

CHRISTOPHE DEUTSCH

**POURSUITE ET LOCALISATION
D'UNE CIBLE COMPLEXE
PAR VISION MONOCULAIRE**

Thèse
présentée
à la Faculté des études supérieures
de l'Université Laval
pour l'obtention
du grade de Philosophiae Doctor (Ph.D.)

DÉPARTEMENT DE GÉNIE ÉLECTRIQUE
FACULTÉ DES SCIENCES ET DE GÉNIE
UNIVERSITÉ LAVAL
QUÉBEC

OCTOBRE 2001



**National Library
of Canada**

**Acquisitions and
Bibliographic Services**

**395 Wellington Street
Ottawa ON K1A 0N4
Canada**

**Bibliothèque nationale
du Canada**

**Acquisitions et
services bibliographiques**

**395, rue Wellington
Ottawa ON K1A 0N4
Canada**

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-66312-4

Canada

Résumé court

L'objectif de cette thèse est de trouver une méthode efficace de poursuite et de localisation d'un objet 3D connu à partir d'une séquence d'images 2D. Les déplacements saccadés sont autorisés. L'approche proposée est basée sur une modélisation originale de la cible par un graphe d'aspect dont les vues sont définies par des contours d'occultation. Pour calculer les six paramètres définissant la pose de l'objet, plusieurs outils ont dû être développés. Une paramétrisation des contours d'occultation permet ainsi de retrouver l'un des paramètres de rotation et le facteur d'échelle. Les deux autres paramètres de rotation sont calculés à partir du graphe d'aspect. Enfin, les deux paramètres de translation sont obtenus grâce à une adaptation de l'algorithme de poursuite 2D de Huttenlocher. Les résultats obtenus sont très bons et laissent présager un bel avenir à cet algorithme innovateur. De plus, plusieurs outils développés tout au long de cette thèse sont généraux et pourront être utilisés avec profit dans d'autres contextes.

Résumé long

L'estimation de la position exacte d'un objet est un problème générique complexe. Les approches sont très différentes suivant la connaissance *a priori* de l'objet, le type de capteurs et le mouvement de la cible. Dans la recherche que nous proposons, il s'agit de retrouver la position d'un objet connu et rigide à partir d'une caméra 2D fixe. Le mouvement de l'objet peut être saccadé.

Les algorithmes classiques utilisés pour la poursuite de cible, requièrent toujours une étape de prédiction. Cette étape peut être plus ou moins complexe et se fait habituellement par un filtrage de Kalman. Plus l'estimation est proche de la pose réelle, plus ces algorithmes sont performants. À cause des contraintes de mouvements saccadés, il n'est donc pas possible de résoudre ce problème en utilisant une approche classique.

La solution que nous proposons est basée sur une modélisation par graphe d'aspect de la cible. Chaque vue 2D du graphe d'aspect correspond à un contour d'occultation. Afin de pouvoir exploiter au mieux les informations des contours d'occultation, nous avons mis au point une paramétrisation dont les propriétés particulières permettent de retrouver l'angle de rotation et le facteur d'échelle.

La détermination des six paramètres de pose de la cible se fait en trois étapes. La rotation dans le plan image et le facteur d'échelle sont obtenus à partir de la paramétrisation du contour

d'occultation. Les deux autres paramètres de rotation sont obtenus à partir du graphe d'aspect. Enfin, les deux derniers paramètres de translation sont calculés à partir de la position de l'objet dans l'image 2D grâce à une adaptation de l'algorithme de Huttenlocher. Comme nous assumons que les images 2D sont formées selon la projection de perspective, les deux paramètres de rotation sont ajustés en fonction des deux paramètres de translation.

Les résultats obtenus par notre approche prouvent que l'utilisation d'un graphe d'aspect pour l'estimation de la position 3D d'un objet est une avenue prometteuse qui avait été négligé jusqu'ici. Plusieurs séquences d'images de synthèse ont été testées. Avec ce type d'images, il est possible de simuler n'importe quel mouvement tout en connaissant parfaitement les paramètres de pose de l'objet 3D. La précision de l'estimation de pose dépend du type d'objet étudié ainsi que de la résolution des images 2D. Dans nos simulations nous avons obtenu des valeurs d'erreur de l'ordre de 2 à 3 degrés pour les paramètres de rotation. Cette estimation pourrait être améliorée si notre approche était prise comme point de départ dans des algorithmes plus classiques. Aucun autre algorithme n'offre actuellement de telles performances avec des mouvements saccadés.

Avant-propos

À mon retour au Québec en 1995, j'ai entrepris des études doctorales au laboratoire de vision numérique de l'université Laval. Il s'agissait plus d'un cheminement naturel que d'un véritable choix éclairé. J'avais toujours aimé étudier, encouragé par des professeurs que j'admirais. J'aimerais citer ici ceux qui m'ont le plus marqué : Georges Divry, instituteur exigeant, humain et bon avec les enfants, Anne Miguet, brillante mathématicienne qui a su me faire apprécier la beauté de cette discipline et enfin André Zaccarin, mon directeur de thèse, qui m'a donné la piquûre pour la recherche et le travail rigoureux.

Une thèse de doctorat c'est long. Il aurait été impossible de traverser ces années où se sont succédés le découragement et l'enthousiasme sans le support de mes amis et de ma famille. Ces êtres chers m'ont soutenu et encouragé, mais ils m'ont surtout permis d'avoir une vie équilibrée. Je voudrais remercier tout particulièrement Marie-Louise et Paul, Michelle et Yvon, Michèle, Laurent, Jérôme et Florian, Isabelle, Jean-Sébastien et Émile, Phillippe, Martine et Alexandre, ainsi que tous ceux qui ont cru en moi. Un grand merci aussi à Annie, Jean-François, Véro, Arnaud, Denis, JF, Agnès, Pascal, Franck, François, Svetlana, Marielle, Jacques, Antoine pour leur amitié.

On commence souvent un doctorat jeune adulte, on le finit parfois père de famille. J'ai eu cette chance puisque en 1999 et en 2001 sont nés Cécile et Léo. J'aurais préféré qu'ils aient un papa plus présent pour leurs premières années de vie. Je leur promets de l'être plus à l'avenir.

On garde souvent le meilleur pour la fin, je le fais aussi. C'est à ma femme Julie que vont mes plus sincères remerciements. Les dernières années de cette thèse ont été difficiles et c'est grâce à elle que je suis allé jusqu'au bout. J'aimerais qu'elle sache combien je l'aime pour tout ce qu'elle est.

Table des matières

| | |
|---|------|
| RÉSUMÉ COURT | I |
| RÉSUMÉ LONG | II |
| AVANT-PROPOS | IV |
| TABLE DES MATIÈRES..... | V |
| Liste des figures..... | IX |
| Liste des tableaux..... | XII |
| Liste des symboles..... | XIII |
| INTRODUCTION..... | I |
| Poursuite de cible et estimation 3D de la pose..... | I |
| Modélisation 3D..... | 4 |
| Plan de la thèse..... | 4 |
| CHAPITRE 1: MODÉLISATION 2D | |
| 1.1 Introduction..... | 7 |
| 1.2 État de l'art..... | 8 |
| 1.3 Modélisation proposée..... | 10 |
| 1.3.1 Esquisse d'une solution..... | 10 |
| 1.3.2 Paramétrisation polaire du modèle silhouette..... | 12 |
| 1.3.2.1 Algorithme de calcul..... | 12 |
| 1.3.2.2 Vérifications théoriques des propriétés..... | 20 |
| 1.3.2.3 Exemple..... | 25 |
| 1.3.2.4 Discussion sur les paramètres de la modélisation..... | 27 |
| 1.3.3 Mesure de comparaison..... | 29 |

| | | |
|------------------------------------|---|----|
| 1.3.3.1 | Choix de la mesure | 29 |
| 1.3.3.2 | Algorithme de calcul de la mesure de comparaison | 32 |
| 1.3.3.3 | Exemple d'utilisation de la mesure de comparaison | 38 |
| 1.3.3.4 | Conclusion | 39 |
| 1.4 | Expérimentation | 40 |
| 1.4.1 | Vérification empirique des propriétés de la modélisation | 40 |
| 1.4.1.1 | Comparaison avec des mesures existantes | 40 |
| 1.4.1.2 | Évaluation des effets du facteur d'échelle | 42 |
| 1.4.1.3 | Évaluation des effets des rotations | 46 |
| 1.4.1.4 | Conclusion | 49 |
| 1.4.2 | Quelques exemples d'utilisation de la modélisation proposée | 50 |
| 1.4.2.1 | Estimation du facteur d'échelle | 50 |
| 1.4.2.2 | Estimation de la rotation | 51 |
| 1.4.2.3 | Estimation de la rotation et du facteur d'échelle | 53 |
| 1.4.2.4 | Retrouver un objet dans une base de données | 56 |
| 1.4.2.5 | Conclusion | 58 |
| 1.4.3 | Variation des paramètres de la modélisation | 58 |
| 1.4.3.1 | Centre de gravité | 58 |
| 1.4.3.2 | Facteur de discrétisation angulaire | 59 |
| 1.4.3.3 | Facteur de discrétisation radiale | 60 |
| 1.5 | Conclusion | 60 |
| | | |
| CHAPITRE 2: GRAPHE D'ASPECT | | |
| 2.1 | Introduction | 62 |
| 2.2 | État de l'art | 64 |
| 2.2.1 | Définition du graphe d'aspect | 64 |
| 2.2.2 | Génération de graphes d'aspect | 66 |
| 2.2.2.1 | Calcul exact | 66 |
| 2.2.2.2 | Calcul approximatif | 67 |
| 2.2.3 | Utilisation des graphes d'aspect | 67 |
| 2.2.4 | Conclusion | 69 |
| 2.3 | Graphe d'aspect et modélisation 2D | 69 |
| 2.3.1 | Création des différentes vues | 69 |
| 2.3.2 | Quelques exemples de graphes d'aspect | 72 |
| 2.3.3 | Modélisation de chacune des vues du graphe | 72 |
| 2.3.4 | Conclusion | 73 |
| 2.4 | Graphe d'aspect et estimation de pose | 75 |
| 2.4.1 | Recherche exhaustive dans le graphe d'aspect complet | 75 |
| 2.4.1.1 | Description de l'algorithme | 75 |
| 2.4.1.2 | Résultats | 77 |

| | |
|--|-----|
| 2.4.1.3 Conclusion | 88 |
| 2.4.2 Analyse par composantes principales. | 88 |
| 2.4.2.1 Principe de base | 89 |
| 2.4.2.2 Alignement des modèles | 91 |
| 2.4.2.3 Résultats | 95 |
| 2.4.2.4 Conclusion sur l'approche par ACP. | 97 |
| 2.5 Conclusion | 98 |
| CHAPITRE 3: POURSUITE ET LOCALISATION DE CIBLE: UNE NOUVELLE APPROCHE | |
| 3.1 Introduction. | 99 |
| 3.2 État de l'art | 100 |
| 3.2.1 Introduction | 100 |
| 3.2.2 Méthode de poursuite 2D | 100 |
| 3.2.2.1 Approche de Huttenlocher | 100 |
| 3.2.2.2 Autres approches de poursuite 2D | 104 |
| 3.2.2.3 Conclusion | 107 |
| 3.2.3 Poursuite de cible avec estimation 3D de la pose. | 108 |
| 3.2.3.1 Principe général | 108 |
| 3.2.3.2 Étude comparative de quelques approches particulières | 110 |
| 3.2.3.3 Autres méthodes. | 115 |
| 3.2.4 Conclusion | 115 |
| 3.3 Approche proposée | 116 |
| 3.3.1 Paramètres d'entrée | 116 |
| 3.3.2 Description détaillée de l'algorithme | 117 |
| 3.3.2.1 Détection des contours d'occultation dans l'image | 117 |
| 3.3.2.2 Extraction des informations pertinentes dans l'image | 118 |
| 3.3.2.3 Calcul de la paramétrisation des contours extraits de l'image | 119 |
| 3.3.2.4 Utilisation du graphe d'aspect | 121 |
| 3.3.2.5 Calcul des paramètres translationnels | 121 |
| 3.3.2.6 Calcul de la pose | 122 |
| 3.3.2.7 Passage à l'image suivante et mise à jour du modèle actif | 127 |
| 3.3.2.8 Initialisation | 128 |
| 3.4 Résultats | 128 |
| 3.4.1 Séquences de tests utilisées | 128 |
| 3.4.2 Analyse des résultats | 130 |
| 3.4.2.1 Évaluation des performances | 130 |
| 3.4.2.2 Analyse globale des résultats | 130 |
| 3.4.2.3 Précision du graphe d'aspect et de la modélisation 2D | 133 |
| 3.4.2.4 Type de mouvement. | 134 |
| 3.4.2.5 Importance du facteur de dilatation | 134 |

Table des matières

| | |
|---|-----|
| 3.4.2.6 Méthode de mise à jour du contour actif | 135 |
| 3.4.2.7 Importance de l'exactitude des contours extraits. | 136 |
| 3.4.3 Considération de complexité algorithmique. | 137 |
| 3.5 Conclusion et travaux futurs. | 138 |
| CONCLUSION | 151 |
| Principales contributions | 151 |
| Travaux futurs. | 153 |
| ANNEXE: Réduction du graphe d'aspect | |
| A1 Introduction | 155 |
| A2 Représentation vectorielle du graphe d'aspect. | 156 |
| A3 Regroupement de différentes vues | 156 |
| A4 Conclusion | 159 |
| BIBLIOGRAPHIE | 163 |

Liste des figures

CHAPITRE 1: MODÉLISATION 2D

| | |
|--|----|
| Figure 1.1 Discrétisation angulaire. | 15 |
| Figure 1.2 Définition des zones de calculs pour les angles extrêmes d'un pixel de contour. | 16 |
| Figure 1.3 Discrétisation radiale. | 18 |
| Figure 1.4 Vérification de la proportionnalité par rapport au facteur d'échelle. | 21 |
| Figure 1.5 Vérification de l'invariance par rapport aux rotations. | 23 |
| Figure 1.6 Exemple de paramétrisation d'un modèle de silhouette. | 26 |
| Figure 1.7 Exemple d'échantillonnage insuffisant sur une courbe particulière. | 27 |
| Figure 1.8 Modèles ayant le même spectre d'amplitude, mais un spectre de phase distinct. | 31 |
| Figure 1.9 Exemples de calcul de la mesure de comparaison. | 38 |
| Figure 1.10 Positions respectives des contours pour le calcul des distances de Hausdorff et de chamfer. | 41 |
| Figure 1.11 Évaluation des performances de la mesure de comparaison - premier exemple. | 43 |
| Figure 1.12 Évaluation des performances de la mesure de comparaison - second exemple. | 44 |
| Figure 1.13 Effets du facteur d'échelle sur la distance. | 45 |
| Figure 1.14 Effets des rotations sur la paramétrisation - premier exemple. | 47 |
| Figure 1.15 Effets des rotations sur la paramétrisation - second exemple | 48 |
| Figure 1.16 Récupération du facteur d'échelle. | 50 |
| Figure 1.17 Récupération de l'angle - contours identiques. | 52 |
| Figure 1.18 Récupération de l'angle - contours différents. | 53 |

| | |
|--|-----|
| Figure 1.19 Récupération du facteur d'échelle et de l'angle de rotation - contours identiques. | 54 |
| Figure 1.20 Récupération du facteur d'échelle et de l'angle de rotation- contours différents | 55 |
| Figure 1.21 Base de données de contours. | 56 |
| Figure 1.22 Utilisation d'une base pour classer des contours. | 57 |
| Figure 1.23 Variation de la position du centre de gravité | 59 |
| CHAPITRE 2: GRAPHE D'ASPECT | |
| Figure 2.1 Graphe d'aspect d'un objet simple. | 65 |
| Figure 2.2 Relation entre la position dans la sphère de points de vue et les angles d'Euler de l'objet 3D. | 71 |
| Figure 2.3 Graphe d'aspect de la théière. | 73 |
| Figure 2.4 Graphe d'aspect de l'avion. | 74 |
| Figure 2.5 Résultats d'estimation de pose avec facteur d'échelle variable. | 81 |
| Figure 2.6 Résultats d'estimation de pose avec angle de rotation 2D aléatoire. | 82 |
| Figure 2.7 Résultats d'estimation de pose avec facteur d'échelle et angle de rotation 2D fixes. | 83 |
| Figure 2.8 Résultats d'estimation de pose entièrement aléatoire. | 84 |
| Figure 2.9 Plus grandes erreurs lors des simulations d'estimation de pose de la théière. | 85 |
| Figure 2.10 Calcul de l'alignement en rotation 2D. | 94 |
| Figure 2.11 Résultats d'estimation de pose avec une approche par composantes principales. | 96 |
| CHAPITRE 3: POURSUITE ET LOCALISATION DE CIBLE: UNE NOUVELLE APPROCHE | |
| Figure 3.1 Poursuite de cible avec la méthode de Huttenlocher (modèle à taille variable) | 105 |
| Figure 3.2 Poursuite de cible avec la méthode de Huttenlocher (modèle à taille fixe). | 106 |
| Figure 3.3 Schéma de principe des algorithmes d'estimation de pose | 108 |
| Figure 3.4 Exemple illustrant le principe d'action des algorithmes d'estimation de pose. | 109 |
| Figure 3.5 Description de l'algorithme de calcul de pose. | 120 |
| Figure 3.6 Calcul des paramètres de translation. | 122 |
| Figure 3.7 Calcul des paramètres de rotation. | 125 |
| Figure 3.8 Erreur d'approximation à cause de la projection de perspective. | 127 |
| Figure 3.9 Séquence d'images 1: l'avion. | 139 |
| Figure 3.10 Séquence d'images 2: le téléphone (mouvement de rotation). | 140 |

| | |
|--|-----|
| Figure 3.11 Séquence d'images 3: le poisson (mouvement de translation) | 141 |
| Figure 3.12 Séquence d'images 4: la voiture (mouvement de zoom). | 142 |
| Figure 3.13 Séquence d'images 5: la théière (mouvement saccadé: rotation et translation). | 143 |
| Figure 3.14 Résultats avec la séquence de l'avion avec des précisions différentes pour le graphe d'aspect et la paramétrisation. | 144 |
| Figure 3.15 Résultats avec la séquence du téléphone avec des précisions différentes pour le graphe d'aspect et la paramétrisation. | 145 |
| Figure 3.16 Résultats avec la séquence du poisson avec des précisions différentes pour le graphe d'aspect et la paramétrisation. | 146 |
| Figure 3.17 Résultats avec la séquence de la voiture. | 147 |
| Figure 3.18 Résultats avec la séquence de la théière avec des précisions différentes pour le graphe d'aspect et la paramétrisation. | 148 |
| Figure 3.19 Ajustement du facteur de dilatation. | 149 |
| Figure 3.20 Résultats obtenus avec une extraction de contour de type détection d'arêtes ou de type segmentation. | 150 |
| ANNEXE: Réduction du graphe d'aspect | |
| Figure A1 Projection des vues du graphe d'aspect de la théière dans le plan. | 157 |
| Figure A2 Projection des vues du graphe d'aspect du poisson dans le plan. | 158 |
| Figure A3 Résultats de l'algorithme du «Fuzzy K-Mean» avec 6 aspects pour le modèle de la théière | 160 |
| Figure A4 Résultats de l'algorithme de «Fuzzy K-Mean» pour le poisson et pour l'avion. | 161 |
| Figure A5 Exemple de graphe d'aspect réduit. | 162 |

Liste des tableaux

CHAPITRE 1 : MODÉLISATION 2D

| | |
|---|----|
| Tableau 1.1 Distance de Hausdorff. | 43 |
| Tableau 1.2 Mesure δ_{symLinf} | 43 |
| Tableau 1.3 Distance de chamfer. | 43 |
| Tableau 1.4 Mesure δ_{symL1} | 43 |
| Tableau 1.5 Mesure δ_{symL1} effective. | 43 |
| Tableau 1.6 Distance de Hausdorff. | 44 |
| Tableau 1.7 Mesure δ_{symLinf} | 44 |
| Tableau 1.8 Distance de chamfer. | 44 |
| Tableau 1.9 Mesure δ_{symL1} | 44 |
| Tableau 1.10 Mesure δ_{symL1} effective. | 44 |

CHAPITRE 2 : GRAPHE D'ASPECT

CHAPITRE 3 : POURSUITE ET LOCALISATION DE CIBLE: UNE NOUVELLE APPROCHE

| | |
|---|-----|
| Tableau 3.1 Résumé des méthodes 3D - Les algorithmes. | 112 |
| Tableau 3.2 Résumé des méthodes 3D - Les performances. | 114 |
| Tableau 3.3 Erreur moyenne et écart-type pour chacun des paramètres de pose. | 131 |
| Tableau 3.4 Erreur absolue moyenne pour chacun des paramètres de pose. | 132 |

Liste des symboles

Notation générale

| | |
|---------------------|---|
| $ $ | Module complexe ou cardinal d'un ensemble (suivant le contexte). |
| $X[n]$ | $n^{\text{ième}}$ composante du vecteur X . |
| $\langle \rangle$ | Produit scalaire entre deux vecteurs. |
| $\sigma^{(k)}$ | Permutation circulaire d'ordre k . |
| $(X \bullet Y)$ | Produit de convolution circulaire entre les vecteurs X et Y . |
| $TFD(X)$ | Transformée de Fourier discrète du vecteur X . |
| $TFI(X)$ | Transformée de Fourier discrète inverse du vecteur X . |
| \bar{X} | Conjugué du vecteur X . |
| $\ \ _{L_p}$ | Norme L_p . |
| $a \equiv b(x)$ | Égalité modulo x . |
| $\{ \{ \}$ | Intervalle de nombre entier. |

Notation reliée au chapitre 1

| | |
|--------------------------|--|
| V_p | Vecteur de paramétrisation d'un contour d'occultation. |
| A_p | Spectre d'amplitude du vecteur de paramétrisation. |
| P_p | Spectre de phase du vecteur de paramétrisation. |
| N | Taille du vecteur de paramétrisation. |
| Θ_s | Facteur de discrétisation angulaire. |
| N_s | Facteur de discrétisation radiale. |
| C | Ensemble des pixels d'un contour d'occultation. |
| f_e | Facteur d'échelle. |
| $f_e^{i/j}$ | Facteur d'échelle permettant de ramener le contour C_j à l'échelle de C_i . |
| $\varphi^{i/j}$ | Angle de rotation permettant de ramener le contour C_j le plus proche possible du contour C_i . |
| $\delta(C_i, C_j)$ | Mesure de comparaison entre les contours C_i et C_j . |
| $\delta_{sym}(C_j, C_i)$ | Mesure de comparaison symétrique entre les contours C_i et C_j . |
| $d_H(P, Q)$ | Distance de Hausdorff entre les contours P et Q. |
| $d_C(P, Q)$ | Distance de chamfer entre les contours P et Q. |

Notation reliée au chapitre 2

| | |
|---------------------|--|
| N_φ, N_ψ | Facteurs de discrétisation de la sphère de point de vue. |
| φ | Premier angle d'Euler |

| | |
|----------------------------|---|
| Ψ | Deuxième angle d'Euler. |
| Γ_d | Ensemble discret de points correspondant à une tessellation régulière de la sphère. |
| $Z_{\varphi\Psi}$ ou Z_g | Distance caméra-objet pour les différentes vues d'un graphe d'aspect. |

Notation reliée au chapitre 3

| | |
|--|---|
| R_{dil} | Rayon de dilatation. |
| I_n | Image binaire correspondant aux contours d'occultation, n est l'indice dans la séquence d'images. |
| $\left(\begin{array}{c} x_n, y_n, z_n \\ \theta_n, \varphi_n, \Psi_n \end{array} \right)$ | Paramètres de pose de l'objet 3D à l'instant n . |
| X_{offset}, Y_{offset} | |
| <i>Focalfactor</i> | Paramètres intrinsèques de la caméra. |
| <i>AspectRatio</i> | |

Introduction

Poursuite de cible et estimation 3D de la pose

Parmi les axes de développement importants en vision, l'analyse de l'environnement à l'aide de différents capteurs tient une place prépondérante. Il est, par exemple, très important de pouvoir situer exactement certains objets dans une scène lorsqu'on désire interagir avec ces objets. Les applications de ce type de situations sont très nombreuses: le déplacement d'un robot dans un environnement hostile ou dangereux, l'aide à la manipulation d'un bras télé-opéré ou encore la poursuite de cible.

L'analyse de l'environnement est un problème bien trop large pour le traiter dans son ensemble. Chaque chercheur n'en aborde qu'un aspect particulier en essayant de trouver une solution dans un contexte défini. Ainsi, nous nous intéresserons ici au cas de la poursuite d'un objet 3D connu, observé par une caméra 2D. L'objectif de cette recherche est de retrouver les six paramètres de pose d'un objet rigide en mouvement à partir d'une séquence d'images 2D. Il s'agit d'un problème difficile qui a déjà été souvent abordé. Il est donc important de définir très clairement la question et toutes les hypothèses qui s'appliquent précisément à notre étude. Ceci permettra au lecteur d'identifier sans équivoque les contributions que nous apporterons au sujet.

La première hypothèse concerne l'objet qui est observé. Celui-ci doit être rigide¹ et sa géométrie parfaitement connue. De plus, on suppose que dans les images 2D, l'objet apparaît toujours en entier. Les occultations partielles ne seront donc pas traitées.

La seconde hypothèse se rapporte au mouvement de l'objet dans la séquence d'image 2D. Dans tous les algorithmes de poursuite de cible connus actuellement, le mouvement doit toujours être limité ou, tout au moins, prédictible.

Il existe cependant de nombreux cas de figure pour lesquels le mouvement est plus complexe. Prenons par exemple le cas de la manipulation d'une nacelle d'Hydro-Québec qui permet le remplacement d'isolateurs sur les lignes de transport d'électricité. Cette nacelle supporte deux bras manipulés par un opérateur au sol. L'opérateur reçoit des images stéréo par l'intermédiaire de caméras situées sur la nacelle. En raison du vent et de la manipulation des bras, la nacelle bouge souvent avec un mouvement saccadé. Les images reçues par l'opérateur le sont donc elles aussi et la télé-opération des bras devient très difficile. L'utilisation d'une autre caméra observant la nacelle à partir du sol et estimant la position de celle-ci permettrait de compenser les mouvements saccadés et d'offrir des images stabilisées de la scène à l'opérateur.

Des mouvements saccadés apparaissent également dans toutes les applications pour lesquelles l'échantillonnage temporel lors de l'acquisition des images 2D est trop lent par rapport au mouvement de l'objet observé.

Ces deux exemples montrent que la résolution du problème de poursuite de cible dans un contexte de mouvements saccadés présente un réel intérêt. Nous limiterons par conséquent le moins possible le mouvement de l'objet observé et des mouvements saccadés devront pouvoir être traités.

Nous assumerons que la projection qui forme les images est une projection de perspective. Cette hypothèse n'est pas restrictive puisqu'il s'agit du cas le plus réaliste et le plus difficile à traiter.

1. Les objets articulés ne sont pas considérés comme rigides dans cette étude.

De nombreux travaux préfèrent se limiter à une projection orthogonale qui simplifie nettement le problème. Nous supposons que la caméra qui fait les acquisitions 2D est entièrement calibrée et que ses paramètres intrinsèques sont connus.

Avant de pouvoir parler de la dernière hypothèse que nous avons faite, nous devons introduire la notion de contour d'occultation. Le contour d'occultation d'un objet correspond à la frontière entre l'objet lui-même et l'arrière-plan dans une image 2D. C'est une information généralement plus facile à extraire que d'autres types de contours tels que ceux formés par les arêtes de discontinuité de courbure, de changement de texture ou de changement de couleur. C'est à cause de cette caractéristique que nous avons choisi ce type de primitive pour effectuer la poursuite d'un objet dans une séquence d'image 2D. Dans le cas d'images réelles, une méthode d'extraction permettant d'identifier très précisément les contours d'occultation n'est toutefois pas encore au point. Il existe cependant de nombreuses méthodes efficaces et nous indiquerons des approches possibles pour résoudre ce problème, sans toutefois les expliquer car leur utilisation dépasserait largement le contexte de nos recherches. Dans nos travaux, nous ferons donc l'hypothèse que l'extraction des contours d'occultation est une étape acquise.

L'objectif de cette thèse est de trouver une nouvelle méthodologie pour aborder le problème de poursuite de cible avec estimation de la pose. Il s'agit d'un défi de taille. Pour atteindre cet objectif, nous effectuerons des tests sur des séquences d'images. Puisque nous avons fait l'hypothèse qu'il est possible d'extraire adéquatement les contours d'occultation, nous travaillerons avec des séquences d'images de synthèse plutôt qu'avec des images réelles. L'utilisation de telles images offrira un avantage de taille puisque le mouvement de l'objet 3D ainsi que l'objet 3D lui-même pourront être parfaitement connus. Il est ainsi possible d'avoir une référence absolue pour tester les performances intrinsèques de l'algorithme.

Modélisation 3D

En fonction des hypothèses que nous venons de présenter, nous aurons besoin d'un modèle géométrique 3D de la cible permettant de générer aisément un contour d'occultation dans le plan image.

Il existe de nombreuses modélisations 3D dans la littérature: les modèles en fil de fer [71], les modèles surfaciques (approximations polygonales [61], triangulation, etc...), la modélisation par CAO, les modèles volumétriques (super-quadriques [66], cylindres généralisés, etc...), et bien d'autres encore. Dans notre cas, une approche surfacique est parfaitement adéquate. En effet, grâce à l'approche par triangulation il est possible de représenter n'importe quelle forme réelle.

De manière concrète, nous utiliserons le langage VRML¹ pour encoder nos modèles surfaciques. De nombreux outils permettent de travailler avec VRML (3DstudioMax, Moonlight Creator, etc...) et il est très facile de projeter un modèle VRML dans une image pour en extraire le contour d'occultation.

De plus, la description d'objet avec VRML peut aussi contenir des informations de texture et de couleur. Celles-ci ne sont pas exploitées directement dans notre approche, mais elles permettent de générer des séquences d'images virtuelles relativement réalistes.

Plan de la thèse

Après avoir décrit l'objectif et les hypothèses de cette thèse ainsi que l'outil de modélisation 3D choisi, nous pouvons passer à la description détaillée de la méthode que nous suggérons pour résoudre le problème.

Comme nous l'avons mentionné, les contours d'occultation nous permettront de calculer la pose 3D de l'objet. Il faut donc introduire une méthode permettant de les comparer entre eux. C'est

1. *Virtual Reality Modeling Language*

ce que nous décrivons dans le premier chapitre. L'approche suivie consiste à transformer un contour d'occultation en vecteur de paramétrisation polaire de celui-ci. En comparant les vecteurs entre eux, on retrouve toute rotation 2D ou changement d'échelle existant entre deux contours. Cette méthode est très efficace comme le prouvent les différentes simulations. La mise au point de cette paramétrisation offre de nouvelles avenues à de nombreuses applications. Nous nous concentrerons cependant à son utilisation pour le calcul de la pose 3D d'un objet.

Afin d'utiliser la paramétrisation 2D introduite au chapitre 1, il faut connaître les différents aspects que peut prendre l'objet 3D dans une image 2D. L'outil le plus approprié pour décrire ces différentes vues est, bien entendu, le graphe d'aspect. Il s'agit d'un outil avec beaucoup de potentiel qui n'a cependant jamais été utilisé pour du calcul de pose à cause de certains défauts. Dans le chapitre 2, nous avons réussi à redonner au graphe d'aspect la place qu'il méritait en le combinant à notre paramétrisation 2D. Nous l'avons en effet utilisé pour retrouver les quatre paramètres de pose d'un objet centré à l'origine du plan image. De nombreuses simulations montrent l'efficacité de la méthode.

En combinant la méthode de poursuite de cible 2D de Huttenlocher à nos outils de paramétrisation 2D et de graphe d'aspect décrit dans les deux premiers chapitres, nous avons pu mettre au point une méthode de poursuite de cible efficace dans un contexte de mouvements saccadés. Le chapitre 3 explique l'approche que nous avons suivie et montre les performances obtenues à partir de séquences d'images de synthèse. La discrétisation du graphe d'aspect et de la paramétrisation 2D limitent la précision de l'estimation de la pose. Cette estimation est toutefois très proche de la pose réelle. Il est donc facile d'imaginer une approche qui permettrait de raffiner notre estimation de la pose par des méthodes classiques comme celles de Lowe [63] ou de Koller [51].

Le dernier chapitre de cette thèse résume les contributions principales de nos travaux en montrant les améliorations qui peuvent être apportées, ainsi que les travaux futurs à effectuer.

Une annexe présente également des travaux préliminaires sur la réduction des graphes d'aspect. Les résultats obtenus pourront servir de points de départ pour des recherches à venir.

1.1 Introduction

Nous supposons que la géométrie de l'objet 3D à poursuivre est entièrement connue et qu'il sera très facile de projeter cet objet dans le plan image pour en extraire son contour d'occultation théorique. L'utilisation des contours d'occultation associée au principe du graphe d'aspect permettra d'effectuer le suivi et la localisation de l'objet 3D. Pour cela, nous aurons besoin d'une modélisation du contour d'occultation pertinente. Elle devra:

1. permettre de décrire aisément un ensemble très varié de contours;
2. permettre de comparer deux contours à l'aide d'une mesure qui est basée sur la ressemblance et qui ne tient compte ni du facteur d'échelle ni de la rotation 2D entre les deux contours;
3. permettre d'isoler le facteur d'échelle entre deux contours;
4. permettre de calculer l'angle de la rotation 2D entre deux contours.

La première propriété permettra de tenir compte des variations importantes du contour d'occultation selon le point de vue d'observation. Les contours composés d'une ou de plusieurs courbes fermées devront donc être considérés.

La seconde propriété rendra possible non seulement la comparaison des différents contours du graphe d'aspect avec les données extraites d'une image, mais aussi la comparaison des contours

du graphe d'aspect entre eux. La première comparaison sera utilisée pour localiser la cible, pendant que la seconde donnera des informations précises sur les variations d'aspects des contours le long de la sphère de point de vue.

Les troisième et quatrième propriétés seront très utiles pour retrouver facilement l'objet dans une scène sans connaître ni le facteur d'échelle ni le paramètre de rotation 2D. Elles permettront aussi de calculer le facteur d'échelle et l'angle de rotation existant entre un contour connu et un autre inconnu.

Les rotations 2D que nous allons considérer dans ce chapitre correspondent à des rotations de l'objet 3D par rapport à un axe parallèle à celui de la caméra et centré à l'origine du repère de l'objet. Pour le contour d'occultation, il s'agit donc d'une rotation 2D dont le centre est la projection de l'origine du repère de l'objet 3D.

Avant de décrire la modélisation que nous proposons, nous ferons un rapide survol de la littérature dans le paragraphe 1.2. La paramétrisation du contour que nous avons retenue sera décrite très précisément dans la section 1.3. La description détaillée de son calcul ainsi que la démonstration théorique de sa performance y seront faites. Le paragraphe 1.4 regroupera toutes les expérimentations et montrera la puissance pratique de l'approche proposée.

1.2 État de l'art

Une des modélisations les plus simples pour représenter un contour 2D est une représentation par image binaire. Les points du contour correspondent alors aux pixels de valeur 1. Nous utiliserons le terme *silhouette*¹ pour faire référence à cette approche. Cette modélisation offre de nombreux avantages puisque le contour représenté ne doit respecter aucune contrainte de structure (ensemble de points connectés, contours fermés, ensemble de segments droits, etc...). H.T.

1. Il est important de noter que le terme *silhouette* est utilisé ici dans un sens différent de celui qui est habituellement utilisé en vision numérique où il représente généralement les contours d'occultation.

Huttenlocher [41-44] a largement utilisé cette modélisation pour effectuer de la poursuite de cible à l'aide d'une séquence d'images monoculaires. Il est cependant difficile de retrouver un facteur d'échelle ou une possible rotation entre deux contours représentés par des silhouettes.

La modélisation par «chain code» décrite par Freeman [31], décrit un contour constitué d'un ensemble de points connexes ayant au moins deux voisins. Cette approche très simple ne permet pas de tenir compte directement du facteur d'échelle, mais en l'associant au principe des descripteurs de Fourier, on peut obtenir une paramétrisation invariante aux rotations et au facteur d'échelle [57]. Il faut noter que les descripteurs de Fourier peuvent aussi être définis directement à partir des coordonnées du contour dans un repère cartésien. Le principe reste cependant sensiblement le même. L'inconvénient de cette approche est la difficulté de donner un sens concret à une métrique définie sur les descripteurs de Fourier. De plus, les contours traités doivent nécessairement être continus et ne contenir qu'une seule courbe fermée. Les contours d'occultation, quant à eux, contiennent souvent plusieurs courbes fermées.

Il existe ensuite toute une panoplie de modélisation de contour basée sur les B-splines. Ce type de modélisation est très pratique parce qu'il permet de représenter des objets relativement complexes et non-rigides [4,10,13,19,20,21,52-54]. Il permet également de représenter un ensemble de contours semblables avec très peu de paramètres en utilisant un principe d'analyse par composantes principales. Pour l'application que nous recherchons, cette approche est cependant inappropriée.

Les contours actifs ou «snakes» représentent un type de modélisation souvent utilisé dans des applications de poursuite de cible 2D. Cette représentation permet de capturer de manière physique¹ les informations extraites des images 2D [23,46,60]. Il s'agit, là encore, d'une approche inutilisable avec des notions de facteurs d'échelle ou de rotation 2D. De plus les mouvements entre deux images 2D successives doivent rester faibles pour que l'approche demeure robuste.

1. À l'aide de forces agissant sur le contour et de fonctions de minimisation d'énergie.

La représentation d'un contour par un ensemble de segments de droite [1] permet, par une méthode d'appariement de primitives, d'estimer une rotation existant entre deux contours différents. La comparaison devient toutefois difficile lorsqu'il faut tenir compte du facteur d'échelle. Des hypothèses sur les positions respectives des deux contours sont alors nécessaires. De plus, les objets complexes nécessitent souvent un grand nombre de segments de petites tailles, ce qui rend la tâche encore plus délicate.

Après un survol rapide de la littérature, la modélisation qui se rapproche le plus de notre objectif de calcul du facteur d'échelle et d'angle de rotation 2D entre deux contours distincts, est certainement celle utilisant les descripteurs de Fourier. Cette approche ne permet malheureusement pas de traiter convenablement des contours d'occultation formés de plusieurs courbes fermées. Nous avons donc décidé de définir notre propre modélisation.

1.3 Modélisation proposée

1.3.1 Esquisse d'une solution

La modélisation la plus simple que nous avons vue est le modèle de silhouette 2D décrit dans la section 1.2. Ce type de modèle répond très bien au premier critère puisqu'il est très simple et qu'il est utilisé avec beaucoup de succès dans les algorithmes de poursuite de cible proposés par Huttenlocher [41]. De plus, en utilisant la distance de Hausdorff, il est possible d'obtenir une mesure très précise pour comparer deux modèles entre eux.

Par contre, la modélisation par silhouette ne permet pas d'estimer facilement l'angle de rotation 2D entre deux modèles. Pour obtenir une représentation qui permettrait ce type de manipulation, il faudrait utiliser une paramétrisation polaire du contour adéquate. Cette paramétrisation pourrait être représentée par un vecteur. Notons-le V_p . La paramétrisation du même modèle, ayant subi une rotation 2D, correspondrait alors à une permutation circulaire du vecteur V_p :

$$V_p' = \sigma^{(k)}(V_p) \quad (\text{EQ 1.1})$$

où σ est la permutation circulaire unité, c'est-à-dire que $\sigma^{(N)} = \text{Identite}$, N est la dimension du vecteur V_p et k une valeur entière comprise entre 0 et $N-1$.

Si on effectue une transformée de Fourier discrète (TFD) sur les vecteurs V_p et V_p' , on constate que les deux transformées ont la même amplitude et ne diffèrent que par la phase. Notons $A_p = |TFD(V_p)|$ le spectre d'amplitude du vecteur V_p et P_p son spectre de phase. Nous obtenons alors:

$$A_p = A_p' \quad (\text{EQ 1.2})$$

et

$$P_p \neq P_p' \text{ avec } P_p'[n] = P_p[n] + \frac{2nk\pi}{N} \quad (\text{EQ 1.3})$$

Une paramétrisation polaire apporte donc une solution logique et relativement simple au problème du calcul de l'angle de la rotation 2D qui peut exister entre deux modèles. Il suffit en effet de retrouver la permutation circulaire qui ramène un vecteur de paramétrisation vers l'autre. Cette opération peut être effectuée soit directement sur les vecteurs de paramétrisation, soit sur leur spectre de phase.

Passons à présent au facteur d'échelle. Deux modèles qui ne diffèrent que par un facteur d'échelle auront des vecteurs de paramétrisation proportionnels. Nous en ferons la démonstration détaillée dans le paragraphe 1.3.2.2.1. Notons f_e le facteur d'échelle, nous obtenons:

$$V_p' = f_e V_p \text{ ainsi que } A_p' = f_e A_p \quad (\text{EQ 1.4})$$

Le spectre de phase des deux vecteurs reste, quant à lui, le même:

$$P_p' = P_p \quad (\text{EQ 1.5})$$

Le facteur d'échelle peut aisément être retrouvé à partir des spectres d'amplitude des deux paramétrisations puisqu'il apparaît sous la forme d'un facteur de proportionnalité.

Les équations 1.2, 1.3, 1.4 et 1.5 nous montrent clairement que, grâce à la transformée de Fourier discrète, il est possible de séparer l'effet d'une rotation 2D de celui d'un changement d'échelle. En effet, les rotations 2D n'ont un effet que sur le spectre de phase du vecteur de paramétrisation alors que le facteur d'échelle ne modifie que le spectre d'amplitude. Ainsi, la modélisation 2D qui semble la mieux adaptée à nos critères devra être basée sur une paramétrisation polaire du contour. Le contour sera représenté comme un modèle de silhouette, c'est-à-dire comme une simple image binaire. Dans la suite de ce chapitre, nous décrirons précisément la paramétrisation qui a été utilisée et nous démontrerons théoriquement et expérimentalement que les objectifs fixés sont bien atteints.

1.3.2 Paramétrisation polaire du modèle silhouette

Dans ce paragraphe, nous expliquerons tout d'abord comment la paramétrisation polaire est calculée à partir de la silhouette. Un exemple nous permettra de mieux visualiser les effets de la paramétrisation. Enfin, la discussion concernant le choix des paramètres de la méthode fera office de conclusion.

1.3.2.1 Algorithme de calcul

Dans la suite, nous supposerons que le contour 2D qui doit être modélisé prend la forme d'une silhouette, c'est-à-dire la forme d'une image binaire pour laquelle les pixels valant un seront considérés comme des points de contour et les pixels valant zéro comme l'arrière-plan. On note S la silhouette, $S_{i,j}$ la valeur du pixel (i, j) (i et j sont entiers) et C l'ensemble des pixels de contour de S . Nous aurons: $C = \{(i, j) | S_{i,j} = 1\}$ et le cardinal de cet ensemble sera noté $|C|$.

1.3.2.1.1 Calcul du centre de gravité

La première étape consiste à calculer le centre de gravité du contour 2D qui sert d'origine au repère du système de coordonnées. La section 1.3.2.2 justifie la pertinence de ce choix.

On note X_g et Y_g les coordonnées du centre de gravité du contour. La formule permettant de les calculer est la suivante:

$$\left\{ \begin{array}{l} X_g = \frac{1}{2} + \sum_{(i,j) \in C} \frac{j}{|C|} \\ Y_g = \frac{1}{2} + \sum_{(i,j) \in C} \frac{i}{|C|} \end{array} \right. \quad (\text{EQ 1.6})$$

Comme i et j font respectivement référence à la ligne et à la colonne du pixel (i, j) , on associe j à X_g et i à Y_g . Les coordonnées du barycentre sont exprimées sous forme réelle plutôt que sous forme entière. Cette conversion risquerait en effet d'introduire des erreurs de discrétisation par la suite. On ajoute 1/2 parce qu'on associe un point de contour au centre du pixel plutôt qu'à son coin supérieur gauche.

1.3.2.1.2 Passage en coordonnées polaires barycentriques

Lorsque les coordonnées du centre du repère polaire sont connues, il suffit de calculer les coordonnées de chaque pixel de contour dans ce repère, avec les formules suivantes:

$$\left\{ \begin{array}{l} r = \sqrt{\left(Y_g - i - \frac{1}{2}\right)^2 + \left(j + \frac{1}{2} - X_g\right)^2} \\ \theta = \operatorname{atan}\left(\frac{Y_g - i - \frac{1}{2}}{j + \frac{1}{2} - X_g}\right) \end{array} \right. \quad (\text{EQ 1.7})$$

1.3.2.1.3 Discrétisation angulaire

Lorsque les coordonnées polaires des points de contour sont calculées, le modèle de silhouette peut être représenté par un ensemble de coordonnées (r, θ) et par la position du centre de gravité (X_g, Y_g) . Cet ensemble de données est relativement volumineux et son utilisation intégrale n'est pas indispensable pour caractériser un contour ou pour le différencier d'un autre. Selon la précision voulue, il est donc souhaitable de diminuer cette quantité d'information. La stratégie adoptée consiste à définir un nombre adéquat de secteurs angulaires et à ne retenir que quelques distances radiales par secteur. Dans un premier temps, nous décrirons comment chaque secteur est défini, puis nous verrons comment calculer les distances radiales.

Les secteurs angulaires ont tous la même taille. Celle-ci est arbitrairement choisie et vaut, par exemple, cinq degrés. Elle sera notée Θ_s à partir de maintenant et sera toujours exprimée en degrés. Les coordonnées polaires précédemment calculées sont alors classées par secteur. Ceci signifie que pour chaque secteur, on génère une liste de distances radiales. Cette opération a pour effet d'éliminer la moitié des données puisque les angles sont supprimés. Cette discrétisation angulaire introduit une distorsion du modèle, mais, en modifiant la taille des secteurs, celle-ci pourra être maîtrisée.

Cette approche algorithmique semble naturelle, mais révèle toutefois des problèmes liés au caractère discret du modèle de silhouette. En effet, faut-il considérer chaque point de contour d'une silhouette comme un point isolé ou comme un point se trouvant sur une courbe? Si on le

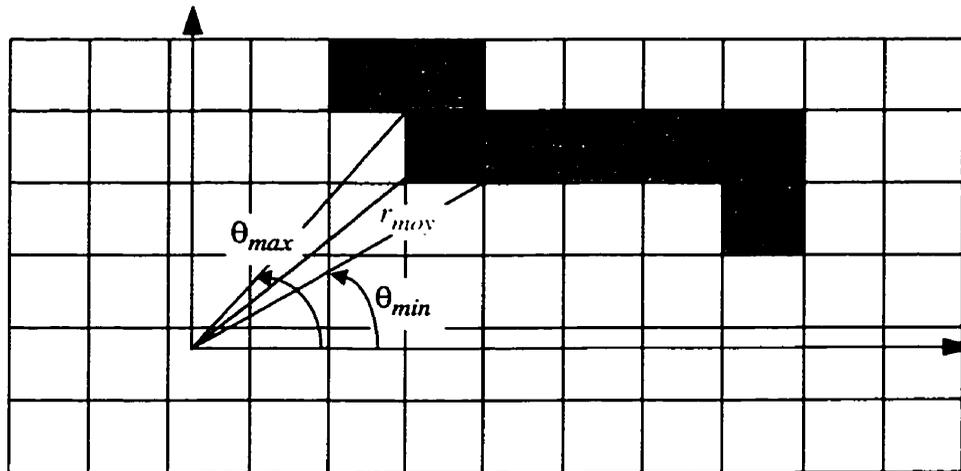


FIGURE 1.1 Discretisation angulaire.

Si on considère que chaque point de contour fait partie d'une courbe, l'angle minimum et l'angle maximum d'un pixel doivent être pris en compte lors de la discrétisation angulaire. La distance radiale notée r_{moy} peut alors être vue comme la distance moyenne de la courbe entre θ_{min} et θ_{max} . r_{moy} correspond à la distance entre le centre de gravité de la courbe et le centre du pixel bleu.

considère comme un point isolé, notre approche est tout à fait valable. Par contre, dans le cas contraire, chaque pixel devrait être représenté non pas par un seul angle, mais plutôt par un intervalle. La figure 1.1 illustre cette situation. Bien entendu, plus les pixels sont éloignés du centre de gravité, plus la variation entre l'angle maximum et l'angle minimum devient négligeable. Il est important de tenir compte de ce phénomène d'une part parce que les modèles 2D représentent des contours d'occultation qui sont continus et d'autre part parce que cette approche limite les variations de la paramétrisation d'un même objet à différentes échelles.

Le calcul de la liste des distances radiales par secteur se fait donc un peu différemment de ce que nous avons décrit jusqu'ici. En fait, pour chaque point de contour de la silhouette, on calcule la distance radiale selon la formule de l'équation 1.7. On évalue ensuite l'angle minimum et l'angle maximum attribué à ce pixel. Le calcul de ces angles dépend de la position du pixel par rapport au centre de gravité. La figure 1.2 montre les 8 zones à considérer. Voici les équations pour le cadran supérieur gauche:

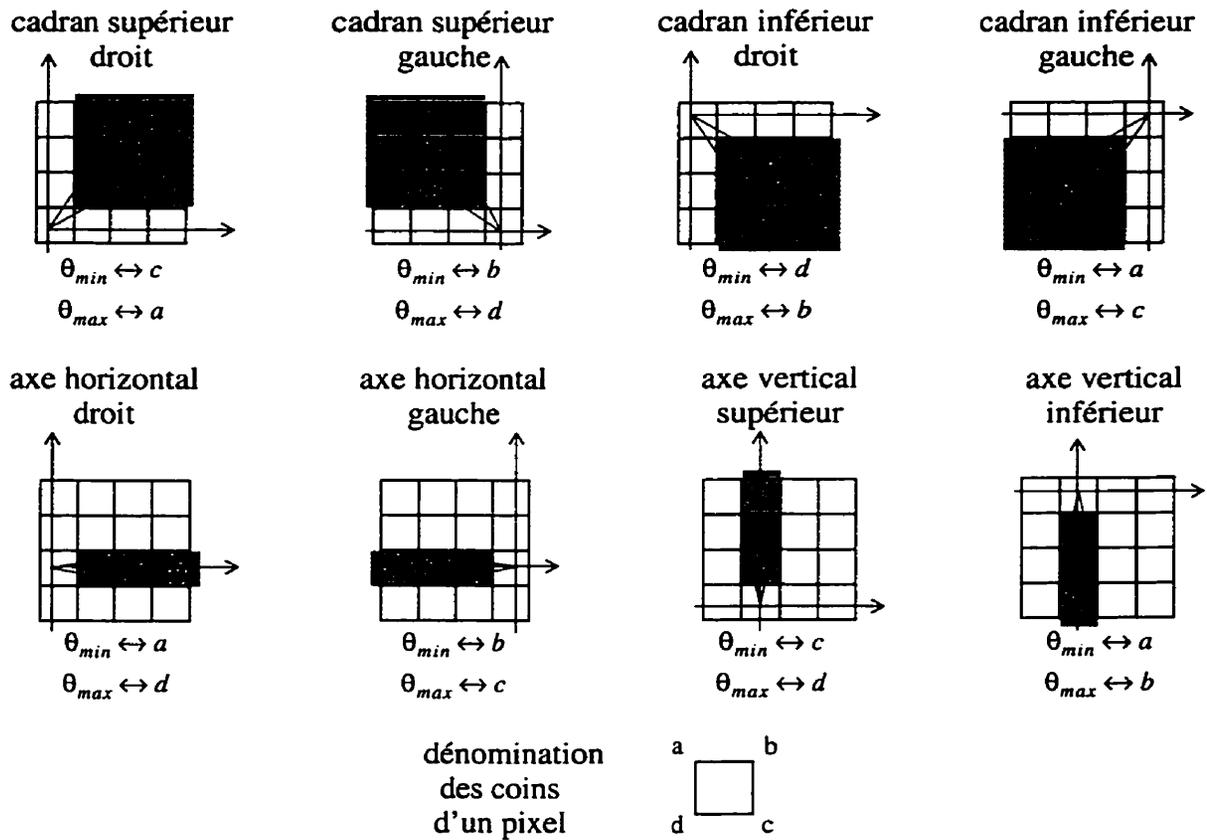


FIGURE 1.2 Définition des zones de calculs pour les angles extrêmes d'un pixel de contour.

$$\left\{ \begin{array}{l} \theta_{min} = \text{atan} \left(\frac{Y_g - i}{j + 1 - X_g} \right) \\ \theta_{max} = \text{atan} \left(\frac{Y_g - i - 1}{j - X_g} \right) \end{array} \right. \quad \text{(EQ 1.8)}$$

Après le calcul de θ_{min} et de θ_{max} , la distance radiale r est ajoutée à la liste de tous les secteurs qui recourent cette région.

1.3.2.1.4 Discrétisation radiale

Après la discrétisation angulaire, nous obtenons une liste de distances radiales pour chaque secteur angulaire. Cet ensemble de données est encore relativement dense et contient beaucoup d'informations redondantes. En effet, dans un secteur angulaire donné, deux pixels voisins auront des distances radiales très proches. À cause de la discrétisation angulaire, celles-ci n'ajouteront pas grand chose au pouvoir discriminant de la paramétrisation. Il est donc possible de limiter le nombre de données sans perdre d'information importante.

Il ne reste qu'à définir une stratégie pour limiter le nombre de valeurs radiales utilisées dans chaque secteur. La méthode retenue est basée sur la nature continue du contour d'occultation. En effet, si on considère que la silhouette n'est qu'une représentation discrète d'une courbe d'occultation, une des manières naturelles de choisir la valeur radiale d'un secteur consiste à prendre le rayon moyen de la partie de la courbe qui intersecte ce secteur. Bien entendu, un contour d'occultation n'est que rarement une fonction de l'angle. Il faut donc prévoir qu'une courbe pourrait intercepter un même secteur plusieurs fois. Dans un tel cas, représenter un secteur angulaire avec une seule valeur n'est pas très pertinent. La solution consiste à utiliser plusieurs valeurs radiales correspondant chacune à une partie de la courbe. La figure 1.3 illustre cette manière de procéder.

Dans la pratique, il faut trier la liste des valeurs radiales obtenues à l'étape de discrétisation pour chaque secteur. Ce tri se fait à l'aide d'un critère d'adjacence. En effet, chaque valeur radiale correspond à un pixel de contour. Il est donc possible de regrouper ces pixels par ensembles adjacents, comme le montre la figure 1.3. Pour chaque groupe, on calcule la valeur radiale moyenne. L'ensemble de ces valeurs moyennes pour un secteur forme alors la paramétrisation finale de celui-ci.

Selon le type de contour d'occultation, un secteur pourra avoir plusieurs valeurs radiales. L'utilisation d'une paramétrisation autorisant un nombre variable de valeurs par secteur poserait un

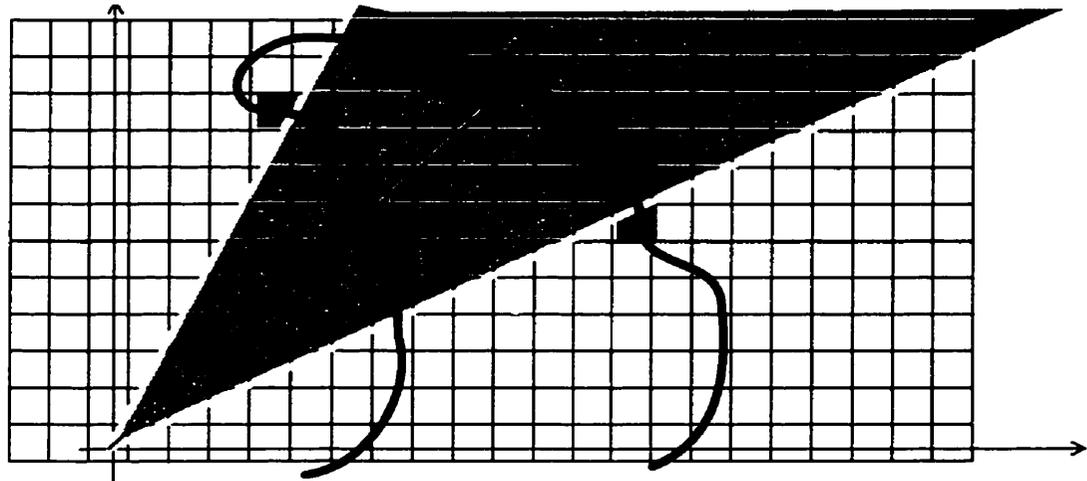


FIGURE 1.3 Discretisation radiale.

Chaque partie de la courbe d'occultation qui traverse un secteur angulaire engendre une valeur radiale. Dans l'exemple ci-dessus, le secteur en vert est donc défini par deux valeurs $r_{moy 1}$ et $r_{moy 2}$.

problème. En effet, il deviendrait difficile de comparer deux modèles entre eux puisque ceux-ci auraient des vecteurs de paramétrisation de dimensions différentes. Si un nombre fixe de valeurs par secteur apparaît comme un choix naturel, son utilisation l'est un peu moins. Voici les différents cas de figure qui peuvent se présenter et voici la manière par laquelle nous avons décidé de les traiter. Nous noterons N_s le nombre fixe de valeurs par secteur et n_s le nombre de valeurs observées dans un secteur en particulier.

1. $n_s = 0$. Il n'y a aucun pixel de contour dans ce secteur et les N_s valeurs sont mises à zéro.
2. $1 \leq n_s \leq N_s$. Les n_s valeurs sont triées dans l'ordre croissant pour être utilisées dans cet ordre. Comme n_s peut être inférieur à N_s , les valeurs manquantes sont toutes fixées à la plus grande des valeurs possibles. Ce choix est arbitraire. Les autres options possibles auraient été de dédoubler une autre valeur que la plus grande ou d'en dédoubler plusieurs pour compléter le vecteur jusqu'au N_s valeurs nécessaires. Toutes ces approches auraient été aussi arbitraire l'une que l'autre. Ce qui est évident, par contre, c'est qu'il ne faut pas compléter le vecteur par des valeurs nulles. Nous aurions à ce moment un problème lors de la comparaison entre deux contours.

3. $n_s > N_s$. Il y a trop de valeurs et dans ce cas ce sont les parties de courbes les moins significatives qui sont supprimées. Ainsi, on ne garde que les N_s valeurs correspondant aux regroupements de pixels les plus importants. Les valeurs sont triées par ordre croissant.

Cette dernière étape nous permet d'aboutir à une paramétrisation du modèle de silhouette sous la forme d'un vecteur de dimension $360N_s/\Theta_s$. Nous noterons ce vecteur de la manière suivante: V_p .

1.3.2.1.5 Transformée de Fourier discrète

Le vecteur de paramétrisation V_p n'est malheureusement pas invariant aux rotations. En utilisant la transformée de Fourier discrète, il est cependant possible de le rendre invariant à toutes les rotations dont l'angle est proportionnel à Θ_s . En effet, toute rotation d'angle $k\Theta_s$ avec $k \in \mathbb{N}$ engendrera une permutation circulaire sur le vecteur V_p . Or, nous avons déjà vu que deux vecteurs égaux - à une permutation circulaire près - ont des transformations de Fourier discrètes égales à un facteur de phase près (voir section 1.3.1).

Ainsi, le vecteur associé au modèle de silhouette sera le spectre d'amplitude du vecteur V_p . Ce spectre est un vecteur de dimension $(360N_s/2\Theta_s + 1)$. Le vecteur V_p étant réel, le spectre d'amplitude est symétrique et il est donc possible de ne conserver que la moitié des données sans perdre d'information. Nous noterons le spectre d'amplitude A_p .

Le vecteur représentant le spectre de phase aura la même dimension que A_p . En effet, lorsque V_p est réel, son spectre de phase est impair et la même remarque que pour le spectre d'amplitude s'applique. Nous noterons P_p le spectre de phase de V_p .

1.3.2.1.6 Résumé de l'algorithme de calcul de la paramétrisation

Voici la liste des différentes étapes du calcul de la paramétrisation:

1. Calcul des coordonnées barycentriques;
2. Calcul des coordonnées polaires (avec angle minimum et maximum);

3. Répartition par secteur angulaire;
4. Regroupement des pixels adjacents dans chaque secteur angulaire et calcul des distances radiales moyennes;
5. Classement des distances radiales pour chaque secteur et élimination des valeurs superflues afin d'obtenir le vecteur V_p ;
6. Calcul de la transformée de Fourier discrète, puis du spectre d'amplitude A_p et du spectre de phase P_p .

Avec cette paramétrisation, nous pourrions remplacer le modèle 2D par trois vecteurs réels: le vecteur de paramétrisation, son spectre d'amplitude et son spectre de phase.

1.3.2.2 Vérifications théoriques des propriétés

Nous avons construit une paramétrisation de notre modèle de silhouette. Il faut maintenant démontrer que cette paramétrisation possède bien les propriétés recherchées par rapport à un changement d'échelle ou à une rotation 2D.

1.3.2.2.1 Changement d'échelle

Dans la section 1.3.1, nous avons supposé que l'effet d'un changement d'échelle du modèle multipliait le vecteur de paramétrisation V_p par un facteur d'échelle $f_e \in \mathbb{R}^+$. Voyons si, dans la théorie, la paramétrisation que nous avons construite vérifie bien cette propriété.

Supposons que le modèle de silhouette S^0 possède une paramétrisation V_p^0 tel que montré à la figure 1.4. Supposons ensuite que le modèle S^1 correspond au même modèle, mais à un facteur d'échelle près (voir figure 1.4). Notons (x^0, y^0) les coordonnées d'un point de contour de S^0 dans le repère ayant pour origine le point de focale de la caméra. Ce point de contour aura pour coordonnées: $(x^1, y^1) = (f_e x^0, f_e y^0)$ dans le modèle S^1 (f_e étant le facteur d'échelle). Les coordonnées du barycentre de S^1 seront donc:

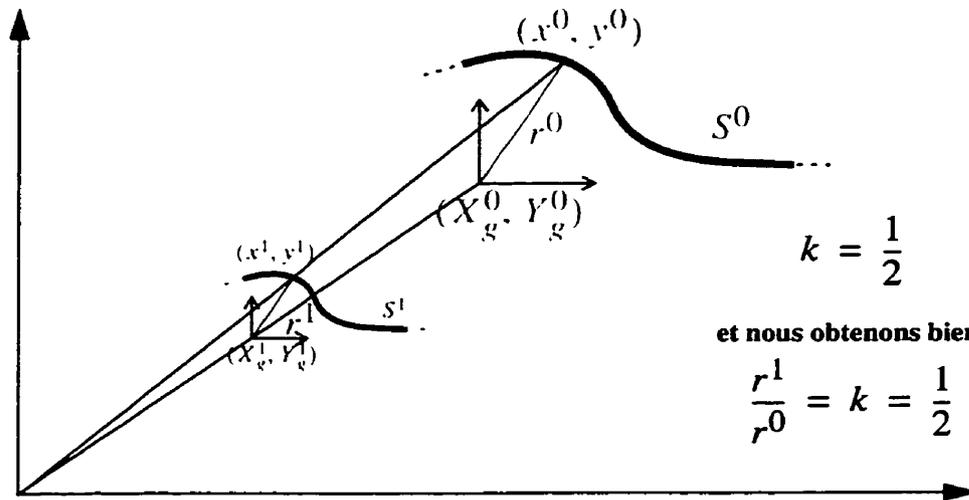


FIGURE 1.4 Vérification de la proportionnalité par rapport au facteur d'échelle.

$$\begin{cases} X_g^1 = f_e\left(X_g^0 - \frac{1}{2}\right) + \frac{1}{2} \\ Y_g^1 = f_e\left(Y_g^0 - \frac{1}{2}\right) + \frac{1}{2} \end{cases} \quad \text{(EQ 1.9)}$$

où (X_g^0, Y_g^0) sont les coordonnées du barycentre de S^0 . Le calcul de la coordonnée polaire radiale d'un point de contour de S^1 sera alors:

$$r^1 = \sqrt{\left(Y_g^1 - y^1 - \frac{1}{2}\right)^2 + \left(x^1 + \frac{1}{2} - X_g^1\right)^2} \quad \text{(EQ 1.10)}$$

En utilisant les équations précédentes, on peut exprimer la coordonnée radiale en fonction du repère d'origine:

$$r^1 = \sqrt{\left(\left(f_e\left(Y_g^0 - \frac{1}{2}\right) + \frac{1}{2}\right) - f_e y^0 - \frac{1}{2}\right)^2 + \left(f_e x^0 + \frac{1}{2} - \left(f_e\left(X_g^0 - \frac{1}{2}\right) + \frac{1}{2}\right)\right)^2} \quad (\text{EQ 1.11})$$

Et ainsi nous obtenons:

$$r^1 = \sqrt{f_e^2\left(Y_g^0 - \frac{1}{2} - y^0\right)^2 + f_e^2\left(x^0 - X_g^0 + \frac{1}{2}\right)^2} = f_e \cdot r^0 \quad (\text{EQ 1.12})$$

De la même manière, on montre que le calcul des angles donne le même résultat pour les deux modèles. Les vecteurs de paramétrisation avant la transformée de Fourier sont donc proportionnels. De plus, comme la transformée de Fourier discrète est une opération linéaire, les deux spectres d'amplitude restent proportionnels eux aussi. Ainsi nous obtenons:

$$A_p^0 = f_e \cdot A_p^1 \quad (\text{EQ 1.13})$$

La démonstration précédente ne tient pas compte de la discrétisation liée au modèle de silhouette. En effet, dans une silhouette, la résolution est limitée à la taille des pixels. Nous verrons l'ampleur des distorsions engendrées par cette discrétisation dans la section 1.4.1.2.

1.3.2.2.2 Rotations

Comme nous l'avons dit dans l'introduction, les modèles de silhouette que nous traitons correspondent tous à des contours d'occultation obtenus à partir de la projection d'objets 3D. Les rotations que nous allons considérer dans cette section correspondent à celles de l'objet 3D par rapport à un axe parallèle à celui de la caméra et centré à l'origine du repère de l'objet. Pour le modèle en silhouette, il s'agit donc d'une rotation 2D dont le centre est la projection de l'origine du repère de l'objet (voir figure 1.5).

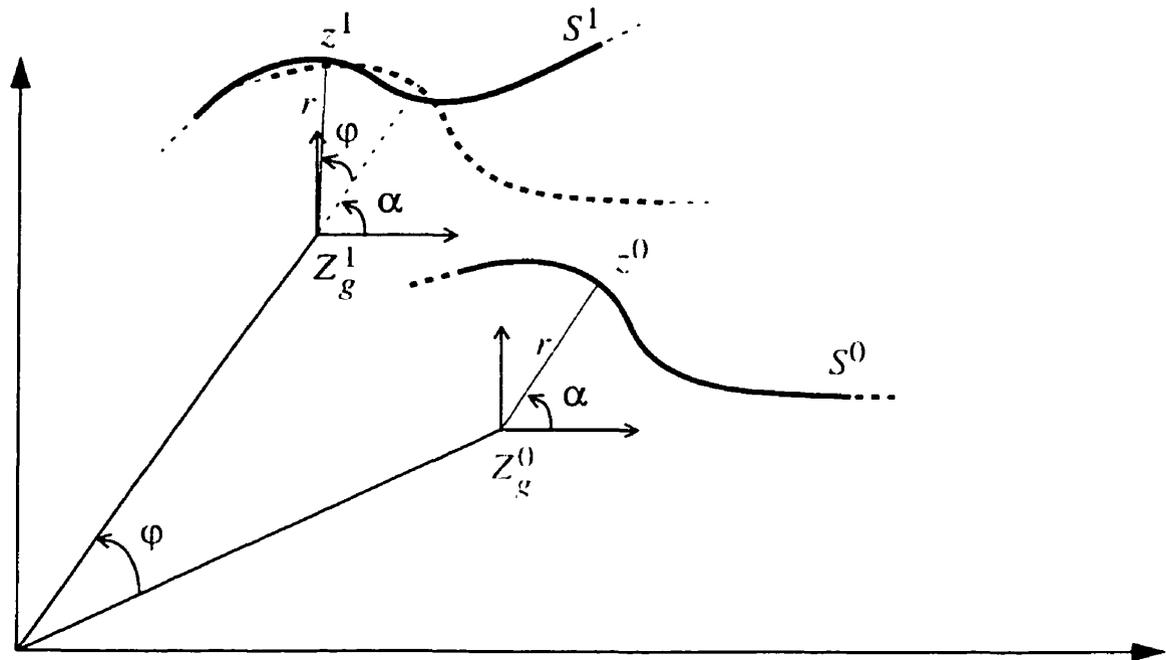


FIGURE 1.5 Vérification de l'invariance par rapport aux rotations.

Dans la section 1.3.1, nous avons supposé que l'effet d'une telle rotation sur la paramétrisation était une permutation circulaire du vecteur de paramétrisation. Pour démontrer que la paramétrisation construite vérifie bien cette propriété, il faut prouver que cette rotation a le même effet qu'une rotation de même angle centré à l'origine du repère barycentrique.

Notons z^0 la coordonnée complexe d'un point quelconque de la silhouette S^0 dans le repère dont l'origine est la projection du centre du repère 3D lié à l'objet. S^1 correspondra à la silhouette de l'objet après une rotation d'angle φ et z^1 sera l'image de z^0 par cette rotation. On note Z_g^0 la coordonnée complexe du centre de gravité de S^0 et Z_g^1 celle de S^1 . Par définition, nous aurons:

$$z^1 = e^{j\varphi} \cdot z^0 \quad (\text{EQ 1.14})$$

Le centre de gravité de S^1 sera l'image par la rotation d'angle φ de celui de S^0 . La relation suivante sera donc vérifiée:

$$Z_g^1 = e^{j\varphi} \cdot Z_g^0 \quad (\text{EQ 1.15})$$

Il existe $r \in \mathbb{R}^*$ et $\alpha \in \mathbb{R}$ tels que

$$z^0 - Z_g^0 = r e^{j\alpha}. \quad (\text{EQ 1.16})$$

(r, α) correspondent aux coordonnées polaires de z^0 dans le repère barycentrique. Calculons maintenant les coordonnées polaires de z^1 dans le repère barycentrique.

$$\begin{aligned} z^1 - Z_g^1 &= e^{j\varphi} \cdot z^0 - e^{j\varphi} \cdot Z_g^0 \\ &= e^{j\varphi} \cdot (z^0 - Z_g^0) \\ &= e^{j\varphi} \cdot (r e^{j\alpha}) = r e^{j(\alpha + \varphi)} \end{aligned} \quad (\text{EQ 1.17})$$

On constate qu'une rotation de l'objet 3D par rapport à un axe perpendiculaire au plan image se traduit par une rotation du même angle dans le repère barycentrique. La figure 1.5 illustre la démonstration.

Si l'angle de rotation est proportionnel à Θ_s , le vecteur de paramétrisation V_p subira simplement une permutation circulaire. L'invariance du spectre d'amplitude par rapport à ces rotations est alors assurée. Par contre, une rotation quelconque introduit des erreurs puisque les secteurs angulaires ne sont plus définis exactement de la même manière par rapport à l'objet. Nous verrons dans la section 1.4.1.3 quelle est l'ampleur de ces erreurs. Il est toutefois assez naturel de penser que plus l'angle Θ_s sera petit, moins ces erreurs seront importantes.

Dans ce paragraphe, tout comme dans le précédent, nous n'avons pas tenu compte de la discrétisation liée au modèle de silhouette. La position des points de contours n'étant connue qu'à une valeur discrète près, une rotation introduit nécessairement des erreurs d'arrondi. La section 1.4.1.3 étudie l'effet de cette discrétisation.

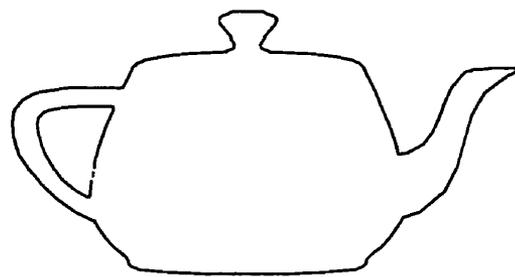
1.3.2.3 Exemple

La figure 1.6 donne un exemple de la paramétrisation obtenue à partir d'un modèle de silhouette représentant une théière. On peut voir l'importance du facteur de discrétisation angulaire. Celui-ci permet en effet de gérer le nombre de points qui représenteront le contour. La valeur du paramètre choisi dans l'exemple ci-dessous est bonne puisqu'elle permet de reconnaître l'objet qualitativement lorsqu'on ne montre que les points de la paramétrisation. Choisir un paramètre de discrétisation trop grand limite sévèrement la qualité de la paramétrisation. Mais le prendre trop petit augmente énormément la dimension des vecteurs de paramétrisation.

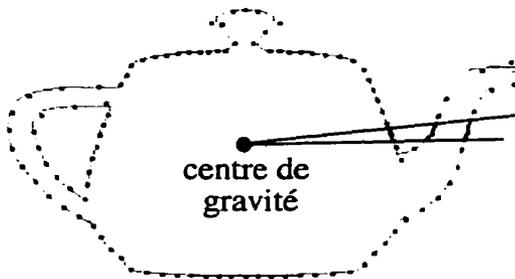
Un aspect important qu'il faut noter concernant la paramétrisation que nous proposons est la non-homogénéité de l'échantillonnage des points le long du contour d'occultation. Comme on peut le constater dans la figure 1.6, toutes les parties de la théière ne sont pas représentées avec la même précision. Le haut du bec verseur est un exemple particulièrement parlant. Il s'agit ici d'un défaut intrinsèque de la paramétrisation que nous proposons. En effet, si on observe la figure 1.7, on peut constater que des courbes qui s'éloignent du centre de gravité en restant dans le même secteur angulaire ne seront représentées que par un seul point. Ce défaut peut être réduit en jouant sur la valeur du facteur de discrétisation angulaire, mais il ne peut pas être complètement éliminé.

L'intérêt du facteur de discrétisation radiale apparaît clairement lorsqu'on observe un secteur angulaire spécifique. La figure 1.6 montre en effet un secteur qui est traversé trois fois par le contour d'occultation. Il est donc important de conserver ce paramètre à une valeur supérieure à un même si ceci engendre une grande redondance dans les composantes du vecteur de paramétrisation.

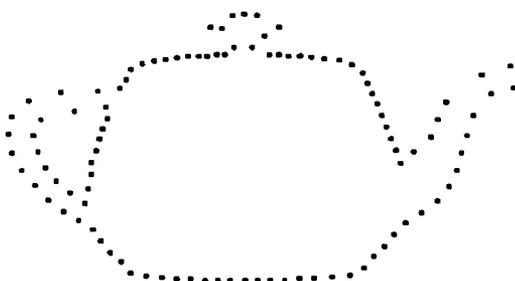
Enfin, la visualisation du vecteur de paramétrisation V_p donne une assez bonne idée de l'aspect du contour. Les variations très rapides de la courbe de paramétrisation proviennent des différentes valeurs radiales associées à chaque secteur angulaire. Ces pics peuvent laisser crain-



Modèle de silhouette d'une théière



Modèle de silhouette et, en sur-impression, le résultat de la paramétrisation (points bleus). La zone rouge représente un des secteurs de la modélisation



Résultat de la paramétrisation

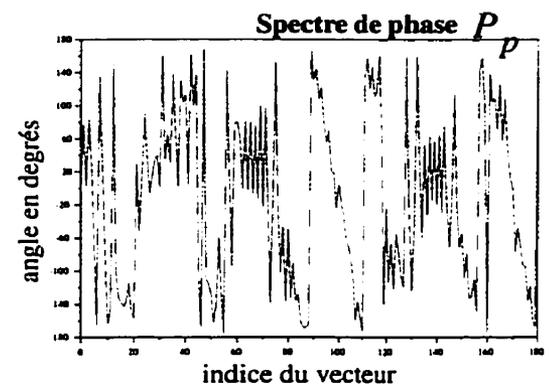
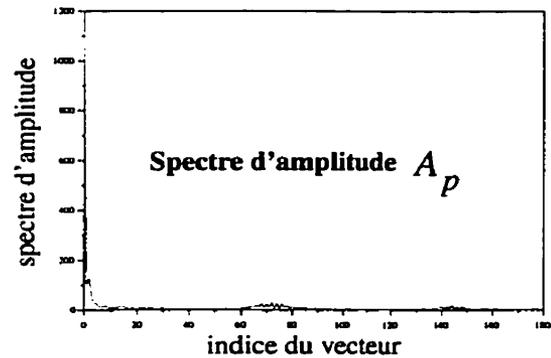
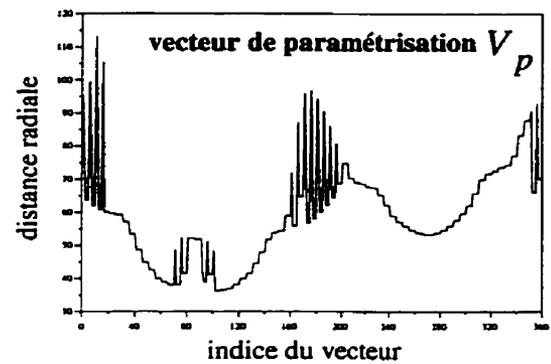


FIGURE 1.6 Exemple de paramétrisation d'un modèle de silhouette.

Cette figure illustre le principe de paramétrisation. En haut à gauche se trouve le modèle de silhouette d'une théière. Au milieu, ce modèle est repris avec, en sur-impression, la position de chaque point du vecteur de paramétrisation V_p . La zone en rouge sur le contour de la théière représente le secteur angulaire $[0^\circ, 5^\circ]$. La figure en bas à gauche reprend la position des différentes composantes du vecteur de paramétrisation V_p . Le graphique du haut représente le vecteur de paramétrisation V_p . Les deux graphiques du bas représentent respectivement son spectre d'amplitude A_p et son spectre de phase P_p . Les valeurs utilisées dans cet exemple sont: $\Theta_s = 5^\circ$ et $N_s = 5$. La dimension de V_p est donc 360 et celles de A_p et de P_p 181.

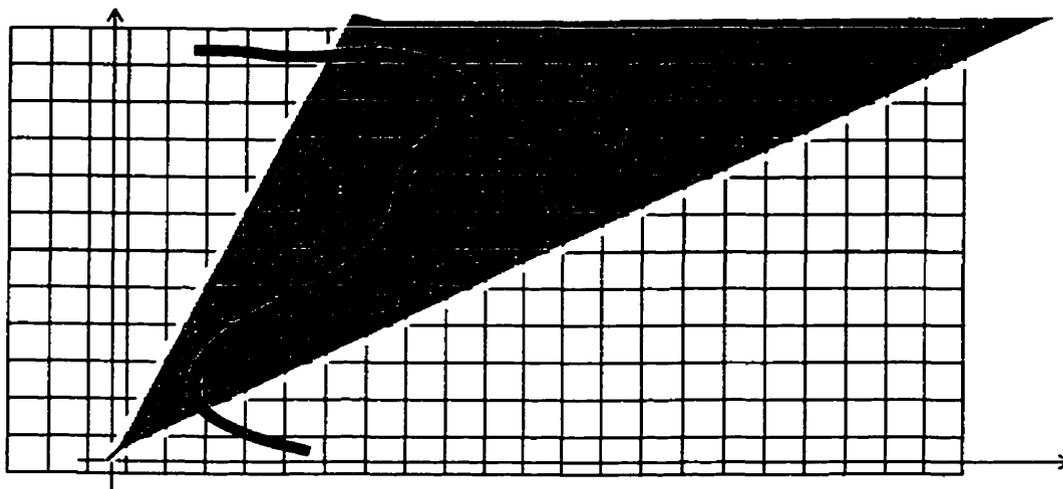


FIGURE 1.7 Exemple d'échantillonnage insuffisant sur une courbe particulière.

Dans cette figure, on peut constater que l'ensemble de la courbe bleue qui se trouve dans le secteur angulaire vert n'est représenté que par un seul point. Il s'agit là d'une faiblesse de notre paramétrisation.

dre une grande sensibilité de la paramétrisation que nous avons choisie. En fait, il n'en est rien. Ces nombreuses variations correspondent uniquement au choix que nous avons fait pour représenter le vecteur de paramétrisation. La représentation aurait en effet été beaucoup moins "saccadée" si nous avions choisi de tracer le vecteur de paramétrisation en commençant par la première coordonnée radiale pour tous les secteurs angulaires puis par la seconde et ainsi de suite. Il s'agit donc simplement d'un artefact de représentation et non d'une limitation de notre approche.

1.3.2.4 Discussion sur les paramètres de la modélisation

Dans la description que nous avons faite de la modélisation, plusieurs paramètres interviennent: le centre de gravité du modèle de silhouette, le facteur de discrétisation angulaire Θ_s et le facteur de discrétisation radiale N_s . Chacun de ces facteurs a une influence directe sur la qualité de la modélisation.

1.3.2.4.1 Centre de gravité

Le calcul du centre de gravité est la première étape de la paramétrisation. Ce paramètre est crucial puisqu'une erreur de quelques pixels aura pour effet de fausser l'ensemble des composantes du vecteur V_p . Si le contour d'occultation est bruité uniformément (c'est-à-dire que la position des points de contour est bruitée de la même manière tout au long du contour), les effets du bruit pourront se compenser et l'erreur sur la position du centre de gravité restera minimale. Lors de l'utilisation de cette méthode de paramétrisation, il faudra donc tenir compte de cette sensibilité par rapport à la position du centre de gravité.

Comme la précision de la position du centre de gravité est vitale, il va sans dire que toute occlusion d'une partie de l'objet est fatale. Ceci explique l'une des hypothèses faites dans l'introduction.

1.3.2.4.2 Facteur de discrétisation angulaire

Le facteur de discrétisation angulaire intervient dans deux aspects de la modélisation. Tout d'abord, il permet de gérer l'échantillonnage angulaire du signal. Il intervient aussi dans l'invariance du modèle obtenu par rapport aux rotations.

Comme nous l'avons déjà dit, l'utilisation d'un angle de discrétisation trop grand limite la qualité de la paramétrisation. Par contre, un angle trop petit engendre un vecteur de paramétrisation de très grande dimension. Il faut donc trouver un compromis entre la qualité de la modélisation et la taille des informations à traiter.

1.3.2.4.3 Facteur de discrétisation radiale

Le facteur de discrétisation radiale est le paramètre dont l'impact sur la qualité de la discrétisation est le moins important. En effet, ce paramètre représente le nombre de fois qu'une courbe peut traverser un même secteur angulaire. En fixant cette valeur à 5 composantes, on s'assure que la plupart des objets classiques pourront être convenablement modélisés, sans pour autant augmenter de manière trop importante la quantité d'informations à gérer. D'autres

valeurs peuvent bien sûr être utilisées, en particulier si on cherche à diminuer le nombre de composantes du vecteur de paramétrisation ou lorsqu'on travaille avec des objets particuliers.

1.3.3 Mesure de comparaison

1.3.3.1 Choix de la mesure

Pour comparer deux contours d'occultation, il existe dans la littérature différentes mesures ayant fait leur preuve. Parmi celles-ci, la distance de Hausdorff et celle de chamfer sont deux mesures très fiables et ont déjà été largement utilisées [33,41]. Ces distances sont relativement simples à évaluer, mais elles nécessitent le calcul préalable d'une surface de Voronoi et ne permettent pas de tenir compte d'un changement d'échelle ou d'une rotation existant entre deux modèles.

La distance de Hausdorff mesure la distance entre deux contours P et Q de la manière suivante:

$$d_H(P, Q) = \max(d_h(P, Q), d_h(Q, P)) \quad (\text{EQ 1.18})$$

avec:

$$d_h(P, Q) = \max_{p \in P} \min_{q \in Q} \|p - q\| \quad (\text{EQ 1.19})$$

L'interprétation de ces équations est simple. La distance de Hausdorff représente la distance maximale entre un point de P (respectivement de Q) et le point le plus proche de Q (respectivement de P). Cette distance permet donc de mesurer l'écart maximum entre deux contours.

La distance de chamfer entre deux contours P et Q est donnée par les équations suivantes:

$$d_C(P, Q) = \max(d_c(P, Q), d_c(Q, P)) \quad (\text{EQ 1.20})$$

avec

$$d_c(P, Q) = \frac{1}{\text{card}(P)} \sum_{p \in P} \min_{q \in Q} \|p - q\| \quad (\text{EQ 1.21})$$

Contrairement à la distance de Hausdorff qui mesure l'écart entre deux contours en un seul point, cette mesure évalue la distance moyenne entre les deux contours.

À l'aide de la paramétrisation présentée, nous allons introduire une nouvelle mesure proche de celle de Hausdorff et de chamfer. Toutes les distances liées aux espaces vectoriels normés pourraient être utilisées sur l'un des trois vecteurs formant la paramétrisation (vecteur de paramétrisation, spectre d'amplitude et spectre de phase). Il s'agit toutefois de choisir une distance adéquate afin d'obtenir une mesure qui réponde bien aux critères décrits dans l'introduction de ce chapitre. Il faut en effet que la mesure permette de comparer deux contours sans tenir compte ni du facteur d'échelle ni de l'angle de rotation qui pourraient exister entre deux modèles.

L'utilisation du vecteur de paramétrisation V_p pour le calcul de cette mesure de comparaison ne serait pas une bonne méthode. En effet, un même modèle pourra avoir des vecteurs de paramétrisation très différents en fonction de son angle de rotation et de son facteur d'échelle. Une mesure basée uniquement sur le spectre d'amplitude A_p permettrait d'évaluer efficacement la ressemblance de deux modèles malgré un facteur de rotation. Malheureusement, deux modèles pour lesquels le spectre d'amplitude est identique, mais dont le spectre de phase diffère ne sont pas nécessairement semblables comme le montre la figure 1.8. De plus, un changement d'échelle modifie le spectre d'amplitude.

Afin de définir une mesure vraiment efficace, la meilleure approche est donc de ramener un modèle vers l'autre en effectuant une rotation 2D et un changement d'échelle. L'angle de rotation peut être obtenu à partir des deux spectres de phase et le facteur d'échelle à partir des spectres d'amplitude, comme nous le verrons dans les sections 1.3.3.2.1 et 1.3.3.2.2. Notons C_i et C_j les deux contours que l'on cherche à comparer et $\delta(C_i, C_j)$ la mesure de comparaison entre ces deux modèles. Supposons que les vecteurs de paramétrisation de ces deux contours soient respectivement V_p^i et V_p^j , que le facteur d'échelle permettant de ramener le contour C_j à l'échelle de C_i soit $f_e^{i/j}$ et que la permutation circulaire correspondant à l'angle de rota-

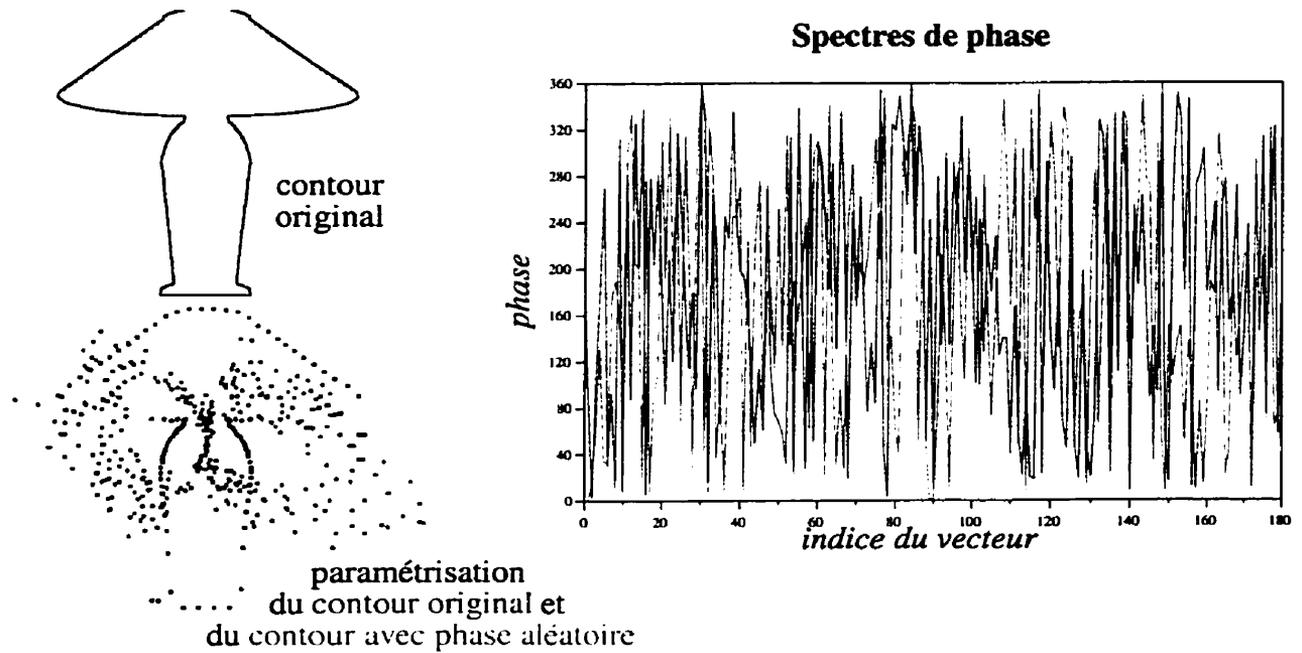


FIGURE 1.8 Modèles ayant le même spectre d'amplitude, mais un spectre de phase distinct.

Le spectre d'un contour original (en haut à gauche) a été calculé. Afin de voir l'effet de la phase sur la paramétrisation, nous avons utilisé le spectre d'amplitude de ce contour et un spectre de phase aléatoire pour reconstruire le vecteur de paramétrisation. On constate aisément que le spectre de phase a une importance capitale pour définir la forme d'un contour.

tion entre les deux contours vaille n . La mesure de comparaison peut alors s'exprimer à l'aide de l'équation suivante:

$$\delta(C_i, C_j) = \frac{1}{N} \left\| V_p^i - f_e^{i/j} \sigma^{(n)}(V_p^j) \right\|_{L_p} \quad (\text{EQ 1.22})$$

Rappelons que $\sigma^{(n)}(\)$ représente la permutation circulaire qui déplace n composantes et que n doit être une valeur entière comprise entre 0 et $N-1$, sachant que N est la dimension des vecteurs de paramétrisation. $\| \cdot \|_{L_p}$ représente la norme L_p définie sur un espace vectoriel et a pour expression:

$$\|x\|_{L_p} = \left(\sum_k |x_k|^p \right)^{1/p} \quad (\text{EQ 1.23})$$

N'importe quelle norme L_p peut être utilisée, mais nous étudierons plus en détail les mesures correspondant aux normes L_1 ¹ et L_{inf} ² parce qu'elles permettent d'effectuer des comparaisons avec les distances de chamfer et de Hausdorff respectivement. Dans la suite de ce chapitre, nous noterons δ_{L_1} et $\delta_{L_{inf}}$ les deux mesures que nous étudierons.

La mesure que nous venons de définir n'est malheureusement pas symétrique. En effet:

$$\delta(C_j, C_i) = \frac{\delta(C_i, C_j)}{f_{i/j}} \quad (\text{EQ 1.24})$$

Afin d'éliminer cette asymétrie, nous modifions la définition de la mesure comme suit:

$$\delta_{sym}(C_j, C_i) = \text{Min}(\delta(C_i, C_j), \delta(C_j, C_i)) \quad (\text{EQ 1.25})$$

Nous noterons δ_{symL_1} et $\delta_{symL_{inf}}$ les deux mesures correspondantes. Pour savoir si cette définition représente une bonne mesure pour le contour d'occultation, nous la comparerons aux distances de Hausdorff et de chamfer. Cette comparaison est présentée en détails à la section 1.4.1.1.

1.3.3.2 Algorithme de calcul de la mesure de comparaison

Nous avons introduit une mesure de comparaison qui permet d'éliminer les effets d'un changement d'échelle ou d'une rotation. Cette mesure est définie à partir de la paramétrisation que nous avons construite. Elle nécessite le calcul préalable du facteur d'échelle et de l'angle de rotation 2D. Dans cette section, nous allons décrire la méthodologie d'estimation de ces deux paramètres.

1. La norme L_1 est définie de la manière suivante: $\|x\|_{L_1} = \sum_k |x_k|$.
2. Normalement, la norme L_{inf} est définie de la manière suivante: $\|x\|_{L_{inf}} = \max_k |x_k|$, mais, afin de conserver le même ordre de grandeur que pour la norme L_1 , nous introduirons la dimension du vecteur dans la définition: $\|x\|_{L_{inf}} = N \cdot \max_k |x_k|$

1.3.3.2.1 Calcul du facteur d'échelle

Soient A_p^i et A_p^j les spectres d'amplitudes des paramétrisations respectives de deux modèles i et j . Si les deux modèles correspondent au même objet à un facteur d'échelle près, alors il existe un facteur de proportionnalité entre ces deux vecteurs (comme nous l'avons montré dans la section 1.3.1) et ce même s'il existe une rotation entre les deux modèles.

Pour retrouver le facteur d'échelle f_e entre les deux modèles, il suffit de trouver le minimum de la fonction suivante:

$$\Psi(f_e) = \|A_p^i - f_e \cdot A_p^j\|_{L_2} \quad (\text{EQ 1.26})$$

L'utilisation de la norme euclidienne L_2 se justifie aisément puisqu'elle permet de trouver de manière analytique le minimum de la fonction $\Psi(f_e)$ en cherchant le passage par zéro de sa dérivée première. Si on note $f_e^{i/j}$ le facteur d'échelle permettant de ramener le modèle j à l'échelle du modèle i , on aura alors:

$$f_e^{i/j} = \frac{\langle A_p^i | A_p^j \rangle}{\|A_p^j\|_{L_2}^2} \text{ où } \langle | \rangle \text{ représente le produit scalaire de deux vecteurs.} \quad (\text{EQ 1.27})$$

Le facteur d'échelle peut ainsi être obtenu pour n'importe quel couple de modèles définis par leur paramétrisation. Il ne représente cependant un véritable facteur d'échelle que si les deux modèles correspondent effectivement au même objet. Il est donc très important de calculer la mesure de comparaison avant de passer aux conclusions. Or, ce calcul nécessite la connaissance d'un autre paramètre: l'angle de rotation. Nous allons maintenant voir comment le calculer.

1.3.3.2.2 Calcul de l'angle de rotation

Soient V_p^i et V_p^j les vecteurs de paramétrisation de deux modèles. Pour savoir s'il existe une rotation entre eux-ci, il faut retrouver l'angle qui permet de passer de l'un à l'autre. Avec la paramétrisation que nous utilisons, cet angle prend deux formes. Il se retrouve sous la forme

d'une permutation circulaire des composantes de V_p^i par rapport à V_p^j avec un éventuel facteur d'échelle. Mais il apparaît également comme un déphasage entre le spectre de phase des deux paramétrisations. Suivant le point de vue que l'on adopte il est possible d'utiliser deux algorithmes différents pour calculer l'angle de rotation. Nous allons maintenant décrire ces deux algorithmes.

1.3.3.2.1 Algorithme de permutation circulaire

En utilisant l'approche basée sur la permutation circulaire des composantes, retrouver l'angle de rotation $\varphi^{i/j}$ permettant de ramener le modèle j sur le modèle i , revient à calculer la permutation qui minimise la distance entre le vecteur V_p^j et le vecteur $f_e^{i/j} \cdot V_p^j$. Il faut en effet tenir compte du facteur d'échelle qui peut exister entre les deux modèles. L'expression que l'on cherche à minimiser est par conséquent:

$$n_r = \underset{n \in \llbracket 0, N-1 \rrbracket}{\operatorname{argmin}} \left\| V_p^i - f_e^{i/j} \sigma^{(n)}(V_p^j) \right\|_{L_2}^2 \quad (\text{EQ 1.28})$$

où $\sigma^{(n)}(\)$ représente la permutation circulaire qui déplace n composantes et N est la dimension des vecteurs de paramétrisation $N = 360N_s/\Theta_s$. Toutes les valeurs de n_r ne sont pas permises à cause du facteur de discrétisation radiale. En effet, pour qu'une permutation corresponde effectivement à une rotation, il faut que n_r soit proportionnel à N_s . Nous aurons donc:

$$n_r = \underset{n \in \llbracket 0, 360/\Theta_s \rrbracket}{\operatorname{argmin}} \left\| V_p^i - f_e^{i/j} \sigma^{(nN_s)}(V_p^j) \right\|_{L_2}^2 \quad (\text{EQ 1.29})$$

L'angle de rotation est alors:

$$\varphi^{i/j} = n_r \Theta_s \quad (\text{EQ 1.30})$$

La résolution directe de cette équation possède une complexité algorithmique élevée puisqu'il faut effectuer $360/\Theta_s$ calculs de distance euclidienne sur un vecteur de taille N . De plus il

faut avoir préalablement calculé le facteur d'échelle, ce qui peut introduire une erreur supplémentaire. Il est toutefois possible de réduire la complexité de l'algorithme en développant l'expression à minimiser. L'équation 1.29 peut en effet s'écrire:

$$\begin{aligned}
n_r &= \underset{n \in \llbracket 0, 360/\Theta_s \rrbracket}{\operatorname{argmin}} \left(\|V_p^i\|_{L_2}^2 + (f_e^{i/j})^2 \|V_p^j\|_{L_2}^2 - 2f_e^{i/j} \langle V_p^i | \sigma^{(nN_s)}(V_p^j) \rangle \right) \\
&= \underset{n \in \llbracket 0, 360/\Theta_s \rrbracket}{\operatorname{argmax}} \left(\langle V_p^i | \sigma^{(nN_s)}(V_p^j) \rangle \right) \\
&= \operatorname{round} \left(\frac{\underset{n \in \llbracket 0, N \rrbracket}{\operatorname{argmax}} (V_p^i \bullet V_p^j)(n)}{N_s} \right)
\end{aligned} \tag{EQ 1.31}$$

$(x \bullet y)$ correspond au produit de convolution circulaire et $\operatorname{round}(x)$ correspond à l'approximation de la valeur entière la plus proche de x . Les opérations que nous avons effectuées nous ramènent au calcul d'un produit de convolution circulaire. Or, ce produit de convolution circulaire peut être calculé à l'aide de la transformée de Fourier des deux vecteurs de paramétrisation. Nous noterons W_p^i et W_p^j les transformées de Fourier respectives des deux vecteurs de paramétrisation. Nous aurons alors:

$$n_r = \operatorname{round} \left(\frac{\underset{n \in \llbracket 0, N \rrbracket}{\operatorname{argmax}} \operatorname{TFI}(W_p^i \otimes \overline{W_p^j})(n)}{N_s} \right) \tag{EQ 1.32}$$

$x \otimes y$ correspond au produit composante par composante des deux vecteurs x et y . Il s'agit donc d'un vecteur de même dimension que x et y . $\operatorname{TFI}(W)$ représente la transformée de Fourier inverse du vecteur W et enfin \overline{W} représente le vecteur conjugué de W . Comme les vecteurs W_p^i et W_p^j doivent toujours être calculés lors de la paramétrisation d'un contour (afin d'accé-

der au spectre d'amplitude et de phase), l'évaluation de l'équation 1.32 peut se faire à l'aide d'un algorithme dont la complexité est raisonnable.

Cette approche pour le calcul de l'angle de rotation est très intéressante puisqu'elle permet de trouver l'angle qui minimise la distance entre les deux paramétrisations. Ainsi, même si deux objets sont très différents, cet angle conserve son sens physique et représente l'angle de rotation permettant de ramener les deux contours l'un vers l'autre en maximisant la ressemblance.

Si on observe l'équation 1.31, on peut constater que le calcul de n_r correspond en fait à un calcul de corrélation entre les vecteurs V_p^i et V_p^j . Nous ferons donc référence à cette approche du calcul de l'angle de rotation comme à une méthode de calcul par corrélation.

1.3.3.2.2 Algorithme du spectre de phase

Il existe une autre approche qui consiste à utiliser directement le spectre de phase du vecteur de paramétrisation. Cette approche est plus rapide sur le plan du calcul puisqu'aucun calcul de transformée de Fourier n'est nécessaire. Lorsqu'il existe une rotation entre deux contours, l'algorithme converge vers le bon résultat. Mais, dans le cas contraire, l'angle calculé par cet algorithme n'a aucune signification physique alors que l'approche précédente en possédait une. Nous allons maintenant décrire plus en détails les étapes de calcul.

Supposons que les spectres de phase des vecteurs V_p^i et V_p^j soient respectivement P_p^i et P_p^j . À cause de l'imparité du spectre de phase pour un signal réel, les vecteurs P_p^i et P_p^j peuvent être ramenés à une dimension deux fois plus petite que celle du vecteur de paramétrisation. Si les deux modèles représentent le même contour, les deux spectres d'amplitude doivent être égaux à un facteur d'échelle près et les spectres de phase vérifient la relation suivante:

$$P_p^j[k] \equiv P_p^i[k] + \frac{2\pi k n_r N_s}{N} \quad (2\pi) \text{ pour } k \in \llbracket 0, N/2 \rrbracket \quad (\text{EQ 1.33})$$

où $N = 360N_s/\Theta_s$ est la dimension du vecteur de paramétrisation et n_r est la valeur entière qui permet de définir l'angle de la rotation.

Nous avons donc:

$$n_r \equiv \frac{(P_p^0[k] - P_p^i[k])N}{2\pi k N_s} \quad \left(\frac{N}{kN_s} \right) \text{ pour } k \in \llbracket 1, N/2 \rrbracket \quad (\text{EQ 1.34})$$

Cette équation permet de calculer directement n_r à partir de la seconde composante du spectre de phase ($k = 1$). La première est toujours nulle puisque le signal est réel. Il suffit de prendre la solution de l'équation qui se situe dans l'intervalle $\llbracket 0, N/N_s \rrbracket$. En effet, pour $k = 1$, cette solution est unique.

Malheureusement, comme nous l'avons déjà vu, les rotations engendrent des perturbations sur la paramétrisation. Ainsi, l'utilisation d'une seule valeur pour trouver l'angle de rotation peut engendrer des erreurs. Il vaut donc mieux utiliser également les autres composantes du spectre de phase.

Pour ce faire, nous avons utilisé une méthode récursive. Supposons que nous avons calculé n_r à l'aide des l premières composantes des spectres de phase. Notons cette valeur $n_r^{(l)}$. On définit l'intervalle I_{l+1} de la manière suivante:

$$I_{l+1} = \left[\left(n_r^{(l)} - \frac{N}{2(l+1)N_s} \right), \left(n_r^{(l)} + \frac{N}{2(l+1)N_s} \right) \right] \quad (\text{EQ 1.35})$$

On définit ensuite la $(l+1)$ ^{ème} estimée de n_r . On la note $\hat{n}_r^{(l+1)}$. Elle correspond à la solution de l'équation 1.34 (pour $k = l+1$) se trouvant dans l'intervalle I_{l+1} .

L'estimée de n_r calculée à l'aide des $l+1$ premières composantes des spectres de phase sera alors la valeur moyenne des $\hat{n}_r^{(k)}$ pour $k \in \llbracket 1, l+1 \rrbracket$.

Pour obtenir les meilleurs résultats possibles, les différentes estimées de n_r sont toutes calculées sous forme réelle. Ce n'est que lorsque toutes les composantes des spectres de phase auront été utilisées qu'une valeur entière sera associée à n_r .

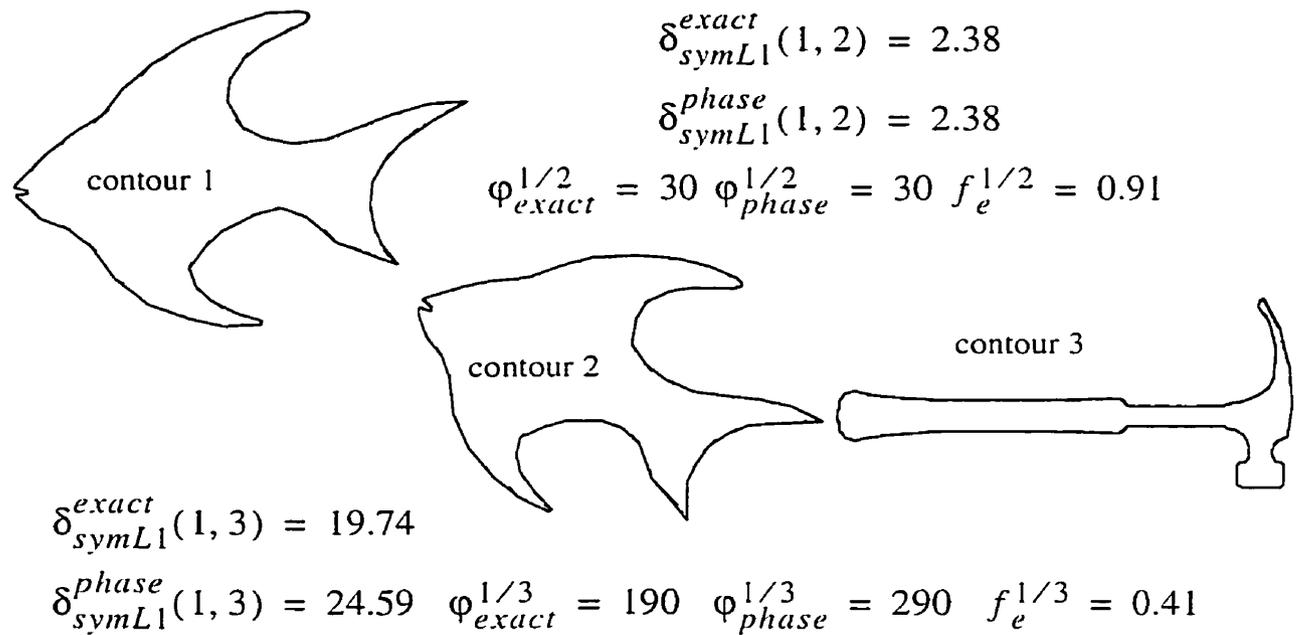


FIGURE 1.9 Exemples de calcul de la mesure de comparaison.

Dans cette figure, les contours 1 et 2 correspondent au même modèle avec une rotation de 30 degrés et un facteur d'échelle de 0.91. Le troisième contour ne correspond pas du tout aux deux autres. On constate que lorsqu'il existe une rotation entre deux contours, les deux algorithmes de calcul de l'angle sont équivalents. Par contre, lorsque les contours sont différents, l'algorithme basé sur le spectre de phase donne un résultat qui ne minimise pas la mesure de comparaison. On peut aussi constater que la mesure de comparaison est un très bon indicateur de la ressemblance de deux contours.

À cause des distorsions apportées par une rotation, il est possible que, lors du calcul de l'angle de rotation, certaines estimées $\hat{n}_r^{(l)}$ ne soient pas dans l'intervalle I_l . On suppose alors que la $l^{\text{ième}}$ composante du spectre de phase est erronée et on passe à l'étape suivante.

1.3.3.3 Exemple d'utilisation de la mesure de comparaison

Dans la figure 1.9, nous avons comparé un contour de base avec le même contour ayant subi une rotation et un changement d'échelle d'une part et avec un contour représentant un autre objet

d'autre part. On peut constater que les deux algorithmes de calcul de l'angle donnent le même résultat lorsque les contours représentent le même objet, mais qu'ils diffèrent tel que nous l'avions prédit pour les autres cas.

Le calcul de l'angle basé sur la permutation circulaire donne un angle qui minimise la mesure de comparaison ce qui lui donne un sens "physique". Il représente la rotation qui rend les deux contours les plus ressemblants parce qu'il tient compte de la corrélation entre les deux vecteurs de paramétrisation.

La seconde approche basée sur le spectre de phase est plus rapide en terme de temps de calcul et peut être utilisée lorsque le but de la comparaison est uniquement de vérifier si deux contours sont égaux ou non.

1.3.3.4 Conclusion

Nous avons défini une mesure de comparaison qui permet de comparer efficacement deux contours entre eux à partir de la paramétrisation que nous avons définie. Cette mesure nécessite le calcul préalable du facteur d'échelle et de l'angle de rotation 2D qui peuvent exister entre les deux contours. Nous avons décrit précisément les algorithmes permettant d'effectuer ces calculs. Pour l'angle de rotation, nous avons proposé deux algorithmes de calcul qui seront utilisés en fonction des besoins.

1.4 Expérimentation

1.4.1 Vérification empirique des propriétés de la modélisation

1.4.1.1 Comparaison avec des mesures existantes

Afin d'avoir une bonne idée des performances de la mesure de comparaison, nous allons l'appliquer à plusieurs modèles différents en la comparant à la distance de Hausdorff et à celle de chamfer.

Les figures 1.11 et 1.12 montrent le calcul des distances entre différents types de contours. Les distances de Hausdorff et de chamfer ont été calculées en minimisant les équations définies dans la section 1.3.3.1 en fonction de la position respective des deux contours. Pour trouver la position dans laquelle les deux contours avaient une distance minimale, nous avons considéré qu'un des deux contours était fixe et nous avons déplacé l'autre, pixel par pixel. Toutes les translations possibles entre les deux contours ont ainsi été étudiées. La figure 1.10 illustre ce principe.

Comme nous l'avons vu dans le paragraphe 1.3.3, la mesure de comparaison basée sur notre paramétrisation est définie de la manière suivante:

$$\delta_{sym}(C_j, C_i) = \text{Min}(\delta(C_i, C_j), \delta(C_j, C_i)) \quad (\text{EQ 1.36})$$

avec

$$\delta(C_i, C_j) = \left\| V_p^i - f_e^{i/j} \sigma^{(\alpha)}(V_p^j) \right\|_{L_p} \quad (\text{EQ 1.37})$$

Les mesures δ_{symL1} et $\delta_{symLinf}$ ont été introduites au paragraphe 1.3.3.1. Elles correspondent à la mesure générique en prenant respectivement les normes L_1 et L_{inf} . Pour comparer nos deux mesures avec les mesures de référence, nous avons toutefois omis le calcul du facteur

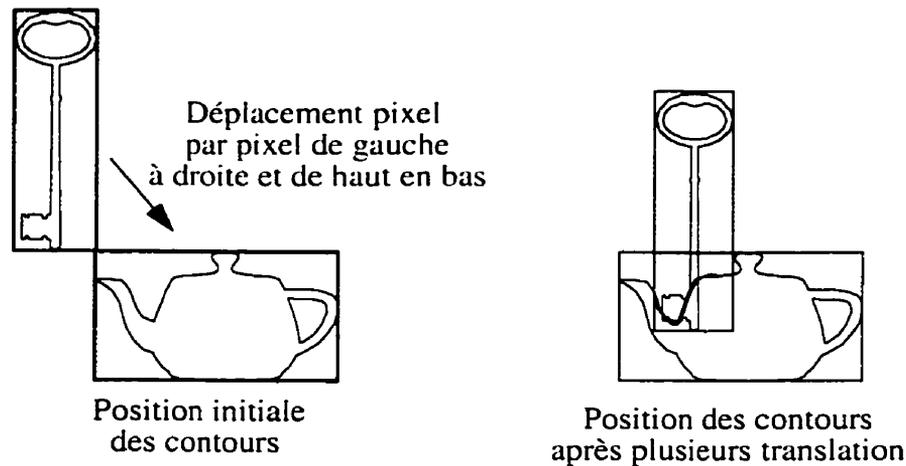


FIGURE 1.10 Positions respectives des contours pour le calcul des distances de Hausdorff et de chamfer

Cette figure illustre les différentes positions que prennent deux contours lors du calcul de leur distance de Hausdorff et de chamfer.

d'échelle et de l'angle de rotation. En effet, ni la distance de Hausdorff, ni celle de chamfer ne tiennent compte de ces deux paramètres.

La mesure $\delta_{\text{sym}L1}$ (sans facteur d'échelle ni angle de rotation) se rapproche énormément de la distance de chamfer. En effet, son interprétation physique correspond à la distance radiale moyenne entre deux modèles dont les centres de gravité ont été superposés. La distance $\delta_{\text{sym}L1}$ mesure donc la distance moyenne entre un point du premier contour et le point le plus proche de l'autre contour qui se trouve dans le même secteur angulaire. La distance de chamfer cherche elle aussi, pour un point donné du premier contour, le point le plus proche dans l'autre contour. Les figures 1.11 et 1.12 montrent que la distance de chamfer et que la mesure $\delta_{\text{sym}L1}$ réagissent globalement de la même manière lorsque les contours sont relativement ressemblants, mais peuvent diverger davantage lorsque ceux-ci sont différents. On constate également que la mesure $\delta_{\text{sym}L1}$ est toujours plus grande que la distance de chamfer. L'explication de ces phénomènes est très simple. La distance de chamfer est calculée de manière optimale sur la position respective des deux contours alors que la mesure $\delta_{\text{sym}L1}$ n'est calculée que pour une position: celle qui met en correspondance les deux centres de gravité. Lorsque les deux contours sont relativement

semblables, la correspondance des centres de gravité est presque optimale pour la distance de chamfer alors que pour des contours différents cette affirmation n'est plus valable.

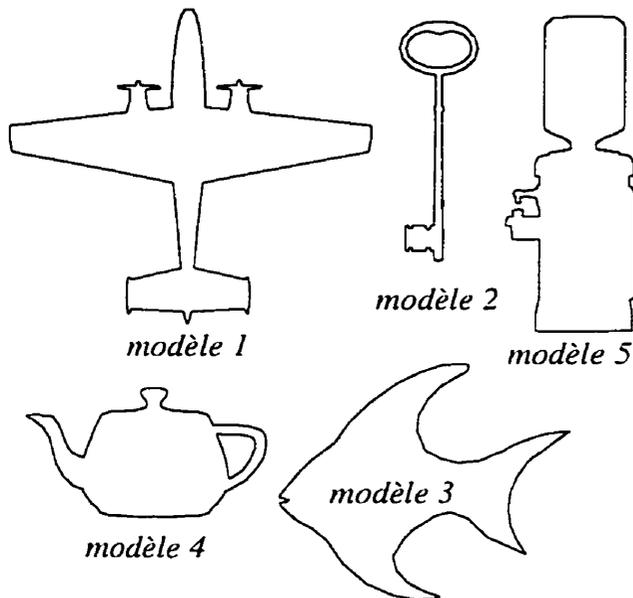
La même analogie peut être faite entre la distance de Hausdorff et la mesure δ_{symLinf} . En effet, la distance de Hausdorff évalue le plus grand écart existant entre deux contours. De la même manière, la mesure δ_{symLinf} permet de trouver la distance radiale maximale qui existe entre deux contours dont les centres de gravité sont superposés. Les figures 1.11 et 1.12 conduisent aux mêmes conclusions que précédemment. Lorsque les contours se ressemblent, les deux mesures sont très proches l'une de l'autre et, lorsque les contours diffèrent davantage, elles s'éloignent puisque la distance de Hausdorff est optimisée par rapport à la position relative des deux contours.

Nous avons également donné les valeurs réelles de la mesure δ_{symL1} (en tenant compte du facteur d'échelle et de l'angle de rotation). On peut constater visuellement que cette mesure permet bien de retrouver les deux contours les plus proches en plus de donner une bonne idée de la ressemblance ou non de deux contours.

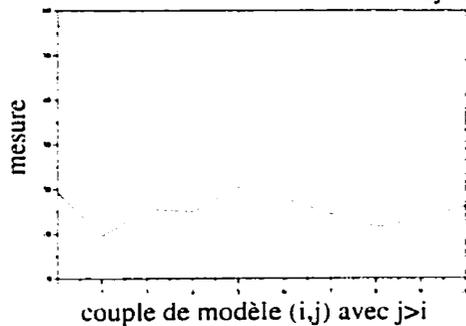
La mesure de comparaison que nous avons introduite à l'aide de notre paramétrisation est donc très bonne puisqu'elle peut être comparée à des distances qui sont des références dans la littérature. Mais, elle va encore bien plus loin que ces distances classiques puisqu'elle permet de tenir compte du facteur d'échelle et de l'angle de rotation 2D. Nous avons évalué les performances de cette mesure pour deux normes particulières qui mènent aux mêmes conclusions. Dans la suite de ce chapitre, nous ne traiterons donc plus que l'une d'entre elle: δ_{symL1} .

1.4.1.2 Évaluation des effets du facteur d'échelle

Dans le paragraphe 1.3.2.2.1 nous avons vu que, en théorie, le spectre d'amplitude du vecteur de paramétrisation est toujours proportionnel au facteur d'échelle. Malheureusement, à cause des effets de la discrétisation du modèle de silhouette, la position des points de contour possède une précision limitée. Des distorsions apparaissent par rapport au modèle théorique idéal. Afin de



distance de chamfer (bleu) et distance δ_{symL1} (vert)



distance de Hausdorff (bleu) et distance δ_{symLinf} (vert)

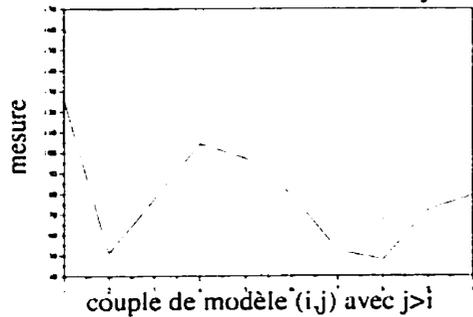


TABLEAU 1.1 Distance de Hausdorff

| Modèles | 2 | 3 | 4 | 5 |
|---------|--------|-------|-------|-------|
| 1 | 125.14 | 50.99 | 77 | 104 |
| 2 | | 96.84 | 79.07 | 51.88 |
| 3 | | | 47.54 | 72.01 |
| 4 | | | | 79 |

TABLEAU 1.2 Mesure δ_{symLinf}

| Modèles | 2 | 3 | 4 | 5 |
|---------|--------|--------|--------|--------|
| 1 | 160.67 | 104.53 | 105.08 | 129.86 |
| 2 | | 126.66 | 113.55 | 136.20 |
| 3 | | | 66.33 | 112.20 |
| 4 | | | | 101.22 |

TABLEAU 1.3 Distance de chamfer

| Modèles | 2 | 3 | 4 | 5 |
|---------|-------|-------|-------|-------|
| 1 | 19.15 | 9.43 | 15.48 | 14.81 |
| 2 | | 20.40 | 17.70 | 14.27 |
| 3 | | | 11.32 | 13.39 |
| 4 | | | | 16.08 |

TABLEAU 1.4 Mesure δ_{symL1}

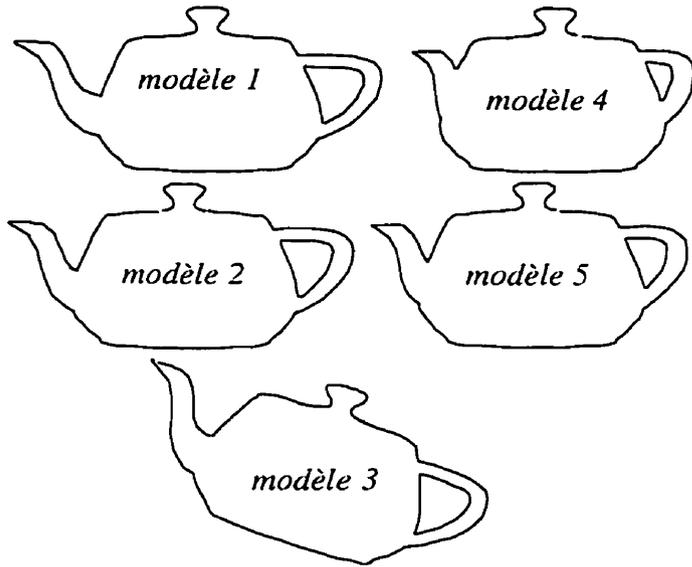
| Modèles | 2 | 3 | 4 | 5 |
|---------|-------|-------|-------|-------|
| 1 | 57.01 | 39.62 | 37.85 | 45.68 |
| 2 | | 59.65 | 51.93 | 53.21 |
| 3 | | | 27.00 | 49.58 |
| 4 | | | | 32.41 |

TABLEAU 1.5 Mesure δ_{symL1} effective

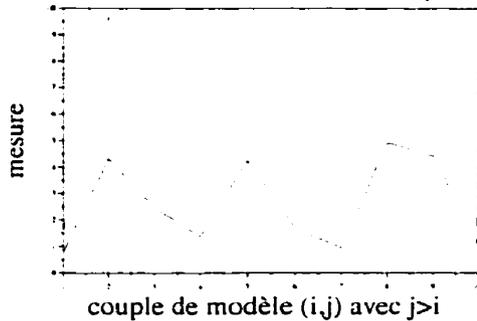
| Modèles | 2 | 3 | 4 | 5 |
|---------|-------|-------|-------|-------|
| 1 | 23.41 | 32.73 | 22.83 | 26.42 |
| 2 | | 17.07 | 17.13 | 15.11 |
| 3 | | | 20.36 | 24.18 |
| 4 | | | | 18.89 |

FIGURE 1.11 Évaluation des performances de la mesure de comparaison - premier exemple.

Les graphiques en haut à gauche représentent les différents modèles testés. Les autres graphiques représentent les courbes de comparaison des mesures pour les différents modèles. Les mesures δ_{symL1} et δ_{symLinf} sont calculées sans facteur d'échelle et sans angle de rotation sauf dans le tableau 1.5. La mesure δ_{symL1} (avec facteur d'échelle et rotation) indique que les deux modèles les plus ressemblants sont 2 et 5, ce qui est visuellement très satisfaisant.



distance de chamfer (bleu) et mesure $\delta_{\text{symL}1}$ (vert)



distance de Hausdorff (bleu) et distance $\delta_{\text{symL}1\text{inf}}$ (vert)

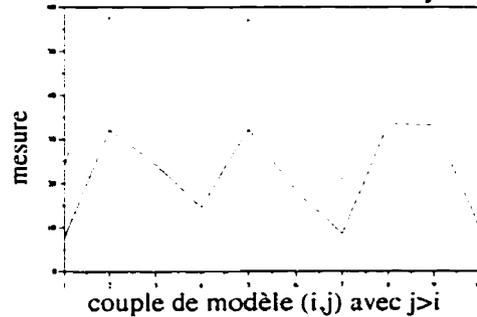


FIGURE 1.12 Évaluation des performances de la mesure de comparaison - second exemple.

Les graphiques en haut à gauche représentent les différents modèles testés. Les autres graphiques représentent les courbes de comparaison des mesures pour les différents modèles. Les mesures $\delta_{\text{symL}1}$ et $\delta_{\text{symL}1\text{inf}}$ sont calculées sans facteur d'échelle et sans angle de rotation sauf dans le tableau 1.10. La mesure $\delta_{\text{symL}1}$ (avec facteur d'échelle et rotation) indique que les deux modèles les plus ressemblants sont 1 et 3, ce qui est très satisfaisant puisqu'il s'agit du même modèle avec une rotation.

TABLEAU 1.6 Distance de Hausdorff

| Modèles | 2 | 3 | 4 | 5 |
|---------|------|-------|-------|-------|
| 1 | 7.28 | 32.06 | 24.08 | 14.32 |
| 2 | | 32.06 | 18 | 8.25 |
| 3 | | | 33.30 | 33 |
| 4 | | | | 10.05 |

TABLEAU 1.7 Mesure $\delta_{\text{symL}1\text{inf}}$

| Modèles | 2 | 3 | 4 | 5 |
|---------|-------|-------|-------|-------|
| 1 | 23.54 | 57.82 | 27.26 | 17.03 |
| 2 | | 57.24 | 31.69 | 20.38 |
| 3 | | | 56.05 | 57.17 |
| 4 | | | | 13.21 |

TABLEAU 1.8 Distance de chamfer

| Modèles | 2 | 3 | 4 | 5 |
|---------|------|------|------|------|
| 1 | 0.73 | 4.29 | 2.42 | 1.33 |
| 2 | | 4.22 | 1.64 | 0.89 |
| 3 | | | 4.89 | 4.39 |
| 4 | | | | 1.03 |

TABLEAU 1.9 Mesure $\delta_{\text{symL}1}$

| Modèles | 2 | 3 | 4 | 5 |
|---------|------|------|------|------|
| 1 | 1.55 | 9.67 | 3.33 | 2.72 |
| 2 | | 9.51 | 2.54 | 1.60 |
| 3 | | | 8.61 | 9.31 |
| 4 | | | | 1.64 |

TABLEAU 1.10 Mesure $\delta_{\text{symL}1}$ effective

| Modèles | 2 | 3 | 4 | 5 |
|---------|------|------|------|------|
| 1 | 1.55 | 1.44 | 3.79 | 2.82 |
| 2 | | 1.62 | 2.78 | 1.68 |
| 3 | | | 3.94 | 3.02 |
| 4 | | | | 1.69 |

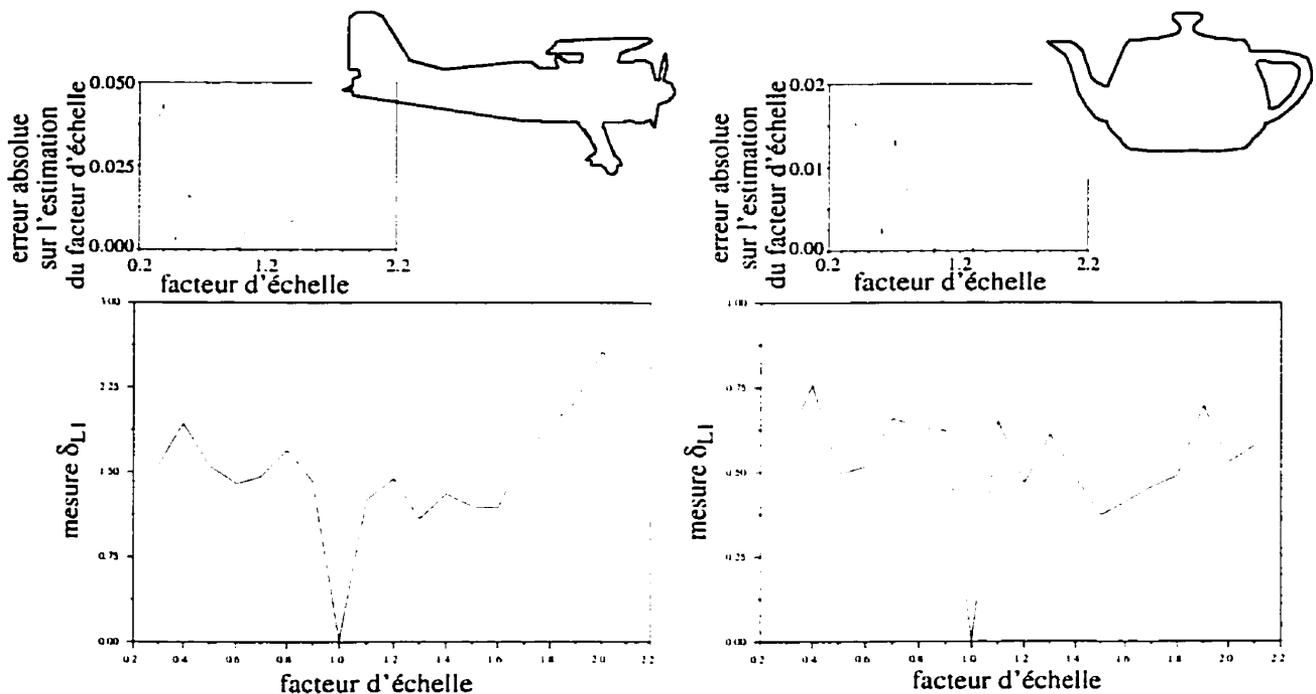


FIGURE 1.13 Effets du facteur d'échelle sur la distance.

Chaque graphique représente les variations de la mesure δ_{symL1} entre la paramétrisation du modèle de base et celle du modèle redimensionné ainsi que l'erreur absolue faite sur l'estimation du facteur d'échelle. Aucune erreur sur le calcul de l'angle de rotation n'a été commise pour les deux types de contours.

visualiser ces distorsions et de mesurer leurs effets, nous avons comparé un modèle avec lui-même à différentes échelles, comme le montre la figure 1.13.

Comme le montrent les graphiques de la mesure δ_{symL1} en fonction du facteur d'échelle, plus les deux modèles à comparer sont différents du point de vue du facteur d'échelle, plus les effets de la discrétisation se font sentir. On peut constater aussi que les courbes de la figure 1.13 ne varient pas de manière uniforme, mais plutôt en dents de scie. Ce phénomène confirme bien que les erreurs introduites sont liées à une discrétisation qui peut avoir plus ou moins d'effets néfastes selon le facteur d'échelle. On peut constater aussi que, malgré la détérioration de la mesure de comparaison, lorsque le facteur d'échelle change, le niveau d'erreur reste acceptable. La conclusion de ces simulations est la suivante: lorsqu'un modèle inconnu est comparé à un modèle

connu, le seuil de reconnaissance devra tenir compte de l'erreur engendrée par la différence d'échelle entre les deux modèles.

1.4.1.3 Évaluation des effets des rotations

1.4.1.3.1 Rotations d'angle proportionnel à Θ_s

Nous avons vu dans le paragraphe 1.3.2.2.2 que des rotations d'angle proportionnel à Θ_s engendrent une permutation circulaire du vecteur de paramétrisation. Cette propriété devrait se traduire par l'égalité des spectres d'amplitude des paramétrisations du modèle avant et après rotation. L'effet de discrétisation du modèle de silhouette vient toutefois perturber les performances théoriques de la paramétrisation. Les deux graphiques en haut à droite dans les figures 1.14 et 1.15, montrent les variations de la mesure δ_{symL1} du contour de base avant et après rotation d'angle proportionnel à Θ_s . On constate que le niveau d'erreur varie en dents de scie en fonction des angles. C'est un phénomène logique puisqu'il est lié à la discrétisation. On remarque aussi que le niveau d'erreur reste relativement faible et varie autour d'une valeur presque constante. Ceci confirme que, même si une erreur est introduite par rapport à la paramétrisation idéale, celle-ci conserve tout de même ses propriétés par rapport aux rotations d'angle proportionnel à Θ_s . De plus, on peut constater que les erreurs dans le calcul du facteur d'échelle restent très faibles et qu'aucune erreur n'est commise sur l'estimation de l'angle de rotation quel que soit le type d'algorithme choisi (à partir du spectre de phase ou par corrélation) pour le calculer.

1.4.1.3.2 Rotation d'angle quelconque

Une question se pose toutefois. Que se passe-t-il pour les rotations d'angle non proportionnel à Θ_s ? Les graphiques au milieu à droite dans les figures 1.14 et 1.15 illustrent les résultats obtenus pour les rotations d'angles entiers compris entre 1° et 40° . Même si ces rotations introduisent une erreur un peu plus grande que les rotations d'angle proportionnel à Θ_s , on constate

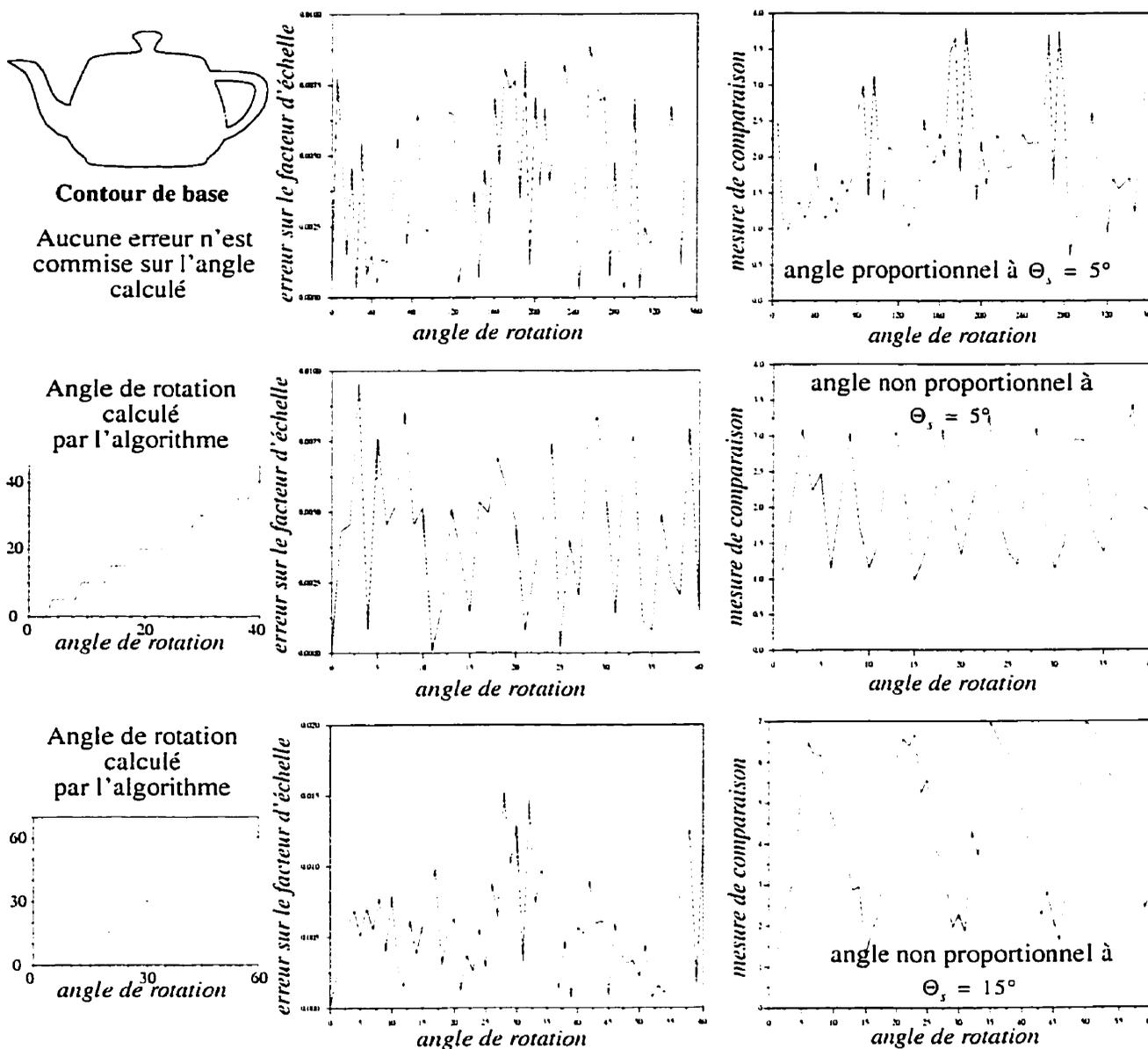


FIGURE 1.14 Effets des rotations sur la paramétrisation - premier exemple

Les graphiques de droite représentent les variations de la mesure δ_{symL1} entre la paramétrisation du contour de base et celle du contour après rotation. Pour la figure du haut, l'angle de rotation varie entre 5° et 355° avec un incrément de 5° . Pour celle du milieu, il varie entre 1° et 40° avec un incrément de 1° . Pour celle du bas, il varie entre 1° et 60° avec un incrément de 1° . Le facteur de discrétisation radiale reste toujours le même: $N_r = 5$. Par contre, le facteur de discrétisation angulaire est $\Theta_s = 5^\circ$ pour les deux graphiques du haut et $\Theta_s = 15^\circ$ pour celui du bas. L'erreur commise sur le facteur d'échelle ainsi que l'angle calculé dans l'algorithme d'évaluation de la mesure sont aussi présentés.

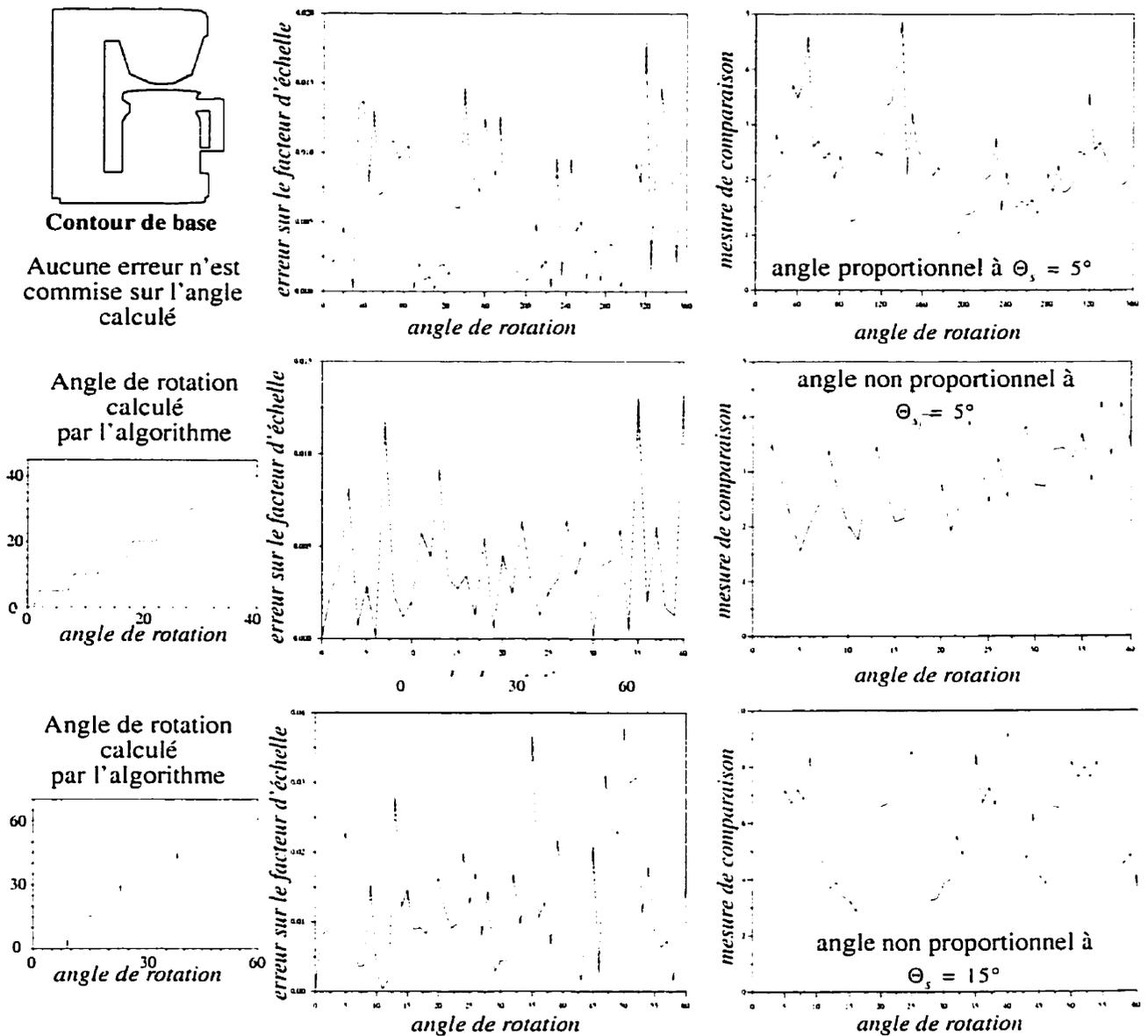


FIGURE 1.15 Effets des rotations sur la paramétrisation - second exemple

Les graphiques de droite représentent les variations de la mesure δ_{symLI} entre la paramétrisation du contour de base et celle du contour après rotation. Pour la figure du haut, l'angle de rotation varie entre 5° et 355° avec un incrément de 5° . Pour celle du milieu, il varie entre 1° et 40° avec un incrément de 1° . Pour celle du bas, il varie entre 1° et 60° avec un incrément de 1° . Le facteur de discrétisation radiale reste toujours le même: $N_r = 5$. Par contre, le facteur de discrétisation angulaire est $\Theta_r = 5^\circ$ pour les deux graphiques du haut et $\Theta_r = 15^\circ$ pour celui du bas. L'erreur commise sur le facteur d'échelle ainsi que l'angle calculé dans l'algorithme d'évaluation de la mesure sont aussi présentés.

que l'angle calculé par l'algorithme correspond en général à la meilleure approximation possible.

L'explication de ce phénomène est simple. Par des rotations d'angle non proportionnel à Θ_s , les secteurs angulaires utilisés pour le calcul des distances radiales sont légèrement décalés. L'effet de ce décalage est minimisé par le moyennage des distances radiales dans chaque secteur. En effet, pour un secteur donné, chaque partie de la courbe d'occultation qui le traverse engendre une valeur radiale. Celle-ci correspond à la valeur moyenne des points de cette partie de courbe. Comme les courbes d'occultations sont continues, le moyennage limite la distorsion. Bien sûr, si l'angle Θ_s devient trop grand, cet effet atténuateur perd de sa puissance, ainsi que le montrent les graphiques du bas des figures 1.14 et 1.15.

Les erreurs commises sur le calcul du facteur d'échelle sont également présentées dans ces deux figures. On constate que lorsque l'angle de discrétisation angulaire n'est pas trop grand, cette erreur reste très faible. La valeur calculée de l'angle de rotation montre que seuls des angles proportionnels à Θ_s sont possibles. Dans presque tous les cas, c'est le meilleur angle qui est calculé, sauf lorsque Θ_s devient trop grand. Dans ce cas, certaines erreurs peuvent survenir (graphique en bas à gauche figure 1.15).

En conclusion, on peut donc dire que, si la valeur de Θ_s est convenablement choisie, la paramétrisation est invariante à toutes les rotations 2D. Pour comparer deux modèles, il faudra toutefois utiliser un seuil tenant compte de l'erreur engendrée par une éventuelle rotation.

1.4.1.4 Conclusion

Nous venons de montrer expérimentalement que la mesure de comparaison que nous avons introduite permet de comparer efficacement deux contours. Cette mesure offre des performances similaires à la distance de Hausdorff ou de "chamfer" (si on ne tient pas compte de l'angle de rotation et du facteur d'échelle). Elle offre donc une évaluation juste de la ressemblance entre deux contours. Mais cette mesure va encore beaucoup plus loin puisqu'il est possible de l'utili-

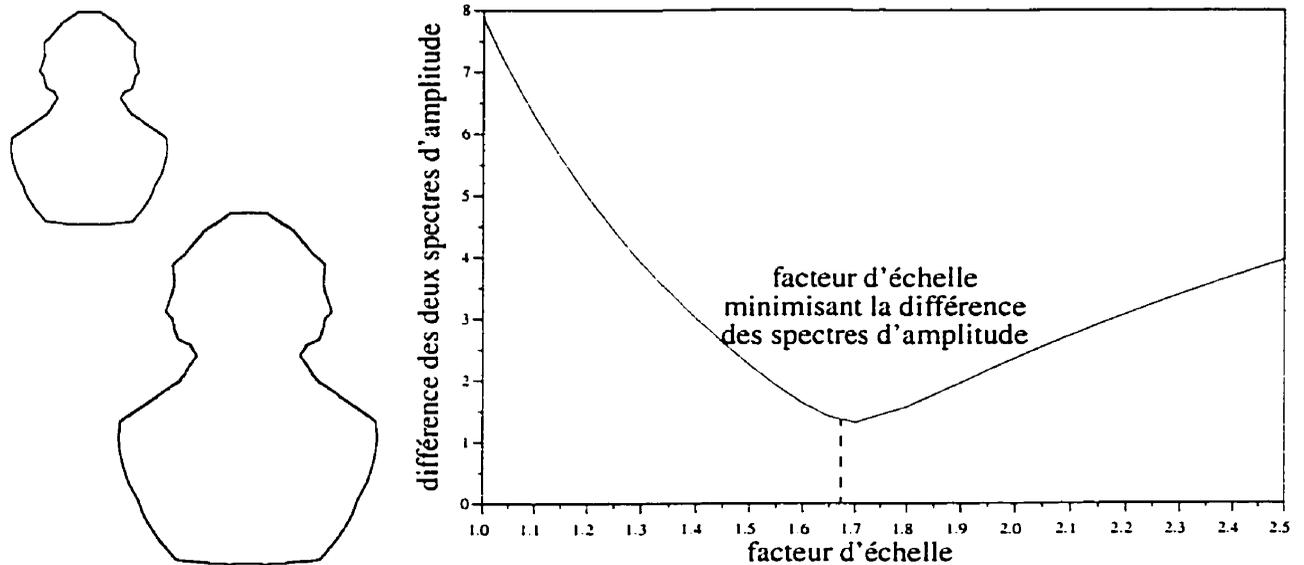


FIGURE 1.16 Récupération du facteur d'échelle

Les deux contours d'occultation ont été obtenus à partir du même objet 3D. Le seul paramètre de pose qui diffère est la distance du modèle à la caméra. Le facteur d'échelle théorique qui existe entre les deux modèles est $f_e = 5/3 \approx 1,67$. Les paramètres utilisés pour le calcul sont $\Theta_s = 5$ et $N_s = 5$. Le facteur d'échelle calculé à partir de la paramétrisation est $f_e = 1,68466$, l'angle de rotation est nul quelle que soit la méthode et la mesure de comparaison $\delta_{\text{sym}L1}$ vaut 1.60.

ser même lorsqu'il existe un angle de rotation ou un facteur d'échelle entre les deux contours à comparer. Des erreurs dues au caractère discret des contours et au facteur de discrétisation angulaire sont introduites. Celles-ci restent toutefois limitées et l'utilisation de la mesure $\delta_{\text{sym}L1}$ est très efficace non seulement pour évaluer la ressemblance de deux contours, mais aussi pour calculer le facteur d'échelle ou l'angle de rotation.

1.4.2 Quelques exemples d'utilisation de la modélisation proposée

1.4.2.1 Estimation du facteur d'échelle

Afin de savoir si les deux modèles sont effectivement identiques à un facteur d'échelle près, il faut voir si la mesure de comparaison possède une valeur raisonnable. Dans l'exemple de la figure 1.16, nous obtenons les résultats suivant: $f_e = 1,68$ et $\delta_{L1\text{sym}} = 1.60$. Les deux

modèles sont donc les mêmes à un facteur d'échelle près. L'angle de rotation calculé par l'algorithme vaut 0 degré quel que soit l'algorithme choisi.

L'utilisation combinée de notre paramétrisation et de la mesure de ressemblance permet non seulement de retrouver un éventuel facteur d'échelle entre deux modèles, mais aussi d'avoir une bonne appréciation de l'erreur qui peut être commise.

1.4.2.2 Estimation de la rotation

Dans ce paragraphe, nous évaluons les performances de notre paramétrisation et de la mesure de comparaison pour des contours égaux à une rotation 2D près. La figure 1.17 montre les résultats obtenus pour deux objets différents. Dans le cas de la théière, l'angle est proportionnel à Θ_s et on retrouve effectivement le bon angle. La mesure de ressemblance vaut $\delta_{L1sym} = 1.26$. Pour le marteau, l'angle de rotation effectif n'est pas proportionnel à Θ_s , mais le résultat est tout de même très bon puisque l'approximation trouvée est la plus proche possible du résultat exact. La mesure de ressemblance dans ce cas est un peu moins bonne $\delta_{L1sym} = 6.24$ à cause de l'angle de rotation. Cette mesure reste toutefois raisonnable et permet malgré tout de conclure à l'égalité des deux contours à une rotation près. Dans les deux cas, le calcul du facteur d'échelle est excellent.

Afin d'avoir une meilleure idée des performances de la paramétrisation proposée nous avons également calculé la ressemblance existant entre deux contours distincts. La figure 1.18 donne ces résultats. On peut constater la valeur élevée de la mesure de ressemblance qui confirme que les deux contours sont distincts. Comme nous l'avons déjà mentionné, lorsque deux contours sont différents, le calcul de l'angle de rotation change selon l'algorithme choisi. En effet, l'algorithme basé sur le spectre de phase donne un résultat qui n'a aucun sens physique. L'utilisation d'un algorithme plutôt qu'un autre dépend uniquement de l'objectif visé par le calcul. L'algorithme basé sur le spectre de phase peut très bien être utilisé lorsqu'on cherche uniquement à savoir s'il s'agit du même objet dans des positions différentes ou non. Mais il n'a plus de sens si on veut évaluer la ressemblance de contours différents.

Superposition des vecteurs de paramétrisation avant et après rotation

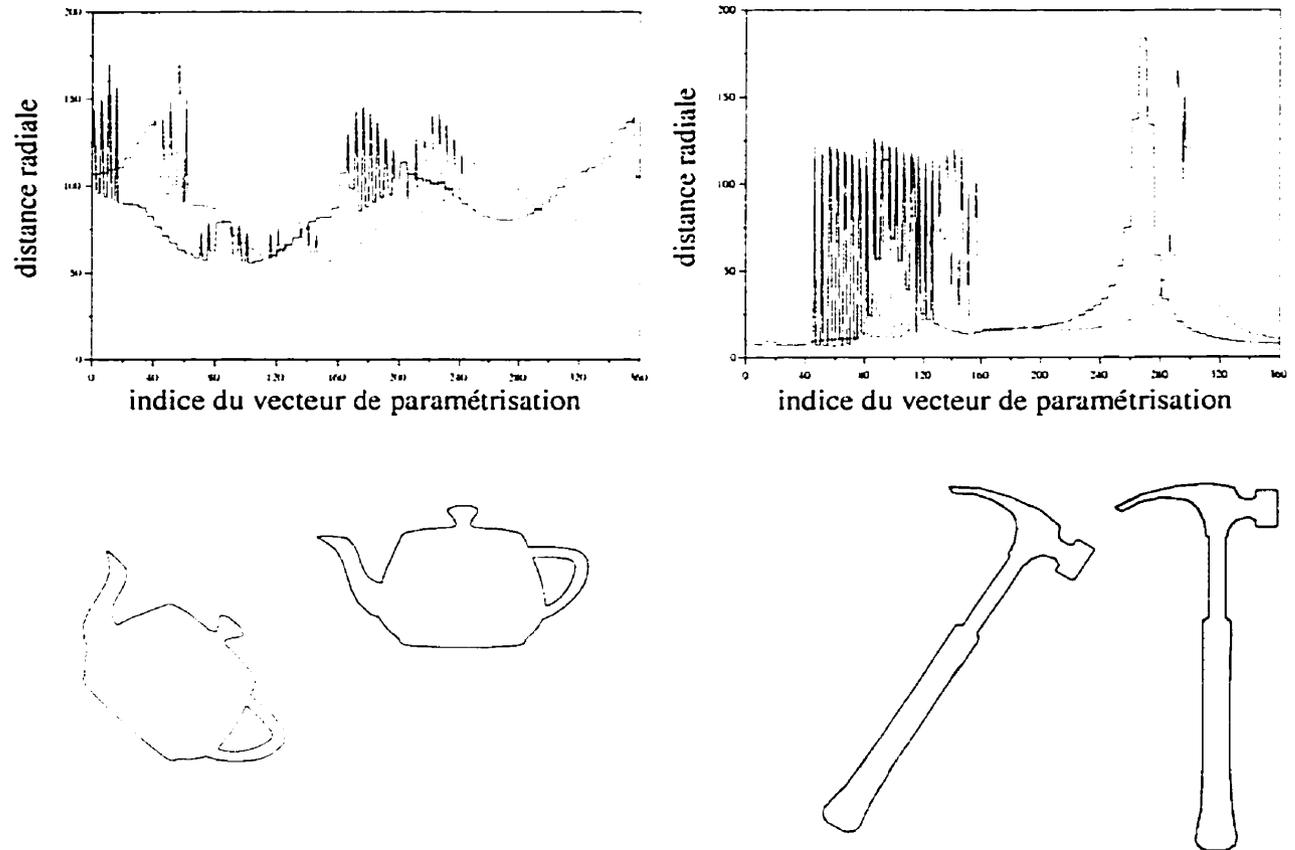


FIGURE 1.17 Récupération de l'angle - contours identiques.

Cette figure illustre le principe d'estimation de l'angle de rotation entre deux contours à l'aide de deux exemples. Pour chaque exemple, les contours d'occultation ont été obtenus à partir du même objet 3D en faisant varier la pose. Pour les deux exemples, les paramètres sont $\Theta_s = 5$ et $N_s = 5$. Pour le premier exemple, l'angle entre les deux contours est de -45° alors qu'il est de -32° pour le second.

L'angle calculé dans le premier cas est de -45° et la mesure de comparaison vaut $\delta_{symL1} = 1.26$. L'angle calculé dans le second cas est de -30° et la mesure de comparaison vaut $\delta_{symL1} = 6.24$. L'angle de rotation ne peut être calculé qu'à 5° près à cause de la valeur de Θ_s . Ceci explique la différence de niveau de la mesure de ressemblance entre les deux exemples.

Le facteur d'échelle calculé est très proche de 1 dans les deux cas: $f_e = 0.999$ pour la théière et $f_e = 1.018$ pour le marteau.

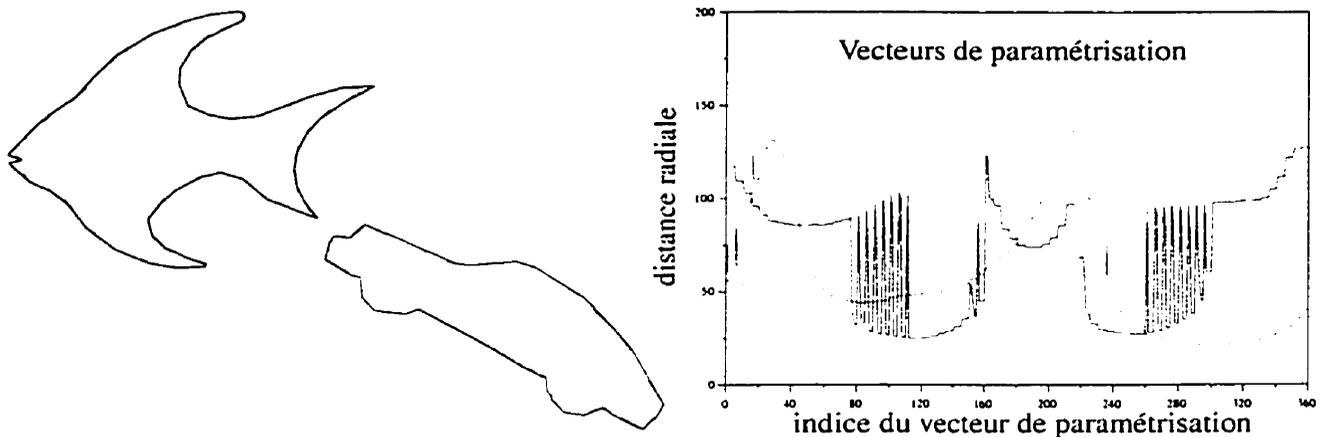


FIGURE 1.18 Récupération de l'angle - contours différents.

Cette figure illustre le principe d'estimation de l'angle de rotation existant entre deux contours. Les contours d'occultation ont été obtenus à partir de deux objets 3D distincts. Les paramètres sont $\Theta_s = 5$ et $N_s = 5$. L'angle calculé entre les deux modèles est de 180° avec la méthode basée sur le spectre de phase et de 210° avec la méthode basée sur la corrélation. La mesure de ressemblance vaut 26.96 dans le premier cas et 22.93 dans le second. À titre indicatif le facteur d'échelle vaut 0.827.

La valeur de la mesure de ressemblance est vraiment très élevée ce qui indique que les deux contours sont distincts.

1.4.2.3 Estimation de la rotation et du facteur d'échelle

La combinaison de la recherche du facteur d'échelle et de l'angle de rotation est très simple puisque les deux calculs sont indépendants. Le facteur d'échelle se retrouve dans le spectre d'amplitude alors que l'angle est obtenu à l'aide du spectre de phase.

La figure 1.19 montre les résultats pour la statue de Beethoven. On retrouve bien le facteur d'échelle et l'angle de rotation. La mesure de ressemblance permet de vérifier que les deux modèles sont bel et bien identiques. On peut constater que lors de la superposition des deux modèles, il semble y avoir une erreur de translation. Cette erreur provient du fait que la précision sur le calcul du centre de gravité du plus petit contour est moins bonne que celle du grand contour.

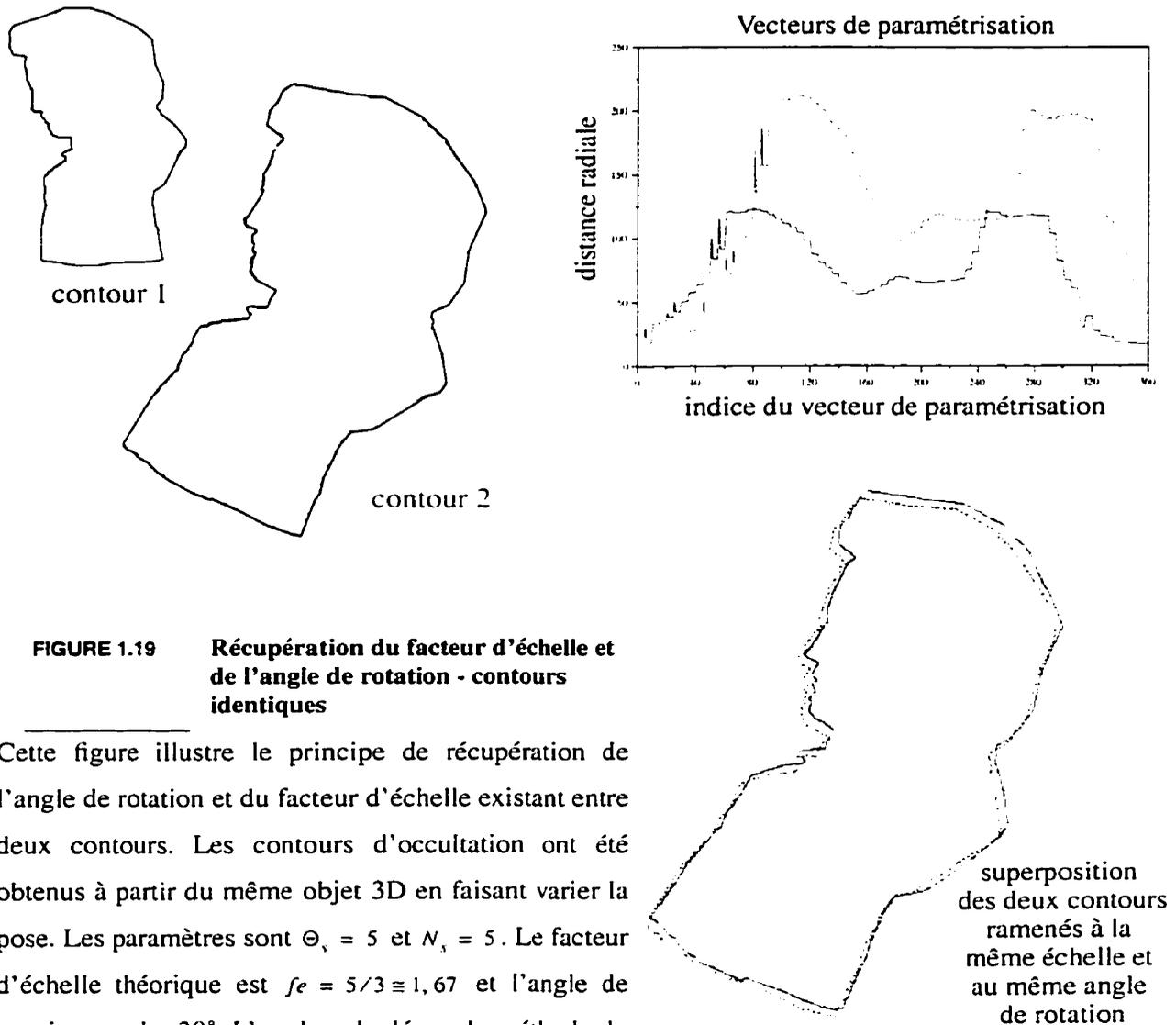


FIGURE 1.19 Récupération du facteur d'échelle et de l'angle de rotation - contours identiques

Cette figure illustre le principe de récupération de l'angle de rotation et du facteur d'échelle existant entre deux contours. Les contours d'occultation ont été obtenus à partir du même objet 3D en faisant varier la pose. Les paramètres sont $\Theta_s = 5$ et $N_s = 5$. Le facteur d'échelle théorique est $f_e = 5/3 \cong 1,67$ et l'angle de rotation est de -30° . L'angle calculé par la méthode du spectre de phase ou par la corrélation donne le même résultat: -30° . Le calcul du facteur d'échelle donne la valeur suivante: $f_e = 1,68308$. La mesure de ressemblance quant à elle vaut 3.14, ce qui nous permet de conclure que les deux contours sont identiques. Le graphique du haut représente les deux vecteurs de paramétrisation. On y voit très clairement le facteur d'échelle et la rotation. La figure en bas à droite montre la superposition du premier contour dilaté et tourné et du second contour. Les différences qui existent entre les deux contours sont dues à des problèmes de discrétisation et expliquent l'imperfection de la mesure de ressemblance.

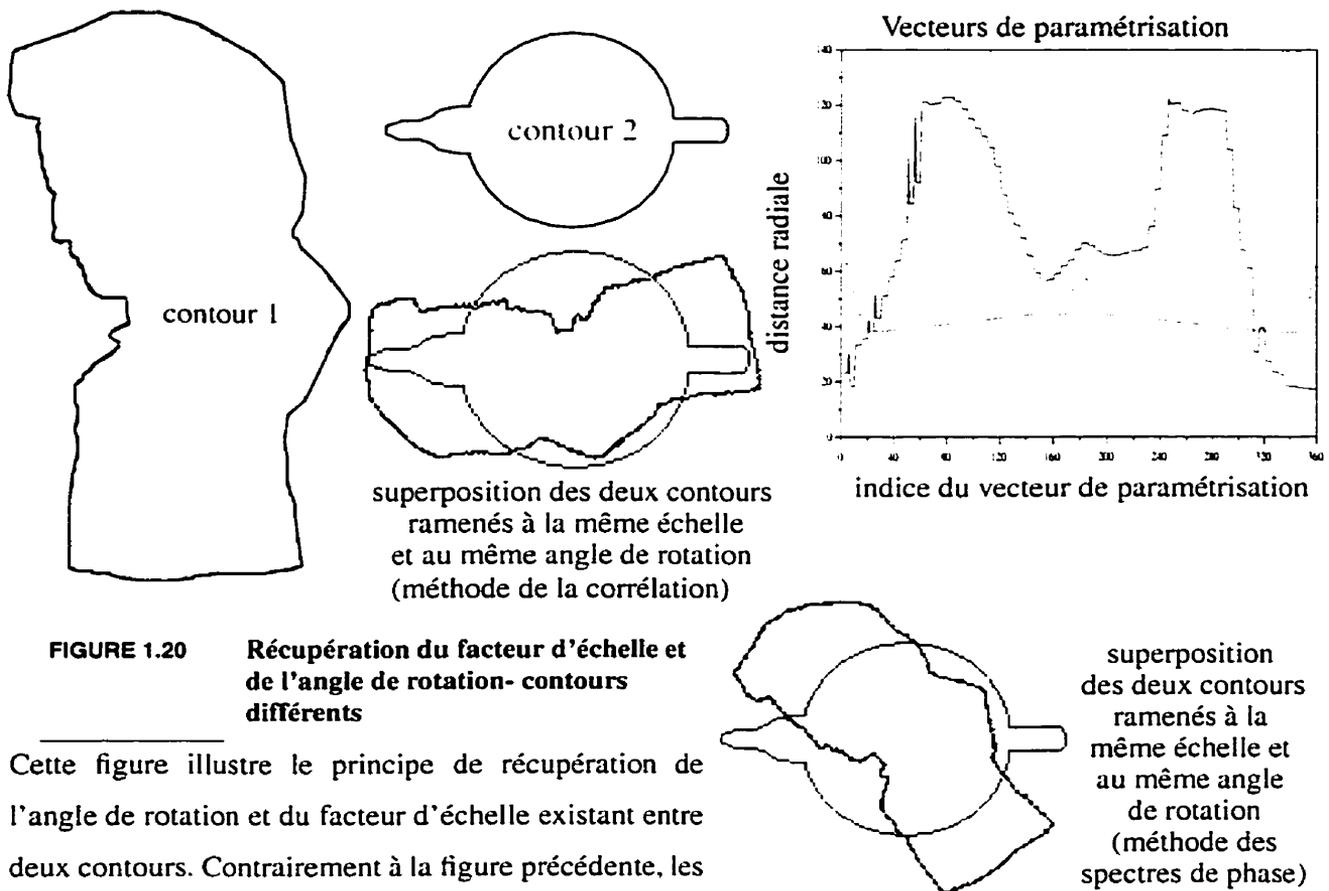


FIGURE 1.20 Récupération du facteur d'échelle et de l'angle de rotation- contours différents

Cette figure illustre le principe de récupération de l'angle de rotation et du facteur d'échelle existant entre deux contours. Contrairement à la figure précédente, les deux contours n'ont cette fois rien à voir l'un avec

l'autre. Les contours d'occultation ont été obtenus à partir de deux objets 3D distincts. Les paramètres sont $\Theta_i = 5$ et $N_i = 5$. Le calcul de l'angle de rotation à partir des spectres de phases donne un résultat de -35° , alors que ce résultat est de 285° avec l'algorithme de corrélation. Le calcul du facteur d'échelle donne la valeur suivante: $f_c = 0,573775$. Cette valeur permet de ramener un contour vers l'autre pour qu'ils soient le plus ressemblants possible. C'est grâce à la mesure de ressemblance qu'on peut dire si les deux contours sont semblables ou non. Avec l'angle issu des spectres de phase, on obtient une mesure de ressemblance de 17.84 alors qu'avec l'autre angle la mesure vaut 14.45. Ces valeurs viennent confirmer que les deux contours sont différents comme le démontre également le graphique des deux vecteurs de paramétrisation. Enfin, la superposition des deux contours ramenés à la même échelle et au même angle de rotation, permet d'illustrer visuellement que l'algorithme de corrélation a bien un sens physique alors que celui basé sur les spectres de phase n'en a pas lorsque les contours sont différents.

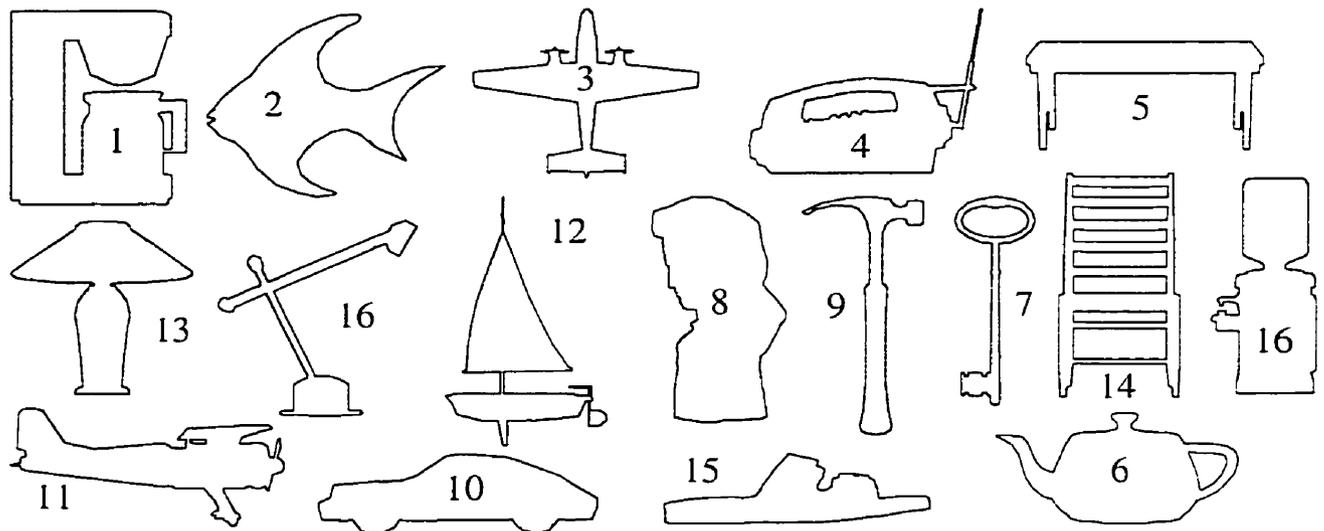


FIGURE 1.21 Base de données de contours.

Cette figure illustre les différents modèles de contour d'une base de données qui est utilisée pour reconnaître des objets. La figure 1.22 montre différents tests effectués avec cette base de modèles.

La figure 1.20 montre aussi que lorsque les objets sont différents, on peut conclure à la non-ressemblance grâce à la mesure de comparaison.

1.4.2.4 Retrouver un objet dans une base de données

L'utilisation d'une base de données de modèles contenant le vecteur de paramétrisation, le spectre d'amplitude et le spectre de phase de chacun des modèles permet de comparer un nouveau contour à un ensemble de modèles connus. Ce principe peut toutefois devenir très coûteux en temps de calcul si la base contient beaucoup d'éléments. En effet, pour chaque modèle de la base, il faut effectuer $2N$ opérations (où N est la dimension du vecteur de paramétrisation).

La figure 1.22 illustre les résultats que l'on peut obtenir avec ce type de base de données. Des contours ont été générés à partir des mêmes objets que ceux de la base. Dans certains cas, des déformations ont été introduites. Elles correspondent à de légères rotations de l'objet 3D selon un axe non perpendiculaire au plan de projection. Un contour ne correspondant pas à la base a

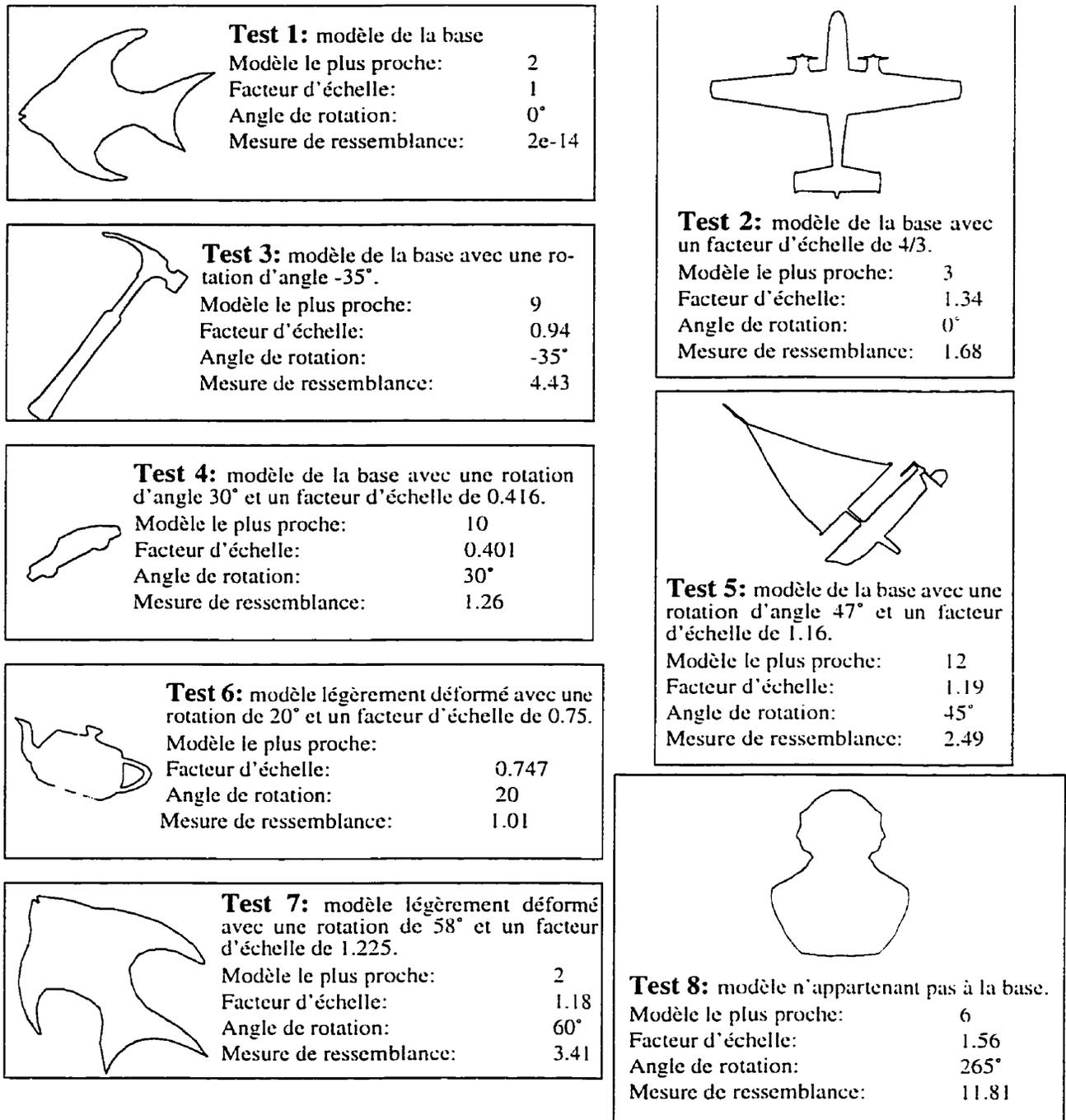


FIGURE 1.22 Utilisation d'une base pour classer des contours.

Cette figure montre l'utilisation qui peut être faite de la paramétrisation proposée. Huit tests permettent de voir que les résultats obtenus sont très bons, même lorsqu'un modèle est légèrement déformé. La base de données utilisée est illustrée dans la figure 1.21.

également été testé. Les résultats sont très satisfaisants puisque, dans tous les cas de figure, le contour a été identifié avec une mesure de ressemblance inférieure à 5. Le contour ne faisant pas partie de la base de modèle a, quant à lui, une mesure de ressemblance supérieure à 10 ce qui indique clairement qu'il ne fait pas partie de la base de données.

Un seuil permettant de décider si un contour fait partie ou non de la base de données doit être déterminé. Celui-ci peut être fixe ou dépendre du modèle le plus proche qui a été trouvé. Il peut aussi dépendre du facteur d'échelle. Les différentes valeurs de ce seuil devraient alors être calculées à partir d'une série de tests.

1.4.2.5 Conclusion

Nous venons de prouver l'efficacité de la paramétrisation et de la mesure de comparaison associée. Nous ne détaillerons pas plus les applications possibles, même s'il reste beaucoup à en dire pour des applications hors des objectifs des travaux présentés dans cette thèse.

1.4.3 Variation des paramètres de la modélisation

1.4.3.1 Centre de gravité

Lorsque nous avons décrit le principe de paramétrisation des contours d'occultation, nous avons été obligé d'introduire certains paramètres. Le premier est le centre de gravité du contour. Nous avons vu, dans le paragraphe 1.3.2.2, que c'est grâce à l'utilisation de ce centre de gravité que les propriétés d'invariance aux rotations et de proportionnalité au facteur d'échelle pouvaient être assurées. Par conséquent, si une perturbation vient déplacer le centre de gravité d'un contour, les performances de la paramétrisation en seront sérieusement affectées.

Cette sensibilité à la position du centre de gravité est très bien illustrée par la figure 1.23. On voit en effet que l'erreur engendrée par une variation sur la position du centre de gravité croît rapidement en fonction de l'erreur sur la position.

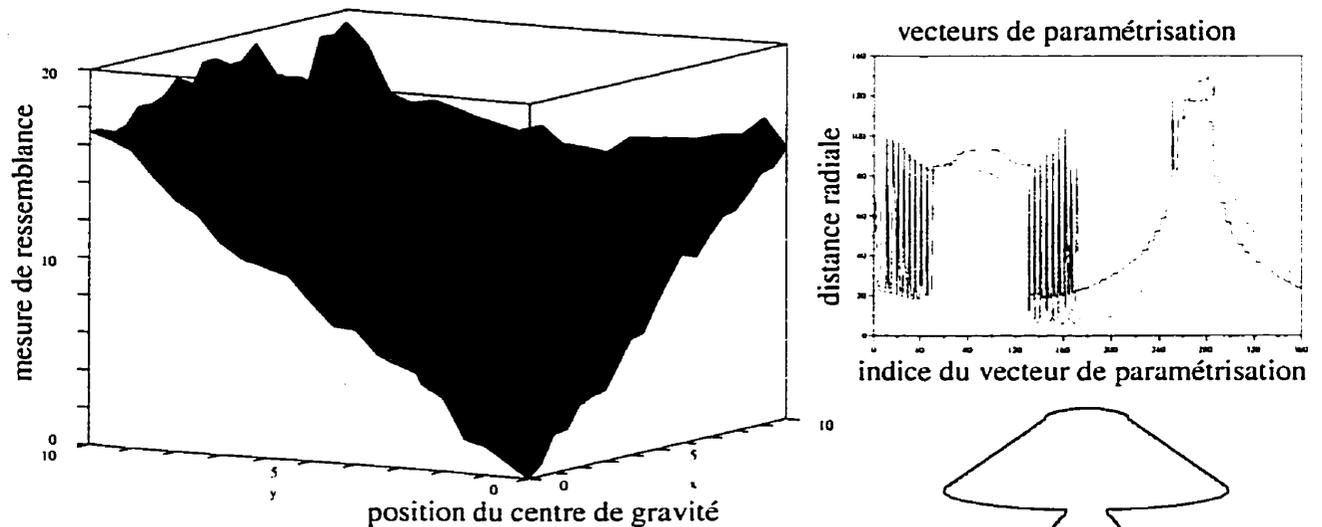


FIGURE 1.23 Variation de la position du centre de gravité

Cette figure illustre l'effet d'une erreur dans la position du centre de gravité sur la paramétrisation. Le graphique en haut à gauche représente la mesure de ressemblance entre la paramétrisation du contour sans erreur et celle du même contour avec une erreur sur la position du centre de gravité. L'erreur sur la position du centre de gravité varie entre 0 et 10 pixels dans les deux directions. On peut constater que la paramétrisation est très sensible à des variations sur la position du centre de gravité puisque la mesure de ressemblance croît rapidement en fonction de l'erreur. À titre indicatif, nous avons également montré les vecteurs de paramétrisation pour le contour sans erreur (courbe en bleu) et pour le contour avec une erreur de 10 pixels dans chaque direction (courbe en rouge). La figure en bas à droite représente le contour utilisé pour cette simulation.

Les nombreux avantages de la paramétrisation que nous proposons ont un prix: la sensibilité par rapport à la position du centre de gravité. Il s'agit d'un facteur dont il faut tenir compte lorsqu'on utilise cette paramétrisation.

1.4.3.2 Facteur de discrétisation angulaire

Le facteur de discrétisation angulaire permet d'obtenir un échantillonnage plus ou moins riche du contour d'occultation. Comme nous l'avons déjà mentionné, choisir un facteur trop grand

limite sérieusement la faculté de discrimination de la paramétrisation. En effet, seules les variations globales du contour sont prises en compte. Par contre, choisir un facteur de discrétisation angulaire trop petit augmente inutilement la quantité de données à traiter. Il s'agit donc de trouver un compromis entre puissance de discrimination et taille des données à traiter.

La description polaire du contour d'occultation amène parfois certains problèmes d'échantillonnage. En effet, certaines zones d'un contour peuvent être très bien décrites alors que d'autres le seront nettement moins. C'est en particulier le cas pour les parties de courbes qui correspondent à des segments de droites se confondant avec des rayons issus du centre de gravité. Ce problème peut partiellement être résolu par une diminution du facteur de discrétisation radiale.

D'après les tests qui ont été effectués, une valeur empirique de 5 degrés semble très bien adaptée.

1.4.3.3 Facteur de discrétisation radiale

Ce facteur de discrétisation n'a pas un impact très important sur le comportement global de la paramétrisation. Il suffit de choisir une valeur raisonnable pour que les courbes relativement complexes soient bien représentées. Une valeur empirique de 5 composantes est amplement suffisante dans les tests que nous avons faits.

1.5 Conclusion

Dans ce chapitre, nous avons effectué un survol des différentes modélisations 2D utilisées dans la littérature. Nous avons ensuite introduit une nouvelle paramétrisation et une mesure de ressemblance offrant des propriétés très intéressantes pour reconnaître un contour ayant subi une homothétie et une rotation dans le plan. La description détaillée du calcul de la paramétrisation ainsi que la démonstration théorique de ses propriétés ont été faites. De nombreuses simulations ont été effectuées et analysées afin d'identifier les forces et les faiblesses de la méthode. Les performances de cette paramétrisation rencontrent bien les objectifs qui ont été fixés. Cette

méthode va donc pouvoir être utilisée de manière avantageuse dans les prochaines étapes du traitement dont l'un des objectifs est de générer un graphe d'aspect intéressant.

2.1 Introduction

La construction d'un graphe d'aspect est une approche prometteuse pour décrire l'apparence que peut prendre un objet 3D dans une image 2D. Cet outil permet en effet de jeter un pont entre l'aspect 2D d'un objet et sa position réelle dans l'espace 3D. Cependant, malgré le fait que ce concept ait été introduit il y a déjà plus de trente ans, on l'utilise peu dans des applications pratiques de calcul de pose ou de reconnaissance d'objets. Plusieurs chercheurs se sont penchés sur les raisons de cette désaffection dans un article intitulé «Why aspect graphs are not (yet) practical for computer vision» [30]. Plusieurs points intéressants ressortent de cette étude. En premier lieu, il est important de noter que les graphes d'aspect ne contiennent aucune information sur l'échelle de l'objet 3D. Cette lacune a des conséquences importantes lors du calcul de sa position. Dans le même ordre d'idées, on peut mentionner qu'aucune information concernant une rotation 2D de l'objet dans le plan image n'est encodée dans le graphe d'aspect. Il apparaît donc clairement que pour utiliser les graphes d'aspect dans des applications de localisation de cible, il faut tenir compte du facteur d'échelle et de la rotation 2D.

Tout objet rigide possède six paramètres de pose, trois de translation et trois de rotation. Deux des trois paramètres de translation correspondent à un déplacement dans un plan parallèle au plan image. Le troisième se rapporte à la distance entre la caméra 2D et l'objet. Les paramètres de rotation correspondent, quant à eux, aux angles d'Euler. Calculer les trois angles d'Euler et la

distance objet-caméra pour un contour d'occultation donné revient à trouver la vue du graphe d'aspect qui, après rotation et homothétie 2D, s'ajuste le mieux sur ce contour. En effet, les coordonnées sphériques de la vue dans le graphe d'aspect donnent les deux premiers angles d'Euler tandis que la rotation 2D donne le troisième angle et que le facteur d'échelle permet de retrouver la distance objet-caméra.

Dans le chapitre précédent, nous avons présenté une paramétrisation des contours d'occultation originale ayant de nombreux avantages. Elle permet non seulement de calculer le facteur d'échelle et l'angle de rotation entre deux contours, mais aussi de vérifier, à l'aide d'une mesure de comparaison, la qualité de la ressemblance de ces deux contours. L'utilisation de cette paramétrisation pour les différentes vues du graphe d'aspect s'impose donc comme une solution élégante qui permet d'exploiter convenablement les graphes d'aspect. Nous décrivons cette approche dans la suite de notre exposé.

Dans le paragraphe 2.2 nous introduirons la notion de graphe d'aspect et nous définirons certains termes. Nous discuterons ensuite des applications qui ont été proposées dans la littérature et de la manière de calculer le graphe d'aspect.

La section 2.3 fera ensuite le lien entre la définition générale des graphes d'aspect et la définition pratique impliquant la paramétrisation des contours d'occultation.

Enfin, le paragraphe 2.4 sera consacré à l'utilisation du graphe d'aspect pour le calcul de pose. Une approche relativement simple, quoique coûteuse en temps de calcul, sera présentée. Les performances obtenues nous portent à croire qu'elle pourrait être exploitée dans plusieurs applications. Une autre méthode, utilisant une analyse par composantes principales des différentes vues d'un graphe d'aspect, sera aussi développée. Cette seconde méthode réduit sensiblement le temps de calcul, sans pour autant diminuer les performances.

Une conclusion résumant l'ensemble des innovations apportées par cette nouvelle approche des graphes d'aspect terminera le chapitre.

2.2 État de l'art

2.2.1 Définition du graphe d'aspect

Koenderink et van Doorn [50] ont introduit le concept de graphe d'aspect dans le milieu des années 1970. Ils ont décrit cette représentation multi-vues d'un objet 3D à l'aide d'une structure de graphe dans laquelle chaque noeud représente un **aspect général** de l'objet. Ces différents aspects sont définis à partir d'un "découpage" en cellules de l'espace des points de vue. Dans chaque cellule, toutes les vues de l'objet 3D sont équivalentes, ce qui signifie que toutes les vues contiennent les mêmes caractéristiques. En se déplaçant dans l'espace des points de vue, des caractéristiques de l'objet 3D apparaissent ou disparaissent en certains points particuliers appelés **événements visuels**. Ceux-ci permettent de calculer la frontière entre les différentes cellules. Selon la terminologie utilisée dans le domaine des graphes d'aspect, on appelle aussi ces frontières des **vues accidentelles** de l'objet 3D. Le choix du terme "accidentel" s'explique par le fait que la probabilité de voir l'objet selon de tels points de vue est très faible. Chaque vue accidentelle permet de passer d'un aspect général à un autre, c'est pourquoi elles correspondent aux arêtes. La partition topologique de l'espace des points de vue définit en quelque sorte une énumération des aspects fondamentalement différents d'un objet 3D. La figure 2.1 illustre ces différents concepts dans le cas d'un objet très simple.

La définition précédente fait référence à des vues 2D. Celles-ci correspondent habituellement aux contours visibles de l'objet lorsqu'il est projeté dans une image 2D, mais elles ne sont pas limitées à cette définition et pourraient contenir des informations de couleur, de texture, ou d'illuminance [68]. Toutefois, l'illuminance est peu utilisée parce qu'elle représente une information de trop bas niveau sémantique et dépend énormément du contexte dans lequel l'image a été prise (importance de l'arrière-plan et de l'éclairage).

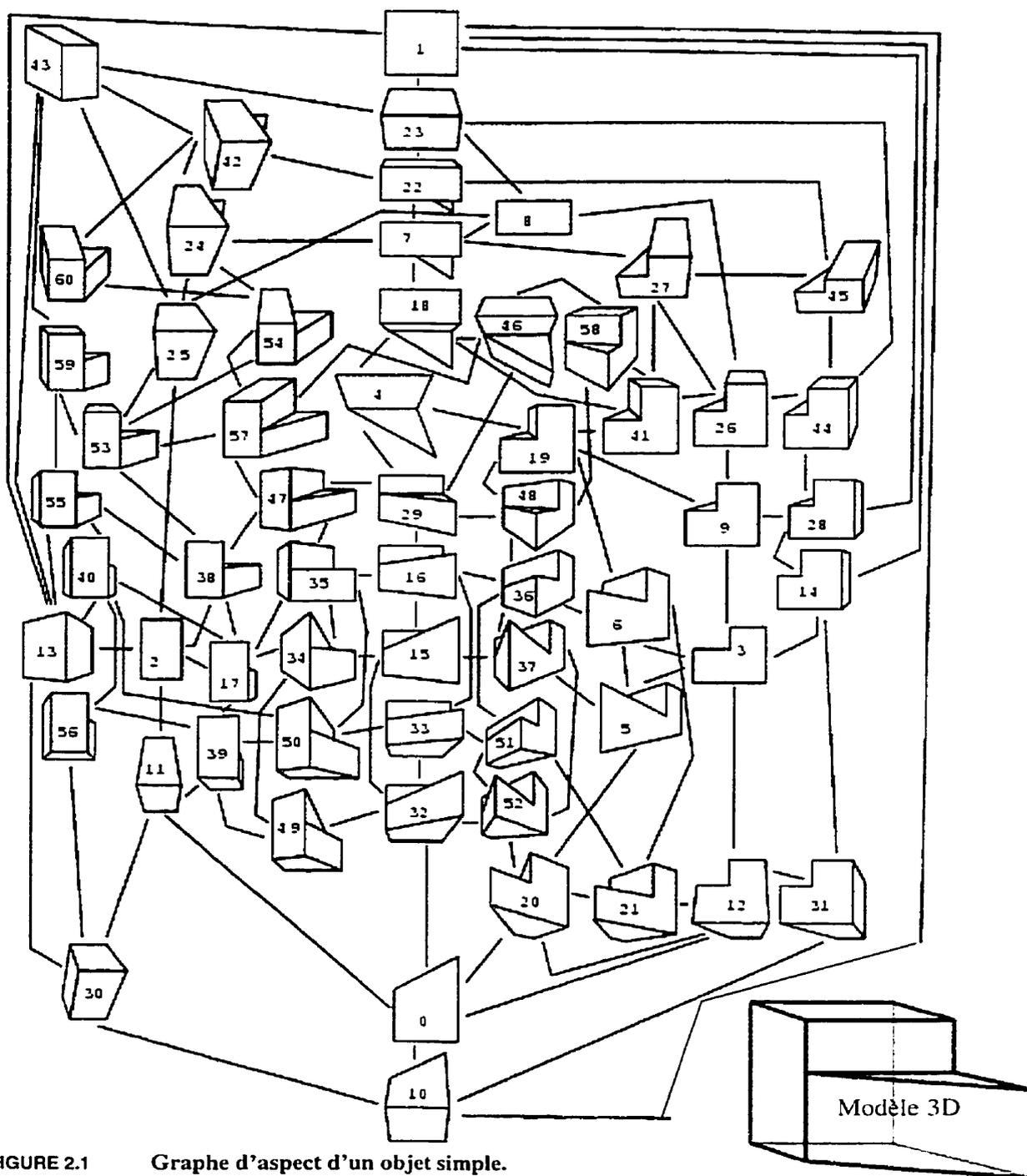


FIGURE 2.1 Graphe d'aspect d'un objet simple.

Dans cet exemple (extrait de [74], page 133), on peut voir les différents aspects 2D que l'objet 3D peut prendre. Chaque aspect est identifié par un nombre et les liens du graphe qui relient deux aspects correspondent aux vues accidentelles permettant de passer d'un aspect à l'autre.

2.2.2 Génération de graphes d'aspect

La définition générale des graphes d'aspect laisse supposer que leur calcul n'est pas une opération très simple parce qu'il s'agit de trouver toutes les vues accidentelles nécessaires à la partition de l'espace des points de vue. Deux approches sont possibles. La première correspond à une analyse mathématique de l'objet 3D et permet un calcul exact du graphe d'aspect. La seconde, en revanche, utilise une méthode approximative basée sur un découpage régulier de l'espace des points de vue.

2.2.2.1 Calcul exact

Dans la littérature, de nombreux articles traitent du calcul exact du graphe d'aspect d'objets 3D. Les algorithmes qu'ils présentent peuvent être classés selon trois critères principaux: le type de modèle utilisé pour décrire l'objet 3D à traiter, le type de projection et enfin le type de primitives qui décrivent chacune des vues 2D.

Le calcul exact du graphe d'aspect ne peut se faire que si l'objet 3D est défini à l'aide d'un modèle précis. On trouve ainsi des algorithmes de calcul pour des formes polyédriques [16,37,58,73], des solides de révolution [25,56], des modèles contenant des surfaces paramétriques [17,67,72] et même des modèles articulés [26]. Mais il n'existe aucune approche générale permettant de traiter n'importe quel objet.

Pour tous les types de modèles ci-dessus, il existe des algorithmes qui traitent la projection orthogonale et la projection de perspective, même si celle-ci augmente la complexité des calculs.

Dans la quasi-totalité des cas, les différentes vues de l'objet 3D sont décrites à l'aide d'un dessin au trait. Celui-ci correspond à un ensemble de segments de droite qui représente les contours de l'objet et à un ensemble de jonctions qui décrivent les liens entre les segments. Cette représentation 2D d'une vue est appelée "*image structure graph*" (ISG). Une description détaillée de cette structure de graphe est donnée dans [64].

Le calcul du graphe d'aspect se fait à l'aide de l'ISG. En effet, une vue accidentelle correspond à un point de l'espace des points de vue tel qu'un changement infinitésimal autour de ce point fasse passer d'un ISG à un autre qui ne lui est pas isomorphe. Cette propriété permet de construire la partition de l'espace des points de vue en fonction des caractéristiques du modèle de l'objet 3D.

La complexité de ces algorithmes dépend directement de celle des objets et peut donc augmenter rapidement. De plus, chaque vue doit être décrite à l'aide de contours et de jonctions, ce qui peut être inutilement complexe selon l'utilisation que l'on cherche à faire du graphe d'aspect.

2.2.2.2 Calcul approximatif

Il existe une seconde approche pour calculer le graphe d'aspect d'un objet 3D. Il ne s'agit plus d'un calcul exact avec une partition parfaite, mais d'un calcul approximatif. La première étape consiste à discrétiser l'espace des points de vue de manière régulière. Ensuite, on associe à chaque point de vue l'aspect de l'objet qui y correspond. Enfin, on regroupe les vues équivalentes pour former un graphe se rapprochant le plus possible du graphe d'aspect exact. La discrétisation de l'espace des points de vue correspond habituellement à une tessellation régulière de la sphère de points de vue. Cette approche a été particulièrement utilisée pour des applications basées sur une modélisation CAO [39,45,55].

L'une des difficultés de cette approche réside dans le regroupement de vues équivalentes. Il faut, en effet, définir une mesure de similitude entre deux vues. De plus, avec cette approche, certaines vues peuvent être négligées. Ceci reste toutefois marginal puisque ces vues sont accidentelles et qu'elles n'ont qu'une très faible probabilité d'apparition.

2.2.3 Utilisation des graphes d'aspect

Les graphes d'aspect ont été utilisés pour différents types d'applications. Ils permettent, par exemple, de prédire les caractéristiques à rechercher dans une image contenant un objet dans

une certaine position. Ces informations peuvent notamment servir à la construction d'un arbre d'interprétation qui permet de reconnaître un objet à partir de caractéristiques extraites d'une image. Différentes approches de ce type existent dans la littérature [18,45].

La vocation première des graphes d'aspect est la reconnaissance d'objets. En effet, comme le graphe d'aspect contient directement les différents aspects 2D d'un objet 3D, il permet, en théorie, de reconnaître un objet particulier dans une image. Il donne aussi des informations sur la position dans l'espace de l'objet 3D puisque les différentes cellules du graphe d'aspect contiennent de manière implicite des paramètres de rotation. Eggert et al. ont ainsi utilisé les graphes d'aspect pour reconnaître différents objets polyédriques simples et calculer certains paramètres de leur pose respective [28].

Comme nous l'avons déjà mentionné dans l'introduction, les graphes d'aspect n'ont pas été très utilisés en vision notamment parce qu'ils ne contiennent aucune information sur l'échelle de l'objet 3D [30]. Eggert et al. ont proposé une solution à ce problème en incluant des informations de ce type dans le graphe d'aspect [27], mais sans que cette approche soit suffisante pour relancer l'utilisation des graphes d'aspect.

Un autre point important concerne la complexité du graphe d'aspect. En effet, plus un objet possède de caractéristiques particulières, plus le graphe d'aspect doit contenir de cellules qui doivent alors être indexées judicieusement afin de permettre la reconnaissance de l'objet.

Mundy évoque la difficulté d'utiliser un graphe d'aspect pour retrouver la pose exacte d'un objet [30]. En effet, la définition générale du graphe d'aspect est basée sur une notion d'aspect équivalent du point de vue de la **présence** de caractéristiques particulières de l'objet. Or, le calcul de la pose d'un objet fait appel à l'utilisation de la **position** de ces caractéristiques.

Le dernier point important concerne les primitives utilisées dans les graphes d'aspect. En effet, celles-ci devront pouvoir être extraites aisément d'une image, même bruitée. Les primitives utilisées dans les graphes d'aspect sont le plus souvent des segments de droites. L'extraction de tel-

les primitives n'est pas invariante lorsqu'il s'agit de contours qui ne sont pas exactement des segments de droites. Il devient alors très difficile de relier un segment de droite extrait d'une image à un segment de droite du graphe d'aspect. Afin de limiter ce problème, il existe différentes techniques utilisant la géométrie 3D de l'objet et permettant d'apparier plusieurs primitives à la fois. Cependant, comme le dit Ahuja dans [30], on perd alors tout l'intérêt des graphes d'aspect, puisqu'on doit revenir à l'objet 3D.

2.2.4 Conclusion

Dans cette section, nous avons décrit ce que sont les graphes d'aspect et comment ils peuvent être calculés de manière exacte ou de manière approximative. Nous avons également vu que leur structure en faisait des candidats intéressants pour résoudre les problèmes de calcul de la position d'un objet connu dans une image 2D. Malheureusement, certaines lacunes les rendent peu adaptés à des applications pratiques, ce qui explique le peu d'intérêt des chercheurs à leur égard. Nous comptons cependant remédier à cet état de fait en combinant les graphes d'aspect et la modélisation 2D des contours d'occultation développée dans le chapitre précédent.

2.3 Graphe d'aspect et modélisation 2D

Après avoir décrit de manière générale les graphes d'aspect ainsi que leurs avantages et leurs inconvénients, nous allons maintenant présenter le type de graphe d'aspect que nous avons construit. Nous justifierons ensuite nos choix en fonction des critiques habituellement faites aux graphes d'aspect.

2.3.1 Création des différentes vues

Nous utiliserons un graphe d'aspect approximatif représenté par un ensemble de vues 2D. Dans notre étude, ces vues correspondent tout simplement au contour d'occultation de l'objet 3D tel que nous l'avons défini dans l'introduction. Nous supposerons que l'objet 3D est modélisé à l'aide du langage VRML. Rappelons que l'intérêt des contours d'occultation tient au fait qu'ils

sont plus simples à retrouver dans une image 2D que des contours de discontinuités de surface ou des contours d'illuminance ou de changement de texture.

Afin de paramétrer le graphe d'aspect, nous prendrons la bijection qui existe entre l'ensemble des vues 2D et la sphère de points de vue. Pour cela, nous utiliserons deux angles φ et ψ qui varient respectivement dans les intervalles $[0, \pi]$ et $[0, 2\pi[$. Pour obtenir un graphe d'aspect avec un nombre discret de vues, il suffit de limiter les valeurs que peuvent prendre ces deux angles. On peut, par exemple, choisir de ne conserver que les angles proportionnels à 10° . On obtient alors une répartition régulière sur la sphère de points de vue. On parle alors d'une tessellation régulière. Il est aussi possible de ne conserver que les angles qui décrivent les changements d'aspects d'un objet. Cette seconde méthode nécessite cependant des algorithmes de calcul complexes et ne s'applique souvent qu'à une classe particulière d'objets comme les solides de révolution. Dans la suite, nous nous limiterons donc à l'utilisation d'une tessellation régulière de la sphère de points de vue.

Celle-ci est obtenue à partir d'un maillage régulier de l'ensemble $\Gamma = [0, \pi] \times [0, 2\pi[$ défini à l'aide de deux valeurs entières que nous noterons N_φ et N_ψ . Ces deux valeurs permettent en effet de définir les pas de discrétisation de l'ensemble Γ . L'ensemble des points d'une tessellation discrète de la sphère a donc la forme suivante:

$$\Gamma_d = \left\{ \frac{k\pi}{N_\varphi} \text{ avec } k \in \llbracket 0, N_\varphi \rrbracket \right\} \times \left\{ \frac{2l\pi}{N_\psi} \text{ avec } l \in \llbracket 0, N_\psi - 1 \rrbracket \right\} \quad (\text{EQ 2.1})$$

Dans la pratique, on utilise une tessellation régulière ayant un pas identique pour les deux angles. La définition de la valeur entière N_φ est alors suffisante puisque $N_\psi = 2N_\varphi$.

L'ensemble Γ_d est utilisé pour générer les différentes vues du graphe d'aspect. Chacune des vues correspond au contour d'occultation de l'objet, suivant un point de vue particulier sur la sphère. Ainsi, si on place le centre de gravité de l'objet 3D au centre de la sphère, chaque couple $(\varphi, \psi) \in \Gamma_d$ permet de définir un point de vue. De manière pratique, au lieu de déplacer le

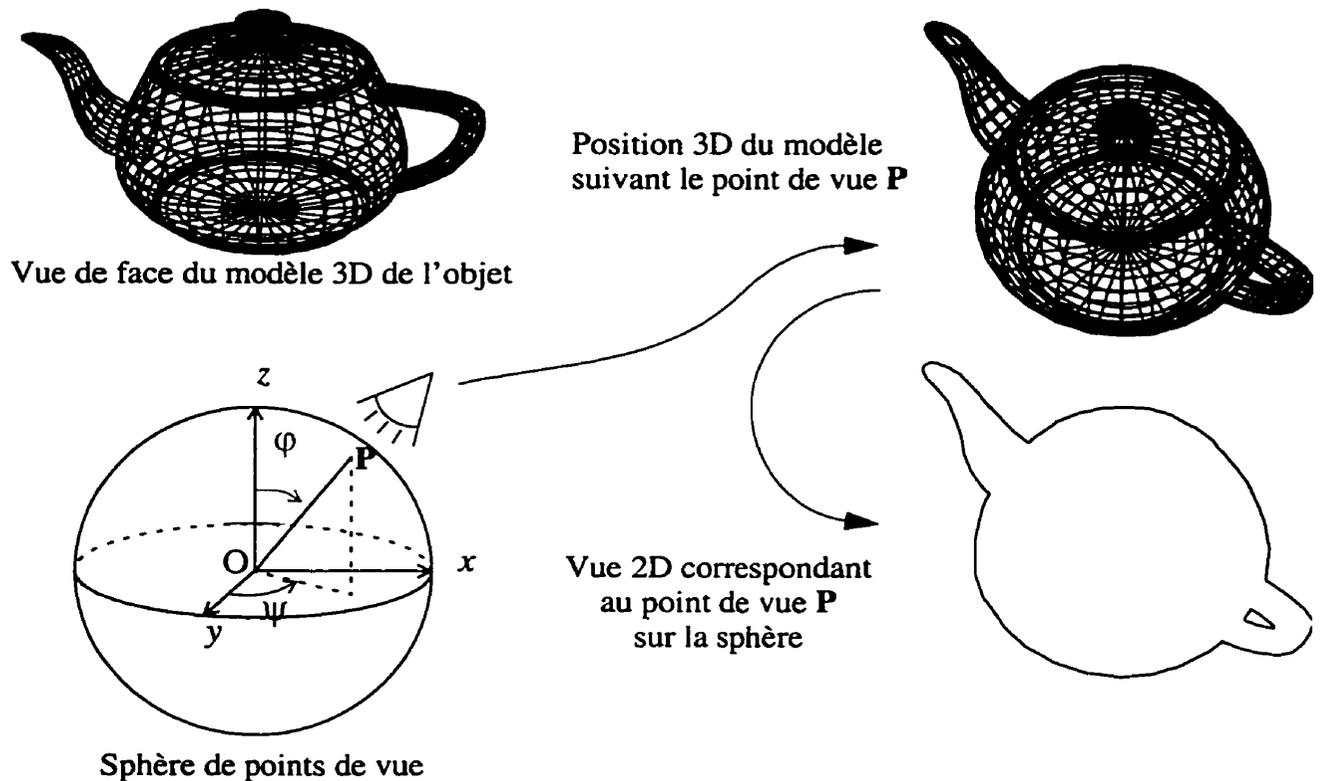


FIGURE 2.2 Relation entre la position dans la sphère de points de vue et les angles d'Euler de l'objet 3D.

plan image le long de la sphère de points de vue, on déplace l'objet 3D. Pour obtenir le point de vue $(\varphi, \psi) \in \Gamma_d$, on effectue une rotation d'angle $-\varphi$ selon l'axe Ox , suivie d'une rotation d'angle $-\psi$ selon l'axe Oz' (qui correspond à l'axe Oz après la première rotation) de l'objet 3D. Ces deux rotations correspondent à une modification de la pose de l'objet 3D selon ses deux premiers angles d'Euler. La figure 2.2 illustre le principe de création des différentes vues du graphe d'aspect tel que nous venons de le décrire.

Dans ce qui précède, nous avons omis de donner l'un des paramètres de pose indispensable: la distance entre l'objet et la caméra. Ce paramètre limite quelque peu la généralité du graphe d'aspect puisque celui-ci est alors défini en fonction d'un facteur d'échelle. Cependant, cette limitation importante des graphes d'aspect évoquée par Faugeras dans [30], n'en est pas une

dans notre cas puisque, grâce à la modélisation 2D que nous avons définie dans le chapitre précédent et que nous utiliserons pour chaque vue du graphe d'aspect, il sera possible de retrouver le facteur d'échelle entre un modèle 2D inconnu et une vue particulière du graphe. C'est là un des grands intérêts de notre modélisation. Elle permet en effet de s'affranchir du facteur d'échelle à l'intérieur des limitations de notre approche, dans le cas de la projection de perspective.

2.3.2 Quelques exemples de graphes d'aspect

Les figures 2.3 et 2.4 représentent les différentes vues des graphes d'aspect d'une théière et d'un avion. Les paramètres de discrétisation de la sphère de points de vue sont différents dans les deux cas. On peut constater que, même avec un maillage très limité, on obtient un grand nombre d'aspects.

Avec le type de représentation que nous avons choisi, on constate que les vues du dessus et du dessous correspondent au même contour, mais qu'elles ont subi une rotation par rapport à un axe perpendiculaire au plan image. Les N_ψ copies du même aspect, à une rotation 2D près, sont ainsi stockées dans la structure de graphe d'aspect. Même si la paramétrisation 2D que nous allons utiliser permet de retrouver la rotation 2D entre deux contours, nous conserverons ces différentes copies, plus pour des raisons de facilité de programmation que pour de réelles raisons scientifiques. Le maillage de la sphère de points de vue contiendra donc $(N_\phi + 1)N_\psi$ aspects distincts.

2.3.3 Modélisation de chacune des vues du graphe

La paramétrisation du contour d'occultation est définie à l'aide de trois vecteurs: le vecteur de paramétrisation noté V_p , le spectre d'amplitude noté A_p et enfin le spectre de phase noté P_p . La création d'un graphe d'aspect nécessite donc, en plus du calcul du contour d'occultation, le

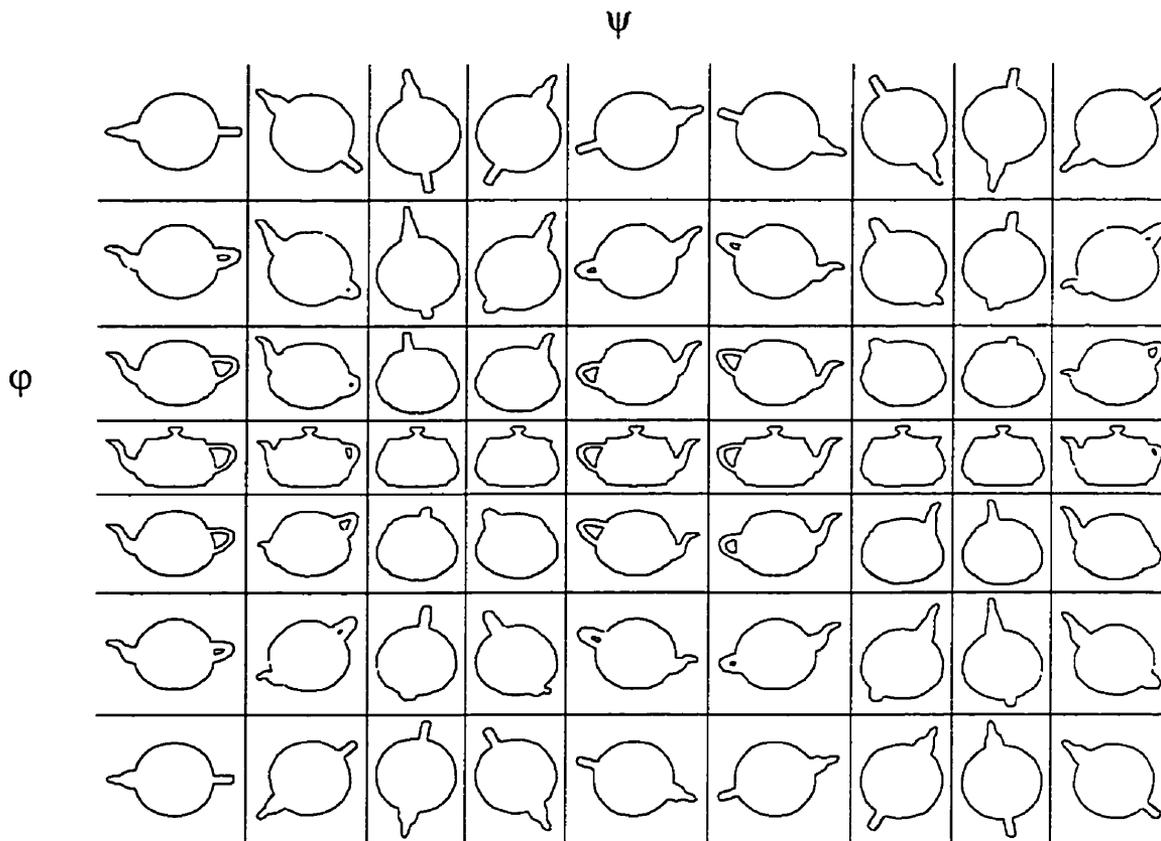


FIGURE 2.3 Graphe d'aspect de la théière

Cette figure représente les différentes vues du graphe d'aspect de la théière. Les paramètres N_ϕ et N_ψ ont été fixés à 6 et à 9 respectivement dans cet exemple. On obtient donc une partition avec un pas de 30° en latitude et un pas de 40° en longitude.

calcul des trois vecteurs de paramétrisation pour chacune des vues. Une autre manière de voir consisterait à construire un graphe d'aspect pour chacun des trois vecteurs.

2.3.4 Conclusion

Dans ce paragraphe, nous avons vu comment générer le graphe d'aspect d'un objet 3D. Ce graphe d'aspect possède deux paramètres N_ϕ et N_ψ qui définissent sa résolution, un paramètre qui définit la distance objet-caméra et deux paramètres Φ_s et N_s liés à la modélisation 2D des différentes vues 2D (voir chapitre 2). Chaque vue du graphe d'aspect est définie par un contour

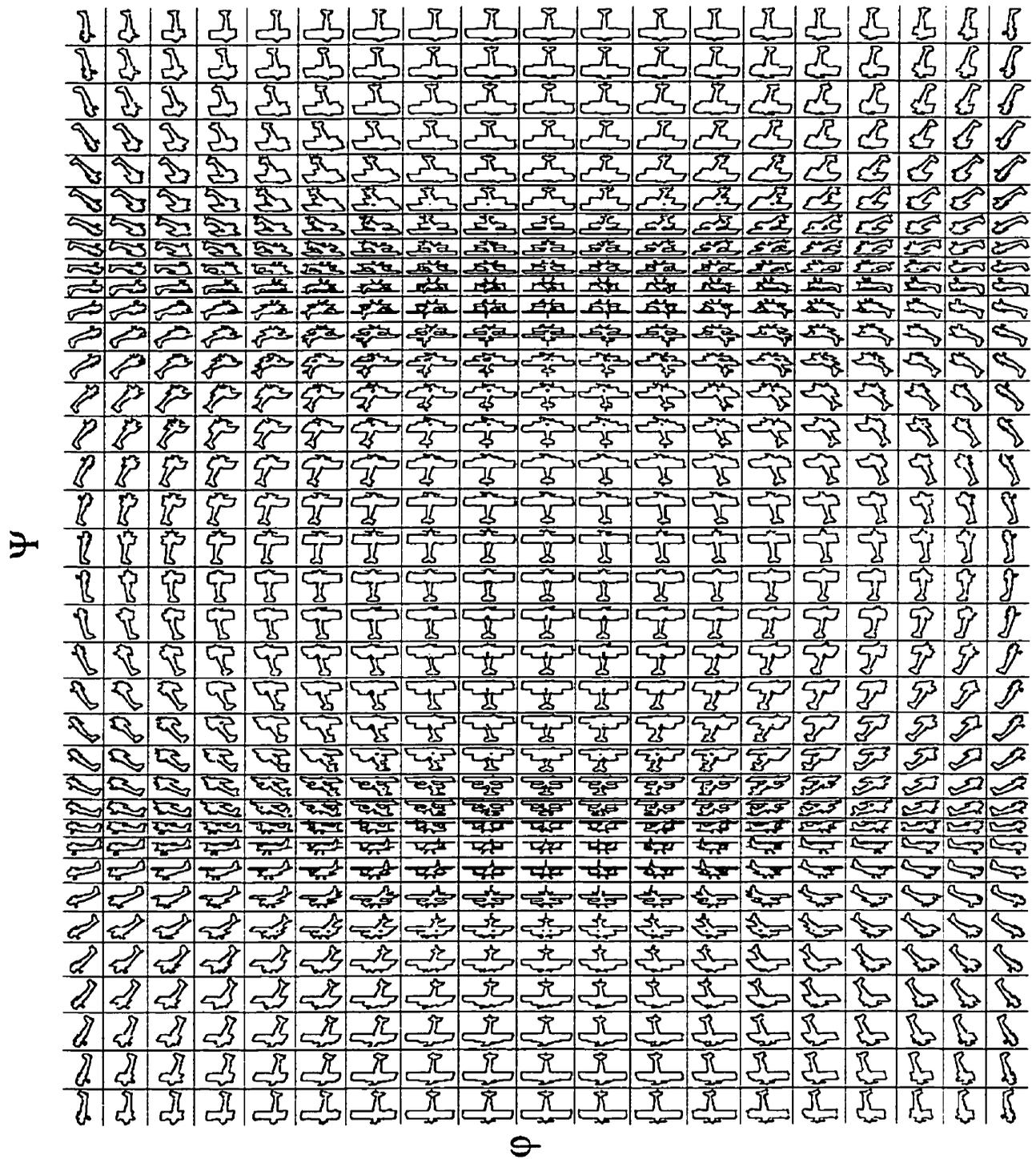


FIGURE 2.4 Graphe d'aspect de l'avion.

Les paramètres N_ϕ et N_ψ ont été fixés à 18 et 36 respectivement. On obtient donc une partition régulière avec un pas de 10° .

d'occultation, un vecteur de paramétrisation polaire, un spectre d'amplitude et un spectre de phase. Nous verrons dans les prochains paragraphes comment exploiter ces informations pour le calcul des paramètres de pose.

2.4 Graphe d'aspect et estimation de pose

2.4.1 Recherche exhaustive dans le graphe d'aspect complet

2.4.1.1 Description de l'algorithme

Dans le chapitre précédent, nous avons vu qu'il était possible de retrouver l'angle de rotation 2D ainsi que le facteur d'échelle existant entre deux contours d'occultation d'un même objet. Nous avons également pu constater qu'il était possible d'utiliser la mesure de comparaison que nous avons définie pour retrouver la pose 2D d'un contour parmi un ensemble de contours distincts (voir section 1.4.2.4). Reprenons le même principe avec les contours d'occultation du graphe d'aspect régulier d'un objet 3D. En recherchant dans cet ensemble le contour le plus proche, il sera possible non seulement de retrouver le facteur d'échelle et l'angle de rotation 2D, mais aussi les deux autres angles de rotation caractérisant la pose 3D d'un objet.

Par la suite, nous noterons $\{C_{n_\phi n_\psi} \mid (n_\phi, n_\psi) \in \Gamma_d\}$ l'ensemble des contours d'occultation du graphe d'aspect obtenu par un maillage régulier Γ_d de la sphère de points de vue. L'ensemble de ces contours correspond à des vues particulières de l'objet 3D. Pour toutes les vues, nous avons choisi une distance constante entre la caméra et l'objet 3D notée $Z_{\phi\psi}$. À chaque contour est associé son vecteur de paramétrisation $V_{n_\phi n_\psi}$ ainsi que les spectres d'amplitude $A_{n_\phi n_\psi}$ et de phase $P_{n_\phi n_\psi}$ correspondants.

On note C_x le contour d'occultation d'un objet 3D dans une position inconnue définie à partir de quatre paramètres de pose $(\phi_x, \psi_x, \alpha_x, z_x)$. Les 3 premiers paramètres représentent les 3 angles d'Euler d'un objet rigide dans l'espace. Le troisième de ces angles correspond par conséquent à la rotation 2D dans le plan image du contour d'occultation. Le quatrième paramètre

représente la distance entre l'objet 3D et la caméra. Rappelons que dans ce chapitre, nous ne nous intéresserons pas aux deux paramètres de pose translationnels qui manquent pour définir parfaitement la position d'un objet rigide dans l'espace. Tous les contours d'occultation correspondent donc à des vues pour lesquelles l'origine du repère de l'objet 3D se trouve sur le même axe perpendiculaire au plan image.

Pour retrouver les paramètres de pose du contour C_x , on utilise donc le graphe d'aspect:

$$(\varphi_x, \psi_x) = \underset{(n_\varphi, n_\psi) \in \Gamma_d}{\operatorname{argmin}} \delta(C_{n_\varphi, n_\psi}, C_x) \quad (\text{EQ 2.2})$$

De plus, si on note f_e^x le facteur d'échelle entre les contours C_x et $C_{\varphi_x \psi_x}$, nous aurons:

$$z_x = f_e^x \cdot Z_{\varphi \psi} \quad (\text{EQ 2.3})$$

Enfin, l'angle de rotation 2D entre les contours C_x et $C_{\varphi_x \psi_x}$ correspondra à α_x :

$$\alpha_x = \operatorname{Angle2D}(C_x, C_{\varphi_x \psi_x}) \quad (\text{EQ 2.4})$$

Le facteur d'échelle et l'angle de rotation 2D sont calculés en même temps que la mesure de comparaison $\delta(\)$ est évaluée.

La précision maximale que l'on peut obtenir pour les deux premiers angles d'Euler dépend du maillage retenu pour le graphe d'aspect. Ainsi, φ_x sera connu à π/N_φ radian près et ψ_x à $(2\pi)/N_\psi$ radian près. De même, la précision sur l'angle de rotation 2D, α_x correspondra au facteur de discrétisation angulaire Φ_s permettant de définir la paramétrisation.

La complexité de l'algorithme est facile à évaluer puisqu'il s'agit de calculer $(N_\varphi + 1)N_\psi$ mesures de comparaison. Le calcul d'une mesure de comparaison est une opération relativement simple dont la complexité dépend uniquement de la taille des vecteurs de paramétrisation et donc des facteurs Φ_s et N_s . La complexité croît linéairement par rapport au nombre d'aspects

présents dans le graphe. Il est ainsi possible d'effectuer la recherche exhaustive dans un laps de temps raisonnable et ce, même pour un nombre élevé d'aspects. De plus, l'algorithme est hautement parallélisable puisque chaque comparaison est indépendante de l'autre.

2.4.1.2 Résultats

Nous venons de voir qu'il est possible de retrouver les quatre paramètres de pose d'un objet 3D à partir d'une vue particulière de son contour d'occultation. Il s'agit maintenant de voir quelles sont les performances de cet algorithme. En effet, retrouver la pose avec une bonne précision peut s'avérer très délicat puisque deux contours voisins dans le graphe d'aspect peuvent être très ressemblants pour ne pas dire semblables. De plus, les effets de discrétisation engendrent toujours des perturbations qui risquent de biaiser l'estimation de la pose. Les effets de la discrétisation apparaissent ici dans plusieurs paramètres. Le premier paramètre de discrétisation est celui relié au maillage du graphe d'aspect, tandis que le second correspond au facteur de discrétisation angulaire de la paramétrisation des contours d'occultation. Enfin, il y a les effets incontournables de la discrétisation liés à la résolution en pixel du contour d'occultation lui-même. Ce dernier effet engendre des erreurs dans les coefficients du vecteur de paramétrisation lors d'une rotation ou d'un changement d'échelle d'un même objet.

L'ensemble de ces imperfections font que les performances de l'algorithme de calcul de pose ne sont pas parfaites. Nous allons cependant pouvoir constater qu'elles sont très bonnes et que la méthode proposée ouvre une voie très prometteuse.

2.4.1.2.1 Description des simulations

Afin d'évaluer les performances, un très grand nombre de simulations ont été effectuées à partir de différents objets 3D. Pour chaque objet, une large collection de poses a été générée aléatoirement. Ceci permet d'obtenir des résultats représentatifs.

Afin de bien voir l'impact des différents paramètres en jeu, les poses aléatoires ont été générées en fixant certains critères. Des simulations ont, par exemple, été effectuées avec la distance

caméra objet égale à la même valeur que celle utilisée dans le graphe d'aspect. Pour d'autres simulations, les premiers angles d'Euler ont été choisis uniquement parmi un ensemble de valeurs discrètes correspondant à la discrétisation de la sphère de points de vue.

Les objets utilisés pour les simulations sont aussi variés qu'une théière, un bateau, une voiture ou un poisson.

2.4.1.2.2 Évaluation des performances

Afin de pouvoir traiter toute l'information générée par ces simulations, nous avons défini un certain nombre de mesures de performance. Précisons ici qu'il est difficile d'évaluer les performances avec une seule valeur. En effet, un calcul de pose peut être considéré comme mauvais parce que les bons paramètres n'ont pas été trouvés, mais il peut aussi être considéré comme acceptable parce qu'avec les données accessibles (à savoir le contour d'occultation), il s'agit de la meilleure estimation de pose réalisable. L'utilisation de plusieurs mesures d'évaluation de performance semble donc indispensable. Voici celles que nous avons retenues.

La première de ces mesures est le pourcentage d'estimation de pose parfaite. Ce pourcentage correspond au nombre de simulations pour lesquelles les estimations des quatre paramètres de pose sont parfaites par rapport au nombre total de simulations. Par "estimation parfaite" nous entendons "estimation la meilleure possible compte tenu des différents facteurs de discrétisation". Ainsi, une estimation de 45 degrés avec un facteur de discrétisation de 5 degrés est considérée comme parfaite, même si la valeur réelle est de 47 degrés. Ce pourcentage est un indicateur de performance «brute». Il indique le pourcentage de réussite, mais ne permet pas de savoir si les échecs sont importants ou non. On lui associe un pourcentage d'estimation de pose «correcte» qui correspond à une bonne estimation des deux premiers angles d'Euler et une erreur sur l'angle de rotation 2D ou sur le facteur d'échelle. On donne également l'erreur moyenne faite sur le calcul de l'angle de rotation 2D et sur le facteur d'échelle. Le pourcentage d'estimation de pose correcte permet d'évaluer les performances du processus purement 2D,

c'est à dire du calcul du facteur d'échelle et de l'angle de rotation, lorsque le bon contour est sélectionné dans le graphe d'aspect.

La deuxième mesure de performance donne le pourcentage de réussite en tenant compte uniquement de la mesure de comparaison. Ce pourcentage correspond au nombre de simulations pour lesquelles la mesure de comparaison est inférieure à un seuil par rapport au nombre total de simulations. Cette mesure est calculée pour différentes valeurs de seuil et la valeur moyenne de la mesure de comparaison est donnée à titre indicatif. Ce pourcentage donne une bonne indication de la performance de l'algorithme en ne tenant compte que de la ressemblance entre un contour inconnu et le contour qui lui est associé.

La troisième mesure permet de connaître le rang du contour qui aurait dû être choisi dans le graphe d'aspect. Le rang est calculé à partir du classement des contours du graphe d'aspect en fonction de la mesure de comparaison. Ainsi, si l'estimation de pose est exacte, ce rang vaut 0. On donne cette mesure en pourcentage par rapport au nombre total de contours contenus dans le graphe d'aspect. Ce pourcentage permet d'avoir une bonne idée de l'efficacité de l'algorithme lorsque l'estimation de la pose n'est pas parfaite. Ainsi, s'il y a 50% de contours qui sont meilleurs que le contour idéal, on pourra conclure que l'algorithme ne permet pas de différencier convenablement les modèles entre eux et que l'estimation de pose sera plus un processus aléatoire qu'un calcul fiable. Par contre, si ce pourcentage reste faible, on pourra supposer que l'estimation de pose est relativement bonne et que les erreurs sont dues à la ressemblance locale des contours entre eux dans une partie du graphe d'aspect.

La quatrième mesure de performance donne l'erreur moyenne commise sur les quatre paramètres de pose pour l'ensemble des simulations. Dans le cas du facteur d'échelle, cette mesure permet de savoir si la distance caméra objet est bien estimée en moyenne. Il est difficile de donner une interprétation pour les trois autres paramètres de pose dans la mesure où des erreurs importantes sur les angles peuvent survenir. En effet, pour certains objets, une vue du dessus et une vue du dessous peuvent être très semblables.

Enfin, la cinquième mesure de performance répertorie les plus mauvaises performances de l'algorithme en terme de mesure de comparaison et en terme de justesse du contour choisi dans le graphe d'aspect. La représentation graphique du contour testé et du contour calculé est donnée. Cette mesure de performance permet ainsi de voir comment l'algorithme fonctionne lorsqu'il donne de mauvais résultats et d'en comprendre les causes.

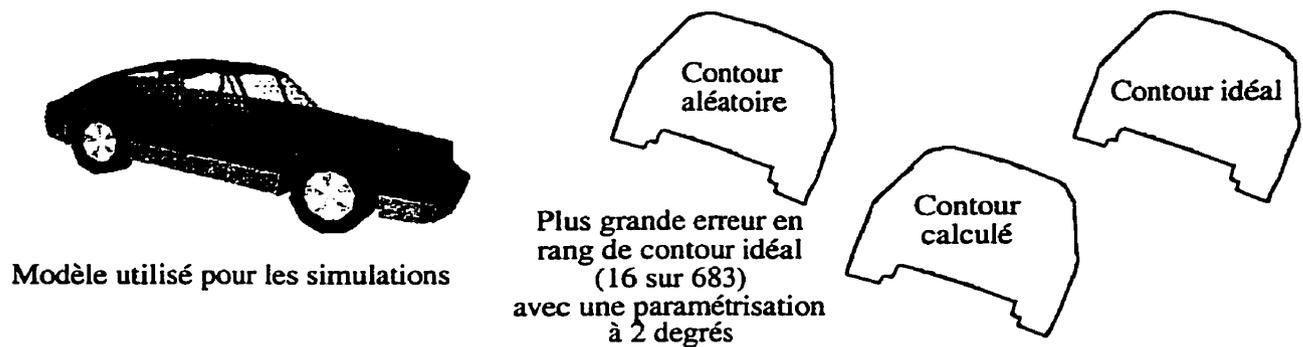
2.4.1.2.3 Simulations

Toutes les simulations ont été effectuées sur des stations unix sparc 10. La durée moyenne de calcul représente par conséquent un bon moyen de comparer la complexité de l'algorithme en fonction de la taille des graphes d'aspect ou du type de paramétrisation utilisée. Aucune optimisation n'a été faite tant au niveau de l'algorithme que de la programmation proprement dite.

Une évaluation de l'effet du facteur d'échelle, de l'angle de rotation 2D et des 2 premiers angles d'Euler a été faite indépendamment des autres paramètres de pose. Des simulations qui faisaient varier les quatre paramètres de pose en même temps ont aussi été faites.

Différents paramètres intrinsèques de l'algorithme ont aussi été testés. Ainsi, différents maillages de la sphère de points de vue ont été utilisés ainsi que des valeurs différentes du facteur de discrétisation angulaire.

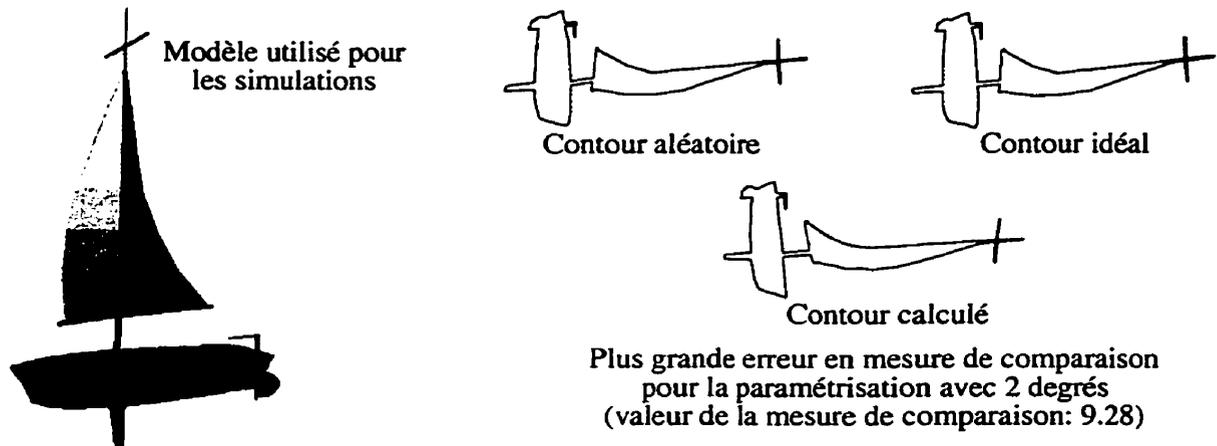
Pour chaque série de paramètres évalués, cent contours ont été générés aléatoirement. Les statistiques résultant de ces simulations sont donc fiables. De plus, différents objets ont été utilisés afin de conserver leur généralité aux conclusions qui pourront être tirées de ces tests. Toutes les simulations pour lesquelles l'objet reste le même ont été effectuées avec la même série de poses aléatoires. Ceci permet de bien évaluer la performance d'un paramètre en particulier puisque l'ensemble de test reste le même.



| Nombre de simulations | Durée moyenne du calcul pour une estimation de pose en secondes | Pourcentage d'estimation de pose parfaite | Pourcentage d'estimation de pose correcte | Erreur moyenne absolue sur le facteur d'échelle | Erreur moyenne absolue sur l'angle de rotation 2D. |
|---|---|---|---|---|--|
| 100 | 173.20 | 88% | 95% | 0.0045 | 0.14 degrés |
| 100 | 27.36 | 95% | 95% | 0.0051 | 0 degré |
| Mesure de comparaison moyenne | Pourcentage de modèles ayant une mesure de comparaison inférieure à 1 | Pourcentage de modèles ayant une mesure de comparaison inférieure à 1.5 | Pourcentage de modèles ayant une mesure de comparaison inférieure à 2 | Rang moyen du contour idéal dans le graphe d'aspect | Rang moyen du contour idéal dans le graphe d'aspect en pourcentage |
| 0.8031 | 78% | 97% | 100% | 0.22 (683) | 0.03% |
| 0.8190 | 73% | 98% | 100% | 0.21 (683) | 0.03% |
| Erreur moyenne absolue sur le facteur d'échelle | Erreur moyenne absolue sur le premier angle d'Euler | Erreur moyenne absolue sur le deuxième angle d'Euler | Erreur moyenne absolue sur l'angle de rotation 2D | Type de simulation | |
| 0.0049 | 3 degrés | 1.7 degrés | 4.02 degrés | 2 degrés | |
| 0.0055 | 2.1 degrés | 1.8 degrés | 3.2 degrés | 5 degrés | |

FIGURE 2.5 Résultats d'estimation de pose avec facteur d'échelle variable.

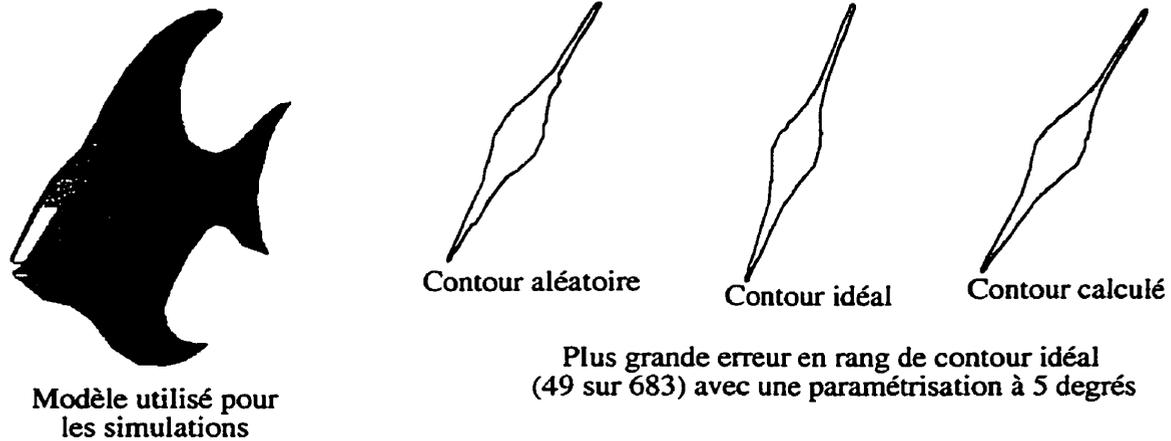
Dans cette figure, nous présentons les résultats obtenus pour la voiture. Pour les simulations, seule la distance entre l'objet et la caméra est testée. En effet, les deux premiers angles d'Euler ne peuvent prendre que des valeurs correspondant au maillage discret de la sphère de points de vue et le troisième angle d'Euler a été laissé fixe à 0 degré. Ces simulations permettent de voir la précision que l'on peut atteindre sur le facteur d'échelle. Deux séries de tests ont été effectuées. La première avec un facteur de discrétisation angulaire de 2 degrés et la seconde avec 5 degrés (Φ_s). Dans les deux cas, le maillage de la sphère de points de vue était de 10 degrés dans les deux axes, soient $N_\phi = 18$ et $N_\psi = 36$. La distance caméra objet varie entre 30 et 100 et le graphe d'aspect a été calculé avec une distance de 50.



| Nombre de simulations | Durée moyenne du calcul pour une estimation de pose en secondes | Pourcentage d'estimation de pose parfaite | Pourcentage d'estimation de pose correcte | Erreur moyenne absolue sur le facteur d'échelle | Erreur moyenne absolue sur l'angle de rotation 2D |
|---|---|---|---|---|--|
| 100 | 239.54 | 50% | 78% | 0.0119 | 0.76 degrés |
| 100 | 36.56 | 70% | 90% | 0.0116 | 1.5 degrés |
| Mesure de comparaison moyenne | Pourcentage de modèles ayant une mesure de comparaison inférieure à 1 | Pourcentage de modèles ayant une mesure de comparaison inférieure à 1.5 | Pourcentage de modèles ayant une mesure de comparaison inférieure à 2 | Rang moyen du contour idéal dans le graphe d'aspect | Rang moyen du contour idéal dans le graphe d'aspect en pourcentage |
| 2.21 | 8% | 30% | 54% | 4.47 (683) | 0.65% |
| 2.72 | 3% | 9% | 25% | 6.16 (683) | 0.90% |
| Erreur moyenne absolue sur le facteur d'échelle | Erreur moyenne absolue sur le premier angle d'Euler | Erreur moyenne absolue sur le deuxième angle d'Euler | Erreur moyenne absolue sur l'angle de rotation 2D | Type de simulation | |
| 0.0181 | 8.1 degrés | 0.2 degré | 6.72 degrés | 2 degrés | |
| 0.172 | 6.4 degrés | 0.1 degré | 6.06 degrés | 5 degrés | |

FIGURE 2.6 Résultats d'estimation de pose avec angle de rotation 2D aléatoire.

Voici les résultats obtenus pour le bateau. Pour les simulations, seule l'angle de rotation 2D est testé. En effet, les deux premiers angles d'Euler ne peuvent prendre que des valeurs correspondant au maillage discret de la sphère de points de vue et la distance caméra objet est égale à celle utilisée pour calculer le graphe d'aspect, soit 600. Deux séries de tests ont été effectuées. La première avec un facteur de discrétisation angulaire de 2 degrés et la seconde avec 5 degrés (Φ_s). Dans les deux cas, le maillage de la sphère de points de vue était de 10 degrés dans les deux axes, soient $N_\phi = 18$ et $N_\psi = 36$.



| Nombre de simulations | Durée moyenne du calcul pour une estimation de pose en secondes | Pourcentage d'estimation de pose parfaite | Pourcentage d'estimation de pose correcte | Erreur moyenne absolue sur le facteur d'échelle | Erreur moyenne absolue sur l'angle de rotation 2D |
|---|---|---|---|---|--|
| 100 | 200.69 | 46% | 77% | 0.0242 | 0.98 degrés |
| 100 | 31.8 | 61% | 70% | 0.0256 | 0.5 degré |
| Mesure de comparaison moyenne | Pourcentage de modèles ayant une mesure de comparaison inférieure à 1 | Pourcentage de modèles ayant une mesure de comparaison inférieure à 1.5 | Pourcentage de modèles ayant une mesure de comparaison inférieure à 2 | Rang moyen du contour idéal dans le graphe d'aspect | Rang moyen du contour idéal dans le graphe d'aspect en pourcentage |
| 2.58 | 6% | 15% | 26% | 2.09 (683) | 0.31% |
| 2.84 | 5% | 10% | 21% | 2.59 (683) | 0.38% |
| Erreur moyenne absolue sur le facteur d'échelle | Erreur moyenne absolue sur le premier angle d'Euler | Erreur moyenne absolue sur le deuxième angle d'Euler | Erreur moyenne absolue sur l'angle de rotation 2D | Type de simulation | |
| 0.0282 | 8.32 degrés | 10.49 degrés | 4.66 degrés | 2 degrés | |
| 0.0319 | 8.64 degrés | 9.01 degrés | 4.05 degrés | 5 degrés | |

FIGURE 2.7 Résultats d'estimation de pose avec facteur d'échelle et angle de rotation 2D fixes.

Voici les résultats obtenus pour le poisson. Pour les simulations, la distance entre l'objet et la caméra et l'angle de rotation 2D sont fixés aux mêmes valeurs que celles utilisées pour calculer le graphe d'aspect (700 pour la distance et 0 pour l'angle). Ces simulations permettent de tester la précision de l'algorithme sur les deux premiers angles d'Euler. Deux séries de tests ont été effectuées. La première avec un facteur de discrétisation angulaire de 2 degrés et la seconde avec 5 degrés (Φ_5). Dans les deux cas, le maillage de la sphère de points de vue était de 10 degrés dans les deux axes, soient $N_\phi = 18$ et $N_\psi = 36$.

| Nombre de simulations | Durée moyenne du calcul pour une estimation de pose en secondes | Pourcentage d'estimation de pose parfaite | Pourcentage d'estimation de pose correcte | Erreur moyenne absolue sur le facteur d'échelle | Erreur moyenne absolue sur l'angle de rotation 2D |
|---|---|---|---|--|--|
| 100 | 11.9 | 13% | 33% | 0.0108 | 14.84 degrés |
| 100 | 22.2 | 18% | 30% | 0.0105 | 2.20 degrés |
| 100 | 121.6 | 14% | 17% | 0.0110 | 1.41 degrés |
| 100 | 68.3 | 5% | 37% | 0.0097 | 22.94 degrés |
| 100 | 136.3 | 10% | 36% | 0.0100 | 2.25 degrés |
| 100 | 642.8 | 9% | 20% | 0.0104 | 1.45 degrés |
| Mesure de comparaison moyenne | Pourcentage de modèles ayant une mesure de comparaison inférieure à 1 | Pourcentage de modèles ayant une mesure de comparaison inférieure à 1.5 | Pourcentage de modèles ayant une mesure de comparaison inférieure à 2 | Rang moyen du contour idéal dans le graphe d'aspect | Rang moyen du contour idéal dans le graphe d'aspect en pourcentage |
| 1.15 | 40% | 82% | 96% | 20.67 (342) | 6.04% |
| 1.05 | 48% | 89% | 98% | 40.01 (684) | 5.84% |
| 0.90 | 66% | 93% | 99% | 63.66 (2663) | 2.39% |
| 1.08 | 50% | 85% | 95% | 15.68 (342) | 4.58% |
| 0.97 | 68% | 89% | 98% | 29.98 (684) | 4.38% |
| 0.80 | 81% | 95% | 100% | 32.18 (2663) | 1.21% |
| Erreur moyenne absolue sur le facteur d'échelle | Erreur moyenne absolue sur le premier angle d'Euler | Erreur moyenne absolue sur le deuxième angle d'Euler | Erreur moyenne absolue sur l'angle de rotation 2D | Modèle utilisé pour les simulations  | |
| 0.0201 | 42.24 degrés | 22.13 degrés | 19.38 degrés | | |
| 0.0192 | 44.62 degrés | 22.01 degrés | 19.57 degrés | | |
| 0.0163 | 39.86 degrés | 18.21 degrés | 14.00 degrés | | |
| 0.0198 | 46.36 degrés | 24.47 degrés | 16.52 degrés | | |
| 0.0170 | 44.46 degrés | 18.53 degrés | 16.06 degrés | | |
| 0.0147 | 36.68 degrés | 14.26 degrés | 11.44 degrés | | |

FIGURE 2.8 Résultats d'estimation de pose entièrement aléatoire.

Voici les résultats obtenus avec la théière. Les simulations sont effectuées en générant les 4 paramètres de pose, de manière complètement aléatoire. La distance caméra objet varie entre 20 et 70 et le graphe d'aspect a été calculé avec une distance de 30. Six séries de tests ont été effectuées. Trois types de maillage ont été utilisés: $N_\phi = 18$ $N_\psi = 18$, $N_\phi = 18$ $N_\psi = 36$ et $N_\phi = 36$ $N_\psi = 72$. Pour chacun de ces maillages, deux facteurs de discrétisation angulaire ont été utilisés. L'un est de 5 degrés et le second de 2 degrés (Φ_s). Les trois premières lignes correspondent à 5 degrés, les trois autres à 2 degrés et les maillages vont du plus petit au plus grand dans les deux cas.

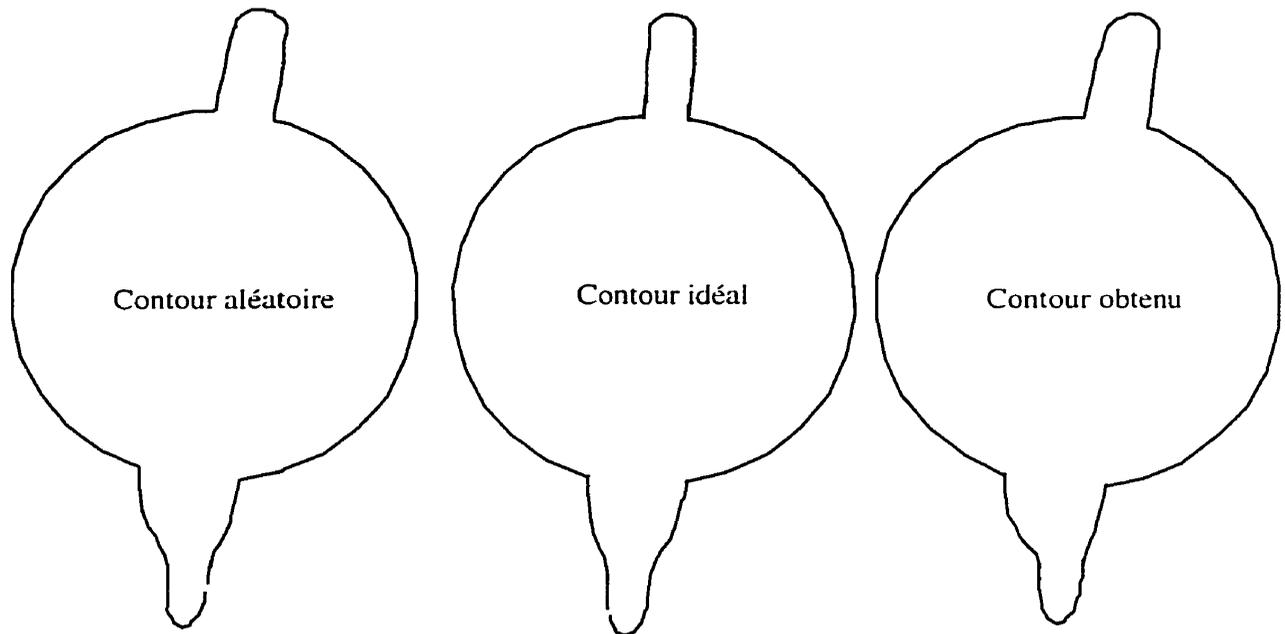


FIGURE 2.9 Plus grandes erreurs lors des simulations d'estimation de pose de la théière.

Cette figure représente les contours pour lesquels l'estimation de la pose a été la plus difficile. Les résultats sont donnés pour un maillage avec $N_\phi = 18$ et $N_\psi = 18$ ainsi que pour un facteur de discrétisation de 2 degrés pour la paramétrisation. Le résultat présenté correspond au pire cas en terme de rang de contour idéal (266 sur 342). On peut constater que le résultat semble mauvais en terme de mesure de performance alors que l'erreur provient du maillage trop peu dense de la sphère de points de vue.

Les figures 2.5 à 2.9 illustrent les résultats obtenus pour les différentes simulations. D'autres simulations ont été effectuées avec d'autres objets et confirment ces résultats. L'analyse des résultats est faite dans le paragraphe suivant.

2.4.1.2.4 Analyse

L'interprétation des résultats est quelque peu complexe à cause des différents paramètres qui ont été mesurés. Nous ne commenterons pas chaque mesure de performance individuellement, mais

nous tirerons les conclusions générales qui apparaissent à la lueur des différents résultats obtenus.

Le premier point important est que qualitativement la méthode est très bonne puisque, si on se fie aux figures des prédictions les plus mauvaises, on différencie difficilement le contour qui aurait effectivement dû être choisi de celui qui l'a été. De plus, comme le montre la figure 2.9, les mauvaises performances sont davantage liées au maillage discret de la sphère de points de vue qu'à l'approche en tant que telle.

L'une des mesures offrant la meilleure évaluation des performances est celle correspondant au rang moyen du bon modèle dans le graphe d'aspect. Cette mesure reste toujours sous la barre des 1% lorsque les différents paramètres de pose sont évalués individuellement et ce, quel que soit l'objet. Lorsque les paramètres de pose sont évalués tous ensemble, ce pourcentage est au pire de 6% avec un graphe d'aspect peu dense et un facteur de discrétisation angulaire de 5 degrés. Par contre, dès que le graphe d'aspect est plus dense et que la paramétrisation des contours est plus fine, ce pourcentage diminue pour atteindre 1.21% avec un maillage de 5 degrés et un angle de paramétrisation de 2 degrés.

L'intérêt de ce pourcentage est énorme. En effet, notre méthode permet de trouver un ensemble de pose initiale très proche de la solution. Avec un algorithme de raffinement de la pose basée sur d'autres critères que le seul contour d'occultation (voir section 3.2.3.2). Il devient possible d'améliorer encore l'estimation. Il suffit en effet de prendre les poses correspondant aux 1 ou 2% des meilleurs contours dans le graphe d'aspect. On est assuré, en moyenne, que la bonne pose fait partie de cet ensemble d'estimation de pose initiale et qu'il permettra d'aboutir au calcul exact de la pose.

En revanche, on peut constater que l'évaluation des performances basées sur la mesure de comparaison est plus difficile à interpréter de manière objective. En effet, la mesure de comparaison dépend fortement du type d'objets traités. Définir un seuil unique permettant de dire si l'estimation de pose est bonne ou non est relativement difficile. L'utilisation de la mesure de comparai-

son moyenne pour l'ensemble des simulations donne une assez bonne indication de la valeur du seuil qui devrait être utilisée pour chaque objet.

On peut constater que, de manière générale, les erreurs sur le calcul du facteur d'échelle sont toujours assez faibles. Elles ne dépassent jamais les 2% d'erreur sur l'ensemble des simulations. Lorsque le facteur d'échelle est la seule inconnue, ce pourcentage tombe d'ailleurs à 0.5%, ce qui doit simplement correspondre à l'erreur liée à la discrétisation des contours d'occultation en terme de pixels.

Il est plus difficile d'interpréter les résultats obtenus pour les angles d'Euler. À cause de certaines symétries de l'objet observé, deux poses très différentes peuvent être aussi acceptables l'une que l'autre. De la même manière, les erreurs de discrétisation rendent difficiles la distinction des deux poses suivantes: $(120^\circ, 90^\circ, 10^\circ)$ et $(120^\circ, 80^\circ, 20^\circ)$.

On peut toutefois utiliser ces résultats pour montrer que plus on augmente le maillage dans le graphe d'aspect et plus on diminue le facteur de discrétisation angulaire, plus on améliore les performances de l'algorithme. En effet, il est évident que dans la figure 2.8, l'erreur moyenne en rotation selon les trois angles diminue lorsque ces paramètres changent. Cette remarque entraîne une autre concernant le pouvoir de différenciation de la mesure de comparaison. En augmentant le maillage et en diminuant l'angle de discrétisation, la mesure de comparaison distingue des contours de plus en plus ressemblants.

Le maillage de la sphère de points de vue doit être choisi adéquatement. En effet, s'il est trop grand (N_φ et N_ψ trop petits) certains aspects importants de l'objet seront oubliés dans le graphe d'aspect et l'estimation de la pose sera erronée, par contre le choisir trop petit (N_φ et N_ψ trop grands) surchargera inutilement le graphe d'aspect avec des vues si ressemblantes que l'algorithme sera incapable de les différencier. Le choix du maillage relève donc essentiellement d'un choix arbitraire qui devra être validé par un humain pour chaque objet.

En ce qui concerne la durée du calcul, on constate qu'elle est directement proportionnelle au nombre de contours dans le graphe d'aspect et qu'elle dépend aussi largement du facteur de discrétisation angulaire. La complexité de l'algorithme n'est pas trop énorme, même avec de gros graphes d'aspect, d'autant plus qu'aucune optimisation n'a été implantée et que l'algorithme est hautement parallélisable. Le prochain paragraphe proposera toutefois une alternative à cette méthode de recherche exhaustive dans le graphe d'aspect complet et en améliorera le temps de calcul.

2.4.1.3 Conclusion

Après avoir analysé les résultats précédents, nous pouvons conclure que la méthode d'estimation de pose est performante. À notre connaissance, aucune autre approche ne permet de retrouver les quatre paramètres de pose d'un objet 3D à partir de son contour d'occultation sans avoir aucune information préliminaire sur la pose. Dans le paragraphe suivant, nous allons améliorer le temps de calcul.

2.4.2 Analyse par composantes principales

Nous avons vu qu'il était possible de calculer avec une bonne précision la pose d'un objet 3D à partir de son contour d'occultation et de son graphe d'aspect. Ce calcul nécessite cependant qu'une recherche exhaustive dans le graphe d'aspect soit effectuée. Or, pour chaque contour du graphe d'aspect, il faut traiter des vecteurs de tailles assez importantes surtout lorsque le facteur de discrétisation angulaire devient relativement petit. Même si ce traitement peut se faire dans une durée acceptable et que l'algorithme peut être optimisé et parallélisé, il semble possible d'utiliser la redondance d'informations contenues dans les vecteurs de paramétrisation pour diminuer la complexité de l'algorithme de recherche exhaustive.

L'utilisation d'une analyse par composantes principales sur l'ensemble des contours du graphe d'aspect est la solution idéale. Elle permet de ramener un vecteur de paramétrisation de plusieurs centaines de composantes à un vecteur n'en ayant que quelques unes. La perte d'informa-

tion de cette approche est minimale si on se fie aux résultats obtenus et le vecteur de petite taille semble bel et bien contenir toute l'information pertinente. Dans ce paragraphe, nous allons présenter l'analyse par composantes principales et comparer les résultats avec ceux obtenus lors de la recherche exhaustive.

2.4.2.1 Principe de base

L'analyse par composantes principales est une technique fréquemment utilisée pour diminuer le nombre de paramètres décrivant un ensemble d'objets semblables. Pour utiliser ce principe, il faut que chaque objet soit représenté par un vecteur qui le caractérise. La représentation vectorielle des objets doit aussi tenir compte de la structure topologique naturellement induite par tout espace vectoriel. Ainsi, deux objets semblables devront être représentés par deux vecteurs proches au sens de la norme euclidienne.

Si la représentation vectorielle viole ce principe, alors l'analyse par composantes principales (ACP) ne donne aucun résultat. En effet, une ACP n'est rien de plus qu'une projection dans un sous-espace vectoriel. Cette projection est faite de manière à minimiser les erreurs commises en remplaçant les variations des vecteurs autour du vecteur moyen par leur projection. Exprimons ce principe de manière mathématique. Notons $\{M_i | i \in I\} \subset \mathbb{R}^n$ l'ensemble des vecteurs de dimension n représentant un ensemble d'objets et $E_k \subset \mathbb{R}^n$ un sous-espace vectoriel de dimension $k < n$. Le but d'une ACP est de trouver le sous-espace de projection E_k en minimisant l'erreur de projection. Cette erreur s'exprime de la manière suivante:

$$\varepsilon(E_k) = \sum_{i \in I} d^2(M_i - \bar{M}, E_k) \quad (\text{EQ 2.5})$$

\bar{M} correspond à la moyenne des M_i et $d(\cdot, E_k)$ représente la distance euclidienne entre un vecteur quelconque et le sous-espace vectoriel E_k . On suppose que les vecteurs $\{U_j | j \in \llbracket k+1, n \rrbracket\}$ représentent les coordonnées de la base E_k^\perp (sous-espace vectoriel orthogonal à E_k) dans la base canonique de \mathbb{R}^n . On définit aussi la matrice de covariance associée aux vecteurs M_i :

$$\Sigma = \frac{1}{\text{Card}(I)} \sum_{i \in I} (M_i - \bar{M})(M_i - \bar{M})^t \quad (\text{EQ 2.6})$$

On peut exprimer l'erreur de projection en fonction de la matrice de covariance:

$$\varepsilon(E_k) = \sum_{j=k+1}^n U_j^t \Sigma U_j \quad (\text{EQ 2.7})$$

Le minimum de $\varepsilon(E_k)$ est atteint lorsque E_k est le sous-espace vectoriel engendré par les vecteurs propres associés au k plus grandes valeurs propres. En effet, comme la matrice Σ est réelle et symétrique, elle est diagonalisable et ses valeurs propres sont positives. Notons $\{\lambda_j | j \in \llbracket 1, n \rrbracket\}$ l'ensemble des valeurs propres de Σ classées dans l'ordre décroissant. Nous aurons alors:

$$\varepsilon(E_k) = \sum_{j=k+1}^n \lambda_j \quad (\text{EQ 2.8})$$

Le choix de la valeur de k se fait en fonction des valeurs propres λ_j . De manière générale, on élimine les composantes dès que 99% de l'énergie est atteinte:

$$k = \underset{j}{\operatorname{argmin}} \left(\frac{\sum_{i=1}^j \lambda_i}{\sum_{i=1}^n \lambda_i} > 0,99 \right) \quad (\text{EQ 2.9})$$

Les applications de cette approche sont très nombreuses. Il est souvent possible de ramener une représentation d'une centaine de composantes à seulement 2 ou 3 composantes significatives. Ces composantes significatives sont des combinaisons linéaires des composantes de base. Plus l'ensemble des vecteurs $M_i - \bar{M}$ sur lequel on effectue une ACP possède de corrélation, plus l'énergie sera concentrée dans les premières valeurs propres de Σ .

2.4.2.2 Alignement des modèles

Nous venons de voir le principe général de l'analyse par composantes principales. Dans le cas particulier des contours d'un graphe d'aspect, les vecteurs permettant de faire l'analyse sont les vecteurs de paramétrisation. L'objectif de l'ACP est de trouver la matrice de projection ramenant les vecteurs de paramétrisation dans un sous-espace tel que l'erreur faite en remplaçant le vecteur par sa projection soit inférieure à 1% en moyenne. On note N_i la projection des vecteurs M_i . N_i comportera moins de composantes que M_i . La matrice de projection peut être utilisée pour projeter le vecteur de paramétrisation de n'importe quel contour inconnu dans ce sous-espace vectoriel. En recherchant le vecteur N_i le plus proche de la projection du contour inconnu, on pourra retrouver les paramètres de pose. Cette recherche se fait très rapidement en terme de temps de calcul puisque les vecteurs N_i sont de faibles dimensions.

Pour que l'approche précédente fonctionne adéquatement, il est indispensable que l'ensemble permettant de trouver la matrice de projection soit représentatif de tous les contours recherchés. En effet, un contour n'ayant pas le même genre de caractéristiques que celles des vecteurs de base ne pourra pas être remplacé par sa projection dans le sous-espace des composantes principales sans perte d'information. Effectuer une recherche dans ce sous-espace n'a alors plus de sens puisque deux vecteurs proches en apparence ne le seront plus si on considère l'espace vectoriel au complet.

En pratique, il faudrait donc que l'ensemble de base pour l'ACP soit composé non seulement de l'ensemble des contours du graphe d'aspect, mais aussi des rotations 2D et des homothéties de chacun de ces contours. Il s'agit là d'un ensemble immense de contours et cette approche exhaustive ne peut pas être envisagée comme une solution élégante ou même réalisable. Afin de ramener l'ensemble des contours de base à une dimension raisonnable, nous allons donc proposer d'aligner tous les contours. Cette procédure d'alignement assure que la projection d'un contour inconnu dans le sous-espace des composantes principales est efficace.

2.4.2.2.1 Critères d'alignement

La procédure d'alignement devra ramener tout vecteur de paramétrisation d'un contour ayant subi une rotation 2D ou un changement d'échelle à une forme de base. Il faut donc trouver un moyen d'évaluer la rotation 2D et le facteur d'échelle de manière absolue.

L'approche que nous avons retenue pour le calcul du facteur d'échelle consiste à ramener n'importe quel contour à l'échelle d'un cercle de rayon connu. Le calcul du facteur d'échelle se résume alors au calcul de la moyenne des composantes du vecteur de paramétrisation. En effet, si on note R le rayon du cercle de base, V_p le vecteur de paramétrisation d'un contour quelconque et N la taille du vecteur de paramétrisation, nous aurons:

$$f_e = \frac{NR}{\sum_{k=0}^{N-1} V_p(k)} \quad (\text{EQ 2.10})$$

L'approche permettant de ramener n'importe quel contour à une rotation 2D de base est un peu plus complexe. Nous avons pris pour acquis que le point le plus remarquable d'un contour d'occultation est le point dont le rayon est maximum. Ramener tous les contours de manière à ce que ce point de rayon maximum ait toujours une coordonnée angulaire nulle semble une approche raisonnable et susceptible d'être assez robuste pour nos besoins.

Pour trouver le point de rayon maximum, nous avons mis le vecteur de paramétrisation sous une forme pour laquelle le passage d'une composante à la suivante correspond à un changement d'angle. Rappelons que, dans sa forme de base, le vecteur de paramétrisation contient une succession de N_s valeurs pour chaque secteur angulaire (voir chapitre 2). Ces différentes valeurs permettent de tenir compte du fait que la paramétrisation angulaire d'un contour d'occultation n'est pas une fonction de l'angle, mais seulement une relation. En effet, pour un angle donné on peut avoir plusieurs rayons qui y sont associés. Afin de trouver le point de rayon maximum, il faut donc ramener le vecteur de paramétrisation à une représentation de N_s courbes successives de contours. Pour certains objets simples, toutes ces courbes sont identiques. Pour d'autres,

elles sont distinctes. Ces courbes correspondent à l'expression du rayon en fonction de l'angle. Rechercher le maximum est donc très simple:

$$\alpha_{max} = \Theta_s \underset{k \in \llbracket 1, N/N_s \rrbracket}{argmax} V_p(kN_s) \quad (\text{EQ 2.11})$$

Le calcul du facteur d'échelle et de la rotation 2D absolus permettent de ramener le vecteur de paramétrisation de n'importe quel contour à une forme de base. Comme nous le verrons, cette forme de base est relativement insensible aux rotations 2D et aux changements d'échelle. Nous pourrions par conséquent remplacer le vecteur de paramétrisation V_p par le vecteur aligné suivant:

$$V_{alig}(k) = f_e \cdot V((k - N_s \cdot \alpha_{max} / \Theta_s) \% N) \quad (\text{EQ 2.12})$$

Cette méthode est beaucoup plus simple que l'approche que nous utilisons jusqu'alors pour trouver un angle de rotation 2D ou un facteur d'échelle. Cette procédure d'alignement n'est cependant pas parfaitement robuste. Afin qu'elle le devienne, nous avons ajouté un facteur d'incertitude sur le calcul de l'angle correspondant au maximum de la courbe. Au lieu de ne tenir compte que d'un seul angle, nous avons défini un ensemble d'angles possibles. Ceux-ci correspondent à des maxima locaux de la courbe de paramétrisation dont la valeur atteint au moins 80% du maximum. La figure 2.10 illustre ce principe. On peut constater que lorsque le facteur d'échelle change, le maximum de la courbe de paramétrisation peut ne plus correspondre à la même portion de contour. Mais, en ajoutant un facteur d'incertitude, l'ensemble des angles associés aux extrema suffisamment élevés reste quant à lui stable.

2.4.2.2.2 Description de l'algorithme

Grâce à la méthode d'alignement des vecteurs de paramétrisation, il est possible d'utiliser une approche par ACP. En effet, on peut définir l'ensemble de base de l'ACP à partir des vecteurs de paramétrisation alignés du graphe d'aspect. L'alignement du facteur d'échelle se fait par rapport à un cercle dont le rayon est la moyenne de toutes les composantes de tous les vecteurs du gra-

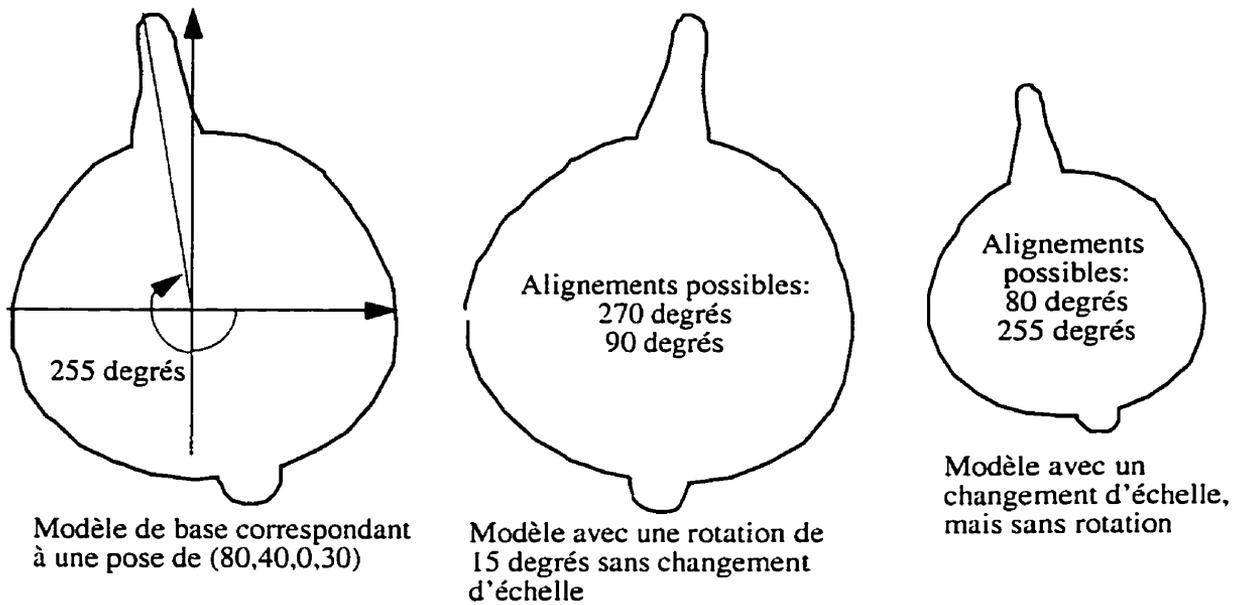


FIGURE 2.10 Calcul de l'alignement en rotation 2D.

Dans cette figure, nous illustrons le principe d'alignement en rotation décrit dans le paragraphe 2.4.2.2. Le premier graphique sur la gauche correspond à l'un des contours du graphe d'aspect de la théière. Le maximum de son vecteur de paramétrisation est obtenu pour 255 degrés. Le second graphique représente le même contour, mais avec une rotation 2D de 15 degrés. En utilisant l'algorithme d'alignement, on obtient deux rotations possibles. L'une est de 270 degrés (celle correspondant au maximum effectif) et l'autre de 90 degrés. Le troisième graphique correspond au contour initial avec un changement d'échelle. Dans ce cas, deux angles d'alignement sont possibles, mais, à présent, celui qui correspond au maximum du vecteur de paramétrisation n'est plus le même que celui du contour initial. C'est en effet le point de contour correspondant à 80 degrés qui correspond au maximum.

phe d'aspect. L'alignement du point de vue de l'angle de rotation se fait en utilisant le maximum effectif de la courbe de paramétrisation.

Le nombre de vecteurs de base pour l'analyse par composantes principales correspond au nombre de contours du graphe d'aspect. Le calcul de la matrice de projection se fait de manière classique en diagonalisant la matrice de covariance.

L'utilisation de l'ACP pour trouver la pose d'un contour quelconque est très simple. La première étape consiste à aligner le vecteur de paramétrisation du contour inconnu. Ensuite, on projette ce vecteur dans le sous-espace des composantes principales et on cherche quel est le vecteur de la base qui en est le plus proche. Comme il est possible que pour un seul contour plusieurs angles de rotation soient possibles, il faudra tester chacun d'eux. On peut utiliser la norme euclidienne ou la norme absolue pour effectuer la recherche du vecteur le plus proche. Dans les simulations du paragraphe suivant, nous avons pris la norme absolue afin de réduire encore le temps de calcul.

Le vecteur de la base le plus proche est associé à un point dans la sphère de points de vue. Celui-ci permet de retrouver les paramètres de pose angulaires. Les deux autres paramètres sont obtenus en utilisant l'approche classique décrite dans le chapitre précédent pour retrouver l'angle de rotation 2D et le facteur d'échelle entre un contour connu et un contour inconnu (voir section 1.4.2.3).

L'algorithme que nous venons de décrire pour calculer la pose d'un contour quelconque à l'aide d'une analyse par composantes principales est beaucoup plus rapide que l'approche exhaustive utilisée jusqu'ici. Les simulations que nous avons faites le prouvent sans qu'aucune optimisation sur le code n'ait été faite. Le paragraphe suivant illustre les résultats obtenus avec cette approche.

2.4.2.3 Résultats

La figure 2.11 montre les résultats obtenus avec cette méthode dans le cas de la théière. Les mêmes poses aléatoires que celles utilisées dans les simulations de la figure 2.8 ont été utilisées. Ceci permet d'effectuer des comparaisons entre la méthode exhaustive et la méthode par ACP.

On constate immédiatement que les résultats sont à peu près les mêmes pour les deux méthodes. La méthode par ACP ne donne toutefois pas exactement les mêmes résultats pour deux raisons.

| Nombre de simulations | Durée moyenne du calcul pour une estimation de pose en secondes | Pourcentage d'estimation parfaite | Pourcentage d'estimation correcte | Erreur moyenne absolue sur le facteur d'échelle | Erreur moyenne absolue sur l'angle de rotation 2D |
|---|---|--|---|---|--|
| 100 | 0.33 | 11% | 32% | 0.0106 | 4.28 degrés |
| 100 | 0.35 | 19% | 26% | 0.0118 | 2.03 degrés |
| 100 | 0.43 | 10% | 10% | 0.0106 | 0.80 degré |
| 100 | 0.56 | 7% | 43% | 0.0101 | 4.51 degrés |
| 100 | 0.57 | 11% | 38% | 0.0103 | 1.86 degrés |
| 100 | 0.73 | 9% | 19% | 0.0115 | 1.15 degrés |
| Facteur de discrétisation angulaire | Maillage | Nombre de composantes principales | Pourcentage de compactage des vecteurs de paramétrisation grâce à l'ACP | Rang moyen du contour idéal dans le graphe d'aspect | Rang moyen du contour idéal dans le graphe d'aspect en pourcentage |
| 5 | 18 par 18 | 49 | 85.7% | 51.5 (342) | 15.10% |
| 5 | 36 par 18 | 52 | 92.4% | 89.4 (684) | 13.06% |
| 5 | 72 par 36 | 55 | 97.9% | 366.5 (2663) | 13.76% |
| 2 | 18 par 18 | 65 | 81.0% | 31.71 (342) | 9.27% |
| 2 | 36 par 18 | 75 | 89.0% | 53.56 (684) | 7.83% |
| 2 | 72 par 36 | 85 | 96.8% | 157.89 (2663) | 5.92% |
| Erreur moyenne absolue sur le facteur d'échelle | Erreur moyenne absolue sur le premier angle d'Euler | Erreur moyenne absolue sur le deuxième angle d'Euler | Erreur moyenne absolue sur l'angle de rotation 2D | Modèle utilisé pour les simulations | |
| 0.0240 | 52.60 degrés | 30.11 degrés | 16.51 degrés |  | |
| 0.0194 | 46.52 degrés | 21.37 degrés | 14.60 degrés | | |
| 0.0187 | 36.37 degrés | 20.24 degrés | 10.66 degrés | | |
| 0.0182 | 50.44 degrés | 15.21 degrés | 14.94 degrés | | |
| 0.0157 | 37.98 degrés | 10.87 degrés | 10.28 degrés | | |
| 0.0156 | 28.72 degrés | 15.29 degrés | 8.14 degrés | | |

FIGURE 2.11 Résultats d'estimation de pose avec une approche par composantes principales.

Dans cette figure, nous présentons les résultats obtenus pour la théière. Les simulations sont effectuées de la même manière que dans la figure 2.8, mais en utilisant la méthode par composantes principales pour le calcul de la pose. La distance caméra-objet varie entre 20 et 70 et le graphe d'aspect a été calculé avec une distance de 30. Six séries de tests ont été effectuées. Trois types de maillage ont été utilisés: $N_\phi = 18$ $N_\psi = 18$, $N_\phi = 18$ $N_\psi = 36$ et $N_\phi = 36$ $N_\psi = 72$. Pour chacun de ces maillages deux facteurs de discrétisation angulaire ont été utilisés. L'un est de 5 degrés et le second de 2 degrés (Φ_1). Les trois premières lignes correspondent à 5 degrés, les trois autres à 2 degrés et les maillages vont du plus petit au plus grand dans les deux cas.

Tout d'abord, le principe d'alignement des modèles est différent pour les deux approches et, ensuite, l'analyse par composantes principales peut engendrer certaines erreurs à cause de la projection dans le sous-espace des composantes principales.

Le temps de calcul est donné ici à titre indicatif. Il ne reflète pas entièrement la réalité parce que les résultats pour chaque calcul de pose sont arrondis à la seconde près. Une résolution supérieure aurait été nécessaire pour obtenir de vraies valeurs. Ceci n'est toutefois pas nécessaire puisque le plus important est de noter que le rapport en terme de temps de calcul peut aller jusqu'à 650 entre la méthode ACP et la méthode exhaustive. L'approche par ACP est donc très rapide, même sans optimisation algorithmique.

On peut aussi constater que le pourcentage de rang moyen du contour idéal est plus mauvais dans le cas de la méthode par ACP. Ce phénomène provient de la méthode d'alignement initial des contours. Lorsqu'une erreur survient à cette étape, la projection du vecteur de paramétrisation aligné dans le sous-espace des composantes principales peut être artificiellement loin du contour idéal. Notons bien que le terme "erreur dans l'alignement" n'est pas entièrement adéquat dans la mesure où ces erreurs sont bien souvent dues au maillage discret de la sphère de points de vue plus qu'à une erreur de l'algorithme. En effet, le contour choisi par l'algorithme est souvent visuellement plus proche du contour réel que celui correspondant à la pose idéale.

Le nombre de composantes principales permet de diminuer de 80 à 98% le nombre de composantes du vecteur de paramétrisation. Ce gain montre bien que le vecteur de paramétrisation contient énormément d'informations redondantes. L'analyse par composantes principales réduit cette redondance et accélère les calculs.

2.4.2.4 Conclusion sur l'approche par ACP

Nous avons introduit une nouvelle approche permettant d'utiliser la technique de l'analyse par composantes principales pour calculer la pose d'un objet 3D à partir d'un contour d'occultation.

Les résultats sont très intéressants car ils montrent que les performances de calcul de pose sont maintenues tout en offrant un gain en rapidité de calcul par rapport à la méthode de recherche exhaustive.

2.5 Conclusion

Dans ce chapitre, nous avons apporté un éclairage nouveau sur le principe du graphe d'aspect. Cette notion très intéressante du domaine de la vision par ordinateur a été quelque peu délaissée à cause de son inadéquation pour résoudre des problèmes de calcul de pose ou de reconnaissance d'objets.

Grâce à la paramétrisation et à la mesure de comparaison qui y est associée, nous avons réussi à redonner au graphe d'aspect tout son intérêt. Nous proposons en effet une approche pour estimer la pose d'un objet 3D en combinant le graphe d'aspect et les contours d'occultation.

La méthode de calcul de la pose à partir du graphe d'aspect a été étudiée en profondeur. Deux approches ont été proposées. La première est une approche de recherche exhaustive. Elle a permis de montrer que de bonnes performances en termes de calcul de pose peuvent être atteintes. La seconde a permis de corriger le plus grand défaut de la première, à savoir sa complexité. Une analyse par composantes principales a en effet permis de réduire sensiblement la redondance d'information et ainsi diminuer le temps de calcul.

À cette étape, nous avons vu qu'il est possible d'obtenir 4 paramètres de pose sur les six décrivant un objet 3D rigide. L'étape suivante, présentée au chapitre 3, consistera à intégrer les deux derniers paramètres de translation afin de rendre la méthode complète.

Poursuite et localisation de cible: une nouvelle approche

3.1 Introduction

Comme nous allons le voir dans la revue de littérature sur la poursuite de cible, toutes les méthodes dont le but est l'estimation de la pose 3D d'un objet connu requièrent une étape de prédiction afin d'obtenir de bons résultats. C'est pourquoi le mouvement qui existe entre deux images successives doit être prédictible. De plus, il doit être relativement faible afin de ne pas nuire aux performances de prédiction.

L'algorithme que nous proposons élimine l'étape de prédiction et permet ainsi de retrouver la pose d'un objet 3D en limitant les hypothèses sur son mouvement au strict minimum. Cette approche utilise la méthode de Huttenlocher [41], qui permet de retrouver avec une grande efficacité la position d'un objet dans une image 2D. En combinant cette méthode aux outils exposés dans les chapitres précédents, nous pourrions estimer les six paramètres de pose d'un objet 3D en mouvement. Les deux paramètres de translation, parallèles au plan image, pourront être estimés à l'aide de l'algorithme de Huttenlocher. Le graphe d'aspect de l'objet 3D, tel qu'il est défini dans le chapitre 2, permet de calculer deux des angles de rotation. Enfin, les deux derniers paramètres sont calculés grâce à la paramétrisation des contours d'occultation introduite au chapitre 1.

Dans le paragraphe 3.2, nous effectuerons un survol rapide de la littérature. Nous décrirons ensuite en détails l'algorithme proposé dans la section 3.3. Les résultats obtenus seront analysés avec diverses séquences d'images 2D au paragraphe 3.4, puis nous conclurons sur les performances de notre méthode ainsi que sur les améliorations à y apporter.

3.2 État de l'art

3.2.1 Introduction

La revue de littérature sera divisée en deux, d'une part les méthodes de poursuite 2D et d'autre part les méthodes de poursuite avec localisation d'une cible 3D. La première partie montrera qu'il existe de nombreuses méthodes de poursuite de contours 2D, mais qu'elles n'ont presque jamais été utilisées pour estimer une pose 3D. La seconde partie présentera les méthodes classiques permettant de calculer la pose 3D d'une cible à partir d'une séquence d'images 2D. La présentation sera relativement succincte, mais une description plus détaillée des algorithmes dont nous parlerons peut être trouvée dans [24].

3.2.2 Méthode de poursuite 2D

3.2.2.1 Approche de Huttenlocher

L'algorithme de Huttenlocher [42] présente une méthode de poursuite 2D avec pour seule hypothèse le fait que l'apparence 2D de la cible ne change pas trop rapidement d'une image à l'autre. Comme l'approche que nous proposons dans la section 3.3 utilise cette méthode, nous donnerons davantage de détails pour cet algorithme que pour les autres que nous citerons plus loin.

3.2.2.1.1 Description globale de l'algorithme

Le but de l'algorithme de Huttenlocher est de trouver la position d'une cible connue dans une image d'arête notée I . La cible est modélisée par une image de contour notée M . Avant d'expli-

quer comment effectuer la recherche de position, il faut préciser quelques notions. On définit un ensemble de *transformations autorisées* de M qu'on note T . Cet ensemble contient généralement que les translations mais on peut l'étendre en y ajoutant des mises à l'échelle [41]. La cible sera présente dans l'image I s'il existe une transformation $t \in T$ tel que $t(M)$ soit exactement égale à une zone de I . De plus la "position" de la cible sera donné par la transformation t .

La position de la cible sera donc trouvée en recherchant une zone de l'image ayant la même "silhouette" que le modèle. Cependant, à cause du bruit de mesure et surtout de la restriction liée au choix de l'ensemble T , on ne peut pas obtenir de transformations qui permettent de retrouver exactement le modèle dans l'image. C'est pourquoi Huttenlocher définit une distance qui permet de comparer I et le modèle M (en tenant compte de l'ensemble T). Cette distance est basée sur la distance de Hausdorff. elle est donnée par:

$$d(M, I) = \min_{t \in T} h_K(t(M), I) = \min_{t \in T} K^{ieme} \min_{p \in M} \min_{q \in I} \|t(p) - q\| \quad (\text{EQ 3.1})$$

Commentons cette équation. Tout d'abord on considère que M et I correspondent à l'ensemble des coordonnées des pixels de contour uniquement. K^{ieme} correspond à la K^{ieme} plus grande valeur. Et enfin, $\| \cdot \|$ représente la distance euclidienne. Si on suppose que M possède m points de contour nous aurons toujours $K < m$. On définit f tel que $K = fm$ (f est bien sûr compris entre 0 et 1). On peut constater que s'il existe $t \in T$ tel que, au moins K points de $t(M)$ soient aussi des points de contour de I , alors $d(M, I) = 0$. D'autre part si $d(M, I) = v$, on peut approximativement dire qu'au moins K points de M (ou encore $f\%$ de M) seront à une distance maximale v de points de contour de I . Cette interprétation de la distance prouve l'intérêt de celle-ci puisqu'en la calculant on pourra aisément dire si la cible est présente ou non, tout en tenant compte des déformations possibles du modèle et du bruit de mesure. En effet, en fixant f à 0.8 et en prenant un seuil maximal τ , si $d(M, I) \leq \tau$ on sera assuré qu'au moins 80% du modèle se trouve à une distance d'au plus τ de points de contour de I . Les deux paramètres importants sont donc f et τ .

On peut aussi constater que le choix de réduire l'ensemble T en ne prenant que les translations n'est pas nécessairement restrictif puisque f et τ permettent d'inclure des modifications dans la forme (dues par exemple à une rotation de la cible par rapport à un axe parallèle au plan image de la caméra 2D).

3.2.2.1.2 L'algorithme de recherche

On note I_t , l'image de contour et M_t le modèle de la cible à l'instant t . Le but de l'algorithme de Huttenlocher est de trouver la position du modèle M_t dans l'image I_{t+1} . Notons T_{t+1} l'ensemble des translations possibles pour M_t dans I_{t+1} . La recherche de la translation, qui correspond le mieux à la nouvelle position de la cible, est faite en minimisant la distance basée sur celle de Hausdorff entre I_{t+1} et M_t sur l'ensemble des positions possibles. Mais au lieu de chercher uniquement la meilleure translation, on va définir X_{t+1} comme étant le sous-ensemble de T_{t+1} défini par:

$$X_{t+1} = \{x \in T_{t+1} \mid h_K(M_t \oplus x, I_{t+1}) \leq \tau\} \quad (\text{EQ 3.2})$$

La notation $M_t \oplus x$ correspond au modèle translaté de x par rapport au coin supérieur gauche de l'image. X_{t+1} correspond à l'ensemble des translations x pour lesquelles au moins K points de contour de $M_t \oplus x$ sont à une distance maximale de τ de points de contour de I_{t+1} . La position exacte de la cible est ensuite calculée en prenant la translation qui donne la plus petite distance. Cependant, si plusieurs translations sont solutions, on utilise un second critère. En effet, on associe à chaque translation le nombre de pixels de contour de $M_t \oplus x$ qui correspond exactement à des pixels de contour de I_{t+1} . La translation pour laquelle ce nombre est maximum déterminera la bonne position.

L'intermédiaire de calcul X_{t+1} présente deux avantages. Tout d'abord, si plusieurs translations donnent sensiblement la même distance de Hausdorff on pourra choisir la "bonne" position de la cible grâce à d'autres critères simples (information de trajectoire par exemple). Ceci est surtout très utile lorsque la cible est présente plusieurs fois dans l'image. Le second avantage réside dans l'implantation puisqu'il existe un algorithme très rapide permettant de calculer

X_{t+1} grâce à la “distance map” de l’image de contour que nous noterons D_{t+1} pour l’image I_{t+1} . D_{t+1} représente pour chaque pixel de I_{t+1} la distance euclidienne du pixel de contour le plus proche. Pour un pixel de contour cette distance est évidemment nulle. Il existe un algorithme rapide qui calcule la distance euclidienne de manière séquentielle (“four-point sequential Euclidean distance mapping algorithm”), l’ensemble des pixels est parcouru deux fois seulement et l’erreur sur la distance n’apparaît qu’en de rares endroits et est au maximum de 0.29 unité de pixel.

3.2.2.1.3 Mise à jour du modèle

Une fois que la position x de la cible M_t est trouvée dans I_{t+1} , on effectue une mise à jour du modèle. Ceci permet de tenir compte des modifications de l’aspect de la cible au cours du temps. C’est cette mise à jour dynamique du modèle qui permet de poursuivre des objets non rigides (à condition bien sûr que les changements ne soient pas trop brusques).

La mise à jour du modèle se fait de la manière suivante: on dilate le modèle avec un disque de rayon δ puis on obtient M_{t+1} en effectuant un ET logique entre le modèle dilaté à la position x et l’image I_{t+1} (la dilatation et le ET logique sont calculés en considérant que les pixels de contour ont une valeur 1 et les autres une valeur 0).

Plus la valeur de δ est grande, plus on pourra incorporer de changement dans le nouveau modèle. Mais on risque aussi d’inclure d’autres informations n’ayant rien à voir avec la cible. Le modèle M_{t+1} peut s’agrandir ou rapetisser en tenant compte du nombre de points d’arêtes qui se trouvent sur les bords. Ce changement de taille permet de tenir compte des changements d’échelle.

Afin de rendre l’algorithme plus robuste à des changements brusques d’apparence, une base de modèles est créée. Celle-ci englobe progressivement tous les aspects que peut prendre la cible. En effet, chaque nouveau modèle est ajouté à la base, s’il est suffisamment différent de tous ceux déjà présents. La comparaison se fait là aussi avec la distance de Hausdorff. L’utilité d’une

telle base est de retrouver la cible lorsque celle-ci a été perdue avec le modèle actif M_t , en remplaçant successivement M_t par chacun des modèles de la base.

3.2.2.1.4 Quelques résultats obtenus avec l'algorithme de Huttenlocher

Les résultats obtenus par cet algorithme sont relativement bons. La figure 3.1 montre l'évolution de l'algorithme sur une séquence de football. Un des joueurs est suivi et on peut voir comment l'utilisation de la base de modèle permet d'inclure progressivement l'évolution de la position du joueur ainsi que l'apparition du bas de son corps qui était initialement caché. Lors de ce test, la taille du modèle pouvait changer, ce qui posait un problème puisque le modèle englobait progressivement des points de contour appartenant à un autre joueur. C'est l'un des inconvénients majeurs de cet algorithme.

La figure 3.2 permet de voir que l'algorithme offre de très bons résultats même lorsque la silhouette change relativement rapidement d'aspect et que le mouvement est grand.

Cet algorithme a aussi été utilisé pour permettre le déplacement d'un robot mobile vers une cible connue [43]. Les principaux avantages de cette approche sont les faibles contraintes sur le mouvement de la cible et le temps de calcul qui peut être optimisé grâce à diverses techniques [44].

Le point le plus délicat est la mise à jour du modèle. Souvent, des caractéristiques non pertinentes sont incluses dans le modèle, ce qui a pour effet de rendre progressivement la poursuite moins précise.

3.2.2.2 Autres approches de poursuite 2D

Il est difficile de parler de poursuite 2D sans parler des approches basées sur les contours actifs. Kass et al. [46] ont été parmi les premiers à proposer d'utiliser ce type de modèle pour la poursuite 2D. Leymarie et Levine [60] ont effectué la poursuite de cellules vivantes - les fibroblastes - se déplaçant sur des surfaces planes. Les résultats qu'ils ont obtenus permettent de suivre correctement les mouvements de cellules vivantes, malgré des images relativement bruitées et des

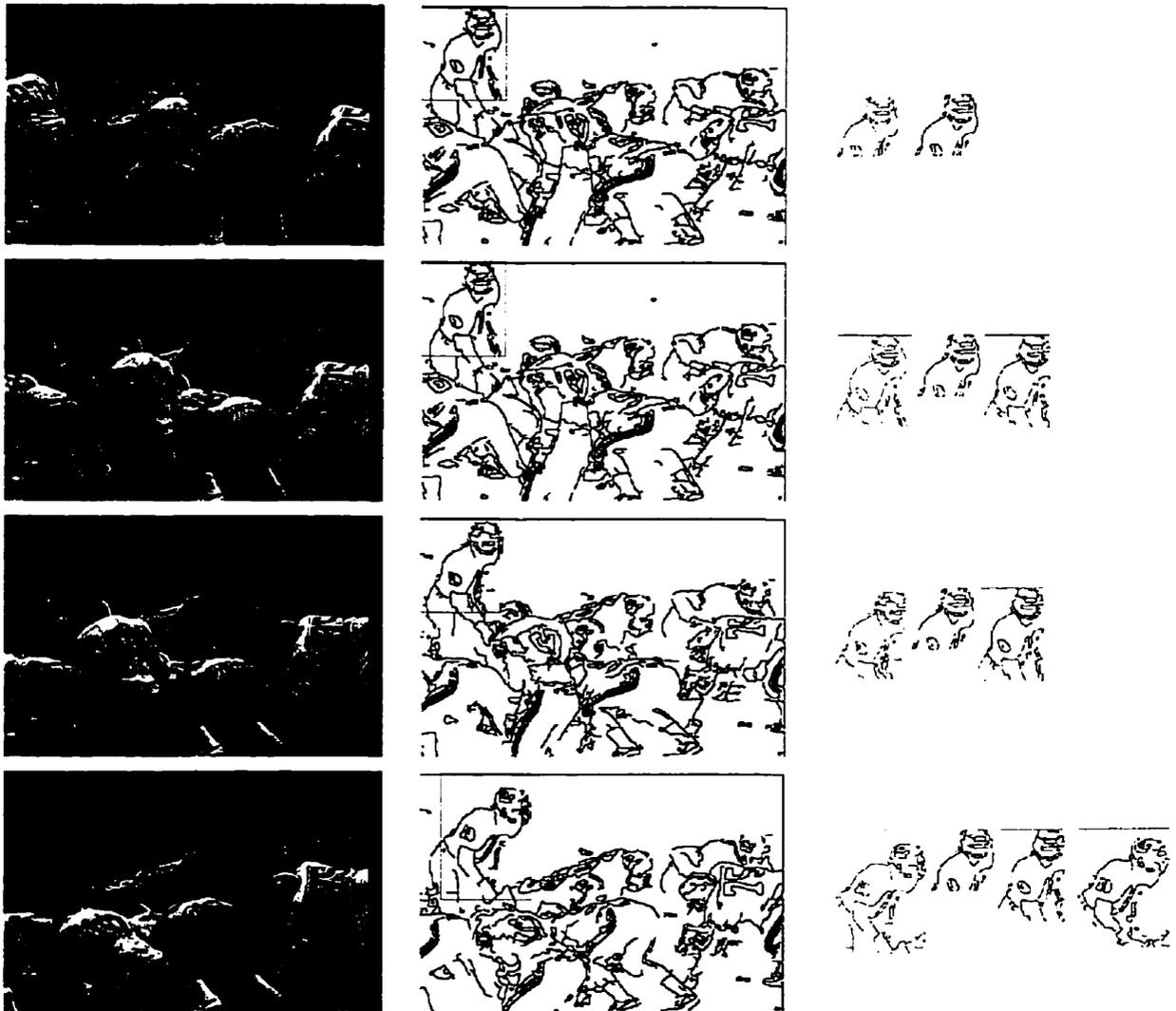


FIGURE 3.1 Poursuite de cible avec la méthode de Huttenlocher (modèle à taille variable)

La poursuite de cible est effectuée sur la séquence football (images 1 à 30). La première ligne de cette figure représente la première image de la séquence ainsi que le modèle initial et sa position dans l'image. Les trois lignes suivantes correspondent aux images 10, 20 et 30. La première colonne représente l'image avec la position du modèle en rouge. La seconde colonne représente l'image d'arêtes et le rectangle magenta représente la position du modèle. Enfin, la dernière colonne représente les modèles actifs en rouge, ainsi que la base de modèles en bleu.

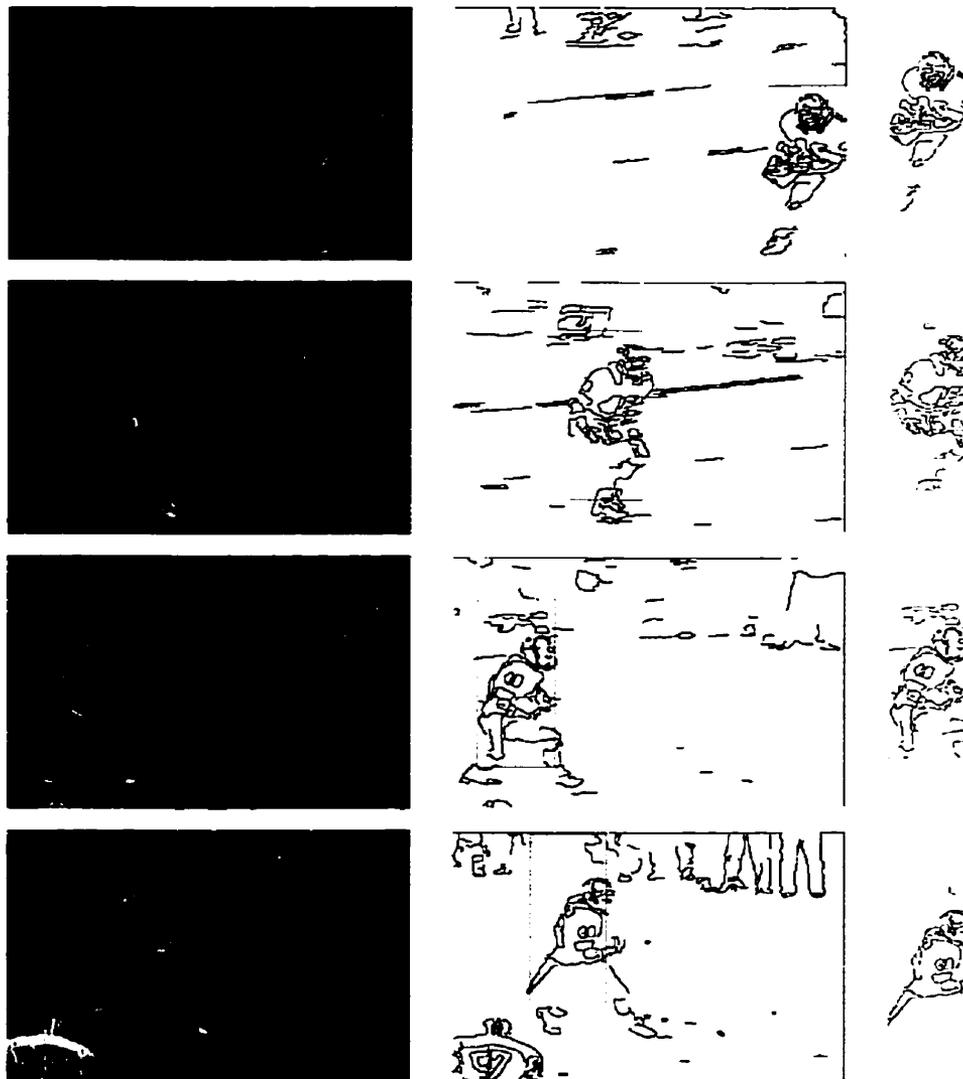


FIGURE 3.2 Poursuite de cible avec la méthode de Huttenlocher (modèle à taille fixe)

La poursuite de cible est effectuée sur la séquence football (images 165 à 195). La première ligne de cette figure représente la première image ainsi que le modèle initial et sa position dans l'image. Les trois lignes suivantes correspondent aux images 175, 185 et 195. La première colonne représente l'image avec la position du modèle en rouge. La seconde colonne représente l'image d'arêtes et le rectangle magenta représente la position du modèle. La dernière colonne représente les modèles actifs en bleu.

contours de cellules souvent difficiles à définir visuellement. Delagnes a donné une définition un peu différente pour les contours actifs [23] pour introduire la notion de polygones ajustables. Grâce à cette notion il devient possible d'effectuer la poursuite d'objets manufacturés possédant des coins. Des résultats sur un suivi de véhicule confirment l'intérêt de ces contours particuliers. Berger a aussi utilisé les contours actifs, que ce soit avec des objets non rigides [5], ou avec des objets rigides, mais non polyédriques [6,7]. Ses résultats démontrent eux aussi l'intérêt de cette approche. Il faut toutefois noter que le choix des nombreux paramètres définissant les contours actifs n'est pas toujours une chose facile. De plus, le mouvement de l'objet ne doit pas être trop important pour que la poursuite se fasse adéquatement. Enfin, il faut tenir compte du bruit dans l'image qui peut attirer le contour vers d'autres objets.

Terzopoulos et Szeliski [78] ont proposé une méthode basée à la fois sur les contours actifs et sur une approche statistique utilisant le filtrage de Kalman. Les résultats obtenus se traduisent par l'apparition d'un phénomène de "mémoire" qui permet de conserver approximativement la forme globale du contour, chose qui n'était pas possible avec la méthode classique. L'algorithme itératif nécessite cependant beaucoup de temps de calcul.

L'utilisation de splines ou, plus généralement, de contours définis par quelques points de contrôle est une autre approche très populaire pour effectuer la poursuite 2D. En effet, Blake, Curwen et *al.* [10,11,12,21,47] l'ont utilisé dans une application de suivi de mouvement de mains ou de lèvres, Koller [52-54] a proposé un algorithme de poursuite de véhicule afin d'analyser le trafic sur une autoroute et Baumberg et Hogg [4] ont mis au point le suivi d'une personne qui marche. L'utilisation du filtrage de Kalman pour mettre à jour les paramètres de contrôle des contours donne une assez bonne robustesse à l'ensemble de ces méthodes. L'utilisation des splines est surtout appropriée pour la poursuite d'objets non rigides.

3.2.2.3 Conclusion

L'ensemble de ces méthodes permet d'effectuer une poursuite de cible 2D, mais sans qu'une information précise sur la pose 3D ne puisse être calculée. L'utilisation de certaines de ces

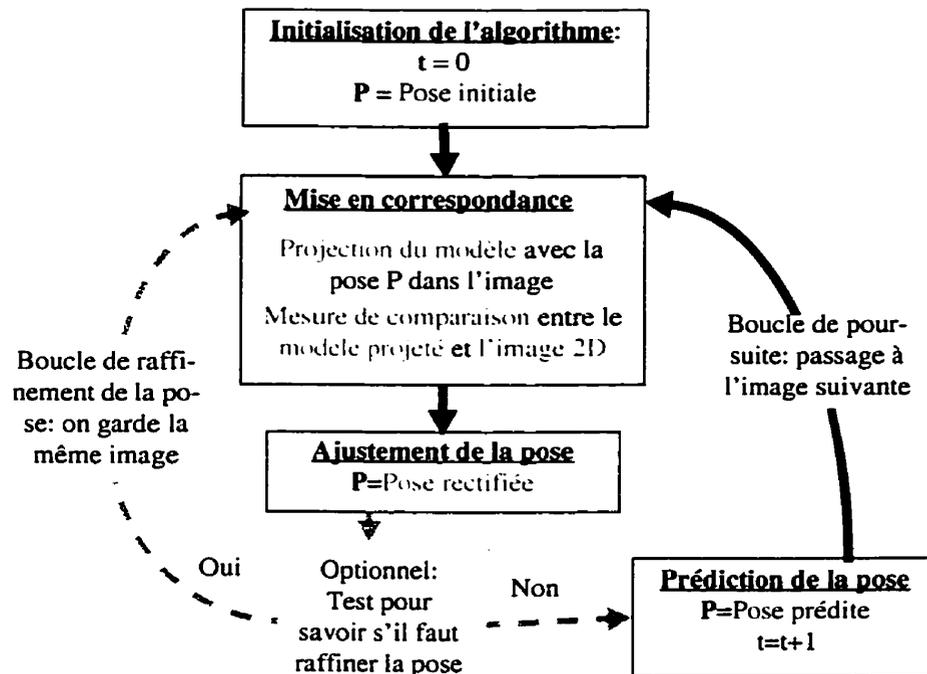


FIGURE 3.3 Schéma de principe des algorithmes d'estimation de pose

méthodes pour obtenir une estimation, même grossière de la pose présenterait cependant un grand intérêt comme nous allons le montrer dans le paragraphe 3.3. Il s'agit là d'un aspect qui n'a pas été véritablement exploré dans la littérature jusqu'ici.

3.2.3 Poursuite de cible avec estimation 3D de la pose

3.2.3.1 Principe général

Après avoir vu les approches de poursuite 2D, voyons à présent quelles sont les méthodes de poursuite avec une estimation 3D de la pose. La plupart des algorithmes permettant une estimation de la pose 3D à partir d'une séquence d'images 2D sont basés sur le même principe. Une pose initiale de l'objet est connue soit par un processus d'initialisation au début de la poursuite, soit par un processus de prédiction utilisant les résultats obtenus dans l'image précédente. Cette pose initiale permet alors de projeter le modèle de la cible dans le plan image. À partir de cette

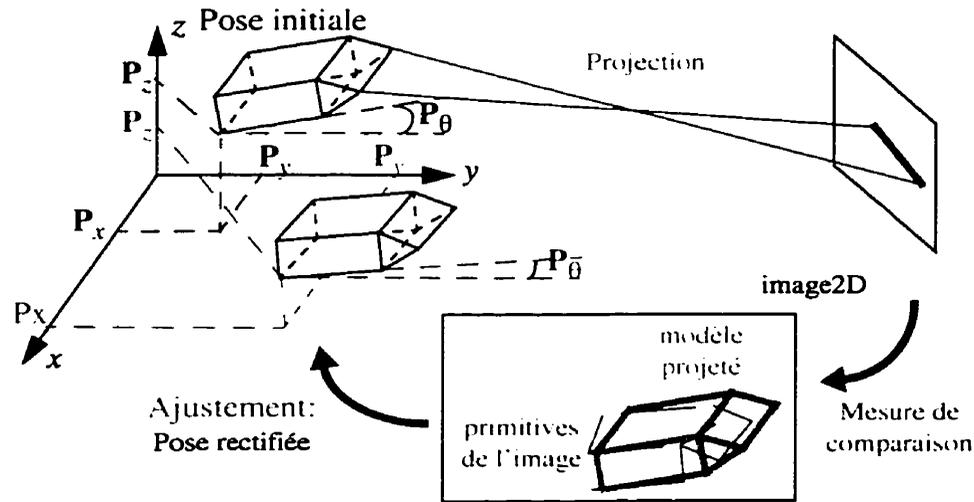


FIGURE 3.4 Exemple illustrant le principe d'action des algorithmes d'estimation de pose

projection, il est possible de savoir si la pose est bonne ou non. Une mesure de comparaison entre l'image 2D et le modèle projeté est utilisée pour effectuer un ajustement de la pose.

Dans certains algorithmes, ce premier ajustement est bon, mais pour d'autres, un processus itératif est nécessaire pour raffiner encore la pose. Le premier ajustement sert alors de nouveau point de départ. Les itérations s'arrêtent soit lorsque la mesure de comparaison est suffisamment petite, soit lorsqu'un nombre suffisant d'ajustements a été fait.

La dernière étape consiste à prédire la pose dans l'image suivante. Cette étape nécessite la connaissance de paramètres tels que la vitesse ou l'accélération. Des modèles de mouvement plus ou moins élaborés existent en fonction du type d'application visée.

On peut schématiser ces algorithmes en y faisant apparaître différents modules tel que montré à la figure 3.3. Chacun de ces modules, à savoir l'initialisation, la mise en correspondance (projection et mesure de comparaison), l'ajustement de la pose et la prédiction de la pose pour l'image suivante, est traité de manière bien différente suivant les applications ou suivant les auteurs. La figure 3.4 montre par un exemple très simple comment ce type d'algorithme fonctionne.

3.2.3.2 Étude comparative de quelques approches particulières

Plutôt que de décrire en détails les différentes approches de la littérature, nous avons choisi d'en sélectionner quelques-unes et de les comparer. Le tableau 3.1 offre une vue d'ensemble des algorithmes. Le type de modèle, la méthode de prédiction, d'initialisation et d'ajustement ainsi que le type de primitive 2D utilisée et la fonction de mise en correspondance entre modèle et image y sont donnés.

Les onze algorithmes qui ont été étudiés sont nommés en fonction de l'auteur qui a apporté la contribution majeure. Le premier auteur cité est Genery. Il a été l'un des premiers à proposer un système complet de poursuite de cibles avec une estimation de la pose en 3D à partir d'images 2D [35,36]. Il a aussi introduit un système de filtrage différent du filtrage de Kalman classique pour l'étape de prédiction.

Baker, Sullivan et Worrall ont largement travaillé sur la poursuite de véhicule. Ils ont proposé une approche de base avec trois grandes variantes pour l'estimation de la pose en 3D [2, 75, 76, 79, 80]. Nous y ferons référence par Baker, Worrall et Worrall². Dans leur cas l'étape de prédiction se fait par filtrage de type Kalman, mais avec une mise à jour originale de la matrice de covariance proposée par Maybank [65].

Lowe a lui aussi proposé une approche pour calculer la pose d'une cible à partir d'images 2D [61, 62, 63]. Les objets suivis sont décrits à l'aide d'un modèle d'approximation polygonale originale permettant de traiter également les objets articulés.

Koller a développé un algorithme de poursuite pour des véhicules [51]. Le modèle utilisé est un modèle en fil de fer et la pose est définie par trois paramètres: deux pour la position dans le plan de la route et un pour l'orientation par rapport à la perpendiculaire à la route. L'étape de prédiction se fait dans son cas par un filtrage proche de celui du filtre de Kalman étendu itératif.

L'algorithme RAPID de Harris permet d'effectuer la poursuite d'un objet rigide en tenant compte de ses 6 degrés de liberté [38]. L'avantage énorme de cette méthode réside dans sa rapidité d'exécution.

Gavrila a, quant à lui, travaillé sur le suivi et la reconnaissance de mouvements humains [33,34]. Les modèles qu'il a utilisés font appel à des super-quadriques associées à un squelette qui permet de définir les articulations entre les différentes parties du corps. L'espace des paramètres à estimer a une dimension proportionnelle au nombre d'articulations qui sont traitées.

Dans sa thèse, Rehg a proposé un algorithme de suivi de mouvement humain [69, 70]. Le mouvement étudié est celui de la main. Un modèle cinématique, semblable à ceux utilisés en robotique, permet de représenter les différents degrés de liberté de la main.

Basu a proposé un algorithme basé sur la régularisation du flux optique pour la poursuite du mouvement de la tête [3, 9]. Le modèle qu'il utilise est une ellipsoïde avec des paramètres fixes. Cette ellipsoïde est échantillonnée pour donner un ensemble de points 3D.

Chan, Métaxas et Terzopoulos ont proposé une approche de poursuite à l'aide de modèles 3D déformables. Ce sont surtout des super-ellipsoïdes déformables qui ont été utilisées. L'algorithme proposé correspond en fait à la généralisation en 3D de la poursuite 2D avec des contours actifs (cf section 3.2.2.2) [15, 77].

En observant le tableau 3.1, on peut constater que l'étape d'ajustement de la pose ne se fait le plus souvent que de deux manières: soit en minimisant une fonction dans le voisinage d'un point de manière systématique (méthode du simplexe par exemple), soit en la minimisant avec une technique par moindres carrés souvent associée à un algorithme itératif (méthode de Newton ou de Levenberg-Marquardt). Il existe aussi des méthodes actives basées sur une approche physique pour lesquelles l'ajustement est fait en intégrant des équations différentielles (méthode de Worrall et de Terzopoulos-Metaxás).

L'étape d'initialisation est, quant à elle, très spécifique et dépend énormément du type d'application. De plus, dans de nombreux cas, elle n'est pas abordée du tout. La prédiction est elle aussi spécifique au type d'applications, mais elle se rapporte presque toujours au filtrage de Kalman.

TABLEAU 3.1 Résumé des méthodes 3D - Les algorithmes.

| méthode | type de modèle | prédiction | initialisation | primitives 2D utilisées | fonction de mise en correspondance | ajustement |
|-----------------|------------------------------------|---|---------------------------------------|--|--|--|
| Gennery | objet polyédrique avec réflectance | filtrage de type Kalman avec calcul de la covariance élaboré | non traitée | points et segments de droite | erreur entre primitive 2D et modèle | minimisation par moindres carrés. méthode non-itérative |
| Baker | modèle en fil de fer | Kalman ou Maybank | non traitée | gradient directionnel | évaluation "iconic" | recherche du maximum de la fonction d'évaluation. méthode itérative |
| Worrall | modèle en fil de fer | Kalman ou Maybank | non traitée | gradient directionnel | évaluation "iconic" et force 3D | déplacement dû à la force 3D. méthode itérative |
| Worrall2 | modèle en fil de fer | Kalman ou Maybank | analyse de mouvement ou vote | gradient directionnel | évaluation "iconic" et fonction d'erreur | minimisation de l'erreur par moindres carrés. méthode itérative |
| Lowe | modèle CAO adapté | aucune | non traitée | segments de droite | distance perpendiculaire entre segments appariés | minimisation par moindres carrés (Levenberg et Marquardt) méthode itérative |
| Koller | modèle en fil de fer | type IEKF (Iterative Extended Kalman Filter), mais faite en même temps que l'ajustement | analyse du mouvement par flux optique | segments de droite (modélisation de Deriche et Faugeras) | mise en correspondance des segments de l'image 2D et des segments projetés du modèle | recherche du Maximum <i>a posteriori</i> (Gauss-Newton ou Levenberg-Marquardt), méthode itérative |

TABLEAU 3.1 Résumé des méthodes 3D - Les algorithmes.

| méthode | type de modèle | prédiction | initialisation | primitives 2D utilisées | fonction de mise en correspondance | ajustement |
|--------------|---|--|--|---|--|--|
| Harris | points d'intérêt | filtrage de Kalman avec accélération nulle | non traitée | gradient dans une direction privilégiée autour des points d'intérêt | distance entre le point d'intérêt et le maximum du gradient | minimisation de la distance par moindres carrés, méthode non-itérative |
| Gavrila | super-quadrilles | simple par ajustement d'un polynôme de degré 2 | détermination d'une région d'intérêt et évaluation itérative de l'axe principal par PCA. | points d'arêtes | distance de chamfer | recherche du minimum de la distance autour de la pose initiale en découpant l'espace de pose, méthode itérative |
| Rehg | modèle cinématique et template de chaque articulation sous la forme d'image d'illuminance | aucune | non traitée | image d'illuminance directement | erreur quadratique entre les template et l'image | minimisation par descente du gradient |
| Basu | ellipsoïde | aucune | méthode de Moghaddan et Pentland | flux optique | erreur quadratique entre le flux optique prédit en fonction de la pose et le flux mesuré | minimisation de l'erreur quadratique en utilisant l'algorithme du simplexe |
| Chan-Metaxas | modèle paramétrique déformable (super ellipsoïde déformable) | filtrage de Kalman | segmentation des primitives en partie suivie de la reconstruction de la forme à partir de plusieurs vues | gradient de l'image | fonction d'énergie dépendant du gradient de l'image | résolution de l'équation dynamique du second ordre par intégration par la méthode de Euler et avec un filtrage de Kalman |

Le tableau 3.2 est un résumé des points forts et des faiblesses des différents algorithmes. Ce qui en ressort c'est l'importance pour toutes les méthodes de la pose initiale. Plus cette pose est bonne, plus l'algorithme peut converger rapidement et plus l'estimation de la pose sera bonne.

On peut aussi constater que, pour l'ensemble des méthodes, la seule technique utilisée pour obtenir une bonne valeur initiale de la pose pour chaque nouvelle image est la prédiction.

TABLEAU 3.2 Résumé des méthodes 3D - Les performances

| Méthode | Type d'application | Points forts | Faiblesses |
|----------|--|--|--|
| Gennery | poursuite d'objet polyédrique | dynamique du modèle permettant de tenir compte de variations d'angle allant jusqu'à 30 | importance de la pose initiale |
| Baker | poursuite de véhicule | bonne convergence (peu d'attracteurs autres que la bonne pose) | grand nombre d'itération |
| Worrall1 | poursuite de véhicule | convergence assez rapide | choix d'un paramètre crucial pour la convergence |
| Worrall2 | poursuite de véhicule | convergence très rapide | Nombreux attracteurs, donc importance d'une bonne pose initiale |
| Lowé | n'importe quel objet articulé | convergence rapide tient compte de non-linéarité | pas de prédiction |
| Koller | poursuite de véhicule | tient compte de l'éclairage de la scène tient compte de non-linéarités | importance de la pose initiale et des appariements pour la convergence modèle dynamique limité pas de traitement pour d'éventuels mauvais appariements |
| Harris | suivi d'objet rigide | rapidité, temps réel vitesse angulaire jusqu'à 10rad/s | l'utilisation de point doit rendre la méthode sensible au bruit |
| Gavrila | suivi et reconnaissance de mouvement humain | résultats corrects malgré le grand nombre de paramètres | importance de la pose initiale pas de déformation possible pour le modèle |
| Rehg | suivi de mouvement humain: mouvement des mains | traitement des occultations | pas du tout de prédiction simulations uniquement pour des scènes avec un background noir. |

TABLEAU 3.2 Résumé des méthodes 3D - Les performances

| Méthode | Type d'application | Points forts | Faiblesses |
|--------------|--|---|---|
| Basu | Suivi de mouvement humain: mouvement de la tête (comme objet rigide) | résultats bons et fiables parce que testés sur une séquence où ceux-ci sont connus d'avance | pas de prédiction entre les images. Importance de la pose initiale car utilisation du simplexe pour optimiser la fonction. Projection d'un patron utilisée pour le calcul du flux optique |
| Chan Metaxas | Suivi d'objets déformables | permet de s'adapter aux déformations des objets permet de traiter des occultations | pas vraiment testé sur des objets déformables (quand des images 2D sont utilisées) nécessité d'images stéréo pour obtenir de bons résultats |

3.2.3.3 Autres méthodes

La revue de littérature présentée jusqu'ici n'est évidemment pas exhaustive. Cependant, la majorité des travaux publiés peuvent être associés aux techniques de poursuite de cible que nous avons décrites ou à des combinaisons de celles-ci. Les algorithmes de Ravela [68] ou de Daucher et Dhome [22] ressemblent, par exemple, à l'approche de Lowe, celui de Lerasle [59] et Yassine [81] se rapproche quant à lui de la méthode de Worrall.

3.2.4 Conclusion

Une analyse de la revue de littérature a montré que l'utilisation de méthodes de poursuite 2D pour obtenir une estimée initiale de la pose 3D est une voie qui n'a pas encore été explorée. Elle pourrait cependant offrir une alternative intéressante aux algorithmes classiques de poursuite de cible avec estimation de la pose 3D qui requièrent toujours une étape de prédiction de la pose.

L'approche que nous proposons, basée sur l'utilisation du graphe d'aspect, permet de faire le lien entre le monde de la poursuite 2D et celui de l'estimation de la pose 3D.

3.3 Approche proposée

3.3.1 Paramètres d'entrée

Avant de décrire notre algorithme, nous allons énumérer ses données d'entrée qui peuvent être classées en trois catégories.

La première catégorie réfère à la séquence d'images 2D. Elle contient les images 2D proprement dites. Ces images peuvent être en couleur ou non, même s'il faut noter que seules les informations d'illuminance seront prises en compte. Les différents paramètres utilisés pour créer la séquence d'images font aussi partie de cette catégorie. Ils sont au nombre de trois: la position du centre de projection défini à partir de ses coordonnées X_{offset} et Y_{offset} dans l'image, la distance focale que nous noterons *Focalfactor* et, enfin, le rapport entre ordonnées et abscisses noté *AspectRatio*. Il s'agit en fait des paramètres intrinsèques de la caméra.

La seconde catégorie correspond à la modélisation de l'objet 3D décrite dans le chapitre d'introduction. Il s'agit d'une modélisation par approximation polygonale. Elle peut être définie à l'aide d'un fichier VRML. Cette modélisation sera utilisée pour générer le contour d'occultation de l'objet 3D placé dans une position particulière. Afin de compléter les informations sur le modèle 3D, nous utiliserons aussi la modélisation sous forme de graphe d'aspect décrite dans le chapitre 2. Le graphe d'aspect est défini à l'aide de 3 paramètres: la distance objet-caméra que nous noterons Z_{graphe} et les facteurs de discrétisation des angles d'Euler N_t et N_p . D'autres paramètres sont aussi implicitement utilisés dans le graphe d'aspect. Ce sont les constantes utilisées pour la paramétrisation des contours 2D, c'est-à-dire le facteur de discrétisation angulaire Θ_s et le facteur de discrétisation radiale N_s . Tous les détails sur la paramétrisation des contours se trouvent dans le chapitre 1.

La troisième catégorie de paramètres d'entrée de l'algorithme concerne les paramètres reliés à la pose. Il s'agit tout d'abord des six paramètres de la pose de l'objet pour chacune des images de la séquence. Ces paramètres sont utilisés uniquement pour l'évaluation des performances de

l'algorithme ainsi que pour l'initialisation. Enfin, nous définissons six paramètres de tolérance qui nous permettent de situer dans quelle plage dynamique la pose peut évoluer entre deux images successives. Ces tolérances permettent de tenir compte de la corrélation temporelle.

3.3.2 Description détaillée de l'algorithme

Après avoir décrit les paramètres d'entrée de l'algorithme, décrivons les détails de l'implantation en supposant que la pose de l'objet 3D a été calculée avec succès pour l'image précédente. L'étape d'initialisation sera vue dans le paragraphe 3.3.2.8

3.3.2.1 Détection des contours d'occultation dans l'image

La première étape de l'algorithme consiste à extraire les contours d'occultation de l'image en cours. Cette étape peut se faire de deux manières différentes. La première consiste à utiliser l'algorithme de Canny décrit dans [14] pour extraire les points d'arêtes. Le calcul se fait à partir d'une image d'illuminance. Par conséquent, lorsque l'image d'entrée est en couleur, il faut la convertir en une image en niveau de gris. Nous verrons que cette méthode possède des inconvénients puisque ce ne sont pas seulement les contours d'occultation qui apparaissent, mais aussi tous les autres types de contours (changement de luminosité, changement de courbure, etc.).

La seconde approche consiste à utiliser une méthode de segmentation de l'image en objets. Les contours d'occultation correspondent alors aux frontières entre les différents objets reconnus par l'algorithme. La méthode que nous utiliserons se résume à un simple seuillage. Dans la pratique, une méthode de segmentation par seuillage n'est souvent pas assez fiable et il faut alors utiliser des méthodes plus sophistiquées pour obtenir des frontières plus lisses et pour éliminer les régions trop petites.

Le but de cette étape est, rappelons-le, d'obtenir les contours d'occultation des objets présents dans la scène visualisée. Si des contours indésirables viennent s'y ajouter, l'étape d'estimation de la pose s'en trouvera affectée. Il est donc important d'avoir une méthode fiable. Développer

une telle méthode dépasse largement le cadre de ce doctorat et nous admettrons donc qu'il est possible de mettre au point un algorithme efficace pour extraire les contours d'occultation. Une approche combinant extraction de points d'arêtes et segmentation serait probablement très efficace. Surtout si on y ajoutait une méthode de détection des contours fermés [29] puisque les contours d'occultation en sont.

La figure 3.5 résume les différentes étapes de l'algorithme. On y voit le résultat de la détection d'arêtes pour une image particulière. À l'issue de cette étape, nous disposons d'une image binaire dont les pixels valant un représentent des points du contour d'occultation. Nous noterons cette image I_n , n représentant l'indice dans la séquence d'images.

3.3.2.2 Extraction des informations pertinentes dans l'image

3.3.2.2.1 Algorithme de Huttenlocher

La seconde étape consiste à retrouver la position 2D de l'objet. Le calcul de cette position se fait à l'aide de l'algorithme de Huttenlocher. Cette méthode est décrite dans le paragraphe 3.2.2.1. Nous n'entrerons donc pas dans les détails ici. Notons simplement que le modèle permettant de retrouver la position de l'objet dans C_n est le contour d'occultation calculé pour l'image précédente. Nous noterons M_{n-1} ce contour.

Cette étape de l'algorithme ne donne de bons résultats qu'à condition de respecter les hypothèses faites par Huttenlocher. Il faut, par conséquent, que le changement de forme du contour d'occultation de l'objet entre deux images successives ne soit pas trop important. Cette contrainte se traduit en terme de pose par une variation limitée entre deux images. Il pourrait s'agir d'une limitation importante de l'algorithme, mais nous verrons dans la suite que cette condition n'est pas trop contraignante.

3.3.2.2.2 Masque d'extraction du contour

Après avoir trouvé la position de M_{n-1} dans l'image I_n , nous pourrions extraire des informations pertinentes sur le contour d'occultation de l'objet dans l'image I_n . Pour cela, nous utiliserons une technique fort simple qui consiste à dilater le contour d'occultation M_{n-1} et à effectuer un ET logique avec l'image I_n . Nous noterons R_{dil} le rayon de la dilatation et M_n^{ext} le contour d'occultation extrait de l'image. Nous aurons:

$$M_n^{ext} = I_n \& Dil(M_{n-1}, R_{dil}) \quad (\text{EQ 3.3})$$

La dilatation du contour M_{n-1} permet de tenir compte des variations de la pose d'une image à la suivante. La valeur du facteur de dilatation est très importante puisque si R_{dil} est trop faible, certains points du contour d'occultation de l'objet seront omis. Par contre, si R_{dil} est trop grand, des points d'arêtes n'ayant aucun rapport avec l'objet seront ajoutés. Ce paramètre permet par conséquent de contrôler en partie la tolérance au mouvement saccadé.

3.3.2.3 Calcul de la paramétrisation des contours extraits de l'image

Le contour d'occultation extrait de l'image M_n^{ext} permet de calculer la paramétrisation polaire qui y est associée. Le calcul se fait en utilisant l'algorithme décrit dans le chapitre 2. Le résultat de cette étape fournit le vecteur de paramétrisation V_n^P ainsi que son spectre d'amplitude A_n^P et son spectre de phase P_n^P .

C'est à cette étape qu'interviennent les paramètres de discrétisation angulaire Θ_s et de discrétisation radiale N_s . Ils permettent d'augmenter ou de diminuer la précision de la paramétrisation.

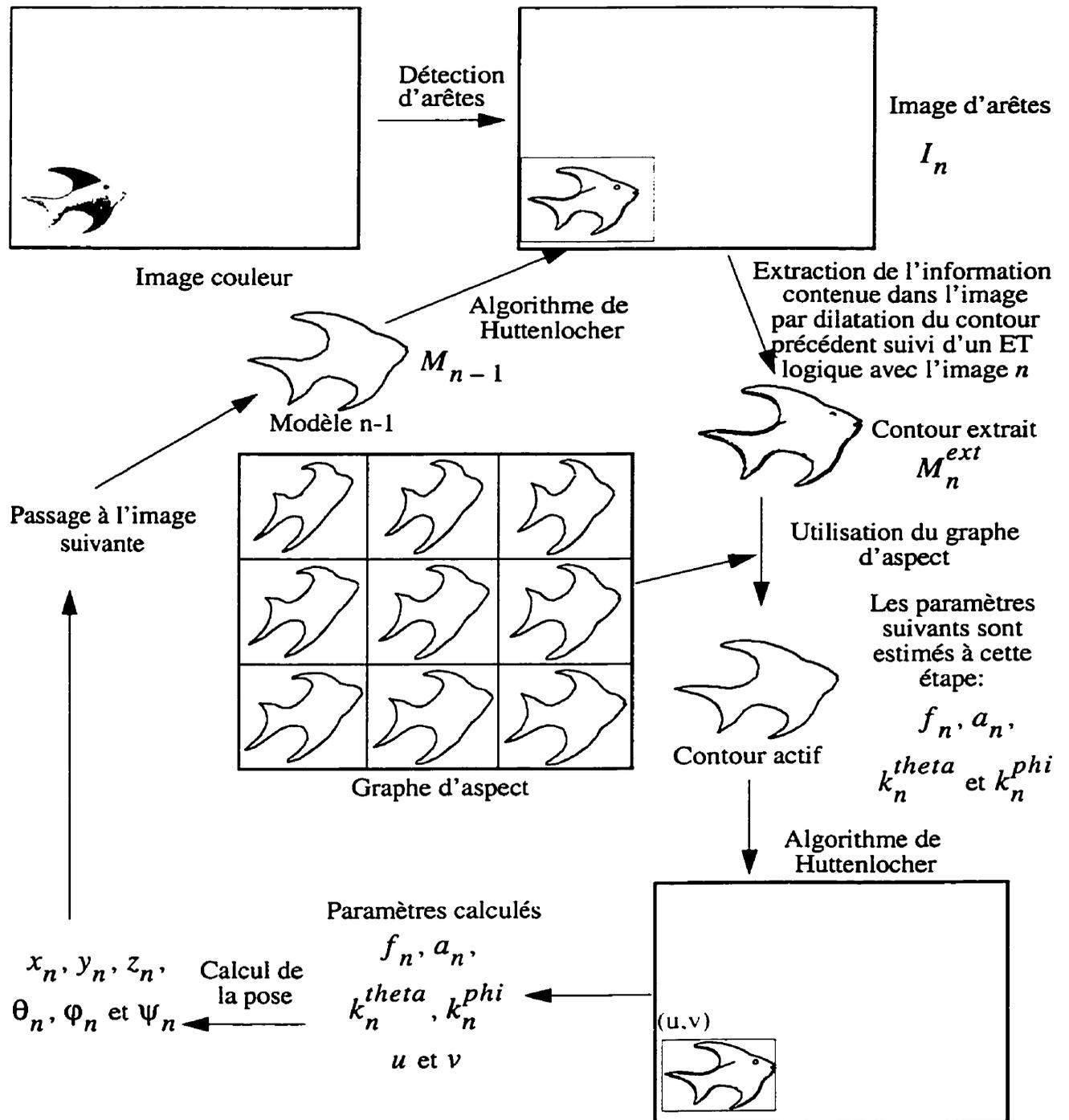


FIGURE 3.5 Description de l'algorithme de calcul de pose.

Cette figure illustre le principe de l'algorithme d'estimation de pose que nous proposons.

3.3.2.4 Utilisation du graphe d'aspect

Les vecteurs V_n^P , A_n^P et P_n^P sont utilisés pour retrouver le contour du graphe d'aspect le plus proche du contour extrait de l'image. Nous avons vu dans le chapitre 2 qu'il est possible d'utiliser soit une méthode exhaustive, soit une méthode par analyse en composantes principales pour effectuer cette recherche. La méthode par analyse en composantes principales est beaucoup plus rapide tout en étant presque aussi performante. Nous verrons si ces conclusions se confirment avec cette application.

À cette étape de l'algorithme, les résultats sont les coordonnées du contour dans le graphe d'aspect. Nous les noterons k_n^{theta} et k_n^{phi} . Nous obtiendrons aussi le facteur d'échelle f_n et l'angle de rotation α_n associés à ce contour.

Il est à noter que c'est à cette étape que l'on peut définir des tolérances sur les angles de rotation. Ainsi, si on le désire, il est possible de limiter la recherche dans le graphe d'aspect autour de la position calculée pour l'image précédente $(k_{n-1}^{theta}, k_{n-1}^{phi})$. De la même manière, il est possible de limiter les choix lors du calcul de l'angle de rotation 2D en ne prenant que des valeurs autour de α_{n-1} .

La définition de tolérances sur les autres paramètres de pose est également possible, mais elle a moins d'impact dans l'algorithme que nous proposons et nous ne l'avons donc pas implantée.

3.3.2.5 Calcul des paramètres translationnels

L'étape précédente permet de trouver le contour du graphe d'aspect le plus proche des informations extraites de l'image en cours. Nous allons donc nous servir de ce contour pour mettre à jour le modèle actif. Ce nouveau modèle actif va aussi nous permettre de recalculer avec une meilleure précision la position 2D de l'objet dans l'image. Nous noterons M_n^g l'image binaire correspondant au contour d'occultation de l'objet 3D. Ce contour est calculé en utilisant les quatre paramètres de pose calculés à l'étape précédente: k_n^{theta} , k_n^{phi} , f_n et α_n .

Commençons par le paramètre le plus simple à calculer: la distance entre la caméra et l'objet. Rappelons que le graphe d'aspect est calculé à partir du modèle 3D situé à une distance Z_g pré-définie. Le facteur d'échelle calculé par l'algorithme est donc évalué en fonction de cette valeur. Nous aurons ainsi:

$$z_n = f_n \cdot Z_g \quad (\text{EQ 3.4})$$

Passons à présent au calcul des deux autres paramètres de translation, ceux parallèles au plan image. Nous avons noté X_{offset} et Y_{offset} les coordonnées du centre de projection dans l'image 2D, $Focalfactor$ la distance focale pour effectuer la projection de perspective et $AspectRatio$ le rapport entre ordonnées et abscisses dans l'image. Nous noterons $X_n^{gOffset}$ et $Y_n^{gOffset}$ les coordonnées de l'origine du repère de l'objet 3D dans l'image binaire M_n^g . D'après notre définition de M_n^g , le centre de projection et le centre du repère de l'objet 3D sont confondus. Ces coordonnées sont donc très faciles à évaluer au cours du déroulement de l'algorithme.

Pour calculer les deux paramètres de translation, il suffit d'évaluer le vecteur permettant de passer du centre de projection au centre du repère de l'objet. Ce vecteur peut se calculer dans le plan image puis être ramené dans l'espace 3D grâce aux équations de la projection de perspective inverse. Nous aurons ainsi:

$$\begin{aligned} x_n &= \frac{-(u_n + X_n^{gOffset} - X_{offset}) \cdot z_n}{Focalfactor} \\ y_n &= \frac{-(v_n + Y_n^{gOffset} - Y_{offset}) \cdot z_n}{AspectRatio \cdot Focalfactor} \end{aligned} \quad (\text{EQ 3.5})$$

3.3.2.6.2 Paramètres de rotation

Le calcul des angles de rotation est, quant à lui, un peu plus complexe parce qu'il faut tenir compte des effets de la projection de perspective. En effet, on ne peut pas déduire directement

les angles d'Euler $(\theta_n, \varphi_n, \psi_n)$ des trois angles fournis par notre algorithme. Nous noterons les angles calculés par l'algorithme $(\theta_n', \varphi_n', \psi_n')$. Nous aurons:

$$\begin{aligned}\theta_n' &= \frac{2\pi k_n^{theta}}{N_t} \\ \varphi_n' &= \frac{\pi k_n^{phi}}{N_p} \\ \psi_n' &= \alpha_n\end{aligned}\tag{EQ 3.6}$$

$(\theta_n', \varphi_n', \psi_n')$ sont évalués sans tenir compte de la translation puisque le graphe d'aspect est calculé en supposant que l'origine du repère de l'objet 3D est centré sur l'axe de projection. Déplacer le repère de l'objet par une translation de vecteur parallèle au plan image correspond approximativement à effectuer une rotation de l'objet 3D, si on observe uniquement les modifications du contour d'occultation.

Pour déterminer cette rotation, utilisons la figure 3.7 qui illustre le phénomène pour une translation selon l'axe des ordonnées. En généralisant cette approche au cas d'une translation quelconque parallèle au plan image, on démontre que la translation est équivalente, en termes de contour d'occultation, à une rotation de l'objet autour d'un axe perpendiculaire au vecteur de translation et d'angle β défini de la manière suivante:

$$\beta = \text{atan}\left(\frac{\sqrt{x_n^2 + y_n^2}}{z_n}\right)\tag{EQ 3.7}$$

L'axe de cette rotation peut être défini de la manière suivante:

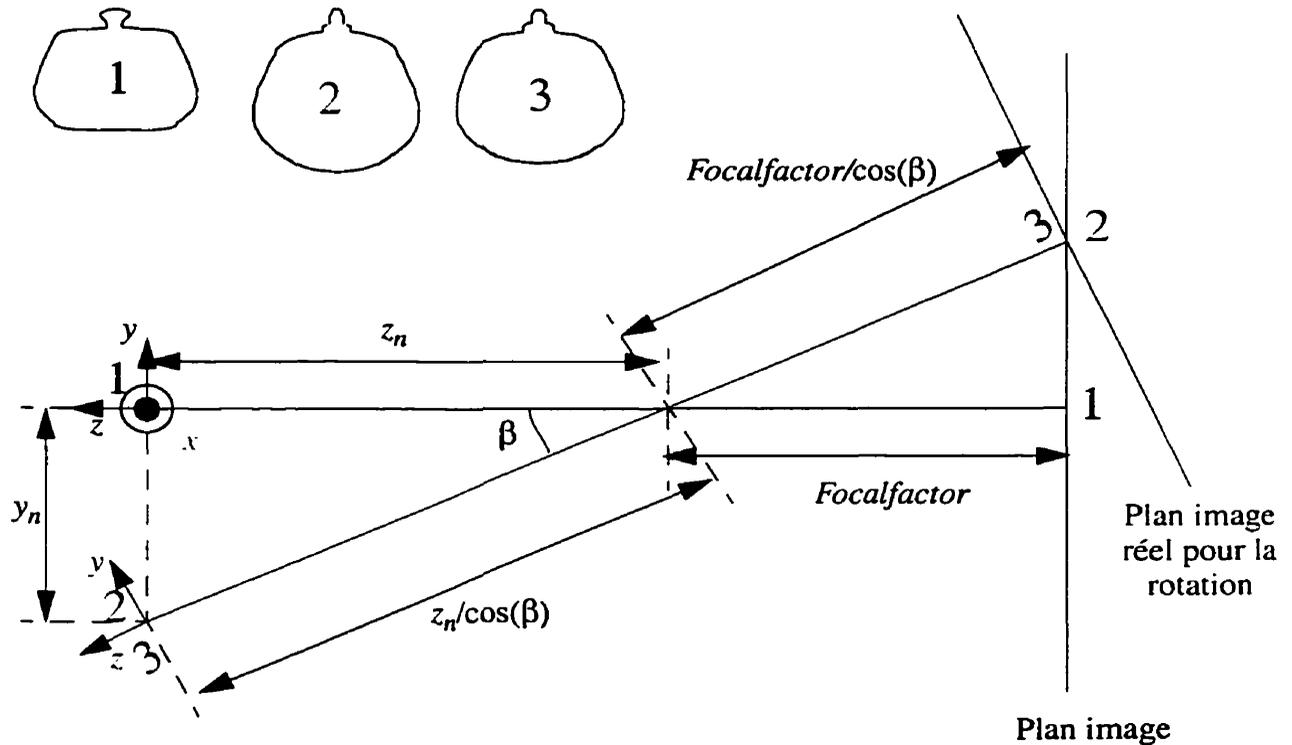


FIGURE 3.7 Calcul des paramètres de rotation.

Dans cette figure, nous allons supposer que nous nous trouvons dans le cas particulier où la translation selon l'axe (Ox) est nulle. La représentation est alors plus simple. Si on ne s'intéresse qu'à la forme du contour d'occultation dans le plan image (sans tenir compte de sa position dans celui-ci), on peut constater qu'une translation de y_n du repère de l'objet a approximativement le même effet en terme de contour d'occultation qu'une rotation d'angle $\beta = \text{atan}(y_n/z_n)$ selon l'axe (Ox).

En effet, prenons la théière de telle sorte que lorsqu'elle est dans la position 1 elle donne le contour d'occultation 1 dans le plan image. Si on effectue une translation de y_n de la théière (position 2), son contour dans le plan image prendra la forme représentée par le contour d'occultation 2. Enfin, si on effectue une rotation d'angle β selon l'axe (Ox) de la théière (position 3) on obtient le contour d'occultation 3 dans le plan image "incliné" (que nous avons appelé le plan image réel pour la rotation). On peut constater que les contours d'occultation 2 et 3 ne sont pas parfaitement identiques. Cette différence provient du fait que le contour 3 est calculé en utilisant un plan image "incliné".

En remplaçant la translation de y_n par une rotation d'angle β , on néglige donc une partie des effets de la translation. Cette approximation est valable tant que l'angle β reste relativement petit.

$$\sigma = \frac{1}{\sqrt{x_n^2 + y_n^2}} \begin{bmatrix} -y_n \\ x_n \\ 0 \end{bmatrix} \quad (\text{EQ 3.8})$$

À partir de l'angle β et de l'axe de rotation σ , on peut définir la rotation $Rot(\beta, \sigma)$ qui permet de retrouver les paramètres de pose réelle à partir de ceux calculés par notre algorithme. Ainsi, si on note $Rot(\theta_n', \varphi_n', \psi_n')$ la composition des rotations selon les angles d'Euler, nous aurons:

$$Rot(\theta_n, \varphi_n, \psi_n) = Rot(\beta, \sigma) \circ Rot(\theta_n', \varphi_n', \psi_n') \quad (\text{EQ 3.9})$$

L'équation ci-dessus illustre le principe de calcul des vrais angles d'Euler $(\theta_n, \varphi_n, \psi_n)$. En effet, en effectuant la composition de la rotation formée à partir des angles d'Euler calculés et de la rotation due à la translation, on obtient une nouvelle rotation. Celle-ci peut être décomposée de manière unique¹ en trois rotations selon les axes d'Euler.

Cette manière de procéder est, comme nous l'avons déjà mentionné, une approximation dans la mesure où la translation ne correspond pas uniquement à une rotation en terme de contour d'occultation. Cette approximation est toutefois valable lorsque la translation n'est pas trop grande. La figure 3.8 reprend les contours d'occultation introduits dans la figure 3.7 en les agrandissant. On peut constater qu'il y a effectivement une différence entre le contour d'occultation de l'objet translaté et le contour d'occultation calculé par rotation. En plus d'effectuer une rotation, il faudrait incliner le plan image (voir figure 3.7) pour que la correspondance soit parfaite. Les erreurs causées par cette approximation ont un effet sur le calcul de la pose de l'objet 3D. Ainsi, plus l'objet 3D aura subi une translation importante plus les angles d'Euler de sa pose seront imprécis.

1. Pour que la décomposition soit unique il faut que $0 \leq \varphi \leq \pi$.

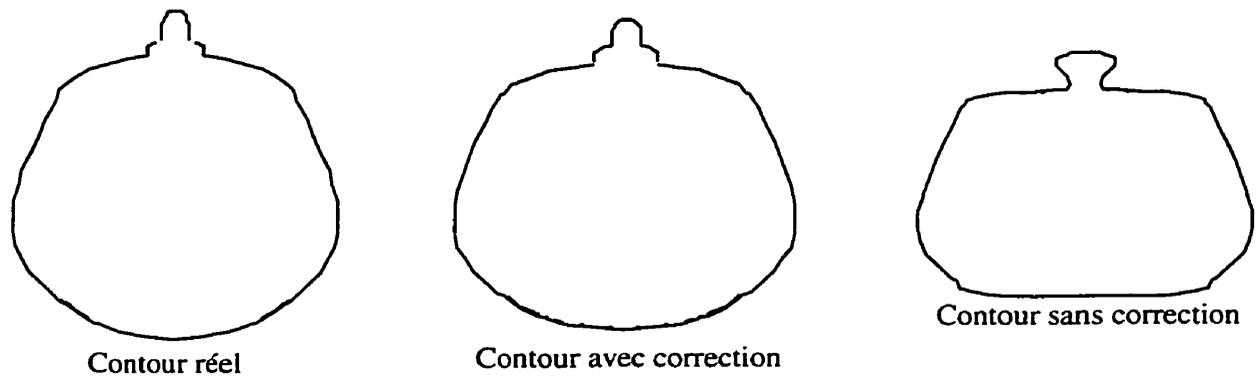


FIGURE 3.8 Erreur d'approximation à cause de la projection de perspective.

Cette figure illustre les effets de la projection de perspective lors d'un mouvement translationnel parallèle au plan image. Le contour de gauche correspond à la théière dans la position suivante: $(0, 10, 20, 90, 90, 0)$. En supposant qu'il n'y a aucune relation de dépendance entre les paramètres de translation et ceux de rotation (comme c'est le cas dans une projection orthogonale) et en recentrant l'objet 3D, il ne devrait y avoir aucun changement d'aspect du contour. Ce n'est malheureusement pas le cas puisque le contour de droite représente la théière dans la pose $(0, 0, 20, 90, 90, 0)$.

L'opération de recentrage nécessite par conséquent une modification des paramètres de rotation comme nous l'avons expliqué. Le contour du centre correspond à ce recentrage. La pose est alors $(0, 0, 20, 90, 63.43, 0)$.

On peut constater que la correction n'est pas parfaite puisque le contour réel et le contour corrigé ne sont pas parfaitement identiques. Cette erreur vient de l'approximation visant à éliminer l'effet d'inclinaison du plan image. Une translation parallèle au plan image ne peut pas être interprétée uniquement par une rotation. L'erreur, tout en restant minime, engendrera des imprécisions dans l'estimation de la pose.

3.3.2.7 Passage à l'image suivante et mise à jour du modèle actif

Nous venons d'expliquer comment se fait le calcul de la pose réelle de l'objet 3D. On peut alors générer le contour d'occultation de l'objet 3D en fonction de cette pose calculée. Nous le noterons M_n^{cal} .

Ce contour permet d'effectuer une mesure de comparaison avec les contours extraits de l'image M_n^{ext} . Cette mesure de comparaison donne alors une bonne idée de la justesse de l'estimation de pose.

La mise à jour du modèle actif M_n qui sera utilisée pour les calculs de l'image suivante peut se faire de deux manières différentes. On utilise soit le modèle calculé M_n^{cal} , soit le modèle correspondant aux contours extraits de l'image M_n^{ext} . Nous verrons les avantages et les inconvénients de chacune des deux approches lorsque nous analyserons les résultats.

3.3.2.8 Initialisation

Avant de passer aux résultats obtenus avec notre algorithme, il reste un dernier point à traiter qui correspond à l'initialisation de l'algorithme. L'initialisation par une méthode algorithmique est très délicate et ferait à elle seule le sujet d'un doctorat. Nous n'avons malheureusement pas pu mettre en place une méthode efficace, même si nous disposons de plusieurs pistes de recherche que nous développerons dans la section de la conclusion générale consacrée aux travaux futurs.

Nous nous contenterons donc d'utiliser ici la pose réelle de la première image pour effectuer l'initialisation.

3.4 Résultats

3.4.1 Séquences de tests utilisées

Les séquences d'images qui ont été utilisées pour évaluer les performances de l'algorithme sont représentées dans les figures 3.9 à 3.13. Ces séquences ont été générées à partir des modèles VRML d'objets 3D. Elles sont formées par un ensemble d'images au format «PPM» représentant l'objet 3D dans une position particulière. Il s'agit d'images de synthèse dont le plus grand intérêt est de connaître parfaitement la pose de l'objet 3D qui a permis de les générer.

Pouvoir générer une séquence d'images en maîtrisant tous les paramètres de pose permet de tester séparément différents aspects de notre algorithme. Ainsi, chacune des séquences que nous avons retenues pour la présentation des résultats possède des caractéristiques particulières.

La séquence de l'avion représente un mouvement avec faible transition. L'avion effectue un mouvement de rotation par rapport au plan image et une légère translation. Cette séquence permet de vérifier les performances de l'algorithme avec un mouvement relativement simple. Ce mouvement correspond exactement à un type de situations traitées par des algorithmes classiques de poursuite de cible.

La deuxième séquence correspond à un mouvement de rotation par rapport à un axe parallèle au plan image. Le mouvement reste relativement lent afin de ne pas modifier la silhouette trop rapidement. Le téléphone change progressivement d'aspect, mais aucun mouvement de translation n'est ajouté.

La troisième séquence correspond à un mouvement purement translationnel. Deux objets semblables sont présents dans la scène afin de vérifier les performances de l'algorithme en présence de plusieurs contours. Seul l'un des poissons est poursuivi lors des simulations. Les translations qu'il subit entre deux images successives sont relativement importantes. Jusqu'à l'image 9, les translations se font parallèlement au plan image. Ensuite, il y a en plus un effet de zoom dû à une composante translationnelle perpendiculaire au plan image.

La quatrième séquence analyse l'effet d'un panoramique. La voiture est au centre de l'image puis s'éloigne progressivement.

La dernière séquence que nous analyserons ici est celle de la théière. À un mouvement translationnel (essentiellement parallèle au plan image) s'ajoute des variations des angles de rotations. Les mouvements en translation sont importants puisqu'ils modifient remarquablement l'aspect de la théière dans l'image.

Aucune simulation n'a pu être faite à partir de séquences d'images réelles. De telles séquences sont difficiles à générer ou à obtenir puisqu'il faut avoir à la fois un modèle exact de l'objet qui se trouve dans la scène et les six paramètres de pose dans le repère de la caméra pour chaque image. Les résultats que nous obtenons avec les images de synthèse donnent cependant une très bonne idée des performances que nous pourrions obtenir dans des cas réels. Les performances que nous exposerons dans les paragraphes suivants pourront être vérifiées avec des séquences d'images réelles dans des travaux futurs.

3.4.2 Analyse des résultats

3.4.2.1 Évaluation des performances

Afin d'évaluer les performances de l'algorithme avec les différentes séquences décrites ci-dessus, nous allons tout simplement afficher les graphiques de comparaison entre la pose réelle et la pose calculée pour chacun des six paramètres. Ces courbes donneront une bonne idée de la précision de l'algorithme. Elles pourront également aider à comprendre certaines contre-performances. En plus des courbes, nous utiliserons également deux tableaux. L'un représentera la moyenne et l'écart type de l'erreur d'estimation de la pose pour chaque séquence. Ces valeurs permettront de savoir si notre estimation introduit un biais et si l'erreur est plus ou moins grande. Le second tableau présentera l'erreur absolue moyenne. Cette erreur donne une idée rapide de la qualité de l'algorithme.

Enfin, nous appuierons les résultats sur des figures représentant la superposition de l'image 2D et du contour d'occultation de l'objet 3D. Ces figures nous permettront d'évaluer qualitativement les performances de l'algorithme.

3.4.2.2 Analyse globale des résultats

Ainsi que le montrent les figures 3.14 à 3.18 et les tableaux 3.3 et 3.4, l'algorithme que nous proposons permet d'effectuer un suivi d'assez bonne qualité avec toutes sortes de mouvements.

L'erreur en translation est toujours très faible comme le biais dans l'estimation. Le fait que le biais ne soit pas nul est le résultat du nombre limité d'images par séquence et de la discrétisation. En effet, la position du contour actif dans l'image 2D ne peut être calculée qu'au pixel près. Cette erreur de discrétisation, ajoutée à celle commise lors du calcul du facteur d'échelle, introduit un léger biais dans l'estimation. L'erreur commise sur les paramètres de rotation est un peu plus grande. Si on se fie aux résultats du tableau 3.4, on peut constater que l'erreur absolue moyenne reste en dessous (ou dans le même ordre de grandeur) des facteurs de discrétisation utilisés pour le graphe d'aspect et pour le facteur de discrétisation angulaire.

TABLEAU 3.3 Erreur moyenne et écart-type pour chacun des paramètres de pose.

| Séquence s^a | x^b | | y^b | | z^b | | θ^b | | φ^b | | ψ^b | |
|------------------------|--------|-------|--------|-------|--------|------|------------|-------|-------------|-------|----------|-----------|
| | | | | | | | | | | | | |
| Avion (36 18 5) | -0.19 | 0.128 | -0.056 | 0.135 | -0.84 | 1.08 | -5.77 | 4.21 | 10.46 | 5.31 | 1.88 | 1.69 |
| Avion (72 36 2) | -0.19 | 0.074 | -0.11 | 0.072 | -1.11 | 0.63 | -3.16 | 2.69 | 7.45 | 3.88 | 1.23 | 0.91 |
| Téléphone (36 18 5) | -0.17 | 0.81 | 0.07 | 0.37 | -2.08 | 3.40 | 1.12 | 2.64 | 7.69 | 8.18 | -2.13 | 5.48 |
| Téléphone (72 36 2) | 0.009 | 0.22 | -0.01 | 0.12 | -0.12 | 1.81 | 2.19 | 2.30 | 1.10 | 7.16 | 1.29 | 2.23 |
| Poisson (36 18 5) | 0.85 | 4.37 | -0.2 | 3.30 | -21.93 | 9.09 | 7.09 | 9.63 | 9.79 | 11.83 | 0.65 | 2.32 |
| Poisson (72 36 2) | 1.49 | 2.41 | -0.46 | 2.06 | -10.67 | 9.95 | 10.79 | 8.34 | 13.54 | 14.15 | -0.71 | 2.98 |
| Voiture (36 18 5) | -0.072 | 0.039 | -0.22 | 0.024 | -1.03 | 0.21 | 3.28 | 0.02 | -1.32 | 0.07 | 1.46 | 0.01 2 |
| Théière (36 18 5) | 0.065 | 0.069 | -0.007 | 0.023 | -0.12 | 0.33 | 3.07 | 16.94 | -2.36 | 3.63 | 4.33 | 16.5 7 |
| Théière (72 36 2) | 0.028 | 0.037 | -0.021 | 0.044 | 0.099 | 0.35 | 10.47 | 18.40 | 0.95 | 3.70 | 9.92 | 17.2 1 |

- a. Les chiffres entre parenthèses qui suivent le nom de la séquence représentent respectivement N_t , N_p et Θ_s . Ainsi (36 18 5) correspond à une séquence testée avec un graphe d'aspect échantillonné tous les 10 degrés et avec un angle de discrétisation de 5 degrés pour la modélisation 2D.
- b. La première colonne représente l'erreur moyenne, la seconde l'écart-type.

Les erreurs angulaires dépendent aussi largement du type d'objet 3D évalué et de la zone angulaire autour de laquelle les variations se font. En effet, suivant le type d'objet, certaines zones du graphe d'aspect présentent de très faibles variations du contour d'occultation. Dans ces zones, une erreur de choix du bon modèle dans le graphe d'aspect peut entraîner des erreurs importantes en terme de mesure d'angle, sans que l'erreur ne soit importante en terme de contour d'occultation.

TABEAU 3.4 Erreur absolue moyenne pour chacun des paramètres de pose.

| Séquences ^a | x | y | z | θ | φ | ψ |
|------------------------|-------|-------|-------|----------|-----------|--------|
| Avion (36 18 5) | 0.19 | 0.12 | 0.99 | 6.52 | 10.46 | 2.09 |
| Avion (72 36 2) | 0.19 | 0.12 | 1.11 | 3.31 | 7.45 | 1.28 |
| Téléphone (36 18 5) | 0.37 | 0.15 | 2.52 | 2.32 | 8.03 | 2.29 |
| Téléphone (72 36 2) | 0.16 | 0.08 | 1.36 | 2.34 | 5.23 | 1.97 |
| Poisson (36 18 5) | 3.26 | 2.67 | 21.93 | 9.52 | 11.86 | 1.78 |
| Poisson (72 36 2) | 2.31 | 1.56 | 12.15 | 11.21 | 14.83 | 2.65 |
| Voiture (36 18 5) | 0.072 | 0.026 | 1.03 | 3.28 | 1.32 | 1.46 |
| Théière (36 18 5) | 0.075 | 0.021 | 0.26 | 13.86 | 3.32 | 12.54 |
| Théière (72 36 2) | 0.035 | 0.034 | 0.29 | 17.63 | 2.45 | 13.05 |

a. Les chiffres entre parenthèses qui suivent le nom de la séquence représentent respectivement N_t , N_p et Θ_s . Ainsi (36 18 5) correspond à une séquence testée avec un graphe d'aspect échantillonné tous les 10 degrés et avec un angle de discrétisation de 5 degrés pour la modélisation 2D.

Dans le cas du poisson, on peut constater que les erreurs sont plus importantes sur les deux premiers angles d'Euler. Ce sont ceux reliés directement au graphe d'aspect. Ceci illustre bien ce que nous venons d'expliquer. Par contre, le troisième angle d'Euler est relativement bien estimé.

Les erreurs plus importantes des paramètres de translation sont simples à expliquer. Alors que pour les autres séquences la distance caméra-objet est de l'ordre de quelques dizaines, elle est, pour le poisson, de l'ordre de quelques centaines. Il est donc normal d'obtenir une erreur dix fois plus importante pour le paramètre z . Les deux autres paramètres de translation étant calculés à partir de z , il est normal que l'ordre de grandeur de l'erreur soit proportionnel.

Dans le cas de la théière, les erreurs se retrouvent dans le premier et le troisième angle d'Euler. Ces erreurs sont relativement importantes. Toutefois, en y regardant d'un peu plus près, on se rend compte que ces deux angles se compensent. En effet, d'un point de vue purement mathématique, la décomposition d'une rotation en angle d'Euler n'est unique que lorsque le deuxième angle d'Euler est différent de 0 ou de 180 degrés. Dans ces deux cas particuliers, les rotations élémentaires correspondant au premier et au troisième angle d'Euler se font autour du même axe. Ainsi, les rotations définies par les triplets $(\theta, 180, \theta)$ et $(0, 180, 0)$ sont identiques, quelle que soit la valeur de θ . Dans l'exemple de la théière, le deuxième angle d'Euler est proche de 180 degrés. On peut constater aisément dans la figure 3.18 que les premier et troisième angles d'Euler se compensent. Dans ce cas, les erreurs apparemment importantes ne le sont pas.

Dans l'ensemble, nous pouvons donc conclure que l'algorithme que nous proposons est relativement robuste. Nous allons maintenant évaluer les résultats en nous concentrant sur certains aspects spécifiques.

3.4.2.3 Précision du graphe d'aspect et de la modélisation 2D

Les simulations montrent que, de manière générale, plus la précision du graphe d'aspect et de la modélisation est grande, plus les résultats sont bons. Cette remarque, qui semble aller de soi, n'est pas aussi triviale qu'il n'y paraît. En effet, en augmentant la précision du graphe d'aspect et de la modélisation, il devient plus difficile de discriminer deux contours voisins dans le graphe d'aspect. Les résultats obtenus prouvent encore une fois que la modélisation 2D introduite dans le chapitre 2 est très robuste et permet de différencier adéquatement des contours très ressemblants.

3.4.2.4 Type de mouvement

On peut constater, que dès que le mouvement contient une composante translationnelle parallèle au plan image, les erreurs d'estimation des angles d'Euler sont plus importantes. Les séquences du poisson et de la théière le prouvent, même si dans le cas de la théière, une part des erreurs est imputable à la définition même des angles d'Euler, comme nous l'avons déjà souligné.

La source de ces erreurs est évidente. Il s'agit de l'approximation que nous faisons par rapport à la projection de perspective. Comme nous le décrivons dans le paragraphe 3.3.2.6.2, nous réduisons l'effet d'une translation parallèle au plan image à une simple rotation. Il faudrait toutefois tenir compte aussi de l'inclinaison du plan image. La figure 3.8 illustre l'erreur que cette approximation introduit. Cette erreur se traduit directement par une erreur dans l'estimation des angles d'Euler et il est relativement difficile de corriger cet effet. On peut toutefois remarquer que plus l'estimation de la translation parallèle au plan image est grande, plus l'incertitude sur le calcul des angles d'Euler est importante.

Il est important aussi de noter qu'un mouvement très saccadé en translation peut être traité par l'algorithme. L'estimation de la pose peut se faire sans aucune difficulté. Par contre, dans le cas d'un mouvement de rotation ou de zoom trop important, l'algorithme risque de devenir inopérant parce que la silhouette de l'objet aura changé plus que ce que l'algorithme peut tolérer. Nous verrons dans le prochain paragraphe que c'est le facteur de dilatation qui permet de gérer la tolérance de l'algorithme en terme de changement de silhouette du contour d'occultation.

3.4.2.5 Importance du facteur de dilatation

Comme le montre la figure 3.19, le facteur de dilatation est très important. En effet, si ce facteur est trop petit, le contour extrait de l'image 2D devient progressivement incomplet. Or, la robustesse de l'algorithme dépend grandement de la qualité du contour extrait. Un contour incomplet entraîne rapidement des erreurs d'estimation de pose importantes.

Le facteur de dilatation permet de tenir compte des changements d'aspects du contour d'occultation de l'objet 3D entre deux images successives. Choisir un facteur trop petit rend l'extraction de l'information de contour impossible. Par contre, le choisir trop grand risque d'inclure des contours voisins n'ayant rien à voir avec l'objet 3D recherché et de perturber la reconnaissance. Il est important d'ajuster ce paramètre en fonction des changements de silhouette acceptables. Une valeur empirique entre 5 et 10 pixels permet généralement d'obtenir de bons résultats.

3.4.2.6 Méthode de mise à jour du contour actif

Nous avons indiqué dans le paragraphe 3.3.2.7 que la mise à jour du modèle actif pouvait se faire de deux manières distinctes. La première consiste à utiliser le contour d'occultation de l'objet 3D en tenant compte de la pose calculée. Nous l'avons noté M_n^{cal} . Le problème de cette approche est qu'une petite erreur de pose peut entraîner une déformation de M_n^{cal} par rapport au contour réel. En passant à l'image suivante, l'extraction des contours peut alors s'avérer incomplète. En effet, la dilatation de M_n^{cal} ne permet pas d'englober toutes les informations pertinentes sur le contour d'occultation. Or, nous venons de voir que l'algorithme ne donne pas de bons résultats lorsque les contours à tester sont incomplets.

Nous avons utilisé cette approche avec la séquence de l'avion. Le décrochage de l'algorithme est assez rapide parce qu'une petite erreur de pose entraîne un grand changement de l'aspect de l'avion. Les ailes deviennent ou un peu trop courtes ou un peu trop longues. Par contre, dans la séquence de la voiture ou du poisson, les performances sont restées les mêmes. En effet, dans ces deux cas les erreurs de pose n'ont pas entraîné un changement significatif du contour d'occultation.

La seconde méthode, celle que nous avons utilisée dans toutes les simulations que nous présentons ici, consiste à prendre comme modèle actif pour l'image suivante les informations extraites de l'image, c'est-à-dire M_n^{ext} . Cette approche n'est pas sensible aux légères erreurs de pose, mais elle possède un autre inconvénient. Elle inclut progressivement des points de contours qui

n'ont pas nécessairement de lien avec le contour d'occultation dans le modèle actif. Au lieu d'obtenir des contours incomplets, on obtient alors des contours bruités. Le paragraphe suivant donne un exemple de décrochage dû à ce type de problème.

En conclusion, on peut dire que chacune des deux méthodes possède certains inconvénients, mais qu'il est néanmoins possible d'obtenir de bons résultats ainsi que le montrent les simulations. Il serait cependant important de développer une méthode de mise à jour du contour actif plus robuste. L'une des avenues à explorer est l'ajustement des paramètres de pose avec une méthode comme celle de Lowe (voir paragraphe 3.2.3.2). En effet, la pose que nous estimons possède une précision limitée à cause de la discrétisation du graphe d'aspect et de la modélisation 2D. Utiliser l'algorithme de Lowe en prenant comme pose de départ le résultat de notre méthode serait un excellent moyen pour rendre le contour M_n^{cal} plus juste. La méthode de Lowe ne peut pas être utilisée directement à cause des mouvements de translation importants qui peuvent survenir dans nos séquences. Il s'agirait cependant d'une excellente méthode d'ajustement.

3.4.2.7 Importance de l'exactitude des contours extraits.

Jusqu'ici, nous avons toujours utilisé le principe de segmentation en objet pour obtenir les contours extraits de l'image. Ceci donne de bons résultats dans le cas particulier des séquences que nous testons. Il faut cependant avouer qu'il s'agit ici du point faible de notre approche puisque nos séquences sont très simples et que dans le cas d'images réelles, l'extraction peut s'avérer plus difficile.

Dans la figure 3.20, nous avons utilisé une extraction de contours basée sur l'algorithme d'extraction d'arêtes de Canny. Le contour d'occultation extrait est perturbé par des arêtes appartenant à l'avion, mais qui ne font pas partie du contour d'occultation proprement dit. Ces perturbations deviennent rapidement assez importantes pour fausser entièrement la prédiction de la pose.

Des développements plus importants pour rendre l'algorithme performant dans un monde réel devront être faits au niveau de l'extraction des contours d'occultation à partir d'une image. Cette étape a été abordée sommairement dans le cadre de ce doctorat, mais mériterait d'être approfondie ailleurs. Une combinaison intelligente de méthodes de segmentation en objets avec une détection d'arêtes par une méthode de Canny devrait toutefois permettre d'aboutir à une première ébauche de solution. De plus, en utilisant le fait qu'un contour d'occultation est toujours formé d'un ou de plusieurs contours fermés, il devrait être possible d'épurer l'image de contours précédemment calculée et d'aboutir à une qualité semblable à celle des contours obtenus pour nos simulations.

3.4.3 Considération de complexité algorithmique

La complexité de l'algorithme que nous proposons dépend de plusieurs étapes. Ainsi l'extraction des contours peut devenir relativement complexe, en particulier lorsque la méthode de Canny est utilisée. Cette étape peut prendre 2 secondes sur une station Sparc 10 pour une image de 640 par 480 pixels (aucune optimisation particulière du code n'est faite).

La deuxième étape importante en terme de complexité algorithmique concerne le calcul de la «distance map» (voir paragraphe 3.2.2.1). Ce calcul est indispensable pour l'algorithme de Huttenlocher. La complexité est proportionnelle à la taille de l'image et elle prend 2 secondes en moyenne sur une station Sparc 10 pour une image de 640 par 480 pixels.

L'algorithme de Huttenlocher n'est pas un noeud en terme de complexité parce qu'il existe de nombreuses optimisations du code possibles grâce à l'utilisation de la «distance map» et à des propriétés intrinsèques de l'algorithme.

Il en va de même pour la recherche dans le graphe d'aspect. En effet, non seulement la recherche ne se fait pas dans le graphe d'aspect au complet, mais en plus il est possible d'utiliser la méthode par analyse en composantes principales qui accélère considérablement la recherche sans trop affecter les performances.

3.5 Conclusion et travaux futurs

Dans ce chapitre, nous avons présenté l'algorithme qui permet d'effectuer une poursuite de cible. Il s'agit du point d'aboutissement de tous les outils développés dans les chapitres précédents. Cet algorithme est prometteur puisqu'il permet notamment de s'affranchir de toute étape de prédiction du mouvement et qu'il est efficace même dans des cas de mouvements importants.

Les résultats montrent qu'il est possible d'effectuer une poursuite de cible dans une séquence d'images 2D sans décrochage. La faisabilité d'une telle approche est démontrée, mais il reste un point important à traiter afin de transformer cette faisabilité en une véritable réussite: il faut trouver une méthode efficace d'extraction des contours d'occultation. La mise au point d'une telle méthode semble possible, mais nécessitera, selon nous, un temps de développement minimum.

La mise au point d'une méthode d'extraction efficace des contours d'occultation ouvre d'autres avenues pour notre algorithme. Elle permet en particulier d'imaginer une méthode de recherche exhaustive parmi tous les contours d'occultation extraits. Cette méthode s'acquitterait de l'étape de recherche du contour de l'image précédente par la méthode de Huttenlocher. L'initialisation de l'algorithme serait alors chose facile et la contrainte de changement lent de forme du contour (voir section 3.3.2.2.1) pourrait être supprimée.

Un autre axe de développement consisterait à reprendre l'algorithme de Lowe (voir paragraphe 3.2.3.2) pour raffiner les estimations de pose à chaque étape. La combinaison de notre méthode et de celle de Lowe devrait donner des résultats d'une très bonne précision.

L'apport de notre méthode dans le domaine de la poursuite de cible et de l'estimation de pose relève d'une utilisation originale du graphe d'aspect. L'algorithme que nous avons proposé possède, en plus de ses propriétés actuelles, un grand potentiel d'amélioration.

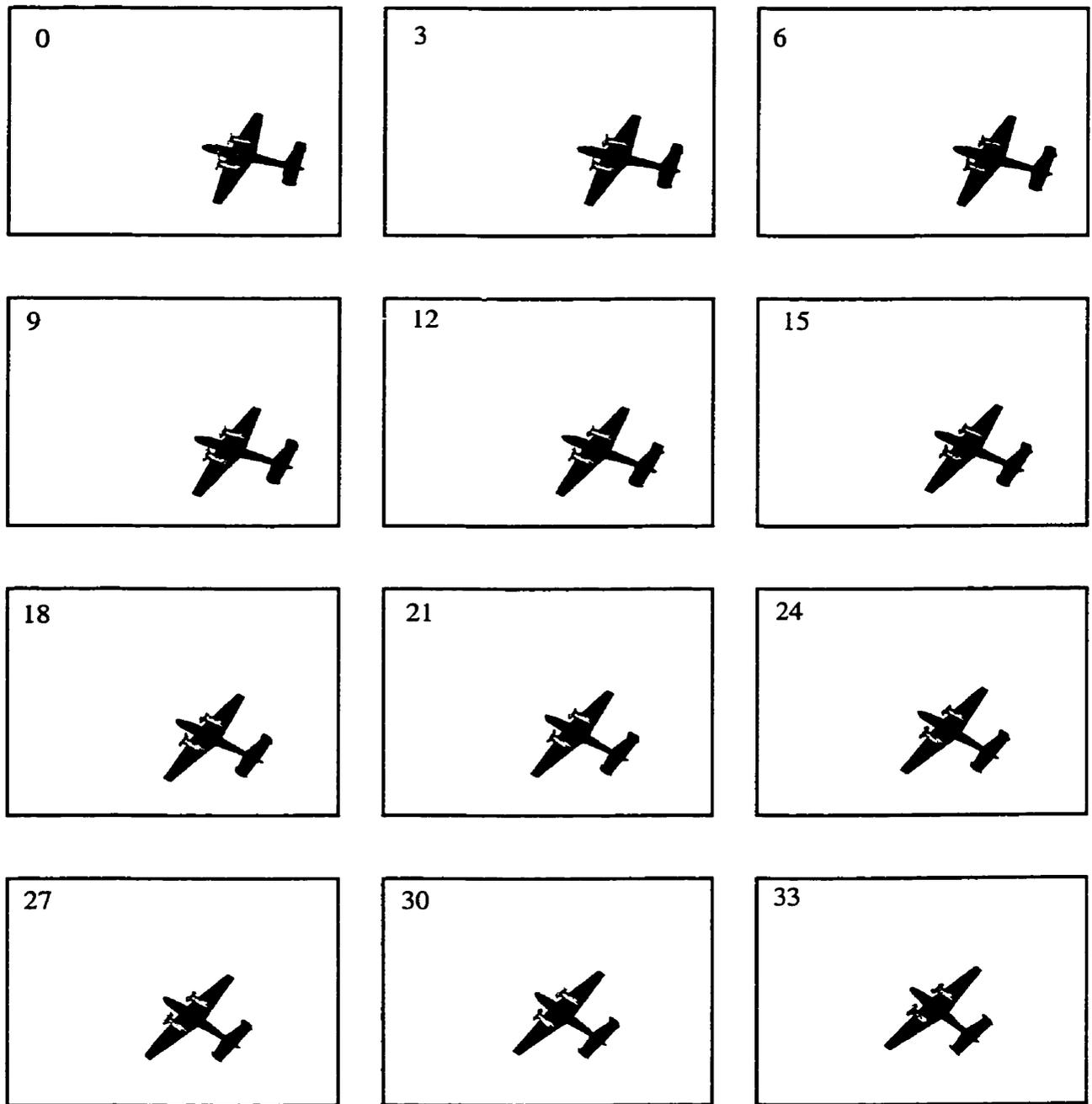


FIGURE 3.9 Séquence d'images 1: l'avion.

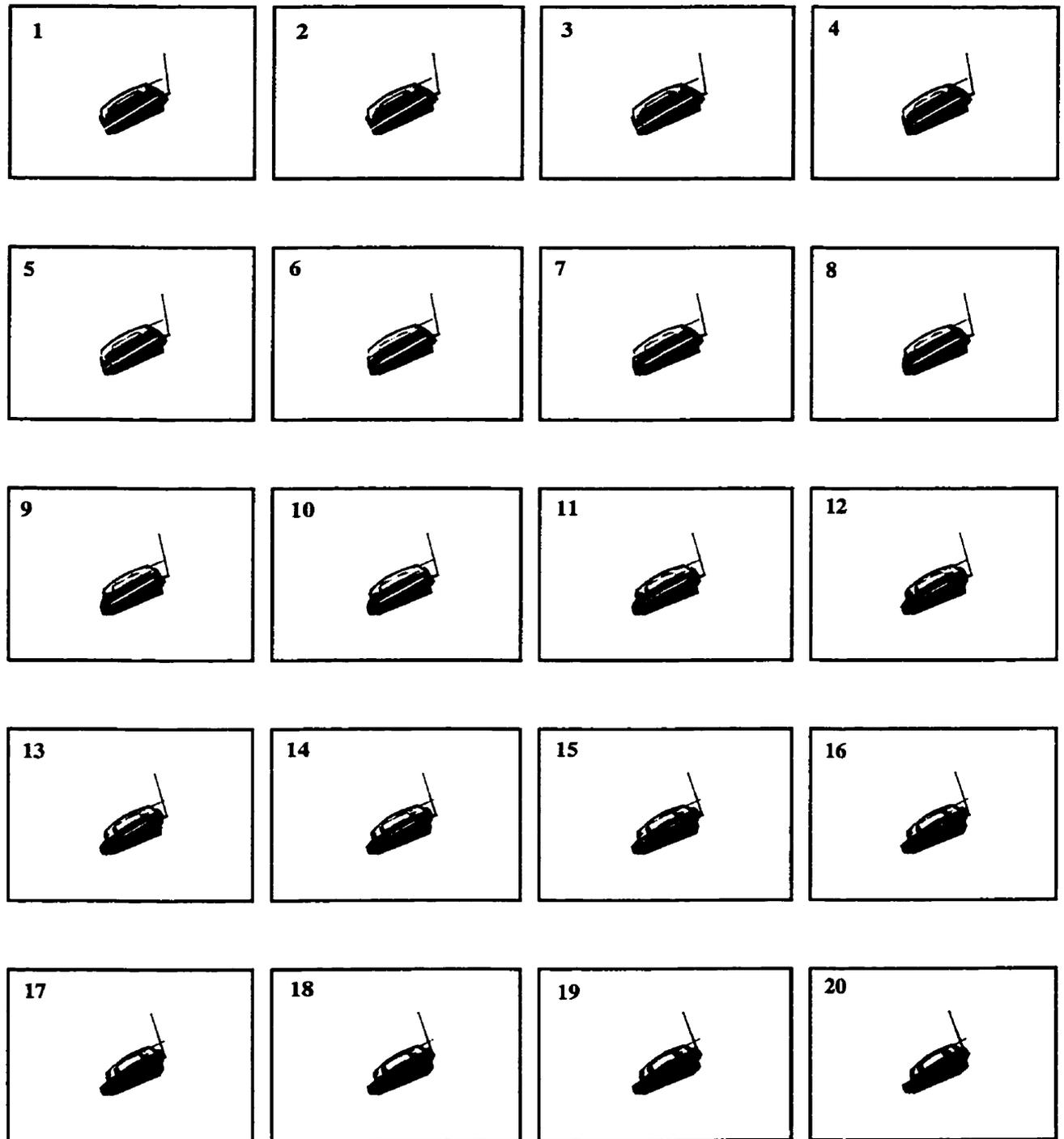


FIGURE 3.10 Séquence d'images 2: le téléphone (mouvement de rotation).

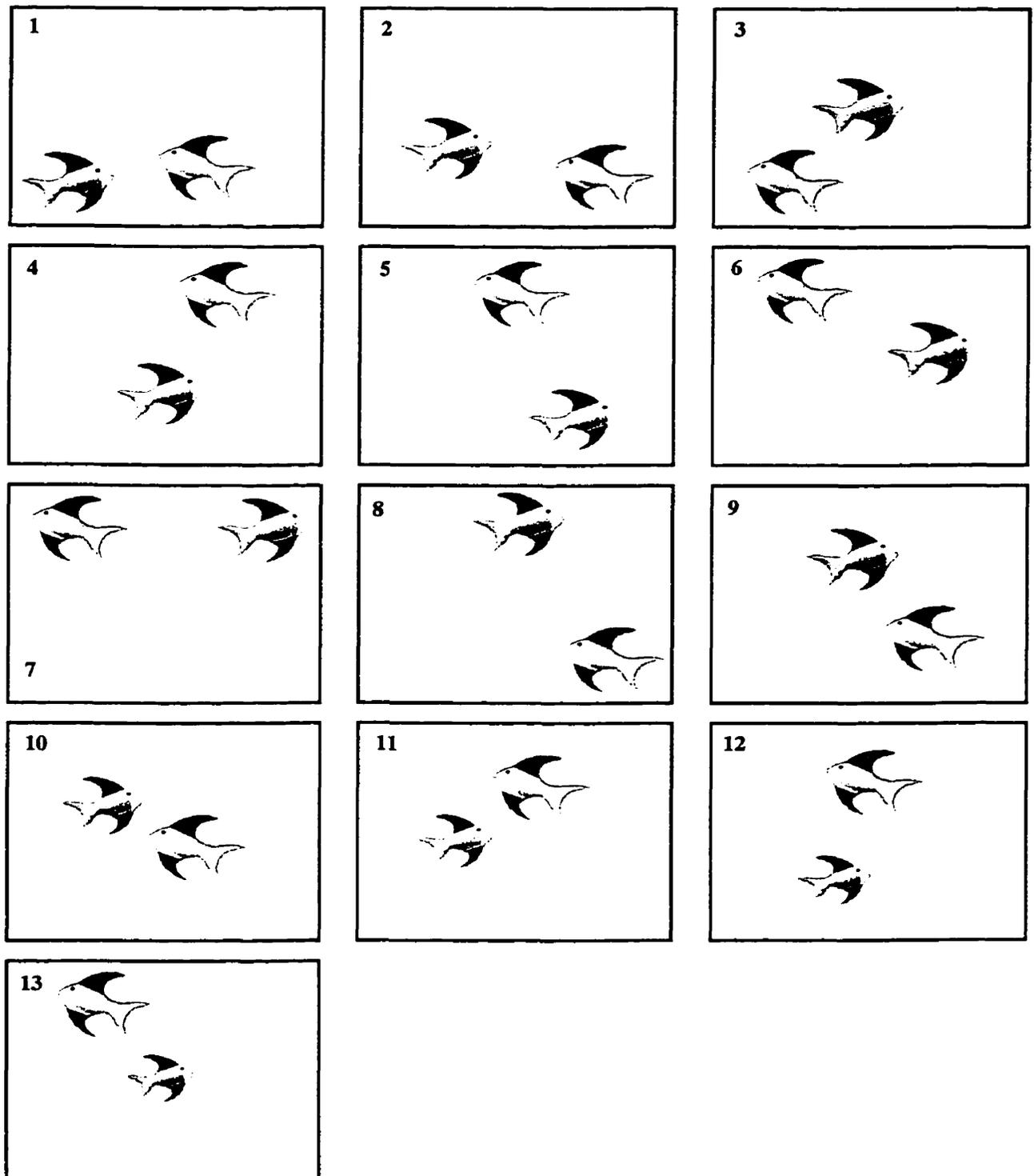


FIGURE 3.11 Séquence d'images 3: le poisson (mouvement de translation)

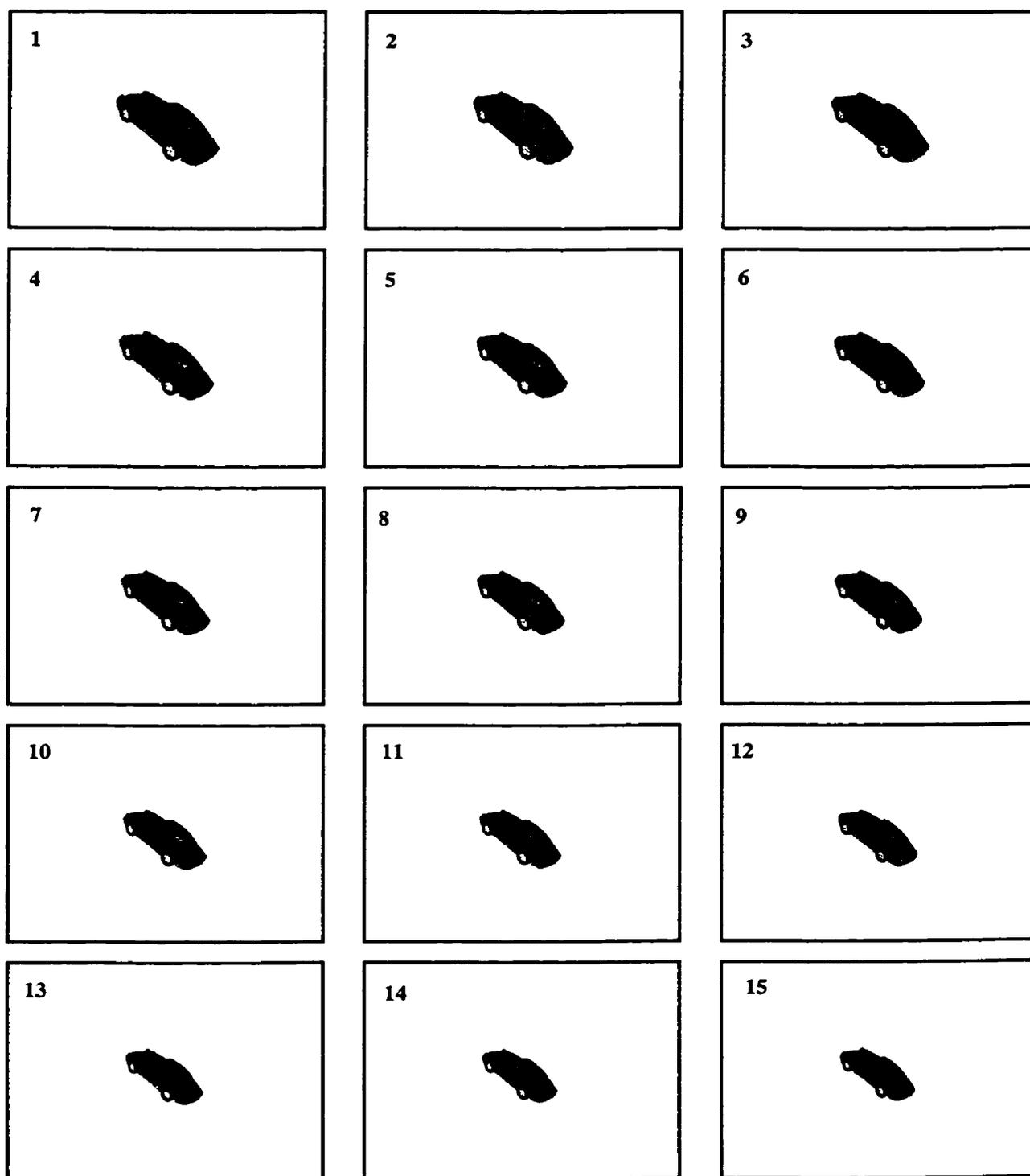


FIGURE 3.12 Séquence d'images 4: la voiture (mouvement de zoom).

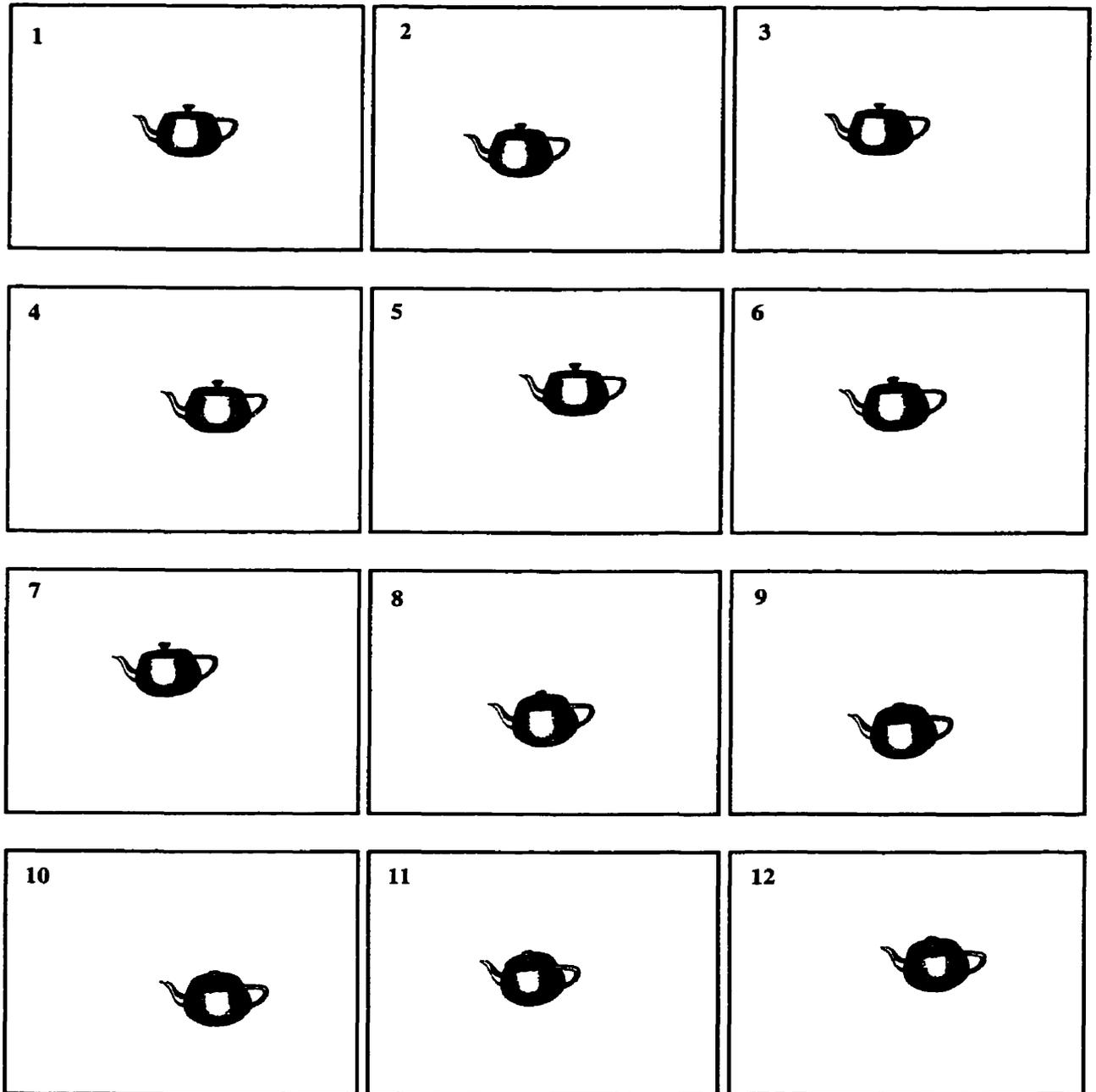
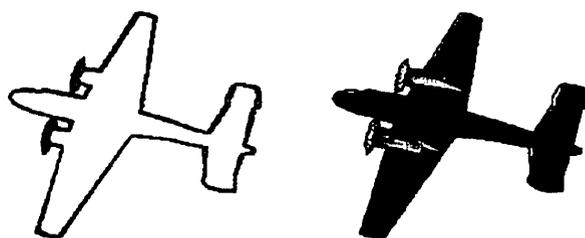
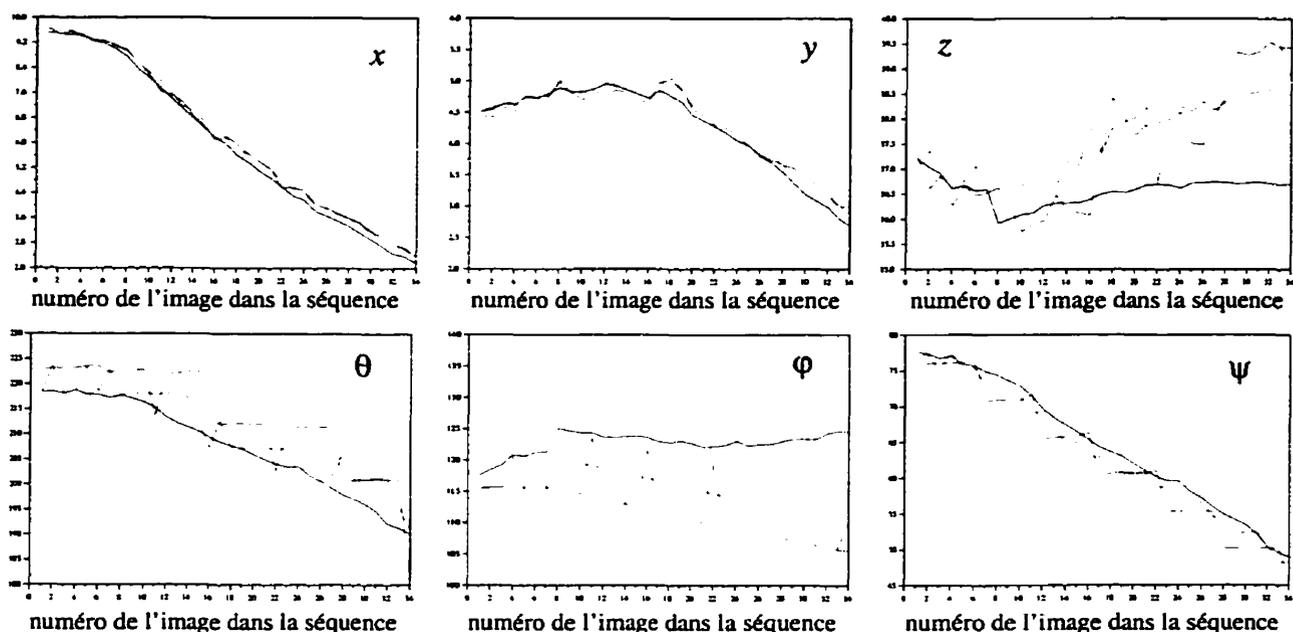
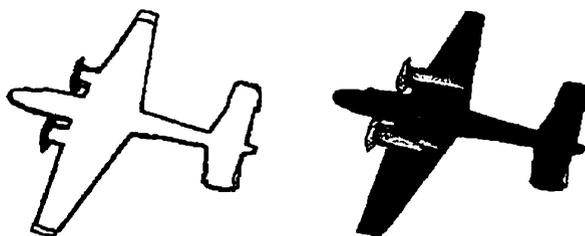


FIGURE 3.13 Séquence d'images 5: la théière (mouvement saccadé: rotation et translation).



Résultats obtenus avec la première image



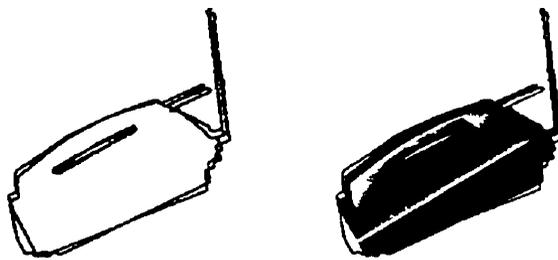
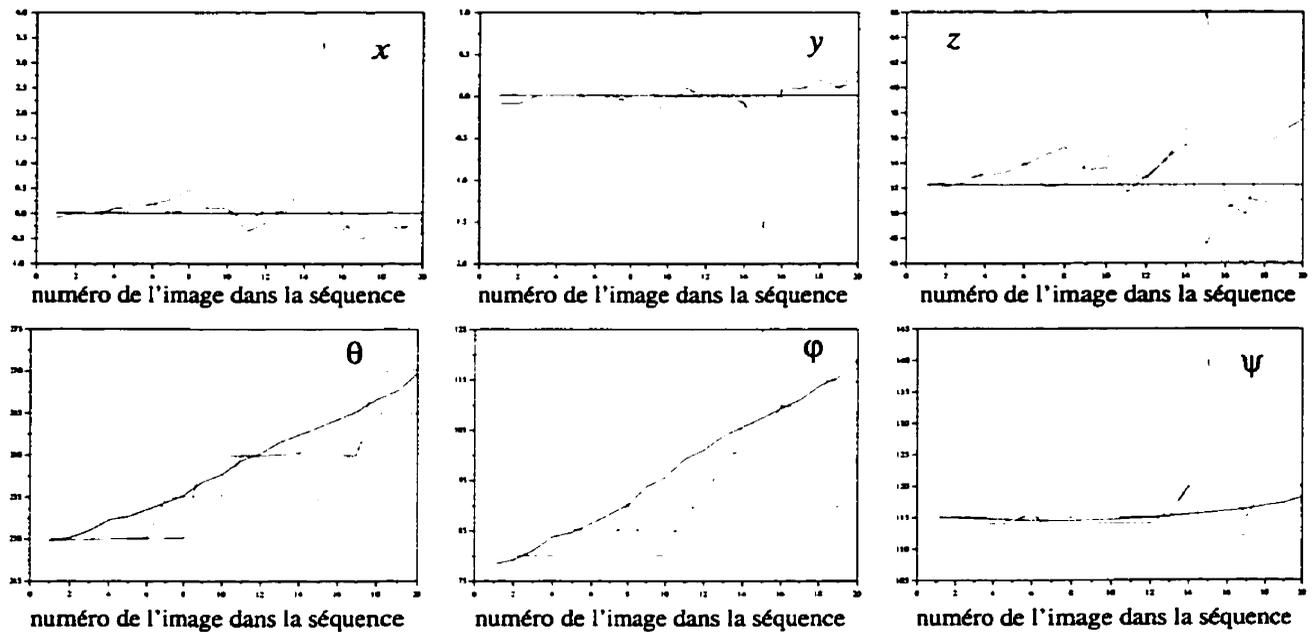
Résultats obtenus avec la deuxième image

FIGURE 3.14 Résultats avec la séquence de l'avion avec des précisions différentes pour le graphe d'aspect et la paramétrisation.

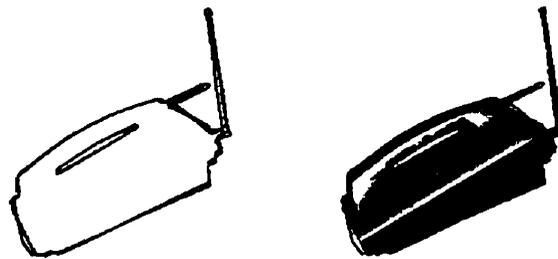
Les graphiques ci-dessus représentent les courbes des six paramètres de pose. Les courbes bleues correspondent à la pose réelle de l'objet alors que les courbes rouges et vertes représentent l'estimation de cette pose avec des graphes d'aspect et des paramétrisations 2D de résolutions différentes.

Ainsi, nous aurons une résolution de 10 degrés (respectivement 5) pour le graphe d'aspect de la courbe rouge (respectivement verte) et un angle de discrétisation de 5 degrés (respectivement 2) pour la paramétrisation. Le facteur de dilatation a été fixé à 8.

Les images de gauche illustrent le principe de fonctionnement de l'algorithme. La zone en jaune correspond aux masques obtenus par dilatation du contour actif de l'image précédente. Le contour en rouge correspond à la projection de l'objet 3D avec la pose estimée et le contour en vert correspond aux contours extraits de l'image. À droite, le contour estimé (en rouge) est superposé à l'image 2D.



Résultats obtenus avec la première image



Résultats obtenus avec la deuxième image

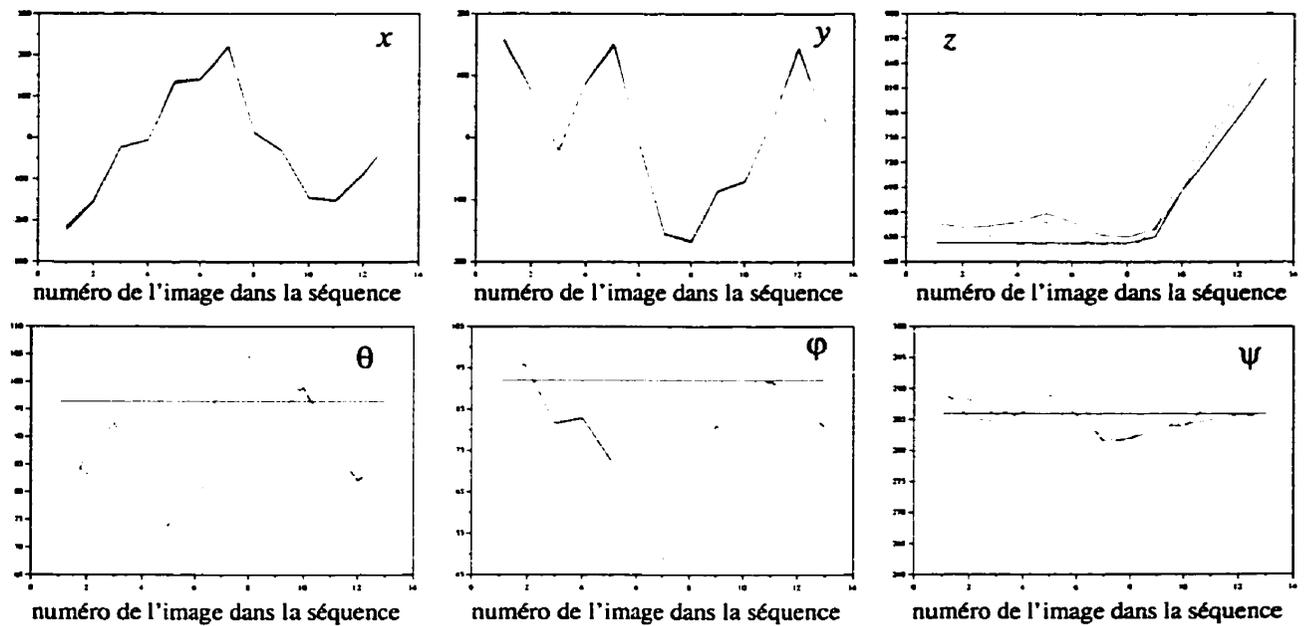
La description des graphiques ainsi que des images de cette figure sont les mêmes que pour la figure 3.14.

On peut constater lors de cette simulation que, pour la quinzième image et à la précision la plus faible, une erreur en translation assez importante apparaît. Cette erreur se traduit aussi dans le paramètre de rotation 2D. Cette erreur disparaît en partie lorsqu'on augmente la précision du graphe d'aspect et de la paramétrisation 2D.

Ceci montre qu'il est important d'avoir une précision suffisante du graphe d'aspect afin de ne pas négliger certains aspects importants du graphe.

FIGURE 3.15 Résultats avec la séquence du téléphone avec des précisions différentes pour le graphe d'aspect et la paramétrisation.

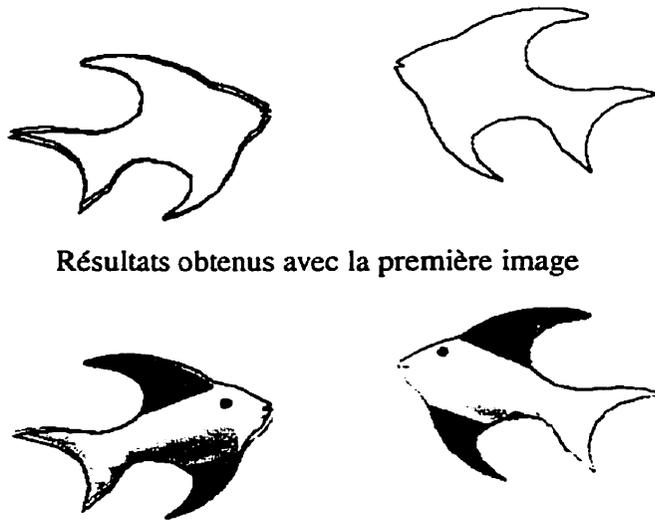
Comme le montrent les courbes bleues des graphiques, le mouvement étudié ici est un mouvement de rotation pure.



On peut constater d'après les graphiques ci-dessus que les deux paramètres qui ne sont pas très bien estimés sont ceux liés au graphe d'aspect.

En fait, l'aspect du poisson dans cette zone du graphe d'aspect ne varie pas énormément. Ainsi les erreurs d'estimation qui semblent importantes ne le sont pas en terme de contour d'occultation.

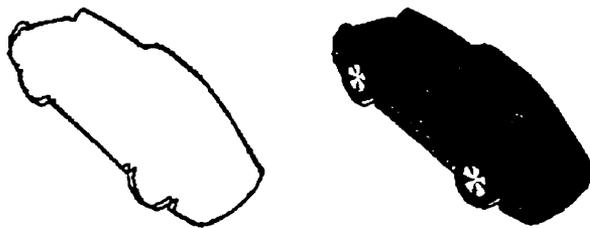
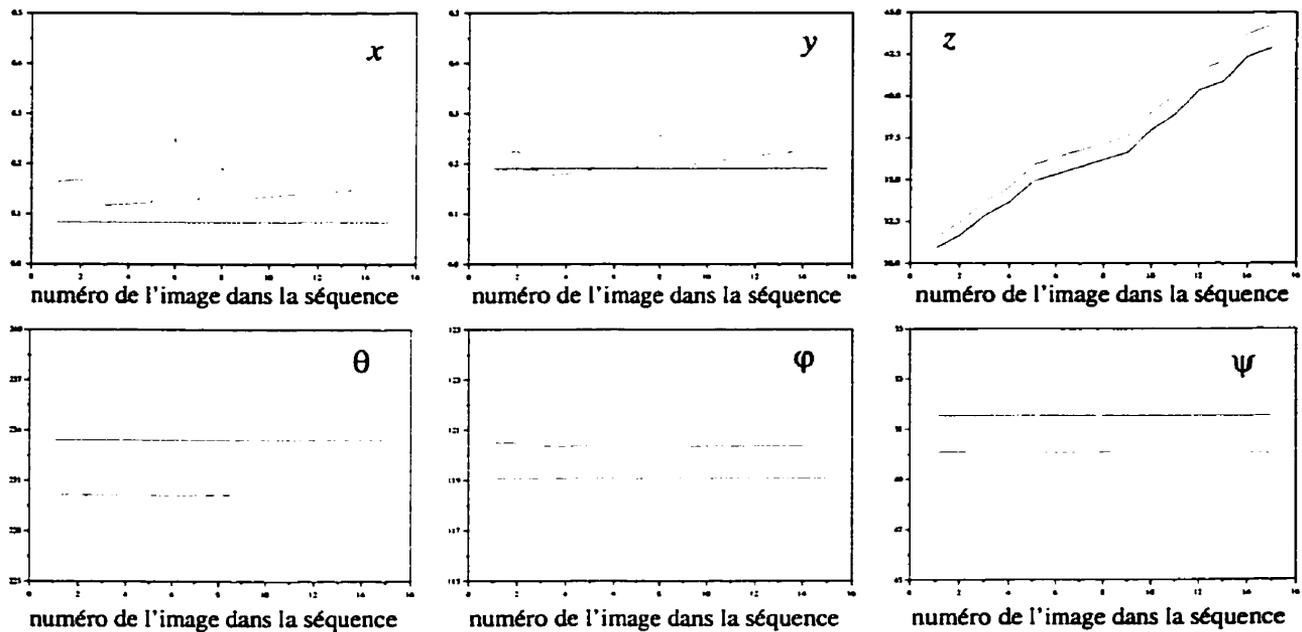
La précision que l'on peut obtenir sur ces deux paramètres est donc limitée pour cet aspect particulier du poisson.



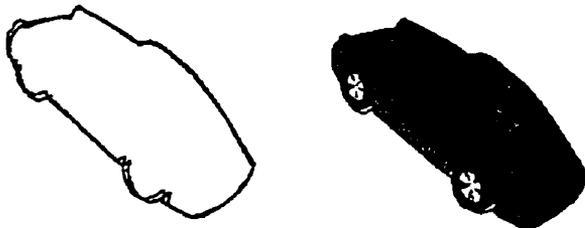
Résultats obtenus avec la première image

FIGURE 3.16 Résultats avec la séquence du poisson avec des précisions différentes pour le graphe d'aspect et la paramétrisation.

La description des graphiques ainsi que des images de cette figure sont les mêmes que pour la figure 3.14. Le mouvement d'un seul des deux poissons est étudié. Il s'agit d'un mouvement translationnel saccadé et relativement important.



Résultats obtenus avec la première image



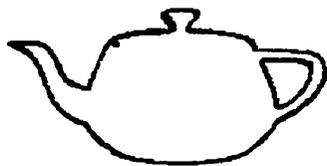
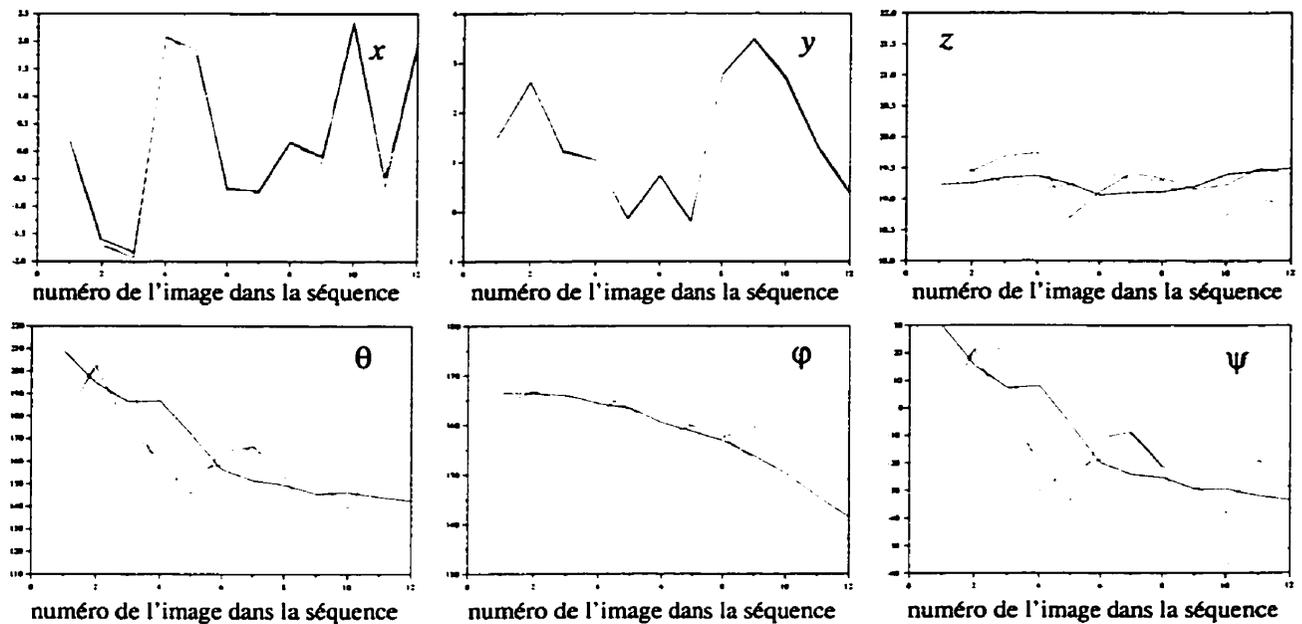
Résultats obtenus avec la deuxième image

FIGURE 3.17 Résultats avec la séquence de la voiture.

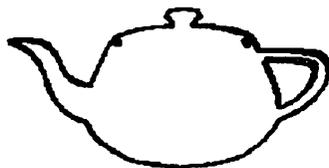
Les graphiques ci-dessus représentent les courbes des six paramètres de pose. Les courbes bleues correspondent à la pose réelle de l'objet alors que les courbes rouges représentent l'estimation de cette pose.

La résolution du graphe d'aspect est de 10 degrés et l'angle de discrétisation est de 5 degrés. Le facteur de dilatation a été fixé à 10 et nous avons utilisé une détection de contour par segmentation en objet.

La description des images de cette figure est la même que pour la figure 3.14. Le mouvement correspond ici à un panoramique de la caméra. L'erreur angulaire est constante tout au long de la séquence et laisse supposer qu'il s'agit essentiellement d'un problème de discrétisation insuffisante du graphe d'aspect.



Résultats obtenus avec la première image



Résultats obtenus avec la deuxième image



Les résultats de cette simulation nécessitent certaines précisions. On pourrait en effet croire que l'estimation des premier et troisième angles d'Euler est relativement mauvaise. En fait, il n'en est rien. Il s'agit simplement d'un cas particulier pour lequel ces deux rotations se font approximativement autour du même axe et se compensent mutuellement. Ceci arrive lorsque le

second angle d'Euler est proche d'une valeur de 0 ou de 180 degrés. On pe deux angles évoluent exactement de la même manière.

FIGURE 3.18 Résultats avec la séquence de la théière avec des précisions différentes pour le graphe d'aspect et la paramétrisation.

La description des graphiques ainsi que des images de cette figure sont les mêmes que pour la figure 3.14. Le mouvement de la théière est un mélange de rotation et de translation saccadée. Il s'agit d'un mouvement très général.

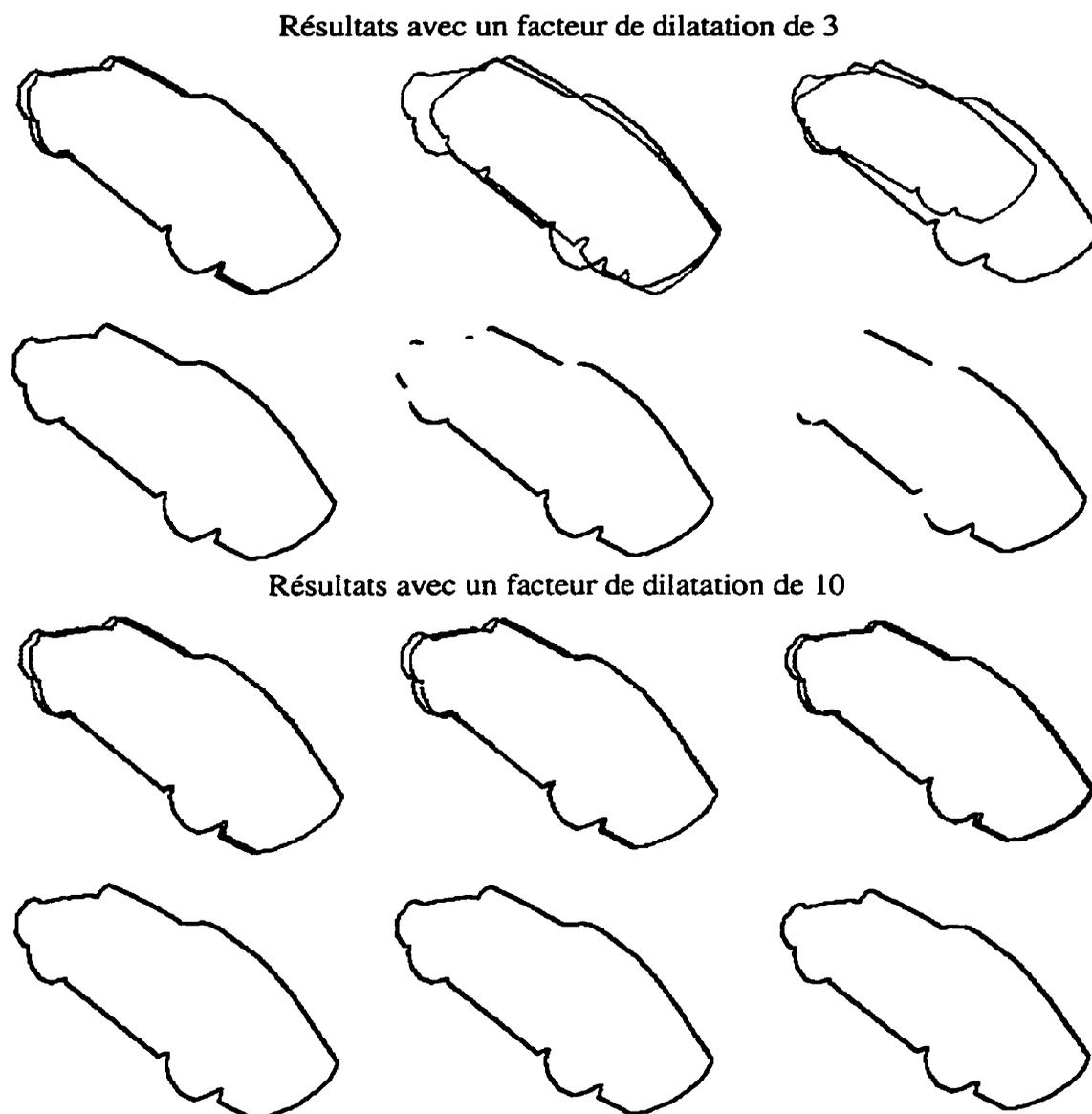
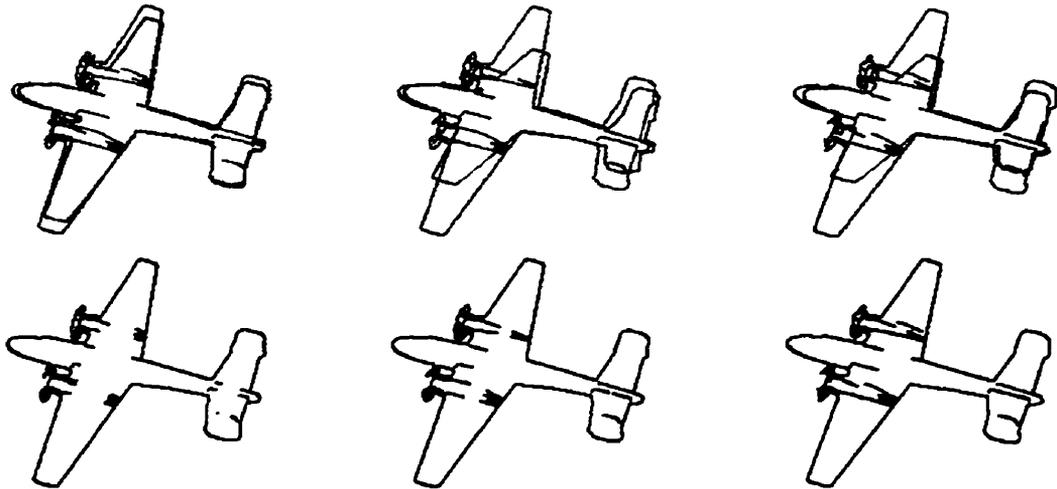


FIGURE 3.19 Ajustement du facteur de dilatation.

Dans cette figure, les trois premières images de la séquence de la voiture ont été testées avec deux facteurs de dilatation distincts (respectivement 3 et 10). Pour chaque facteur de dilatation, les images du haut représentent la zone de dilatation en jaune, le contour estimé en rouge et le contour extrait par segmentation en vert. Les images du bas, quant à elles, correspondent aux contours extraits notés M_n^{ext} dans notre taxonomie. On constate que lorsque le facteur de dilatation n'est pas assez grand, on perd rapidement de l'information dans M_n^{ext} . Or, cette information est capitale pour que l'algorithme donne de bonnes performances.

Résultats avec une extraction de contour de type Canny



Résultats avec une extraction de contour de type segmentation

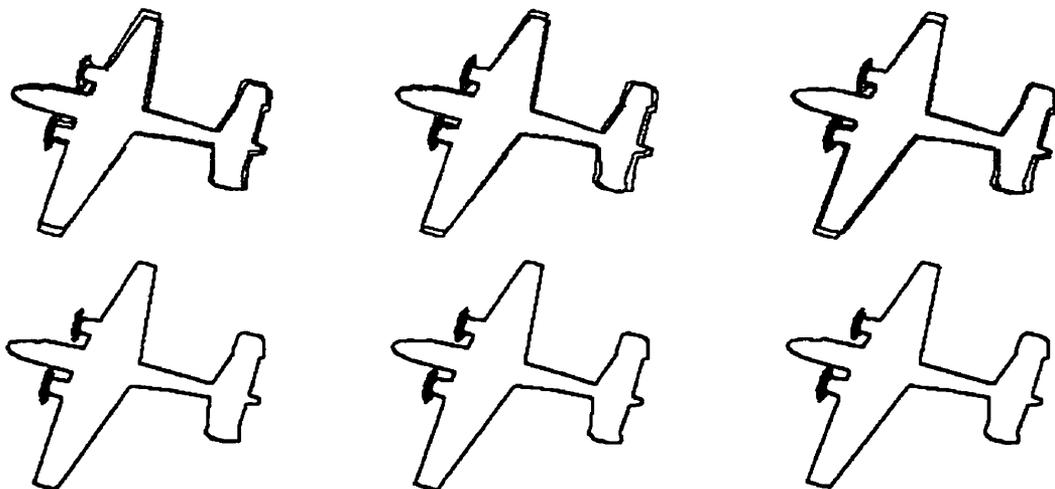


FIGURE 3.20 Résultats obtenus avec une extraction de contour de type détection d'arêtes ou de type segmentation.

Dans cette figure, les trois premières images de la séquence de l'avion ont été testées avec une détection de contour de type Canny et une autre de type segmentation (par seuillage simple). Pour chacune des méthodes, les images du haut représentent la zone de dilatation en jaune, le contour estimé en rouge et le contour extrait par segmentation en vert. Les images du bas, quant à elles, correspondent aux contours extraits notés M_n^{ext} dans notre taxonomie. On constate que lorsque la méthode de Canny est utilisée, M_n^{ext} contient des arêtes qui n'appartiennent pas au contour d'occultation. Ces informations "en trop" perturbent le bon fonctionnement de l'algorithme et prennent de plus en plus d'importance.

Conclusion

Principales contributions

L'objectif de la recherche présentée dans cette thèse était la poursuite d'une cible connue et l'estimation de sa pose à partir d'une séquence d'images 2D. La cible devait être un objet rigide et les occultations, même partielles, n'étaient pas autorisées. Nous avons présenté une méthode qui répond aux attentes que nous pouvions avoir. L'algorithme proposé permet en effet de suivre les mouvements, même saccadés, d'un objet 3D et de calculer les six paramètres de pose. La précision et la robustesse de notre approche sont raisonnables et sa simplicité laisse présager un bel avenir.

La première contribution de cette thèse est la mise au point d'une paramétrisation 2D des contours d'occlusion. Cette paramétrisation est basée sur une représentation en coordonnées barycentriques du contour 2D. Deux paramètres permettent d'en ajuster la précision. Ce sont deux propriétés particulières qui font la puissance de cette paramétrisation. La première de ces propriétés indique que tout changement d'échelle du contour 2D se traduit par la multiplication du vecteur de paramétrisation par un nombre scalaire. La seconde propriété montre qu'une rotation 2D d'un contour correspond à un déphasage du spectre de la paramétrisation. En utilisant ces deux propriétés, il devient possible de comparer des contours sans tenir compte du facteur d'échelle ou de la rotation 2D qui peuvent exister entre eux. Une mesure de comparaison basée

sur cette paramétrisation a été introduite et permet d'évaluer quantitativement la ressemblance de deux contours.

Les applications de cette approche sont d'autant plus nombreuses que l'utilisation de la paramétrisation ne se limite pas aux contours d'occultation. Nous avons d'ailleurs obtenu d'excellentes performances de reconnaissance lorsque nous avons testé la paramétrisation sur un ensemble de contours différents. La sensibilité aux bruits peut introduire certaines erreurs, mais celles-ci restent généralement faibles.

La deuxième contribution est d'avoir réussi à redorer le blason des graphes d'aspect. Cet outil, très efficace en théorie, n'a jamais été utilisé dans des applications de calcul de pose. Les seuls travaux qui s'y rapportent concernent son calcul dans le cas d'objets particuliers comme les solides de révolution. L'une des principales raisons de cet état de disgrâce provient de la dépendance du graphe d'aspect au facteur d'échelle.

Nous avons réussi à éliminer ce défaut en combinant la paramétrisation 2D, introduite dans le premier chapitre, aux différentes vues du graphe d'aspect. Nous avons utilisé le graphe d'aspect afin de retrouver des paramètres de pose 3D à partir d'information 2D. Un algorithme peu coûteux en terme de temps de calcul a ainsi été mis en place pour déterminer les trois paramètres de rotation et le facteur d'échelle qui définissent la position d'un objet 3D centré sur l'axe perpendiculaire au plan image passant par le centre de projection. Les résultats obtenus à partir de plusieurs tests montrent que notre algorithme est robuste et performant.

Enfin, notre troisième contribution consiste à avoir prouvé qu'il est possible d'effectuer le suivi d'un objet, même lorsque son mouvement est saccadé. En effet, grâce aux résultats obtenus avec le graphe d'aspect, il est possible de prédire quatre des six paramètres de pose. Il suffit alors de trouver un moyen efficace de calculer les deux derniers paramètres de translation afin de pouvoir effectuer un suivi avec prédiction de pose à partir d'une séquence d'images 2D. Le calcul de ces deux paramètres est possible grâce à l'algorithme de Huttenlocher. Cet algorithme permet en effet de retrouver la position du contour d'occultation dans l'image 2D. En tenant

compte des effets de la projection de perspective, il est possible de calculer les six paramètres de pose.

La précision de l'estimation est limitée par les différents facteurs de discrétisation qui permettent de générer le graphe d'aspect et la modélisation 2D de chacune de ses vues. Les résultats que nous avons obtenus avec des séquences d'images de synthèse montrent que l'estimation que nous obtenons est proche de la solution réelle malgré des mouvements importants entre deux images. La précision pourrait cependant être encore améliorée.

L'algorithme que nous proposons possède un autre avantage: sa complexité algorithmique. En effet, notre méthode reste très simple et les temps de calcul obtenus lors des tests montrent qu'un calcul en temps réel est possible.

Travaux futurs

Les méthodes d'estimation de la pose classique nécessitent tous une estimation précise de la pose initiale afin de converger vers la solution idéale. Plus la pose initiale est proche de la pose réelle, plus la convergence est rapide. Ces méthodes seraient donc incapables d'offrir de bonnes performances avec les séquences que nous avons utilisées. En effet, même avec un filtrage de Kalman, il est impossible d'obtenir une pose initiale proche de la pose réelle dans le cas de mouvements saccadés. Il est cependant facile d'imaginer un scénario dans lequel la pose initiale serait calculée par notre méthode puis raffinée par un algorithme plus conventionnel. Il s'agit là de la première avenue que nous suggérons pour des travaux futurs.

La seconde avenue de recherche nécessiterait un peu plus de travail. Il faudrait en effet s'attaquer à une limitation importante de notre approche. Tous les tests ont en effet été effectués sur des images de synthèse et pour extrapoler les résultats dans le cas d'images réelles, nous avons supposé que l'extraction des contours d'occlusion était possible. La mise au point d'une telle méthode semble possible grâce aux développements actuels en vision. Nous avons d'ailleurs

proposé quelques axes de recherche pour régler ce problème dans la dernière section du chapitre 3.

Mettre au point une méthode efficace d'extraction des contours d'occultation permettrait non seulement de s'assurer que notre approche est efficace avec des images réelles, mais elle ouvrirait aussi la porte à de nouvelles améliorations. L'algorithme de Huttenlocher pourrait, par exemple, être remplacé par une recherche exhaustive de tous les contours d'occultation de l'image 2D.

L'étape d'initialisation des algorithmes de poursuite de cible est toujours un problème délicat. Nous avons supposé ici que cette étape était réalisée par un utilisateur externe. Il s'agit là d'une pratique courante dans le domaine de la poursuite de cible. Notre approche présente toutefois un avantage de taille puisque s'il est possible de remplacer l'algorithme de Huttenlocher tel que nous venons de le décrire, l'étape d'initialisation pourrait alors se faire de la même manière que la poursuite de cible elle-même.

Une dernière avenue qu'il semble intéressant d'explorer concerne la simplification du graphe d'aspect qu'on obtiendrait en regroupant les vues semblables. Cette approche permettrait en effet de tenir compte des particularités de chaque objet et de la précision limitée qu'il est possible d'obtenir pour les deux paramètres de rotation calculés à partir du graphe d'aspect. Une approche préliminaire est proposée en annexe.

La méthode de poursuite de cible que nous avons mise au point au fil des années présente un fort potentiel et nous a permis de développer de nombreux outils. Il reste maintenant à explorer les avenues de recherche que nous avons proposées afin d'en faire une méthode utilisable dans des systèmes réels de vision.

Réduction du graphe d'aspect

A1 Introduction

Dans cette annexe, nous allons présenter les premiers résultats d'une approche permettant de réduire le nombre de vues dans un graphe d'aspect construit à partir d'une tessellation régulière de la sphère de point de vue. Cette approche dépasse le cadre des recherches de cette thèse et c'est pour cette raison que nous la présentons en annexe. Elle pourra servir de base à des travaux futurs.

Nous considérerons que le graphe d'aspect est constitué de l'ensemble de contours d'occultation obtenu à partir de la tessellation régulière de la sphère de point de vue, tel que nous l'avons décrit dans le chapitre 2. La réduction du nombre de vues du graphe d'aspect se fait par une stratégie de regroupement basée sur la mesure de comparaison introduite dans le chapitre 1.

L'approche que nous proposons se fait en deux étapes. La première consiste à projeter l'ensemble des vues du graphe d'aspect dans un espace vectoriel de dimension deux. La seconde étape permet alors de regrouper les vues à l'aide d'un algorithme de «Fuzzy K-Mean».

Nous terminerons cette annexe en commentant les résultats obtenus et en montrant les opportunités qu'offre cette approche.

A2 Représentation vectorielle du graphe d'aspect

Afin d'obtenir une représentation schématique de la comparaison entre les différentes vues d'un graphe d'aspect, nous avons utilisé l'algorithme de visualisation de Klock et Buhman [40, 48]. Il permet en effet de représenter les différentes vues du graphe d'aspect par des vecteurs de faible dimension. Nous utiliserons des vecteurs de dimension deux.

Cette opération de projection dans un espace vectoriel se fait à partir de la mesure de comparaison entre deux contours d'occultation introduite dans le premier chapitre. Celle-ci permet de créer une matrice carrée dont la dimension est égale au nombre total de vues dans le graphe. Chaque composante de la matrice correspond ainsi à la mesure de comparaison entre deux vues. Cette matrice possède une diagonale remplie de zéros et est symétrique car la mesure de comparaison l'est. Elle représente la matrice de dissimilarité dans la nomenclature de Klock et Buhman.

Les figures A1 et A2 montrent les résultats obtenus avec la théière et le poisson. Dans les deux cas, on peut constater que la mesure de comparaison définie dans le chapitre 1 est fiable puisque des contours visuellement semblables sont répartis dans les mêmes régions du plan.

A3 Regroupement de différentes vues

Après avoir obtenu une représentation permettant de visualiser la comparaison entre les différentes vues du graphe d'aspect, il faut définir la stratégie permettant d'effectuer le regroupement. Nous avons utilisé l'algorithme classique de «Fuzzy K-Mean». Cet algorithme itératif de classification est très bien décrit dans [8, 32].

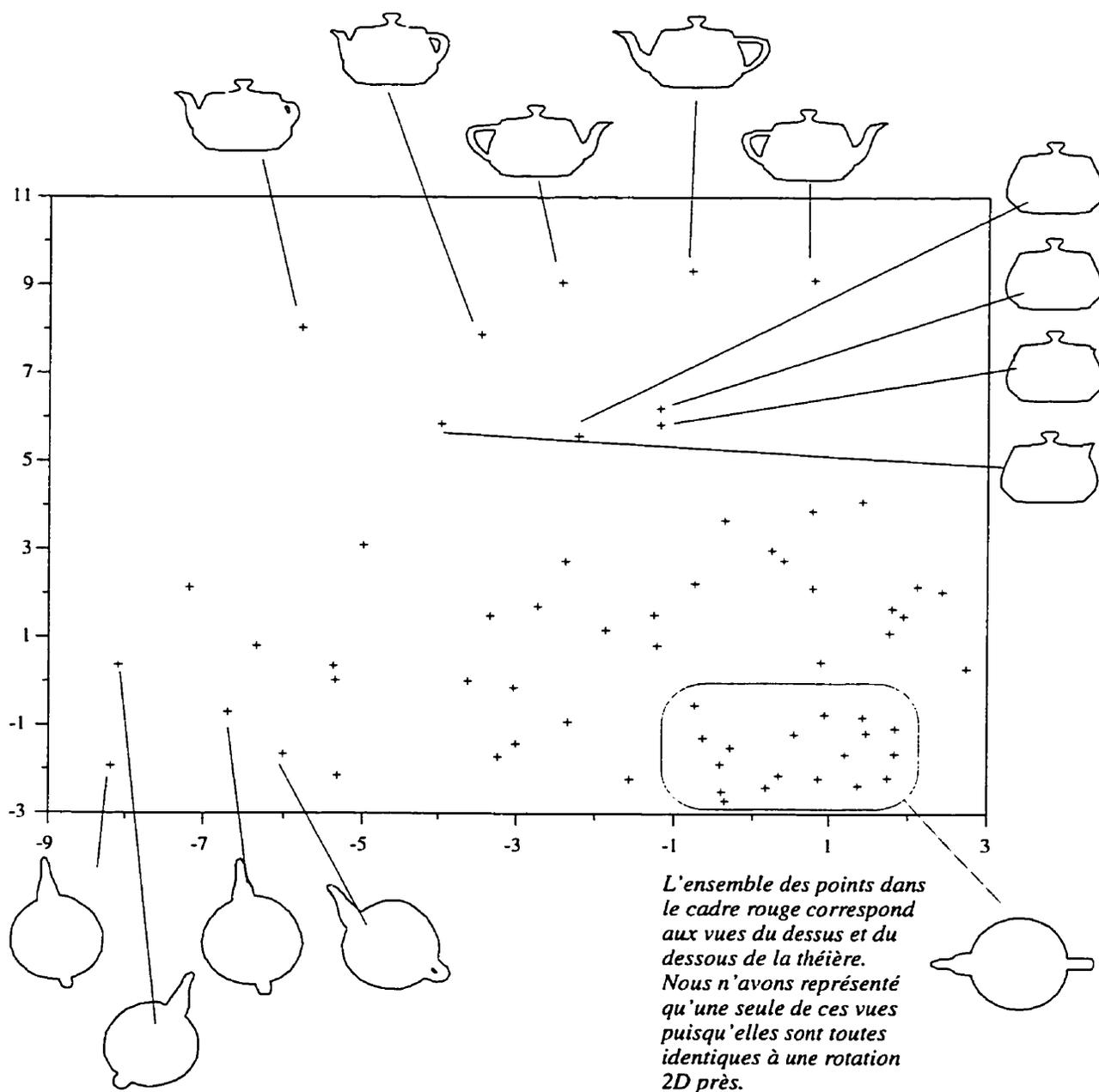


FIGURE A1 Projection des vues du graphe d'aspect de la théière dans le plan.

Dans cette figure, l'ensemble des vues du graphe d'aspect de la théière ont été projetées dans un espace de dimension 2 à l'aide de l'algorithme de Klock et de Buhman.

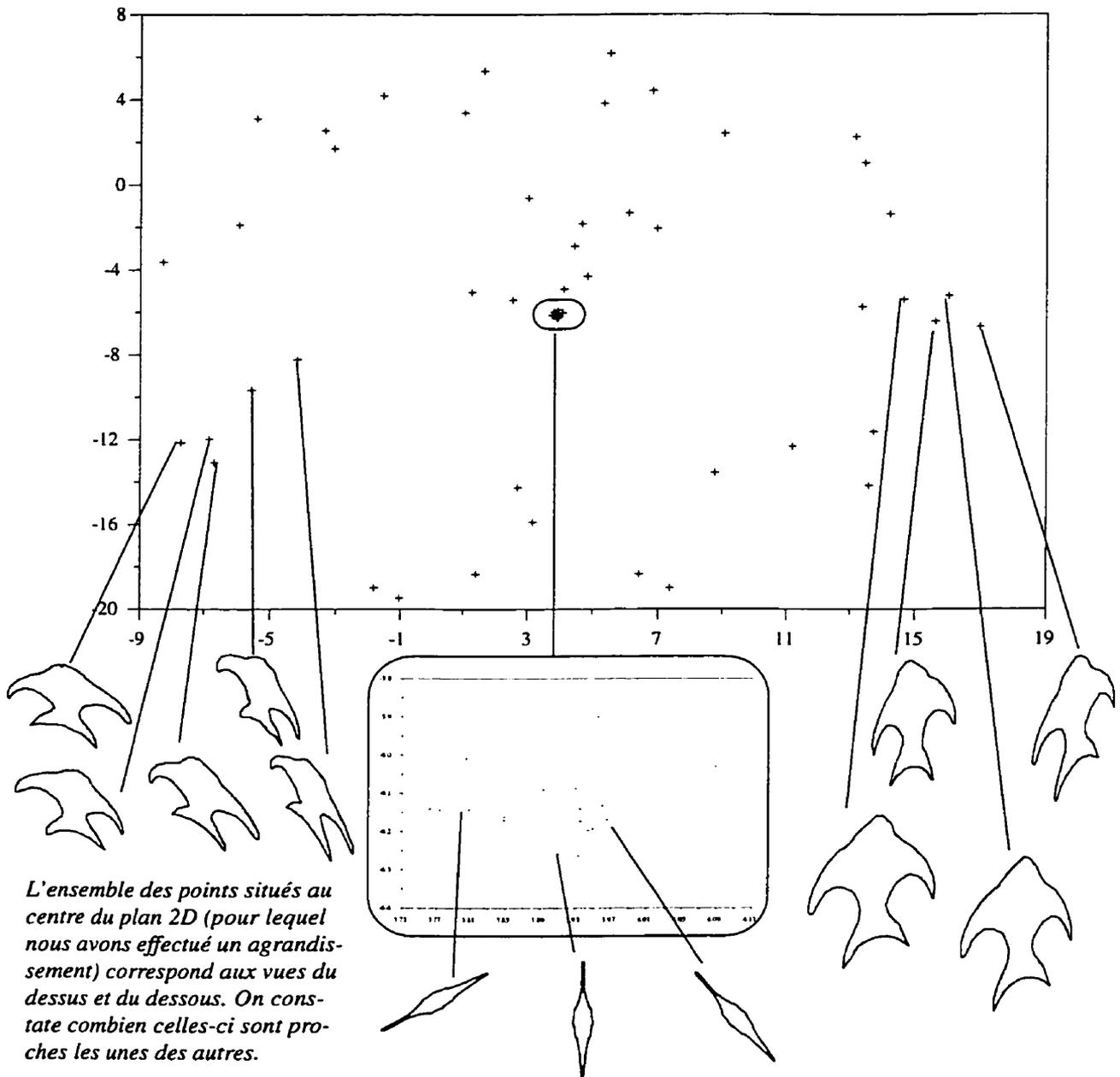


FIGURE A2 Projection des vues du graphe d'aspect du poisson dans le plan.

Dans cette figure, les vues du graphe d'aspect du poisson ont été projetées dans un espace de dimension 2 à l'aide de l'algorithme de Klock et de Buhman.

Les figures A3 et A4 illustrent les résultats de l'algorithme de «Fuzzy K-Mean» avec 6 aspects pour les modèles de la théière, du poisson et de l'avion. On peut constater que la classification donne des résultats assez naturels visuellement. L'approche adoptée prouve donc que le concept est bon.

À partir des ensembles obtenus par l'algorithme de «Fuzzy K-Mean», on peut finalement procéder à la construction du graphe d'aspect simplifié. Pour cela, il faut tenir compte non seulement des ensembles de regroupement, mais aussi du voisinage dans la sphère de point de vue. La figure A5 représente ainsi le graphe que l'on peut obtenir pour le modèle de l'avion en créant des vues à partir d'un modèle à quatre voisins.

A4 Conclusion

Nous venons de voir qu'il est possible de générer un graphe d'aspect simplifié en regroupant les vues les plus ressemblantes grâce à notre approche.

La combinaison de l'algorithme de Klock et Buhman et du principe du «Fuzzy K-Mean» n'est pas nécessairement la meilleure, mais elle nous a permis de produire un résultat intéressant pour un minimum d'investissement. Des algorithmes basés sur une approche du même type que celui de Klock et Buhman adaptés aux regroupements [49] seraient à ce titre sûrement plus justifiés.

La mise au point d'un graphe d'aspect simplifié présente un intérêt certain pour le calcul de la pose. En effet, puisque dans certaines régions du graphe d'aspect les contours d'occultation sont très semblables, il est difficile de donner la pose avec une grande précision. L'utilisation d'un graphe d'aspect simplifié permettrait donc de mieux tenir compte de données intrinsèquement liées à l'objet et pour lesquelles il n'est pas possible d'obtenir une plus grande précision.

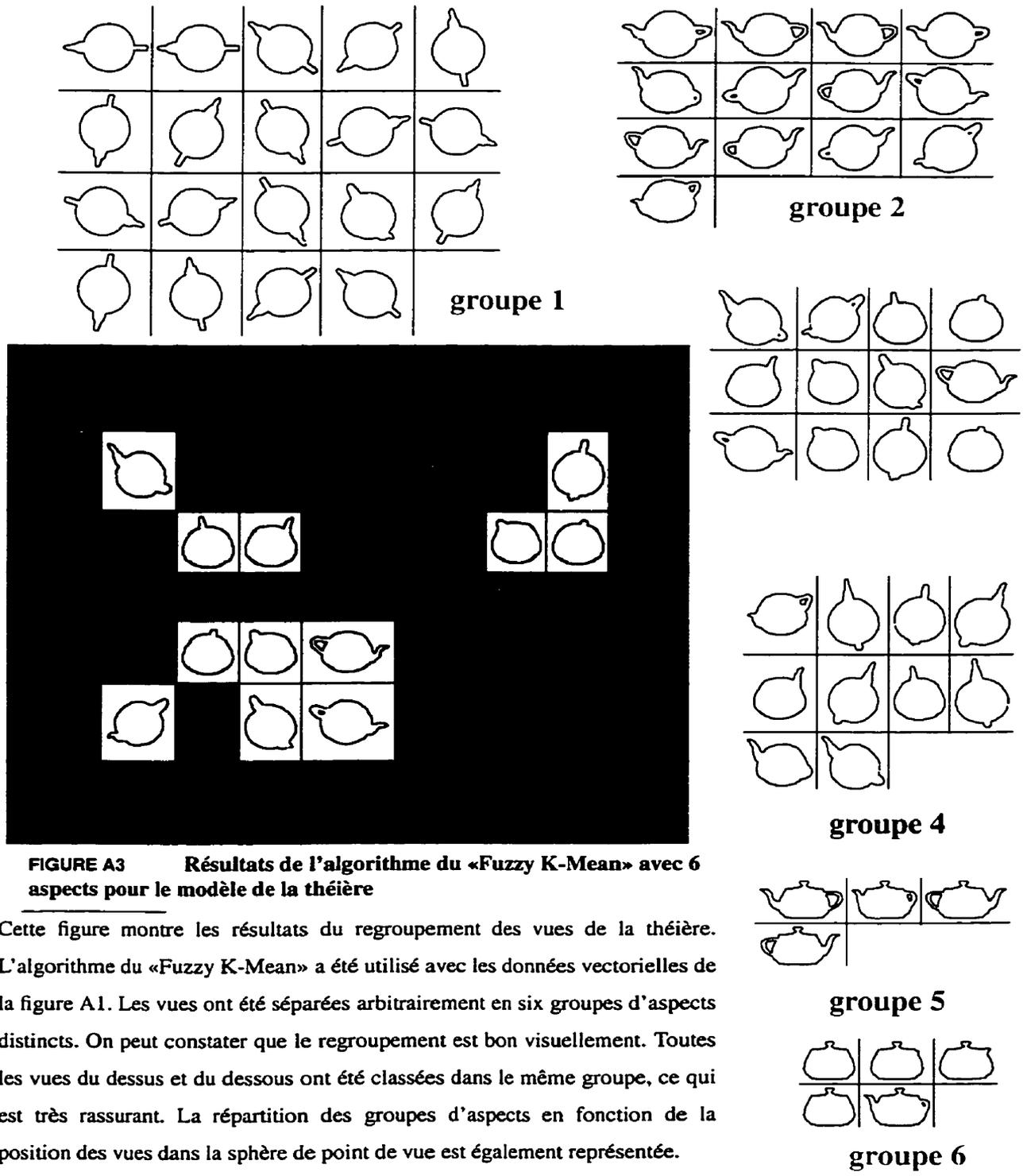


FIGURE A3 Résultats de l'algorithme du «Fuzzy K-Mean» avec 6 aspects pour le modèle de la théière

Cette figure montre les résultats du regroupement des vues de la théière. L'algorithme du «Fuzzy K-Mean» a été utilisé avec les données vectorielles de la figure A1. Les vues ont été séparées arbitrairement en six groupes d'aspects distincts. On peut constater que le regroupement est bon visuellement. Toutes les vues du dessus et du dessous ont été classées dans le même groupe, ce qui est très rassurant. La répartition des groupes d'aspects en fonction de la position des vues dans la sphère de point de vue est également représentée.

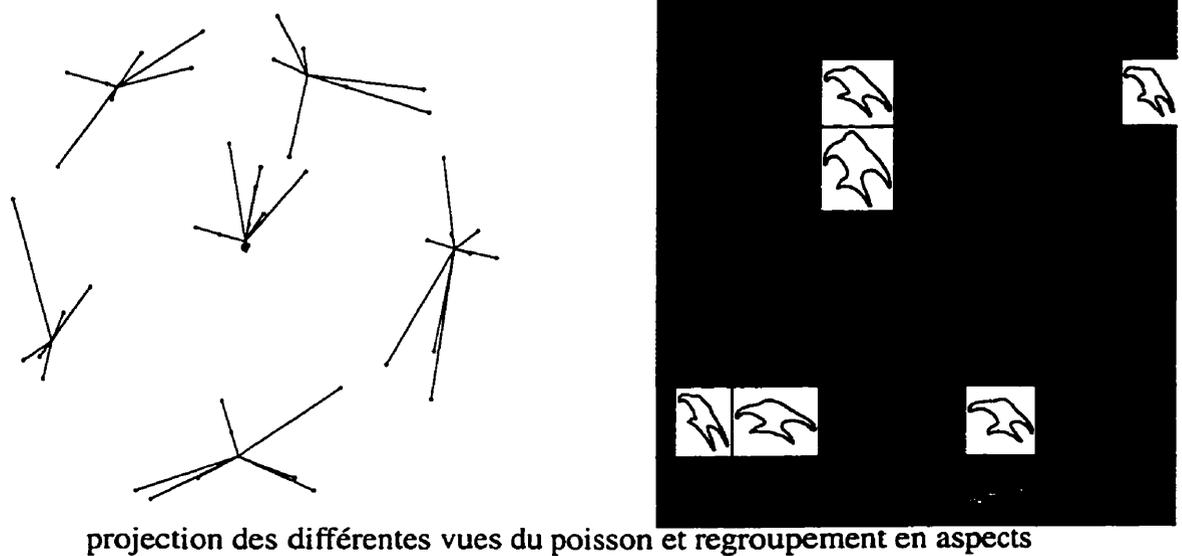
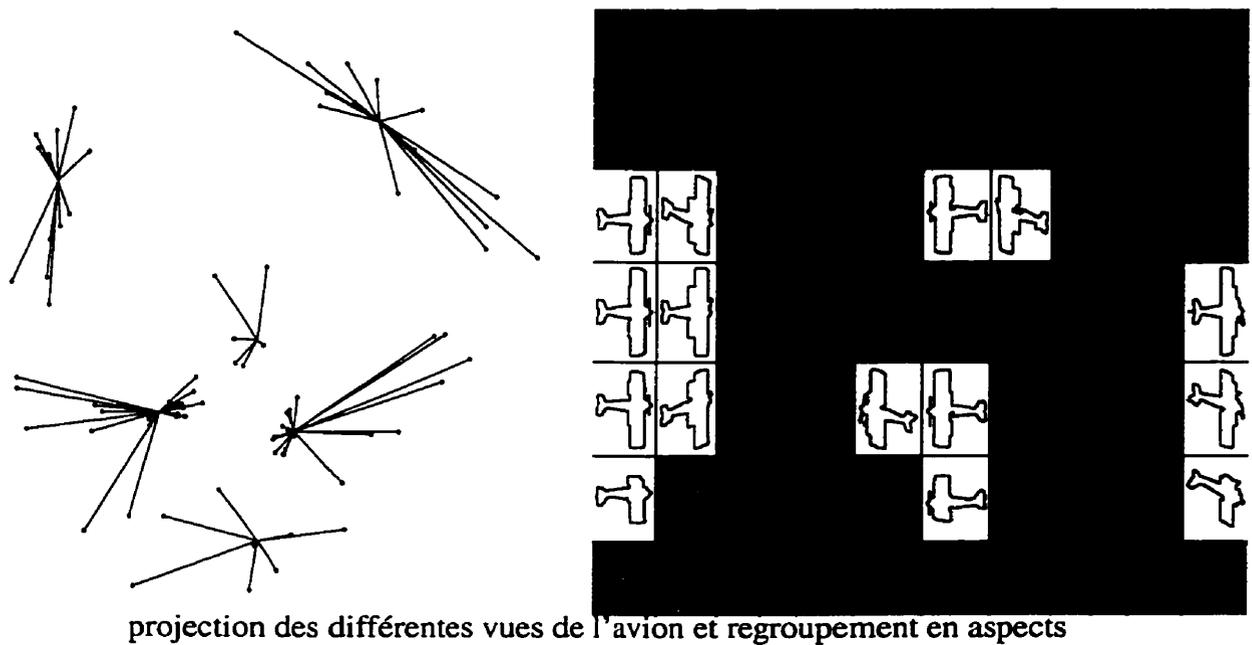


FIGURE A4 Résultats de l'algorithme de «Fuzzy K-Mean» pour le poisson et pour l'avion.

Cette figure représente les projections des vues des deux modèles dans l'espace 2D avec les regroupements correspondant à l'algorithme du «Fuzzy K-Mean».

Les graphes d'aspect de chacun des deux modèles sont également représentés avec des couleurs de fonds différentes pour chacun des groupes d'aspect.

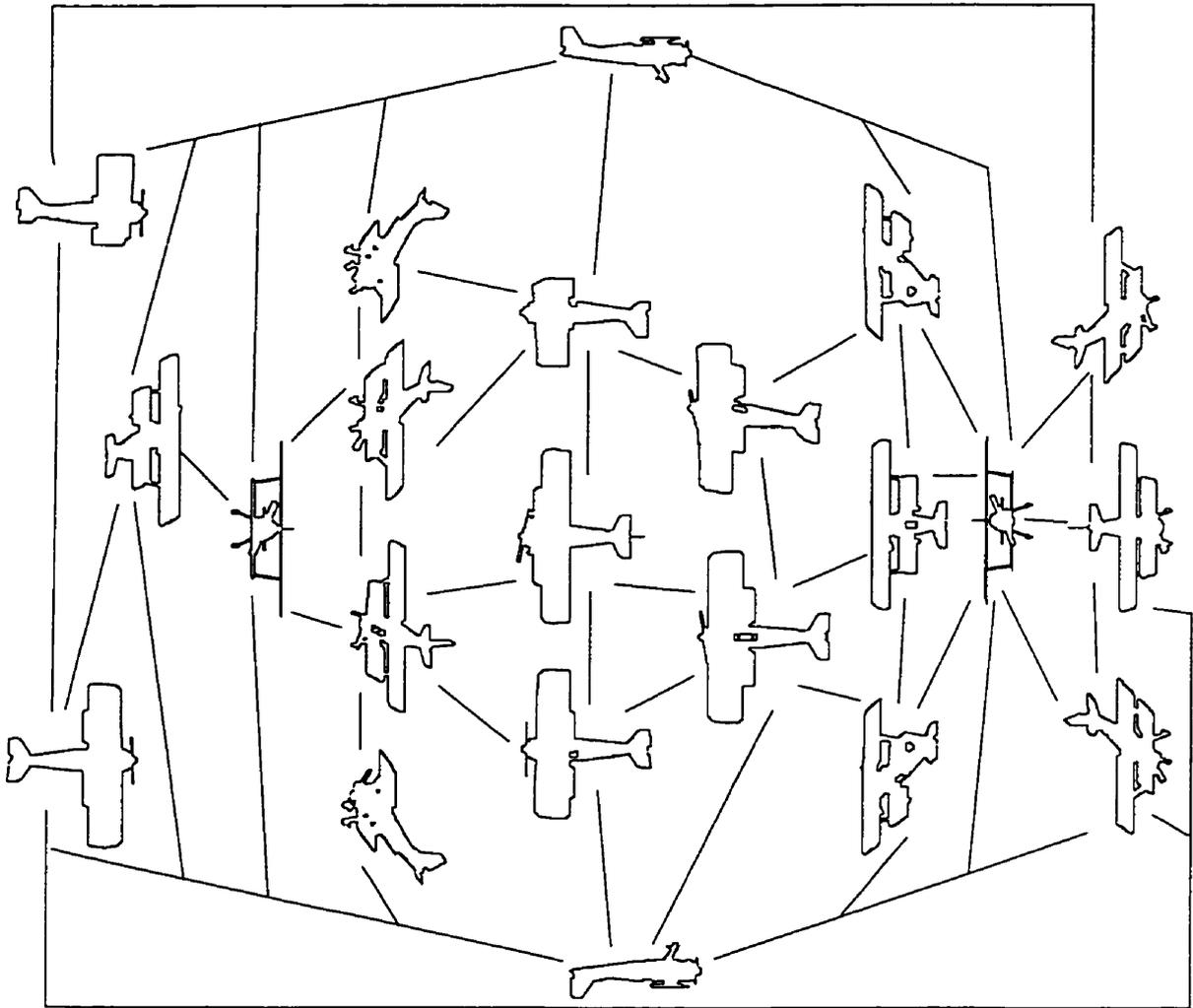


FIGURE A5 Exemple de graphe d'aspect réduit.

Ce graphe d'aspect de l'avion a été obtenu à partir du regroupement en aspects effectué sur un graphe régulier (voir figure A4). Chaque segment de droite qui relie deux aspects correspond à une transition possible dans la sphère de point de vue. Les transitions ont été générées manuellement à partir d'un modèle à quatre voisins appliqué au quadrillage de la sphère de point de vue.

Bibliographie

A - C

- [1] Andersson, M., "*Tracking Methods in Computer Vision*". Ph. D., Dept. of Numerical Analysis and Computing Science, Royal Institute of Technology, Stockholm, Sweden. May 1994.
- [2] Baker K. D. and G. D. Sullivan, "Performance Assessment of Model-Based Tracking." Proc. *1st Workshop on Applications of Computer Vision*, Palm Springs, CA. 1992. pp. 28-35.
- [3] Basu S., I. Essa and A. Pentland, "Motion Regularization for Model-Based Head Tracking," Proc. *13th International Conference on Pattern Recognition*, Vienna, Austria, 1996, pp. 611-616.
- [4] Baumberg A. M. and D. C. Hogg, "An Efficient Method for Contour Tracking Using Active Shape Models," University of Leeds, School of computer studies, research report series. Report 94.11, April 1994.
- [5] Berger M.-O. and Y. Laprie, "Tracking Articulators in X-ray Images with Minimal User Interaction: Example of the Tongue Extraction". Proc. *Third International Conference on Image Processing*, Lausanne, Switzerland, September 16-19, 1996, Vol. 2. pp. 289-292.
- [6] Berger M.-O.. "A Snake Based Tracking Tool for 3D Reconstruction: Snake Strategies for Tracking the Right Contour," Proc. *9th Scandinavian Conference on Image Analysis*. Uppsala, Sweden, 1995, pp. 913-920.
- [7] Berger, M.-O, "How to Track Efficiently Piecewise Curved Contours with a View to Reconstructing 3D Objects," Proc. *12th International Conference on Pattern Recognition*, Jerusalem, Israel, October 9-13, 1994. A, pp. 32-36.
- [8] Bezdek J. C., "*Fuzzy Mathematics in Pattern Classification*," Ph. D. dissertation. Cornell University, Ithaca, NY. 1973.

- [9] Black M. J., and Y. Yacoob, "Tracking and Recognizing Rigid and Non-Rigid Facial Motions Using Local Parametric Model of Image Motion," Proc. *International Conference on Computer Vision*, IEEE Computer Society, Cambridge, MA, 1995, pp. 374-381.
- [10] Blake A., R. Curwen, and A. Zisserman, "A Framework for Spatio-Temporal Control in the Tracking of Visual Contours," In *Real-Time Computer Vision*, C.M. Brown and D. Terzopoulos (eds.), Publications of the Newton Institute of Mathematical Sciences, Cambridge University Press, chap. I, 1994, pp. 3-34.
- [11] Blake A., M. Isard, and D. Reynard, "Learning to Track Curves in Motion," *IEEE Int. Conf. Decision Theory and Control*, 1994, pp. 3788-3793.
- [12] Blake A., M. Isard, and D. Reynard, "Learning to Track the Visual Motion of Contours", *Artificial Intelligence*, 1995, 78, pp. 101-134.
- [13] Byrne N. J., A. M. Baumberg, and D. C. Hogg, "Using Shape and Intensity to Track Non-Rigid Objects," University of Leeds, School of computer studies, Research Report Series, report 94.11, May 1994.
- [14] Canny J. F., "A computational approach to edge detection," *IEEE Trans. Pat. Anal. and Mach. Intel.*, Vol 8, No. 6, 1986, pp. 34-43.
- [15] Chan M., D. Metaxas and S. Dickinson, "A New Approach to Tracking 3D Objects in 2D Image Sequences," Proc. *12th National Conference on Artificial Intelligence*, Seattle, WA, August 1994, pp. 960-965.
- [16] Chang I. C. and C. L. Huang, "Aspect Graph Generation for Non-Convex Polyedra from Perspective Projection View," *Pattern Recognition*, Vol. 25, 1992, pp. 1075-1096.
- [17] Chen S. and H. Freeman, "Characteristic View Modeling of Curved-Surface Solids," *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 10, 1996, pp. 537-560.
- [18] Chen C. and A. Kak, "A Robot Vision System for Recognizing 3D Objects in Low Order Polynomial Time," *IEEE Trans. on Syst. Man, and Cyber.*, Vol. 19, 1989, pp. 1535-1563.
- [19] Cootes T. F. and C. J. Taylor, "Active Shape Models - Smart Snakes," Proc. *British Machine Vision Conference*, Springer-Verlag, 1992, pp.266-275.
- [20] Cootes T. F., C. J. Taylor, D. H. Cooper and J. Graham, "Training Models of Shape from Sets of Examples," Proc. *British Machine Vision Conference*, Springer-Verlag, 1992, pp.9-18.
- [21] Curwen R. and A. Blake, "Dynamic Contours: Real-time Active Splines," *Active Vision*, A. Blake and A. Yuille (eds.), MIT Press, Cambridge, MA, Ch. 1, 1992, pp. 39-58.

D - E

- [22] Daucher N., M. Dhome, J. T. Lapresté and G. Rives, "Modelled Object Pose Estimation and Tracking by Monocular Vision," Proc. *British Machine Vision Conference*, Springer-

Verlag, 1993, pp. 249-258.

[23] Delagnes P., J. Benois and D. Barba, "Active Contours Approach to Object Tracking in Image Sequences With Complex Background," *Pattern Recognition Letters*, Vol. 16, No. 2, 1995, pp. 171-178.

[24] Deutsch C., "Suivi et Localisation d'une Cible Complexe Connue par Vision Monoculaire," Rapport pré-doctoral, RT-LVSN-97-04, Laboratoire de Vision et Systèmes Numériques, Université Laval, Québec, Canada, 1997.

[25] Eggert D. W. and K. W. Bowyer, "Computing the Perspective Projection Aspect Graph of Solids of Revolution," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 2, 1993, pp. 109-128.

[26] Eggert D. W. and K. W. Bowyer, "Computing the Generalized Aspect Graph for Objects with Moving Parts," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 6, 1993, pp. 605-610.

[27] Eggert D. W., K. W. Bowyer, C. R. Dyer, H. I. Christensen and D. B. Goldgof. "The Scale Space Aspect Graph," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 11, 1993, pp. 1114-1130.

[28] Eggert D. W., L. Stark and K. W. Bowyer, "Aspect Graphs and their Use in Object Recognition," *Annals of Mathematics and Artificial Intelligence*, Vol. 13, 1995, pp. 347-375.

[29] Elder J. H. and S. W. Zucker, "Computing Contour Closure," *Proc. Fourth European Conference on Computer Vision*, Cambridge, England, 1996, Vol. I, pp. 399-412.

F - K

[30] Faugeras O., J. Mundy, N. Ahuja, C. Dyer, A. Pentland, R. Jain, K. Ikeuchi and K. Bowyer, "Why Aspect Graphs Are Not (Yet) Practical for Computer Vision," *Computer Vision Graphics and Image Processing: Image Understanding*, Vol. 55, No. 2, March 1992, pp. 212-218.

[31] Freeman H., "Computer Processing of Line Drawing Images," *ACM Computer Survey*, Vol. 6, 1974, pp. 57-97.

[32] Gath I. and A. B. Geva, "Unsupervised Optimal Fuzzy Clustering," *IEEE Trans. on Pattern Anal. and Machine Intell.*, Vol. 11, No. 7, July 1989, pp.773-781.

[33] Gavrilă D. M. and L. S. Davis, "3-D Model-Based Tracking of Humans in Action: a Multi-View Approach," *IEEE Symposium on Computer Vision*, Miami Beach, 1995, pp. 253-258.

[34] Gavrilă D. M. and L. S. Davis, "Towards 3-D Model-Based Tracking and Recognition of Human Movement: a Multi-View Approach," *Proc. Int. Workshop on Face and Gesture Recognition*, Zurich, 1995, pp. 264-279.

-
- [35] Gennery D. B., "Visual Tracking of Known Three-Dimensional Objects," *International Journal of Computer Vision*, Vol. 7, No. 3, April 1992, pp. 243-270.
- [36] Gennery D. B., "Tracking Known Three-Dimensional Objects," Proc. *2nd National Conference on Artificial Intelligence*, Pittsburgh, PA, August, 1982, pp. 13-17.
- [37] Gigus Z., J. Canny and R. Seidel, "Efficiently Computing and Representing Aspect Graphs of Polyhedral Objects," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 13, 1991, pp. 542-551.
- [38] Harris Chris, "Tracking with Rigid Models," In *Active Vision*, A. Blake and A. Yuille (eds.), MIT Press, Cambridge, MA, Ch. 1, 1992, pp. 59-74.
- [39] Hébert M. and T. Kanade, "The 3D-Profile Method for Object Recognition," Proc. *IEEE Conf. on Computer Vision and Pattern Recognition*, 1985, pp. 458-463.
- [40] Hofmann T. and J. M. Buhmann, "Multidimensional Scaling and Data Clustering". *Advances in Neural Information Processing Systems*, Vol. 7, Morgan Kaufmann Publishers, 1995.
- [41] Huttenlocher D. P. and W. J. Rucklidge, "A Multi-Resolution Technique for Comparing Images Using the Hausdorff Distance," Technical Report TR92-1321, Computer Science Department, Cornell University, 1992.
- [42] Huttenlocher D. P., J. J. Noh and W. J. Rucklidge, "Tracking Non-Rigid Objects in Complex Scenes," Technical Report TR92-1320, Computer Science Department, Cornell University, 1992.
- [43] Huttenlocher D. P., M. E. Leventon and W. J. Rucklidge, "Visually-Guided Navigation by Comparing Two-Dimensional Edge Images," Technical Report TR94-1407, Computer Science Department, Cornell University, 1994.
- [44] Huttenlocher D. P., G. A. Klanderman and W. J. Rucklidge, "Comparing images using the Hausdorff distance," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 15, 1993, pp. 850-863.
- [45] Ikeuchi K., "Generating an Interpretation Tree from a CAD Model for 3-D Object Recognition in Bin-Picking Tasks," *International Journal of Computer Vision*, Vol. 1, 1987, pp. 145-165.
- [46] Kass M., A. P. Witkin and D. Terzopoulos, "Snakes: Active Contour Model". *International Journal of Computer Vision*, Vol. 1, No. 4, 1988, pp. 321-331.
- [47] Kaucic R., B. Dalton and A. Blake, "Real-Time Lip Tracking for Audio-Visual Speech Recognition Applications," Proc. *European Conference on Computer Vision*, vol. 2, pp. 376-387, Cambridge UK, 1996.
- [48] Klock H. and J. M. Buhmann, "Multidimensional Scaling by Deterministic Annealing." *International Workshop on Energy Minimization Methods in Computer Vision and Pattern*
-

Recognition, Venice, 1997, pp. 245-260.

[49] Klock H. and J. M. Buhmann, "Data Visualization by Multidimensional Scaling: A Deterministic Annealing Approach," *Pattern Recognition*, 1999, Vol. 33, No. 4, pp. 651-669.

[50] Koenderink J.J. and A. J. van Doorn, "The internal representation of solid shape with respect to vision," *Biological Cybernetics*, 1979, pp. 211-216.

[51] Koller D., K. Daniilidis and H. H. Nagel, "Model-Based Object Tracking in Monocular Image Sequences of Road Traffic Scenes," *International Journal of Computer Vision*, Vol. 10, No. 3, June 1993, pp. 257-281.

[52] Koller D., J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao and S. Russell. "Towards Robust Automatic Traffic Scene Analysis in Real-Time," Proc. *12th International Conference on Pattern Recognition*, 12th, Jerusalem, Vol. I, 1994, pp. 126-131.

[53] Koller D., J. Weber, and J. Malik, "Towards realtime visual based tracking in cluttered traffic scenes," Proc. *Intelligent Vehicles Symposium*, Paris, France, 1994, pp. 201-206.

[54] Koller, D., J.W. Weber, J. Malik, "Robust Multiple Car Tracking with Occlusion Reasoning," Proc. *Third European Conference on Computer Vision*, Stockholm, Sweden, Vol. A, 1994, pp.189-196.

[55] Korn M. R. and C. R. Dyer, "3D Multiview Object Representations for Model-Based Object Recognition," *Pattern Recognition*, Vol. 20, 1987, pp. 91-103.

[56] Kriegman D. J. and J. Ponce, "Computing Exact Aspect Graphs of Curved Objects: Solids of Revolution," *International Journal of Computer Vision*, Vol. 5, No. 2, 1990, pp. 119-135.

[57] Kuhl F. P. and C. R. Giardina, "Elliptic Fourier Features of a Closed Contour," *Computer Graphics and Image Processing*, Vol. 18, 1982, pp.236-258.

L - Q

[58] Laurentini A., "Comments on: Efficiently Computing and Representing Aspect Graphs of Polyhedral Objects," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 1, January 1996, pp. 57-58.

[59] Lerasle F., G. Rives, M. Dhome and A. Yassine. "Human Body Tracking by Monocular Vision," Proc. *Fourth European Conference on Computer Vision*, Vol. II, 1996, pp. 518-527.

[60] Leymarie F. and M. D. Levine, "Tracking Deformable Objects in the Plane Using an Active Contour Model," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 6, June 1993, pp. 617-634.

[61] Lowe, D. G., "Fitting Parameterized Three-Dimensional Models to Images," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 13, No. 5, May 1991, pp. 441-450.

[62] Lowe D. G., "Three-dimensional object recognition from single two-dimensional

images," *Artificial Intelligence*, Vol. 31, No. 3, 1987, pp. 355-395.

[63] Lowe, D. G., "Robust Model-Based Motion Tracking Through the Integration of Search and Estimation," *International Journal of Computer Vision*, Vol. 8, No. 2, August 1992, pp. 113-122.

[64] Malik J., "Interpreting line drawings of curved objects," *International Journal of Computer Vision*, Vol. 1, 1987, pp. 73-103.

[65] Maybank J., A.D. Worrall and G.D. Sullivan, "Filter for Car Tracking Based on Acceleration and Steering Angle," Proc. *Fourth European Conference on Computer Vision*, Vol. II, 1996, pp. 540-549.

[66] Pentland A.P., "Parts: Structured Descriptions of Shape," Proc. *National Conference on Artificial Intelligence*, 1986, pp. 695-701.

[67] Petitjean S., J. Ponce and D.J. Kriegman, "Computing Exact Aspect Graphs of Curved Objects: Algebraic Surfaces," *International Journal of Computer Vision*, Vol. 9, No. 3, 1992, pp. 231-255.

R - Z

[68] Ravela S., B. Draper, J. Lim and R. Weiss, "Adaptative Tracking and Model Registration Across Distinct Aspects." Proc. *IEEE International Conference on Intelligent Robots and Systems*, Vol.1, Pittsburgh, August 5-9, 1995, pp. 174-180.

[69] Rehg J. and T. Kanade, "Model-Based Tracking of Self-Occluding Articulated Objects." Proc. *5th International Conference on Computer Vision*, Cambridge, MA, USA. June 20-22, 1995, pp.612-617.

[70] Rehg J., "*Visual Analysis of High DOF Articulated Objects with Application to Hand Tracking*," Ph. D. Thesis, Carnegie Mellon University, April 1995, #CMU-CS-95-138.

[71] Roberts L. G., "*Machine Perception of Three-Dimensional Objects*," MIT Press, 1966.

[72] Sripradisvarakul T. and R. Jain, "Generating Aspect Graphs for Curved Objects." Proc. *IEEE Workshop on Interpretation of 3D Scenes*, 1989, pp. 109-115.

[73] Stewman J. H., L. Stark and K. E. Bowyer, "Creating the Perspective Projection Aspect Graph of Convex Polyhedra," Proc. *2nd Int. Conf. on Computer Vision*, 1998, pp. 494-500.

[74] Stewman J. H., "*Viewer Centered Representations for Polyhedral Objects : Computing the Exact Perspective Projection Aspect Graph of an Object Bounded by Planar Faces*," PhD Thesis, University of south Florida, Departement of Computer Science and Engineering, December 1991.

[75] Sullivan G. D., A. D. Worrall and J. M. Ferryman, "Visual Object Recognition Using Deformable Models of Vehicles," Proc. *Workshop on Context-Based Vision*, Cambridge Massachusetts, 1995, pp. 75-86.

-
- [76] Sullivan G.D., "Model-based Vision for Traffic Scenes Using the Ground-Plane Constraint," In *Real-Time Computer Vision*, C. M. Brown and D. Terzopoulos (eds.), chap II, Publications of the Newton Institute of Mathematical Sciences, Cambridge University Press, 1994, pp. 93-113.
- [77] Terzopoulos D. and D. Metaxas, "Tracking Nonrigid 3D Objects," In *Active Vision*, A. Blake and A. Yuille (eds.), MIT Press, Cambridge, MA, 1992, Ch. 1, pp. 75-90.
- [78] Terzopoulos D. and R. Szeliski, "Tracking with Kalman Snakes," In *Active Vision*, A. Blake and A. Yuille (eds.), MIT Press, Cambridge, MA, 1992, Ch. 1, pp. 3-20.
- [79] Worrall A. D., J. M. Ferryman, G. D. Sullivan and K. D. Baker, "Pose and Structure Recovery using Active Models," Proc. *6th British Machine Vision Conference*, 11-14th September, Birmingham, 1995, pp. 137-146.
- [80] Worrall A. D., G. D. Sullivan and K. D. Baker. "Pose Refinement of Active Models Using Forces in 3D," Proc. *Third European Conference on Computer Vision*, Stockholm, Sweden, Vol. I, 1994, pp. 341-350.
- [81] Yassine A. "*De la localisation et du suivi par vision monoculaire d'objets polyédriques articulés modélisés*," Thèse de doctorat, Université Blaise Pascal, Clermond-Ferrand, France. Novembre 1995.