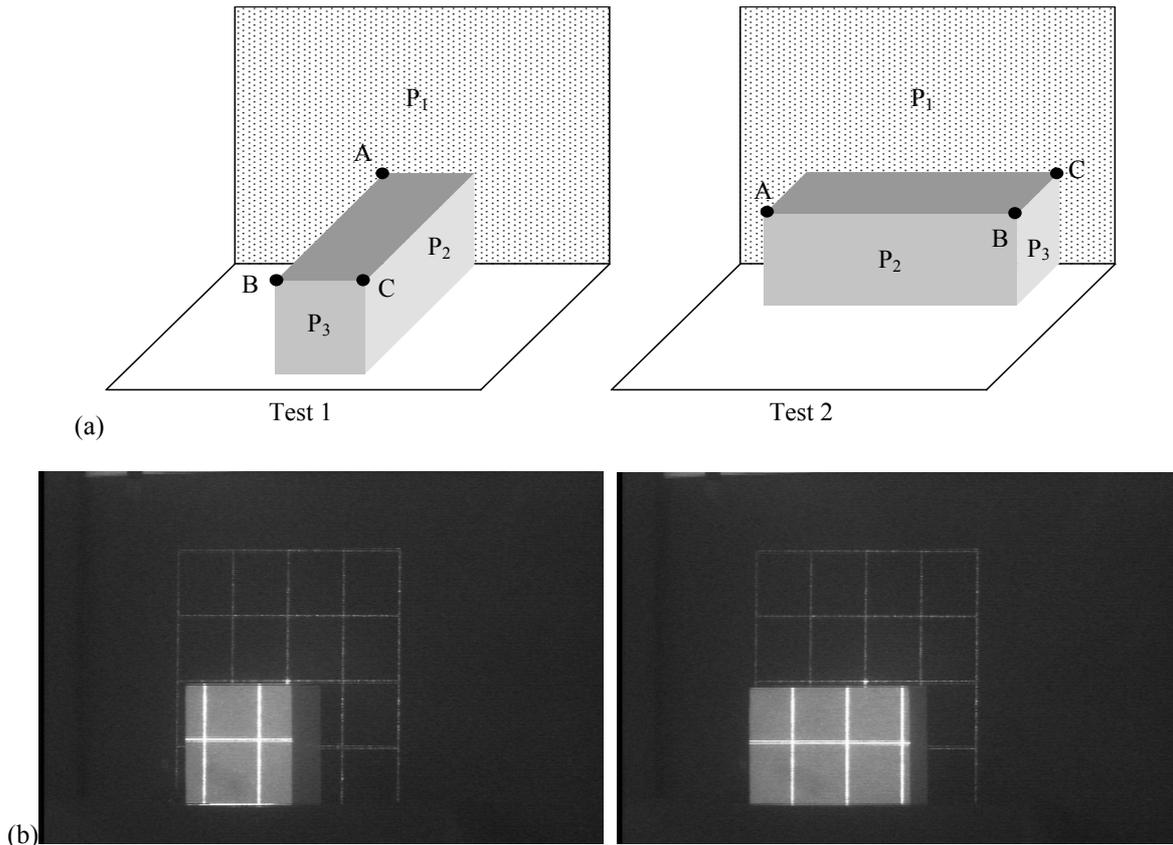


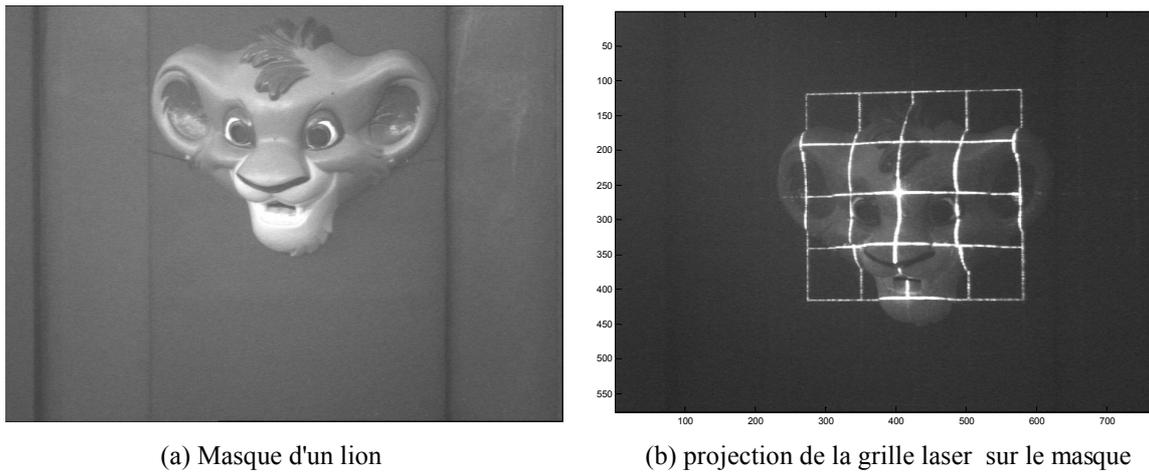
Figure 4.11 : (a) Tests sur l'objet étalon (b) deux vues de la grille projetée sur l'objet étalon

### 4.3 Construction de la base de données

Dans ce paragraphe, nous allons décrire, étape par étape, la procédure que nous avons élaborée pour construire la base de données du modèle de l'objet de forme libre.

#### 4.3.1 Numérisation de l'objet de forme libre

Afin d'illustrer l'ensemble de la méthode clinométrique, nous avons considéré l'exemple particulier d'un masque de lion (cf. figure 4.12a), ce dernier possédant une surface de forme libre. Nous avons utilisé notre système de vision pour numériser quelques régions particulières de cet objet. Pour cela, nous plaçons le masque à une distance du système de vision de telle façon à ce que les nœuds de la grille laser se projettent tous sur lui (cf. figure 4.12b), ceci permet de numériser plusieurs régions de l'objet en même temps. Ensuite, afin d'avoir un ensemble structuré de points 3D pour chaque région de l'objet, nous effectuons un balayage horizontal et vertical de la grille au voisinage des points de référence qui vont constituer les centres des patches de surface reconstruits. Ce balayage est obtenu grâce à la tourelle sur laquelle est monté le projecteur, nous avons fixé le pas angulaire de balayage à  $\Delta\varphi = \Delta\theta = 0.4^\circ$  (ces valeurs sont déterminées expérimentalement). La figure (4.13) illustre le résultat de la numérisation de 9 régions différentes du masque de lion.



(a) Masque d'un lion

(b) projection de la grille laser sur le masque

Figure 4.12 : Objet de forme libre

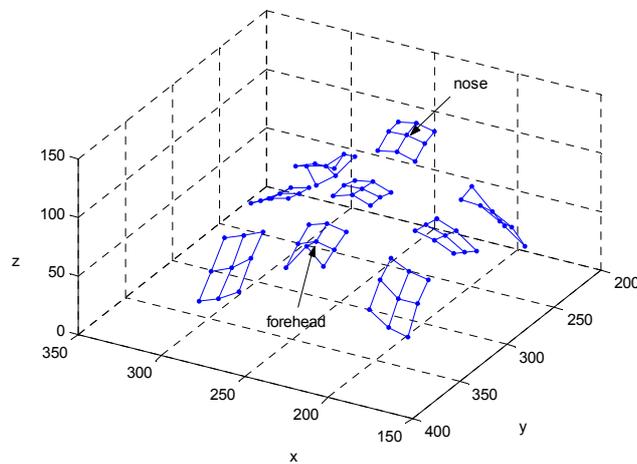


Figure 4.13 : Numérisation de 9 régions du masque de lion

### 4.3.2 Construction des patches de surface

A partir des ensembles structurés de points 3D reconstruits lors de la phase de numérisation, nous construisons les patches de surface pour caractériser chaque région locale de l'objet. Pour cela, nous construisons un maillage triangulaire à partir de chaque ensemble structuré de points 3D. Ainsi, pour chaque patch de surface, nous obtenons deux listes ordonnées  $V$  et  $F$ , où  $V = \{V_1, \dots, V_{N_v}\}$  représente la liste des  $N_v$  sommets  $v_i = (x_i, y_i, z_i)^T$  qui constituent le patch, et  $F = \{f_1, f_2, \dots, f_{N_f}\}$  la liste des faces triangulaires telle que chaque face  $f_i$  est définie par la liste des indices des sommets qui la constituent, c'est à dire  $f_i = \{V_{i,1}, V_{i,2}, V_{i,3}\}$ . Nous calculons aussi, pour chaque patch les normales aux sommets. Chaque normale au sommet est définie comme étant la moyenne des normales à toutes les faces qui partagent ce sommet (voir § 3.1.2). Enfin, nous stockons les informations concernant les différents patches de surface ainsi générés sous forme d'une

table multidimensionnelle. Chaque entrée de cette table correspond à un patch donné et contient la liste des index de tous les sommets appartenant à ce patch (cf. figure 4.14). Chaque index pointe d'une part, vers les coordonnées 3D du point en question dans le tableaux "Sommets", et d'autre part, vers les coordonnées 3D de la normale à ce point dans le tableau "Normales".

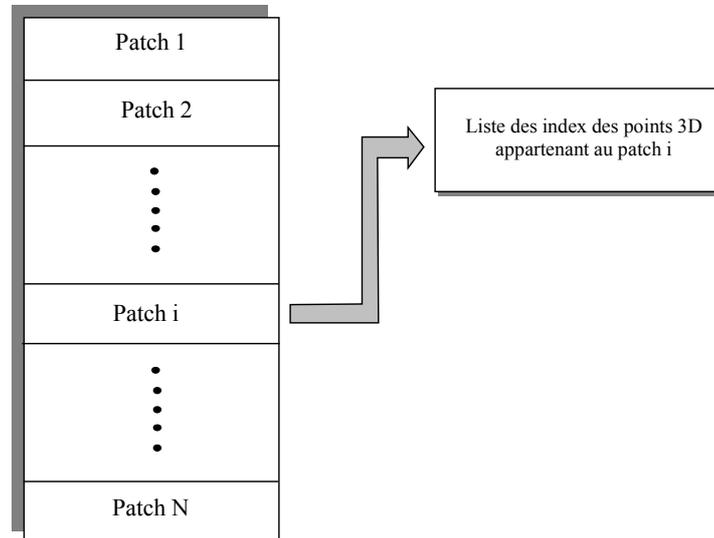


Figure 4.14 : Structure de la table des patches de surface

### 4.3.3 Table d'indexation

Une fois que les patches de surface sont construits, il est impératif de les ordonner dans une structure informatique qui soit simple, rapide d'accès et facilement modifiable par insertion ou suppression de nouveaux patches de surface. Ainsi, l'indexation consiste à ordonner les patches de surface en fournissant des index précalculés, caractérisant chaque patch et qui permettent de l'identifier lors de la phase en ligne. Pour cela, il est important que l'index résiste au bruit et à l'imprécision des données de mesure. Ceci implique directement la gestion des erreurs et des imprécisions. De plus, lors de la création de la table d'indexation, une discrétisation des axes de la table est nécessaire générant ainsi une imprécision supplémentaire. Pour gérer ces sources d'erreurs, nous avons choisi d'enregistrer dans la table d'indexation une seule instance par patch de surface. Par contre, lors de la phase de reconnaissance, la vérification sera effectuée dans les cases voisines de celle où se situe l'index calculé.

La discrimination des patches de surface est assurée par l'utilisation de deux types d'invariants clinométriques, le premier type est calculé à partir de la distribution angulaire des patches de surface, il s'agit en l'occurrence de la moyenne  $m$  et de l'écart type  $\sigma$  de cette distribution qui vont servir comme axe pour la table d'indexation. La moyenne  $m$  nous informe sur l'orientation moyenne du patch de surface par rapport à la normale au point de référence, par contre l'écart type  $\sigma$  nous indique la dispersion des normales aux sommets du patch autour de l'orientation moyenne.

D'une manière plus formelle, nous prenons pour chaque patch, le sommet de référence (qui correspond au sommet central du patch) puis nous calculons les angles  $\theta_i$  entre la normale à ce sommet de référence et les normales aux autres sommets du patch en utilisant l'équation suivante:

$$\theta_i = \arccos(\vec{N}_{ref} \cdot \vec{N}_i) \quad i=1, \dots, N_v \quad (4.18)$$

où  $\vec{N}_{ref}$  est la normale au sommet de référence et  $\vec{N}_i$  la normale au sommet  $i$  du patch considéré.

Nous construisons ensuite, à partir de ces angles, la distribution angulaire du patch de surface que nous notons  $D(\theta)$ . Pour cela, nous discrétisons l'axe des angles (de  $0^\circ$  à  $90^\circ$ ) et nous considérons une tolérance angulaire de  $2^\circ$  (choisie expérimentalement). Une fois la distribution angulaire déterminée, nous calculons sa moyenne  $m$  et son écart type  $\sigma$  tels que:

$$m = \frac{1}{N_v-1} \cdot \sum_{i=1}^{N_v-1} \theta_i \cdot D(\theta_i) \quad (4.19)$$

$$\sigma = \sqrt{\left( \frac{1}{N_v-1} \cdot \sum_{i=1}^{N_v-1} \theta_i^2 \cdot D(\theta_i) \right) - m^2} \quad (4.20)$$

Pour chaque patch de surface, nous calculons les courbures moyenne  $H$  et gaussienne  $K$  de son sommet de référence, définies par [Nutbourne et Martine, 1988] :

$$H = \frac{K_1 + K_2}{2} \quad \text{et} \quad K = K_1 \cdot K_2 \quad (4.21)$$

où  $K_1$  et  $K_2$  représentent les courbures principales au point de référence considéré (voir annexe C).

Les fonctions de courbure  $H$  et  $K$  sont deux invariants algébriques de surface, ils sont indépendants de la pose de l'objet. De plus, leur signe permet de classifier la catégorie de surface à laquelle appartient chaque patch. On peut définir ainsi huit types de surface fondamentaux donnés par le tableau suivant et représentés dans la figure (4.15):

	<b>H&lt;0</b>	<b>H=0</b>	<b>H&gt;0</b>
<b>K&lt;0 (hyperbolique)</b>	Col convexe	Minimal	Col concave
<b>K=0</b>	Cylindre convexe	Plan	Cylindre Concave
<b>K&gt;0 (elliptique)</b>	Pic	indéterminé	Vallée

Tableau 4.3 : Caractérisation locale des patches de surface en fonction des courbures moyenne et gaussienne

Ainsi, les courbures moyenne et gaussienne assurent une discrimination supplémentaire des patches de surface dans le cas où la moyenne  $m$  et l'écart type  $\sigma$  donnent plusieurs hypothèses d'appariements entre les patches du modèle et ceux de l'objet de la scène, ceci

permet d'écarter des mises en correspondance incorrectes avant même la phase de vérification d'hypothèses et ainsi de réduire significativement la complexité de la méthode.

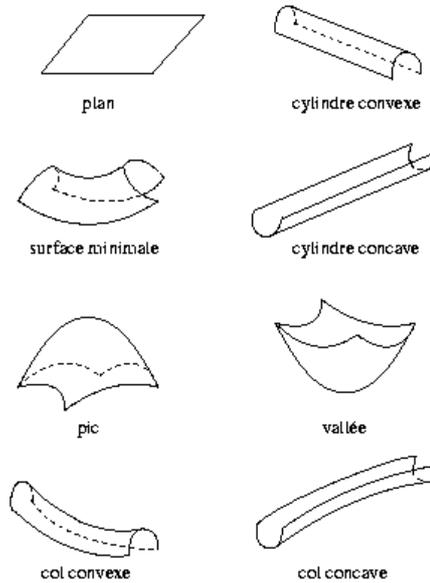


Figure 4.15 : Huit types de surface fondamentaux

Pratiquement, nous utilisons les courbures discrètes [Thirion, 1996] qui sont définies directement à partir du maillage triangulaire des données. Ainsi, la courbure gaussienne discrète  $K$  du sommet de référence du patch de surface est calculée en fonction des angles et des faces connectés à ce sommet (cf. figure 4.16a), elle est donnée par :

$$K = \frac{2\pi - \sum_{i=1}^k \phi_i}{\frac{1}{3}A} \quad (4.22)$$

où  $A = \sum_{i=1}^k S_i$  est la somme des surfaces des faces triangulaires incidentes au sommet de référence, et  $\phi_i$  représentent les angles des faces autour de ce sommet.

De même, la courbure moyenne discrète  $H$  du sommet de référence est donnée par (4.16b):

$$H = \frac{\sum_{i=1}^k m(e_i)}{\frac{1}{3}A} \quad (4.23)$$

où  $e_i$  est une arête incidente du sommet de référence et  $m(e_i)$  est la fonction de courbure définie par :

$$m(e_i) = \begin{cases} \gamma & \text{si } e_i \text{ est convexe} \\ 0 & \text{si } e_i \text{ est plan} \\ -\gamma & \text{si } e_i \text{ est concave} \end{cases} \quad (4.24)$$

où  $\gamma$  est l'angle entre les normales des deux faces adjacentes qui forment l'arête  $e_i$  (cf figure 4.16b).

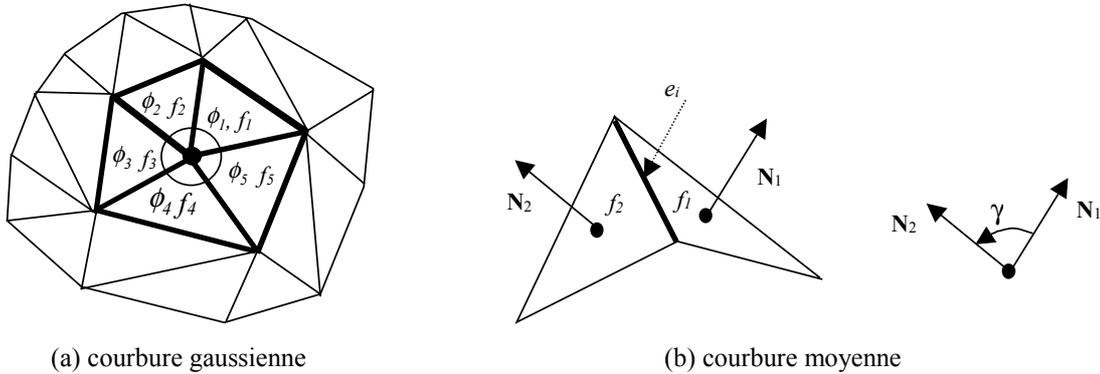


Figure 4.16 : Définition des courbures gaussienne et moyenne discrètes

Nous créons ainsi notre table d'indexation qui est semblable à une table de hachage géométrique [Lamdan et Wolfson, 1988] (cf. figure 4.17). Cette table est repérée respectivement en abscisse et en ordonnée par la moyenne  $m$  et l'écart type  $\sigma$  de la distribution angulaire  $D(\theta)$ . Pour cela, nous avons effectué un échantillonnage adéquat de  $m$  et de  $\sigma$  ( $1^\circ$  pour  $m$  et de  $0.5^\circ$  pour  $\sigma$ , voir § 4.7.3). Dans chaque case de cette table, nous stockons, d'une part, les différents patches de surface du modèle qui sont associés aux valeurs des index  $m$  et  $\sigma$ , et d'autre part, le signe des courbure  $K$  et  $H$  de chacun de ces patches. Ainsi, chaque case de cette table va contenir une ou plusieurs entrées du type : (patch,  $sign(K)$ ,  $sign(H)$ ), ou patch désigne l'index du patch de surface dans la table des patches. Nous verrons que cette table d'indexation est très rapide d'accès grâce à ses deux index  $m$  et  $\sigma$ .

$\sigma \backslash m$	0	1	.....	20	21	.....	90
0.0							
0.5							
1.0				[patch i, sign( $K_i$ ), sign( $H_i$ )]			
⋮							
30.0					[patch l, sign( $K_l$ ), sign( $H_l$ )]		
30.5		[patch j, sign( $K_j$ ), sign( $H_j$ )]					
⋮							
90							

Figure 4.17 : Table d'indexation

### 4.3.4 Résultats expérimentaux

Dans les résultats expérimentaux que nous présentons, nous avons considéré deux zones locales du masque de lion, en l'occurrence la région de l'oreille droite (région N°1) et celle du nez (région N°2) de la figure (4.18) qui représente le masque de lion dans sa position initiale et dans laquelle l'objet à été numérisé en partie.

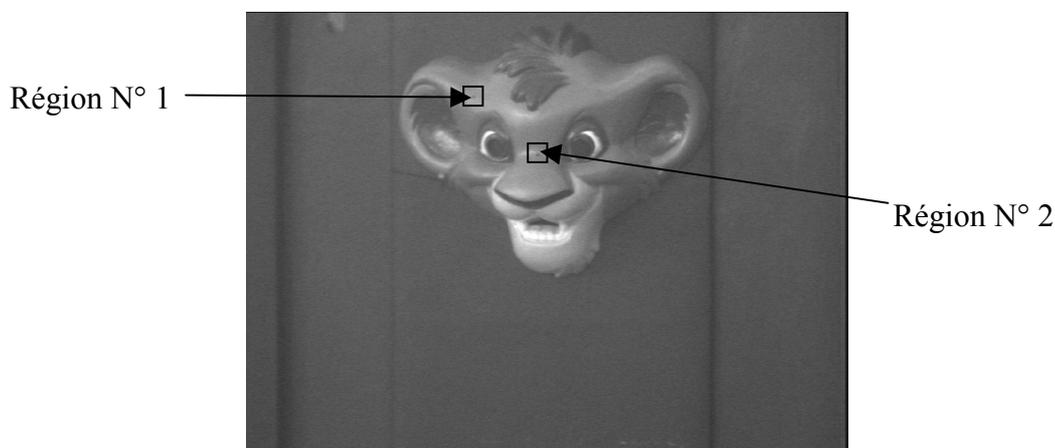
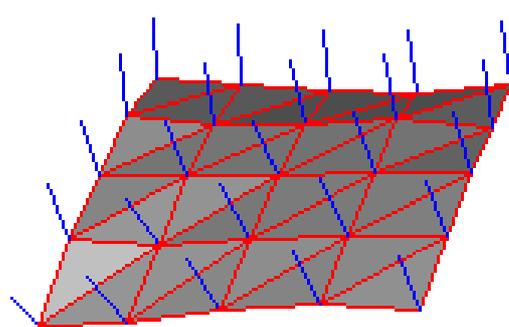
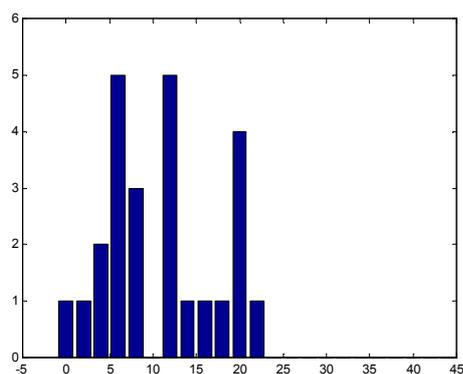


Figure 4.18 : exemples de construction de patch de surface

La figure (4.19a) illustre le patch de surface N°1 construit à partir d'un ensemble structuré de 25 points 3D déterminés grâce à notre système de vision. Ainsi, nous avons représenté ce patch sous forme d'un maillage triangulaire sur lequel nous avons indiqué les normales aux sommets. D'un autre côté, la figure (4.19b) illustre la distribution angulaire associée à ce patch de surface. De même, les figures (4.19c) et (4.19d) illustrent le patch de surface N° 2 et sa distribution angulaire.



(a) Patch N°1



(b) Distribution angulaire

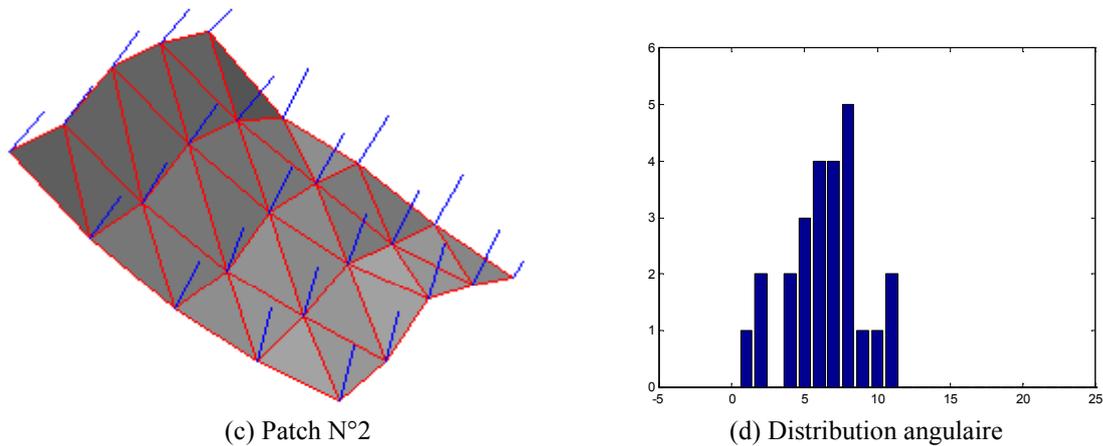


Figure 4.19 : Exemples de patch de surface appartenant au modèle du masque

Le tableau (4.4) présente les résultats obtenus pour les deux patches de surface. Ainsi, pour chaque patch, nous donnons d'une part, les valeurs des index  $m$  et  $\sigma$  qui définissent l'emplacement de la case correspondante dans la table d'index, et d'autre part, les signes des courbures gaussienne et moyenne respectivement.

	$m$	$\sigma$	Sign(K)	Sign(H)	N° de la case
Patch N°1	11	6.0	+	-	(11;12)
Patch N°2	26	10.0	+	+	(26;20)

Tableau 4.4 : Résultats expérimentaux obtenus

#### 4.4 Mise en correspondance

Lorsque l'objet de la scène subit une transformation rigide quelconque l'amenant de sa pose initiale à sa pose courante, le processus de mise en correspondance, qui se déroule en ligne, doit être capable, d'identifier avec précision les patches de surface extraits de l'objet dans sa pose courante avec ceux du modèle. Le résultat de cette étape est très important car si il est correct alors le temps d'exécution de l'algorithme de recalage se trouve énormément réduit.

Pour extraire les patches de surface de l'objet dans sa nouvelle pose, nous utilisons notre capteur de vision stéréo actif. Nous allons démontrer dans la prochaine section que deux appariements de patches objet/modèle suffisent pour déterminer la transformation rigide qu'a subi l'objet. Ainsi, nous procédons à l'extraction de deux patches de surface de l'objet de la même manière que celle expliquée lors de la construction du modèle de l'objet. Dans la mesure du possible, nous choisissons de reconstruire les régions à forte courbure car ces dernières sont facilement identifiables dans la table d'indexation. En effet, le nombre de régions de ce type sur un objet de forme libre est très réduit par rapport au nombre de régions semi-planes, de plus ces régions sont caractérisées par un écart type des distributions angulaires assez important qui traduit une dispersion caractéristique des normales du patch autour de la normale moyenne. Ainsi, les cases de la table d'indexation

correspondantes à ces régions vont contenir un nombre très limité d'entrées voire une seule entrée, ce qui permet de réduire énormément les hypothèses d'appariement.

Une fois les deux patches de surface de l'objet reconstruits, nous déterminons leur distributions angulaires ainsi que leurs courbures gaussienne et moyenne. A partir de ces distributions angulaires, nous calculons les index  $(m, \sigma)$  de chaque patch reconstruit. Les index calculés nous permettent d'accéder directement à la table d'indexation et d'identifier les patches reconstruits avec un ou plusieurs patches du modèle. En pratique, pour gérer les erreurs dues à l'échantillonnage des valeurs des index  $(m, \sigma)$ , la vérification sera aussi effectuée dans les cases voisines de celle où se situe l'index calculé.

L'étape suivante consiste à générer les hypothèses de mise en correspondance à partir du résultat d'indexation. Chaque hypothèse sera constituée de deux appariements de patch objet/modèle parmi tous les appariements possibles. La re-projection des patches de surface du modèle considérés dans l'image de luminance en utilisant la transformation optimale trouvée nous permet de vérifier la validité de l'hypothèse en cours. En effet, les points image correspondant aux points 3D de chaque patch sont connus, car calculés auparavant par le système de vision. Il est clair que, pour la méthode clinométrique comme pour la méthode photoclinométrique, la vérification d'hypothèses fait appel au module de localisation 3D (voir § 4.5) qui permet, à partir des appariements trouvés, de déterminer la transformation rigide correspondante, ce module sera détaillé dans la section suivante.

Pour tester, la validité de la méthode de mise en correspondance, nous avons choisi de reconstruire les mêmes régions du masque, en l'occurrence les régions N°1 et N°2, après que ce dernier ait subi une transformation rigide (cf. figure 4.20).

Ainsi, les figures (4.21a) et (4.21c) illustrent les résultats de la construction des patches de surface correspondant aux régions N°1 et N°2 de l'objet dans sa nouvelle pose, que nous avons appelés patch\_bis N°1 et patch\_bis N°2, respectivement. Les figures (4.21b) et (4.21d) montrent, quant à elles, les répartitions angulaires des deux patches ainsi reconstruits.

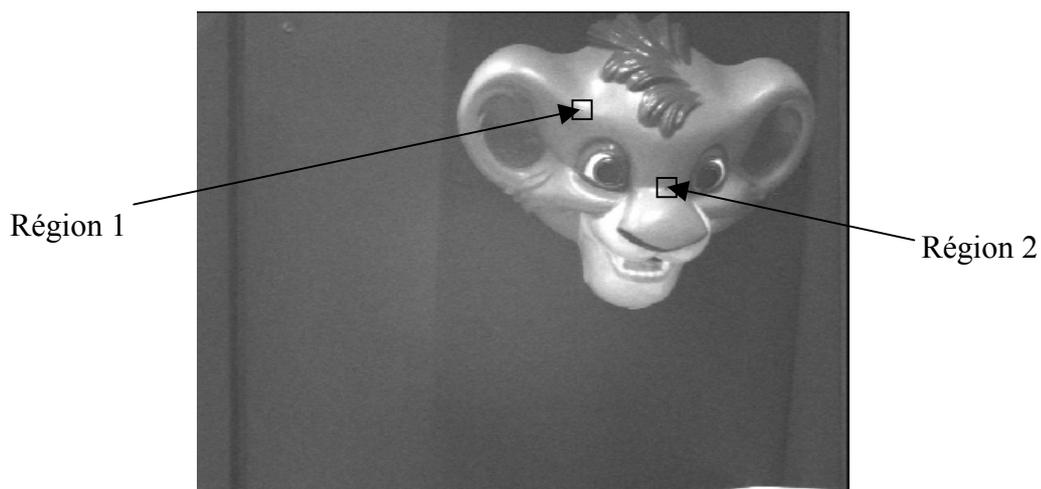


Figure 4.20 : L'objet de la scène après qu'il ait subi une transformation rigide

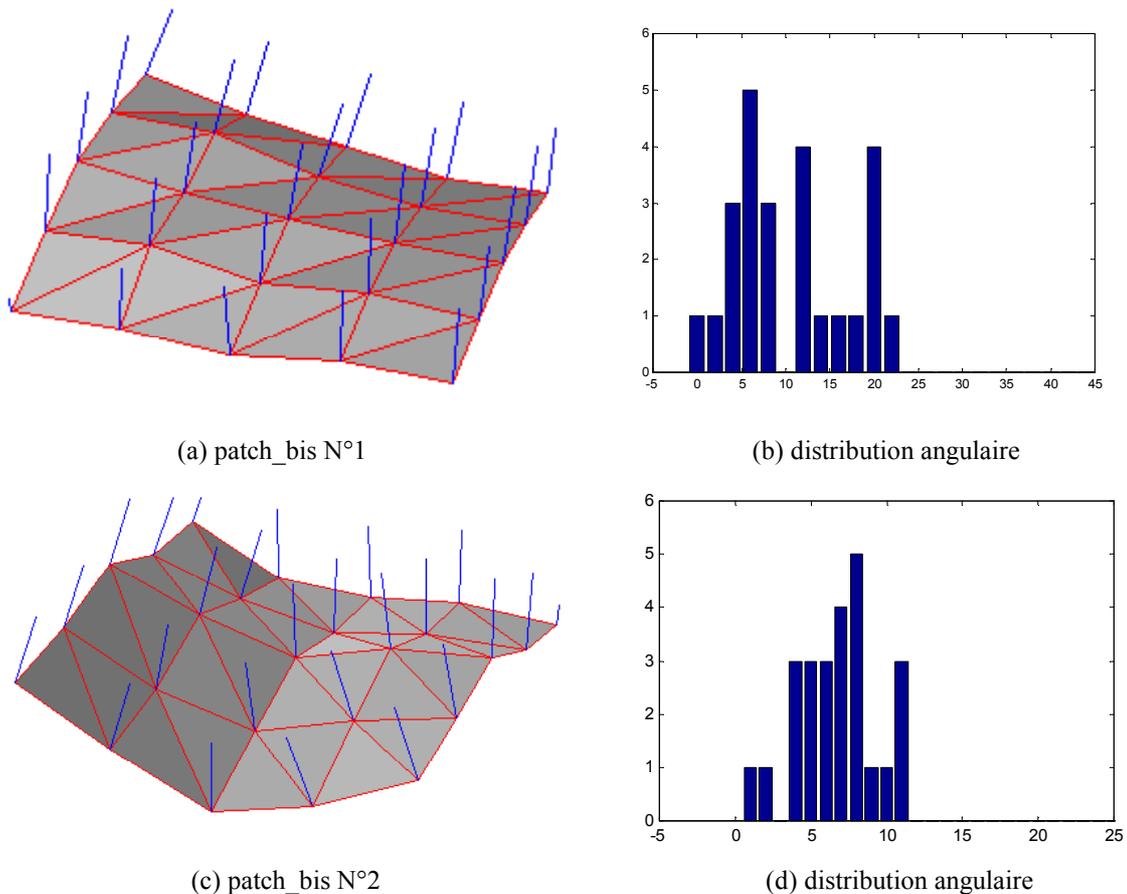


Figure 4.21 : Reconstruction des régions N°1 et N°2 de l'objet dans sa nouvelle pose

Le tableau (4.5) présente les résultats obtenus pour les deux patches bis de surface. Nous remarquons, que le calcul des index  $m$  et  $\sigma$  pour le patch\_bis N°1 donne les mêmes valeurs que celles obtenues pour le patch N°1 (voir tableau 4.4), ce qui implique que l'appariement de ces deux patches est immédiat, par contre, le calcul des index  $m$  et  $\sigma$  pour le patch\_bis N°2 donne des valeurs qui pointent vers une case voisine de celle où se trouve le patch N°2, et donc c'est grâce à la vérification des cases voisines que l'appariement entre les deux patches a pu être établi. De plus, les courbures moyenne et gaussienne des deux patches bis ont les mêmes signes que celles des deux patches du modèle, ce qui, dans ce cas, confirme la mise en correspondance.

	$m$	$\sigma$	Sign(K)	Sign(H)	N° de la case
<b>Patch bis N°1</b>	11	6.0	+	-	(11;12)
<b>Patch bis N°2</b>	27	10.5	+	+	(27;21)

Tableau 4.5 : Résultats obtenus pour les patch\_bis N°1 et N°2

La mise en correspondance des patches de l'objet avec ceux de son modèle constitue le résultat de l'appariement automatique 3D/3D recherché. Ces informations seront utilisées

par l'algorithme de localisation 3D afin de déterminer la position courante de l'objet dans la scène.

## 4.5 Localisation 3D

Cette section a pour but de détailler les formalismes mathématiques utilisés pour élaborer les deux algorithmes de recalage implantés dans l'approche clinométrique, en l'occurrence : le recalage 3D et le recalage 3D fin.

### 4.5.1 Recalage 3D

Le but de ce module est de déterminer, à partir d'une hypothèse de mise en correspondance générée par le module d'appariement 3D/3D décrit ci-dessus, la transformation rigide (translation  $T$  + rotation  $R$ ) qu'a subi l'objet pour apparaître dans sa nouvelle pose (cf. figure 4.22).

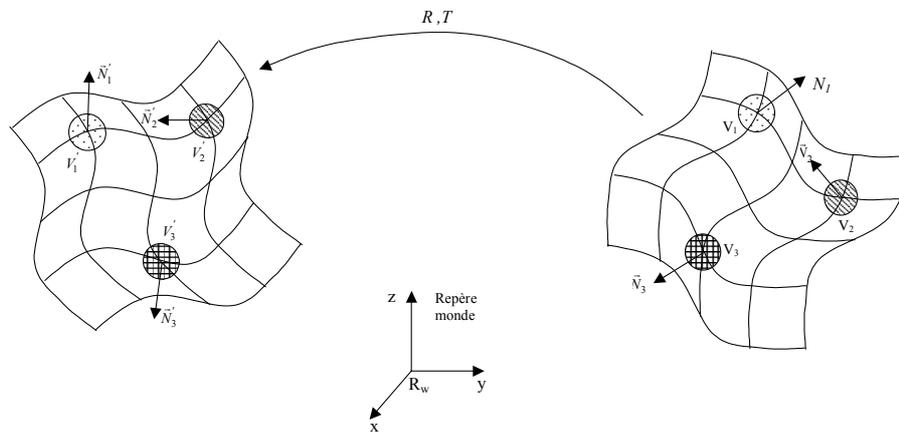


Figure 4.22 : Recalage 3D

Chaque appariement entre un patch de surface de l'objet, lorsque celui-ci est dans sa pose courante, et un patch du modèle permet d'identifier le sommet de référence du patch ainsi que sa normale. Soient  $V_i(x_i, y_i, z_i)$  les coordonnées 3D dans le repère monde du sommet de référence  $V_i$  (sommet du modèle) lorsque l'objet se trouve dans sa pose initiale (nous avons supposé, sans perte de généralité, que le repère de l'objet est confondu avec le repère monde à l'instant initial), et soient  $V'_i(x'_i, y'_i, z'_i)$  les coordonnées 3D dans le repère monde obtenues par notre système de vision stéréo actif du sommet de référence  $V'_i$  (sommet de l'objet) qui correspond au sommet  $V_i$  après que l'objet ait subi la transformation rigide. De même, soient  $\vec{N}_i(N_{xi} \ N_{yi} \ N_{zi})^T$  et  $\vec{N}'_i(N'_{xi} \ N'_{yi} \ N'_{zi})^T$  les coordonnées 3D dans le repère monde des normales aux sommets  $V_i$  et  $V'_i$ , respectivement.

Nous pouvons donc écrire pour tout  $i$  (soit pour toute mise en correspondance  $V_i \leftrightarrow V'_i$ ) :

$$V'_i = R \times V_i + T \quad (4.25)$$

En raison du bruit de mesure il n'y aura pas de superposition parfaite entre les points du modèle et les points mesurés sur l'objet. La distance entre un point du modèle transformé et le point correspondant de l'objet dans sa nouvelle pose est donné par:

$$d_i = \|V'_i - R \times V_i - T\| \quad (4.26)$$

En sommant les carrés de ces distances nous obtenons un critère quadratique :

$$Q = \sum_{i=1}^n \|V'_i - R \times V_i - T\|^2 \quad (4.27)$$

Le problème de localisation 3D peut maintenant être défini de la façon suivante : sous l'hypothèse d'une mise en correspondance terme à terme de  $n$  points, trouver la transformation rigide qui minimise le critère fourni par l'équation (4.27).

Parallèlement, la relation qui lie les normales aux sommets de référence est donnée par :

$$\vec{N}'_i = R \times \vec{N}_i \quad (4.28)$$

L'équation (4.28) est très intéressante car elle permet de séparer la rotation  $R$  de la translation  $T$ , et donc le critère à minimiser peut se décomposer en deux critères [Horaud et Monga, 1993] :

$$\min_R \left( \sum_{i=1}^n \|\vec{N}'_i - R \cdot \vec{N}_i\|^2 \right) \quad (4.29)$$

$$\min_T \left( \sum_{i=1}^n \|V'_i - R_{opt} \times V_i - T\|^2 \right) \quad (4.30)$$

tel que dans le critère (4.30),  $R_{opt}$  correspond à la rotation optimale trouvée grâce au critère (4.29).

L'avantage de cette décomposition est que chaque critère est plus simple et, comme nous allons le voir, la solution au problème d'optimisation de la rotation est très élégante et très fiable d'un point de vue numérique. Le critère de translation se ramène à un problème d'optimisation linéaire classique.

#### 4.5.1.1 Rotation optimale

La minimisation du critère (4.29) dépend du choix d'une représentation pour la rotation  $R$ , nous avons opté pour la représentation par axe et angle de rotation  $(\vec{n}, \theta)$ , car dans ce cas nous aurons à estimer 3 paramètres uniquement, soit les composantes du vecteur directeur de l'axe de rotation  $\vec{n}$ , la valeur de l'angle  $\theta$  étant proportionnelle au module de ce vecteur (voir annexe D). Il s'agit d'une représentation intéressante car minimale en ce qui concerne le nombre de paramètres à estimer. Le critère de l'équation (4.29) signifie que nous cherchons une solution approchée d'un système de  $n$  équations où chaque équation du système est semblable à l'équation (4.28).

Premièrement, nous utilisons le fait que la rotation conserve le produit scalaire:

$$\vec{n} \cdot \vec{N}_i = (R \cdot \vec{n}) \cdot (R \cdot \vec{N}_i) \quad (4.31)$$

or le vecteur  $\vec{n}$  est un vecteur propre de la matrice  $R$  associé à la valeur propre 1 (voir annexe D), ainsi l'équation (4.31) peut se réécrire:

$$\vec{n} \cdot \vec{N}_i = \vec{n} \cdot \vec{N}'_i \quad (4.32)$$

d'où on déduit la propriété de perpendicularité suivante:

$$\vec{n} \cdot (\vec{N}_i - \vec{N}'_i) = 0 \quad (4.33)$$

Par ailleurs le module d'un vecteur est conservé par rotation, on a donc pour tout  $i$ :

$$\|\vec{N}_i\| = \|\vec{N}'_i\| \quad (4.34)$$

On déduit facilement une deuxième propriété de perpendicularité :

$$(\vec{N}_i + \vec{N}'_i) \cdot (\vec{N}_i - \vec{N}'_i) = 0 \quad (4.35)$$

Puisque le vecteur  $(\vec{N}_i - \vec{N}'_i)$  est perpendiculaire aux vecteurs  $\vec{n}$  et  $(\vec{N}_i + \vec{N}'_i)$  il est donc proportionnel à leur produit vectoriel :

$$(\vec{N}_i + \vec{N}'_i) \times \vec{n} = k (\vec{N}_i - \vec{N}'_i) \quad (4.36)$$

Afin de déterminer la valeur de  $k$ , nous allons décomposer les vecteurs  $\vec{N}_i$  et  $\vec{N}'_i$  selon deux directions, une direction colinéaire à l'axe de rotation  $\vec{n}$  et une autre direction orthogonale à cet axe de rotation (cf. figure 4.23) :

$$\vec{N}_i = \vec{N}_{i\perp} + \vec{N}_{i\parallel} \quad (4.37)$$

$$\vec{N}'_i = \vec{N}'_{i\perp} + \vec{N}'_{i\parallel} \quad (4.38)$$

En substituant ces formules dans l'équation (4.36), nous obtenons:

$$(\vec{N}'_{i\perp} + \vec{N}'_{i\parallel} + \vec{N}_{i\perp} + \vec{N}_{i\parallel}) \times \vec{n} = k (\vec{N}'_{i\perp} + \vec{N}'_{i\parallel} - \vec{N}_{i\perp} - \vec{N}_{i\parallel}) \quad (4.39)$$

En développant et en remarquant que  $\vec{N}'_{i\parallel} = \vec{N}_{i\parallel}$ , nous obtenons:

$$(\vec{N}'_{i\perp} + \vec{N}_{i\perp}) \times \vec{n} = k (\vec{N}'_{i\perp} - \vec{N}_{i\perp}) \quad (4.40)$$

Nous pouvons remarquer que les vecteurs  $(\vec{N}'_{i\perp} + \vec{N}_{i\perp})$  et  $(\vec{N}'_{i\perp} - \vec{N}_{i\perp})$  sont orthogonaux et par ailleurs ils sont tous les deux orthogonaux au vecteur unitaire  $\vec{n}$ . Nous obtenons donc  $k$  comme le rapport suivant :

$$k = \frac{\|\vec{N}'_{i\perp} + \vec{N}_{i\perp}\|}{\|\vec{N}'_{i\perp} - \vec{N}_{i\perp}\|} \quad (4.41)$$

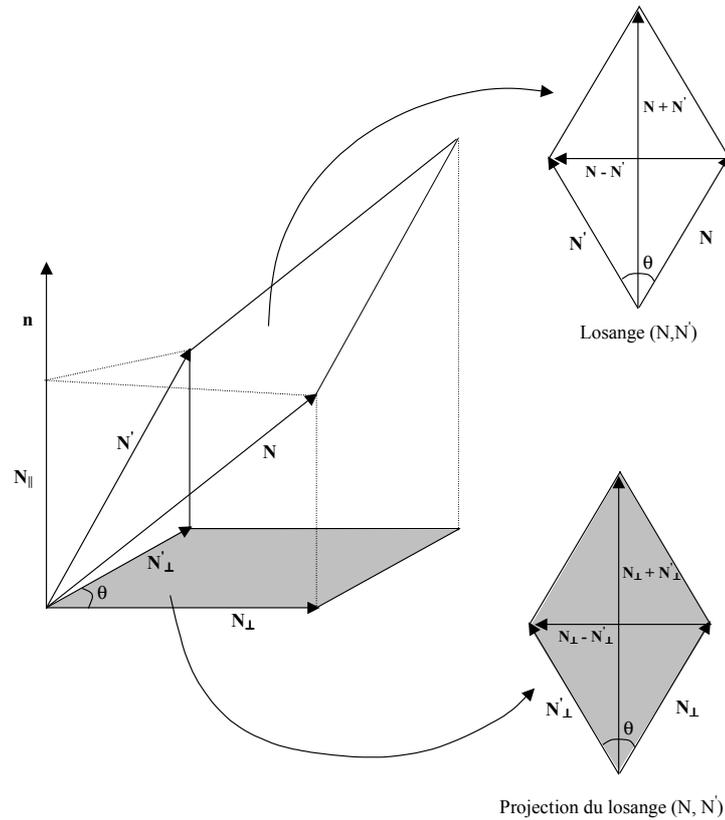


Figure 4.23 : Calcul de la rotation optimale

Or l'angle entre les vecteurs  $\vec{N}'_{i\perp}$  et  $\vec{N}_{i\perp}$  est l'angle de rotation que l'on cherche, soit  $\theta$ . D'après la figure (4.23), nous remarquons qu'il s'agit d'un des angles d'un losange qui possède comme diagonales  $(\vec{N}'_{i\perp} + \vec{N}_{i\perp})$  et  $(\vec{N}'_{i\perp} - \vec{N}_{i\perp})$ . La relation entre cet angle et le rapport des longueurs de ces diagonales est:

$$k = \frac{1}{\tan(\theta/2)} \tag{4.42}$$

En posant :

$$\vec{\Omega} = \tan(\theta/2) \vec{n} \tag{4.43}$$

et remplaçant dans l'équation (4.36) nous obtenons un système de trois équations et trois inconnues pour chaque  $i$ :

$$S(\vec{N}'_i + \vec{N}_i) \vec{\Omega} = \vec{N}'_i - \vec{N}_i \tag{4.44}$$

où  $S(\vec{N}'_i + \vec{N}_i)$  est la matrice antisymétrique associée au vecteur  $(\vec{N}'_i + \vec{N}_i)$  (voir annexe D).

Le rang de la matrice  $S(\ )$  est 2 et chaque mise en correspondance  $i$  fournit donc 2 équations et non pas 3 équations. Il faut donc au moins 2 mises en correspondance pour trouver un axe de rotation. En général, on dispose de  $n$  mises en correspondance. On

obtient alors un système de  $2n$  équations à 3 inconnues. Si l'on considère deux équations du système d'équations (4.44) nous obtiendrons :

$$\begin{pmatrix} \vdots & \vdots & \vdots \\ 0 & -(N'_{zi} + N_{zi}) & N'_{yi} + N_{yi} \\ N'_{zi} + N_{zi} & 0 & -(N'_{xi} + N_{xi}) \\ \vdots & \vdots & \vdots \end{pmatrix} \cdot \begin{pmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{pmatrix} = \begin{pmatrix} \vdots \\ N'_{xi} - N_{xi} \\ N'_{yi} - N_{yi} \\ \vdots \end{pmatrix} \quad (4.45)$$

La solution optimale pour  $\vec{\Omega}$  est obtenue en résolvant ce système d'équations surcontraint. Nous obtenons finalement le vecteur unitaire de l'axe de rotation et l'angle de rotation :

$$\vec{n} = \frac{\vec{\Omega}}{\|\vec{\Omega}\|} \quad (4.46)$$

$$\theta = 2 \cdot \text{atan}\left(\|\vec{\Omega}\|\right) \quad (4.47)$$

et grâce à la formule de Rodrigues (voir annexe D) nous obtenons facilement la matrice de rotation  $R$  à partir du vecteur unitaire  $\vec{n}$  et de l'angle  $\theta$ .

Comme nous l'avons déjà mentionné, la méthode proposée ci-dessus nécessite au moins deux appariements de vecteurs pour trouver la rotation optimale. Ceci explique notre choix de reconstruire uniquement deux patches de surface.

#### 4.5.1.2 Translation optimale

Une fois que nous avons estimé la rotation optimale  $R_{opt}$  le calcul de la translation optimale est possible grâce au critère de l'équation (4.30). La méthode classique consiste à dériver ce critère par rapport aux 3 composantes de  $T$  et à annuler ces dérivées, nous obtenons alors un système de trois équations linéaires en trois inconnues qui se résout sans aucune difficulté. Or comme nous avons établi des appariements de points (les sommets de référence des patches), le calcul de la translation optimale est encore plus simple, elle est donnée par le vecteur qui relie les barycentres des deux ensembles de sommets de référence :

$$T_{opt} = \vec{v}' - R_{opt} \times \vec{v} \quad (4.48)$$

où

$$\vec{v} = \frac{1}{n} \sum_{i=1}^n V_i \quad \text{et} \quad \vec{v}' = \frac{1}{n} \sum_{i=1}^n V'_i \quad (4.49)$$

### 4.5.2 Recalage 3D fin

Lorsque nous avons appliqué la transformation optimale calculée ci-dessus aux patches de surface du modèle et avons projeté les patches ainsi transformés sur l'image de l'objet, nous avons noté une erreur de recalage entre les primitives image (les nœuds exacts) et les sommets de référence projetés dans l'image qui leur correspondent (cf. figure 4.24). Ceci est dû aux erreurs de calculs introduites par les algorithmes de traitement de données utilisés dans les différents modules du système de recalage et qui sont généralement cumulatives.

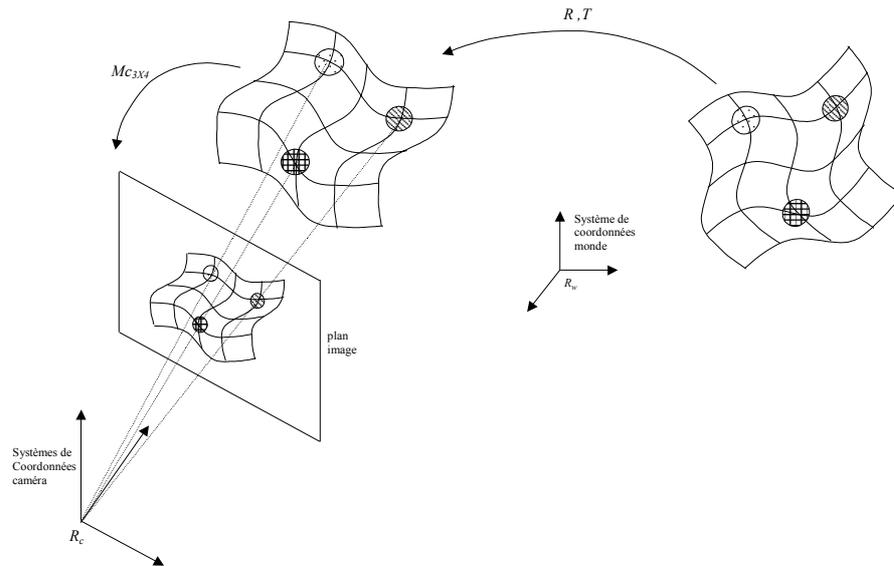


Figure 4.24 : Recalage 3D fin

Ainsi, le but du recalage fin est de déterminer la transformation optimale qui permet de minimiser l'erreur quadratique moyenne de distance entre les points dans l'image correspondant aux centres exacts des patches reconstruits et les projetés dans l'image des sommets de référence du modèle transformé.

Soit  $D$  le critère à minimiser, il est défini par:

$$D = \sum_{i=1}^n \|\hat{m}_i - m_i\|^2 \quad (4.50)$$

où  $\hat{m}_i$  sont les points extraits de l'image qui correspondent aux centres exact des patches, et  $m_i(u_i, v_i)$  sont définis par :

$$\begin{bmatrix} s \cdot u_i \\ s \cdot v_i \\ s \end{bmatrix} = Mc_{3 \times 4} \cdot \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \cdot V_i \quad (4.51)$$

où  $M_{c_{3 \times 4}}$  est la transformation globale entre le plan image et le repère monde déterminée lors de la calibration de la caméra,  $R$  et  $T$  sont la matrice de rotation et le vecteur de translation affinés à déterminer.

Nous avons utilisé la méthode de Levenberg-Marquardt pour résoudre ce problème d'optimisation non linéaire. Le vecteur de paramètres à estimer est donc constitué des éléments de  $R$  et  $T$ . Nous avons choisi de représenter la rotation  $R$  par un quaternion unitaire  $q$ . Cette représentation nous permet d'avoir une solution unique au problème inverse de la rotation  $R$ . Le quaternion unitaire  $q$  est donné par ses 4 composantes comme suit :

$$q = [q_0 \ q_1 \ q_2 \ q_3]^T \quad \text{où} \quad q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \quad (4.52)$$

L'expression de la matrice de rotation  $R$  en fonction du quaternion unitaire  $q$  est donnée par:

$$R = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (4.53)$$

Ainsi, le vecteur de paramètres  $P$  à estimer est composé de 6 paramètres : les trois composantes du vecteur de translations  $T$  et 3 des quatre éléments du quaternion unitaire  $q$ :

$$P = [q_1 \ q_2 \ q_3 \ T_x \ T_y \ T_z]^T \quad (4.54)$$

Comme nous l'avons déjà mentionné (voir § 4.2.2), la méthode de Levenberg-Marquardt nécessite une bonne initialisation du vecteur de paramètres (proche de l'optimum) pour qu'elle puisse converger. Nous utilisons la transformation ( $R_{opt}$ ,  $T_{opt}$ ) obtenue par le module de recalage 3D décrit dans la section précédente pour initialiser le vecteur  $P$ .

### 4.5.3 Résultats expérimentaux

Pour l'exemple considéré, nous avons injecté l'hypothèse de mise en correspondance générée par le module d'appariement au module de recalage 3D afin de déterminer la pose courante de l'objet. La transformation optimale déterminée est la suivante:

Vecteur de translation (mm) T	52.7234	30.6261	110.5023
Matrice de rotation R	0.8821	0.2160	0.4185
	-0.1465	0.9704	-0.1922
	-0.4476	0.1082	0.8876

Tableau 4.6 : Transformation optimale

La figure (4.25a) montre la projection des deux patches du modèle dans l'image de luminance après qu'on leur ait appliqué la transformation optimale trouvée, nous remarquons qu'effectivement les patches projetés dans l'image ne se superposent pas d'une manière exacte sur les régions de l'objet qui leur correspondent.

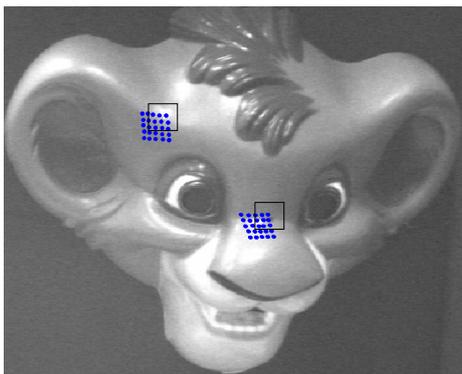
Pour affiner le recalage de l'objet, nous injectons la transformation optimale obtenue ci-dessus au module de recalage 3D fin, nous obtenons la transformation suivante:

Vecteur de translation $T$ (mm):	55.9683	28.0648	107.6215
Matrice de rotation $R$	0.8872	0.1498	0.4927
	0.0454	0.9310	-0.3621
	-0.5130	0.3328	0.7921

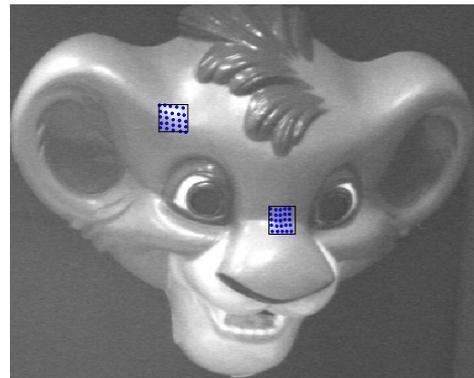
Tableau 4.7 : Transformation affinée

De même, la figure (4.25b) illustre la re-projection des deux patches du modèle dans le plan image en utilisant cette fois-ci la transformation affinée, nous remarquons que dans ce cas ces patches se superposent nettement mieux avec les régions de l'objet qui leur correspondent, ce qui implique une meilleure qualité du recalage 3D. Le tableau (4.8) illustre justement la qualité du recalage 3D, il donne la moyenne  $E_d$  des distances entre les projetés des sommets de référence des patches et les nœuds exacts dans l'image correspondant à ces sommets, pour la transformation optimale et la transformation fine, avec  $E_d$  définie par :

$$E_d = \frac{1}{n} \sum_{i=1}^n \|\hat{m}_i - m_i\| \quad (4.55)$$



(a) re-projection des deux patches du modèle en utilisant la transformation optimale



(b) re-projection des deux patches du modèle en utilisant la transformation affinée

Figure 4.25 : Re-projection des patches du modèle dans le plan image

	Transformation optimale	Transformation fine
Erreur moyenne $E_d$ (pixels)	3.736	$0.274 \times 10^{-2}$

Tableau 4.8 : Erreurs apparentes de recalage

Ainsi, les résultats obtenus démontrent l'amélioration importante qu'apporte le recalage fin aux calculs de la pose courante de l'objet.

## 4.6 Vérification d'hypothèses

La re-projection des patches du modèle dans l'image en utilisant la transformation optimale déterminée nous permet de vérifier la validité de l'hypothèse en cours. Pour cela nous vérifions que chaque centre de patch dans l'image coïncide avec un sommet de référence du patch du modèle projeté. Un sommet de référence est validé si la distance entre son projeté dans l'image et le nœud exact correspondant à un centre de patch dans l'image est inférieur à un seuil donné (5 pixels, voir § 4.7.4). L'hypothèse en cours est retenue si tous les sommets de référence des patches du modèle, de l'hypothèse en cours, sont vérifiés, en l'occurrence 2 sommets dans le cas que nous avons traité.

## 4.7 Evaluation des performances

Comme nous l'avons déjà mentionné dans le chapitre précédent, l'évaluation des performances d'un système de reconnaissance est une étape primordiale. Elle permet, d'une part, de valider le fonctionnement global du système grâce à un ensemble de tests effectués dans des conditions différentes, et d'autre part, d'optimiser les paramètres qui influencent la rapidité et la robustesse du système.

Le but de cette section est d'identifier, parmi tous les paramètres qui entrent en jeu dans la méthode clinométrique de recalage 3D, ceux qui influencent d'une manière directe les performances du système, et montrer quels sont les critères que nous avons définis pour les optimiser.

Ainsi, pour chaque étape constituant le système clinométrique, nous allons présenter les critères de jugement retenus. Les résultats obtenus par application de ces critères sont ensuite analysés puis discutés pour aboutir à un jugement global du système.

### 4.7.1 Système de vision

Le système de vision représente l'organe sensoriel utilisé par le système clinométrique pour percevoir et reconstruire les patches de surface des objets de la scène afin de les recalibrer dans leur environnement de travail. Ainsi, les performances du système clinométrique dépendent en grande partie de la précision de la reconstruction 3D du système de vision. Les paramètres que nous avons optimisés pour aboutir à une précision de reconstruction satisfaisante sont :

- la calibration de la caméra et du projecteur : en effet, la reconstruction 3D dépend directement des éléments des matrices de projection perspective de la caméra et du projecteur. Pour avoir un bon calibrage de ces deux dispositifs, nous devons disposer d'un objet de calibrage fournissant des points dont la position doit être connue avec une très bonne précision dans le repère monde. Nous avons utilisé la grille laser comme mire de calibrage, les points de calibrage sont l'ensemble des intersections des lignes horizontales et verticales de la grille laser projetée sur un plan vertical, et dont la position est connue au 1/10 de millimètre près.

- la détection des projections des nœuds de la grille dans l'image : en effet, le traitement d'image doit être capable de détecter avec précision ces projections, nous avons élaboré une méthode de détection subpixel, cette méthode étant basée sur l'optimisation d'un modèle, nous avons fixé sa précision arbitrairement à 1/100 du pixel.

### 4.7.2 Construction de la base de données

Le critère à optimiser dans cette étape est la complexité en taille de la table des patches de surface du modèle. En effet, nous devons faire en sorte que, d'une part, le nombre de patch construits pour modéliser l'objet soit le plus réduit possible pour ne pas aboutir à une dimension trop prohibitive pour une application concrète, et d'autre part, s'assurer que les patches ainsi construits restent discriminants, c'est à dire caractérisant d'une manière fiable les régions locales du modèle. En pratique, nous avons choisi de numériser quelques régions de l'objet pour mettre en œuvre la solution clinométrique. Les valeurs des pas de balayage angulaire sont choisies de sorte à reconstruire, à chaque fois, un ensemble structuré de points 3D qui, d'une part, est assez dense pour discriminer localement la région numérisée de l'objet, et d'autre part, permet de construire des patches de surface ayant une dimension raisonnable afin de ne pas alourdir le processus de reconnaissance. Après plusieurs tests, nous avons trouvé que les pas de balayage  $\Delta\varphi = \Delta\theta = 0.4^\circ$  permettaient d'obtenir des patches de surface de dimension (5×5) satisfaisants pour notre application.

### 4.7.3 Indexation

L'indexation est une étape délicate car elle doit permettre de retrouver dans la base de données, un patch de surface du modèle sensé être mis en correspondance pour le recalage 3D. De nombreux paramètres dans cette étape sont à optimiser afin d'obtenir des résultats satisfaisants. Le premier concerne l'échantillonnage des index ( $m, \sigma$ ). En effet, des pas d'échantillonnage  $\Delta m$  et  $\Delta\sigma$  trop élevés entraînent une indexation qui n'est pas suffisamment discriminante, les cases de la table d'indexations vont contenir un nombre important d'entrées de patches impliquant ainsi une génération d'un nombre très élevé d'hypothèses lors de la phase en ligne. Ceci va augmenter considérablement le temps d'exécution. D'un autre côté, des pas d'échantillonnage  $\Delta m$  et  $\Delta\sigma$  très petits entraînent, d'une part, une augmentation considérable de la taille de la table d'indexation, et d'autre part, une discrimination trop sélective des patches du modèle qui ne prend pas en compte la gestion des erreurs et des imprécisions des données de mesure, ce qui a pour effet une mauvaise indexation des patches de l'objet construits lors de la phase en ligne. Nous avons donc choisi  $\Delta m = 1^\circ$  et  $\Delta\sigma = 0.5^\circ$ , ces valeurs ont été déterminées expérimentalement, elles constituent le meilleur compromis entre qualité de discrimination et taille de la table d'indexation. De plus, afin de gérer les erreurs de mesures, nous avons choisi de vérifier, lors de la phase de reconnaissance, les cases voisines de celle où se situe les index calculés.

#### 4.7.4 Vérification d'hypothèses

Comme pour la méthode photoclinométrique, la vérification d'hypothèses dans la méthode clinométrique est basée sur la mesure de distance dans l'image (en pixel) entre un centre de patch dans l'image et un sommet de référence du patch du modèle projeté. Ainsi, nous avons suivi la même procédure que celle utilisée dans la méthode photoclinométrique pour déterminer le rayon de vérification qui définit la distance maximale à ne pas dépasser afin de vérifier un appariement entre le centre de patch dans l'image et un sommet de référence projeté. Nous avons retenu dans ce cas aussi, après plusieurs tests, un rayon de 5 pixels qui permet d'avoir une bonne précision de la vérification.

### 4.8 Conclusion

Ce chapitre décrit la méthode clinométrique de recalage 3D des objets de forme libre que nous avons élaborée. Nous avons démontré que l'exploitation des normales à la surface d'un objet de forme libre permet d'extraire des invariants géométriques locaux qui sont utilisés pour établir des appariements 3D/3D entre des régions de l'objet de la scène et celles du modèle de la base de données, il s'agit en l'occurrence de la distribution angulaire ainsi que des courbures gaussienne et moyenne. La mise en œuvre de la solution clinométrique a nécessité la conception et la réalisation d'un système de vision constitué d'une caméra CCD combinée avec un projecteur de grille laser. Nous avons développé une méthode de détection subpixel des primitives (nœuds de la grille projetée) dans l'image afin d'assurer une bonne précision de reconstruction des points 3D, les différents tests expérimentaux ont démontré une précision  $< 1\%$  sur la reconstruction. Par ailleurs, Nous avons montré comment construire, à partir d'un ensemble structuré de points 3D, des patches de surface pour modéliser et caractériser les régions locales de l'objet. Nous avons choisi la méthode d'indexation pour stocker dans une structure simple et rapidement accessible les différents patches du modèle ainsi construits. Nous avons utilisé les invariants clinométriques, en l'occurrence la moyenne et l'écart type associés à la distribution angulaire de chaque patch du modèle, pour indexer la base de données. Les résultats obtenus pendant l'étape en ligne sont intéressants car, d'une part, ils démontrent un temps d'exécution rapide de part l'indexation, et d'autre part, une mise en correspondance réussie entre les patches de surface de l'objet et ceux du modèle. Nous avons élaboré deux algorithmes de localisation 3D, le premier est l'algorithme de recalage 3D, il prend en entrée les appariements de patches générés par le module d'appariement et donne en sortie la transformation rigide subie par l'objet, par contre le second algorithme permet d'affiner la transformation précédente. Enfin, une évaluation des performances du système nous a permis d'identifier puis d'optimiser les paramètres qui influencent directement les résultats de la méthode clinométrique.

Notre système clinométrique possède la faculté de recalcr un objet de forme libre même si ce dernier est partiellement occlus grâce à la localité de l'invariant utilisé. En effet, notre système de vision nous permet toujours d'extraire des patches de surface qui font partie des zones visibles de l'objet. Cependant, malgré les résultats satisfaisants obtenus, il paraît intéressant de pouvoir établir une évaluation plus poussée de l'efficacité de ce système, qui passe nécessairement par la numérisation complète de l'objet de forme libre. L'ensemble

structuré de points 3D obtenu par cette numérisation va permettre de construire une table d'indexation plus riche qui va contenir la totalité des patches de surface caractérisant les régions locales de l'objet, augmentant ainsi l'efficacité de la mise en correspondance.

## Conclusion Générale

Nous avons abordé dans ce mémoire un problème important en vision, celui du recalage 3D automatique d'objets de forme libre. Ce problème est d'autant plus important qu'il permet d'apporter une aide à la commande et à la perception d'une scène complexe, à un opérateur en situation de téléopération ou de supervision de tâches, en cas de vision indirecte.

Notre objectif consistait donc, d'une part, à démontrer que l'utilisation de la clinométrie des objets de forme libre suffisait pour les recalculer dans leur environnement de travail, et d'autre part, à mettre en œuvre des solutions efficaces de recalage 3D de ce type d'objets basés uniquement sur la clinométrie.

Afin de résoudre ce problème, nous nous sommes attachés au schéma traditionnel de la reconnaissance d'objets en vision. Ce schéma est composé de trois grandes parties respectivement désignées de bas niveau (extraction de primitives), moyen niveau (construction de la base de données du modèle) et haut niveau (appariement et recalage). Chacune de ces parties est elle-même le fruit de traitements successifs qui nécessitent la plus grande rigueur du fait de leur interaction dans le système de reconnaissance global.

Nous avons élaboré deux approches clinométriques qui respectent ce schéma mais dont les modules diffèrent en fonction du type d'images utilisé en entrée. Dans l'approche photoclinométrique, nous avons démontré que, sous certaines conditions, la photoclinométrie permet d'extraire à partir d'une seule image de luminance d'un objet de forme libre un invariant projectif de type photométrique (non géométrique) permettant ainsi de reconnaître les points du modèle vus dans l'image, il s'agit en l'occurrence des répartitions angulaires. Nous avons montré comment exploiter cet invariant pour construire, hors ligne, une base de données qui répertorie d'une manière simple et efficace les informations pertinentes sur le modèle permettant ainsi de l'apparier avec les primitives image (les barycentres des taches de forte intensité lumineuse). Afin de démontrer l'efficacité de cette méthode, nous l'avons testé sur une série d'images synthétiques représentant le modèle dans des poses différentes. Les résultats obtenus sont très satisfaisants et nous confortent dans notre approche. Cela dit, la mise en œuvre pratique de la solution photoclinométrique reste difficile à réaliser. En effet, l'hypothèse de base qui impose que la source de lumière illuminant la scène soit directionnelle est trop contraignante et difficile à mettre en œuvre, ce qui nous a amené à travailler sur des images de synthèse. Ainsi, les conclusions établies à partir de ces images possèdent une portée plus ou moins limitée.

Afin de remédier aux défauts de l'approche photoclinométrique (conditions de mise en œuvre contraignantes), nous avons élaboré la solution clinométrique. En effet, cette solution ne dépend pas des conditions d'éclairage de la scène. Dans l'approche clinométrique, nous avons démontré que l'exploitation des normales à la surface d'un objet de forme libre permet de construire des invariants géométriques (à la rotation et à la translation) locaux qui peuvent être utilisés pour établir des appariements 3D/3D entre des régions de l'objet de la scène et celles du modèle. Il s'agit en l'occurrence de la distribution angulaire ainsi que des courbures gaussienne et moyenne. Pour mettre en œuvre la solution clinométrique, nous avons mis en œuvre un système de vision constitué d'un capteur stéréo actif (caméra CCD + projecteur de grille laser). Nous avons développé une méthode de détection subpixel des primitives dans l'image afin d'assurer une bonne précision de reconstruction des points 3D (précision <1%). Nous avons choisi la méthode d'indexation pour stocker dans une structure simple et rapidement accessible les patches de surface du modèle. L'originalité, dans notre application, réside dans la discrimination entre les différentes régions de surface d'un seul modèle grâce aux invariants clinométriques, ceci a une implication directe dans l'efficacité de la mise en correspondance. En effet, le calcul des invariants clinométriques permet des appariements robustes et rapides entre les régions de l'objet de la scène et celles du modèle de la base de données. Ceci simplifie le processus de recalage et le rend fiable. Nous avons testé l'approche clinométrique sur un objet réel de forme libre, en l'occurrence un masque de lion, les résultats obtenus pour les tests effectués sont très probants.

Ainsi, nous avons réussi à développer deux méthodes de recalage 3D automatique d'objets de forme libre en utilisant uniquement la clinométrie. Au travers de ces travaux, nous avons abordé plusieurs niveaux de la vision en cherchant à exploiter les ressources existantes pour la calibration, la segmentation et la mise en correspondance. Notre contribution essentielle a été apportée dans l'organisation perceptive des primitives de bas niveaux (la construction des aspects photométriques à partir des barycentres des taches de forte intensité lumineuse pour l'approche photoclinométrique, et la construction des patches de surface à partir des ensembles structurés de points 3D pour l'approche clinométrique), l'indexation et la mise en correspondance. L'avantage des méthodes élaborées repose sur leur modularité qui permet de pouvoir réutiliser les différentes parties dans d'autres applications de vision. Finalement, nous avons proposé une caractérisation de performances, des modules essentiels de chaque méthode pour évaluer leur efficacité ainsi que leurs limites.

Parmi les perspectives de nos travaux on peut citer les points suivants :

- Amélioration des méthodes élaborées : pour cela nous devons améliorer les traitements effectués dans les trois niveaux du système de recalage. Ainsi, pour la partie bas niveau de l'approche photoclinométrique, l'extraction de primitives de types géométriques (telle que la silhouette de l'objet) et leur combinaison avec les primitives photométriques ne peut qu'apporter à l'efficacité de la reconnaissance. En effet, une telle combinaison nous permettra de construire des invariants projectifs locaux qui assureront une meilleure mise en correspondance en présence d'occultations. De même, pour l'approche clinométrique, nous envisageons d'utiliser des portions de courbes 3D

de l'objet afin de construire des patches de surface plus discriminants, car plus riches en informations. Ceci permettra d'améliorer la robustesse de la méthode. Dans la partie moyen niveau, pour l'approche photoclino-métrique, la mise en œuvre d'une méthode de classification des taches permettra d'accélérer leur processus de reconnaissance. En effet, actuellement les taches sont comparées une à une ce qui rend le processus d'appariement assez lent. Enfin, dans la partie haut niveau des deux approches, nous pensons améliorer l'indexation de façon à pouvoir obtenir des appariements objet/modèle encore plus pertinents, pour cela nous envisageons l'utilisation d'information supplémentaires telle que la couleur.

- Afin de pallier les difficultés de mise en œuvre de l'approche photoclino-métrique avec des conditions d'éclairage réel, nous envisageons d'intégrer cette approche dans une application de réalité mixée. En effet, notre méthode s'adapte parfaitement à ce type d'applications où il s'agit généralement d'insérer ou de recalibrer des objets réels dans une scène complètement virtuelle dont l'éclairage est contrôlable par l'opérateur.
- Nous envisageons d'étendre l'utilisation des méthodes élaborées à la reconnaissance d'objets multiples dans une scène. En effet, les invariants que nous avons construits peuvent être utilisés pour indexer dans une table de hachage, non plus des régions particulières d'un même objet, mais des modèles d'objets différents.
- Une suite logique de nos travaux concerne le recalibrage dynamique d'objets 3D de forme libre (suivi d'objets dans une séquence d'images) en vision indirecte. Le domaine d'applications de telles techniques inclut la surveillance, la commande de robots mobile et/ou manipulateur et la réalité augmentée.
- Enfin, la réalisation d'un système de recalibrage 3D hybride qui combine les deux approches développées, en l'occurrence l'approche photoclino-métrique et l'approche clinométrique, constituerait une finalité de nos travaux. En effet, grâce à la richesse des informations utilisées, ce système augmentera les performances du recalibrage 3D des objets de forme libre dans des scènes complexes.



# Bibliographie

- [Ababsa et al., 2002a] F. Ababsa, D. Roussel, M. Mallem, J. Y. Didier (2002). 2D/3D Automatic Matching Technique for 3D Recovering of Free Form Objects. *Proceeding of the 16<sup>th</sup> IAPR International Conference on Pattern Recognition (ICPR 2002)*. Quebec City, Canada, August 11-15, 2002. IEEE Computer Society Press.
- [Ababsa et al., 2002b] F. Ababsa, D. Roussel, M. Mallem, J. Y. Didier (2002). 3D free form objects recognition technique using photoclinometry. *Proceedings of the 4<sup>th</sup> International Workshop on Advanced Concepts for Intelligent Vision Systems (ACIVS 2002)*. Ghent, Belgium, September 9-11, pp. 130-135.
- [Ababsa et al., 2002c] F. Ababsa, D. Roussel et M. Mallem (2002). 3D Free Form Object Recovery Using An Accurate 3D Low Cost Vision System Based on Structured Light. *Proceeding of The 7<sup>th</sup> International Conference on Control, Automation, Robotics and Vision (ICARCV 2002)*. 2-5 December 2002. Singapore
- [Ababsa et al., 2002d] F. Ababsa, D. Roussel et M. Mallem (2002). An Optimal Low Cost Solution For The 3D Free Form Object Recovering Pose Problem. *Proceeding of the 8<sup>th</sup> IAPR International Workshop on Machine Vision Applications (MVA2002)*. 11-13 December 2002. Japan
- [Arun et al., 1987] K. S. Arun, T. S. Huang et S. D. Blostein (1987). Least-square fitting of two 3D points sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9:698-700.
- [Aloimonos, 1986] Y. Aloimonos (1986). Detection of surface orientation from texture in the case of planes. *IEEE Conf. On Computer Vision and Pattern Recognition*, pages 584-593.
- [Aström, 1993] K. Aström (1993). Affine invariants of planar sets. TR, Department of Mathematics, Lund Institute of Technology, Sweden.
- [Ayache et Faugeras, 1986] N. Ayache et O. D. Faugeras (1986). HYPER : a new approach for the recognition and positioning two dimensional objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):44-54.
- [Bajaj et al., 1995] C. L. Bajaj, J. Chen et G. Xu (1995). Modeling with cubic A-patches. *ACM Transactions on Graphics*, 14(2):103-133.
- [Bak, 1991] J. Bak (1991). Three dimensional object recognition and its application to biological imaging. *The 7<sup>th</sup> Scandinavian Conference on Image Analysis*, vol. 2, pages:580-587, Aalborg, Danemark.

- [Ballard et Brown, 1982] D. Ballard et C. Brown (1982). Computer vision. Prentice-Hall.
- [Barequet et Sharir, 1997] G. Barequet et M. Sharir (1997). Partial surface and volume matching in three dimensions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(9):929-948.
- [Basri, 1993] R. Basri (1993). Recognition by prototypes. *IEEE Conference on Computer Vision and Pattern Recognition*, pages: 164-167.
- [Ben-Arie, 1990] J. Ben-Arie (1990). Probabilistic models of observed features and aspects with application to weighted aspect graphs. *Pattern recognition letters*, vol. 11, pages:421-427.
- [Besl, 1990] P. J. Besl (1990). Machine vision for three dimensional scenes. In H. Freeman, editor, Academic Press. Pages 25-71.
- [Besl et Jain, 1985] P. J. Besl et R. C. Jain (1985). Three-dimensional object recognition. *Computing Surveys*, 17(1):75-145.
- [Besl et McKay, 1992] P. J. Besl et N. D. McKay (1992). A Method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239-256.
- [Biederman, 1987] I. Biederman (1987). Recognition-by-components : A theory of human image understanding, *Psychological Review*.
- [Bischof et Leonardis, 1998] H. Bischof et A. Leonardis (1998). Robust recognition of scaled eigenimages through a hierarchical approach. *IEEE Conference on Computer Vision and Pattern Recognition*, pages: 664-670.
- [Black et Jepson, 1998] M. J. Black et A. D. Jepson (1998). Eigen tracking : robust matching and tracking of articulated objects using a view-based representation. *International Journal of Computer Vision*, 26(1):63-84.
- [Blake et Marinos, 1990] A. Blake et C. Marinos (1990). Shape from texture: estimation, isotropy and moments. *Artificial Intelligence*, 45(3):323-380.
- [Blum, 1967] H. Blum. A transformation for extracting new descriptors of shape. Symposium Models for Perception of Speech and Visual Form, MIT Press, Cambridge, 362-380.
- [Brady et Asada, 1984] M. Brady et H. Asada (1984). Smoothed local symmetries and their implementation. *International Journal of Robotics Research*.
- [Brill et al., 1992] M. H. Brill, E. B. Barrett et P. M. Bayton (1992). Projective invariants for curves in two and three dimensions. Dans [Mundy et Zisserman, 1992], chapitre 9.
- [Burns et al., 1990] J. B. Burns, R. Weiss et E. M. Riseman (1990). View variation of point set and line segment features. *DARPA Image Understanding Workshop*, pages : 650-659. Pittsburgh.
- [Califano et Mohan, 1991] A. Califano et R. Mohan. Multidimensional indexing for recognizing visual shapes. *IEEE Conference on Computer Vision and Pattern Recognition*. Pages: 28-34.

- [Camps et al., 1998] O. I. Camps, C. Y. Huang et T. Kanungo (1998). Hierarchical organization of appearance-based parts and relations for object recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, pages:685-691.
- [Christensen et Shapiro, 1993] H. Christensen et L. G. Shapiro (1993). Three-dimensional shape from color photometric stereo. *International Journal of Computer Vision*, 13(2):213-227.
- [Chen et Stockman, 1998] J. L. Chen et G. C. Stockman (1998). 3D Free-form objet recognition using indexing by contour features. *Computer Vision and Image Understanding*, 71(3):334-353.
- [Chu et Aggarwal, 1993] C. Chu et J. K. Aggarwal (1993). The integration of image segmentation maps using region and edge information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(12):1241-1252.
- [Chua et Jarvis, 1997] C. S. Chua et R. Jarvis (1997). Point signatures : a new representation for 3D object recognition. *International Journal of Computer Vision*, 25(1):63-58.
- [Cinque et al., 1996] L. Cinque, D. Yasuda, L. G. Shapiro, S. Tanimoto et B. Allen (1996). An improved algorithm for relational distance graph matching. *Pattern Recognition*, 29(2):349-359.
- [Cohen et Wang, 1992] F. S. Cohen et J. Y. Wang (1992). 3d recognition and shape estimation from image contours using invariants 3-d object curve models. Dans *International Conference on Computer Vision and Pattern Recognition*, pages 722-725.
- [Costa et al., 1992] M. Costa, R. Haralick et L. Shapiro (1992). Optimal affine-invariant matching: performance characterization. *SPIE Conference on Image Storage and Retrivial*, San José, CA.
- [Coster et Chermant, 1989] M. Coster et J. L. Chermant (1989). Précis d'analyse d'images. Presses du CNRS.
- [Curless et Levoy, 1996] B. Curless et M. Levoy (1996). A volumetric method for building complex models from range images. *SIGGRAPH 96 Conference Proceedings*, Annual Conference Series, pages 303-312, New Orleans, Louisiana. ACM SIGGRAPH, Addison Wesley.
- [Curless et Seitz, 2000] B. Curless et S. Seitz (2000). 3D Photography. *ACM Siggraph'00 Course Notes*. Course No. 19.
- [Dargentont et al., 1994] P. Dargentont, N. Vincent et H. Emptoz (1994). Appariement de deux graphes structurels quelconques pour la reconnaissance de lettres manuscrites. *9<sup>ème</sup> Congrès AFCET RFIA*, Paris, pages:461-471.
- [Delingette et al., 1992] H. Delingette, M. Herbert et K. Ikeushi (1992). Shape representation and image segmentation using deformable surfaces. *Image and Vision Computing*, 10:132-144.

- [Delingette et al., 1993] H. Delingette, M. Herbert et K. Ikeushi (1993). A spherical representation for the recognition of curved objects. *IEEE International Conference on Computer Vision*, pages:103-112.
- [Deriche et Giraudon, 1993] R. Deriche et G. Giraudon (1993). A computational approach for corner and vertex detection. *International Journal of Computer Vision*, 10(2):101-124.
- [Dhome et al.,1986] M. Dhome, M. Richetin et G. Rives (1986). Model-Based recognition and location of local patterns in polygonal contours via hypothesis accumulation. *E. S. Gelsema and L. N. Kanal, editors, Pattern Recognition in Practice II*, North Holland, Amsterdam, pages:211-218.
- [Dorai et Jain, 1997] C. Dorai et A. K. Jain (1997). COSMOS- a representation scheme for 3D free form objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19(10), 1115-1130.
- [Dudani et al., 1977] S. A. Dudani, K. J. Breeding et R. B. McGhee (1977). Aircraft identification by moment invariants. *IEEE Transactions on Computers*, 26(1):39-45.
- [Eggert et al., 1993] D. W. Eggert, K. W. Bowyer, C. R. Dyer, H. I. Christensen et D. B. Goldgof (1993). The scale-space aspect graph. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(11):1114-1130.
- [Faugeras, 1993] O. Faugeras (1993). Three dimensional computer vision: a geometric viewpoint. MIT Press.
- [Fischer et al., 1992] D. Fischer, O. Bachar, R. Nussinov et H. Wolfson (1992). An efficient automated computer vision-based technique for detection of three dimensional structural motifs in proteins. *Journal of Biomolecular Structures and Dynamics*, 9(4):769-769.
- [Flynn et Jain, 1992] P. Flynn et A. Jain (1992). 3D object recognition using invariant feature indexing of interpretation tables. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 55(2):119-129.
- [Foley et al., 1990] J. D. Foley, A. Van Dam, S. K. Feiner, et J. F. Hughes (1990). *Computer Graphics Principles and Practice*. The Systems Programming Series. Addison-Wesley Publishing Company, Inc.
- [Feldmar, 1995] J. Feldmar (1995). Recalage rigide, non rigide et projectif d'images médicales tridimensionnelles. Thèse de Doctorat, Ecole Polytechnique.
- [Freuder, 1978] E. C. Freuder, Synthesizing constraint expressions. *Communications of ACM*, 21(11):958-966.
- [Gagalowicz et Monga, 1986] A. Gagalowicz et Olivier Monga (1986). A new approach for image segmentation. *Proceedings, of the 8<sup>th</sup> International Conference on Pattern Recognition*, IEEE Publ. 86CH2342-4, pages 265-267, Paris, France.
- [Gherbi, 1992] R. Gherbi (1992). Mise en Correspondance de Chaînes de Contour. Application à la Stéréovision. thèse de doctorat, Université Paris XI, Orsay.

- [Gigus et Malik, 1990] Z. Gigus et J. Malik (1990). Computing the aspect graph for line drawings of polyhedral objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(2):113-122.
- [Gigus et al., 1991] Z. Gigus, J. Canny et R. Seidel (1991). Efficiently computing and representing aspect graphs of polyhedral objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(6):542-551.
- [Greespan, 1998] M. Greespan (1998). The sample tree : a sequential hypothesis testing approach to 3D object recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, pages:772-779.
- [Grimson et Lozano-Pérez, 1984] W. E. L. Grimson et T. Lozano-Pérez (1984). Model-Based recognition and localization from sparse range of tactile data. *International Journal of Robotics Research*, 3(3):3-35.
- [Grimson, 1985] W. E. L. Grimson (1985). Computational experiments with a feature based stereo algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(1):17-34.
- [Grimson et Lozano-Pérez, 1987] W.E.L. Grimson et T. Lozano-Perez (1987). Localizing Overlapping Parts by Searching the Interpretation Tree. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(4).
- [Grimson, 1990] W. E. L. Grimson (1990). The combinatorics of objects recognition in cluttered environment using constrained search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1):121-165.
- [Grimson et al., 1992] W. E. L. Grimson et D. P. Huttenlocher (1992). Recognizing 3D objects from 2D images : an error analysis. *IEEE Conference on Computer Vision and Pattern Recognition*.
- [Gros et al., 1995] P. Gros, O. Bournez, et E. Boyer (1995). Using geometric quasi-invariants to match and model images of line segments. Rapport de recherche N° 2608, INRIA.
- [Gros, 1995] P. Gros (1995). Matching and clustering : two steps towards objects modelling in computer vision. *International Journal of Computer Vision*. 14(5).
- [Guisser et al., 2000] L. Guisser, R. Payrissat, et S. Castan (2000). PGSD: an accurate 3D vision system using a projected grid for surface descriptions. *Image and Vision Computing*. 18, 463-491.
- [Haralick, 1979] R. M. Haralick (1979). Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786-804.
- [Haralick et Shapiro, 1993] R. M. Haralick et L. G. Shapiro (1993). Computer and robot vision. Addison-Wesley, Vol. 2.
- [Horaud et Skordas, 1989] R. Horaud et T. Skordas (1989). Stereo correspondance through feature grouping and maximal cliques. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(11):1168-1180.

- [Horaud et Monga, 1993] R. Horaud et O. Monga (1993). *Vision par Ordinateur : Outils Fondamentaux. Traité Des Nouvelles Technologies*. Editions Hermès.
- [Horaud et Sossa] R. Horaud et H. Sossa (1995). Polyhedral object recognition by indexing. *Pattern Recognition*, 28(12):1855-1870.
- [Horn, 1986] B. K. P. Horn (1986). *Robot Vision*. The MIT Press, ninth printing, 1993 edition.
- [Horn, 1987] B. K. P. Horn (1988). Closed-form solution of absolute orientation using unit quaternion. *Journal of Optical Soc. Am.*, vol. 4: 629-642.
- [Horn et al., 1988] B. K. P. Horn, H. M. Hilden et S. Negahdaripour (1988). Closed-form solution of absolute orientation using orthonormal matrices. *Journal of Optical Soc. Am.*, vol. 5: 127-135.
- [Hough, 1962] P. V. C. Hough (1962). Method and means for recognizing complex patterns. U.S. Patent 3, 069,654.
- [Hu, 1962] M. K. Hu (1962). Visual pattern recognition by moments invariants. *IRE Transactions on Information Theory*. IT-8:179-187.
- [Hu et Stockman, 1989] G. Hu et G. Stockman (1989). 3-D surface solution using structured light and constraint propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(4):390-402.
- [Huang et al., 1997] C. Y. Huang, O. I. Camps et T. Kanungo (1997). Object recognition using appearance-based parts and relations. *IEEE Conference on Computer Vision and Pattern Recognition*, pages:877-883.
- [Huttenlocher et Ullman, 1987] D. P. Huttenlocher et S. Ullman (1987). Object recognition using alignment. *IEEE International Conference on Computer Vision*, pages:72-78.
- [Huttenlocher et Kleinberg, 1994] D. P. Huttenlocher et J. M. Kleinberg (1994). Comparing point sets under projections. *The 5<sup>th</sup> Annual ACM-SIAM Symposium on Discrete Algorithms*, pages : 1-7, Arlington, Virginia.
- [Ikeuchi et Kanade, 1988] K. Ikeuchi et T. Kanade (1988). Automatic generation of object recognition programs. *IEEE*, 76(8):1016-1035.
- [Izquierdo et al., 1999] M.A.G. Izquierdo, M.T. Sanchez, A. Ibanez, et L. G. Ullate (1999). Sub-pixel measurements of 3D surfaces by laser scanning. *Sensors and Actuators Journal* 76. 1-8.
- [Jarvis, 1983] R. A. Jarvis (1983). A perspective on range finding techniques for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 5(2), 122-139.
- [Jonhson et Hebert, 1999] A. E. Johnson et M. Hebert (1999). Using spin-images for efficient object recognition in cluttered 3-D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):433-449.
- [Joshi et al., 1994] S. Joshi, J. Ponce, B. Vijayakumar et D. J. Kriegman (1994). HOT curves for modeling and recognition of smooth curved 3D objects. *IEEE Conference on Computer Vision and Pattern Recognition*, pages: 876-880.

- [Kass et al., 1988] M. Kass, A. Witkin, et D. Terzopoulos (1988). Snakes : Active contour models. *International Journal of Computer Vision*, pages 321-331.
- [Khotanzad et Lu, 1990] A. Khotanzad et J. H. Lu (1997). Classification of invariant image representations using a neural network. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(6):1028-1038.
- [Kirby et Sirovich, 1990] M. Kirby et L. Sirovich (1990). Application of the Karhunen-Loeve procedure for the characterization of human faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):103-108.
- [Koenderink et Van Door, 1979] J. J. Koenderink et A. J. Van Doorn (1979). The internal representation of solid shape with respect to vision. *Biological Cybernetics*, 32:211-216.
- [Kriegman et Ponce, 1990] D. Kriegman et J. Ponce (1990). Computing exact aspect graphs of curved objects: solids of revolution. *International Journal of Computer Vision*, vol. 5, pages: 119-135.
- [Krishnamurthy et Levoy, 1996] V. Krishnamurthy et M. Levoy (1996). Fitting smooth surfaces to dense polygon meshes. *Proc. of SIGGRAPH'96*, pages : 313-324.
- [Krumm, 1996] J. Krumm (1996). Eigenfeatures for planar pose measurements of partially occluded objects. *IEEE Conference on Computer Vision and Pattern Recognition*, pages: 55-60.
- [Leonardis et Bischof, 1996] A. Leonardis et H. Bischof (1996). Dealing with occlusions in the eigenspace approach. *IEEE Conference on Computer Vision and Pattern Recognition*, pages: 453-458.
- [Lamdan et Wolfson, 1988] Y. Lamdan et H. Wolfson (1988). Geometric Hashing : a general and efficient model-based recognition scheme. *IEEE International Conference on Computer Vision*, pages: 238-249.
- [Lim et Binford, 1987] H. S. Lim et T. O. Binford (1987). Stereo correspondence: A hierarchical approach. *Image Understanding Workshop*, volume 1, pages 234-241, Los Angeles, California. DARPA.
- [Lowe, 1985] D. G. Lowe (1985). Perceptual organization and visual recognition. Kluwer Academic Publishers.
- [Lowe, 1987] D. G. Lowe (1987). Three dimensional object recognition from single two-dimensional images, *Artificial Intelligence*, (31); pages: 355-395.
- [Lowe, 1991] D. G. Lowe (1991). Fitting parametrized three-dimensional models to images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):441-450.
- [Marr, 1982] D. Marr (1982). VISION a Computational Investigation Into the Human Representation and Processing of Visual Information. W.H. Freeman and Company.
- [Marr et Nishihara, 1976] D. Marr et H. K. Nishihara (1976). Representation and recognition of the spatial organization of three dimensional shapes. MIT AI Memo 377.

- [Mokhtarian, 1995] F. Mokhtarian (1995). Silhouette-Based isolated object recognition through curvature scale space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):539-544.
- [Mohan et al., 1993] R. Mohan, D. Weinshall et R. Sarukkai. 3D object recognition by indexing invariants from multiple views. *IEEE International Conference on Computer Vision*. Pages: 264-268.
- [Montanari, 1974] U. Montanari (1974). Networks of constraints: Fundamental properties and applications to picture processing. *Information Science*, 7, 95-132.
- [Moons et al., 1995] T. Moons, E. Pauwels, L. Van Gool et A. Oosterlinck (1995). Foundations of semi-differential invariants. *International Journal of Computer Vision*, 14(1):25-47.
- [Mundy et Zisserman, 1992] J. Mundy et A. Zisserman, editors (1992). Geometric invariance in computer vision. MIT Press.
- [Mundy et al., 1996] J. Mundy, A. Liu, N. Pillow, A. Zisserman, S. Abdallah, S. Utke, S. Nayar et C. Rothwell (1996). An experimental comparison of appearance and geometric model based recognition. Dans J. Ponce, A. Zisserman et M. Hebert, editors, *Proc. Object Representation in Computer Vision II*, LNCS 1144, pages:247-269. Springer-Verlag.
- [Murase et Nayar, 1994] H. Murase et S. K. Nayar (1994). Illumination planning for object recognition using parametric eigenspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(12):1219-1227.
- [Murase et Nayar, 1995] H. Murase et S. K. Nayar (1995). Visual learning and recognition of 3D objects from appearance. *International Journal of Computer Vision*, 14:5-24.
- [Naf et al., 1997] M. Naf, G. Szekely, R. Kikinis, M. E. Shenton et O. Kubler (1997). 3D Voronoï skeletons and their usage for the characterization and recognition of 3D organ shape. *Computer Vision and Image Understanding*, 66(2):147-161.
- [Nevatia et Binford, 1977] R. Nevatia et T. O. Binford (1977). Description and recognition of curved objects. *Artificial Intelligence*, 8(1):69-76.
- [Ning et Bloomenthal, 1993] P. Ning et J. Bloomenthal (1993). An evaluation of implicit surfaces tilers. *Computer Graphics and Applications*, 13(3):33-41.
- [Nutbourne et Martin, 1988] A. W. Nutbourne et R. R. Martin (1988). Differential geometry applied to curve and surface design. Ellis Horwood.
- [Ohba et Ikeushi, 1997] K. Ohba et K. Ikeushi (1997). Detectability, Uniqueness and reliability of eigen windows for stable verification of partially occluded objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(9):1043-1048.
- [Pajdla et Van Gool, 1995] T. Pajdla et L. Van Gool (1995). Matching of 3D curves using semi-differential invariants. *IEEE International Conference on Computer Vision*. Pages: 390-395.

- [Pennec, 1998] X. Pennec (1998). Toward a generic framework for recognition based on uncertain geometric features. *Videre : Journal of Computer Vision Research*, MIT Press, pages:58-87.
- [Pentland, 1986] A. P. Pentland (1986). Perceptual organization and the representation of natural form. *Artificial Intelligence*, 28(3):293-331.
- [Persoon et Fu, 1977] E. Persoon et K. S. Fu (1977). Shape discrimination using Fourier descriptors. *IEEE Transactions on Sys. Man., Cybern.*, 7(3): 388-397.
- [Petitjean et al., 1992] S. Petitjean, J. Ponce et D.J Kriegman (1992). Computing exact aspect graphs of curved objects : algebraic surfaces. *International Journal of Computer Vision*, 9(3):231-255.
- [Pillow et al., 1995] N. Pillow, S. Utcke et A. Zisserman (1995). Viewpoint-invariant representation of generalized cylinders using the symetry set. *Image and Vision Computing*, 13(5):355-365.
- [Plantinga et Dyer, 1990] H. Plantinga et C. R. Dyer (1990). Vissibility, occlusion and the aspect graph. *International Journal of Computer Vision*, vol. 5, pages:137-160.
- [Ponce et Kriegman, 1989] J. Ponce et D. J. Kriegman (1989). On recognition and positioning curved 3D objects from image contours. *IEEE Workshop on Interpretation of 3D Scenes*, pages : 61-67.
- [Raja et Jain, 1992] N. S. Raja et A. K. Jain (1992). Recognizing geons from superquadrics fitted to range data. *Image and Vision Computing*, 10(3):179-190.
- [Rao, 1997] R. Rao (1997). Dynamic appearance-based recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, pages : 540-546.
- [Richards et Hoffman, 1985] W. Richards et D. Hoffman (1985). Codon constraints on closed 2D shapes. *CVGIP*, 31, 265-281.
- [Rieger, 1990] J. H. Rieger (1990). The geometry of view space of opaque objects bounded by smooth surfaces. *Artificial Intelligence*, vol. 44, pages: 1-40.
- [Rieger, 1993] J. H. Rieger (1993). Computing view graphs of algebraic surfaces. *Journal of Symbolic Computation*, 16:259-272.
- [Rigoutsos et Hummel, 1992] I. Rigoutsos et R. Hummel (1992). Massively parallel model matching : geometric hashing on the connection machine. *IEEE Computer : special issue on Parallel Processing for Computer Vision and Image Understanding*, 25(2):33-42.
- [Rigoutsos et Hummel, 1995] I. Rigoutsos et R. Hummel (1995). A Bayesian approach to model matching with geometric hashing. *Computer Vision and Image Understanding*, 61(7).
- [Rigoutsos et al., 1996] I. Rigoutsos, D. Platt et A. Califano (1996). Flexible 3D-substructure matching & novel conformer derivation in very large databases of molecular information. *Technical Report RC20497*, IBM T. J. Watson Research Center.

- [Rocchini et al., 2001] C. Rocchini, P. Cigoni, C. Montani, P. Pingi, et R. Scopigno (2001). A low cost 3D scanner based on structured light. *Eurographics Conference* (pp. 299-308). Interlaken, Switzerland.
- [Rothwell et al., 1992] C. Rothwell, A. Zisserman, D. Forsyth et J. Mundy (1992). Canonical frames for planar object recognition. *European Conference on Computer Vision*, LNCS 588, pages: 757-772. Springer-Verlag.
- [Rothwell et al., 1995] C. Rothwell, A. Zisserman, D. Forsyth et J. Mundy (1995). Planar object recognition using projective shape representation. *International Journal of Computer Vision*, 16(2).
- [Roussel, 1999] D. Roussel (1999). Reconstruction de courbes et de surfaces 3d en stéréo-acquisition. thèse de doctorat, Université Paris XI, Orsay.
- [Sallam et al., 1990] M. Y. Sallam, J. S. Stewman et K. W. Bowyer (1990). Computing the visual potential of articulated assemblies. *IEEE International Conference on Computer Vision*, pages: 636-643.
- [Schiffenbauer, 2001] R. D. Schiffenbauer (2001). A survey of aspect graphs. *Technical Report*, TR-CIS-2001-01, Polytechnic University, Brooklyn-Long Island–Westchester.
- [Schwartz et Sharir, 1986] J. Schwartz et M. Sharir (1986). Identification in partially obscured objects in two and three dimensions by matching noisy characteristic curves. *The International Journal of Robotics Research*, 6(2):29-44.
- [Shapiro et Haralick, 1981] L. G. Shapiro et R. M. Haralick (1981). Structural descriptions and inexact matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3(5):504-519.
- [Shen et Yang, 1998] J. Shen et Y. H. Yang (1998). Deformable object modeling using the time dependent finite element method. *Graphical Models and Image Processing*, 60:461-487.
- [Shimshoni et Ponce, 1997] I. Shimshoni et J. Ponce (1997). Finite-resolution aspect graphs of polyhedral objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):315-327.
- [Stein et Medioni, 1992] F. Stein et G. Medioni (1992). Structural indexing : efficient 3D object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):125-145.
- [Solina et Bajcsy, 1990] F. Solina et R. Bajcsy (1990). Recovery of parametric models from range images : the case for superquadrics with global deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(2):131-147.
- [Stewman et Bowyer, 1990] J. Stewman et K. Bowyer (1990). Direct construction of the perspective projection aspect graph of convex polyhedra. *Computer Vision Graphics Image Processing*, vol. 51, pages: 20-37.

- [Subbarao et Choi, 1995] M. Subbarao et R. Choi (1995). Accurate recovery of three-dimensional shape from image focus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(3):266-274.
- [Taubin, 1991] G. Taubin (1991). Estimation of planar curves, surfaces and nonplanar space curves defined by implicit equations with applications to edges and range image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(11):1115-1138.
- [Thirion, 1996] J. P. Thirion (1996). The extremal mesh and the understanding of 3D surfaces. *International Journal of Computer Vision*, 19(2):115-128.
- [Thirion et Ronse, 1996] E. Thirion et C. Ronse (1996). Self calibration and 3d reconstruction from lines with a single translating camera. Technical Report 96/06, Laboratoire des Sciences de l'Image, de l'Informatique et de la Télédétection LSIIIT - URA CNRS 1871, Université Louis Pasteur, Strasbourg.
- [Tsai, 1987] R. Y. Tsai (1987). 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*, 3(4):323-344.
- [Turk et Levoy, 1994] G. Turk et M. Levoy (1994). Zippered polygon meshes from range images. *Computer Graphics Proceedings, SIGGRAPH 94*, Annual Conference, pages 311-318, Orlando, Florida.
- [Turk et Pentland, 1991] M. Turk et A. Pentland (1991). Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71-86.
- [Valkenburg et McIvor, 1998] R. J. Valkenburg et A. M. McIvor (1998). Accurate 3D measurement using structured light system. *Image and Vision Computing* 16(2), 99-110.
- [Wang et Freeman, 1990] R. Wang et H. Freeman (1990). Object recognition based on characteristic view classes. *The 10<sup>th</sup> International Conference of Pattern Recognition*, pages: 8-12.
- [Weiss, 1993] I. Weiss (1993). Geometric invariants and object recognition. *International Journal of Computer Vision*, 10(3): 207-231.
- [Wolfson et Lamdan, 1992] H. J. Wolfson et Y. Lamdan. Transformation invariant indexing. Dans [Mundy et Zisserman, 1992]. Pages: 335-353.
- [Worthington et Hancock, 2001] P. L. Worthington et E. R. Hancock (2001). Object Recognition Using Shape-from-Shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(5): 535-542.
- [Xia et Varshney, 1996] J. Xia et A. Varshney (1996). Dynamic view-dependent simplification for polygonal models. *IEEE International Conference on Visualization*, pages : 327-334.
- [Yang et al., 1997] L. Yang, F. Albrechtsen et T. Taxt (1997). Fast computation of three-dimensional geometric moments using a discrete divergence theorem and a

generalization to higher dimensions. *Graphical Models and Image Processing*, 59(2):97-108.

[Zhang et Hebert, 1997] D. Zhang et M. Hebert (1999). Multi-scale classification of 3-D objects. *IEEE Conference on Computer Vision and Pattern Recognition*, pages:864-869.

[Zisserman et al., 1995] A. Zisserman, D. Forsyth, J. Mundy, C. Rothwell, C. Liu et N. Pillow (1995). 3D object recognition using invariance. *Artificial Intelligence*, (78), pages : 239-288.