
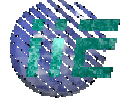





MASTER : **SCIENCES ET INGENIERIE**  
 MENTION : *Sciences Pour l'Ingénieur*  
 SPECIALITE RECHERCHE : **Réalité Virtuelle et Systèmes Intelligents**  
 – RVSI –  
 (Ex DEA RVMSC)

#### Etablissements co habilités

		
<b>INT</b> Institut National des Télécommunications	<b>IIE-CNAM</b> Institut d'Informatique d'Entreprise	<b>ENSMP</b> Ecole Nationale Supérieure Des Mines de Paris

## Cours : Vision pour la Réalité Augmentée

*Malik Mallem*  
(UEVE/LSC)

### Table des matières

1. Données de l'environnement.....	3
2. Relevé 3D en ligne.....	3
2.1. Modélisation de prisme symétrique .....	5
2.2. Modélisation de cylindre.....	6
2.3. Modélisation d'objets à symétrie de révolution .....	10
3. Reconstruction 3D: Mise à jour de la représentation informatique de l'environnement.....	13
3.1. Introduction .....	13
3.2 Le problème .....	14
3.3 Méthode géométrique des 3 segments.....	16
3.4 Méthode linéaire à petit mouvement .....	17
3.5 Méthode non-linéaire .....	19
3.5. Méthode employée (Mixte).....	20
3.6. Méthode de l'itération orthogonale.....	21
3.6.1 Définition du problème .....	21
3.6.2 Algorithme d'itération orthogonale (IO).....	23
3.6.3 Problème de l'orientation absolue .....	23
3.6.4 Algorithme .....	24
4. Représentation informatique des données de l'environnement.....	27
4.1. Représentation filaire .....	28
4.2. Représentations surfaciques .....	28
4.3. Représentations volumiques.....	29
4.4 Codage des données géométriques de l'environnement .....	29
5. Appariement d'une image 2D avec son modèle 3D .....	38
5.1. Bref état de l'art.....	38
5.2. Appariement 2D/3D polyédrique basée sur liaisons .....	39
5.3. Appariement 2D/3D polyédrique .....	39
5.4. Traitement et Modélisation de l'image 2D.....	40
5.5. Construction des graphes d'aspects pour polyèdre .....	43
5.6. Méthodes basées sur le graphe d'appariement et sur le hachage géométrique.....	46
5.7 Méthode basée sur la photogrammétrie.....	50
5.8 Méthode basée sur la clinométrie.....	55
Conclusion .....	12
BIBLIOGRAPHIE.....	61

# MODELISATION D'ENVIRONNEMENT 3D

La modélisation 3D de l'environnement est indispensable pour la réalisation de tâches de Télérobotique.

L'accomplissement de cette modélisation nécessite de répondre à 4 questions :

1. quelles données pertinentes extraire de l'environnement ?
2. comment les extraire ?
3. comment les maintenir à jour ?
4. comment les représenter dans le calculateur ?
5. comment les reconnaître dans le calculateur à partir de leur image caméra.

Des éléments de réponse à ces différentes questions sont fournis respectivement dans les paragraphes suivants.

## 1. Données de l'environnement

Les données pertinentes à extraire de l'environnement sont liées à la tâche à réaliser. L'environnement de télérobotique est constitué de deux types d'objets: les objets connus (robots, outils, pièces à manipuler, ...) et les objets non connus (obstacles,...). Les données géométriques concernant les obstacles sont suffisantes pour une planification de trajectoire du robot afin d'éviter ceux-ci. En revanche, des données cinématiques ou dynamiques des objets à manipuler peuvent être indispensables pour réaliser des tâches de manipulation. Dans ce chapitre, nous décrivons des procédés de relevé 3D et de mise à jour de ces données.

Les objets modélisés sont généralement polyédriques. Leurs données sont saisies, soit hors ligne par une méthode de type CAO, soit en ligne par des méthodes de modélisation utilisant des capteurs tels que caméra et télémètre. Ces données concernent la frontière de l'objet représenté par ses faces, arêtes et sommets connus dans un repère de travail. Ces données servent à la création d'un modèle géométrique de l'environnement.

## 2. Relevé 3D en ligne

Ce paragraphe répond à la question « Comment extraire les données 3D des objets de l'environnement ? »

La modélisation géométrique 3D d'objets nécessite l'utilisation d'un dispositif de relevé 3D. Celui-ci peut être constitué par une caméra mobile, de deux voire trois caméras, de télémètre déposé sur un support rotatif ou à triangulation.

Des travaux concernant la vision monoculaire dynamique pour la modélisation d'objet de révolution sont décrits dans [DHO 90], [SAY 96].

D'autres concernant la vision stéréoscopique voire trinoculaire sont nombreux [FAU 93], [AYA 91].

Ceux concernant la modélisation d'images de profondeur le sont moins :

Ils portent sur de relevé 3D télémétrique par balayage de routes [TRA 93], sur le relevé 3D d'objets en utilisant un télémètre à temps de vol monté sur un dispositif rotatif [COL 94], sur la modélisation de polyèdres en utilisant un télémètre à triangulation. Les données fournies, par plusieurs points de vue de l'objet à modéliser, sont fusionnées afin d'obtenir les sommets et les arêtes de l'objet [BAD 94].

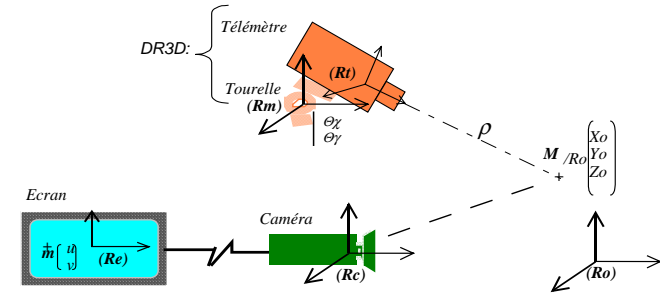
Neveu et al [NEV 95], Solina et al [SOL 94] ont présenté une modélisation par des superquadriques de la surface d'objets de forme courbe et présentant une symétrie.

Kumar et al [KUM 95] ont travaillé sur la modélisation par des hyperquadriques de la surface d'objets, de forme courbe non forcément symétrique, en utilisant des données télémétriques.

Dans cette partie, des méthodes de modélisation de volumes englobants d'objets polyédriques et cylindriques, présentant une symétrie, sont présentées en utilisant différents types de capteurs 3D. Cette modélisation est réalisée intégralement avec la coopération de l'opérateur qui indique la classe d'objet à modéliser et télécommande les capteurs afin de réaliser le relevé 3D nécessaire au calcul des dimensions et l'attitude spatiale de l'objet.

Trois classes d'objets sont considérées : celle des prismes symétriques (parallélépipède, pyramides tronquées, pyramides,...), celle des cylindres (cylindres, cônes tronqués, cônes) et celle des objets à symétrie de révolution, à axe droit, n'appartenant pas aux deux catégories précédentes et à génératrice non forcément rectiligne.

L'idée originale pour toutes ces méthodes, consiste à utiliser un capteur mixte constitué par une caméra et un DR3D(Dispositif de Relevé 3D). Le DR3D(Figure 2. 1) est constitué d'un télémètre laser, de mesure de temps de vol, monté sur une tourelle à 2 degrés de liberté(site et azimut) .



- Rm: Repère lié à la tourelle
- Rt: Repère du télémètre
- Rc: Repère de la caméra
- Ro: Repère global
- Re: Repère de visualisation

$$(X_o, Y_o, Z_o) = f_{DR3D}(\theta_x, \theta_y, \rho)$$

$$(u, v) = f_{camera}(X_o, Y_o, Z_o)$$

Figure 2. 1. DR3D(Dispositif de Relevé 3D)

L'image caméra sert à positionner l'objet dans une pyramide infinie ayant pour sommet le centre optique de la caméra. Le DR3D fournit l'information de profondeur qui permet de situer l'objet dans la pyramide.

Des méthodes, de relevé 3D, pour chacune des classes des primitives sont décrites. Auparavant, nous présentons la notion de plan d'interprétation indispensable pour la compréhension de ces méthodes.

Le plan d'interprétation est défini à l'aide du modèle de la caméra (modèle du sténopé, cf chapitre calibration de capteurs). Il exprime la relation entre un segment de droite réel, sa projection dans le plan image et le centre optique de la caméra (Figure 2. 2).

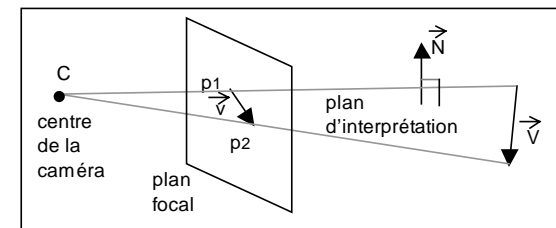


Figure 2. 2. Plan d'interprétation

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

2.1. Modélisation de prisme symétrique

Pour un prisme symétrique, le relevé 3D de deux faces suffit pour le modéliser. Une face extrême dite principale et une des faces latérales dite secondaire sont utilisées. Les faces choisies sont celles pour lesquelles les mesures télémétriques sont les meilleures et qui sont vues simultanément par la caméra et par le télémètre.

L'image caméra et le modèle géométrique inverse de celle-ci permettent de déterminer les plans tangents (plans d'interprétation) aux contours de chacune des deux faces - un plan d'interprétation est un plan qui passe par une arête de l'objet et le centre optique de la caméra - . Ces plans permettent de définir une pyramide infinie ayant pour sommet le centre optique de la caméra. Le télémètre permet de déterminer l'équation du plan porteur de la face. L'intersection entre la pyramide et le plan porteur de la face permet de définir le contour de celle-ci.

Le volume englobant est ensuite généré en traduisant la face principale selon un vecteur ayant une extrémité commune avec la face secondaire.

La Figure 2. 3 représente les plans d'interprétation  $P_i (i=1..4)$  qui sont déterminés grâce aux images des arêtes AB, BC, CD et DA, correspondantes à une face d'un prisme vu simultanément par la caméra et par le DR3D.

Le DR3D permet de relever un ensemble de points 3D appartenant à la surface à modéliser. La modélisation précédente est réitérée sur la surface secondaire (Figure 2. 3) , qui doit également être vue par la caméra et le DR3D simultanément. Des contraintes type point commun/arête commune (Figure 2. 3) avec la face préalablement modélisée sont prises en compte pour le calcul de la nouvelle face. Le volume englobant l'objet est calculé par la translation de la face principale selon un vecteur défini par la face secondaire.

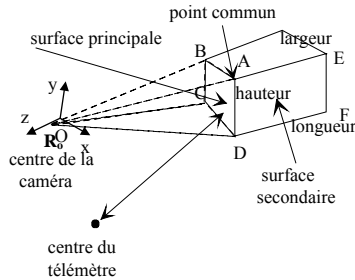


Figure 2. 3. Modélisation 3D de polyèdre

Pour une face qui n'est pas parallèle à l'axe des z, le coefficient directeur suivant z est toujours différent de zéro, donc une normalisation est possible.

l'équation du plan devient alors:

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

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

$$A \cdot X = B \text{ avec :}$$

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

Ce système permet de calculer le plan porteur d'une face.

2.2. Modélisation de cylindre

Dans cette partie sont présentées différentes méthodes employées pour la détermination des caractéristiques d'une primitive cylindrique observée à l'aide d'une caméra et du DR3D.

Pour chaque méthode nous déterminons la position de l'axe de la primitive cylindrique et son rayon.

La définition des différents plans d'interprétation est illustrée (Figure 2. 4). La caméra permet de définir l'équation des deux plans d'interprétation  $P_{c1}$  et  $P_{c2}$  relatifs aux génératrices du cylindre.

En tenant compte de l'attitude du cylindre (vertical, horizontal ou incliné) nous effectuons avec le DR3D deux balayages à deux niveaux différents. A l'aide des points, obtenus après balayage, on détermine l'équation d'un plan frontal  $P_{tl3}$  tangent au cylindre. Compte tenu du diamètre du faisceau laser, le profil du cylindre relevé ne correspond pas au profil exact (Figure 2. 5). Cependant on peut détecter les extrêmes. Chaque balayage permet d'obtenir les coordonnées des points latéraux et de déterminer les plans  $P_{tl1}$  et  $P_{tl2}$  (Figure 2. 4).

Pour la définition du plan  $P_{tl3}$ , on considère deux droites tangentes au cylindre. Chaque balayage permet de définir une droite. Pour chaque droite, on ne considère qu'un nombre réduit de points (Figure 2. 5).

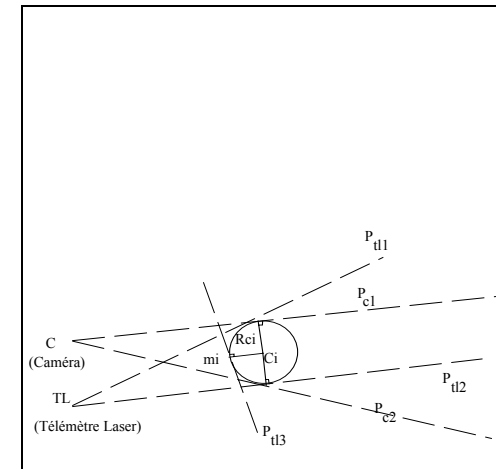


Figure 2. 4. Définition des différents plans d'interprétation

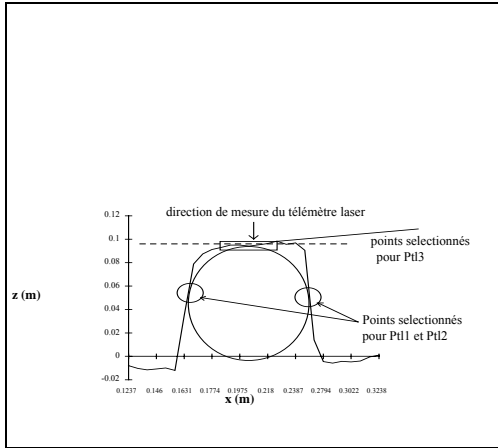


Figure 2.5. Profil d'un cylindre

Les différents plans d'interprétation étant définis, nous décrivons trois méthodes de modélisation de cylindre ainsi que les résultats des expériences d'évaluation menées.

2.2.1. Coopération caméra/DR3D: plans tangents

Dans cette approche les plans  $Pc1$ ,  $Pc2$  et  $Pt13$  (Figure 2. 5.), qui constituent trois plans distincts tangents au cylindre, sont utilisés afin de calculer deux points  $C_1$  et  $C_2$  se trouvant sur l'axe du cylindre et qui sont équidistants à ces trois plans. Les deux altitudes de  $C_1$  et  $C_2$  sont connues et correspondent à celles des balayages télémétriques. Pour chaque balayage, les projections orthogonales de  $C_j$  ( $j=1,2$ ) sur les plans  $Pc1$ ,  $Pc2$  et  $Pt13$  sont calculées.

Ces plans sont définis par les équations suivantes:

$$(P_i : a_i x + b_i y + c_i z + d_i = 0, i=1..3)$$

Ces  $P_i$  correspondent au plan  $Pc1$ ,  $Pc2$  et  $Pt13$

Un premier groupe de trois relations permet d'exprimer que  $C_j$  est à la distance  $R_{c_j}$  de ces trois plans:

$$\frac{|a_i x_{c_j} + b_i y_{c_j} + c_i z_{c_j} + d_i|}{\sqrt{a_i^2 + b_i^2 + c_i^2}} = R_{c_j} \text{ avec } i = 1, 2, 3 \text{ et } j=1, 2.$$

les inconnues sont ici  $x_{c_j}, z_{c_j}, R_{c_j}$  une des coordonnées peut être fixée a priori:  $y_{c_j}$  par exemple (dans le cas d'un balayage horizontal avec le télémètre); dans le cas d'un balayage vertical le paramètre fixé est  $x_{c_j}$ .

La valeur absolue est levée en calculant l'orientation du vecteur normal au plan  $P_i$ . Le signe de la projection de  $C_j$  sur  $Pt13$  est l'opposé du signe de la projection du point  $TL$  (connu), centre du télémètre, sur  $Pt13$ . Le signe de la projection de  $C_j$  sur  $Pc1$  ou  $Pc2$  est le même que le signe de la projection d'un point  $mi$ , point donné par le balayage, sur  $Pc1$  ou  $Pc2$  (Figure 2. 4).

En posant:

$$x = \begin{pmatrix} x_{c1} \\ z_{c1} \\ R_{c1} \end{pmatrix} A = \begin{pmatrix} a_1 & c_1 & -\sqrt{a_1^2 + b_1^2 + c_1^2} \\ a_2 & c_2 & -\sqrt{a_2^2 + b_2^2 + c_2^2} \\ a_3 & c_3 & -\sqrt{a_3^2 + b_3^2 + c_3^2} \end{pmatrix} \text{ et } B = \begin{pmatrix} -b_1 y_{c1} - d_1 \\ -b_2 y_{c1} - d_2 \\ -b_3 y_{c1} - d_3 \end{pmatrix}$$

Le système peut alors s'écrire:

$$AX = B$$

Que l'on résoud alors par une méthode type moindres carrés.

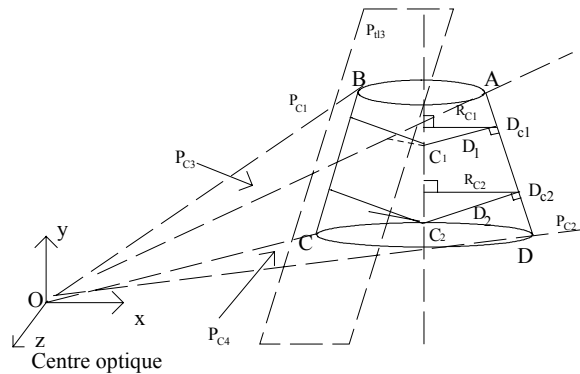
$$X = (A^t A)^{-1} A^t B$$

La solution exprimant X permet de définir les inconnues  $x_{c1}, z_{c1}, R_{c1}$ .

On procède de la même manière avec le deuxième point  $C_2$  et on détermine alors les inconnues  $x_{c2}, z_{c2}, R_{c2}$  avec ici  $y_{c2}$  fixé.

Ainsi les coordonnées des points  $(x_{c1}, y_{c1}, z_{c1})$  et  $(x_{c2}, y_{c2}, z_{c2})$  permettent de déterminer l'équation de l'axe du cylindre.

La hauteur du cylindre est déterminée en utilisant les plans  $Pc3$ ,  $Pc4$  et  $Pt13$ .



**Figure 2. 5. Méthode des trois plans tangents sur deux altitudes différentes**

Pour cette approche sont utilisés deux plans latéraux tangents au cylindre et définis par la caméra et un plan facial tangent au cylindre et défini par le télémètre.  
 Une expérience a été réalisée avec un cylindre droit dont les caractéristiques sont les suivantes [NZI 95]:  
 Diamètre : 10.2 cm , hauteur : 15.6 cm  
 Cette approche permet d'obtenir des erreurs inférieures, sur le rayon à 2%, la hauteur à 8‰ et la profondeur à 5.5‰ sur une distance objet-capteur de 1m à 1.6m. Le temps de modélisation est inférieur à 20 secondes.

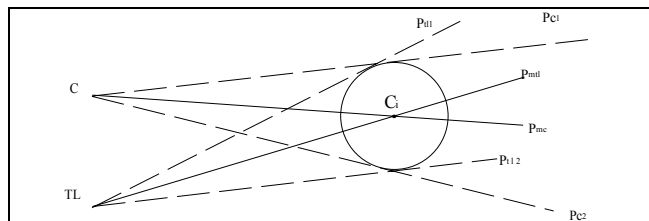
**2.2.2. Télémètre seul: plans tangents**

Cette approche s'apparente à la précédente, la résolution est identique. Cependant, la détermination de l'axe du cylindre s'appuie désormais sur des plans tangents  $P_{t11}$ ,  $P_{t12}$  et  $P_{t13}$  donnés par le DR3D (Figure 2. 4).

Cette approche est identique à la précédente. La détermination de l'axe du cylindre s'appuie sur trois plans qui lui sont tangents. Ces plans sont obtenus à l'aide des mesures données par le télémètre.  
 Les erreurs sur le rayon et sur la hauteur sont du même ordre de grandeur que pour la méthode précédente. Par contre les erreurs sur la profondeur sont plus grandes, inférieures à 5.5‰. Le balayage nécessite un temps plus important (2 à 4 minutes) ce qui augmente considérablement le temps pour la modélisation.

**2.2.3. Coopération caméra télémètre: plans bissecteurs**

Les plans d'interprétation,  $P_{c1}$  et  $P_{c2}$ , fournis par la caméra permettent de définir l'équation du plan bissecteur  $P_{mc}$ .  
 Les plans d'interprétation donnés par le DR3D  $P_{t11}$  et  $P_{t12}$  permettent de définir le plan bissecteur  $P_{mt1}$ . La détermination de l'équation de l'axe du cylindre est alors obtenue par l'intersection des deux plans bissecteurs  $P_{mc}$  et  $P_{mt1}$  (Figure 2. 7).



**Figure 2. 7. Plans bissecteurs de la caméra et du télémètre**

Les plans,  $P_{t11}$  et  $P_{t12}$  étant obtenus par balayage télémétrique, le temps de modélisation est de l'ordre de 2 à 3mn.  
 Les erreurs sur le rayon et sur la hauteur sont du même ordre que les deux précédentes méthodes. Les erreurs sur la profondeur sont inférieures à 1.5%. Cependant l'erreur en profondeur augmente considérablement quand la distance capteur-objet augmente.  
 La méthode des plans tangents (caméra + télémètre) est la plus rapide en temps de modélisation (moins de 20s). C'est aussi la plus précise pour des distances de 1.4m à 1.6m.  
 Pour la méthode basée sur les plans bissecteurs, l'erreur en profondeur augmente rapidement avec la distance capteur-objet. En effet, plus l'objet est éloigné, plus l'angle entre les deux plans bissecteurs est faible. Une erreur sur la détermination d'un plan se traduit par une erreur importante sur la position de l'intersection de ces plans.

**2.3. Modélisation d'objets à symétrie de révolution**

La modélisation d'objets à symétrie de révolution est basée sur l'utilisation des courbes B-Spline. Elle est applicable à des objets d'axe droit. Le principe de modélisation consiste à représenter une partie du contour du bord de l'objet appelé contour de base (Figure 2. 8), par une courbe B-spline, et à effectuer un balayage rotoïde de celui-ci autour de l'axe afin de générer le corps de révolution.

Nous commençons par décrire les données nécessaires à cette modélisation avant d'expliquer la méthode mathématique.

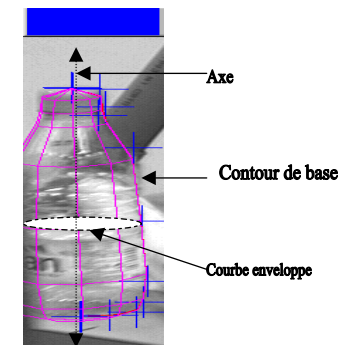
**2.3.1. Données de modélisation**

Des points caractéristiques (croix sur la Figure 2. 8) du contour de base de l'objet sont désignés sur l'image caméra de celui-ci. L'axe est connu, il peut être calculé par la méthode décrite dans le § 2.2.1.

**2.3.2. Méthode de modélisation**

La courbe B-spline est calculée pour le contour de base. La détermination des coordonnées 3D des points du contour de base est possible car ceux-ci sont situés sur la bordure visible de l'objet, et qu'ils sont coplanaires avec l'axe. Aux points obtenus sont alors appliquées des rotations autour de l'axe.

Ci-dessous un exemple de résultat obtenu :



**Figure 2. 8. Exemple de modélisation d'une bouteille**

Nous présentons la méthode de génération de la courbe représentant la contour de base, ainsi que la méthode de construction du volume englobant l'objet de révolution :

#### **Génération des courbes :**

La génération des courbes se fait par la méthode des B-Spline, leur formalisme mathématique est décrit dans l'annexe 1.

#### **Génération du volume englobant :**

Afin de générer le volume englobant l'objet de révolution, il est nécessaire de calculer les coordonnées 3D du contour de base et ceux des courbes enveloppes.

#### **Détermination des coordonnées 3D des points du contour de base :**

On sait que les points du contour de base sont situés sur la bordure visible de l'objet, et cet objet est à symétrie de révolution, donc ils sont inclus dans un plan  $P$ , qui contient d'une part l'axe de révolution  $\vec{A}$ , et d'autre part un vecteur  $\vec{V}$  parallèle au plan image de la caméra. Ce 2<sup>nd</sup> vecteur est défini par :

$$\vec{V} = \vec{T} \times \vec{A}$$

où  $\vec{T}$  est la normale au plan image de la caméra, et  $\times$  le produit vectoriel.

Ainsi, on connaît le vecteur normal au plan  $P$  ( $\vec{N} = \vec{A} \times \vec{V}$ ), on connaît un point de ce plan (une des extrémités du segment formant l'axe), on peut donc calculer les coordonnées 3D des points du contour de base.

Soient

$x$  et  $y$  les coordonnées d'un point sur l'image vidéo, correspondant à un point  $X$  du contour de base,

$\vec{N}$  : le vecteur normal au plan contenant le contour de base, et  $\vec{P}$  un vecteur contenant les coordonnées d'un point appartenant à ce plan.

Le point  $X$  appartenant à ce plan, d'où (en notations algébriques) :

$$\vec{N} \cdot \vec{X} = \vec{N} \cdot \vec{P}$$

D'autre part, soient  $C$  et  $D$  deux points du rayon optique ( $C$  : centre optique de la caméra) passant par  $X$ , alors :

$$X = C + \lambda \vec{CD} \quad \text{ou } \lambda \text{ est une inconnue et } \vec{CD} \text{ le vecteur directeur du rayon optique.}$$

D'où :

$$\lambda = \frac{\vec{N} \cdot \vec{P} - \vec{N} \cdot \vec{C}}{\vec{N} \cdot \vec{CD}}$$

On obtient alors  $X$ .

#### **Détermination des courbes enveloppes:**

Pour calculer l'image  $X'$  d'un point  $X$ , du contour de base, par la rotation d'angle  $\theta$  autour d'un axe dirigé par un vecteur  $\vec{A}$ , on calcule la matrice de passage  $P$  du repère de travail à un repère orthonormé dont  $\vec{A}$  est l'axe des abscisses, on a alors :

$$X' = \text{rot}_{\vec{A}, \theta}(X) = P^{-1} \cdot R \cdot P \cdot X$$

où  $R$  est la rotation d'angle  $\theta$  autour de  $\vec{A}$ .

Cette méthode présente l'avantage de ne nécessiter aucun appariement. En revanche, l'acquisition des points caractéristiques de l'axe de rotation est contraignante. Cette approche de modélisation des corps de révolution est une 1<sup>ère</sup> ébauche d'un travail qui doit se poursuivre afin de permettre une modélisation plus précise et plus rapide.

Une fois les objets de l'environnement modélisés, les données les représentant sont intégrées dans une structure de données informatique. Si ces objets changent de place, leurs données nécessitent une mise à jour. Au lieu de refaire une modélisation qui est coûteuse en temps comme nous l'avons vu dans le paragraphe précédent,

il est plus judicieux de calculer la transformation subie par les objets qui ont changé de place. C'est l'objet du prochain paragraphe.

#### **Conclusion**

Les méthodes de modélisation 3D de volumes englobants présentées sont basées sur l'utilisation des capteurs. Cette modélisation permet d'introduire des objets non préalablement modélisés dans la base de données. Cet ajout en ligne d'objet permet à la tâche effectuée par le robot de se poursuivre. Ces méthodes ont permis également d'évaluer la précision du relevé 3D basé sur l'utilisation de différentes combinaisons de capteurs. Par ailleurs, la connaissance de la géométrie des objets permet de remettre à jour celle-ci si ceux-ci changent de situation. Nous avons présenté quelques méthodes de reconstruction 3D représentatives de l'état de l'art. En effet, nous avons présenté une méthode géométrique, une seconde résolution valable pour les faibles mouvements, ce qui permet de linéariser le problème et d'introduire des informations redondantes, une troisième numérique valable quelque soit le mouvement subis par l'objet et une dernière mixte.

### 3. Reconstruction 3D: Mise à jour de la représentation informatique de l'environnement

Ce paragraphe répond à la question : "comment mettre à jour les données 3D des objets ?"

#### 3.1. Introduction

La reconstruction 3D est basée sur la détermination des transformations géométriques rigides qui permettent d'obtenir la nouvelle situation de l'objet dans l'espace. L'obtention de celle-ci permet d'effectuer un recalage de l'objet, c'est à dire la mise en correspondance de ses représentations réelle et virtuelle. Les données 3D de la situation antérieure de l'objet à recalcr ainsi que ses données 2D (extraites de l'image caméra) de la position courante sont supposées connues.

L'approche de reconstruction 3D décrite est basée sur la détermination des transformations géométriques rigides qui permettent d'obtenir la nouvelle situation de l'objet dans l'espace. L'obtention de celle-ci permet d'effectuer un recalage de l'objet, c'est à dire la mise en correspondance de ses représentations réelle et virtuelle.

On distingue deux types de recalages qui dépendent de l'ampleur du mouvement :

- recalage à grande amplitude dans le cas de grand mouvement,
- recalage à faible amplitude ou fin pour petit mouvement.

Ces deux types de recalages sont applicables successivement à un objet. Deux cas de figure peuvent se présenter:

① Le cas de la création de l'objet, cette création peut se faire de deux manières:

- les dimensions de l'objet sont fournis par l'opérateur, dans ce cas en général, l'écart entre la représentation graphique obtenue de l'objet et sa représentation réelle peut être important.

- les coordonnées de l'objet sont fournies par les capteurs, dans ce cas l'écart entre la représentation graphique obtenue de l'objet et sa représentation réelle est généralement faible. Dans ce cas nous n'appliquons qu'un recalage fin.

② Le cas de la modification de l'environnement. L'objet étant modélisé et ses coordonnées 3D intégrées dans la base de données. Il s'agit de retrouver sa nouvelle situation en appliquant le recalage.

Dans le cas d'une aide à la commande (exemple d'une réalisation de tâche), il faut permettre au robot de manipuler correctement les objets présents dans son environnement; pour cela il est indispensable de bien situer ces objets par rapport à un repère de référence. Egalement dans le cas d'une assistance à la perception de la scène en vision indirecte, on superpose au monde réel un monde virtuel grâce à la base de données géométriques 3D des objets de l'environnement. Donc si la situation du monde réel change, les mondes réel et virtuel ne sont plus superposés; il faudra alors procéder à un recalage.

Nous présentons un état de l'art contenant les principaux travaux portant sur la reconstruction 3D polyédrique.

Deux approches sont utilisées pour la reconstruction 3D polyédrique:

- Une approche numérique basée sur l'utilisation d'un nombre important de données tout en cherchant à optimiser un critère de mise en correspondance du modèle de l'objet avec son image. Les méthodes correspondantes sont citées dans [HOR 93].
- Une approche analytique basée sur la résolution d'équations permettant d'obtenir la nouvelle situation de l'objet géométriquement [DHO 89].

Ganapathy [GAN 84] considère simplement les neuf composantes de la matrice de rotation de manière indépendante. Le système étant linéaire, il est facile à résoudre. Cependant, aucune contrainte d'orthogonalité n'est appliquée à la matrice de rotation, de plus cette méthode est extrêmement sensible au bruit.

Lowe [LOW 87] a proposé une méthode itérative fournissant une solution unique en manipulant un triplet de segments de droite. Celle-ci est obtenue en optimisant, par la méthode de Newton, d'un critère d'erreur quadratique. Celui-ci est constituée par la somme des carrés des distances entre chaque projection d'un segment et de son correspondant dans l'image. L'optimisation pose des problèmes de minima locaux et d'initialisation des paramètres à estimer.

Yuan [YUA 89] a séparé la rotation de la translation et a proposé de déterminer la rotation en tenant compte des contraintes d'orthogonalité de la matrice de rotation. La solution commune à ces contraintes est obtenue en utilisant la méthode de Newton. L'auteur a noté que des minima locaux peuvent être trouvés avec cette méthode. Le minimum global peut être atteint si on fournit une initialisation proche de la solution.

Dhome [DHO 89] a proposé une méthode géométrique basée sur l'utilisation d'un triplet de segments de droite. Cette méthode utilise des repères intermédiaires permettant de simplifier le calcul de la rotation. Des règles sont utilisées pour réduire le nombre de solutions.

Horaud [HOR 93] a proposé une méthode qui minimise un critère basé sur l'utilisation du formalisme  $N.R.V$  décrit dans le § 3.2. La rotation est représentée par les quaternions unitaires. L'optimisation utilise l'algorithme de la région de confiance. Il s'avère que l'algorithme est très rapide quand il converge, mais qu'il faut l'initialiser avec de bonnes conditions initiales pour obtenir le résultat attendu. Cet algorithme semble avoir une bonne robustesse au bruit.

Nous allons décrire quatre méthodes de reconstruction 3D représentative de l'état de l'art :

La 1<sup>ère</sup> est géométrique qui consiste à faire un choix judicieux de repères afin de simplifier les calculs de la transformation rigide. Elle basée sur la méthode de Dhome [DHO 89] (§ 3.3). Cette méthode peut être utilisée uniquement pour les objets polyédriques.

La 2<sup>ème</sup> est numérique est basée sur la dérivation du modèle de la caméra. Cette méthode est valable uniquement pour les faibles mouvements de l'objet (§ 3.4).

La 3<sup>ème</sup> est numérique, elle est valable quelque soit l'amplitude du mouvement subis par l'objet (§ 3.5).

La 4<sup>ème</sup> est mixte (§ 3.5).

Avant de décrire ces méthodes, nous rappelons le principe de la reconstruction 3D basée sur les plans d'interprétation.

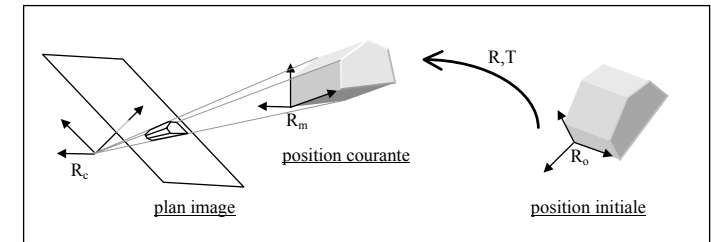


Figure 3.1. Reconstruction 3D

#### 3.2 Le problème

L'objectif de la reconstruction polyédrique 3D est de retrouver la rotation et la translation subies par un objet dans l'espace (Figure 3.1.).

La situation initiale de l'objet est exprimée dans  $(R_0)$ . Il s'agit de calculer  $R$  et  $T$ , c'est à dire la situation de  $(R_m)/(R_0)$  et donc la nouvelle situation de l'objet.

Toutes les méthodes de localisation tridimensionnelle reposent sur des bases mathématiques communes exploitant des indices visuels 2D provenant de la caméra. Parmi tous les indices visuels, les droites sont facilement extractibles, ce sont de plus des indices robustes. C'est la raison pour laquelle ce sont les plus utilisés.

La caméra étant considérée calibrée, on connaît donc la correspondance entre les points dans l'espace et leurs projections sur le plan rétinien. Le vecteur 3D,  $\vec{V}'$  (Figure 3.1) donne le vecteur  $\vec{v}'$  par projection perspective de centre  $C$  sur le plan rétinien. A l'aide des coordonnées 2D, on définit le plan d'interprétation qui passe par le vecteur  $\vec{v}'$  et le centre optique  $C$ . Ce plan contient également le vecteur  $\vec{V}'$  de l'espace,  $\vec{N}$  étant la normale à ce plan.

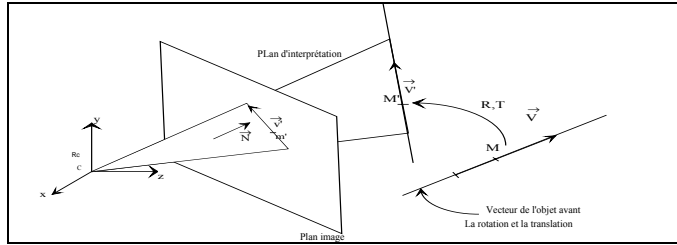


Figure 3.2. Formalisme N.R.V

La localisation fait intervenir trois repères principaux (Figure 3.1.)

- ❶ (Rc) repère lié à la caméra.
- ❷ (Rm) repère lié à l'objet représentant la position courante.
- ❸ (Ro) repère de travail, représente (Rm) à l'instant initial.

Localiser l'objet évoluant dans son environnement revient à déterminer la position de (Rm) par rapport à (Ro). Les repères (Rc) et (Ro) sont connus l'un par rapport à l'autre. On supposera par la suite pour simplifier les expressions, mais sans rien enlever à la généralité du problème, que (Ro) est confondu avec (Rc). L'estimation de la transformation amenant (Rm) à (Ro) revient à déterminer une rotation R et une translation T. On peut alors connaître les coordonnées de l'objet dans le repère (Ro) donc dans (Rc). On a ainsi pour tout point M et pour tout vecteur  $\vec{V}$  de l'objet exprimés dans (Rm), leurs expressions respectives  $M/Rc$  et  $\vec{V}/Rc$  dans (Rc) :

$$M/Rc = T + RM/Rm \quad [1]$$

$$\vec{V}/Rc = R\vec{V}/Rm \quad [2]$$

A partir de ces relations, on peut poser les équations de base de la localisation. Pour un vecteur 3D,  $\vec{V}'$ , auquel on associe son projeté  $\vec{v}'$  et la normale  $\vec{N}$  au plan d'interprétation on a :

$$\vec{N} \cdot \vec{V}' = 0 \quad [3]$$

$$\vec{N} \cdot \vec{CM}' = 0 \quad [4]$$

Les vecteurs présents dans ces deux relations sont exprimés dans une même base. Comme les expressions sont données dans (Rc), repère de centre C; en tenant compte des équations [1] et [2] on obtient :

$$\vec{N} \cdot (R\vec{V}) = 0 \quad [5]$$

$$\vec{N} \cdot (T + RM) = 0 \quad [6]$$

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

Le problème est donc le suivant: étant donné un ensemble de points 3D décrits dans un référentiel 3D "Objet" et leurs projections 2D décrites dans un référentiel "caméra". Si on connaît les paramètres de la caméra, il faut déterminer la transformation rigide (rotation et translation) entre le référentiel objet à un instant donné et le référentiel caméra.

Les solutions proposées peuvent se regrouper en deux catégories:

- ❶ Une solution numérique, Le système d'équation à résoudre est non linéaire. Ce type d'approche permet de tenir compte des données redondantes obtenues quand le nombre de points de correspondance est grand. Ces méthodes numériques sont basées sur la résolution des équations [5] et [6] en minimisant un critère d'erreur qui

est, généralement, la somme des carrés des distances entre chaque projection d'un point (ou d'une arête) et de son correspondant(e) dans l'image.

L'inconvénient majeur de cette approche est la lenteur de convergence des algorithmes d'optimisation.

- ❷ Une solution analytique est présentée dans le paragraphe suivant.

### 3.3 Méthode géométrique des 3 segments

La détermination directe de l'orientation de (Rm)/(Ro) est complexe car elle fait intervenir trois angles. Afin de simplifier les calculs, une approche analytique présentée dans [DHO 89] consiste à introduire des repères intermédiaires permettant de ne faire intervenir que deux angles.

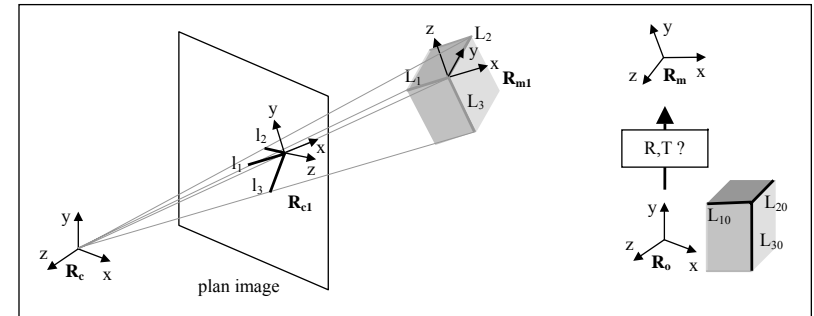
Soient trois arêtes  $L_{i=1,2,3}$  de l'objet définis dans (Rm), et leurs images  $I_{i=1,2,3}$  définies dans (Rc). L'objectif est de déterminer la rotation R et la translation T à appliquer à  $L_i$  afin que leurs images se projettent sur les droites supports de  $I_i$  (Figure 3.3). Le principe est de résoudre les deux équations [5] et [6].

Les différents repères mis en jeu sont :

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

Les matrices de passage suivantes sont définies :

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



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

$$R = Roc1 Rc1m1 Rm1m \quad [7]$$

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

Le développement des équations [ 5]et [ 6] aboutit à un polynôme de degré 8 , auquel peut être

**Figure 3.3.** Les différents repères mis en jeu

appliqué une méthode numérique basée sur le calcul de valeurs propres de la matrice "compagnon" construite à partir du polynôme [NZI 95]. Les angles  $\alpha$  et  $\beta$  qui caractérisent la matrice Rc1m1 sont obtenus. La rotation R subie par l'objet est ensuite calculée par l'équation [ 7]. Une fois la rotation obtenue, la translation est déduite avec également trois points appartenant aux segments 3D et leur images. 3 points formant trois segments sont utilisés pour appliquer la méthode.

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

**3.4. Méthode linéaire à petit mouvement**

L'objectif est d'affiner le recalage à grande amplitude, à l'aide d'une méthode de reconstruction appliquée aux petites rotations, pour réduire les erreurs [NZI 97]. Le vecteur instantané de rotation utilisé, ensuite la méthode de calcul de R et T sont définis.

• **VECTEUR INSTANTANE DE ROTATION**

Une rotation quelconque peut être représentée par trois angles  $\theta_x, \theta_y, \theta_z$  dénommés *angles nautiques*. Chacun de ces angles quantifie la rotation autour d'un axe d'un repère donné. L'avantage de ce formalisme est qu'il représente directement la commande de l'articulation d'un robot. L'inconvénient est que le produit de ces trois rotations n'est pas commutatif, l'ordre dans lequel s'effectuent ces rotations influe sur la rotation finale.

Egalement, il présente pour le problème inverse plusieurs solutions et quelques positions singulières

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

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

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

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

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

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

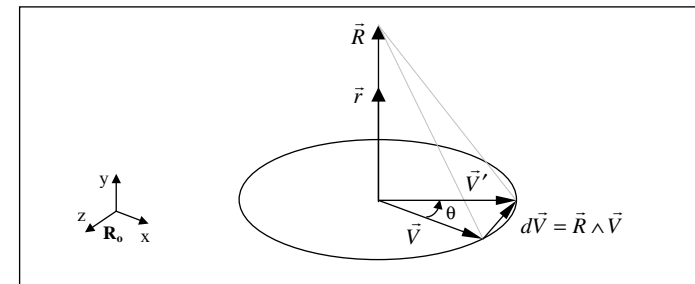
• **DETERMINATION DE R ET T**

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

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

$\vec{R}$  étant le vecteur instantané de rotation, l'écriture de cette équation revient à assimiler l'arc issu de la rotation à la tangente (Figure 3.4).

Pour un point P, ces expressions deviennent :



**Figure 3.4.** Approximation des petits angles

$$dP = T + \vec{R} \times P \quad [ 8]$$

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

$$\begin{cases} n_u \cdot P + a_u = 0 \\ n_v \cdot P + a_v = 0 \end{cases}$$

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

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

dp est donné par l'équation [3.7] et en utilisant une propriété du produit mixte de trois vecteurs i, j, k quelconques :  $i.(jAk) = j.(kAi)$  ces deux équations sont écrites de la manière suivante :

$$\begin{cases} n_u.T + (P \times n_u).R + dn_u.P + da_u = 0 \\ n_v.T + (P \times n_v).R + dn_v.P + da_v = 0 \end{cases} \quad [9]$$

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

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

La résolution de ce système linéaire permet de déterminer R et T. Pour obtenir les 6 inconnues un minimum de n=3 points est nécessaire. Ces trois points doivent avoir des images distinctes et non alignées. Le système peut être résolu par une méthode de type moindres carrés lorsque n>3. Ce recalage dit de petit déplacement permet ainsi de réduire les erreurs obtenues sur la méthode de recalage pour les grands mouvements (rotation de : 0 à  $2\pi$  et translation quelconque) et peut être itérer.

### 3.5. Méthode non-linéaire

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

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

$$\text{Critère} = \text{Min} \left[ \sum_{i=1}^N \left( (\hat{u}_i - u_i)^2 + (\hat{v}_i - v_i)^2 \right) \right] \quad [10]$$

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

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

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

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

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

Le vecteur de paramètres à estimer  $\vec{p}$  est constitué des paramètres de rotation et de translation.

L'application d'une méthode numérique au critère [10] permet d'identifier les paramètres recherchés à partir des points 3D et de leurs images 2D. La méthode d'optimisation de Levenberg-Marquardt [Numerical recipes] est utilisée, car son coefficient de réglage  $\lambda$  permet d'éviter les minima locaux.

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

### 3.5. Méthode employée (Mixte)

L'évaluation des trois méthodes de recalage étudiées ci-dessus [SHA 99], a permis d'adopter la méthode mixte. Celle-ci utilise les trois méthodes pour assurer la convergence vers l'optimum quasiment dans tous les cas. En fait, les méthodes linéaire et non-linéaire nécessitent une bonne initialisation contrairement à la méthode des 3 segments. En revanche, Elles donnent toutes les deux une meilleure précision que cette dernière. Nous utilisons alors la méthode des trois segments qui donne une solution approximative pour initialiser les deux autres méthodes. En tant que précision, la méthode linéaire est meilleure que la non-linéaire, car elle donne une solution analytique optimale dans le sens du critère, tandis que l'autre converge souvent vers des minima locaux. Par ailleurs la méthode linéaire représente des positions singulières que la méthode non-linéaire ne subit pas.

Voici l'algorithme qui décrit la méthode mixte :

Appliquer la méthode des trois segments pour obtenir la matrice de recalage (les trois segments en triangle doivent être non-colinéaires : la superficie du triangle > seuil)

Utiliser la méthode linéaire pour affiner le recalage,

- si le critère d'erreur est suffisamment petit : le résultat est considéré bon, fin
- sinon : position singulière, continuer

Chercher la convergence à l'aide de la méthode non-linéaire,

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

Appliquer à nouveau la méthode linéaire,

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

- sinon : pas de solution, fin

**3.6. Méthode de l'itération orthogonale [Lu et Hager, 2000]**

**3.6.1 Définition du problème**

Soit un ensemble de points de référence 3D non colinéaires  $p_i = (x_i, y_i, z_i), i=1, \dots, n \quad n \geq 3$  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:

$$q_i = R \cdot p_i + T \tag{11}$$

où

$$R = \begin{pmatrix} r'_1 \\ r'_2 \\ r'_3 \end{pmatrix} \text{ tel que } R^t R = I \quad \text{et} \quad T = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \tag{12}$$

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 3.5) :

- ( $R_c$ ) 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>1</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 3.5). 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:

$$u_i = \frac{r'_1 p_i + t_x}{r'_3 p_i + t_z} \tag{13}$$

$$v_i = \frac{r'_2 p_i + t_y}{r'_3 p_i + t_z} \tag{14}$$

ou bien

$$m_i = \frac{1}{r'_3 \cdot p_i + t_z} (R \cdot p_i + T) \tag{15}$$

<sup>1</sup> Nous supposons que la caméra a été préalablement calibrée, et de ce fait, ses paramètres intrinsèques et extrinsèques sont connus.

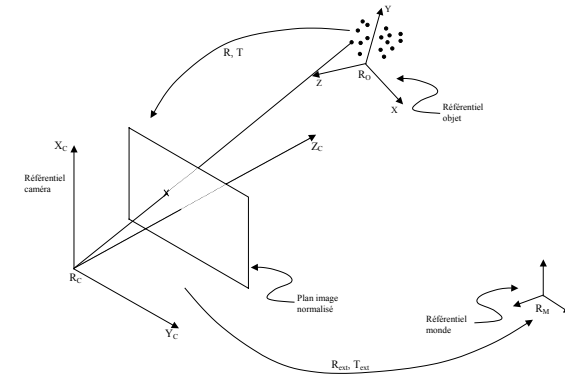


Figure 3.5 : Les différents repères mis en jeu dans le problème de recalage 3D

l'équation [15] 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 :

$$R \cdot p_i + T = V_i \cdot (R \cdot p_i + T) \tag{16}$$

où

$$V_i = \frac{m_i \cdot m_i^t}{m_i^t \cdot m_i} \tag{17}$$

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 [15] représente l'équation de colinéarité dans l'espace image tandis que l'équation [16] 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 3.6).

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

$$Q = \sum_{i=1}^n \left[ \left( \hat{u}_i - \frac{r'_1 p_i + t_x}{r'_3 p_i + t_z} \right)^2 + \left( \hat{v}_i - \frac{r'_2 p_i + t_y}{r'_3 p_i + t_z} \right)^2 \right] \tag{18}$$

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].

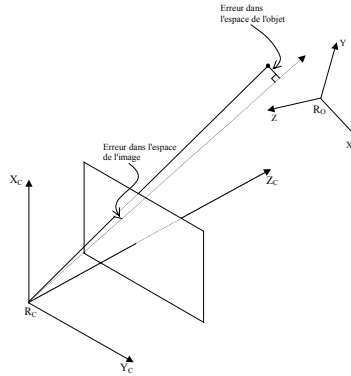


Figure 3.6 : Les erreurs de colinéarité dans l'espace image et l'espace objet

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.

### 3.6.2 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.

### 3.6.3 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:

$$q_i = R \cdot p_i + T \quad [19]$$

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 :

$$\min_{R,T} \sum_{i=1}^n \|R \cdot p_i + T - q_i\|^2 \quad \text{tel que } R \cdot R^t = I \quad [20]$$

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:

$$\bar{p} = \frac{1}{n} \sum_{i=1}^n p_i \quad \bar{q} = \frac{1}{n} \sum_{i=1}^n q_i \quad [21]$$

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

$$p'_i = p_i - \bar{p} \quad q'_i = q_i - \bar{q} \quad [22]$$

et la matrice

$$M = \sum_{i=1}^n q'_i p'_i{}^t \quad [23]$$

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 [20] alors elles satisferont:

$$R^* = \arg \max_R \text{trace}(R^t M) \quad [24]$$

$$T^* = \bar{q} - R^* \cdot \bar{p} \quad [25]$$

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 [20] est donnée par:

$$R^* = V U^t \quad [26]$$

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 [23]. 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.

### 3.6.4 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):

$$e_i = (I - \hat{V}_i)(R.p_i + T) \quad [27]$$

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:

$$\hat{V}_i = \frac{\hat{m}_i \hat{m}_i^t}{\hat{m}_i^t \hat{m}_i} \quad [28]$$

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

$$E(R, T) = \sum_{i=1}^n \|e_i\|^2 = \sum_{i=1}^n \|(I - \hat{V}_i)(R.p_i + T)\|^2 \quad [29]$$

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 :

$$T(R) = \frac{1}{n} \left( I - \frac{1}{n} \sum_{j=1}^n \hat{V}_j \right)^{-1} \sum_{j=1}^n (\hat{V}_j - I) R.p_j \quad [30]$$

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

$$q_i(R) = \hat{V}_i(R.p_i + T(R)) \quad \text{et} \quad \bar{q}(R) = \frac{1}{n} \sum_{i=1}^n q_i(R) \quad [31]$$

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

$$E(R) = \sum_{i=1}^n \|R.p_i + T(R) - q_i(R)\|^2 \quad [32]$$

cette équation ressemble à l'équation [29] 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:

$$M(R) = \sum_{i=1}^n q_i(R).p_i^t \quad \text{où} \quad p_i = p_i - \bar{p} \quad \text{et} \quad q_i(R) = q_i(R) - \bar{q}(R) \quad [33]$$

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  $q_i^{(k)} = R^{(k)}.p_i + T^{(k)}$ . L'estimée suivante,  $R^{(k+1)}$ , est déterminée en résolvant le problème de l'orientation absolue suivant:

$$R^{(k+1)} = \arg \min_R \sum_{i=1}^n \|R.p_i + T - \hat{V}_i.q_i^{(k)}\|^2 = \arg \max_R \text{tr}(R^t M(R^{(k)})) \quad [34]$$

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 [20]. Sous cette forme, la solution pour  $R^{(k+1)}$  est donnée par l'équation [26]. Nous calculons ensuite l'estimée suivante ( $k+1$ ) de la translation en utilisant l'équation [30], telle que:

$$T^{(k+1)} = T(R^{(k+1)}) \quad [35]$$

et nous répétons le processus. La solution optimale  $R^*$  est obtenue lorsque l'équation [34] reste inchangée, en d'autres termes, lorsque  $R^*$  satisfait:

$$R^* = \arg \min_R \sum_{i=1}^n \|R.p_i + T - \hat{V}_i(R^*.p_i + T(R^*))\|^2 \quad [36]$$

L'avantage de cet algorithme est qu'il converge globalement vers l'optimum de l'équation [34] 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:

$$A = A_{ext}^{-1}.A_{opt} \quad [37]$$

où

$$A_{ext} = \begin{pmatrix} R_{ext} & T_{ext} \\ 0 & 1 \end{pmatrix} \quad \text{et} \quad A_{opt} = \begin{pmatrix} R^* & T^* \\ 0 & 1 \end{pmatrix} \quad [38]$$

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.

**4. Représentation informatique des données de l'environnement**

Le modèle informatique de la scène constitue l'élément indispensable à tout système d'aide à la perception et à la commande. En effet, il permet de traduire les caractéristiques de la scène sous une forme exploitable par le calculateur.

L'environnement de téléopération est constitué de deux types d'objets: les objets connus (robots, outils, pièces à manipuler, ...) et les objets non connus (obstacles,...).

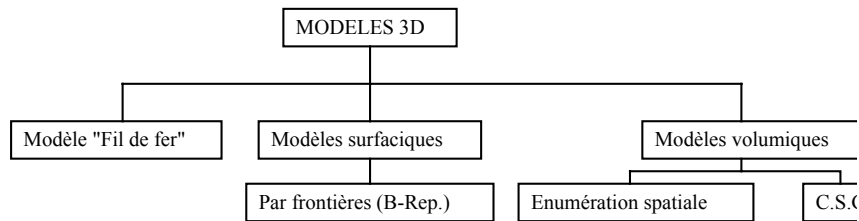
Concernant les objets connus, leur géométrie est bien définie a priori. L'opérateur peut intégrer hors ligne le modèle de ces objets dans la base de données géométriques 3D (BD3D).

Dans le cas où la position de l'objet de ce type n'est pas connue a priori, ou bien s'il est déplacé, l'opérateur utilise les capteurs d'environnement (caméra et/ou DR3D) pour mettre à jour cette base de données. Pour les objets non connus une coopération entre l'opérateur humain et le système est établie pour construire le modèle informatique de la scène.

Nous présentons brièvement les principales méthodes de modélisation informatique d'objets solides . Les structures de données associées sont également présentées ainsi que l'organisation des données topologiques.

Nous définissons le modèle informatique comme une représentation de l'objet comportant des informations de classe (pyramide, cylindre, corps de révolution), de géométrie (coordonnées 3D, équations,...), de topologie(liens entre les éléments géométriques), ...

L'acquisition des données sensorielles permet de prélever l'information du monde réel, par l'intermédiaire de divers capteurs. De ces données brutes (informations de profondeur, de luminance, ...) doit être extraite une description symbolique, de plus ou moins haut niveau, utilisable (point 3D, droite, plan, cylindre,...). Cette opération de description des données perceptuelles est appelée modélisation. Ces données peuvent être organisées sous différentes formes appelées modèles[REQ 80] (figure 4.1).



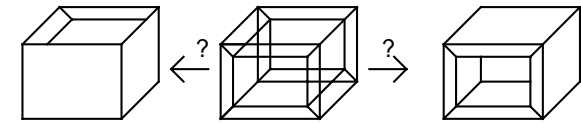
**Figure 4.1. Taxinomie des modèles 3D**

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

**4.1. Représentation filaire**

Le modèle " fil de fer" ("wire frame") retient de l'objet les coordonnées des sommets et les arêtes les joignant. La figure 4.2 montre les ambiguïtés qui peuvent découler de son utilisation, puisque ne connaissant que les arêtes et les sommets, plusieurs interprétations d'un même modèle peuvent être faites. D'autre part, les calculs géométriques (dimensions, volumes) sont difficiles, voire impossibles car les informations concernant les surfaces des objets sont inexistantes.

Bien que rarement utilisée, cette méthode de modélisation a l'intérêt de permettre une création et une visualisation rapide de la scène, car elle permet une modification aisée des points et des arêtes. La place mémoire utilisée est également très faible.



**Figure 4.2. Ambiguïté du modèle "fil de fer"**

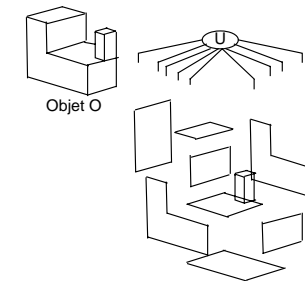
**4.2. Représentations surfaciques**

Ce modèle représente l'objet par ses bords. Le modèle le plus utilisé est la B-REP ("Boundary REPresentation").

Les surfaces représentées peuvent être : planes, surfaces de révolution, surfaces super/hyper quadriques.

La scène est construite interactivement par assemblage de primitives géométriques de base qui sont: point, segment, polygone et polyèdre. Les liaisons entre ces primitives sont définies par des relations topologiques d'incidence, de contiguïté et d'inclusion (figure 4.3).

La principale méthode utilisée pour l'élaboration de polyèdre consiste à balayer un contour de base plan ou courbe le long d'une trajectoire linéaire, circulaire ou quelconque permettant de modéliser le solide



correspondant.

**Figure 4.3. La B-REP ("Boundary REPresentation").**

**4.3. Représentations volumiques**

La représentation *constructive solid geometry* CSG : Il s'agit d'une méthode de construction d'objets par l'application d'opérations d'assemblage d'où l'appellation. La construction de l'objet se fait par combinaison de volumes élémentaires. La démarche est semblable à celle de l'artisan, qui usine les pièces élémentaires, les assemble pour obtenir une plus complexe (Figure 4.4). Elle est décrite par un arbre de construction où les feuilles sont des solides de base et les nœuds sont des opérations comme: l'union, l'intersection, la différence, les transformations géométriques composées de translation ou de rotation.

L'objet est construit à partir de primitives géométriques de base ou volumes élémentaires (cube, cylindre...) auxquels sont appliqués les opérateurs logiques afin de les assembler et de créer d'autres volumes. Les primitives géométriques de base sont paramétrées. Les données canoniques simples caractérisent chacune des primitives (centre, rayon et hauteur pour le cylindre par exemple).

La liste des modèles présentée n'est pas exhaustive. Les plus utilisées sont la CSG et la B-REP. Plusieurs méthodes peuvent être associées les unes aux autres: par exemple la CSG et la B-REP. La représentation CSG permet plus facilement de créer une représentation de la scène que la B-REP. Cette simplicité d'utilisation se fait au détriment de la rapidité d'animation de l'image. En effet, les données canoniques simples ayant servi à la représentation de la scène ne sont pas utilisées directement par le matériel graphique. Une conversion de ces données en points et segments est nécessaire. De plus, le calcul des surfaces très utiles pour les applications de robotique, est difficile avec la CSG, puisque les formules mathématiques de calcul de surface nécessitent la connaissance de points de frontières. Il est certain que la disponibilité des deux méthodes est souhaitable. La CSG permettrait la saisie des données pour la création d'une représentation de la scène et la B-REP l'animation et la visualisation. La solution retenue est celle utilisant la B-REP, elle est plus générale car elle permet de modéliser un grand nombre d'objets et plus proche des systèmes graphiques d'où une animation rapide de l'image synthétique.

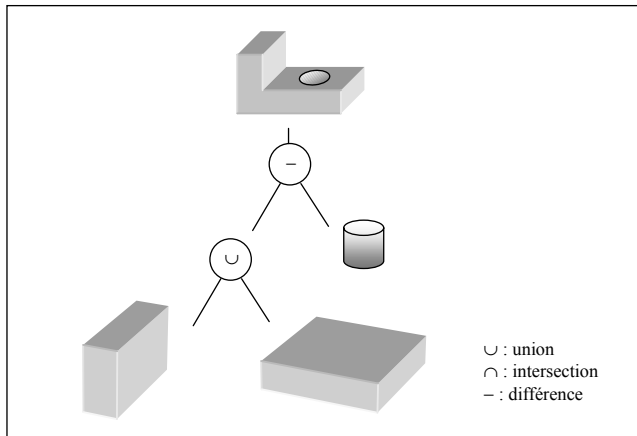


Figure 4.4. Représentation CSG simple

**4.4 Codage des données géométriques de l'environnement**

L'algorithmique graphique ou synthèse d'images est née peu de temps après l'informatique. Les premiers algorithmes et logiciels datent de 1963 (algorithme d'élimination des lignes cachées de L.G. Roberts - logiciel SKETCHPAD de I.E. Sutherland). Pourtant aujourd'hui, cette discipline occupe encore de nombreux chercheurs. C'est dire la complexité du problème posé : fabriquer des images, le plus proche de la réalité, ou plutôt faut-il dire, de la vision humaine.

Le but de ce document est de résumer les principales techniques utilisées dans la synthèse d'images et de proposer les principales références pour une étude approfondie. Le processus menant à l'image de synthèse se décompose en trois parties. La première est la construction du modèle représentant l'univers à visualiser. La seconde essentiellement algorithmique, est celle de l'introduction d'un observateur et du calcul des parties visibles du modèle. La troisième précise les mécanismes de la vision en fonction de l'éclairage. Avant d'aborder les différents chapitres, la figure illustre les structures de données nécessaires aux différentes transformations de l'image.

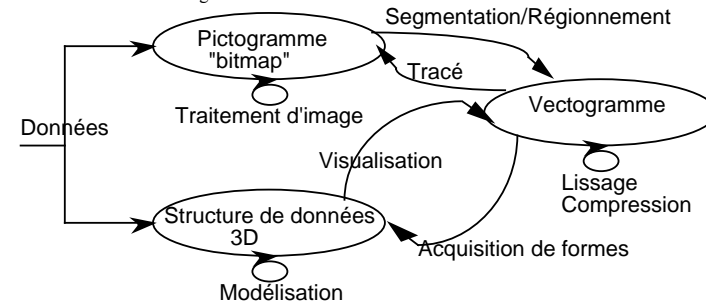


Figure 4.5 : Transformations de l'image

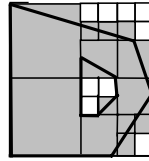
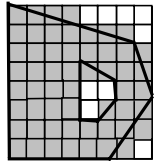
**4.4.1 - La modélisation bidimensionnelle - Quadtree**

**4.4.1.1. Définition**

Le quadtree, ou arbre quaternaire, est utilisé pour représenter informatiquement des objets 2D. A l'espace considéré est superposée une grille constituée de 2\*\*q x 2\*\*q cellules (q entier positif, représentant la profondeur de l'arbre). Les feuilles de l'arbre représentent des parties homogènes(remplies/vides) de l'espace modélisé et les nœuds représentent des parties mixtes (partiellement remplies/vides).

**4.4.1.2 Exemple**

Dans cet exemple, est représenté le quadtree correspondant à un polygone constitués par une liste de points coplanaires, le premier point étant relié au dernier, et une liste d'arêtes reliant les points entre eux. Les nœuds du quadtree sont divisés en 3 classes :  
 . nœud intérieur (feuille) : représente les points intérieurs au polygone(symbolisé par un carré noir),  
 . nœud extérieur (feuille) : représente les points n'appartenant pas au polygone (symbolisé par un carré blanc),  
 . nœud frontière : représentent les cellules de la grille qui intersectent la frontière du polygone.



1	2
3	4

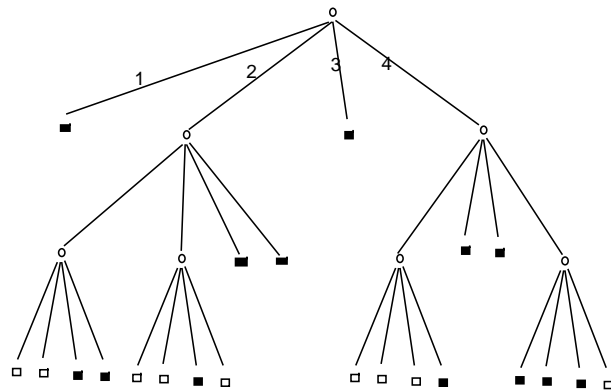


Figure 4.6: Quadtree

4.4.1.3 Structures de données

. Linéaire statique

L'arbre est représenté sous forme d'un tableau linéaire. Ci-dessous un exemple de structure représentant un nœud/feuille de l'arbre :

```
struct t_nœud {
    int profondeur ;
    int couleur ;
}
```

Exo. : calculer la place mémoire occupé par l'arbre de l'exemple 4.4.1.2.

Cette structure de données est valable pour la visualisation de la scène. Si un usinage d'une pièce mécanique, par exemple est nécessaire, une allocation dynamique des données est préférable.

. Structure chaînée de tableaux 1D

L'arbre est représenté cette fois-ci par des tableaux à une dimension chaînés entre eux. Chaque tableau est constitué de 4 entrées correspondantes à 4 fils. Chaque fils pointant sur sa descendance.

```
Struct t_freres { union {
    struct t_freres *fils ;
```

```
int couleur ;
} freres[4]
}
```

Exo. : calculer la place mémoire occupé par l'arbre de l'exemple 4.4.1.2.

. Structure chaînée de tableaux 2D (1 à 4)

L'idée de cette structure est de faire pointer chaque génération sur sa descendance directe :

```
Struct freres {
    struct generation *ptr;
    int couleur[4];
}
Struct generation {
    struct freres T[4] ;
}
```

Exo. : calculer la place mémoire occupé par l'arbre de l'exemple 4.4.1.2.

4.4.1.4 Opérations sur quadtree

Soient 3 quadrees de même dimension q1, q2, et q3, chaque nœud n de ces quadtree peut prendre 3 valeurs, n := (N/B/G). N(noir) correspond à un nœud intérieur, B( blanc) correspond à un nœud extérieur et G(gris) correspond à un nœud frontière.

Les algorithmes suivants effectuent une opération logique sur q1, q2 et stockent le résultat dans q3.

. Union de 2 quadrees

```
si n(q1) = N, ∀ n(q2) alors n(q3) := N
si n(q2) = N, ∀ n(q1) alors n(q3) := N
si n(q1) = B, ∀ n(q2) alors n(q3) := n(q2)
si n(q2) = B, ∀ n(q1) alors n(q3) := n(q1)
si n(q1)=n(q2)=G alors parcourir
```

. Intersection de 2 quadrees

```
si n(q1) = N, ∀ n(q2) alors n(q3) := n(q2)
si n(q2) = N, ∀ n(q1) alors n(q3) := n(q1)
si n(q1) = B, ∀ n(q2) alors n(q3) := B
si n(q2) = B, ∀ n(q1) alors n(q3) := B
si n(q1)=n(q2)=G alors parcourir
```

Exercice : opérations booléennes sur les quadrees

Ecrire l'algorithme qui réalise un OU-EXCLUSIF entre deux quadrees q1, q2 et place le résultat dans q1. Les fenêtres associées aux nœuds racines des deux quadrees sont supposées identiques.

Vous disposez des fonctions suivantes :

Fonctions d'interrogations

Fonctions de construction

```
Interrogation ou positionnement du nœud racine d'un quadree
inq_racine(q : quadtree) : nœud set_racine(q : quadtree, n : nœud) : (OK, ERREUR)

Interrogation ou positionnement des fils d'un nœud
(no : nord-ouest, ne : nord-est, so : sud-ouest, se : sud-est)

inq_no(n : nœud) : nœud set_no(n, no : nœud) : (OK, ERREUR)
```

inq\_ne(n : nœud) : nœud  
 inq\_so(n : nœud) : nœud  
 inq\_se(n : nœud) : nœud

set\_ne(n, ne : nœud) : (OK, ERREUR)  
 set\_so(n, so : nœud) : (OK, ERREUR)  
 set\_se(n, se : nœud) : (OK, ERREUR)

Interrogation ou positionnement de la couleur d'un nœud  
 (N : Noir, B : Blanc, G : Gris)

inq\_couleur(n : nœud) : (N, B, G)      set\_couleur(n : nœud, c : (N,B,G)) : (OK, ERREUR)

4.4.2. - La modélisation tridimensionnelle

4.4.2.1. Introduction

La première étape du processus de création d'images de synthèse est la modélisation. Elle définit les structures de données utilisées par les algorithmes d'élimination des parties cachées, d'ombrage et d'éclairage.

Ce chapitre se limite à l'étude des représentations d'objets solides. Un objet solide est rigide, sa forme est invariante par des transformations affines telles que la rotation ou la translation. Il est tridimensionnel, possède un intérieur et sa frontière ne contient pas d'arêtes ou de faces pendantes. Il est fini et forme une partie bornée de  $\mathfrak{R}^3$ .

Les principaux critères de validité d'un modèle sont l'unicité et la non-ambiguïté. Tout modèle réunit ces deux conditions s'il réalise une bijection entre l'ensemble des solides représentables et l'image qu'il donne de cet ensemble (1).

Nous aborderons successivement les différents modèles existants, les conversions qui leur sont associées et enfin les opérateurs ou modeleurs permettant leur construction.

4.4.2.2. Les modèles

4.4.2.2.1. Le modèle fil de fer

Historiquement c'est le premier modèle. Les objets sont représentés sous la forme d'une suite de segments 3D définis par les coordonnées (x,y,z) de leurs extrémités.

Ce modèle simple ne demande pas une grande puissance de calcul et peut être implanté dans un espace mémoire restreint. Particulièrement adapté pour les applications en temps réel, il génère toutefois de fâcheuses ambiguïtés :

- il ne distingue pas le plein du vide (figure 4.7)
- il n'assure pas les critères que doivent vérifier les objets solides (cf.4.4.2.1.)
- il ne permet pas les calculs géométriques (volume, masse,...) et l'élimination des parties cachées.

Il est, essentiellement, utilisé pour la visualisation.

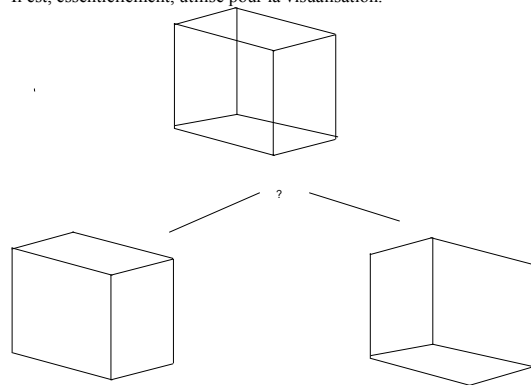


Figure 4.7 : Ambiguïté de la représentation fil de fer

4.4.2.2.2. Le modèle surfacique

Ce modèle permet de distinguer l'intérieur de l'extérieur donc le plein du vide. L'objet est dans ce cas représenté par un ensemble de facettes. Chaque facette est définie par son contour qui est une suite ordonnée de segments 3D. Le modèle surfacique peut générer des objets non cohérents avec des facettes pendantes. Il est néanmoins utilisé, pour des raisons historiques, par la plupart des algorithmes d'élimination des parties cachées.

4.4.2.2.3. Le modèle arbre de construction (CSG)

C'est la représentation la plus utilisée par les logiciels de CAO. L'objet est modélisé sous la forme d'un arbre binaire où les feuilles sont des volumes élémentaire (boite, sphère, cylindre, ...) et les nœuds sont des opérations booléennes régulières ( $\cup^*$ ,  $\cap^*$ ,  $-^*$ ) (figure 4.8).

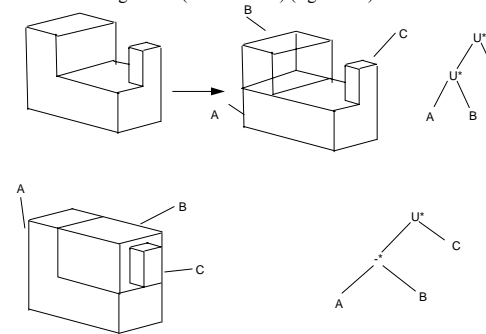


Figure 4.8 : la représentation CSG

Pour respecter les critères définis sur les objets solides, on impose la régularité des opérations booléennes. Ainsi :

$$A \cup^* B = r(A \cup B), A \cap^* B = r(A \cap B), A -^* B = r(A - B)$$

où r est l'application qui associe à un sous-ensemble de  $\mathfrak{R}^3$  l'adhérence de son intérieur.

En d'autres termes, les opérations booléennes régulières sont des lois de composition interne sur l'ensemble des objets solides (figure 4.8).

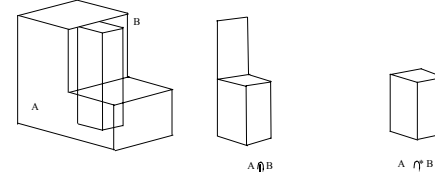
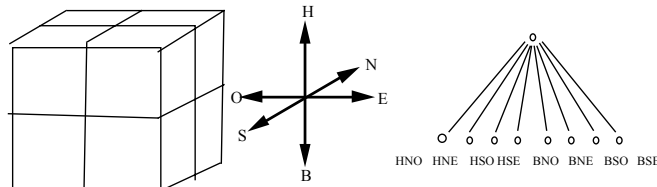


Figure 4.8 : opérations régulières

La modélisation par arbre de construction se réalise simplement; elle est généralement très conviviale. Si les opérations booléennes sont régulières, les objets générés sont cohérents. Les calculs géométriques sont par contre difficilement réalisable et l'unicité de la représentation n'est pas assurée (figure 4.7).

4.4.2.2.4. Le modèle octree

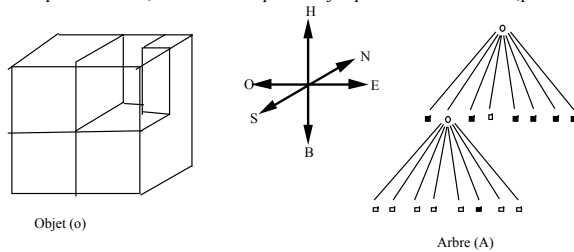
Un octree est un arbre à huit branches. Les nœuds de l'arbre sont des cubes inclus dans l'espace de modélisation. Le nœud racine est l'espace lui-même et ses huit fils sont des cubes de même volume qui le divise (figure 4.9).



Les noeuds sont ordonnés du Haut vers le Bas, du Nord vers le Sud et de l'Ouest vers l'Est

Figure 4.9 : Octree

Une feuille de l'arbre est dite blanche (B) resp. noire (N) si le cube qui lui correspond est à l'extérieur de l'objet (resp. à l'intérieur). Les nœuds internes ont la couleur grise (G) et possèdent huit fils. Modéliser un objet sous forme d'octree consiste donc à diviser récursivement l'espace en huit parties égales. Lorsque nécessaire, la division se poursuit jusqu'à une taille de cube préalablement définie.



Informtiquement, un octet n'est pas représenté sous la forme d'un arbre mais par une liste. La liste de l'arbre A est :

G (N G(B B B B N B B) N B N N N N)

Figure 4.10 : la modélisation par octree

La modélisation par octree génère des objets cohérents. Elle est unique et non- ambiguë. Elle permet d'effectuer très simplement tous les calculs géométriques ainsi que les opérations booléennes entre les objets. Certains algorithmes d'élimination de parties cachées peuvent lui être appliqués. Malheureusement, aucun modeleur ne génère directement une telle représentation.

4.4.2.5. Le modèle de représentation par frontières (BREP)

a) Définition :

Cette représentation décrit explicitement la frontière de l'objet. La structure de données qui lui est associée est de type listes de faces/arêtes/sommets . Plusieurs organisations des données sont possibles ; la Winged-Edge (3) ou Double Edge List (4) est la plus appropriée..

La structure Winged-Edge (figure 4.11) sous sa forme simplifiée définit un solide ou un polyèdre (P) comme étant un ensemble de faces. Une face (F) est une suite orientée d'arêtes. Une arête (A) est un couple de sommets. Un sommet (S) est un triplet (x,y,z) de coordonnées. Figure 4.11 : la structure Winged edge

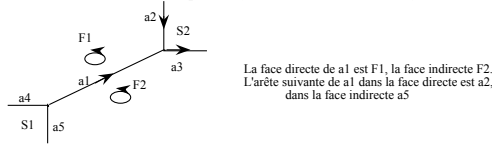


Figure 4.11 : la structure Winged edge

La structure Winged-Edge permet d'obtenir toutes les informations géométriques quelque soit l'entité considérée: coordonnées de sommets, équation des arêtes, équation du plan support des faces. Elle fournit toutes les relations topologiques soit: les contours (C), les incidences (I), les adjacences (A), des entités S, A, F, P (figure 4.12).

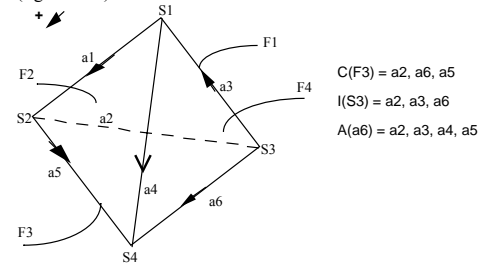


Figure 4.12 : les relations topologiques de la structure Winged edge

b) Structures de données et opérateurs

.Structure de données classique :

Ci-dessous un exemple de structure de données classique utilisée dans le cadre de la représentation par frontières :

```

struct t_polyedre {
    struct t_face *f;
    struct t_arete *a;
    struct t_sommet *s;
}

struct t_face {
    struct t_contour *c;
    struct t_face *suiv;
}

struct t_contour {
    struct t_arete *a;
    struct t_contour *suiv;
}

struct t_arete {
    struct t_sommet *o, *e;
    struct t_arete *suiv;
}

struct t_sommet {
    double x, y, z;
    struct t_sommet *suiv;
}
    
```

**Exo. :** - Ecrire les programmes en C correspondants aux opérateurs C(f) , I(s) , I(a) .

- Déterminer la complexité en temps d'exécution de ces programmes.
- Déterminer la complexité en temps d'exécution des autres opérateurs.

**. Structure de données pour Winged Edge**

```
struct t_polyedre {
    struct t_face *f;
}
struct t_face {
    struct t_arete *a;
    struct t_face *suiv ;
}
struct t_arete {
    struct t_face *fdir, *find;
    struct t_arete *asfd, *asfi;
    struct t_sommet *o, *e;
}
struct t_sommet {
    double x, y,z ;
    struct t_arete *a;
}
```

**Exo. :** - Ecrire les programmes en C correspondants aux opérateurs C(f) , I(s) , A(a) .

- Déterminer la complexité en temps d'exécution de ces programmes.
- Déterminer la complexité en temps d'exécution des autres opérateurs.

**c) La relation d'Euler :**

**S-A+F=2** (valable pour des polyèdres sans trous)

**S-A+F-H=2(C-G)**

où :

S est le nombre de sommets, A d'arêtes , F de faces , H d'ouvertures , C de composantes connexes et G de conduits.

Lorsqu'elle est vérifiée par une structure Winged-Edge assure la cohérence topologique des objets. La représentation par frontière est théoriquement unique et non ambiguë, dans la pratique, des problèmes de précision numérique mettent en défaut ces qualités. Les « Brep » sont bien adaptées au calcul géométrique et à l'élimination des parties cachées. Les opérations booléennes sont applicables. Certains modeleurs génèrent directement une représentation par frontière.

La figure qui suit résume les avantages et les inconvénients des différents modèles.

critères modèles	le modèle représente un objet cohérent	la représentation est unique et non ambiguë	le modèle possède un modeleur adapté	le modèle permet les calculs géométriques	l'élimination des parties cachées est réalisable
Fil de fer	non	non	tous le sont	non	non
Surfacique	non	non	tous le sont	non	oui peut être interactive
V O L U M I Q U E	CSG	oui	non	oui	non
	Octree	oui	oui	non	oui de façon interactive
	Brep	oui	oui	oui mais non-ergonomique	oui mais complexe

Figure 4.13 : Comparaison des méthodes de représentation des solides

**5. Appariement d'une image 2D avec son modèle 3D**

**5.1. Bref état de l'art**

Le problème de la mise en correspondance est l'un des problèmes les plus difficiles de la vision par ordinateur. Ce problème peut être traité en 2D/2D, ce qui est le cas de la vision stéréoscopique ou de l'analyse de séquence d'images, ou en 2D/3D basé sur l'utilisation de modèles 3D. Nous nous intéressons à cette dernière catégorie d'appariement. Nous présentons les principales approches de résolution de ce problème.

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

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

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

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

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

Lamdan et Wolfson [LAM DAN 1988] ont introduit une méthode géométrique. Celle-ci consiste en la création hors ligne d'une table de hachage et à une reconnaissance en ligne. La table de hachage contient les différents aspects des objets du modèle 3D exprimés dans des bases différentes (une base peut être représentée par 2, 3 points ou arêtes). Cette expression est basée sur des invariants ou des quasi invariants géométriques [BIN FORD 1993], [GROS 1995]. La reconnaissance en ligne est obtenu par mise en correspondance de points 2D, exprimés à leur tour dans différentes bases, avec le contenu de la table de hachage.

Jain and Hoffman [JAIN 1988] ont développé une méthode utilisant le raisonnement par évidence pour reconnaître un objet 3D dans une image de profondeur. Pour cela, ils utilisent un système à règles de production. La prémisse d'une règle est composée de conditions sur les différentes primitives de l'objet et la partie action attribue un degré de probabilité sur l'appartenance de l'objet à la base de données.

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

Pampagnin et Devy [PAMPAGNIN 1988] ont introduit le graphe de compatibilité qui regroupe dans ses nœuds toutes les hypothèses d'appariement entre une chaîne de segments 2D et une face d'un aspect du modèle 3D et dont les arcs représentent les compatibilités entre hypothèses d'appariement.

Nous avons développé deux méthodes d'appariement d'objets polyédriques basées sur la connaissance de leur modèle 3D et de leur image caméra 2D:

La 1<sup>ère</sup> est particulière car elle suppose une connaissance géométrique a priori. Elle utilise la technique de la recherche arborescente et le raisonnement par l'évidence. L'originalité de cette méthode est qu'elle prend en compte les contraintes physiques (liaisons) entre les objets.

La 2<sup>ème</sup> est plus générale. Elle utilise les méthodes du graphe d'appariement et celle du hachage géométrique. La méthode du graphe d'appariement permet d'apparier des faces du modèle 3D avec des chaînes de segments dans une image. La méthode du hachage géométrique permet de mettre en correspondance des sommets du modèle 3D avec des sommets de segments dans une image. L'originalité réside dans l'utilisation de l'une de ces méthodes, selon la qualité du traitement de l'image, afin de permettre une reconnaissance de l'objet même si le traitement de l'image ne fournit pas suffisamment de chaînes cohérentes de segments.

Nous décrivons dans les paragraphes suivants chacune de ces méthodes.

## 5.2. Appariement 2D/3D polyédrique basée sur liaisons

La 1<sup>ère</sup> est basée sur l'utilisation d'un modèle 3D tenant compte des liaisons entre objets polyédriques. Nous nous sommes restreints à la liaison appui plan, car c'est la plus couramment rencontrée, mettant en jeu 3 degrés de liberté entre 2 objets. Les liaisons permettent, si elles ne sont pas rompues dans la scène réelle, de restreindre l'appariement dans une zone privilégiée de l'espace.

La méthode consiste à reconnaître un objet polyédrique parmi plusieurs posés sur un support plan. L'équation de celui-ci est connue (fournie par la calibration de la caméra). Les segments extraits de l'image réelle sont projetés dans ce plan de pose. Les segments obtenus sont appariés avec les arêtes du modèle 3D en utilisant leur longueur et l'angle formé entre leur support et la perpendiculaire au plan image comme critères d'appariement. L'étape de vérification se trouve également simplifiée car elle met en œuvre une reconstruction à 3 degrés de liberté au lieu de 6.

## 5.3. Appariement 2D/3D polyédrique

Cette méthode résout le problème de l'appariement des objets polyédriques en s'appuyant sur trois techniques : l'organisation perceptive des indices visuels pour décrire un modèle 2D de l'image, le graphe d'aspects pour décrire la topologie du modèle 3D et un graphe d'appariement ou une table de hachage pour la construction des hypothèses d'appariement. Ensuite, la transformation rigide entre le modèle 3D et le repère de travail est calculée dans la phase de vérification durant laquelle sont appliquées les méthodes de reconstruction 3D décrites dans le chapitre 3. Les invariants géométriques, basés sur les rapports des surfaces du modèle 3D et de l'image, sont également appliqués pour éliminer les hypothèses invalides. Des tests partiels sur la précision du système de reconnaissance sont effectués.

La figure 5.1 représente la structure générale du système de reconnaissance automatique d'objets polyédriques :

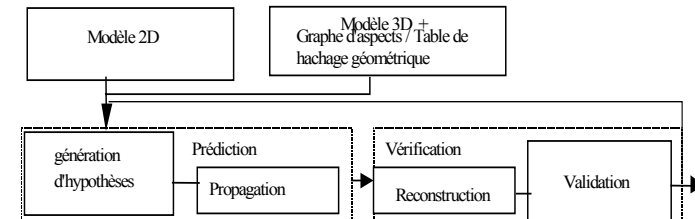


Figure 5.1: Structure générale du système de reconnaissance d'objet.

Nous décrivons brièvement les étapes principales du système de reconnaissance automatique, qui sont :

- le traitement de l'image, à l'issue duquel un modèle 2D de la scène est établi,
- la construction automatique des graphes d'aspects à partir du modèle 3D,
- l'appariement entre modèle 2D et le modèle 3D, qui permet la reconnaissance d'objet.

## 5.4. Traitement et Modélisation de l'image 2D

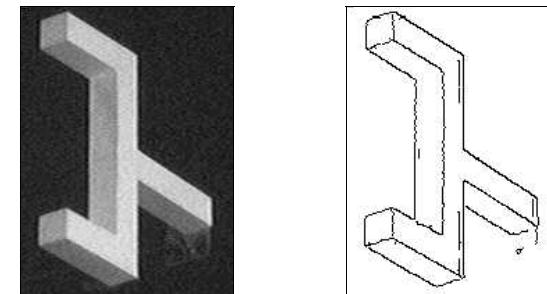
Le traitement d'image appliqué consiste à extraire des primitives 2D (sommets, segment, chaîne de segments) formant le modèle 2D.

Ce traitement est constitué du filtrage du gradient, de l'amincissement et du suivi de contour par hystérésis, de l'extraction des segments et des chaînes de segments. L'extraction de segments est réalisée à partir des points de contours en appliquant la Transformée de Hough (HT).

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

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

La Figure 5.2 montre les points de contour obtenus (image binaire), à partir de l'image d'entrée (a).



(a) image de l'objet Pince

(b) le résultat de l'extraction des points de contour

Figure 5.2 : Traitement de l'image de l'objet pince

La transformée de Hough (TH) est appliquée pour la segmentation de l'image. Ce choix vient du fait que la TH est robuste vis à vis du bruit. L'application de la TH classique, sans connaissance de l'angle de contour, est lente car il faut représenter chaque point du contour par une sinuséide complète dans la matrice de Hough (MH). Chaque point  $(x, y)$  du contour peut correspondre à la cellule  $(a_i, d_i)$  dans la MH ( $a_i$  : angle de contour par rapport à la verticale,  $d_i$  : distance du centre de l'image à la droite support du point  $i$ ), tel que :

$$d_i = x \cdot \cos \alpha_i - y \cdot \sin \alpha_i$$

Nous avons appliqué l'angle de contour, calculé à partir de l'angle du gradient, dans l'équation ci-dessus pour avoir une seule correspondance dans la MH. La segmentation est devenue nettement plus rapide, mais l'erreur sur l'angle de contour a entraîné des segments incomplets. Pour pallier ceci, nous avons développé deux méthodes :

La 1<sup>ère</sup> consiste à explorer les cellules voisines du segment à extraire. L'inconvénient de cette méthode est la difficulté de fixer un critère d'arrêt.

La 2<sup>ème</sup> méthode consiste à définir une tolérance sur l'angle du gradient et à explorer uniquement la cellule correspondante à celui-ci. Chaque point du contour est alors représenté par une portion de sinusé dans la MH.

La figure 5.3 illustre le fonctionnement de la 2<sup>ème</sup> méthode, dont l'algorithme itératif est constitué des étapes suivantes :

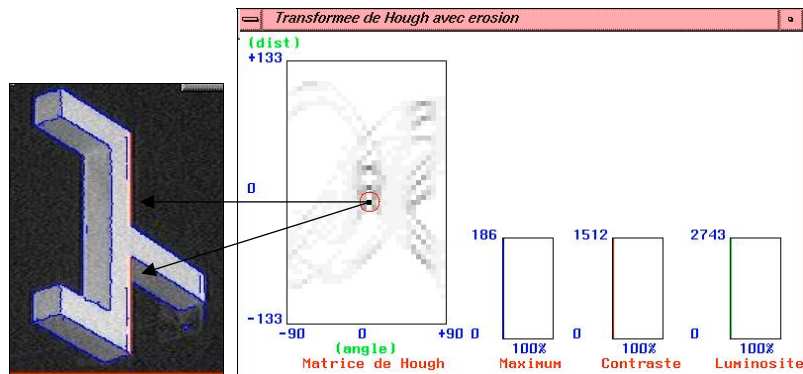
- 1- calcul du vote (nombre de points) pour toutes les cellules de la MH,
- 2- détermination de la cellule à vote maximal : elle représente le segment courant,
- 3- retrait de la MH de tous les points appartenant au segment courant,
- 4- test du critère d'arrêt, s'il n'est pas satisfait retourner à 1.

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

Le 1<sup>er</sup> critère, appelé maximum, représente le vote maximal dans une itération donnée.

Le 2<sup>nd</sup> critère, appelé contraste, la somme des différences avec la moyenne des votes des différentes cellules restantes à explorer.

Le 3<sup>ème</sup> critère, appelé luminosité, la somme des différences au carré avec la moyenne des votes des différentes cellules restantes à explorer.



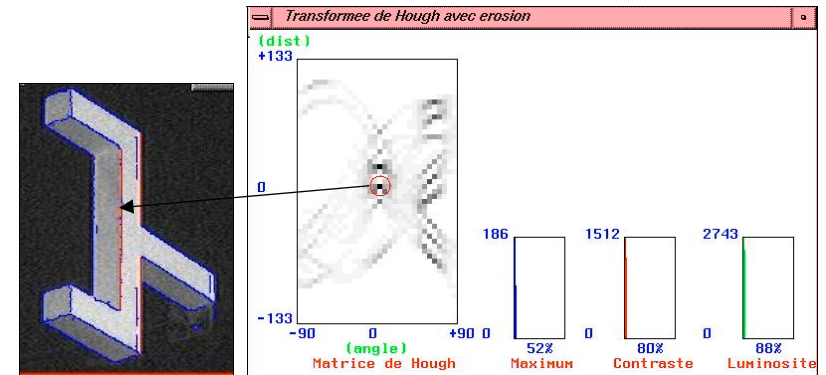
(a) première itération :

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

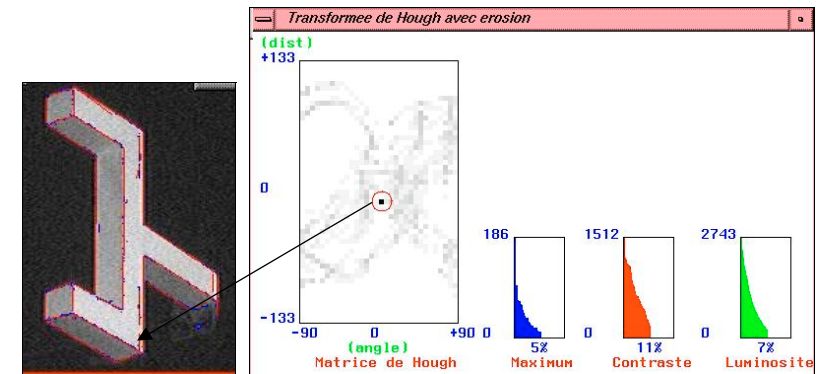
Notes :

- Un point est représenté par ses coordonnées polaires dans la matrice de Hough (l'angle de contour par rapport à la verticale et la distance de la droite support du point au centre de l'image).
- Dans les rectangles représentant les critères(maximum,contraste,luminosité), l'axe des abscisses représente le nombre de segments traités, l'axe des ordonnées représente le nombre de

points par segment(taux qui correspond à la longueur du segment en cours de traitement par rapport à celle du 1<sup>er</sup> segment traité).



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



(c) dernière itération : maximum, contraste et luminosité sont proche de 10%(limite fixée empiriquement pour l'arrêt du programme)

Figure 5.3 : La segmentation par la transformée de Hough améliorée

Pour la fusion de segments, nous utilisons les formules développées par Lowe [LOWE 1987] pour regrouper les segments de droite selon leurs proximité, parallélisme. A partir des informations de proximité entre les segments 2D, les points d'intersection (sommets 2D) sont calculés.

L'étape suivante consiste à parcourir la liste de segments qui se trouvent en proximité pour établir des chaînes qui contiendront certaines descriptions : chaîne ouverte ou fermée, nombre de segments,... Les sommets, segments et chaînes de segments sont organisés dans une structure de données appelée *modèle 2D*.

L'opérateur peut intervenir pour changer ou tester ce modèle (figure 5.4).

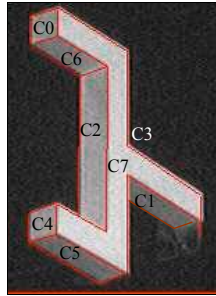


Figure 5.4 : Le modèle 2D de l'image de l'objet Pince

**5.5. Construction des graphes d'aspects pour polyèdre**

Le graphe d'aspects donne une description topologique du modèle 3D où chaque aspect représente des éléments géométriques (sommet, arête, face), du modèle 3D, visibles simultanément par un observateur (caméra).

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

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

Un graphe d'aspects est la représentation sous forme de graphe de la parcellisation de l'espace en cellules. Il a pour nœuds les aspects et pour arcs les événements visuels.

Deux méthodes de calcul du graphe d'aspects ont été développées. La 1<sup>ère</sup> est basée sur l'utilisation de la sphère de Gauss et la 2<sup>ème</sup> sur la parcellisation de l'espace par des plans pour la constitution des différentes cellules.

Nous décrivons brièvement chacune de ces méthodes.

**5.6.1. Méthode exhaustive**

Cette méthode a été développée pour servir à l'appariement basée sur le hachage géométrique, utilisant les sommets du modèle 3D. Un aspect est alors représenté par un ensemble de sommets visibles simultanément.

On considère une sphère de Gauss centrée sur l'objet que l'on discrétise selon la longitude et la latitude (figure 5.5) pour obtenir une série de points repartis sur la surface de cette sphère. Un point P représente ainsi un point de vue adopté pour projeter le modèle.

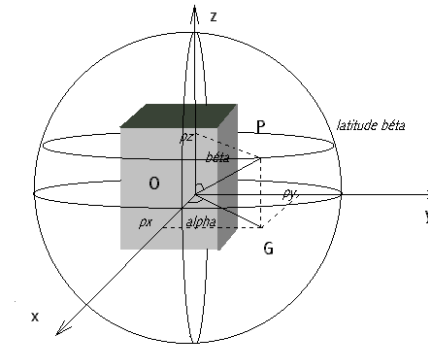
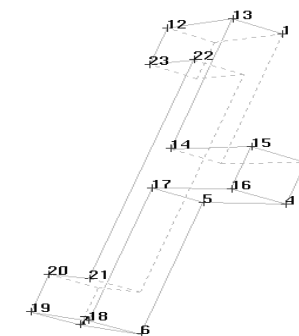


Figure 5.5 : sphère englobant un modèle de type 'boite'

Pour obtenir un aspect concret du modèle les points cachés sont éliminés lors de la projection (Figure 5.6).



La Figure 5.6 : un aspect de l'objet pince  
( tous les segments contenant au moins un point caché sont en pointillé)

Un ensemble d'aspects est ainsi défini, dans lequel il suffit alors d'éliminer la redondance. Deux aspects sont reliés dans le graphe, si et seulement si, les points correspondants sur la sphère sont voisins.

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

**5.6.2. Méthode analytique**

Cette méthode a été développée pour servir à l'appariement basé sur un graphe. Elle fournit le graphe complet d'un objet polyédrique. Le problème a été abordé en deux étapes de complexité croissante ; il s'agissait tout d'abord de traiter les polyèdres convexes, puis de s'étendre au cas général en incluant les polyèdres concaves. Afin d'obtenir l'ensemble des aspects d'un polyèdre, un graphe d'incidence est créé.

Le graphe d'incidence (GI) est la représentation d'une parcellisation de l'espace. Pour le construire, la liste des plans support des faces de l'objet est calculée. L'intersection entre trois plans, supports de faces de l'objet, constitue le noyau du graphe d'incidence. A chaque itération, un nouveau plan de la liste subdivisera les plans antérieurs. En exploitant toute la liste des plans, le graphe d'incidence est entièrement construit. Chaque cellule du graphe représente une parcelle de l'espace, contenant tous les points de vue à partir desquels on voit toujours le même aspect du polyèdre. Un codage est attribué à la cellule. Celui-ci est représenté par n bits où n est le nombre de faces du polyèdre. L'état d'un bit définit la visibilité d'une face. Une face n'est considérée visible, d'un point de vue donné, que si ce celui-ci est situé du bon côté du plan support de la face, c'est à dire du côté où se trouve la normale à la face (la normale à une face est orientée vers l'extérieur de la matière).

Les feuilles du GI représentent les différents aspects, les nœuds constituent les codages intermédiaires et les arcs les événements visuels représentant le passage d'un aspect à un autre.

La parcellisation de l'espace est complétée pour le cas concave en ajoutant celle obtenue par des plans auxiliaires. Ceux-ci sont issus des événements visuels de type segment/point.

Plantiga & Dyer [PLATINGA 1990] classifient les événements visuels que l'on va rencontrer d'une manière très accessible : événements point/segment ou segment/segment/segment. Sur la figure 5.7 nous constatons comment un simple plan (ici celui en diagonale) peut séparer ce qui est une cellule simple dans un

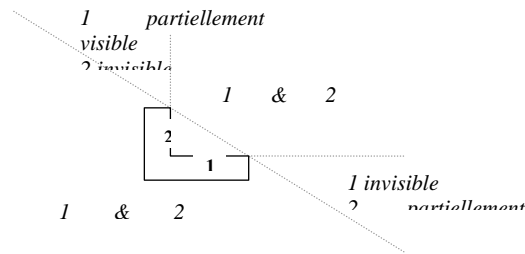


figure 5.7 : exemple d'un plan limite d'un événement visuel

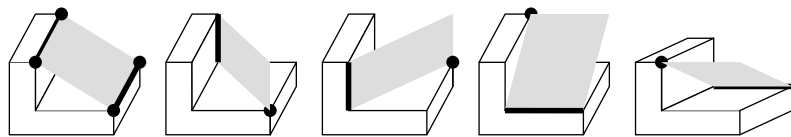


figure 5.8 : les plans intervenant dans les événements segment/point d'un polyèdre concave simple

graphe d'incidence en deux parties : l'une où une face est entièrement visible, l'autre où elle est partiellement cachée.

On parle d'événement visuel segment/point lorsque l'on passe d'un côté à l'autre d'un tel plan. Sur la figure 5.8 les points et les segments impliqués sont en gras.

Cet algorithme est issu des travaux de [STEWMAN 1988] qui utilise celui de EDELSBRUNNER 1986] pour la construction du graphe d'incidence. L'idée générale provient de leur algorithme mais de nombreuses modifications y ont été apportées afin de limiter le nombre de plans auxiliaires pris en compte.

La construction du graphe d'aspects pour un polyèdre convexe fonctionne parfaitement et elle est relativement rapide, les modèles de tests convexes sont traités en quelques secondes. Les modèles concaves pour les tests sont traités en quelques secondes pour les modèles simples (figure 5.11) et en plusieurs minutes pour les modèles complexes.

Pour les polyèdres concaves il faudrait incorporer le traitement des événements du type segment/segment/segment afin de couvrir l'ensemble des aspects..



Figure 5.9 : quelques aspects obtenus de l'objet Pince

Le graphe d'aspects sert à vérifier si une hypothèse de la visibilité de ses éléments. La construction du graphe d'aspects de chaque modèle 3D est réalisée hors ligne. L'utilisation du graphe d'aspects, en ligne, permet de simplifier et d'accélérer la reconnaissance d'un polyèdre dans une image.

**5.6. Méthodes basées sur le graphe d'appariement et sur le hachage géométrique**

L'appariement est constitué d'une étape de prédiction et d'une autre de vérification.

La prédiction confronte un aspect (vue) de l'objet, extrait d'un graphe de l'ensemble des aspects de l'objet, à l'image de celui-ci.

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

**5.6.1. Méthode du graphe d'appariement**

La méthode du *graphe d'appariement* (Gapp) permet d'incorporer les contraintes topologiques des objets, grâce au graphe d'aspects. Cela mène à une réduction importante du nombre d'hypothèses possibles sur l'appariement entre le modèle 2D et le modèle 3D. Les nœuds du GApp représentent les hypothèses d'appariement entre une chaîne de segments 2D et une face d'un aspect du modèle 3D. Les nœuds sont directement issues de la topologie des faces 3D et des chaînes de segments 2D. Un arc relie deux nœuds si les 3 critères suivants sont satisfaits:

- Critère de visibilité : les deux faces 3D apparaissent dans l'un des aspects du modèle 3D (la visibilité des deux faces est possible).
- Critère de connexité : les deux chaînes ont la même relation de connexité que les deux faces (sommet en commun, segment en commun ou rien).

- Critère d'invariance : le test des invariants géométriques, basés sur des rapports de surfaces entre faces et chaînes fermées de segments, est satisfait[ZHU 1995].

Nous cherchons, ensuite, des sous graphes complets du Gapp, appelés *cliques maximales*. Une clique maximale doit, aussi, vérifier les critères de visibilité et de connexité :

- toutes les faces 3D de la clique doivent apparaître dans un aspect,
- tous les appariements, entre les segments 2D et les arêtes 3D de la clique, doivent être cohérents,
- l'erreur issue de l'application des invariants géométriques ne dépasse pas un seuil fixé.

Les cliques maximales forment les hypothèses plausibles sur l'appariement. Elles sont munies d'un score, basé sur le nombre d'appariement sommet3D/point2D obtenu.

Le Gapp est appliqué sur le modèle 2D et chacun des modèles 3D sélectionnés. Les hypothèses issues de tous les Gapps sont ordonnées selon leur score. Une méthode de prédiction/vérification classique est appliquée, aux hypothèses une par une. Si une hypothèse réussit, ses éléments sont retirés du modèle 2D de l'image. Nous continuons cette opération jusqu'à l'exploration de tout le modèle 2D ou de toutes les hypothèses. La transformation rigide, entre le modèle 3D et le repère de l'affichage, est calculée dans la phase de vérification, pour chaque hypothèse réussie. Enfin, tous les modèles reconnus sont superposés sur l'image.

Cette méthode est tributaire de la qualité de l'image caméra. Si celle-ci est dégradée, le résultat du traitement de l'image s'avère insuffisant - difficulté d'extraction des chaînes de segments - pour appliquer la méthode du graphe d'appariement. D'où l'utilisation de la méthode du hachage géométrique basée sur les sommets des objets.

### 5.6.2. Méthode du hachage géométrique

De la même manière que la précédente méthode, un ensemble d'aspects d'un objet est calculé hors ligne et l'appariement est effectué en ligne. Les aspects sont calculés par la méthode exhaustive.

Dans notre implémentation, un aspect est représenté uniquement par les sommets visibles de l'objet du point de vue considéré. Une base de transformation dans le plan de l'aspect est définie, elle peut être formée de deux ou de trois points de l'aspect. Un nombre défini de ces bases, dans lesquelles nous projetons les aspects, est utilisé. Après avoir projeté les points dans chaque base, nous les stockons dans une structure appelée *table de hachage*. Celle-ci est un tableau à deux dimensions, de taille fixe, dont chaque entrée est une coordonnée (les deux dimensions correspondent donc à l'abscisse et à l'ordonnée d'un point). Les dimensions de la table de hachage étant fixes, nous sommes obligés d'approximer les coordonnées des points projetés. Un élément de la table de hachage est donc un point exprimé dans un repère absolu normalisé. Il sera modélisé par une liste dont chaque élément est un triplet : le modèle auquel il appartient, un aspect de celui-ci et une base dans laquelle l'aspect est exprimé.

L'appariement en ligne est effectué en confrontant une série de points, extraite de la scène réelle, avec le contenu de la table de hachage. Ces points, qui ont une valeur expérimentale, sont pour la plupart des sommets de l'objet à reconnaître. Ils sont ensuite exprimés dans différentes bases, ces bases étant construites à l'aide des points reconnus de l'image. En insérant ces points dans la table de hachage, nous obtenons une série de correspondances (triplets : modèle/aspect/base) avec un ou plusieurs aspects préalablement enregistrés lors de la phase hors ligne de préparation de la table de hachage. Pour chaque correspondance obtenue, nous ajoutons un vote pour un triplet modèle/aspect/base, dans une structure réservée à cet effet. Nous définissons ensuite un nombre de votes minimal à obtenir, afin qu'un triplet modèle/aspect/base soit considéré comme une hypothèse plausible. C'est ici que les propriétés d'invariances interviennent. En effet, quelque soit la position ou la taille de l'objet, il sera reconnu. Car, un ou plusieurs aspects, qui lui sont topologiquement semblables, ont été préalablement introduits dans la table de hachage. Il suffira alors d'appliquer une série de transformations affines sur un aspect reconnu, afin de réaliser l'appariement avec l'objet représenté dans la scène. Les hypothèses données par la table de hachage, sont toutes testées par une procédure de prédiction/vérification.

Des tests expérimentaux ont montré que les bases à deux points sont plus performantes que les bases à trois. La figure 5.10b montre que les bases à deux points rendent la table de hachage mieux équilibrée. Le niveau de gris d'une cellule représente la taille de sa liste. Le noir représente une liste vide, tandis que le blanc est une liste à taille maximale.

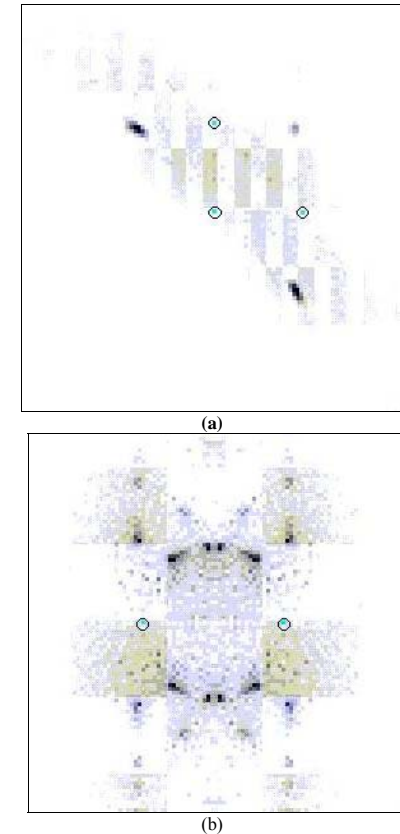


Figure 5.10: La table de hachage (a) a été réalisée à l'aide de bases à trois points, la (b) à l'aide de bases à deux points.

Grâce à la reconnaissance 2D/3D présentée ci-dessus, le suivi d'objet dans l'image caméra est rendu possible.

### 5.6.3 Evaluation de la méthode du graphe d'appariement

Afin de valider cette méthode de reconnaissance d'objet polyédrique, nous utilisons l'objet pince (figure 5.2), monté sur le robot à 4ddl, auquel le protocole expérimental suivant a été appliqué :

- Calibration du robot à 4 ddl. (on note ses coordonnées par rapport à la mire, ainsi que les coordonnées des moteurs du banc correspondantes, et ce, dans 3 positions différentes. On détermine alors les 6 paramètres de la tige.
- Calibration automatique de la caméra.
- Calibration de la pince montée sur le robot à 4 ddl selon la même méthode que celle appliquée au robot.

- Saisie d'une série d'images de la pince avec la caméra calibrée, en notant pour chacune d'entre elles les positions des moteurs du robot.

- Reconnaissance automatique de l'objet, et enregistrement des coordonnées 3D trouvées.

- Evaluation de l'erreur global du système.

Un ensemble de modèles 3D fut sélectionné de la base pour effectuer la reconnaissance (Figure 5.11). La méthode de graphes a été appliquée pour l'appariement. Elle donne un nombre total de 423 hypothèses. C'est l'hypothèse 2 sur le modèle de l'objet Pince, ayant un score 80 qui a réussi.

Nous donnons une description de cette hypothèse :

Chaîne /Face : C7F1 C5F2 C0F9 C6F10 C4F13

Segment /Arête: S3A0 S12A7 S11A8 S7A11 S23A12 S22A13 S0A14 S16A15 S15A16 S20A17 S21A18 S17A19 S13A20 S1A21 S10A22 S8A23 S4A24 S24A25 S18A31 S14A32 S2A33 S9A35

Point / Sommet: P13S0 P12S1 P9S7 P6S8 P14S9 P8S11 P11S12 P0S13 P18S14 P4S16 P17S17 P1S18 P3S19 P5S20 P16S21 P15S22 P7S23

Le modèle reconnu, vérifié par projection, est superposé sur son image Figure 5.12.

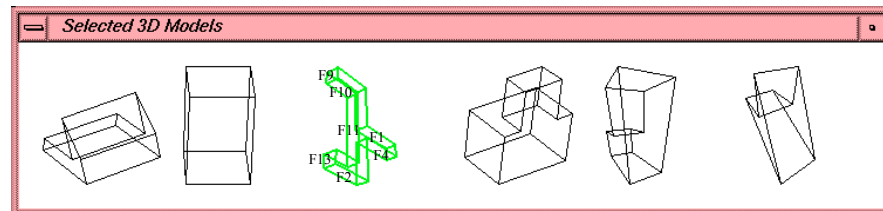


Figure 5.11 : les modèles 3D sélectionnés de la base, le modèle en double épaisseur a été reconnu et ses faces visibles sont marquées

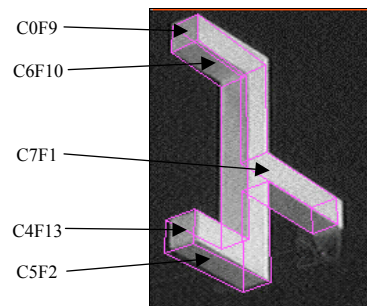


Figure 5.12 : le modèle de l'objet Pince est superposé sur son image avec erreur moyenne de 2.6 pixels, les hypothèses chaîne/face sont marquées

Les erreurs de reconstruction dans le plan image et dans le repère objet ont été calculées :

- l'erreur moyenne est de 2,6 pixels dans le plan image,
- les erreurs moyennes et écarts type dans le repère objet sont présentées dans le tableau ci-dessous :

	X	Y	Z
Moyenne de l'erreur(mm)	4,03	-2,75	-7,90
Ecart-type(mm)	1,67	1,04	6,50

Il est intéressant d'examiner les erreurs relevées par sommet de l'objet. Par exemple, le graphique suivant (figure 5.13) reprend la moyenne de l'erreur, exprimées en mètre, en X pour chacun des sommets :

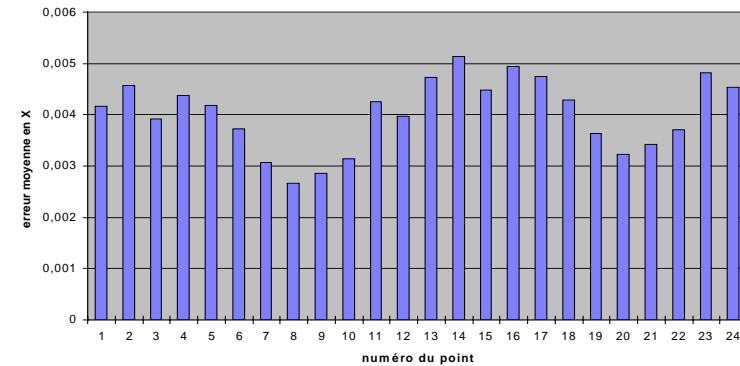


Figure 5.13 Erreur moyenne en X pour chaque sommet

On se rend compte que l'erreur pour certains sommets (8-9 et 20-21) est presque 2 fois moins élevée en moyenne que pour d'autres (2-4, 13-17 et 23-24). En fait la première série de points faisait partie des chaînes utilisées lors de l'appariement, et a donc été prise en compte dans le processus de minimisation de l'erreur d'appariement. Ce n'est pas le cas des sommets de la deuxième série.

Il paraît judicieux de sélectionner les sommets visibles apportant la meilleure précision dans le processus d'optimisation.

### 5.7 Méthode basée sur la photoclinoétrie[Ababsa 2000]

Dans la mise en œuvre de l'approche photoclinoétrie du recalage automatique nous avons considérées 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 a priori.
- 2) la prise de vue de la scène est effectuée par une seule caméra CCD, calibrée au préalable et colinéaire avec la source de lumière (cf. figure 5.7.1).
- 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 une luminosité constante de tous les points de vue qui ne dépend que de l'angle entre la direction de la source de lumière et 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 défini par sa représentation en polygones maillés.

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

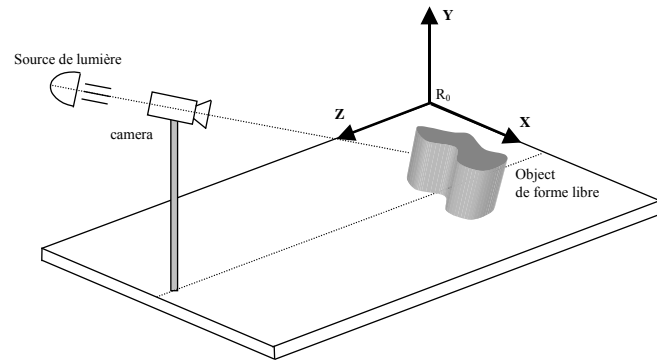


Figure 5.7.1 : Système photométrique

**Modèle de la réflexion diffuse**

Comme nous l'avons mentionné ci-dessus, la luminosité 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 5.7.2).

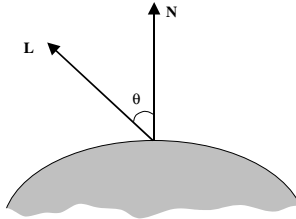


Figure 5.7.2 : Réflexion diffuse

La figure (5.7.3) 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.

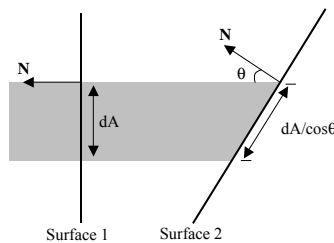


Figure 5.7.3 : Aire interceptée par les faisceaux élémentaires de section  $dA$

D'un autre côté, les surfaces Lambertiennes partagent la propriété connue sous le nom de loi de Lambert, selon laquelle la quantité de lumière réfléchiée 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 réfléchiée 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 :

$$I = I_p K_d \cos(\theta) \tag{5.7.1}$$

$I_p$  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.

En supposant que les vecteurs  $\vec{N}$  et  $\vec{L}$  sont normalisés, on peut réécrire l'équation (5.7.1) en utilisant le produit scalaire :

$$I = I_p K_d (\vec{N} \cdot \vec{L}) \tag{5.7.2}$$

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.

**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 (5.7.4).

Ainsi, le modèle géométrique 3D initial de la Vénus est donné par (cf. figure 5.7.4):

- 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.

```
# vertex
v -1.0800000 0.1300000 5.8000000
v -1.0500000 -0.1300000 5.1100000
v -1.2000000 0.3700000 5.3300000
v -0.9500000 -0.5300000 5.6100000
v -0.6000000 -0.4900000 4.9000000

# normals
vn 0.876291 0.132770 -0.463127
vn 0.837341 0.543765 0.056382
vn 0.919523 0.123972 -0.372972
vn 0.519292 0.677319 -0.521129
vn 0.224243 0.861935 0.454734

# faces
f 1 2 3
f 4 3 2
f 2 5 4
f 4 5 6
f 6 5 7
```

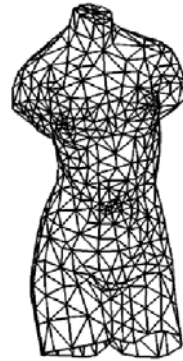


Figure 5.7.4 : Modèle géométrique 3D de la Vénus

De plus, pour des besoins de la méthode, nous avons enrichi ce modèle en lui rajoutant les normales aux surfaces. Chaque normale à la surface est déterminée par le produit vectoriel de deux vecteurs directeurs de deux de ses trois arêtes, pris dans le sens trigonométriques (cf. figure 5.7.5):

$$\vec{V}_f = \vec{V}_{s_1s_2} \times \vec{V}_{s_1s_3} \quad (5.7.3)$$

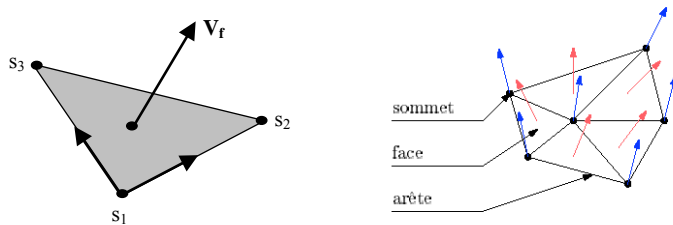


Figure 5.7.5 : Définition des normales aux surfaces du maillage

**Principe de la méthode**

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 (5.7.1). Pour exploiter à bon escient cette propriété, nous avons choisi de déterminer les zones de l'image les plus brillantes que nous avons appelé tâches et qui vont constituer nos primitives images. Ces tâches correspondent à des portions de surfaces de l'objet orientées dans la direction de la source de lumière, or comme cette dernière est connue a priori dans le repère de la scène (hypothèse 1), les normales 3D des tâches exprimées dans le repère monde sont déterminées. Pour chaque tâche nous calculons les angles entre la normale extraite et les normales aux points qui sont tout autour de la tâche. 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 tâche 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. Chaque aspect dans cette table sera défini par tout 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

détecter les différents aspects de l'objet et déterminer une représentation simple et pertinente. L'utilisation des méthodes classiques des graphes d'aspects nécessite de répertorier les 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 des 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, 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 primitives image (juste le nombre qu'il faut pour déterminer la transformation rigide).

La figure (7.5.6) reprend les deux grandes parties du système en ligne et hors ligne qui permettent de former 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.

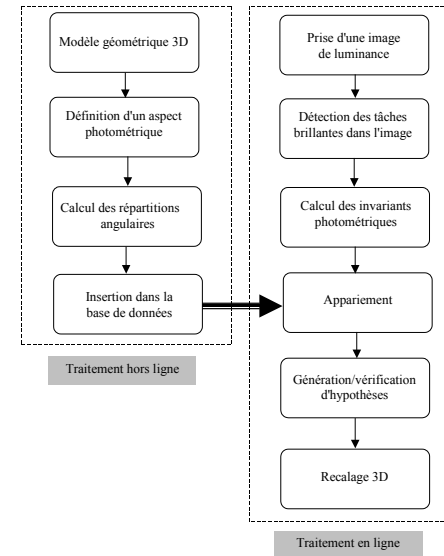


Figure 5.7.6 : Synoptique global de l'approche photométrique

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.

### 5.8 Méthode basée sur la clinométrie [Ababsa 2000]

La méthode clinométrique est une approche de recalage qui exploite, comme la méthode photoclométrique, 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 carrément 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. A partir de ces points, nous construisons des patch 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 5.8.1), de plus elles sont discriminantes pour les objets de forme libre, et de ce fait, caractérisent, à chaque fois, une région locale de l'objet, ce qui permettra de l'identifier dans la base de données.

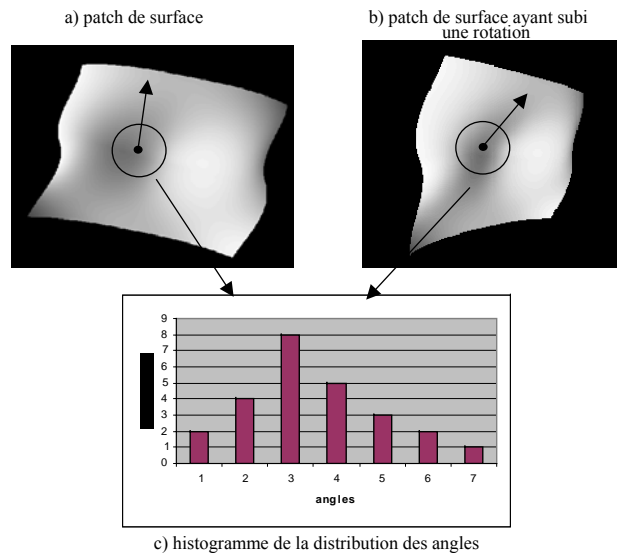


Figure 5.8.1: Principe de l'invariance de la distribution angulaire associée à un patch de surface

Ainsi, 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 patch<sup>3</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 va contenir la liste de tous les patch 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 patch 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 écarter de la mise en correspondance, ce qui réduit d'une manière significative la complexité du processus d'appariement.

<sup>3</sup> Les patch de surface sont construits à partir du nuage de points 3D obtenu lors de la phase de numérisation de l'objet de forme libre.

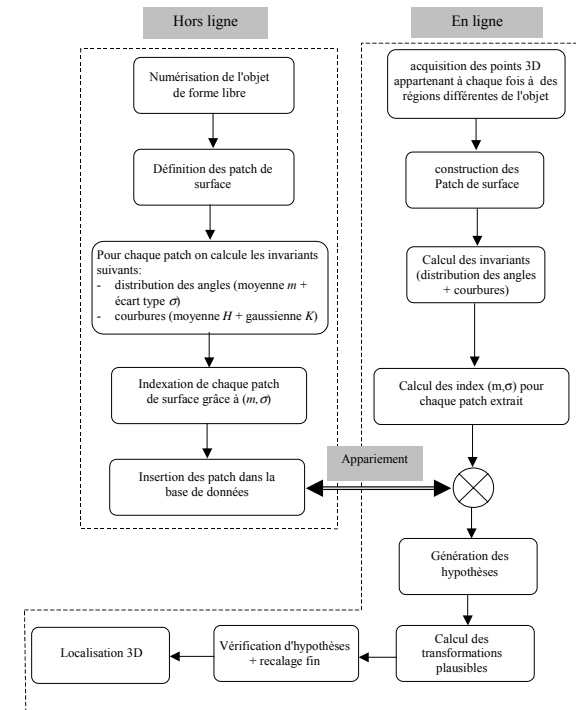


Figure 5.8.2 : Architecture du système de recalage clinométrique

Enfin, comme pour l'approche photoclométrique, nous avons choisi le paradigme de génération/vérification d'hypothèses pour appairier l'ensemble des patch de surface extraits de l'objets dans sa position courante avec ceux de la base de données [Huttenlocher et Ullman, 1987]. La vérification d'une hypothèse étant effectuée par projection dans l'image de luminance des patch du modèle transformés par la transformation rigide calculée à chaque fois.

La figure (5.8.2) 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., 2002].

### 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 laser et d'une caméra CCD. Le projecteur projète 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>4</sup> de la grille déformée projetée sur l'objet, à partir de l'image de cette grille acquise par la caméra (cf. figure 5.8.3). 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,

<sup>4</sup> Un nœud est défini comme étant le point d'intersection entre une ligne horizontale et une ligne verticale de la grille laser.

projecteur}, et de la qualité de l'appariement des nœuds homologues qui est fortement liée à la détection de l'image de la grille projetée.

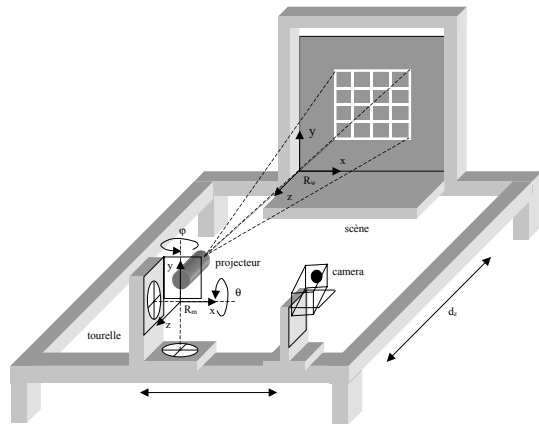


Figure 5.8.3 : Capteur stéréo actif 3D

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 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 ( $oy$ ). De cette façon, nous pouvons balayer et acquérir n'importe quelle région de l'objet avec la résolution que l'on veut. 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 ( $d_t$ ) au capteur 3D (projecteur+caméra) selon l'axe ( $oz$ ) du référentiel de la scène.

**Calibration du système de vision**

La méthode de calibration développée est basée sur l'utilisation du modèle sténopé pour la caméra et le projecteur [Faugeras, 1993]. En effet, d'un point de vue mathématique un projecteur fonctionne exactement comme une caméra et peut donc 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).

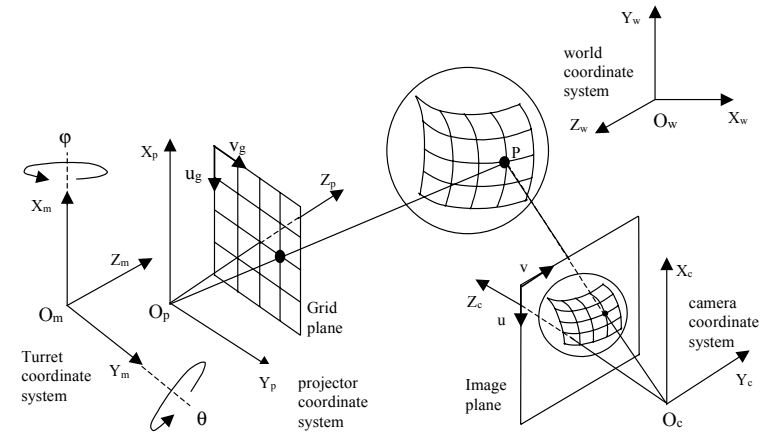


Figure 5.8.4 : Systèmes de coordonnées

**Calibration de la caméra**

Soit  $P(x,y,z)$  un point 3D représenté dans le repère monde (cf. figure 5.8.4), la relation entre le point  $P$  et son correspondant  $p(u,v)$  dans le plan image est donnée par :

$$\begin{bmatrix} s.u \\ s.v \\ s \end{bmatrix} = M_c \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{4.1}$$

où  $s$  est un facteur d'échelle, et  $M_c$  la matrice ( $3 \times 4$ ) de projection perspective.

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 quintuplés  $\{(X_i, Y_i, Z_i, u_i, v_i)\}_{i=1, \dots, n}$  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 au nombre d'inconnues (voir Annexe). 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  $\phi$  et  $\theta$ . Ensuite, nous mesurons les coordonnées 3D dans le repère monde des nœuds de la grille projetée. Les angles  $\phi$  et  $\theta$  sont choisis de telle manière à ce que les points 3D générés soient éparpillés dans l'espace de travail afin d'augmenter la précision du recalage 3D. Ce processus nous permet, d'extraire un certain nombre de points 3D coplanaires. Il doit être effectué au moins à deux distances différentes (nous avons choisi  $d_2=300\text{ mm}$  et  $d_3=600\text{ mm}$ ) car la calibration impose d'avoir des points non 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. D'un autre côté, pour estimer les coordonnées image, Nous avons élaboré un algorithme 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 subpixelique, 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 quintuplés nécessaire pour déterminer  $M_c$ .

### Calibration du projecteur

Comme nous l'avons déjà mentionné, notre projecteur peut tourner de deux angles,  $\varphi$  et  $\theta$ , 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), 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 initial, 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 :

$$\begin{bmatrix} t.u_g \\ t.v_g \\ t \end{bmatrix} = M_p \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (5.8.2)$$

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 quintuplés  $\{(X_i, Y_i, Z_i, u_{gi}, v_{gi})\}_{i=1, \dots, n}$ .

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 (5.8.2).

D'un autre côté, nous remarquons que lorsque le projecteur tourne 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  $A_p(3 \times 4)$  respectivement (voir annexe), tel que :

$$M_p = I_p \times A_p \quad (5.8.3)$$

et

$$A_p = \begin{pmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0_{1 \times 3} & 1 \end{pmatrix} \quad (5.8.4)$$

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

Lorsque la tourelle effectue deux rotations en même temps, 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 :

$$R' = R_\varphi \times R_\theta \times R \quad (5.8.5)$$

De même, le vecteur de translation devient :

$$T' = C_{m/w} \times (R_\varphi \times R_\theta \times O_{p/m}) \quad (5.8.6)$$

où  $C_{m/w}$  exprime la transformation rigide du repère de la tourelle vers le repère monde. L'équation (4.6) veut dire 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 :

$$A_p' = \begin{pmatrix} R'_{3 \times 3} & T'_{3 \times 1} \\ 0_{1 \times 3} & 1 \end{pmatrix} \quad (5.8.7)$$

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

$$M_p' = I_p \times (A_p')^{-1} \quad (5.8.8)$$

L'équation (5.8.8) est très intéressante car elle nous permet de recalibrer le projecteur laser 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.

### 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 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 recaler.

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

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

### 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. Le modèle proposé permet de modéliser des croix de forme quelconque (des croix obliques et/ou droites, voir figure 5.8.5), il est défini par l'équation suivante :

$$L(x,y) = a + b \left( e^{c(x-d-y-e)^2} + e^{-f(y-g-x-h)^2} - e^{-c(x-d-y-h)^2 - f(y-g-x-h)^2} \right) \quad (5.8.9)$$

Il est clair que le modèle de l'équation (5.8.9) 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.

## BIBLIOGRAPHIE

## Bibliographie sur la modélisation 3D

- [AYA 91] AYACHE N., LUSTMAN F., *Trinocular Stereo Vision for Robotics*, IEEE Trans. on PAMI,13(1), pp.73-85, January 1991.
- [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.
- [BAD 94] BADCOCK J.M., JARVIS R.A., *Wire-frame modeling of polyhedral objects from rangefinder data*, Robotica, Vol 12, pp. 65-75, 1994.
- [COL 94] COLLE E. , CHEKHAR Y., *Processus de construction d'images 3D*, RAPA, Ed. HERMES, Vol. 7(2), 1994.
- [DHO 89] DHOME M., RICHETIN M., LAPRESTE J.T., RIVES G., *Determination of the Attitude of 3D Objects from a Single Perspective View*, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.11, n°12 pp 1265-1278,December 198
- [DHO 90] DHOME M., LAPRESTE J.T., RIVES G.,and RICHETIN M., *Spatial Localisation of Modelled Objects of Revolution in Monocular Perspective Vision*, European Conference on Computer Vision, volume 24, 1990, pp. 475-488.
- [FAU 93] FAUGERAS O. G., *Three Dimensionnal Computer Vision: A Geometric Viewpoint*, MIT Press, Boston 1993.
- [FOL 90]FOLEY D.F., VAN DAM A. , FEINER S.K. , HUGHES J.F. *Computer Graphics: Principle and Practice*,2<sup>nd</sup> Edition, Addison-Wesley Publishing Compagny, 1990.
- [GAN 84] GANAPATHY S., *Decomposition of transformation matrices for robot vision*, Proceedings Int. Conf. on Robotics and Automation; Atlanta; pp 130-139; March 13-15,1984.
- [Haralick et Shapiro, 1993] R. M. Haralick et L. G. Shapiro (1993). Computer and robot vision. Addison-Wesley, Vol. 2.
- [HOR 87] HORAUD R., *New Methods for Matching 3D Objects with Single Perspective Views*, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol PAMI-9, n°3,pp 401-412, May 1987
- [HOR 93] HORAUD R., MONGA O., *Vision par ordinateur, outils fondamentaux*, Hermes, Paris, France, 1993
- [KUM 95] KUMAR S. , HAN S., GOLDFOG D. and BOWYER K., *On Recovering Hyperquadrics from Range Data*, IEEE Transaction on PAMI, Vol. 17(11), pp. 1079-1083, 1995.
- [LOW 87] LOWE D., *Three dimensionnal Object Recognition from Single Two-dimensional Images*, Artificial Intelligence, 31, pp 355-395, 1987.
- [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.
- [Lu et Hager, 2000] Lu C., Hager G. D. et Mjolsness E., Globally Convergent Pose Estimation from Video, Transactions on Pattern Analysis and Machine Intelligence, 22, 610-622, 2000.
- [NZI 1995] N'ZI E.C., *Modélisation et Reconstruction 3D d'Environnement : Application à la Téléopération*, Thèse de doctorat de l'université d'Evry Val d'Essonne, Evry, 7/12/1995.
- [NZI 97] N'ZI E.C., MALLEM M., CHAVAND F., *Interactive building and updating of a 3D database for teleoperation*, Robotica, International Journal of Information, Education and Research in Robotics and Artificial Intelligence, Edited by J. Rose, Cambridge University Press, England vol. 15, part 5, pp. 493-510, oct.1997.
- [NEV 95] NEVEU M., FAUDOT D., DERDOURI B., *Superquadriques B-déformables pour la reconstruction 3D*, Technique et Sciences Informatiques, Vol. 14(10), pp. 1291-1314, 1995.
- [PER 90] B. PEROUCHE, D. GHAZANFARPOUR, J. ARJENCE, D.MICHELUCCI "La synthèse d'images" HERMES 1990
- [PRE 85]PREPARATA F.P. and SHAMOS M.I., *Computational Geometry: an Introduction*, Springer Verlag 1985.
- [PRESS 1992] PRESS W. H., TEUKOLSKY S. A., VETTERLING W. T. and FLANNERY B. P. (1992): *Numerical Recipes in C: the art of scientific computing*; Cambridge University Press 1992.
- [REQ 80]REQUICHA A.A.G., *Representations for Rigid Solids: Theory, Methods and Systems*, ACM Computing Surveys, vol.12, N°4, pp. 437-464,Dec. 1980.
- [SAY 96] SAYD P. , *Modélisation de cylindres généralisées en vision artificielle*, Thèse de doctorat de l'université Blaise Pascal , Clermont Ferrand, 18/12/1995.

[SOL 94], SOLINA F., LEONARDIS A., MARCEL A., *A Direct Part-Level Segmentation of Range Images Using Volumetric Models*, IEEE, Robotics and Automation, San Diego, pp. 2254-225 California, May 8-13, 1994.

[TRA 93] TRASSOUDAIN L., *Solution multisensorielle temps reel pour la detection d'obstacles sur route*, thèse de doctorat de l'université Blaise Pascal de Clermont-Ferrand, 16 FEV93.

[YUA 89] YUAN J.S.C., *A General Photogrammetric Method for Determining Object Position and Orientation*, IEEE Trans. on Robotics and Automation, 5(2), pp 129-142, April 198

**Bibliographie sur l'appariement 2D/3D**

- [Ababsa et al., 2000] F. Ababsa « Recalage automatique d'objets 3D de forme libre en utilisant la clinométrie »Thèse de doctorat , LSC, Université d'Evry , 16 décembre 2002.
- [Ababsa et al., 2002] F. Ababsa, D. Roussel et M. Malleme (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
- [AYACHE 1986] Ayache N. and Faugeras O.D., "HYPER: A new approach for the recognition and positioning of two-dimensional objects", *IEEE Trans. PAMI*, 8(1), January 1986, pp.44-54.
- [BINFORD 1993] BINFORD T.O. and LEVITT T.S. "Quasi Invariants : Theory and exploitation" In *Proceeding of DARPA, Image understanding Workshop*, 1993, pp. 819-82
- [BOLLES 1981] Bolles R.C. and Fischler M.A. "A RANSAC-based approach to model fitting and its application to finding cylinders in range data" *Intern. joint conference on Artificial Intelligence*, pp. 637-643, Vancouver, Canada, 1981.
- [BOLLES 1982] Bolles R.C. and Cain R.A. "Recognizing and locating partially visible objects : the local feature-focus method" *Intern. Journal of Robotics Research* , 1(3), 1982, pp. 57-82.
- [BOLLES 1986] Bolles R.C. and Horaud R. "3DPO : A three Dimensional part orientation System " , *Intern. Journal of Robotics Research* , 5(3), 1986, pp. 3-25.
- [BOWYER 1992] Bowyer K.W., "Workshop panel report - why aspect graphs are not (yet) practical for computer vision", *CVGIP: Image Understanding*, vol. 55, No. 2, March 1992, pp. 212-218.
- [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.
- [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.
- [EDELBRUNNER 1986] Edelsbrunner H., Orourke J. and Seidel R., "Constructing arrangements of lines and hyperplanes with applications", *SIAM Journal on Computing*, 15(2), May 1986, pp. 341-363.
- [FAU 93] FAUGERAS O. G., *Three Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press, Boston 1993.
- [GROS 1995]GROS P. "Matching and Clustering : Two steps towards objects modelling in Computer Vision". *The International Journal of Robotics Research*, 14(5), October 1995.
- [HANSEN 1989] Hansen C and Henderson T.C. "CAGD-based Computer Vision", *IEEE Transactions on PAMI*, 11(11), November 1989, pp. 1181-1193.
- [HORAUD 1984] Horaud R. and Bolles R.C. "3DPO's strategy for matching three-dimensional objects in range data", *IEEE Conf. on Robotics & Automation*, Atlanta, March 13-15 1984, pp. 78-85.
- [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.
- [Huttenlocher et Ullman, 1987] D. P. Huttenlocher et S. Ullman (1987). Object recognition using alignment. *IEEE International Conference on Computer Vision*, pages:72-78.
- [GRIMSON 1984] Grimson W.E.L. and Lozano-Perez T., "Model-based recognition and localization from sparse range or tactile data", *International Journal of Robotics Research*, 5(3), Fall 1984, pp. 3-34.
- [IKEUCHI 1987] Ikeuchi K, "Generating an interpretation tree from a CAD model for 3D-Object recognition in bin-picking tasks". *International Journal of Computer Vision* 1(2), 1987, pp. 145-165.
- [IKEUCHI 1988] Ikeuchi K and Kanade T., "Applying sensor models to automatic generation of 3D Object recognition program " , 2<sup>nd</sup> International Conference on Computer Vision, Tampa(FL), November 1988, pp. 228-237.
- [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* 75. 1-8.
- [JAIN 1988] Jain A.K. and Hoffman R., "Evidence-based recognition of 3-D objects", *IEEE Trans. on PAMI*, 10(6), November 1988, pp. 783-802.
- [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.
- [LAMDMAN 1988] Lamdan Y. and Wolfson H.J., "Geometric hashing: A general and efficient model-based recognition scheme", *Second IEEE Int'l Conf. Computer Vision*, Tampa, FL, December 1988, pp. 238-24
- [LOWE 1987] Lowe D.G., "Three dimensional object recognition from single two-dimensional images", *Artificial Intelligence* 31(1987), Elsevier Science Publishers B.V.(North-Holland), pp. 355-395.
- [PAMPAGNIN 1988] Pampagnin L.H. and Devy M., "Identification d'un objet 3D par mise en correspondance entre une seule image et un modèle", *Rapport interne*, LAAS-France, 1988.
- [PLATINGA 1990] Plantinga H. and Dyer C., "Visibility, occlusion and the aspect graph", *Int. Journal of Computer Vision*, 5(2), 1990, pp. 137-160.

- [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.
- [STEWMAN 1988] Stewman J.H. and Bowyer K.W., "Creating the perspective projection aspect graph of polyhedral objects", in *IEEE International Conference on Computer Vision*, 1988, pp. 494-500.
- [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.
- [ZHU 1995] Zhu y., Seneviratne L.D. and Earles S.W.E. "A new structure of invariants for 3D point sets from single view" *IEEE international conference on Robotics and Automation*, Nagaya(Japan), May 21-27, 1995, pp. 1726-1731.
- [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.

Annexe 1 :

**Formalisme mathématique des B-splines :**La courbe B-spline selon n+1 points de contrôle  $P_i$  est donnée par [FOLEY 1990] :

$$Spline(t) = \sum_{i=0}^n \bar{P}_i N_{i,k}(t)$$

où  $N_{i,k}(t)$  est le polynôme de Riesenfeld défini récursivement par :

$$N_{i,1}(t) = \begin{cases} 1 & \text{si } T[i] \leq t \leq T[i+1] \\ 0 & \text{sin on} \end{cases}$$

$$N_{i,j}(t) = \frac{t - T[i]}{T[i+j-1] - T[i]} N_{i,j-1}(t) + \frac{T[i+j] - t}{T[i+j] - T[i+1]} N_{i+1,j-1}(t)$$

Avec comme convention pour le calcul de la formule ci-dessus  $0/0 = 0$  et où T est défini par :

$$T[i] = \begin{cases} 0 & \text{si } i < j \\ i - j + 1 & \text{si } j \leq i \leq d \\ d - k + 1 & \text{si } i \leq n \end{cases}$$