N° d'ordre : 2137

THÈSE

présentée devant

L'UNIVERSITÉ DE RENNES I

Pour obtenir

le grade de : **Docteur de l'Université de Rennes I**

Mention : **Traitement du Signal et Télécommunications**

par

Christophe COLLEWET

Équipe d'accueil : ***Cemagref* Rennes**

École doctorale : **Informatique, Traitement du Signal et Télécommunications**

Composante universitaire : **Structures et Propriétés de la Matière**

Titre de la thèse :

Contributions à l'élargissement du champ applicatif des asservissements visuels 2D

Soutenue le 4 Février 1999, devant la commission d'examen composée de :

M.	**Jean-Jacques**	**FUCHS**	Président
MM.	**Michel**	**DEVY**	Rapporteurs
	Michel	**DHOME**	
MM.	**François**	**CHAUMETTE**	Examinateurs
	Bernard	**ESPIAU**	
	Gilles	**RABATEL**	

Remerciements

Je tiens à exprimer tout d'abord ma plus vive gratitude et mes remerciements les plus sincères à :

- Monsieur Jean-Jacques Fuchs, professeur à l'université de Rennes I, qui m'a fait l'honneur de présider ce jury ;
- Messieurs Michel Devy, ingénieur de recherche au LAAS et Michel Dhome, directeur de recherche au LASMEA pour avoir accepté la lourde tâche de rapporteur ;
- Messieurs Bernard Espiau, directeur de recherche à l'INRIA Rhônes-Alpes et Gilles Rabatel, directeur de recherche au *Cemagref* de Montpellier pour l'intérêt qu'ils ont bien voulu porter à mes travaux ;
- Monsieur François Chaumette, chargé de recherche à l'INRIA de Rennes, pour m'avoir encadré et encouragé dans les périodes de doute. Je lui suis reconnaissant pour ses conseils et ses remarques. Je n'oublie pas non plus sa tenacité, elle me fut une aide précieuse.

Je remercie également les membres de mon unité de recherche, plus particulièrement Laurence, Michel et Stéphane pour m'avoir aidé lors de la réalisation de notre site expérimental, Guylaine pour les discussions que nous avons eues.

Mes remerciements vont aussi à ma famille qui m'a toujours soutenue durant toutes ces années d'études et sans qui ce travail ne serait pas.

Enfin, je remercie mes enfants, Laurick, Maël et Chloé d'avoir accepté mes soirées et mes week-ends loin d'eux, Guylaine pour m'avoir aidé, supporté, reconforté et à qui j'ai laissé la lourde tâche d'organiser la vie famiale pendant cette fin de thèse, je lui dois beaucoup.

Table des matières

Introduction **9**

1 Cadre de l'étude **13**

2 État de l'art en asservissement visuel **17**

2.1 Les asservissements visuels 3D 18

2.1.1 Les techniques photogrammétriques 18

2.1.2 Les techniques basées sur la triangulation 19

2.1.3 Les techniques basées sur l'obtention du mouvement et de la structure à partir du mouvement du capteur 21

2.1.4 Les techniques basées sur l'apprentissage 22

2.2 Les asservissements visuels 2D 23

2.2.1 Généralités 23

2.2.2 Le choix des informations visuelles 24

2.2.3 Le calcul de la matrice d'interaction 26

2.2.4 Les types d'architecture matérielle 28

2.2.5 Les lois de commande 28

2.2.6 L'élargissement du champ applicatif 29

3 Asservissement visuel par rapport à un objet de forme complexe **33**

3.1 L'approche fonction de tâche 33

3.2 Méthode générale de calcul de la matrice d'interaction 35

3.3 Méthode de calcul d'une approximation de la matrice d'interaction 37

3.4 Modélisation d'une courbe plane 38

3.4.1 Représentation sous forme implicite 38

3.4.2 Représentation sous forme polaire 39

3.4.3 Représentation sous forme paramétrique 41

3.4.3.1 Utilisation d'une décomposition sur une base de fonctions 41
3.4.3.2 Utilisation d'une décomposition en fonctions rationnelles 43
3.5 Asservissement visuel basé sur la signature polaire 45
3.5.1 Calcul de la matrice d'interaction 45
3.5.2 Extraction des informations visuelles 50
3.5.3 Résultats de simulations 51
3.5.3.1 Simulations non bruitées 52
3.5.3.2 Simulations bruitées 56
3.5.3.3 Influence de la précision de l'estimation de la profondeur une fois la tâche réalisée sur le comportement de la loi de commande 60
3.5.3.4 Influence du nombre d'harmoniques sur le comportement de la loi de commande 64
3.5.4 Résultats expérimentaux 67
3.5.4.1 Description de la plate-forme expérimentale . . . 67
3.5.4.2 Vérification des résultats 68
3.5.4.3 Résultats expérimentaux 70
3.5.5 Conclusion 86
3.6 Asservissement visuel basé sur une représentation paramétrique . 88
3.6.1 Calcul de la matrice d'interaction 88
3.7 L'exemple des descripteurs de Fourier 92
3.7.1 Calcul de la matrice d'interaction 92
3.7.2 Extraction des informations visuelles 96
3.7.3 Convention concernant l'origine de la variable u 97
3.7.4 Résultats de simulations 98
3.7.4.1 Simulations non bruitées 98
3.7.4.2 Simulations bruitées 103
3.7.4.3 Influence du nombre d'harmoniques sur le comportement de la loi de commande 103
3.7.4.4 Influence de la précision de l'estimation de la profondeur une fois la tâche réalisée sur le comportement de la loi de commande 106
3.7.5 Résultats expérimentaux 110
3.7.5.1 Cas d'un objet plan 110

3.7.5.2 Cas d'un objet plan dont le contour est décrit par une fonction non holomorphe 118
3.7.5.3 Cas d'un objet non plan 122
3.7.6 Conclusion 122
3.8 Conclusion 122

4 Positionnement par rapport à un objet plan de forme inconnue 125
4.1 Le formalisme de redondance de la fonction de tâche 126
4.2 Étude d'un objet simple : le segment 127
4.2.1 Spécification de la tâche 127
4.2.2 Centrage de l'objet dans l'image 129
4.2.2.1 Utilisation de quatre informations visuelles 129
4.2.2.2 Utilisation de trois informations visuelles 130
4.2.3 Détermination d'une fonction de tâche secondaire 131
4.2.3.1 Résultats de simulations non bruitées 134
4.2.3.2 Résultats de simulations bruitées 136
4.2.4 Scrutation d'informations visuelles 139
4.2.4.1 Modélisation de la taille de l'image à l'aide d'informations dynamiques 139
4.2.4.2 Modélisation de la taille de l'image à l'aide d'informations statiques 140
4.2.4.3 Application à la commande 144
4.2.4.4 Simulations non bruitées 146
4.2.4.5 Simulations bruitées 146
4.2.4.6 Discussion sur l'erreur de positionnement 148
4.3 Étude d'un objet plan de forme quelconque 150
4.3.1 Étude d'un objet présentant trois points caractéristiques . 151
4.3.1.1 Spécification de la tâche principale 151
4.3.1.2 Modélisation de la surface de l'objet dans l'image 152
4.3.2 Résultats expérimentaux 155
4.3.2.1 Vérification des résultats 155
4.3.2.2 Résultats expérimentaux 155
4.3.2.3 Comportement de la loi de commande en environnement non calibré 165
4.3.3 Quelques méthodes d'extraction de points caractéristiques 166

4.3.3.1 Application à un objet plan de forme complexe . 168
4.3.4 Étude d'un objet plan de forme quelconque 174
4.3.4.1 Choix des informations visuelles et obtention d'une loi de commande 175
4.3.4.2 Résultats expérimentaux 176
4.4 Conclusion 184

Conclusion et perspectives **185**

A Modélisation de la longueur d'un segment dans l'image à l'aide d'informations visuelles dynamiques **191**
A.1 Obtention de la relation de récurrence 191
A.2 Modélisation de la taille d'un segment de l'image 193

B Conservation de certaines propriétés propres à un objet plan par transformation perspective **195**
B.1 Définitions 195
B.2 Étude des points stationnaires 197
B.3 Étude des points d'inflexion 199

Bibliographie **201**

Introduction

Les « machines », ou plus précisément les systèmes automatisés, font maintenant partie intégrante de notre vie. Toutefois, mais est-il besoin de le rappeler, il n'en a pas toujours été ainsi, loin de là ! En effet, ce n'est que vers la fin du XIXe siècle qu'apparurent les premières machines « évoluées », au Royaume-Uni et tout d'abord dans le secteur agricole, pour assurer à l'homme une nouriture suffisante en le préservant des disettes grâce à une forte augmentation du rendement des récoltes. Puis, vint ensuite la révolution manufacturière ou industrielle où, d'une part la maîtrise de l'énergie et d'autre part le développement des technologies permit l'essor des économies occidentales. L'augmentation de la productivité fut alors de plus en plus un souci majeur et de nouvelles méthodes d'organisation du travail virent le jour, le travail à la chaîne étant un exemple des plus illustrateurs.

Cette façon d'appréhender l'outil de production conduisit à une vision rationnelle des tâches à effectuer, plus parcellaires, et donc propices à une automatisation de plus en plus poussée jusqu'à l'apparition des premiers robots aux USA dans les années 1960. Ils étaient alors cantonnés à des rôles d'automates mécaniques programmables où les trajectoires effectuées étaient apprises puis répétées quelque soit leur environnement.

L'évolution des capteurs extéroceptifs, et en particulier des capteurs visuels, a révélé, avec l'émergence de l'électronique et de son corollaire l'informatique, les potentialités d'utilisation des robots. Tout d'abord, leur degré de flexibilité a été accru. Ils ne sont plus dédiés à une unique tâche bien définie, mais à plusieurs tâches ou à des tâches évoluées où l'environnement proche du robot est susceptible de se modifier.

C'est ainsi que l'on peut entrevoir, maintenant, le robot comme une aide à la réalisation de tâches complexes particulièrement pénibles ou astreignantes (robotique agricole, travaux en salles réfrigérées en usine agro-alimentaire) voire dangereuses ou inhospitalières (travaux sous-marins, en centrale nucléaire ou relevant de la conquête spatiale) ou en ambiance protégée où tout risque de contami-

nation par contact doit être écarté (industries électronique et pharmaceutique, conditionnement de produits alimentaires). Signalons par ailleurs, qu'en ce qui concerne ce dernier point, le capteur visuel devient alors un capteur de choix.

Cette évolution dans la complexité des tâches à réaliser doit s'accompagner aussi, à notre avis, d'une évolution des lois de commande par vision. En particulier, nous pensons qu'elles devraient jouir d'une plus grande indépendance vis-à-vis des objets à traiter garantissant, par la même occasion, un élargissement du champ applicatif. C'est dans ce but que les travaux rapportés dans le présent document ont été orientés. Ce dernier a été organisé en quatre chapitres de la manière suivante :

- Un premier chapitre situe succinctement le cadre dans lequel les travaux ont été réalisés. Dans ce contexte, quelques perspectives d'applications sont évoquées.

- Dans le second chapitre, consacré à l'état de l'art en *asservissement visuel*, nous décrivons tout d'abord les principales structures de commande s'en réclamant. Nous nous focalisons ensuite sur l'approche *asservissement visuel 2D* ; nous détaillons notamment certains points qui nous ont paru les plus importants d'aborder comme le choix des *informations visuelles*, le calcul de la *matrice d'interaction*, les types d'architecture matérielle, les lois de commande, ou encore l'élargissement du champ applicatif.

- Le chapitre 3, quant à lui, traite le cas où l'objet observé a une forme particulièrement complexe. Par forme complexe, on entend ici une forme pour laquelle on ne dispose pas de connaissances suffisamment précises pouvant conduire à l'élaboration d'un modèle. Il s'en suit, dans la même acception du terme, une image souhaitée complexe. Ce chapitre peut être vu, en fait, comme un prolongement des travaux menés à l'INRIA par F. Chaumette [Chaumette 90]. En effet, après avoir rappelé brièvement la méthode de calcul de la matrice d'interaction qu'il propose, nous montrons une façon d'obtenir une approximation de cette matrice, indépendamment de l'objet observé, et ce pour des primitives images de type contour. Deux méthodes sont proposées, une première basée sur la description des courbes en coordonnées polaires, et une seconde basée sur une description paramétrique. Elles sont toutes deux validées grâce à des simulations et à des expérimentations.

- D'un tout autre point de vue, le chapitre 4 traite le cas de figure où *la consigne visuelle* est inconnue — c'est-à-dire le cas où l'image finale souhaitée n'est pas connue précisément — et ce dans le contexte d'une tâche particulière de positionnement consistant à amener le capteur en face de

l'objet considéré. Pour simplifier le problème, nous traitons tout d'abord le cas d'un objet filiforme et montrons comment cette approche se généralise à un objet plan de forme quelconque dès que trois *points caractéristiques* existent. Cette notion, pour l'instant intuitive, sera approfondie. Une approche, ne faisant cette fois aucune hypothèse, sera proposée en fin de chapitre. Des résultats de simulations et d'expérimentations valideront, tout au long du chapitre, les lois de commande proposées.

Une conclusion et deux annexes terminent le document. Nous résumons, dans cette conclusion, les principaux résultats obtenus et évoquons quelques perspectives de recherches.

Chapitre 1

Cadre de l'étude

Les travaux décrits ci-après ont été réalisés au sein de l'unité de recherche « technologie des équipements agro-alimentaires » du département Génie des Équipements Agricoles et des Procédés Agro-alimentaires (GEAPA) du *Cemagref*[1]. Les recherches qui sont menées dans ce département sont centrées sur trois axes thématiques correspondant à des secteurs d'activité majeurs :

- Technologies pour le développement et la maîtrise d'une agriculture propre et raisonnée ;
- Méthodes et équipements pour le diagnostic, la maîtrise et le contrôle de la qualité alimentaire ;
- Évaluation et ingénierie des équipements.

Les techniques d'asservissement visuel, comme nous le verrons ci-après, permettent de répondre efficacement à certains grands problèmes du département GEAPA.

Le premier type d'applications identifiées concerne la caractérisation rapide de la qualité des produits alimentaires soit sur des lignes d'approvisionnement en tête de procédé, soit sur des lignes de transformation. En effet, la sensibilité du consommateur au problème général de la qualité de ces produits devient un élément prépondérant pour l'industrie agro-alimentaire. Cette qualité peut se décliner en qualité bactérioloqique ou sanitaire, qualité nutrionnelle et gustative.

[1]Pour une description détaillée des activités du *Cemagref* le lecteur intéressé peut consulter le site http ://www.cemagref.fr.

Pour contrôler ces différents aspects de la qualité il est souvent fait appel à des analyses physico-chimiques ou rhéologiques mais il s'agit de méthodes destructives ou difficilement utilisables directement en ligne. Par ailleurs, il reste souvent un ensemble de contrôles visuels. Toutefois, ces contrôles se heurtent à la subjectivité de l'appréciation humaine et sont limités par l'opérateur qui ne peut réaliser, en ligne, que des mesures très simples. De ce fait, elles n'expriment que partiellement la qualité du produit.

L'essor des techniques de vision numérique doit permettre, comme dans les industries manufacturières, une amélioration des contrôles visuels en prenant en compte des nouveaux critères et en assurant des résultats plus objectifs et plus répétables tout en respectant les cadences de production. La difficulté, principalement due à la nature des produits observés, rend son application au domaine de l'agro-alimentaire encore très marginale et du domaine de la recherche.

Les travaux décrits dans le présent document, couplés à ceux menés dans notre unité de recherche, par exemple dans la filière volaille [Marty-Mahé 97b, Marty-Mahé 97a, Raffy 99] ou dans la filière charcuterie salaisonnerie [Legeard 99], doivent permettre à terme d'inspecter, d'identifier puis de classifier les produits observés par rapport à leur morphologie ou à leur conformation. Dans le futur, les techniques de *vision dynamique active* permettront de détecter la présence de défauts sur l'ensemble du produit observé (plaie, déchirure, hématome, etc.) voire de permettre, si besoin est, l'examen de fibres musculaires ou de tissus adipeux.

D'autres opérations plus complexes peuvent être envisagées comme l'automatisation du positionnement de capteurs pour certaines mesures (taux de viande maigre, ph, etc.), la découpe sur chaînes d'abattage ou sur le produit fini au tranchage et à l'assemblage, et ce dans diverses filières.

Un outil indirect d'assurance de la qualité d'un produit alimentaire, généralement une pièce de viande, est sa traçabilité. La crise récente liée à l'encéphalite spongiforme bovine est un exemple des plus parlants. D'autres filières, en particulier la filière porcine, ont conduit des études sur les moyens de garantir une identification fiable des animaux, des carcasses et des différentes pièces de viandes. La solution retenue est le tatouage à l'encre alimentaire des porcelets juste après la naissance. Cette marque, qui subit une déformation pendant sa croissance, doit être lue à l'abattoir sur l'animal adulte. Dans ce contexte, les techniques d'asservissement visuel et en premier lieu celles qui seront développées dans le chapitre 4, sont particulièrement bien adaptées à la réalisation d'un système d'identification automatique, l'emplacement du tatouage étant *a priori* inconnu, et la lecture du code par vision déjà possible [Chanet 98]. À cet effet, une tête de lecture à

positionnement manuel est actuellement en cours de développement industriel.

L'autre grand domaine d'activité du département GEAPA concerne les équipements agricoles. Diverses applications peuvent être à ce titre envisagées comme par exemple le guidage d'engins agricoles [Debain 96, Billingsley 97, Pla 97] et l'entretien des espaces verts [Debain 96].

Par ailleurs, le souci actuel d'une production respectant l'environnement conduit à l'émergence d'une agriculture propre et raisonnée. Il s'en suit un besoin d'équipements nouveaux permettant des traitements localisés, herbicides ou phytosanitaires, directement sur la plante [Sanchiz 96, Giles 97, Rabatel 97, Tillett 98].

D'autres secteurs de l'agriculture nécessitent des manipulations précises comme la récolte des fruits et légumes [Kondo 91, Sarig 93, Kondo 96] ou de végétaux fragiles (petits fruits, champignons) [Tillett 93] mais aussi leurs inspections en vue d'un tri [Meyers 90] et leurs empaquettages [Tillett 95]. Le taillage intelligent des végétaux est aussi possible [McFarlane 97].

L'horticulture s'avère être aussi une source d'applications potentielles, par exemple pour la réalisation automatique d'opérations de taille des plantes [Kinase 91] voire d'opérations plus complexes comme les transplantations ou les bouturages [Kinase 91, Simonton 91].

Enfin, signalons, pour conclure ce court chapitre, que dans tous les exemples rapportés ci-dessus l'emploi des techniques d'asservissement visuel est particulièrement intéressant dans la mesure où un couplage étroit est nécessaire entre les phases d'action et de perception pour la bonne réalisation de la tâche demandée.

Le chapitre suivant est dédié aux principaux travaux concernant ce domaine de recherche.

Chapitre 2

État de l'art en asservissement visuel

Dans ce second chapitre, nous décrivons tout d'abord les principales structures de commande permettant la réalisation de tâches robotiques à l'aide d'un capteur de vision. Nous nous focalisons ensuite sur les différents axes de recherche menés en asservissement visuel 2D en portant, pour terminer, une attention toute particulière à ceux contribuant à l'élargissement du domaine d'application de cette approche.

Outre la grande richesse d'informations fournies par un capteur de vision, le simple fait de l'inclure dans une boucle de commande permet d'augmenter considérablement la précision d'un robot. En effet, les premières interactions vision et commande fonctionnaient selon le principe de la boucle ouverte : le robot se déplaçait après une reconstruction de la scène puis, le cas échéant, se déplaçait une nouvelle fois suite à une modification de cette scène. D'une part, la précision obtenue dépendait directement des capteurs, des asservissements articulaires ou encore de la justesse des modèles utilisés et des phases de calibration associées et, d'autre part, ce type de commande, très simple, ne pouvait se révéler efficace que dans le cas d'une scène statique. De tels travaux sont décrits par exemple dans [Rosen 76], [Tani 77].

Depuis, la littérature relate un nombre considérable de travaux permettant la réalisation de tâches aussi complexes que de suivre un objet mobile, de le saisir éventuellement, ou de modifier sa trajectoire en cas d'obstacle ou d'occlusion de

l'objet et cela, de plus en plus, à l'aide d'un ensemble robot/caméra partiellement calibré. Nous allons les décrire ci-après, tout au moins, les principales approches.

La première utilisation de la vision en boucle fermée est due à Shirai et Inoue [Shirai 73] qui décrivirent comment un capteur de vision pouvait augmenter la précision du positionnement. On parlait alors de retour par vision (*visual feedback*) mais, c'est à Hill et Park [Hill 79] que l'on doit l'apparition du terme asservissement visuel (*visual servoing*). Plusieurs approches ont ensuite vu le jour, elles ont été caractérisées dans [Sanderson 80] en deux grandes classes suivant l'utilisation faite des informations visuelles : une première approche, nommée asservissement en situation ou asservissement visuel 3D (*position-based control*), et une seconde, nommée asservissement basé image ou asservissement visuel 2D (*image-based control*). Depuis peu sont venues s'ajouter deux autres approches issues des travaux menés à l'INRIA [Malis 97, Malis 98] et [Crétual 97, Crétual 98] : respectivement les asservissements visuels 2D 1/2 et $d2D/dt$.

Ces différents types de commande et les approches qui s'en réclament font l'objet des paragraphes suivants.

2.1 Les asservissements visuels 3D

Dans ce type de commande, les informations visuelles sont extraites de l'image afin d'obtenir la position et l'orientation de la caméra par rapport à l'objet. Dans le cas idéal où le capteur peut être modélisé par une simple transformation perspective (voir figure 2.1), on s'aperçoit immédiatement que la connaissance de la projection d'un point de l'objet observé n'est pas suffisante pour remonter aux coordonnées 3D du point considéré. Il s'en suit différentes façons d'aborder le problème dit de *reconstruction tridimensionnelle*. Elles sont détaillées ci-après.

2.1.1 Les techniques photogrammétriques

L'hypothèse de base de ces techniques repose sur la connaissance du modèle de l'objet mais aussi sur celui de la caméra. On peut alors montrer que, théoriquement, trois points suffisent pour résoudre le problème[1]. En effet, un point de l'objet fournit par transformation perspective deux équations, et le changement de repère de la caméra à l'objet est entièrement défini par six paramètres (trois translations et

[1]Ce problème est connu dans la littérature sous le non du problème PnP pour *perspective-n-point-problem* selon la dénomination introduite par Fischler et Bolles [Fischler 81].

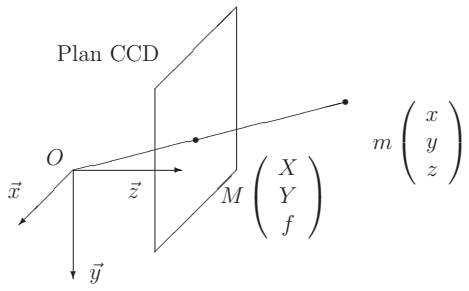


FIG. 2.1 – Transformation perspective d'un point

trois rotations). Cependant, la résolution de ces équations, non linéaires, conduit à quatre configurations caméra/objet possibles. Précisons toutefois que, d'une part, une solution unique est obtenue dès la connaissance de quatre points et que, d'autre part, des points supplémentaires, non coplanaires, améliorent les résultats [Yuan 89].

Le système d'équations étant non linéaire, des méthodes de résolution itérative ont été proposées. Par exemple, dans [Yuan 89] et [Lowe 91] les auteurs utilisent une méthode de Newton-Raphson avec le principal inconvénient de devoir fournir un estimé initial correct. Ce problème est résolu dans [Wunsch 97b] grâce à l'utilisation d'un réseau de neurones qui apprend à associer une image à la situation caméra/objet. Dans [Dementhon 95], une méthode basée sur la transformation orthographique (cette transformation considère que tous les points observés sont à la même distance du plan focal) fournit une estimation de la situation caméra/objet. Quelques itérations suffisent ensuite pour obtenir un résultat précis. Notons que cette méthode, suffisamment rapide, a été appliquée dans [Daucher 97] pour la réalisation d'une tâche de positionnement. D'autres approches, basées sur l'utilisation d'un filtre de Kalman, étendu ou non, permettent d'aboutir [Wilson 96, Wunsch 97a]. Par ailleurs, compte tenu de contraintes inhérentes à la tâche à réaliser, une selection automatique des informations visuelles peut être réalisée au cours du mouvement [Janabi-Sharifi 97].

2.1.2 Les techniques basées sur la triangulation

Deux techniques principales sont employées : la stéréovision et l'utilisation d'un éclairage structuré, le plus souvent un plan laser. L'intérêt de telles approches

réside alors dans le fait qu'un modèle de l'objet n'est plus nécessaire. Détaillons plus avant ces deux méthodes :

- La stéréovision appréhende le problème de la reconstruction 3D grâce à l'utilisation de plusieurs caméras (deux le plus souvent). On obtient alors facilement les coordonnées du point observé par triangulation. Par contre, *la mise en correspondance* de la projection du point considéré sur les deux plans images devient le point clé de cette approche.

  Pour résoudre ce problème, différentes hypothèses peuvent être formulées. Elles conduisent à l'écriture de contraintes (unicité, continuité, ordre, etc.) qui permettent de lever certaines ambiguïtés. Dans le même but, d'autres primitives autres que des points sont souvent utilisées tels que les segments ou les contours. Signalons aussi l'intérêt d'augmenter le nombre de caméras [Ayache 87].

  Nous ne détaillerons pas plus avant les techniques propres à la reconstruction 3D à l'aide de plusieurs caméras mais invitons le lecteur intéressé à la lecture de [Faugeras 88] et [Ozanian 95] où un état de l'art complet est présenté.

- L'utilisation combinée d'un éclairage structuré et d'une caméra, généralement équipée d'un filtre accordé sur la longueur d'onde de la source lumineuse, permet d'obtenir les coordonnées 3D du point observé par intersection de la droite de visée avec, suivant les cas, le rayon ou le plan de lumière incident. Il s'en suit, par ailleurs, une image de la scène très simple et de plus quasiment unimodale.

  Cette technique est donc habituellement mise en œuvre dans les cas de scènes complexes [Agin 85]. Toutefois, notons qu'elle possède l'inconvénient majeure de nécessiter un balayage de la scène qui peut pénaliser les performances dynamiques du système, à moins de différencier les phases de régulation ou de poursuite comme dans [Collewet 99].

Par ailleurs, pour les deux méthodes de triangulation décrites ci-dessus, il existe une différence notable de comportement du système bouclé suivant l'architecture matérielle du dispositif. Dans le cas d'un capteur fixe, indépendant du porteur, un changement de repère intervient dans la loi de commande pour exprimer les données calculées dans le repère du capteur dans le repère de l'effecteur. Cette approche est donc très sensible à la calibration du dispositif [Hager 95, Grosso 96] et, en ce sens, le robot ne réalise pas une réelle boucle fermée sur la position 3D de la cible. Ce problème est, par contre, résolu si le capteur devient solidaire du manipulateur du robot (*eye-in-hand system*). Toutefois, à notre connaissance, seuls

des sytèmes basés sur l'utilisation d'un éclairage structuré [Agin 85, Chen 87] sont de ce type. En revanche, dans le cas d'un système stéréoscopique observant à la fois la scène et l'effecteur du robot, par exemple comme dans [Grosso 96], la loi de commande devient relativement insensible aux éventuelles erreurs de calibration. De plus, les auteurs décrivent des résultats expérimentaux montrant que des tâches 3D de positionnement ou de suivi peuvent être réalisées de façon très précise, comme par exemple mettre en place le capuchon d'un stylo.

Dans le cas d'une scène statique — les coordonnées 3D du point observé étant connues — les techniques classiques de commande en position ou en vitesse peuvent être mises en œuvre [Coiffet 86]. Par contre, dans le cas d'un objet en mouvement, une estimation de sa trajectoire doit être réalisée pour calculer celle du robot [Andersson 88, Rizzi 93]. Signalons que, dans ces deux cas, elle est basée sur un modèle de connaissance, en l'occurence le mouvement d'une balle. En revanche, dans [Allen 91], un filtre à coefficients constants $\alpha - \beta - \gamma$ est utilisé. Dans [Allen 93], ces mêmes auteurs préfèrent une méthode locale ce qui permet une bonne modélisation du mouvement et des caractéristiques du bruit. Il s'en suit une loi de commande conduisant à des résultats plus précis que dans leur première approche. Signalons, par ailleurs, qu'ils proposent de mettre en correspondance des images de vitesse par calcul du flot optique dans les deux caméras. Cette approche leur permet de saisir des objets mobiles.

Concernant la poursuite de cibles, signalons de nombreux travaux impliquant des *têtes stéréoscopiques actives* [Nordlund 96, Lim 97, Mitsuda 97, Spindler 97, Marefat 97, Srinivasa 97]. Cependant, la plus grande part des travaux menés dans ce domaine reste centrée sur les aspects poursuite en tant que telle, notamment pour la réalisation de tâches de focalisation d'attention (*gaze control*), plutôt que sur les aspects commande d'un robot dans le but de réaliser une tâche particulière (voir cependant [Dias 98] pour la poursuite d'objet à l'aide d'un robot mobile).

2.1.3 Les techniques basées sur l'obtention du mouvement et de la structure à partir du mouvement du capteur

Le fait de disposer d'une image acquise selon des points de vue différents[2], par exemple par une caméra montée à l'extrémité d'un manipulateur, peut, de la même façon qu'avec deux caméras, fournir des informations 3D concernant l'objet (*depth from motion*). De même, l'analyse du mouvement est aussi un moyen d'obtenir de telles informations (*depth from dynamics*). On retrouve là l'ensemble des travaux menés en vision dynamique.

[2]En général on dispose plutôt d'une séquence d'images.

Toutefois, nous ne développerons pas plus ces approches dans la mesure où, à notre connaissance, elles n'ont pas été appliquées à la réalisation de tâche robotique telle que des tâches de positionnement ou du suivi. Signalons, en revanche, leur utilisation en robotique mobile pour la création de cartes. Cependant, nous invitons le lecteur intéressé par ce domaine à se référer à [Mitiche 96].

2.1.4 Les techniques basées sur l'apprentissage

Les méthodes décrites ci-dessus, mises à part celles basées sur le mouvement, font appel à des primitives géométriques relativement simples, de type points ou segment par exemple [3]. L'intérêt, à nos yeux, d'une méthode d'apprentissage peut alors résider dans la nature des informations visuelles qu'elle est susceptible d'appréhender, particulièrement des informations pouvant caractériser la globalité de l'image. Certains travaux relatent de tels cas de figures :

Dans [Wells 96], les auteurs utilisent les descripteurs de Fourier et les moments centrés. Pour ce faire, une approche neuronale est mise en œuvre. Le principe consiste à disposer de couples $(\underline{r} - \underline{r}^*, \underline{s})$ où, $\underline{r}$ est la situation courante du porteur (vecteur de dimension n où n est le nombre de degrés de liberté contrôlés), $\underline{r}^*$ sa situation désirée et $\underline{s}$ les informations visuelles (de dimension m), et de réaliser un apprentissage supervisé de la relation liant $\underline{r} - \underline{r}^*$ à $\underline{s}$. Des résultats expérimentaux sont présentés sur des objets réels, en l'occurence une vanne et une pièce d'automobile. Les résultats obtenus s'avèrent toutefois imprécis.

Une démarche originale est, par ailleurs, proposée dans [Nayar 96]. Elle consiste à opérer une analyse en composante principale sur l'image désirée. Pour ce faire, il est nécessaire de disposer d'un ensemble de couples $(\underline{q} - \underline{q}^*, \underline{s})$ dans un voisinage de $\underline{s}^*$ (où $\underline{q}$ représente le vecteur des variables articulaires courantes, $\underline{q}^*$ celles désirées et $\underline{s}^*$ le motif désiré). Ce domaine particulier de l'espace, *l'espace propre*, permet le calcul de la base de vecteurs orthogonaux (représentant en fait des images). Toute image I peut alors se décomposer, dans cette base de vecteurs E_i, sous la forme :

$$I = \sum_{i=1}^{i=m} a_i E_i + C \tag{2.1}$$

Les informations visuelles choisies sont alors constituées des composantes a_i de l'image I projetée dans l'espace propre. Puis, dans un second temps, mais toujours hors ligne, les auteurs modélisent la relation liant $\underline{q} - \underline{q}^*$ à $\underline{a}$ (avec $\underline{a} = (a_1, \ldots, a_m)^T$) grâce à une fonction B-spline multidimensionnelle. La réalisation

[3] Exception faite des méthodes de triangulation laser où le problème se pose différemment.

d'une tâche se déroule alors de la manière suivante : (i) décomposition de l'image dans la base des vecteurs E_i pour obtenir les informations visuelles courantes, (ii) calcul du chemin optimum et mouvement du manipulateur. Les auteurs présentent des résultats expérimentaux sur des images aussi complexes qu'un visage humain ou une carte électronique pour le positionnement d'un composant. Par contre, on peut regretter que les résultats ne soient obtenus que pour des situations initiales proches de celle désirée, vraisemblablement en limite de validité de la fonction B-spline interpolante, et dans le cas trois axes uniquement.

2.2 Les asservissements visuels 2D

2.2.1 Généralités

Contrairement aux asservissements en situation, les techniques d'asservissement visuel 2D ne nécessitent pas de reconstruction 3D, les raisonnements sont réalisés directement au niveau du capteur. En effet, à partir d'un vecteur $\underline{s}$ représentant les informations visuelles courantes, on cherche à atteindre un motif $\underline{s}^*$ correspondant à la position finale souhaitée du porteur. Le choix des informations visuelles et l'obtention de la relation les liant aux mouvements du porteur sont deux points fondamentaux de cette approche. Cette relation, obtenue par dérivation de $\underline{s}$ par rapport à $\underline{r}$, est en fait une matrice de dimension m x n (en conservant les notations introduites en 2.1.4) appelée, suivant les auteurs, matrice d'interaction, jacobien de la tâche ou jacobien de l'image. Elle peut être obtenue sous forme analytique dans le cas d'informations visuelles plus ou moins simples (points [Feddema 89a] et [Feddema 89b], droites, cercles, cylindres, sphères, ellipses [Chaumette 90]). Ce sont en fait les primitives géométriques les plus couramment utilisées [Bien 93].

Par ailleurs, même si nous n'avons pu trouver dans la littérature que très peu de résultats rapportant le comportement d'une tâche robotique réalisée par asservissement visuel 2D comparée à la même tâche réalisée par asservissement visuel 3D (voir néanmoins dans [Crowley 95] une comparaison des deux approches pour la commande d'une tête stéréoscopique active), il est maintenant communément admis les points suivants :

- Les asservissements visuels 2D sont optimisés d'un point de vue temps de calcul puisque la phase de reconstruction 3D est évitée. Ce gain de temps n'est pas forcément négligeable puisque le delai d'application de la commande est réduit, ce qui contribue à la stabilité du système [Wunsch 97a].

- Pour une structure où le capteur est embarqué sur le porteur, structure réputée comme étant beaucoup plus performante qu'une structure où le capteur est déporté [Wilson 96], les raisonnements étant réalisés dans le capteur, il est relativement aisé de s'assurer que l'objet restera tout au long du mouvement dans le champ de vision de la caméra. Tout cela suppose, bien entendu, que la loi de commande converge, ce qui peut ne pas être toujours le cas. En effet, quand la situation initiale est très éloignée de celle désirée, un asservissement visuel 2D peut échouer. Rapportons, dans ce cas de figure, l'utilisation avantageuse des asservissements visuels 2D 1/2 qui utilisent, comme leur nom le laisse supposer, à la fois des données 3D et 2D mais, sans toutefois nécessiter la connaissance du modèle de l'objet [Malis 97, Malis 98].

Par contre, il est vrai qu'il est, par essence, difficile de contrôler la tâche à réaliser dans l'espace opérationnel de l'effecteur du robot [Wilson 96]. Cependant, l'utilisation d'informations hybrides permet d'aboutir [Castano 94].

Comme nous venons de le voir, les asservissements visuels basés image possèdent les propriétés essentielles garantissant au manipulateur la bonne réalisation de la tâche à effectuer. C'est ainsi que ce type de commande se rencontre de plus en plus et cela dans divers domaines d'application. Citons par exemple : la conduite d'un véhicule sous-marin [Rives 97], d'un engin agricole [Debain 96, Martinet 97], le suivi du contour d'un objet inconnu [Coste-Manière 95], le contrôle du regard [Spindler 97], la saisie d'objet en mouvement [Smith 97b] ou encore la réalisation de tâche de positionnement pour des manipulations très précises [Hager 97].

Cependant, bien que ces applications aient été menées avec succès, certains points font l'objet d'une littérature abondante. Nous détaillons ci-après ceux qui nous ont paru les plus importants d'aborder.

2.2.2 Le choix des informations visuelles

Un des points, certainement des plus importants, concerne la nature des informations visuelles. Elles sont principalement de deux types :

- Les informations visuelles dynamiques [Crétual 97, Santos-Victor 97, Crétual 98] : elles se révèlent particulièrement intéressantes dans le cas où la scène observée est complexe. En effet, il peut être alors difficile, voire impossible, d'exprimer la tâche à réaliser à partir d'informations visuelles exclusive-

ment géométriques, le cas de figure typique étant celui où l'objet observé est inconnu [4].

- Les informations visuelles de nature géométrique : dans ce cas, c'est la primitive point qui reste encore la plus couramment employée [Bien 93] : par exemple dans [Feddema 89b, Chaumette 90, Coste-Manière 95, Mitsuda 97, Spindler 97, Hager 97] ou comme dans [Khadraoui 96] grâce à l'utilisation d'un laser. Des primitives de type droite peuvent être aussi utilisées [Chaumette 90, Debain 96], voire des primitives de type cylindre, sphère ou ellipse [Chaumette 90].

Cependant, si l'on veut appréhender des objets de nature plus complexe, de telles informations visuelles de type géométrique paraissent insuffisantes comme le soulignent Wells et al. dans [Wells 96]. D'autres types d'informations visuelles sont proposés pouvant caractériser la globalité de l'image, les moments d'inertie, la projection de l'image sur une ligne, la surface et la transformée de Fourier [Bien 93]. Récemment, des travaux concernant des images réelles ont permis la réalisation de tâche de positionnement [Deguchi 96], la méthode employée est identique à celle décrite dans [Nayar 96], déjà exposée en 2.1.4, seules les lois de commande sont différentes.

Par ailleurs, à partir d'un même objet, plusieurs primitives peuvent souvent être extraites. La question posée est alors que faire de ces informations visuelles potentielles ? Deux alternatives sont proposées :

- Dans [Jang 91] les auteurs utilisent des informations visuelles pour réaliser une trajectoire dans l'espace opérationnel. Ils mettent clairement en évidence dans leurs travaux l'intérêt d'employer des informations visuelles redondantes. Signalons que des conclusions similaires sont rapportées dans [Jägersand 97].

- Par contre, une autre alternative est décrite dans [Feddema 89a]. En effet, à partir des informations visuelles disponibles, une selection est opérée pour améliorer le comportement de la loi de commande en termes de sensibilité par rapport au bruit de mesure et du coût du calcul de la matrice d'interaction. Dans [Papanikolopoulos 95], la sélection est réalisée de manière à éviter des singularités locales de la matrice d'interaction et à ne prendre en compte que des informations visuelles extraites avec suffisamment de précision.

[4] On verra cependant au chapitre 4 une façon de faire à l'aide d'informations visuelles de nature géométrique.

2.2.3 Le calcul de la matrice d'interaction

Comme déjà évoqué en 2.2.1, cette matrice est fondamentale pour l'élaboration de lois de commande. Revenons dans un premier temps à sa définition.

En reprenant les notations introduites en 2.1.4, les informations visuelles $\underline{s}$ ne dépendent que de la situation $\underline{r}$ du porteur, elle même dépendante de la valeur $\underline{q}$ des variables articulaires, et du temps t, soit :

$$\underline{s} = \underline{s}(\underline{q}, t) \tag{2.2}$$

En différenciant cette équation par rapport au temps, on obtient la façon dont les informations visuelles varient suivant les mouvements du porteur de la caméra :

$$\dot{\underline{s}} = \frac{\partial \underline{s}}{\partial \underline{r}} \frac{\partial \underline{r}}{\partial \underline{q}} \dot{\underline{q}} + \frac{\partial \underline{s}}{\partial t} \tag{2.3}$$

où l'on reconnait :

- La matrice d'interaction : $\partial \underline{s}/\partial \underline{r}$ qui ne dépend que de la configuration de la caméra par rapport à l'objet et des informations visuelles choisies. On la notera $L_{\underline{s}}^T$.
- Le jacobien du robot : $\partial \underline{r}/\partial \underline{q}$ qui ne dépend que de la géométrie du robot et de la valeur des variables articulaires. On le notera J.
- Un terme, $\partial \underline{s}/\partial t$, dû au mouvement propre de l'objet par rapport à la caméra.

Par conséquent, en supposant l'objet observé immobile, la connaissance de J et $L_{\underline{s}}^T$ permet de relier $\dot{\underline{s}}$ à $\dot{\underline{q}}$.

Différentes méthodes existent pour obtenir cette relation. Nayar les a séparées en deux grandes catégories [Nayar 96] :

- Les méthodes basées sur un modèle de l'objet ou méthodes analytiques. On retrouve là les travaux de Feddema [Feddema 89a, Feddema 89b] et de Chaumette [Chaumette 90]. Rappelons que dans [Chaumette 90] l'auteur propose une méthode générale d'obtention de la matrice d'interaction [5]. Elle a notamment été mise en œuvre dans [Khadraoui 96] dans le cas d'informations visuelles issues de l'intersection d'un plan laser et d'une sphère.

[5] C'est d'ailleurs à partir de cette méthode que seront obtenus les résultats du chapitre 3.

- Les méthodes basées sur l'apprentissage. On pourrait encore introduire deux sous-catégories :

  – D'une part les méthodes d'apprentissage ou d'estimation en ligne ou même, comme par exemple dans [Hosoda 94] et [Jägersand 97], des méthodes itératives. Dans ces deux cas, les auteurs traitent des informations visuelles de type point et c'est directement le produit $L_{\underline{s}}^T J$ qu'ils estiment. Précisons par ailleurs que l'initialisation de l'algorithme n'est que partiellement résolu. De plus, dans [Jägersand 97] les auteurs soulignent l'influence indirecte de la durée du mouvement sur la précision de l'estimation, qui, en fait, dépend directement de la durée de l'apprentissage.

    Par contre, dans [Bien 93], les auteurs approximent la matrice $L_{\underline{s}}^T$ dans le cas particulier où la caméra ne peut se mouvoir qu'en translation suivant les axes x, y et z mais cette méthode a l'avantage d'être indépendante de la nature des informations visuelles. Pour ce faire, ils évaluent la conséquence d'une translation virtuelle ξ_j du porteur sur l'information visuelle s_i et approximent les éléments de $L_{\underline{s}}^T$ par l'expression $\Delta s_i / \Delta \xi_j$ en supposant les déplacements suffisamment faibles.

  – D'autre part des méthodes hors lignes. Le but est alors d'approximer en différents points de l'espace la matrice recherchée. Pour ce faire, dans [Domingo 96], les auteurs utilisent une approche similaire à [Bien 93] mais, en ligne, ils approximent $L_{\underline{s}}^T J$ par interpolation grâce aux différentes matrices $L_{\underline{s}}^T J$ obtenues hors lignes. Le fait d'utiliser une interpolation en ligne semble être une méthode courante, on la retrouve par exemple dans [Suh 93] pour approximer directement $\left(L_{\underline{s}}^T\right)^{-1}$ grâce à une approche neuronale utilisant une décomposition en fonctions de base. Ce sont, en fait, les fonctions d'appartenance d'un système flou, l'interpolation est ensuite réalisée par un autre réseau de neurones. Cette structure permet d'appréhender le calcul de l'inverse de la matrice d'interaction dans le cas d'informations visuelles relativement complexes telles l'orientation de l'image par rapport aux axes d'inertie ou des rapports de distance. Une démarche équivalente est décrite dans [Wu 97], par réseaux de neurones classiques, excepté le choix du réseau de neurones à utiliser qui est issu d'une décision émanant d'un réseau neuro-flou.

Remarque : concernant les informations visuelles issues d'une décomposition en composantes principales, un apprentissage est, certes, nécessaire mais pas dans le but de calculer la matrice d'interaction. Par exemple, dans [Deguchi 96] une méthode itérative est proposée (sans toutefois proposer de

méthode d'initialisation).

2.2.4 Les types d'architecture matérielle

La mise en œuvre d'un asservissement visuel 2D peut conduire à différents types de configurations matérielles. La plus répandue est certainement celle où la caméra est embarquée sur l'effecteur du robot [Chaumette 90, Jang 91, Papanikolopoulos 91, Coste-Manière 95, Deguchi 96, Hashimoto 96, Smith 97a]. En effet, une analyse de convergence de la loi de commande, du type de celle menée dans [Chaumette 90], permet de montrer une relative insensibilité du système bouclé vis-à-vis d'erreur de calibration du porteur comme du capteur. Signalons que des conclusions similaires sont rapportées dans [Weiss 87, Hashimoto 91, Espiau 93, Crowley 95, Hager 95, Wunsch 97a].

Une autre configuration intéressante, par exemple relatée dans [Kelly 96], consiste en une caméra fixe observant à la fois la scène et l'effecteur du robot. Les auteurs montrent, de même, l'insensibilité relative de leur loi de commande vis-à-vis des erreurs éventuelles de calibration, certes, dans le cas plus simple d'un robot à deux degrés de liberté.

Par ailleurs, des résultats similaires ont été récemment mis en exergue dans [Hager 97] et [Mitsuda 97] dans le cas d'une tête stéréoscopique observant à la fois l'effecteur et la scène (dans [Mitsuda 97] cette tête est active). Dans [Hager 97], grâce à la traduction de contraintes 3D en une fonction de tâche 2D, l'auteur réalise des tâches de positionnement particulièrement précises comme l'insertion d'une disquette dans un lecteur ou une opération de vissage. Dans [Mitsuda 97], les auteurs montrent qu'une approximation linéaire de la relation liant les coordonnées de l'image de l'effecteur à ses variables articulaires permet de l'amener simplement au point de fixation de la tête stéréoscopique active.

2.2.5 Les lois de commande

D'une manière générale, l'approche fonction de tâche introduite par Samson [Samson 88] en commande référencée capteur s'applique parfaitement au cas du capteur vision comme l'a montré Chaumette dans [Chaumette 90]. Les lois de commande qui en découlent s'expriment alors sous deux formes principales suivant les hypothèses émises sur le porteur :

- Dans le cas où sa dynamique est prise en compte, une loi de commande

en couple permet un découplage et une linéarisation dans l'espace de la fonction de tâche [Coste-Manière 95, Rives 97] ou permet de satisfaire d'autres critères (intégrant bien évidemment la dynamique du porteur) [Koivo 91, Kelly 96, Chuang 97].

- Dans le cas contraire, une commande cinématique est élaborée. On cherche alors une loi de commande assurant, par exemple, un comportement exponentiel découplé de l'erreur de tâche : $\dot{\underline{e}} = -\lambda \underline{e}$. La plus grande part des travaux en asservissement visuel 2D utilise une telle loi. Citons par exemple [Chaumette 90, Castano 94, Khadraoui 96, Hager 97, Hashimoto 97, Mitsuda 97]. Signalons, cependant, des variantes de cette approche où, soit des informations visuelles dynamiques peuvent être appréhendées [Martinet 96], soit des consignes visuelles fonction du temps [Hosoda 94, Berry 97].

D'un autre point de vue, certains auteurs abordent l'élaboration de la loi de commande sous le regard de l'automatique. Par exemple, Weiss a développé une commande anticipative avec adaptation paramétrique du modèle du système à commander [Weiss 87], Papanikolopoulos une commande de type PI, ou par placement de pôles [Papanikolopoulos 91]. Plus généralement, des commandes optimales basées sur la minimisation d'un critère temps-énergie peuvent être mises en œuvre [Hashimoto 96, Smith 97a].

Par ailleurs, lors de la réalisation de tâche de suivi ayant pour but de saisir un objet en mouvement, le mouvement propre de cet objet n'est que rarement pris en compte. Il est même souvent considéré comme un facteur exogène, comme dans [Smith 97a] par exemple. Toutefois, certains auteurs mettent en œuvre des observateurs, généralement basé sur un filtre de Kalman, pour l'estimer et ainsi le prendre en compte dans la loi de commande [Papanikolopoulos 91, Koivo 91]. Signalons que le pendant peut tout aussi bien être réalisé grâce à l'approche fonction de tâche [Bensalah 96].

2.2.6 L'élargissement du champ applicatif

Les lois de commande issues d'un asservissement basé image dépendent par nature des informations visuelles choisies permettant ainsi la réalisation de la tâche escomptée. Dans ce contexte, la tâche à exécuter est dépendante de l'objet observé par la présence ou non d'informations visuelles que l'on sait appréhender, c'est-à-dire, plus précisément, dont on est capable de calculer la matrice d'interaction associée. C'est pourquoi, comme nous l'avons vu en 2.2.2, de nombreux travaux traitent d'informations visuelles pouvant caractériser l'aspect global de

l'image. Rappelons, cependant, qu'ils sont tous orientés sur des méthodes d'apprentissage (cf. 2.2.3). De ce fait, à chaque nouvel objet, cette façon de faire conduit à une nouvelle phase d'apprentissage généralement assez lourde [Deguchi 96] et ce, même si le type d'information visuelle reste inchangé. En fait, on ne sait pas, indépendamment de l'objet, exprimer la variation de telles informations par rapport au mouvement du porteur. De plus, l'utilisation d'une telle matrice issue d'une approximation influe sur la vitesse de convergence de la loi de commande et peut-être pénalisante à cette égard. Il y aurait donc tout intérêt, autant du point de vue applicatif que théorique, à élaborer des lois de commande basées sur une approche analytique. C'est en fait ce que nous nous proposons de faire dans le chapitre 3. Nous y traitons, plus précisément, d'informations visuelles issues d'un contour que nous supposons quelconque et complexe.

D'un tout autre point de vue, l'ensemble des travaux menés en asservissement visuel 2D suppose la consigne visuelle connue ou, en d'autres termes, que l'image, une fois la tâche réalisée, puisse être parfaitement décrite [6]. Il s'agit là, en fait, d'une hypothèse plutôt forte dont la conséquence principale est de limiter grandement le champ applicatif. En effet, exceptés les produits manufacturiers pour lesquels un modèle de l'objet est souvent disponible [7] et par voie de conséquence les informations visuelles désirées, tous les autres produits ne permettent pas une description précise de l'image que l'on souhaite obtenir. Le plus souvent il s'agit soit de produits déformables, soit encore de produits que l'on rencontre dans la nature ou même de type alimentaire. Effectivement, le genre de produit rencontré dans les industries agro-alimentaires illustrent parfaitement ce cas de figure. En effet, considérons par exemple la réalisation d'une tâche de positionnement par rapport à un des objets du type de ceux représentés sur la figure 2.2, même si le produit ou l'objet en question appartient à une certaine classe, la classe « jambon » en l'occurrence, ils sont suffisamment différents les uns des autres pour qu'une seule consigne visuelle ne puisse décrire entièrement cette classe. Dans de telles applications, la consigne visuelle est en fait inconnue puisqu'il existe finalement autant de consignes que d'objets de la classe. Ce nouvel aspect des asservissements visuels nous a conduit à mener les travaux rapportés au chapitre 4. Nous y décrivons une façon de réaliser une tâche particulière de positionnement, consistant à amener la caméra en face de l'objet observé, objet supposé quelconque et inconnu.

Les travaux relatés dans ces deux chapitres permettrons, en somme, d'élargir significativement le domaine d'application des asservissements visuels 2D.

[6] Précisons que nous ne considérons ici que des tâches évoluées comme par exemple la réalisation d'une tâche de positionnement précis.

[7] Par exemple par le biais des modèles fournis par un logiciel de CFA0.

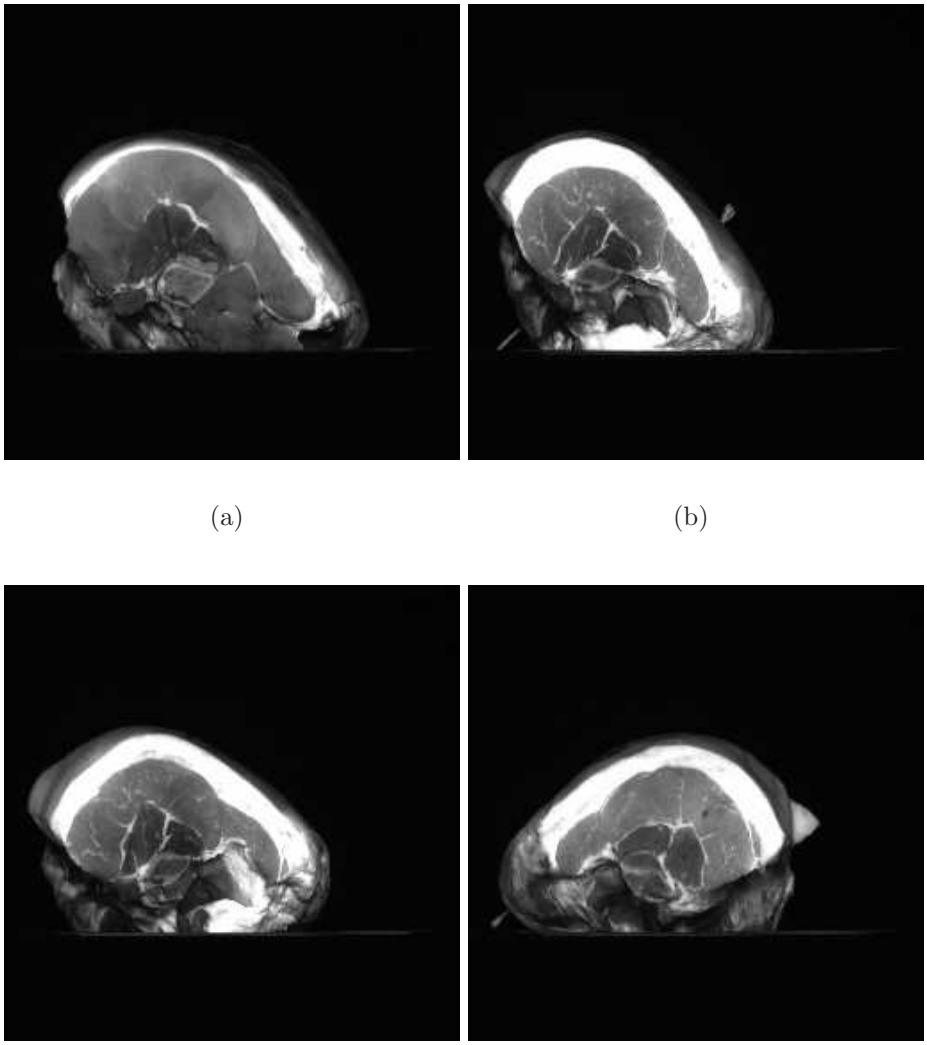


FIG. 2.2 – Exemple de cas où la consigne visuelle est inconnue

Chapitre 3

Asservissement visuel par rapport à un objet de forme complexe

Comme nous l'avons vu au chapitre précédent, la réalisation d'une tâche robotique par asservissement visuel basé image nécessite la connaissance de la matrice dite d'interaction. Aussi, allons-nous nous intéressé, tout au long de ce chapitre, au calcul de cette matrice dans le cas particulier où les informations visuelles sont issues du contour d'un objet observé, contour que l'on suppose de forme complexe pour ne pas limiter le champ applicatif. Après un bref rappel de l'approche fonction de tâche puis de la méthode générale de calcul de la matrice d'interaction, proposée par F. Chaumette [Chaumette 90], nous montrons comment il est possible d'en obtenir une approximation. Pour ce faire, nous reformulons l'approche qu'il propose en un problème de minimisation d'une erreur de modélisation du contour. Puis, après un court état de l'art des types de représentation des courbes planes, nous illustrons cette nouvelle façon de procéder sur deux exemples en réalisant différentes tâches de positionnement. Des résultats de simulation et d'expérimentation valident les développements théoriques.

3.1 L'approche fonction de tâche

Selon la définition rappelée en 2.2.3 et dans le cas où l'objet observé est immobile, les informations visuelles sont reliées à la vitesse du porteur par la

relation :

$$\dot{\underline{s}} = L_{\underline{s}}^T T_c \tag{3.1}$$

où :

- T_c est le torseur cinématique. Il représente la vitesse de la caméra par rapport à l'objet et s'écrit $T_c = (\underline{V}^T, \underline{\Omega}^T)^T$ où $\underline{V} = (v_x, v_y, v_z)^T$ et $\underline{\Omega} = (\omega_x, \omega_y, \omega_z)^T$ sont respectivement les vitesses de translation et de rotation exprimées dans le repère de la caméra ;
- $\dot{\underline{s}}$ est la vitesse des informations visuelles ;
- $L_{\underline{s}}^T$ est la matrice d'interaction relative aux informations visuelles $\underline{s}$.

Dans ce contexte, la réalisation d'une tâche par asservissement visuel 2D peut très bien s'exprimer, comme l'a montré Chaumette [Chaumette 90], en termes de régulation à zéro de la fonction de tâche suivante :

$$\underline{e}(\underline{r}, t) = \widehat{L_{\underline{s}}^T}^+ (\underline{s}(\underline{r}, t) - \underline{s}^*) \tag{3.2}$$

où :

- $\widehat{L_{\underline{s}}^T}^+$ est la pseudo-inverse d'une approximation de la matrice d'interaction, elle est de dimension n x m et permet de prendre en compte les cas où m est différent de n (avec m et n définis en 2.1.4) ;
- $\underline{s}^*$ représente la valeur des informations visuelles à la position finale souhaitée du porteur ;
- $\underline{e}$ est le vecteur fonction de tâche, de dimension n. Il dépend du temps t et de la situation du porteur $\underline{r}$.

Dans le cas où on l'on désire obtenir un comportement exponentiel découplé de la fonction de tâche, à savoir :

$$\dot{\underline{e}} = -\lambda \underline{e} \tag{3.3}$$

on peut montrer que le torseur cinématique s'écrit [Chaumette 90] :

$$T_c = -\lambda \underline{e} \tag{3.4}$$

Signalons, pour terminer ces rappels, que dans le cas d'une tâche référencée vision, la condition suffisante de stabilité établie par Samson [Samson 91] devient :

$$\widehat{L_{\underline{s}}^T}^+ L_{\underline{s}}^T > 0 \tag{3.5}$$

3.2 Méthode générale de calcul de la matrice d'interaction

Soit $\{p_1, \ldots, p_k\}$ l'ensemble des k primitives tridimensionnelles propres à l'objet considéré et soit $\{P_1, \ldots, P_k\}$ l'ensemble des k primitives bidimensionnelles extraites de l'image de cet objet. Ces primitives sont obtenues par projection perspective de tous les points appartenant aux primitives tridimensionnelles observées, soit, pour un point décrit par $\underline{x} = (x, y, z)^T$, par :

$$\underline{X} = f\frac{\underline{x}}{z} \tag{3.6}$$

où $\underline{X} = (X, Y, f)^T$ représente la projection de $\underline{x}$ dans le capteur et f la distance focale que nous supposerons, sans perte de généralité, égale à l'unité par la suite.

Calculons à présent la matrice d'interaction relative à l'information visuelle s constituée des primitives P_i $(i = 1 \ldots k)$:

$$L_s^T = \frac{\partial s}{\partial \underline{r}} = \sum_{i=1}^{i=k} \frac{\partial s}{\partial \underline{P_i}} \frac{\partial \underline{P_i}}{\partial \underline{r}} \tag{3.7}$$

où $\underline{P_i}$ est le paramétrage associé à P_i, supposé de dimension m_i.

Cette équation fait apparaître une somme de k termes constitués eux-mêmes de deux termes :

- $\partial s / \partial \underline{P_i}$ qui s'obtient générallement de façon triviale ;
- $\partial \underline{P_i} / \partial \underline{r}$ qui représente la matrice d'interaction relative à l'information visuelle P_i et dont la méthode de calcul est détaillée ci-après.

Dans le souci de ne pas alourdir inutilement les notations, nous ometterons maintenant les indices i.

D'une façon générale, les primitives tridimensionnelles peuvent s'écrire sous la forme d'une équation implicite :

$$h(\underline{x}, \underline{p}) = 0 \tag{3.8}$$

où $\underline{p}$ est le paramétrage de p et $\underline{x}$ les coordonnées d'un point de la primitive considérée. En introduisant la projection de $\underline{x}$ dans l'image, il vient d'après (3.6) :

$$h'(\underline{X}, 1/z, \underline{p}) = 0 \tag{3.9}$$

En supposant maintenant que $\partial h'/\partial(1/z) \neq 0$ et en appliquant le théorème des fonctions implicites, on peut extraire une expression de l'inverse de la profondeur z :

$$1/z = \mu(\underline{X}, \underline{p}) \tag{3.10}$$

Par ailleurs, notons $g(\underline{X}, \underline{P}) = 0$ l'équation implicite de la primitive 2D issue de la projection dans l'image de la primitive 3D décrite par (3.8). L'hypothèse de rigidité de la primitive décrite par g, signifiant simplement qu'elle ne va pas changer de nature au cours du mouvement, se traduit par :

$$\forall \underline{X} \in P,\ \dot{g}(\underline{X}, \underline{P}) = 0 \tag{3.11}$$

soit :

$$\frac{\partial g}{\partial \underline{X}} \dot{\underline{X}} + \frac{\partial g}{\partial \underline{P}} \dot{\underline{P}} = 0 \tag{3.12}$$

et qui implique :

$$\frac{\partial g}{\partial \underline{P}} \dot{\underline{P}} = -\frac{\partial g}{\partial \underline{X}} L_{\underline{X}}^T T_c \tag{3.13}$$

où $L_{\underline{X}}^T$ est la matrice d'interaction relative à l'information visuelle $\underline{X}$. Elle est donnée par ([Feddema 89a, Chaumette 90]) :

$$L_{\underline{X}}^T = \begin{pmatrix} -1/z & 0 & X/z & XY & -1-X^2 & Y \\ 0 & -1/z & Y/z & 1+Y^2 & -XY & -X \end{pmatrix} \tag{3.14}$$

$1/z$ étant fourni par (3.10).

L'équation (3.13) est valide pour l'ensemble des points constituant P. Par conséquent, en en choisissant m, on obtient le système linéaire suivant :

$$\begin{pmatrix} \left.\dfrac{\partial g}{\partial \underline{P}}\right|_{\underline{X}_1} \\ \vdots \\ \left.\dfrac{\partial g}{\partial \underline{P}}\right|_{\underline{X}_m} \end{pmatrix} \dot{\underline{P}} = -\begin{pmatrix} \dfrac{\partial g}{\partial \underline{X}_1} L_{\underline{X}_1}^T \\ \vdots \\ \dfrac{\partial g}{\partial \underline{X}_m} L_{\underline{X}_m}^T \end{pmatrix} T_c \tag{3.15}$$

Si les m points sont choisis de façon indépendante et si la représentation de g par le paramétrage $\underline{P}$ est minimale alors ce système peut s'inverser et on obtient la matrice d'interaction recherchée :

$$L_{\underline{P}}^T = -\begin{pmatrix} \left.\dfrac{\partial g}{\partial \underline{P}}\right|_{\underline{X}_1} \\ \vdots \\ \left.\dfrac{\partial g}{\partial \underline{P}}\right|_{\underline{X}_m} \end{pmatrix}^{-1} \begin{pmatrix} \dfrac{\partial g}{\partial \underline{X}_1} L_{\underline{X}_1}^T \\ \vdots \\ \dfrac{\partial g}{\partial \underline{X}_m} L_{\underline{X}_m}^T \end{pmatrix} \tag{3.16}$$

3.3 Méthode de calcul d'une approximation de la matrice d'interaction

La méthode developpée ci-dessus repose essentiellement sur les deux points suivants :

- D'une part, sur l'obtention de la fonction μ. Cependant, il peut être analytiquement impossible d'exprimer explicitement $1/z = \mu(\underline{X}, \underline{p})$. Citons par exemple le cas où la primitive 3D est une fonction algébrique de degré élevé.
- D'autre part, sur la connaissance de la fonction $g(\underline{X}, \underline{P}) = 0$. De même, cette fonction peut ne pas être disponible. Par exemple, dans le cas d'une primitive 3D quelconque, la fonction $g(\underline{X}, \underline{P}) = 0$ est en fait inconnue et, seules des approximations peuvent être disponibles.

Ces deux points nous conduisent à ne raisonner que dans l'image ; nous espérons ainsi proposer une loi de commande indépendante d'un modèle de l'objet observé. Seule est supposée connue la consigne visuelle, ou en d'autres termes, une représentation 2D de l'image de l'objet, en somme une photo de ce que l'on désire voir une fois le porteur arrivé en position souhaitée. Il en découle deux conséquences importantes :

- Nous devons considérer maintenant que nous ne connaissons plus les relations liant la primitive 2D à la primitive 3D, en particulier la façon dont varient les profondeurs. Dans le cas d'un objet plan, on pourrait toujours les modéliser par $1/z = aX + bY + c$, par contre, dans le cas général, on devrait imaginer des modèles plus complexes. Cependant, comme nous allons le voir par la suite, on peut aussi les supposer constantes sans pour autant perturber outre mesure la convergence de la loi de commande.
- Ne connaissant plus la forme de la primitive 2D, nous devons nous contenter d'une approximation $\hat{g}$ de g.

Toutefois, avec cette façon de procéder, comme la fonction $\hat{g}$ n'est plus l'image de la primitive décrite par h mais une approximation, la contrainte de rigidité n'est plus analytiquement vérifiée. En effet, rien ne prouve qu'à chaque instant la primitive 3D se projette formellement sous la forme $\hat{g} = 0$. Par contre, si à chaque instant on sait réaliser $g - \hat{g} \longrightarrow 0$, on aura alors $\dot{g}(\underline{X}, \underline{P}) \longrightarrow 0$ et on pourra opérer ensuite comme en 3.2 pour obtenir une matrice d'interaction approximée.

Finalement, le point clé de cette manière d'aborder le calcul de la matrice d'interaction, réside dans l'obtention de la fonction $\hat{g}$, c'est-à-dire à un problème de modélisation d'une courbe plane. Aussi, allons-nous, dans le prochain paragraphe, nous intéresser aux différentes approches existantes.

3.4 Modélisation d'une courbe plane

D'un point de vue général, il n'existe que quatre grandes familles de représentation d'une courbe plane : les représentations cartésienne, sous forme implicite, polaire et paramétrique. Nous ne développerons pas la représentation cartésienne car elle ne permet pas de décrire les contours fermés ce qui est par trop limitatif vis-à-vis des objectifs que l'on s'est fixé (signalons toutefois son utilisation dans le cas particulier de la droite en stéréovision [Ayache 89] ou en asservissement visuel [Chaumette 90]). En revanche, nous allons maintenant nous intéresser, brièvement, aux autres types de représentation.

3.4.1 Représentation sous forme implicite

Les fonctions de ce type les plus utilisées, et depuis longtemps, sont les fonctions algébriques. Elles sont de la forme :

$$g(X,Y) = \sum_{\substack{i+j\leq p \\ i\geq 0 \\ j\geq 0}} a_{ij} X^i Y^j = 0 \tag{3.17}$$

Elles ont tout d'abord été utilisées en aéronautique dans les années 1940 (voir [Iglésias 92] pour un état de l'art exhaustif sur le sujet) mais sont maintenant de plus en plus délaissées au profit des fonctions définies sous une forme paramétrique. En effet, les courbes algébriques n'étant définies que par un petit nombre de paramètres, elles ne possèdent que très peu de degrés de liberté et ne permettent donc pas de modéliser des formes complexes. Par ailleurs, l'identification des a_{ij} peut s'avérer délicate. En effet, on peut, par exemple, montrer que les matrices intervenant dans la méthode choisie (le plus souvent la méthode des moindres carrés) deviennent mal conditionnées dès que le degré p augmente. Néanmoins, elles sont utilisées en stéréovision en tant que primitives pour la mise en correspondance [Ma 93, An 96, Li 96]. Signalons, de plus, qu'elles possèdent la propriété suivante très intéressante [An 96] :

Propriété 1 *La transformation projective d'une courbe algébrique plane de degré d est une courbe algébrique de même degré.*

Par conséquent, dans le cas d'un objet plan de forme simple, la méthode rappelée en 3.2 est tout à fait applicable.

Plus récemment, d'autres types de fonction implicites ont vu le jour, essentiellement en imagerie graphique :

- les superquadriques, popularisées par Piet Hein [Gardiner 65]. Elles s'expriment de façon canonique sous la forme :

$$\left|\frac{X}{s_1}\right|^{\gamma_1} + \left|\frac{Y}{s_2}\right|^{\gamma_2} = 1 \tag{3.18}$$

  Ce type de fonction permet de décrire continûment, suivant les valeurs de γ_1 et γ_2, des formes convexes ou concaves comme des ellipses, des rectangles, des losanges ou encore des losanges pincés. Malheureusement, leurs représentations graphiques sont symétriques par rapport aux axes X et Y.

  Toutefois, comme elles ne sont décrites que par très peu de paramètres, elles sont relativement insensibles aux occlusions partielles, cette propriété intéressante a notamment été mise en évidence dans [Bascle 94] lors de la réalisation de suivi de contours dans des séquences d'images.

- les hyperquadriques, introduites par Hanson [Hanson 88]. Elles sont décrites par une équation de la forme :

$$\sum_{i=1}^{i=N} |a_i X + b_i Y + c_i|^{\gamma_i} = 1 \tag{3.19}$$

  Contrairement aux superquadriques, les hyperquadriques ne présentent pas, dans le cas général, d'axe de symétrie. En revanche, la modélisation de formes complexes entraîne l'emploi d'un nombre conséquent de paramètres [Han 93]. De plus, leur identification n'est pas très aisée [Han 93], de surcroît à la cadence vidéo.

3.4.2 Représentation sous forme polaire

Considérons un repère associé au plan image et choisissons le tel que son centre O et l'axe des z coïncident respectivement avec le centre optique et avec

l'axe optique de la caméra. On lui associe le trièdre orthonormé direct $(\underline{u}_x, \underline{u}_y, \underline{u}_z)$. Un point M du contour considéré s'écrit alors sous forme polaire par :

$$\underline{OM} = \rho \underline{u}_\theta \tag{3.20}$$

avec $\underline{u}_\theta = \cos\theta \underline{u}_x + \sin\theta \underline{u}_y$.

Telle qu'elle, cette représentation n'est pas utilisée dans le domaine de la vision, on lui préfère *la signature polaire*. C'est une fonction $\rho_c = \phi(\theta_c)$ définie de la façon suivante : ρ_c est la distance du point M, de coordonnées $X = \rho\cos\theta$ et $Y = \rho\sin\theta$, au point G de coordonnées X_c et Y_c ; G étant le centre de gravité de l'image de l'objet observé et M un point du contour (voir figure 3.1).

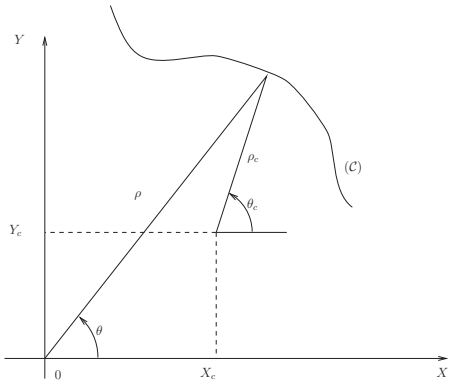


FIG. 3.1 – Signature polaire

Cette façon de faire permet, en fait, de transcrire le problème initial de modélisation bidimensionnelle $g(\underline{X}, \underline{P}) = 0$ en un problème, plus simple, de modélisation monodimensionnelle $\rho_c = \phi(\theta_c, \underline{P})$.

Habituellement, ϕ étant 2π périodique, on la décompose en série de Fourier. Cela permet de manipuler le contour considéré en une somme de signaux élémentaires ; cette approche est, par conséquent, très utilisée en reconnaissance de formes d'objets complexes pour extraire des informations pertinentes du contour de l'image de l'objet [Liao 94, Tao 95, Russ 95], elle est même souvent disponible sur des systèmes industriels [Besançon 88].

Par ailleurs, signalons, d'une part, l'apparition d'un capteur CCD basé sur une transformation polaire [Tistarelli 93] réalisant ainsi un capteur à fovéa et, d'autre part, l'intérêt de ce type de capteur pour l'obtention d'informations de type dynamique, comme le flot optique ou le temps à collision [Tistarelli 93], ou géométrique pour la reconnaissance de primitive image [Lim 97].

3.4.3 Représentation sous forme paramétrique

La description sous forme paramétrique d'une courbe plane revient à considérer un arc γ défini par une paramétrisation $f\ : I \longrightarrow \mathbb{R}^2$ telle que :

$$\begin{array}{rcl} f: & I \longrightarrow & \mathbb{R}^2 \\ & u \longrightarrow & f(u) = \begin{pmatrix} X(u) \\ Y(u) \end{pmatrix} \end{array}$$

où I est un intervalle de $\mathbb{R}$ décrivant le domaine de variation de la variable u.

Signalons que, contrairement à une description basée sur la signature polaire, les fonctions paramétriques ont l'avantage d'appréhender des fonctions non holomorphes. Un exemple de contour conduisant à une telle fonction est représenté sur la figure 3.2.

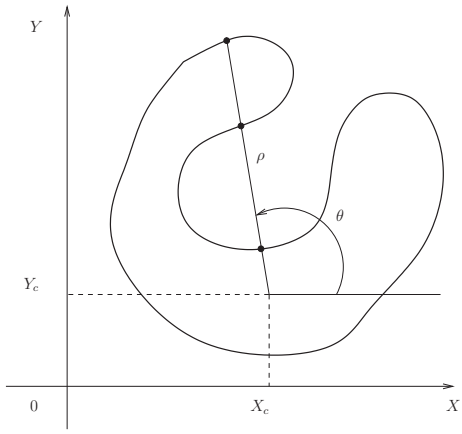


FIG. 3.2 – Un exemple de contour dont la signature polaire n'est pas une fonction holomorphe

Il existe de nombreuses façons d'écrire une fonction paramétrique, toutefois deux grandes familles émergent.

3.4.3.1 Utilisation d'une décomposition sur une base de fonctions

Une première famille de fonctions paramétriques peut s'écrire sous la forme :

$$\underline{X}(u) = \sum_{k=0}^{k=p} \underline{A}_k \Phi_k(u) \tag{3.21}$$

où $\underline{A_k}$ est un coefficient vectoriel et Φ_k une fonction de la variable réelle u.

Habituellement les fonctions Φ_k sont choisies de telle façon à réaliser une base de fonctions orthogonales. Il s'en suit deux types de représentations très utilisées dans la communauté de la vision par ordinateur :

- Les descripteurs de Fourier ; on choisit alors la famille de fonctions orthogonales définie par :

$$\Phi = \{1, \cos u, \ldots, \cos hu, \sin u, \ldots, \sin hu\} \tag{3.22}$$

  où h est le nombre d'harmoniques.

  Nous ne saurions relater ici tous les travaux utilisant les descripteurs de Fourier tant leurs applications sont répandues. Néanmoins, citons quelques un de leurs domaines de prédilection : la modélisation de contours, ouverts ou fermés [Kuhl 82, Staib 92] et la reconnaissance de formes [Mitchell 84, Persoon 86].

- Les courbes à pôles ; dans ce cas, les fonctions Φ_k vérifient la condition de Cauchy :

$$\forall u \in I, \ \sum_{k=0}^{k=p} \Phi_k(u) = 1 \tag{3.23}$$

  Cette famille de courbes regroupe pour l'essentiel deux types de courbes :

  - Les courbes de Bézier ; les Φ_k sont alors les polynômes de Bernstein :

$$\Phi_k(u) = B_{k,p} = C_p^k u^k (1-u)^{p-k} \tag{3.24}$$

    avec $C_p^k = p!/k!(p-k)!$ et les $\underline{A_k}$, p+1 points constituant ainsi *le polygone convexe* (voir un exemple sur la figure 3.3).

    Remarquons que le degré de Φ_k est exactement le nombre de points A_k. Par conséquent, si le contour à modéliser est particulièrement complexe, le degré des Φ_k sera nécessairement élevé, ce qui est considéré comme l'inconvénient majeur de ce type de courbe. En effet, il est bien connu que des polynômes de degrés élevés conduisent à des graphes présentant des oscillations de fréquence d'autant plus élevée que le degré du polynôme est élevé.

  - Les fonctions B-spline ; les Φ_k sont alors des polynômes $N_{k,r}(u)$ de degré r définis par la relation de récurrence :

$$\Phi_k = N_{k,r}(u) = \frac{u - u_k}{u_{k+j} - u_k} N_{k,j-1}(u) + \frac{u_{k+j+1} - u}{u_{k+j+1} - u_{k+1}} N_{k+1,j-1}(u) \tag{3.25}$$

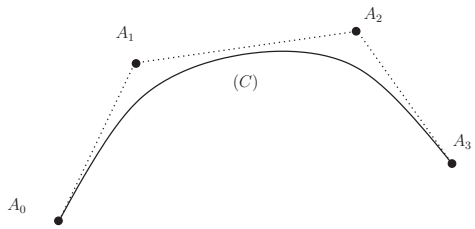


FIG. 3.3 – Contour décrit par une courbe de Bézier de degré 4 et représentation de son polygone convexe

initialisée par :

$$\text{si } u_k \leq u < u_{k+1} \text{ alors } N_{k,0}(u) = 1 \text{ sinon } 0 \tag{3.26}$$

où les u_i forment une suite croissante de valeurs réelles, cette suite est appelée *le vecteur nodal*.

Contrairement aux courbes de Bézier, le degré des polynômes peut être choisi indépendamment du nombre de points constituant le polygone convexe. En outre, on peut montrer que le cas particulier où $r = p + 1$ conduit à une courbe de Bézier. C'est pourquoi les fonctions B-spline sont le plus souvent préférées aux courbes de Bézier, elles en sont une généralisation. On les retrouve maintenant dans de nombreux domaines, dès les phases de conception et de fabrication assistées par ordinateur [Marty 93], en modélisation de forme [Iglésias 92], et en traitement d'images. Nous renvoyons le lecteur intéressé à la lecture de [Bascle 94] où un état de l'art concernant ce dernier point est proposé, les contours actifs y étant plus particulièrement discutés.

3.4.3.2 Utilisation d'une décomposition en fonctions rationnelles

Les courbes à pôles décrites ci-dessus peuvent en fait se généraliser, grâce à l'utilisation des coordonnées homogènes, sous la forme suivante [Besl 88] :

$$\underline{X}(u) = \frac{\sum_{k=0}^{k=l} \underline{A}'_k u^k}{\sum_{k=0}^{k=l} w_k u^k} \tag{3.27}$$

où les w_k sont des valeurs réelles.

Dans le cas des fonctions B-spline, cette façon de procéder conduit alors aux B-spline rationnelles ; dans le cas des courbes de Bézier aux courbes de Bézier rationnelles. On peut, de même, introduire des descripteurs de Fourier ¡¡ rationnels ¿¿ :

$$\begin{cases} X(u) = \dfrac{A_0 + \sum_{k=1}^{k=h} A_k \cos ku + B_k \sin ku}{1 + \sum_{k=1}^{k=h} E_k \cos ku + F_k \sin ku} \\ Y(u) = \dfrac{C_0 + \sum_{k=1}^{k=h} C_k \cos ku + D_k \sin ku}{1 + \sum_{k=1}^{k=h} G_k \cos ku + H_k \sin ku} \end{cases} \tag{3.28}$$

que l'on peut transformer en fonction rationnelle polynômiale à partir de la formule de Moivre puis en exprimant les lignes trigonométriques sinus et cosinus en fonction de la tangente de l'arc moitié. Soit :

$$\begin{cases} \cos ku &= \sum_{0 \leq 2l \leq k} (-1)^l C_k^{2l} (\cos u)^{k-2l} (\sin u)^{2l} \\ \sin ku &= \sum_{0 < 2l+1 \leq k} (-1)^l C_k^{2l+1} (\cos u)^{k-2l-1} (\sin u)^{2l+1} \end{cases} \tag{3.29}$$

puis :

$$\begin{cases} \cos u &= \dfrac{1-t^2}{1+t^2} \\ \sin u &= \dfrac{2t}{1+t^2} \end{cases} \tag{3.30}$$

où on a posé $t = \tan u/2$.

On dispose ainsi, sous une forme unique, des fonctions paramétriques les plus réputées quant à leur possibilité de modélisation de formes quelconques.

Ce type de représentation (3.27) possède alors la propriété intéressante suivante [Böhm 84] :

Propriété 2 *La projection perspective d'une courbe rationnelle est une courbe rationnelle.*

Dans ce cas particulier, une telle représentation conduit alors une matrice d'interaction non approximée puisque la contrainte de rigidité est alors vérifiée.

Les principales formes de représentation d'une courbe plane étant rappelées, nous allons maintenant, dans les paragraphes suivants, développer l'approche décrite en 3.3 dans les cas de figure qui nous ont paru les plus intéressants, à savoir le calcul de la matrice d'interaction relatif à des informations visuelles extraites d'un contour décrit soit à l'aide de sa signature polaire, soit d'une fonction paramétrique. Ces résultats, une fois le processus d'extraction des informations visuelles décrit, seront ensuite appliqués à la réalisation de différentes tâches de positionnement conformément à l'approche fonction de tâche rappelée en 3.1.

3.5 Asservissement visuel basé sur la signature polaire

3.5.1 Calcul de la matrice d'interaction

Soit un point m de coordonnées (x, y, z) exprimées dans un repère associé à la caméra comme celui décrit en 3.4.2. Il se projette, par transformation perspective, sur le plan image en M tel que $\underline{OM} = (X, Y, 1)^T$.

Par la suite, pour ne pas alourdir les notations, on exprimera, sous forme polaire, les coordonnées d'un point dans le repère barycentrique par (ρ, θ). $\underline{OM}$ s'écrit alors :

$$\underline{OM} = (X_c + \rho\cos\theta)\underline{u}_x + (Y_c + \rho\sin\theta)\underline{u}_y \tag{3.31}$$

On en déduit une expression de la vitesse de M :

$$\begin{cases} \dot{X} &= \dfrac{d}{dt}(\rho\cos\theta) + \dot{X}_c \\ \dot{Y} &= \dfrac{d}{dt}(\rho\sin\theta) + \dot{Y}_c \end{cases} \tag{3.32}$$

et en développant, il vient :

$$\begin{cases} \dot{X} &= \dot{\rho}\cos\theta - \rho\dot{\theta}\sin\theta + \dot{X}_c \\ \dot{Y} &= \dot{\rho}\sin\theta + \rho\dot{\theta}\cos\theta + \dot{Y}_c \end{cases} \tag{3.33}$$

La signature polaire étant périodique, on peut la décomposer en série de Fourier, soit :

$$\rho(\theta) = a_0 + \sum_{k=1}^{k=h} a_k \cos k\theta + b_k \sin k\theta \tag{3.34}$$

où h est le nombre d'harmoniques pris en compte.

En notant $\underline{P} = (a_0, \cdots, a_h, b_1, \cdots, b_h)^T$, (3.33) devient alors :

$$\left\{ \begin{array}{rcl} \dot{X} & = & \dot{X}_c + \left(\dfrac{\partial \rho}{\partial \theta} \cos\theta - \rho \sin\theta \right) \dot{\theta} + \dfrac{\partial \rho}{\partial \underline{P}} \dot{\underline{P}} \cos\theta \\ \dot{Y} & = & \dot{Y}_c + \left(\dfrac{\partial \rho}{\partial \theta} \sin\theta + \rho \cos\theta \right) \dot{\theta} + \dfrac{\partial \rho}{\partial \underline{P}} \dot{\underline{P}} \sin\theta \end{array} \right. \tag{3.35}$$

et en éliminant $\dot{\theta}$, il vient :

$$\dot{X} - \alpha \dot{Y} = \dot{X}_c - \alpha \dot{Y}_c + \beta \frac{\partial \rho}{\partial \underline{P}} \dot{\underline{P}} \tag{3.36}$$

où on a posé :

$$\alpha = \frac{\dfrac{\partial \rho}{\partial \theta} \cos\theta - \rho \sin\theta}{\dfrac{\partial \rho}{\partial \theta} \sin\theta + \rho \cos\theta} \tag{3.37}$$

et :

$$\beta = \cos\theta - \alpha \sin\theta \tag{3.38}$$

On choisit alors pour information visuelle le vecteur $\underline{Q} = (X_c, Y_c, \underline{P}^T)$. La relation (3.36) s'écrit alors :

$$\dot{X} - \alpha \dot{Y} = \left(1, -\alpha, \beta \frac{\partial \rho}{\partial \underline{P}} \right) \dot{\underline{Q}}^T \tag{3.39}$$

et en choisissant $m = 2h + 3$ couples (ρ, θ) indépendants, puis en inversant le système ainsi obtenu, il vient :

$$\dot{\underline{Q}}^T = \begin{pmatrix} 1 & -\alpha_1 & \beta_1 \left. \dfrac{\partial \rho}{\partial \underline{P}} \right|_{\rho_1, \theta_1} \\ \vdots & \vdots & \vdots \\ 1 & -\alpha_m & \beta_m \left. \dfrac{\partial \rho}{\partial \underline{P}} \right|_{\rho_m, \theta_m} \end{pmatrix}^{-1} \begin{pmatrix} \dot{X}_1 - \alpha_1 \dot{Y}_1 \\ \vdots \\ \dot{X}_m - \alpha_m \dot{Y}_m \end{pmatrix} \tag{3.40}$$

par suite, en remplacant $\dot{X}$ et $\dot{Y}$ grâce à $\dot{X} = L_X^T T_c$ et $\dot{Y} = L_Y^T T_c$, L_X^T et L_Y^T étant donnés par (3.14), on obtient finalement la matrice recherchée :

$$\widehat{L_{\underline{Q}}^T} = \begin{pmatrix} 1 & -\alpha_1 & \beta_1 \left. \dfrac{\partial \rho}{\partial \underline{P}} \right|_{\rho_1, \theta_1} \\ \vdots & \vdots & \vdots \\ 1 & -\alpha_m & \beta_m \left. \dfrac{\partial \rho}{\partial \underline{P}} \right|_{\rho_m, \theta_m} \end{pmatrix}^{-1} \begin{pmatrix} L_{X_1}^T - \alpha_1 L_{Y_1}^T \\ \vdots \\ L_{X_m}^T - \alpha_m L_{Y_m}^T \end{pmatrix} \tag{3.41}$$

soit pour une information visuelle particulière :

$$\widehat{L_{Q_i}^T} = \sum_{j=1}^{j=m} m_{i,j} \left(L_{X_j}^T - \alpha_j L_{Y_j}^T \right) \tag{3.42}$$

où $m_{i,j}$ est le terme générique de la matrice M telle que :

$$M = \begin{pmatrix} 1 & -\alpha_1 & \beta_1 \left.\dfrac{\partial \rho}{\partial \underline{P}}\right|_{\rho_1,\theta_1} \\ \vdots & \vdots & \vdots \\ 1 & -\alpha_m & \beta_m \left.\dfrac{\partial \rho}{\partial \underline{P}}\right|_{\rho_m,\theta_m} \end{pmatrix}^{-1} \tag{3.43}$$

Étudions plus en détail la matrice $\widehat{L_{\underline{Q}}^T}$. Pour ce faire, on considère une ligne i particulière. L'utilisation explicite de la matrice d'interaction dans le cas d'un point (3.14), de (3.31) et (3.34) dans (3.42) conduit alors aux relations suivantes :

$$\begin{cases} \widehat{L_{\underline{Q}_{i,1}}^T} = \sum_j m_{i,j}(-1/z_j) \\ \widehat{L_{\underline{Q}_{i,2}}^T} = \sum_j m_{i,j}\alpha_j/z_j \\ \widehat{L_{\underline{Q}_{i,3}}^T} = X_c \sum_j m_{i,j}/z_j - Y_c \sum_j \alpha_j m_{i,j}/z_j + \sum_j m_{i,j}\beta_j\rho_j \\ \widehat{L_{\underline{Q}_{i,4}}^T} = X_c Y_c \sum_j m_{i,j} - (1+Y_c^2)\sum_j m_{i,j}\alpha_j + \varphi_i(\underline{X}_c, \underline{\Theta}, \underline{\mathcal{M}}_i) \\ \widehat{L_{\underline{Q}_{i,5}}^T} = -(1+X_c^2)\sum_j m_{i,j} + X_c Y_c \sum_j m_{i,j}\alpha_j + \psi_i(\underline{X}_c, \underline{\Theta}, \underline{\mathcal{M}}_i) \\ \widehat{L_{\underline{Q}_{i,6}}^T} = Y_c \sum_j m_{i,j} + X_c \sum_j m_{i,j}\alpha_j + \sum_j m_{i,j}\gamma_j\rho_j \end{cases} \tag{3.44}$$

relations dans lesquelles :

- $\underline{X}_c = (X_c, Y_c)^T$;
- $\underline{\Theta} = (\theta_1, \ldots, \theta_m, \rho_1, \ldots, \rho_m)^T$;
- $\underline{\mathcal{M}}_i = (m_{i,1}, \ldots, m_{i,m})^T$;
- φ_i et ψ_i sont des fonctions des variables $\underline{X}_c$, $\underline{\Theta}$, $\underline{\mathcal{M}}_i$ dont les formes ne sont pas très sympathiques et qu'il n'est pas utile de présenter ici. Pour simplifier les notations ultérieures, nous noterons $\varphi_i = \varphi_i(\underline{X}_c, \underline{\Theta}, \underline{\mathcal{M}}_i)$ et $\psi_i = \psi_i(\underline{X}_c, \underline{\Theta}, \underline{\mathcal{M}}_i)$;

- $\gamma_j = \sin\theta_j + \alpha_j \cos\theta_j$.

Les relations (3.44) font exhiber les termes $\sum_j m_{i,j}$, $\sum_j m_{i,j}\alpha_j$, $\sum_j m_{i,j}\beta_j\rho_j$ et $\sum_j m_{i,j}\gamma_j\rho_j$. On peut vérifer, numériquement ou à l'aide d'un calculateur symbolique, que ces termes se simplifient, suivant les valeurs de i, et quelques soient les couples (ρ_j, θ_j) en :

$$\left\{\begin{array}{lcl} \sum_j m_{i,j} & = & 1, \underbrace{0, \ldots, 0}_{m-1 fois} \\ \sum_j m_{i,j}\alpha_j & = & 0, -1, \underbrace{0, \ldots, 0}_{m-2 fois} \\ \sum_j m_{i,j}\beta_j\rho_j & = & 0, 0, a_0, a_1, \ldots, a_h, b_1, \ldots, b_h \\ \sum_j m_{i,j}\gamma_j\rho_j & = & 0, 0, 0, b_1, \ldots, kb_k, \ldots, hb_h, -a_1, \ldots, -ka_k, \ldots, -ha_h \end{array}\right. \tag{3.45}$$

où $1 \leq k \leq h$.

En supposant par ailleurs toutes les profondeurs z_j égales à une valeur $\hat{z}^*$, approximation de la profondeur des z_j une fois la tâche réalisée, on en déduit une expression simplifiée de la matrice d'interaction recherchée :

$$\widehat{\underline{L_Q^T}}\Big|_{\hat{z}^*} = \begin{pmatrix} -\frac{1}{\hat{z}^*} & 0 & \frac{X_c}{\hat{z}^*} & X_cY_c + \varphi_1 & -1 - X_c^2 + \psi_1 & Y_c \\ 0 & -\frac{1}{\hat{z}^*} & \frac{Y_c}{\hat{z}^*} & 1 + Y_c^2 + \varphi_2 & -X_cY_c + \psi_2 & -X_c \\ 0 & 0 & \frac{a_0}{\hat{z}^*} & \varphi_3 & \psi_3 & 0 \\ 0 & 0 & \frac{a_1}{\hat{z}^*} & \varphi_4 & \psi_4 & b_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \frac{a_k}{\hat{z}^*} & \varphi_{k+3} & \psi_{k+3} & kb_k \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \frac{a_h}{\hat{z}^*} & \varphi_{h+3} & \psi_{h+3} & hb_h \\ 0 & 0 & \frac{b_1}{\hat{z}^*} & \varphi_{h+4} & \psi_{h+4} & -a_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \frac{b_k}{\hat{z}^*} & \varphi_{k+h+3} & \psi_{k+h+3} & -ka_k \\ 0 & 0 & \frac{b_h}{\hat{z}^*} & \varphi_{2h+3} & \psi_{2h+3} & -ha_h \end{pmatrix} \tag{3.46}$$

Cette expression nous permet de mettre en évidence plusieurs points importants :

- On s'aperçoit à la vue des première et deuxième lignes de $\widehat{L_{\underline{Q}}^T}\Big|_{\hat{z^*}}$ que la matrice d'interaction relative à l'information visuelle (X_c, Y_c) diffère de celle de l'information visuelle (X_c, Y_c), quand elle est considérée en tant que la projection d'un point, uniquement part l'ajout des fonctions φ_1, ψ_1, φ_2 et ψ_2, c'est-à-dire, sur les termes relatifs aux rotations d'axes x et y.

- Par ailleurs, comme on pouvait le pressentir intuitivement, les informations visuelles représentées par $\underline{P}$ ne sont pas affectées par des translations suivant x ou y ou, en d'autres termes la forme du contour n'est pas modifiée par ces translations. En outre, ces deux translations sont découplées et, par conséquent, un mouvement suivant l'axe x (respectivement y) ne modifie que la coordonnée suivant x (respectivement y) du centre de gravité.

- On pourrait s'étonner que les termes de $\widehat{L_{\underline{Q}}^T}\Big|_{\hat{z^*}}$ suivant les rotations ω_x et ω_y dépendent, par l'intermédiaire des fonctions φ et ψ, des valeurs choisies pour les couples (ρ_j, θ_j) et pas les autres termes. Ceci est en fait une conséquence directe de l'approximation du contour réel par la signature polaire qui conduit à la non vérification de la contrainte de rigidité. En effet, le contour supposé décrit formellement par (3.31) et (3.34) n'est ni modifié par les translations, ni par rotation d'axe z. Par contre, il l'est en cas de rotation suivant les axes x ou y, puisqu'il ne correspond pas formellement à la projection de l'objet sur le plan image.

  Toutefois, en choisissant une découpage homogène des θ_j, des simulations et des expérimentations confirmeront que cette matrice $\widehat{L_{\underline{Q}}^T}\Big|_{\hat{z^*}}$ est une bonne approximation de $L_{\underline{Q}}^T\Big|_{\hat{z^*}}$ qui serait obtenue à partir d'une approche basée sur des informations tridimensionnelles de l'objet comme il est fait habituellement. En outre, compte tenue de la condition de stabilité de la loi de commande rappelée en 3.1 :

$$\widehat{L_{\underline{Q}}^T}^{+} L_{\underline{Q}}^T > 0 \tag{3.47}$$

  une approximation peut convenir.

La matrice d'interaction étant obtenue, nous allons à présent nous intéresser au processus d'obtention des informations visuelles.

3.5.2 Extraction des informations visuelles

Le problème posé ici concerne l'extraction des informations visuelles, c'est-à-dire l'obtention du vecteur défini par :

$$\underline{s} = (X_c, Y_c, a_0, \cdots, a_h, b_1, \cdots, b_h)^T \tag{3.48}$$

(X_c, Y_c) représente les coordonnées du centre de gravité de l'image de l'objet. Elles sont calculées directement par la plus part des produits rencontrés en traitement d'image comme les cartes de traitement ou les logiciels.

Par contre, les a_k et b_k restent à calculer. Il s'agit, plus précisément, d'identifier la signature polaire réelle du contour à son modèle décrit par (3.34), soit l'extraction de $2h+1$ paramètres. Par conséquent, $2h+1$ points extraits du contour considéré sont théoriquement suffisants. Toutefois, ces points seront nécessairement bruités, de plus, on peut légitimement supposer que le contour est constitué d'un nombre N de points tel que $N \gg 2h+1$. Une méthode du type des moindres carrés semblent indiquée.

Pour ce faire, posons $\alpha_{1,i} = 1$ et pour $k = 2 \cdots h$:

$$\begin{cases} \alpha_{k,i} & = & \cos(k-1)\theta_i \\ \alpha_{k+h,i} & = & \sin(k-1)\theta_i \end{cases}$$

pour l'ensemble des points du contour, soit pour $i = 1 \cdots N$.

La signature polaire réellement mesurée s'écrit donc en θ_i :

$$\rho_r(\theta_i) = \left(\sum_{k=1}^{k=2h+1} \alpha_{k,i} P_k \right) + b_i \tag{3.49}$$

où b_i est le bruit de modélisation.

Le problème consiste alors, à partir des N mesures $(\rho_r(\theta_i), \alpha_{1,i}, \cdots, \alpha_{2h+1,i})$, à trouver le meilleur estimé du vecteur $\underline{\hat{P}}$ au sens du critère J des moindres carrés :

$$J = \sum_{i=1}^{i=N} \left(\rho_r(\theta_i) - \underline{\alpha}_i^T \underline{\hat{P}} \right)^2 \tag{3.50}$$

dans lequel $\underline{\alpha}_i^T = (\alpha_{1,i}, \cdots, \alpha_{2h+1,i})$. La solution s'obtient en écrivant que le gradient de J par rapport à $\underline{\hat{P}}$ est nul :

$$\left(\sum_{i=1}^{i=N} \underline{\alpha}_i \underline{\alpha}_i^T \right) \underline{\hat{P}} - \sum_{i=1}^{i=N} \underline{\alpha}_i \rho_r(\theta_i) = 0 \tag{3.51}$$

si les observations sont indépendantes, la solution est donnée par :

$$\underline{\hat{P}} = \left(\sum_{i=1}^{i=N} \underline{\alpha}_i \underline{\alpha}_i^T \right)^{-1} \sum_{i=1}^{i=N} \underline{\alpha}_i \rho_r(\theta_i) \tag{3.52}$$

Dans le cas où le bruit b est de moyenne nulle, on peut montrer que cette estimation est non biaisée (voir par exemple [Labarrere 88]).

Remarque : on peut à présent souligner l'intérêt d'une approche basée sur des informations visuelles issues du contour de l'objet plutôt que sur l'extraction éventuelle de points caractéristiques (en supposant que l'objet observé en contienne). En effet, le processus d'extraction d'un point caractéristique, au même titre qu'un point du contour, fournit une information bruitée qui, dans le cas d'un asservissement visuel basé sur des primitives points est à utiliser telle quelle, alors que dans le cas d'une approche contour la méthode d'identification du modèle de ce contour par les moindres carrés permet, comme nous allons le voir par la suite, de limiter fortement son influence. Nous résumerons les avantages de cette nouvelle approche dans la conclusion de ce chapitre, une fois connu le comportement de la loi de commande associée.

La matrice d'interaction étant connue et les informations visuelles étant extraites, nous allons tester dans le paragraphe suivant le comportement de la loi de commande conformément à l'approche fonction de tâche décrite en 3.1 :

$$T_c = -\lambda \left(L_{\underline{s}}^T \big|_{z=\hat{z}^*, \underline{s}=\underline{s}^*} \right)^+ (\underline{s} - \underline{s}^*) \tag{3.53}$$

3.5.3 Résultats de simulations

Au cours des simulations présentées ci-dessous, nous réalisons tout d'abord une tâche de positionnement consistant à placer la caméra en face d'un objet plan. Puis, une autre tâche, relative à une orientation différente, mais par rapport au même objet est ensuite réalisée. Ces deux tâches sont alors une nouvelle fois examinées, mais dans le cas d'un objet non plan. Le comportement de l'algorithme est ensuite décrit en environnement bruité pour ces deux tâches et pour ces deux types d'objets. Pour terminer ces simulations, nous discutons de l'influence sur la loi de commande, de la précision de l'estimation de la profondeur une fois la tâche réalisée et du nombre d'harmoniques, étant entendu qu'ils consituent les deux paramètres les plus importants intervenant dans son calcul, l'influence de λ étant connue.

3.5.3.1 Simulations non bruitées

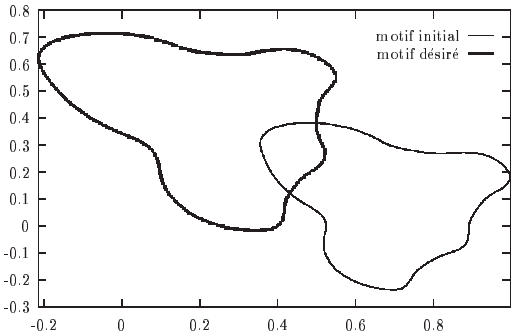
Cas d'un objet plan

Nous considérons tout d'abord le cas d'un objet plan et supposons qu'à tout instant le contour de l'objet reste entièrement visible. La tâche consiste à positionner la caméra en face de l'objet observé.

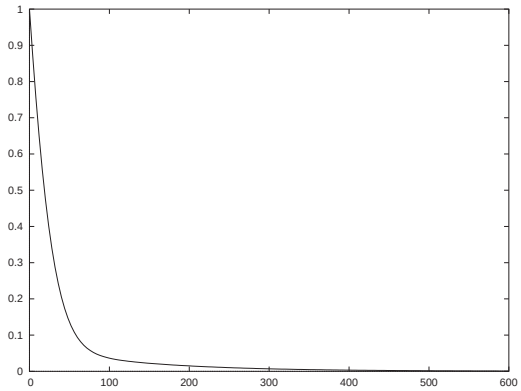
La figure 3.4a représente le motif initial et le motif désiré. Les figures 3.4b et 3.4c représentent respectivement l'erreur normalisée relative aux informations visuelles [1] et les composantes du torseur cinématique. Les conditions de simulations étant les suivantes :

- La période d'acquisition des images est supposée réalisée à la cadence vidéo, soit $\Delta t = 40$ ms ;
- Le nombre h d'harmoniques a été choisi égal à 4 ;
- Le nombre N de points du contour a été choisi égal à 300 ;
- Le gain λ, intervenant dans la loi de commande, a été fixé à 1 ;
- La profondeur, une fois la tâche réalisée, vaut $z^* = 50$ cm ;
- L'approximation de z^* a été fixé à $\hat{z}^* = 65$ cm ;
- La matrice d'interaction a été maintenue constante et a été calculée en $\underline{s} = \underline{s}^*$ et pour $z = \hat{z^*}$;
- La position initiale de la caméra par rapport à l'objet a été obtenue à partir d'un positionnement en face de l'objet à une distance de z^* puis, des rotations successives d'axe x, y et z d'angles θ_x, θ_y et θ_z puis des translations de valeurs Tr_x, Tr_y et Tr_z. Les valeurs suivantes ont été utilisées :
  - $\theta_x = 10$ degrés
  - $\theta_y =$ -10 degrés
  - $\theta_z = 10$ degrés
  - $Tr_x =$ -20 cm
  - $Tr_y = 20$ cm
  - $Tr_z =$ -20 cm

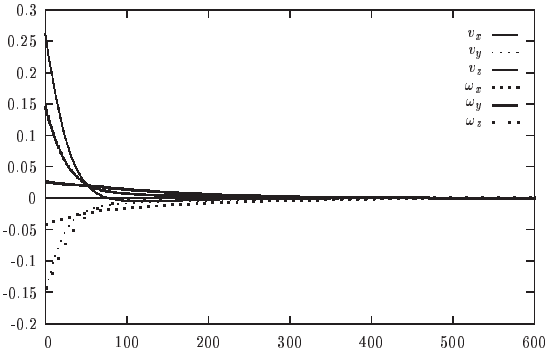
[1] définie comme étant $\frac{\|\underline{s}-\underline{s}^*\|_k}{\|\underline{s}-\underline{s}^*\|_0}$



(a) motif initial et motif désiré



(b) Erreur normalisée



(c) Composantes du torseur cinématique (m/s ou rad/s)

FIG. 3.4 – Positionnement en face d'un objet plan

- L'algorithme se termine quand l'erreur normalisée devient inférieure à 10^{-4}.

Les résultats obtenus sont tout à fait satisfaisants. De plus, ils nous permettent de vérifier que la manière de calculer la matrice d'interaction conduit à une bonne approximation de la façon réelle dont varient les informations visuelles par rapport aux mouvements du porteur.

Une autre tâche de positionnement a été également testée, elle a consisté en un positionnement souhaité à 30 degrés par rapport à l'axe x de la position finale précédente. La figure 3.5a représente le motif initial et le motif désiré. Les figures 3.5b et 3.5c représentent respectivement l'erreur normalisée relative aux informations visuelles et les composantes du torseur cinématique, les conditions de simulations étant inchangées. Une nouvelle fois, la convergence est atteinte bien que toutes les profondeurs utilisées soient grossièrement approximées. En fait, dans ce cas de figure, pratiquement aucune information sur l'objet ou sur sa localisation par rapport à la caméra n'est requise exceptées les images courante et désirée.

Cas d'un objet non plan

Nous considérons maintenant le cas d'un objet non plan et supposons toujours qu'à tout instant le contour de l'objet est entièrement visible. La figure 3.6 représente l'objet utilisé dans un repère absolu, c'est en fait l'objet précédent projeté sur un cylindre, il s'en suit une variation de profondeur d'environ 15 cm entre les différents points de l'objet.

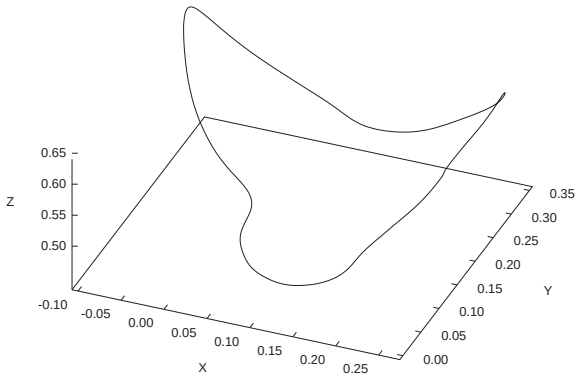
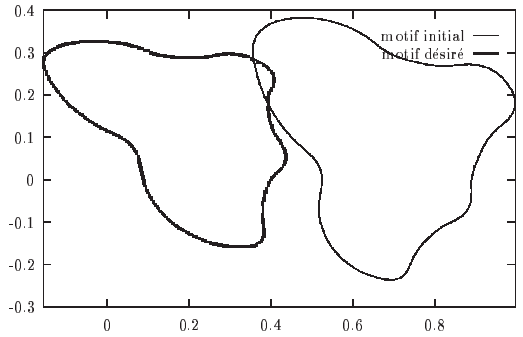
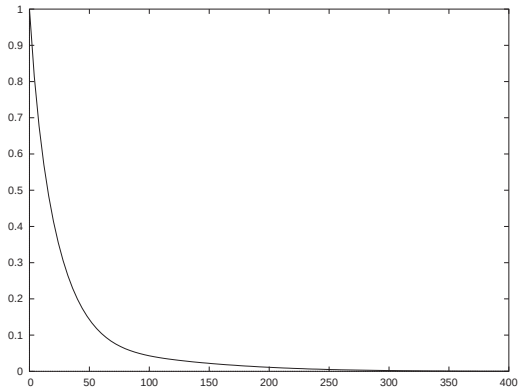


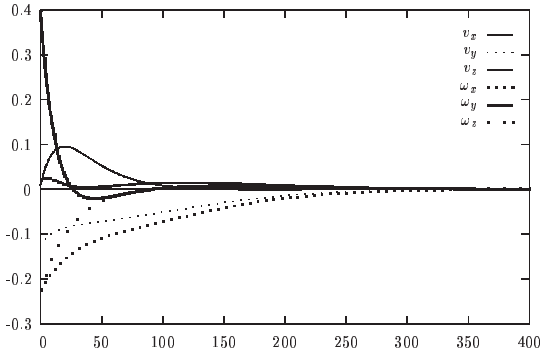
FIG. 3.6 – Objet utilisé



(a) motif initial et motif désiré



(b) Erreur normalisée



(c) Composantes du torseur cinématique (m/s ou rad/s)

FIG. 3.5 – Positionnement à 30 degrés d'un objet plan

Le motif initial et le motif désiré sont représentés sur la figure 3.7a. Les figures 3.7b et 3.7c rapportent respectivement les comportements de l'erreur normalisée et des composantes du torseur cinématique. Les conditions de simulations sont identiques à celles du paragraphe précédent exceptée la valeur de $\hat{z}^*$ qui a été portée à la profondeur moyenne de l'objet (soit 57 cm) et le nombre d'harmoniques qui est passé à 8. La convergence est obtenue sans problème particulier.

Par ailleurs, comme dans le cas d'un objet plan, un positionnement a été réalisé à 30 degrés de la position finale précédente. La figure 3.8a représente le motif initial et le motif désiré, les figures 3.8b et 3.8c l'erreur normalisée et les composantes du torseur cinématique. Les conditions de simulations étant inchangées excepté $\hat{z}^*$ qui a été porté à 70 cm pour tenir compte de la non planéité de l'objet[2] et le nombre d'harmoniques qui est passé à 8. La convergence est une nouvelle fois obtenue sans problème malgré les approximations faites sur les profondeurs.

D'une façon générale, ont peut remarquer que la réalisation d'une tâche de positionnement par rapport à un objet non plan nécessite un nombre d'harmoniques plus élevé que pour un objet plan.

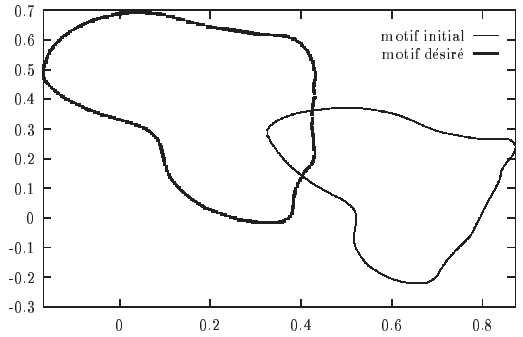
3.5.3.2 Simulations bruitées

Cas d'un objet plan

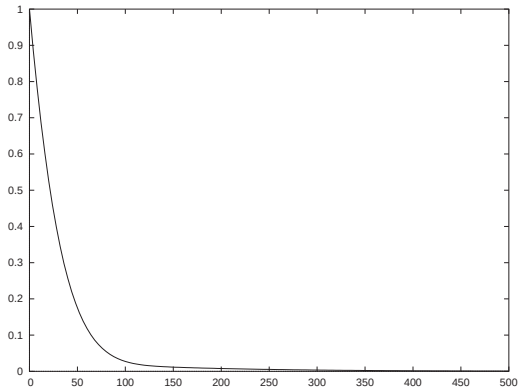
Nous considérons, dans ce paragraphe, le cas où à chaque point $(X, Y) \in (\mathcal{C})$ un bruit uniforme d'amplitude maximum $\pm$ 2 pixels a été ajouté. Les figures 3.9a et 3.9b représentent respectivement le comportement du signal d'erreur normalisé et des composantes du torseur cinématique. Les conditions de simulations étant inchangées par rapport à 3.5.3.1. L'introduction de bruit ne modifie pas le comportement de la loi de commande. Cela se conçoit assez bien dans la mesure où les informations visuelles sont obtenues à l'aide d'une identification par la méthode des moindres carrés qui en fournit une estimation non biaisée, comme nous l'avons vu en 3.5.2.

Remarque : on pourrait être tenté dans (3.41) d'exprimer $L^T_{X_i}$ et $L^T_{Y_i}$ directement à partir des points du contour mesurés dans l'image. En fait, il est préférable d'utiliser les expressions de X_i et Y_i en fonction des informations visuelles grâce à (3.31) et (3.34). On dispose ainsi d'une loi de commande peu sensible au bruit de mesure.

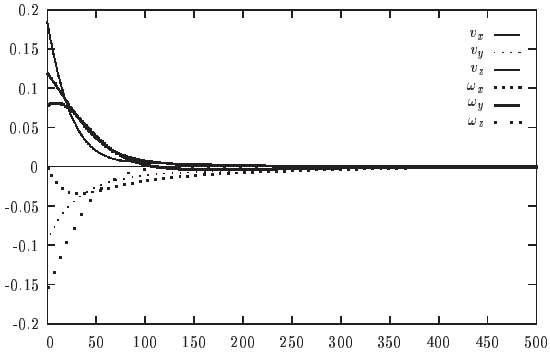
[2]C'est approximativement la profondeur moyenne (z_{moy} = 67 cm)



(a) motif initial et motif désiré

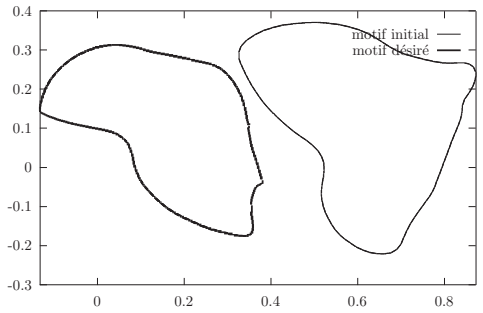


(b) Erreur normalisée

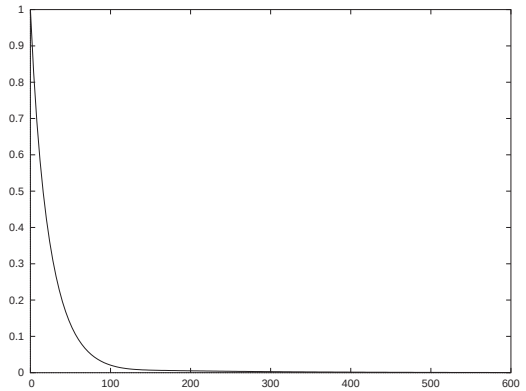


(c) Composantes du torseur cinématique (m/s ou rad/s)

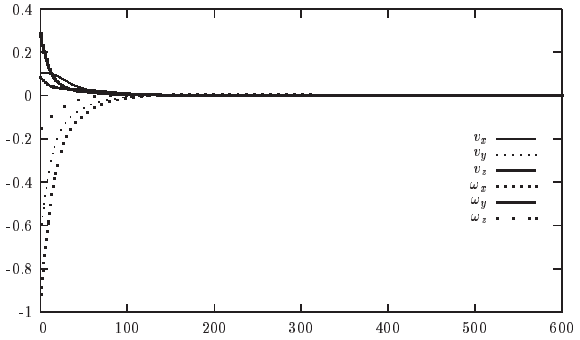
FIG. 3.7 – Positionnement en face d'un objet non plan



(a) motif initial et motif désiré

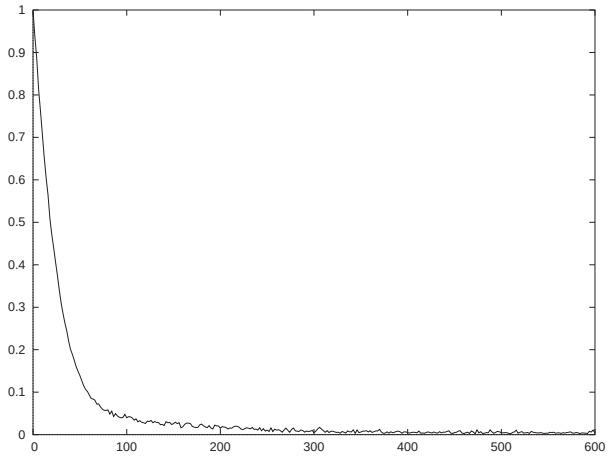


(b) Erreur normalisée

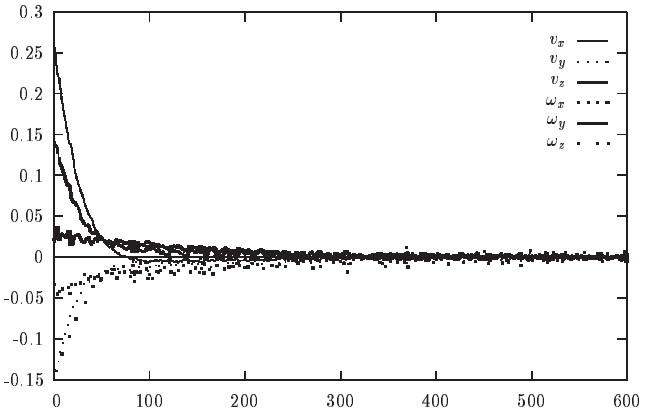


(c) Composantes du torseur cinématique (m/s ou rad/s)

FIG. 3.8 – Positionnement à 30 degrés d'un objet non plan



(a) Erreur normalisée



(b) Composantes du torseur cinématique (m/s ou rad/s)

FIG. 3.9 – Positionnement en face d'un objet plan — Informations visuelles bruitées —

Cas d'un objet non plan

La figure 3.10 rapporte le comportement respectif de l'erreur normalisée et des composantes du torseur cinématique dans le cas le plus délicat où l'objet est non plan et où la situation désirée est orientée de 30 degrés par rapport à un positionnement en face de l'objet. Une nouvelle fois, on observe de bons résultats.

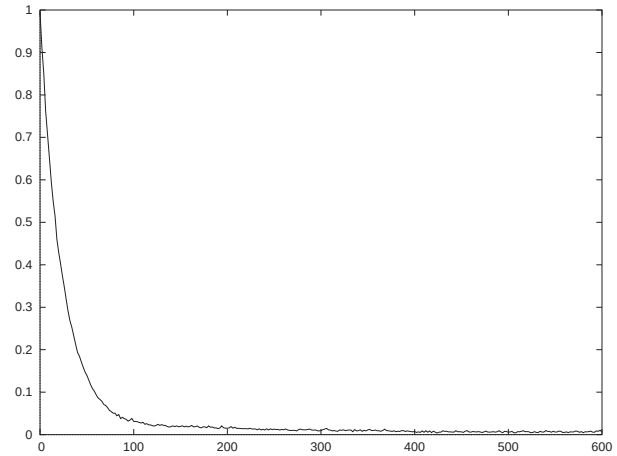
Le comportement de l'algorithme étant connu dans diverses configurations particulières et pour différents types d'objets, nous allons discuter maintenant de l'influence de $\hat{z}^*$ puis de h sur la loi de commande.

3.5.3.3 Influence de la précision de l'estimation de la profondeur une fois la tâche réalisée sur le comportement de la loi de commande

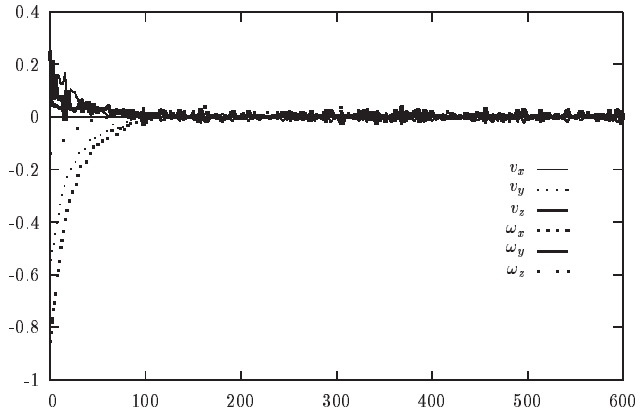
Nous considérons, tout d'abord, la réalisation de la tâche consistant à amener la caméra en face de l'objet non plan représenté sur la figure 3.6. Les conditions de simulation sont identiques à celles fixées pour l'objet plan exceptée la valeur de h qui a été portée à 8.

Nous avons pu vérifier que la valeur de $\hat{z}^*$ n'est pas critique. En effet, une convergence peut tout aussi bien être obtenue pour $\hat{z}^* \in [10 \text{ cm}, 250 \text{ cm}]$. Toutefois, $\hat{z}^*$ influe sur la vitesse de convergence comme en témoigne la figure 3.11a, une faible valeur de $\hat{z}^*$ conduit à une convergence lente alors qu'une forte valeur de $\hat{z}^*$ conduit, au contraire, à une convergence rapide. Remarquons sur la figure 3.11b que, dans ce dernier cas, le chemin parcouru par la caméra n'est pas du tout optimisé par rapport aux cas où les valeurs de $\hat{z}^*$ sont plus faibles. Il s'en suit, comme le montre clairement la figure 3.12, des valeurs de commande beaucoup plus élevées (puisque le chemin parcouru est plus long et la convergence plus rapide). Ce phénomène s'explique aisément en observant qu'il apparaît dans la matrice d'interaction (3.46) des termes en $1/\hat{z}^*$ qui conduisent à des termes en $\hat{z}^*$ dans le calcul de la pseudo-inverse nécessité par (3.53).

Nous avons ensuite considéré la tâche consistant à positionner la caméra à 30 degrés de la position finale précédente. Ici aussi la valeur de $\hat{z}^*$ n'est pas critique, un choix de $\hat{z}^*$ dans l'intervalle [30 cm, 250 cm] conduit à une convergence. De plus, son influence concernant le chemin parcouru par la caméra et les valeurs des composantes du torseur cinématique est identique. Pour terminer ces simulations concernant le rôle joué par $\hat{z}^*$, nous avons considéré ces deux tâches de positionnement mais pour l'objet plan, là encore des résultats similaires ont été obtenus.

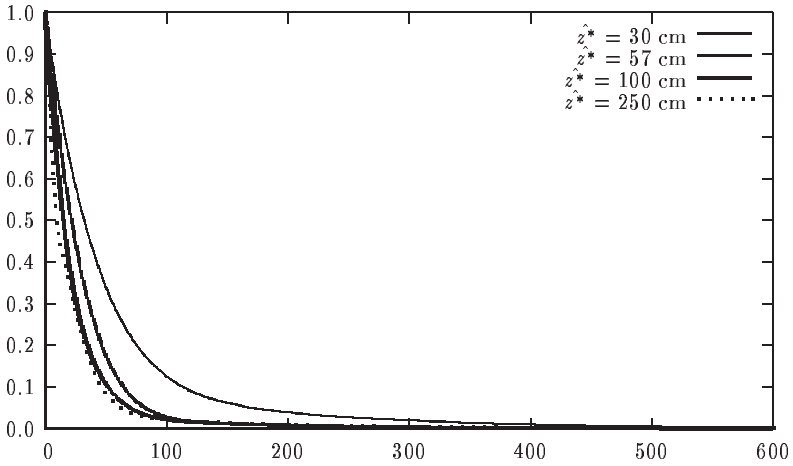


(a) Erreur normalisée

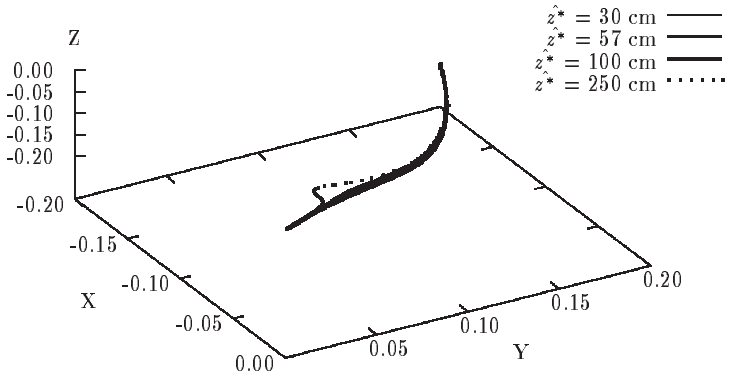


(b) Composantes du torseur cinématique (m/s ou rad/s)

FIG. 3.10 – Positionnement à 30 degrés d'un objet non plan — Informations visuelles bruitées —

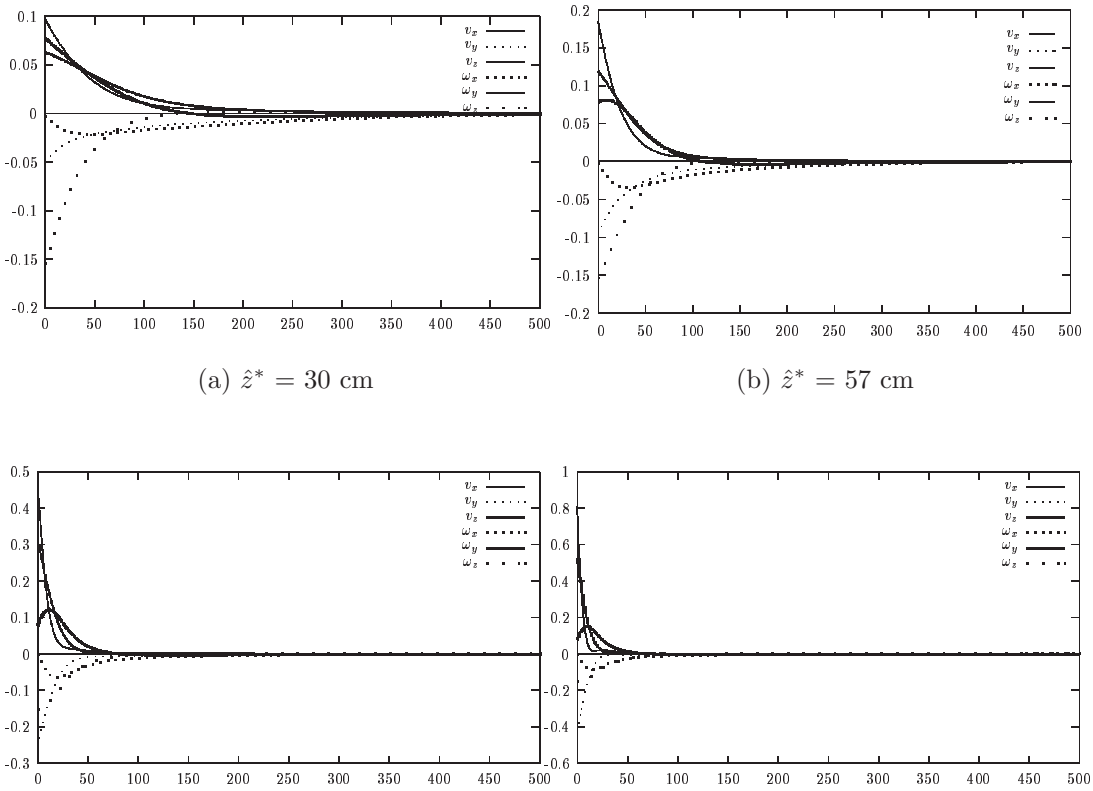


(a) Erreur normalisée



(b) Trajectoire du centre optique de la caméra dans le repère de l'objet

FIG. 3.11 – Positionnement en face d'un objet non plan — Influence de $\hat{z}^*$



(a) $\hat{z}^* = 30$ cm

(b) $\hat{z}^* = 57$ cm

(c) $\hat{z}^* = 100$ cm

(d) $\hat{z}^* = 250$ cm

FIG. 3.12 – Positionnement en face d'un objet non plan — Influence de $\hat{z}^*$ sur les composantes du torseur cinématique (m/s ou rad/s)

Par conséquent, ces résultats mettent en valeur la robustesse de la loi de commande vis-à-vis d'une imprécision de l'estimation de la profondeur une fois la tâche réalisée.

3.5.3.4 Influence du nombre d'harmoniques sur le comportement de la loi de commande

La première contrainte, certainement la plus importante, à laquelle doit se soumettre h est de réaliser bien évidemment la tâche de positionnement. C'est-à-dire d'assurer un rang plein pour la matrice $\widehat{L^T}_{|\underline{s}=\underline{s}^*}$. Il vient donc en particulier la condition nécessaire $2h + 3 \geq 6$. Cette condition étant respectée, seules d'éventuelles singularités isolées peuvent apparaître, mais indépendamment de h. La plus petite valeur possible pour h est donc, théoriquement, $h = 2$.

Par ailleurs, d'un tout autre point de vue, les informations visuelles s_i interviennent dans l'algorithme de deux façons :

- dans la commande : la manière dont elles varient est un point primordial pour la loi de commande. En effet, elles doivent satisfaire :

$$\underline{s}|_{k+1} = \underline{s}|_k + \Delta t L^T_{|\underline{s}} T_c \tag{3.54}$$

- dans la modélisation du contour dans l'image : les informations visuelles sont ajustées de manière à minimiser la distance entre la signature polaire réelle et celle modélisée.

Ces deux points peuvent se transcrire de la façon suivante :

$$\underline{s}|_{k+1} = \underline{s}|_k + \Delta t L^T_{|\underline{s}} T_c + \delta \underline{s}|_{k+1} \tag{3.55}$$

où $\delta \underline{s}|_{k+1}$ est un terme de variation introduit pour passer de la modélisation de la signature polaire de l'instant k à l'instant $k + 1$.

En somme, pour que la loi de commande converge, il faut que h soit suffisamment important pour que (3.54) reflète, au moins grossièrement mais à chaque itération, la variation de $\underline{s}$ quand T_c est appliqué au porteur. À titre d'exemple, considérons le cas de l'objet non plan et de la tâche consistant à positionner le capteur en face de l'objet. Nous avons pu vérifier qu'un nombre trop faible d'harmoniques pouvait conduire à une convergence très lente de la loi de commande, voire uniquement pour des positions initiales proches de la position finale

souhaitée, ou même à une divergence. Par exemple, dans ce cas de figure, un choix $h = 4$ conduit à une divergence, un choix $h = 6$ conduit à une convergence très lente (le critère d'arrêt est atteint pour 5000 itérations au lieu de 500) alors qu'un choix $h = 8$ assure une convergence pour différentes configurations initiales caméra / objet. D'autre part, nous avons remarqué que ces valeurs de h sont très liées à la complexité de la forme de l'objet dans le sens où, dans le cas d'un objet beaucoup plus simple, une convergence avec $h = 4$ est possible. Un exemple d'un tel objet est représenté sur la figure 3.13.

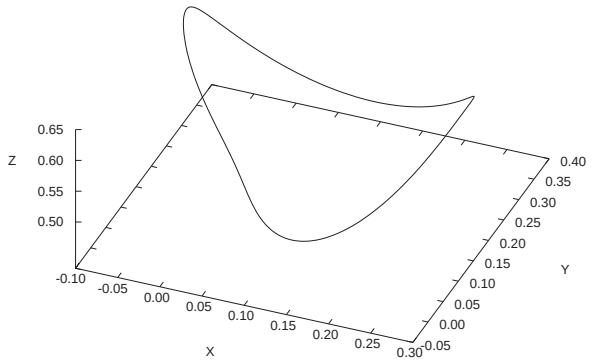
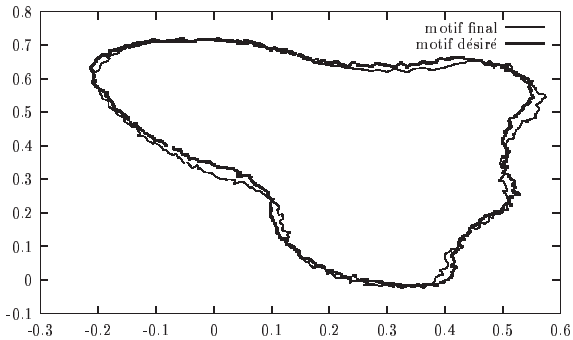
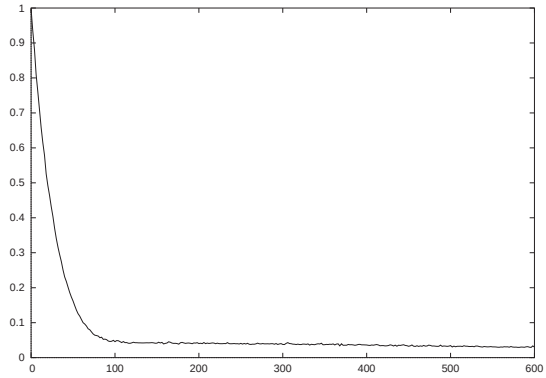


FIG. 3.13 – Objet de forme simple permettant une convergence pour $h = 4$

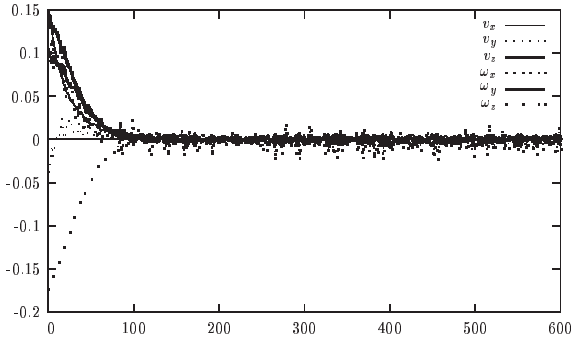
On pourrait penser à imposer une valeur de h importante garantissant ainsi une modélisation, à chaque itération, parfaite et donc un terme $\delta\underline{s}|_{k+1}$ quasiment nul. En pratique, cette façon de faire conduirait à des temps de calcul incompatibles avec un fonctionnement à la cadence vidéo. En outre, si h est choisi grand, il faut alors augmenter le nombre N de points du contour pour conserver une bonne précision sur la valeurs des informations visuelles. Ce nombre de points étant en fait limité physiquement par la quantification introduite par le capteur par l'intermédiaire des pixels, on peut de ce fait, en augmentant h vis-à-vis de N, assister à l'apparition d'un phénomène très gênant, l'existence d'un minimum local. La figure 3.14 rapporte un tel cas de figure. Les images finale et désirée, représentées sur la figure 3.14a, sont proches les unes des autres, comme en témoigne l'erreur normalisée représentée sur la figure 3.14b. Par contre, la matrice exprimant l'orientation de la caméra par rapport au repère de l'objet n'est pas égale à la matrice unité. En effet, le calcul des angles d'Euler γ, β et α représentant respectivement les rotations autour des axes x, y et z ont pour valeurs respectives : $\gamma =$



(a) Motif final et motif désiré



(b) Erreur normalisée



(c) Composantes du torseur cinématique (m/s ou rad/s)

FIG. 3.14 – Positionnement en face d'un objet plan — Informations visuelles bruitées — Cas d'un minimum local

15.95 degrés, $\beta = 3.9$ degrés et $\alpha = 1.78$ degrés. De plus les translations d'axes x, y et z suivantes sont rapportées : -5.1 cm, 16.7 cm, 46.5 cm au lieu de 0 cm, 0 cm et 50 cm. Pour cette simulation, concernant des informations visuelles bruitées, la valeur de h a été portée à 15, N valant toujours 300. En outre, on a pu vérifier que si N est augmenté à la valeur 500 par exemple, ce minimum local disparaît. On retrouve là le problème classique du choix du nombre de points nécessaires à une bonne estimation de paramètres en présence de bruit. De même, on aurait certainement pu montrer que ces minima locaux apparaissent pour de plus faibles valeurs de h dès lors que le bruit augmente (pour N constant).

Après avoir précisé, en simulation, l'influence des deux paramètres principaux intervenant dans la loi de commande, à savoir h et $\hat{z}^*$, et constaté le bon comportement de l'algorithme proposé, nous allons à présent le valider sur le site expérimental.

3.5.4 Résultats expérimentaux

Avant de présenter les résultats obtenus, nous allons tout d'abord décrire, brièvement la plate-forme expérimentale, puis la méthode utilisée pour valider ces résultats. Les tests expérimentaux portent sur la validation des résultats obtenus en simulation mais aussi sur l'influence de la calibration du dispositif expérimental ou d'une occlusion partielle de l'objet.

3.5.4.1 Description de la plate-forme expérimentale

Dans le cadre des travaux rapportés dans ce document, une plate-forme expérimentale a été conçue et réalisée sur le site de Rennes du *Cemagref*. Nous allons la décrire brièvement.

Elle est constituée d'un robot cartésien, nécessaire aux trois degrés de liberté assurant les translations, auquel a été ajoutée une rotule (trois axes orthogonaux et concourants) assurant la réalisation des rotations (Figure 3.15).

Sur la rotule est fixée une caméra CCD Sony XC-8500CE. Elle possède la particularité d'exposer simultanément les trames paires et impaires et d'éviter ainsi le flou inter-trame. Elle dispose d'un capteur ayant la même résolution horizontale et verticale (pixels carrés de 8.3 μm x 8.3 μm) et permet l'acquisition d'images de taille 782(H) x 582(V) au format CCIR. Elle est dotée d'un objectif de focale 12.5 mm. Les acquisitions et traitements d'images sont réalisés par une carte GENESIS de la société Matrox. Elle est au format PC (bus PCI) et permet grâce à

FIG. 3.15 – Plate-forme expérimentale

un processeur TMS320C80 de réaliser des traitements à grande vitesse. De plus, elle possède une carte graphique qui permet une visualisation indépendante du superviseur. La programmation de la carte est réalisée de manière transparente grâce à des librairies C 32 bits dédiées à la carte.

Les asservissements articulaires sont réalisés par une carte d'axe DMC-1060 de la société GALIL. Elle est au format PC et permet l'asservissement des six axes du robot. Elle se programme en C.

Le superviseur est un PC équipé d'un Pentium à 200 MHz. Il permet de réaliser les calculs nécessaires à l'élaboration des lois de commande et de gérer les différentes cartes.

Par ailleurs, l'ensemble du dispositif a été calibré (jacobien du robot, caméra, matrice de passage de la caméra à la rotule).

3.5.4.2 Vérification des résultats

Lors des simulations, il était facile d'obtenir la situation de la caméra par rapport à l'objet. Par contre, expérimentalement, cette situation est beaucoup plus difficile à évaluer, en particulier l'orientation. Il s'agit en fait, plus précisément, de situer le capteur CCD par rapport à l'objet. La technique utilisée devra donc nécessairement faire intervenir la caméra. Deux approches sont possibles :

- mesurer à l'aide d'un dispositif à définir la situation de l'objet par rapport

à un repère fixe. Puis, grâce à des calibrations (repère fixe → porteur, porteur → poignet, poignet → caméra) obtenir cette situation par rapport à la caméra.

- utiliser une technique de localisation 3D. Pour ce faire une modélisation précise de l'objet ainsi qu'une caméra calibrée sont nécessaires comme on l'a vu en 2.1.1.

L'une et l'autre des deux méthodes vont conduire, du fait d'erreurs de mesure, à une situation biaisée de l'objet par rapport à la caméra. Mais la seconde méthode est moins lourde et fait intervenir moins d'intermédiaires susceptibles d'entraîner des erreurs (passages repère fixe → poignet et poignet → caméra). En conséquence, notre choix s'est porté sur la deuxième méthode.

Nous avons choisi la méthode de localisation décrite dans [Dementhon 95]. Pour ce faire un objet comportant quatre points non coplanaires est nécessaire, nous l'avons représenté sur la figure 3.16. Il est constitué de trois points coplanaires et d'un quatrième point distant de 9 cm du plan formé des trois autres. Cette méthode permet de calculer l'orientation et la position de la caméra par rapport à l'objet. On a caractérisé la matrice de rotation correspondante par les angles d'Euler notés γ, β et α, ils ont la même signification qu'en simulation. Les translations par rapport aux axes x, y et z sont notées respectivement t_x, t_y, t_z. De plus, nous avons appliqué cette méthode sur 100 images pour fournir un intervalle de validation.

Fig. 3.16 – Objet permettant la recontruction 3D

Par ailleurs, nous avons traduit en millimètre les variations des variables articulaires des axes assurant les translations entre les situations initiale et finale du porteur. Nous les notons Δx, Δy et Δz pour les translations suivant les axes respectifs. Ces valeurs permettent d'évaluer la distance séparant la position initiale du porteur de la situation obtenue une fois la tâche de positionnement réalisée.

Enfin, pour chaque expérimentation présentée, l'objet étant immobile, nous avons consigné la situation de la caméra par rapport à l'objet $(\gamma, \beta, \alpha, t_x, t_y, t_z)$ pour les situations désirée, initiale et réellement obtenue ainsi que les valeurs Δx, Δy et Δz. Pour ce faire, nous avons opéré de la manière suivante. Nous amenons le porteur à la situation désirée, par exemple par asservissement visuel sur un objet parfaitement connu, puis, nous remplaçons l'objet ayant servi au positionnement du porteur par l'objet utile à la méthode de Dementhon que nous appliquons. Nous remplaçons ensuite cet objet par l'objet par rapport auquel la tâche doit être réalisée. Cette façon de procéder est renouvelée à chaque fois qu'une localisation est nécessaire. Compte tenu du dispositif expérimental permettant de fixer les objets sur leur support, dispositif que l'on devine sur la figure 3.16, la précision sur la translation en x (obtenue par t_x) est de l'ordre d'un millimètre.

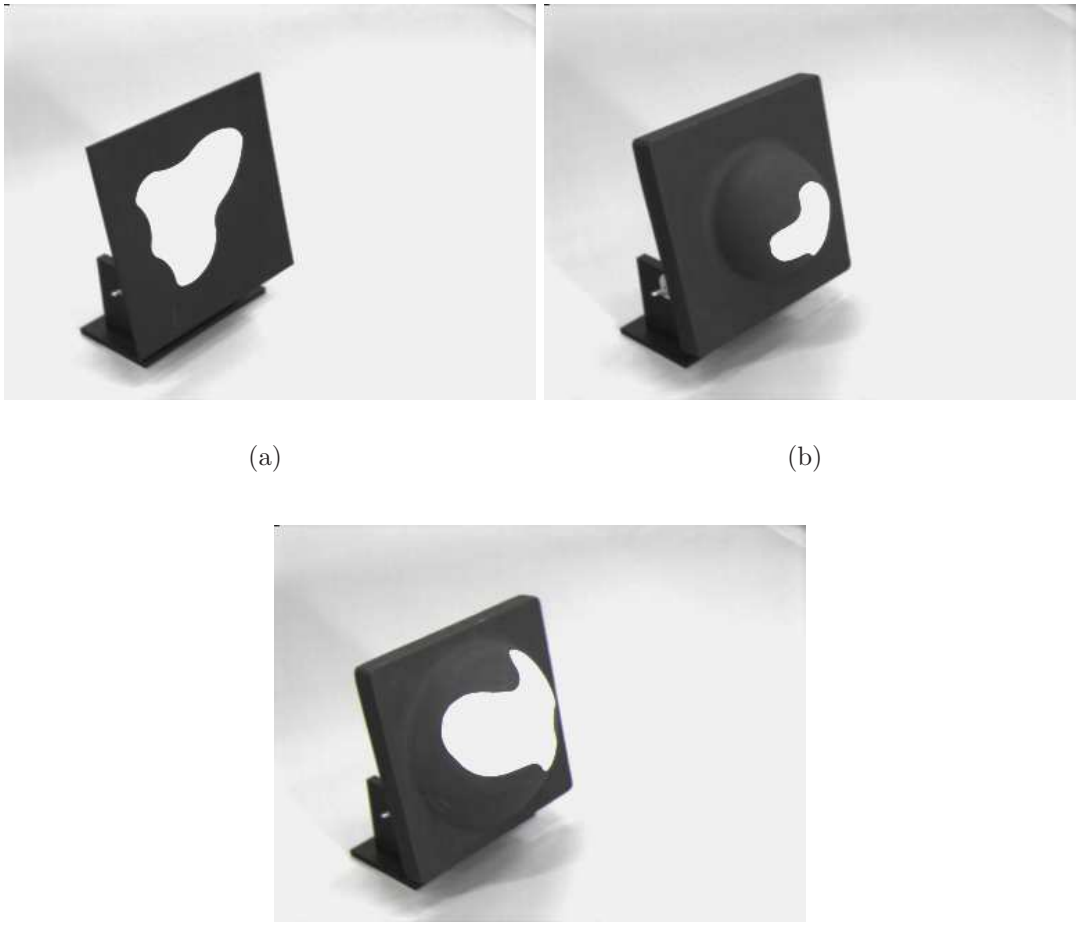
3.5.4.3 Résultats expérimentaux

Pour ces expérimentations, nous avons considéré trois types d'objet, nommés a, b et c :

- l'objet de type a est un objet plan de forme quelconque ;
- l'objet de type b est un objet non plan de profondeur maximum approximativement 2 cm ;
- l'objet de type c est un objet non plan de profondeur maximum approximativement 4 cm.

Ils sont représentés sur le figure 3.17. Comme on peut le voir, ils ont été choisis de telle sorte que leur contour dans l'image soient des fonctions holomorphes. De plus, ils sont binaires, pour s'affranchir d'éventuels problèmes liés au traitement des images.

Afin de valider les résultats obtenus en simulation, nous testons ci-dessous les mêmes cas de figure, puis, nous terminons par l'étude de l'influence de $\hat{z}^*$, de la calibration et des occlusions partielles sur la convergence de la loi de commande.



(a) (b)

(c)

Fig. 3.17 – Les objets utilisés lors des expérimentations

Cas d'un objet plan

La première expérimentation consiste en la réalisation d'une tâche assurant un positionnement du capteur en face de l'objet plan, l'objet a. Les conditions d'expérimentations et les résultats quantitatifs sont présentés ci-dessous (le choix de la valeur de ces paramètres sera justifié par la suite) :

- $h = 15 \Rightarrow \Delta t = 120$ ms
- $\lambda = 0.3$
- $z^* = 68$ cm
- Approximation de z^* : $\hat{z}^* = 65$ cm
- La matrice d'interaction a été maintenue constante et a été calculée en $\underline{s} = \underline{s}^*$. Toutes les profondeurs ont été supposées égales à $\hat{z}^*$.
- Situation initiale :
  - Angles d'Euler (en degrés) :

$$\gamma = 11.39 \pm 0.02 \quad \beta = 2.70 \pm 0.04 \quad \alpha = 9.36 \pm 0.06$$

  - Amplitude de translation par rapport à la situation désirée (en mm) :

$$\Delta_x = -283 \quad \Delta_y = 58 \quad \Delta_z = 288$$

- Situation désirée :
  - Angles d'Euler (en degrés) :

$$\gamma = -0.25 \pm 0.01 \quad \beta = -0.06 \pm 0.02 \quad \alpha = 0.02 \pm 0.02$$

  - Translations (en mm) :

$$t_x = 0.08 \pm 0.03 \quad t_y = -0.61 \pm 0.01 \quad t_z = 610.30 \pm 0.17$$

- Situation obtenue :
  - Angles d'Euler (en degrés) :

$$\gamma = -0.75 \pm 0.01 \quad \beta = 0.11 \pm 0.02 \quad \alpha = -0.04 \pm 0.02$$

– Translations (en mm) :

$$t_x = 0.33 \pm 0.05 \quad t_y = -1.75 \pm 0.01 \quad t_z = 611.00 \pm 0.15$$

De plus, la figure 3.18a représente le motif initial et le motif désiré, la figure 3.18b le motif obtenu. Les figures 3.18c et 3.18d, quant à elles, rapportent respectivement les comportements des composantes du torseur cinématique et de l'erreur normalisée relative aux informations visuelles.

Ces résultats sont tout à fait corrects et permettent ainsi de valider la façon dont a été obtenue la matrice d'interaction. Toutefois, on peut noter que le nombre d'harmoniques nécessaires à une bonne convergence est plus élevé qu'en simulation (il est passé de 4 à 15). Il conduit, de ce fait, à une cadence d'acquisition d'images de 120 ms et par conséquent à une valeur faible de λ. Cela explique les faibles vitesses atteintes par le porteur.

En fait, ce nombre important d'harmoniques est utilisé pour éviter l'apparition de minima locaux, phénomène non apparu dans ce cas de figure lors des simulations. En effet, si la valeur de h est trop faible, l'algorithme peut très bien converger comme le montre les figures 3.19b et 3.19c vers une erreur faible dans l'image, les images courante et désirée sont alors presque identiques comme le rapporte la figure 3.19c mais pour une position qui peut-être très éloignée de la position désirée. En effet, nous avons obtenu les valeurs suivantes :

- Angles d'Euler (en degrés) pour la situation désirée :

$$\gamma = 0.24 \pm 0.01 \quad \beta = -0.22 \pm 0.02 \quad \alpha = -0.08 \pm 0.02$$

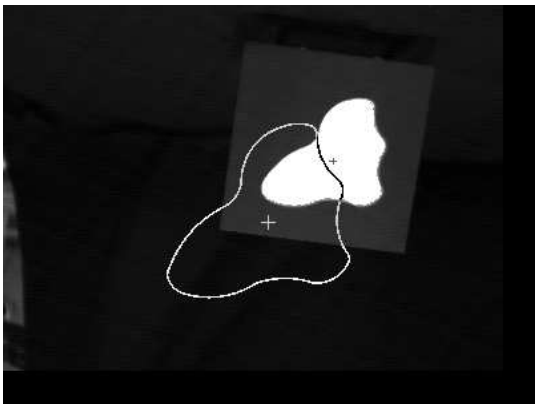
- Angles d'Euler (en degrés) pour la situation obtenue :

$$\gamma = 9.49 \pm 0.01 \quad \beta = -1.03 \pm 0.01 \quad \alpha = 1.27 \pm 0.02$$

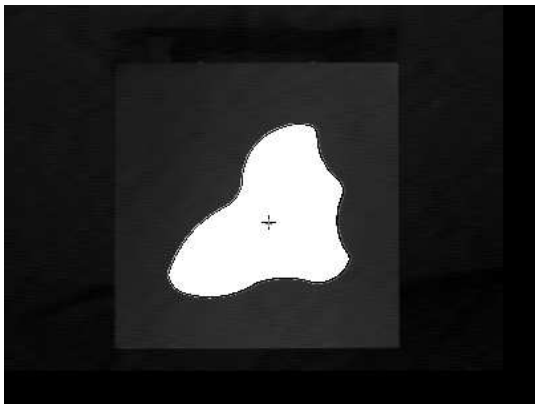
et d'autre part, concernant l'amplitude des translations entre les situations désirée et réellement obtenue (en mm) :

$$\Delta_x = 22 \quad \Delta_y = 115 \quad \Delta_z = -3$$

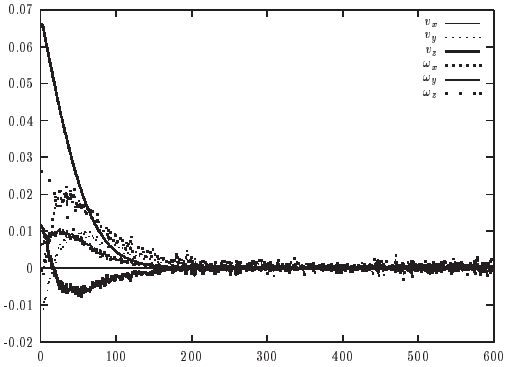
Cette expérimentation a été réalisée dans les mêmes conditions que précédemment, exceptée la valeur de h qui est passée à 20 et l'objet utilisé qui est l'objet de type c (on verra par la suite qu'une valeur de $h = 25$ conduit à une bonne réalisation de la tâche). En effet, ce phénomène, somme toute assez rare, peut être plus facilement mis en évidence sur un objet non plan.



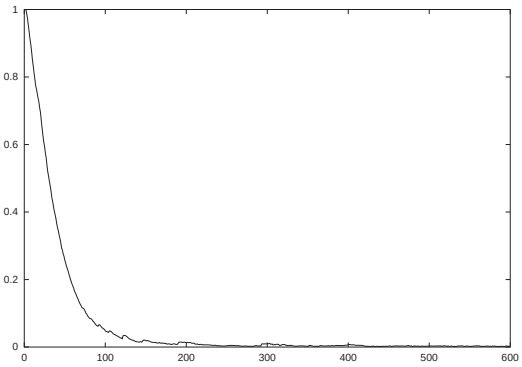
(a) Images actuelle et désirée



(b) Image finale

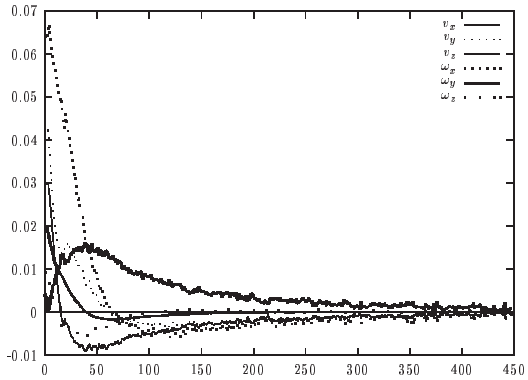


(c) Composantes du torseur cinématique (m/s ou rad/s)

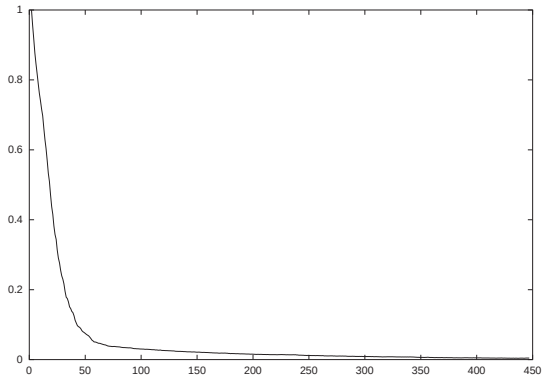


(d) Erreur normalisée

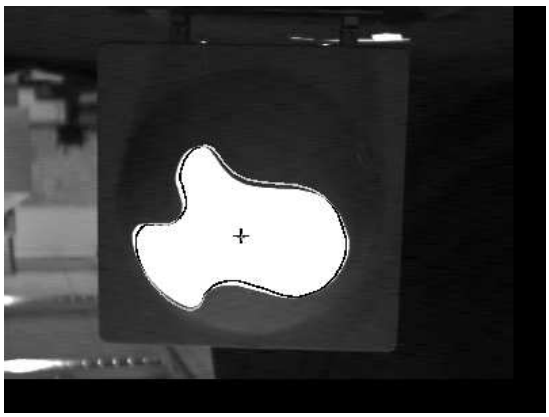
FIG. 3.18 – Résultats expérimentaux — Objet de type a — Positionnement en face de l'objet



(a) Composantes du torseur cinématique (m/s ou rad/s)



(b) Erreur normalisée



(c) Images finale et désirée

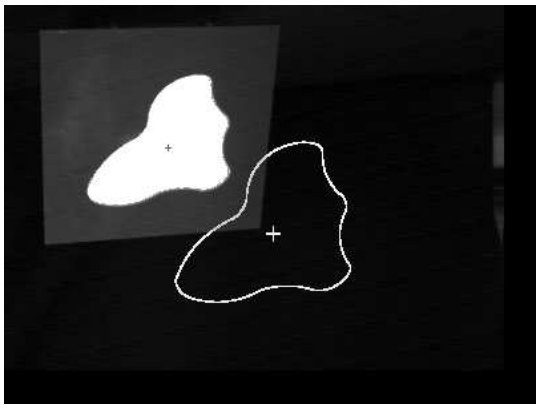
FIG. 3.19 – Résultats expérimentaux — Objet de type c — Positionnement en face de l'objet — Cas d'un minimum local —

Par ailleurs, signalons que les cas de minima locaux obtenus en simulations, n'ont pas pu être reproduit avec les objets de type a ou c le nombre de points du contour, compte tenu de la résolution du capteur et de la taille des objets, étant important ($N \approx 700$). Par contre, ils se sont produits pour l'objet b dans les cas où la profondeur désirée était grande ($z^* \approx 1\text{m} \Rightarrow N \approx 400$, $h = 30$).

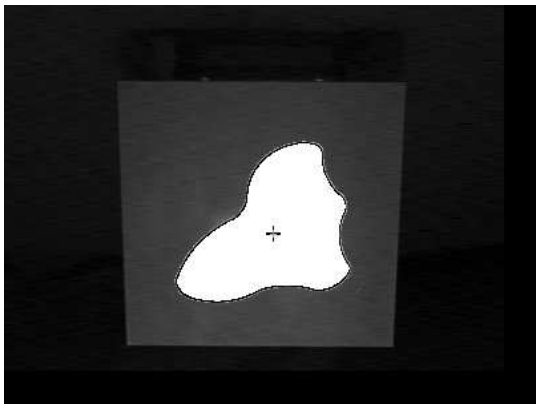
L'expérimentation suivante consiste en la réalisation d'une tâche de positionnement par rapport à l'objet de type a amenant, non plus la caméra en face de l'objet, mais suivant une orientation différente comme nous l'avons déjà fait en simulation. Les conditions d'expérimentations et les résultats quantitatifs sont présentés ci-dessous :

- $h = 20 \Rightarrow \Delta t = 160$ ms
- Situation initiale :
  - Angles d'Euler (en degrés) :
    $$\gamma = 31.21 \pm 0.00 \quad \beta = 16.42 \pm 0.04 \quad \alpha = -4.14 \pm 0.03$$
  - Amplitude de translation par rapport à la situation désirée (en mm) :
    $$\Delta_x = -245 \quad \Delta_y = 103 \quad \Delta_z = 61$$
- Situation désirée :
  - Angles d'Euler (en degrés) :
    $$\gamma = 17.90 \pm 0.01 \quad \beta = 0.06 \pm 0.03 \quad \alpha = 0.31 \pm 0.03$$
  - Translations (en mm) :
    $$t_x = 2.21 \pm 0.03 \quad t_y = 37.64 \pm 0.01 \quad t_z = 634.66 \pm 0.01$$
- Situation obtenue :
  - Angles d'Euler (en degrés) :
    $$\gamma = 17.62 \pm 0.01 \quad \beta = -0.01 \pm 0.02 \quad \alpha = 0.28 \pm 0.02$$
  - Translations (en mm) :
    $$t_x = 3.24 \pm 0.03 \quad t_y = 37.27 \pm 0.02 \quad t_z = 633.46 \pm 0.15$$

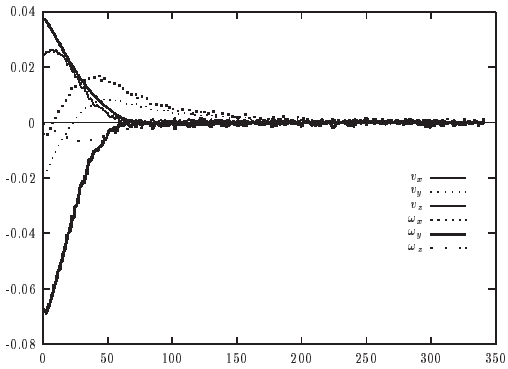
Comme le montre la figure 3.20, le comportement de l'algorithme n'apporte pas de remarques particulières. Signalons, cependant, que le nombre d'harmoniques a dû, une nouvelle fois être augmenté (il est passé de 15 à 20), pour éviter les minima locaux qui apparaissent pour certaines positions initiales.



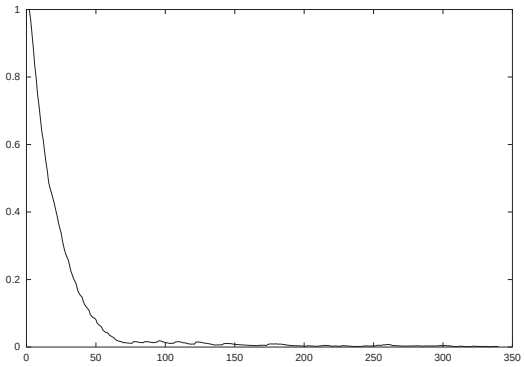
(a) Images actuelle et désirée



(b) Image finale



(c) Composantes du torseur cinématique (m/s ou rad/s)



(d) Erreur normalisée

FIG. 3.20 – Résultats expérimentaux — Objet de type a — Positionnement non en face de l'objet

Cas d'un objet non plan

Cette expérimentation consiste en la réalisation d'une tâche de positionnement en face de l'objet pour lequel il existe les plus grandes différences de profondeurs entre les points du contour, l'objet de type c. Les situations initiale et désirée sont identiques à celles concernant le positionnement en face de l'objet plan du paragraphe précédent. Les conditions d'expérimentations et les résultats quantitatifs sont présentés ci-dessous :

- $h = 25 \Rightarrow \Delta t = 240$ ms
- Situation obtenue :
  - Angles d'Euler (en degrés) :

$$\gamma = -0.38 \pm 0.01 \quad \beta = -0.05 \pm 0.02 \quad \alpha = 0.02 \pm 0.02$$

  - Translations (en mm) :

$$t_x = 1.89 \pm 0.02 \quad t_y = -0.36 \pm 0.02 \quad t_z = 610.72 \pm 0.12$$

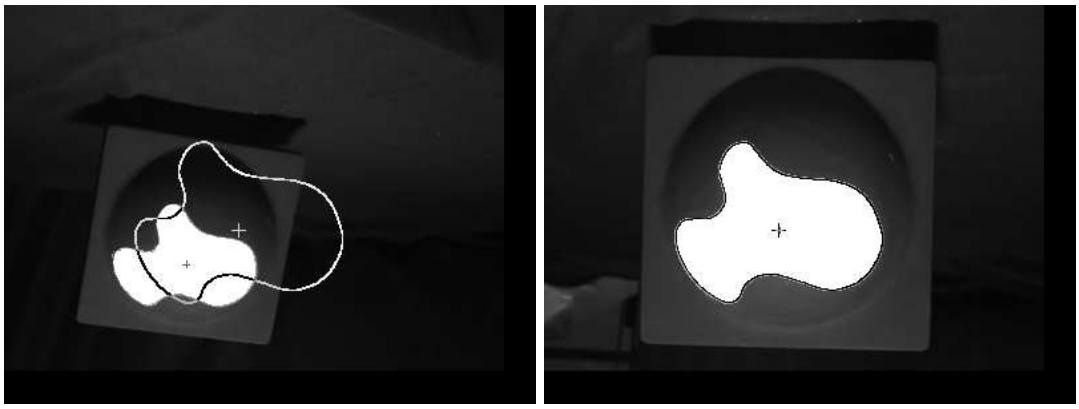
La figure 3.21a représente le motif initial et le motif désiré, la figure 3.21b le motif obtenu, les figures 3.21c et 3.21d les comportements des composantes du torseur cinématique et de l'erreur normalisée.

Ces résultats sont une nouvelle fois tout à fait corrects bien que, rappelons le, toutes les profondeurs soient grossièrement approximées par $\hat{z}^*$. Toutefois, on peut noter que le nombre d'harmoniques a dû être augmenté par rapport à celui utilisé pour un objet plan comme nous l'avions déjà remarqué lors des simulations. La cadence d'acquisition d'images est de ce fait passée de 120 ms à 240 ms.

À présent, intéressons-nous, toujours au même objet, mais pour la réalisation d'une tâche de positionnement où l'orientation désirée ne correspond plus à un positionnement en face de l'objet. Les conditions d'expérimentations et les résultats quantitatifs sont présentés ci-dessous :

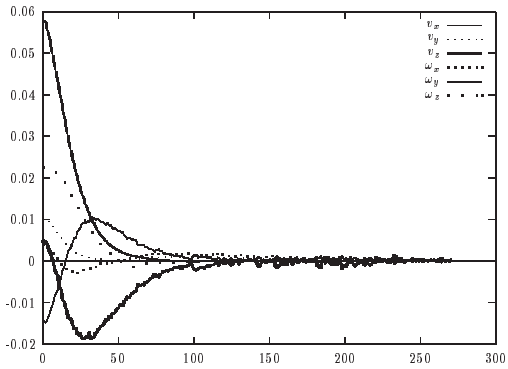
- Situation initiale :
  - Angles d'Euler (en degrés) :

$$\gamma = 31.21 \pm 0.00 \quad \beta = 16.42 \pm 0.04 \quad \alpha = -4.14 \pm 0.03$$

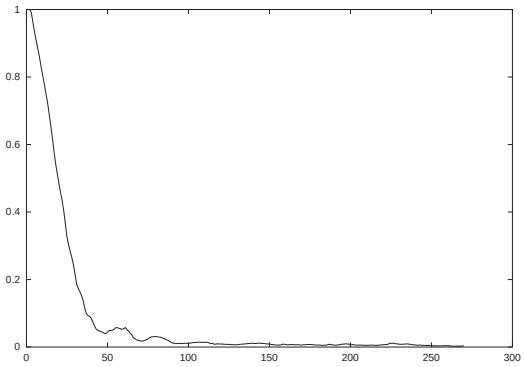


(a) Images actuelle et désirée

(b) Image finale



(c) Composantes du torseur cinématique (m/s ou rad/s)



(d) Erreur normalisée

FIG. 3.21 – Résultats expérimentaux — Objet de type c — Positionnement en face de l'objet

– Amplitude de translation par rapport à la situation désirée (en mm) :

$$\Delta_x = -245 \quad \Delta_y = 103 \quad \Delta_z = 61$$

- Situation désirée :
  – Angles d'Euler (en degrés) :

$$\gamma = 17.99 \pm 0.01 \quad \beta = 0.11 \pm 0.02 \quad \alpha = 0.25 \pm 0.02$$

  – Translations (en mm) :

$$t_x = 3.88 \pm 0.02 \quad t_y = 38.12 \pm 0.02 \quad t_z = 632.94 \pm 0.12$$

- Situation obtenue :
  – Angles d'Euler (en degrés) :

$$\gamma = 17.96 \pm 0.01 \quad \beta = 0.12 \pm 0.02 \quad \alpha = 0.27 \pm 0.01$$

  – Translations (en mm) :

$$t_x = 3.02 \pm 0.02 \quad t_y = 38.00 \pm 0.01 \quad t_z = 632.37 \pm 0.10$$

Au vu de ces résultats et de la figure 3.22, la réalisation de cette tâche n'apporte pas de remarques particulières, elle confirme ce qui a déjà été dit.

Influence de la calibration du dispositif expérimental

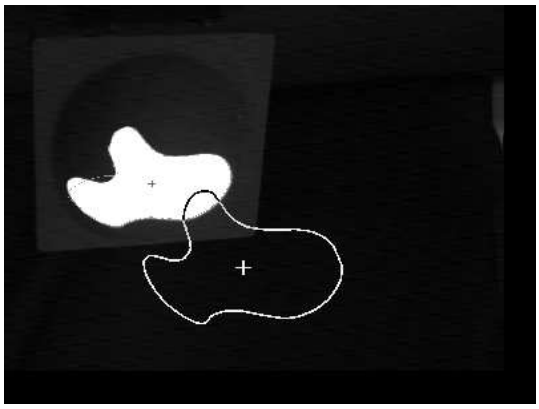
Par ailleurs, dans le but de tester l'influence de la calibration sur les performances de l'algorithme proposé, nous avons considéré le cas où la calibration réalisée est entachée d'erreurs. Deux types de calibration entrent en jeu :

- La calibration de la caméra : elle a été modélisée en considérant un déplacement (X_c, Y_c) par rapport au centre du capteur et en prenant en compte la distorsion radiale K_d. On obtient ainsi le modèle suivant :

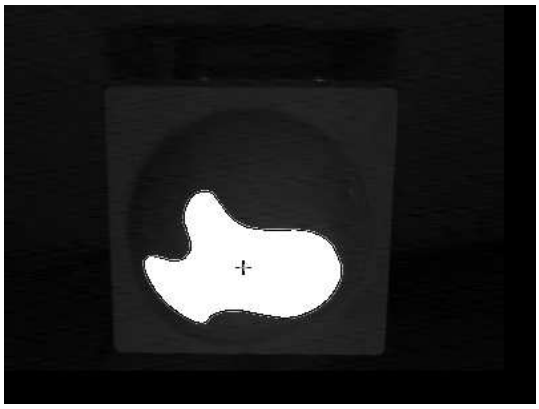
$$\left\{ \begin{array}{rcl} X_p & = & X_c + F_x X \left(1 + K_d \left(X^2 + Y^2\right)\right) \\ Y_p & = & Y_c + F_y Y \left(1 + K_d \left(X^2 + Y^2\right)\right) \end{array} \right. \tag{3.56}$$

  Les résultats de la calibration ont donné les valeurs suivantes :

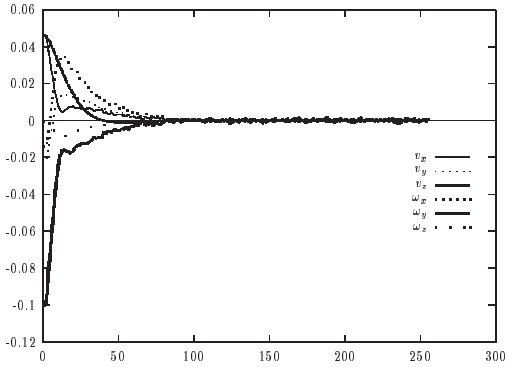
$$\left\{ \begin{array}{rcrlrcrl} X_c & = & 378.2 & \text{(pixels)} & Y_c & = & 299.8 & \text{(pixels)} \\ F_x & = & 1508.5 & & F_y & = & 1508.9 & \\ K_d & = & -0.08 & & & & & \end{array} \right.$$



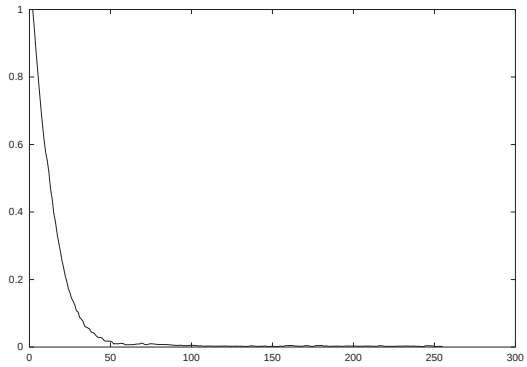
(a) Images actuelle et désirée



(b) Image finale



(c) Composantes du torseur cinématique (m/s ou rad/s)



(d) Erreur normalisée

FIG. 3.22 – Résultats expérimentaux — Objet de type c — Positionnement non en face de l'objet

- La calibration du porteur : elle intervient lors du calcul du jacobien inverse pour fournir, à partir du torseur cinématique, les consignes de vitesse articulaire. Dans cette transformation, le point le plus délicat est l'identification de la matrice M_{pc}. Elle exprime l'orientation et la translation du repère du porteur par rapport au repère de la caméra. L'identification de cette matrice a donné les valeurs suivantes :

$$\begin{pmatrix} -0.0024 & 0.9999 & -0.0001 & -89 \text{ mm} \\ -0.9999 & -0.0024 & -0.0062 & 3 \text{ mm} \\ -0.0062 & 0.0001 & 0.9999 & 46 \text{ mm} \end{pmatrix}$$

Nous avons, dans le cas d'un ensemble robot/caméra calibré, réalisé une tâche de positionnement consistant à placer la caméra en face d'un objet de type b. Les conditions d'expérimentations et les résultats quantitatifs sont présentés ci-dessous :

- $h = 25 \Rightarrow \Delta t = 180$ ms
- Situation désirée :
  - Angles d'Euler (en degrés) :

$$\gamma = -0.21 \pm 0.01 \quad \beta = 0.06 \pm 0.03 \quad \alpha = 0.31 \pm 0.03$$

  - Translations (en mm) :

$$t_x = 0.01 \pm 0.02 \quad t_y = -0.17 \pm 0.02 \quad t_z = 611.03 \pm 0.12$$

- Situation obtenue :
  - Angles d'Euler (en degrés) :

$$\gamma = 0.16 \pm 0.01 \quad \beta = -0.48 \pm 0.02 \quad \alpha = 0.00 \pm 0.01$$

  - Translations (en mm) :

$$t_x = 0.12 \pm 0.02 \quad t_y = 0.26 \pm 0.01 \quad t_z = 611.61 \pm 0.13$$

Nous avons considéré ensuite :

- D'une part, une mauvaise identification des paramètres intrinsèques de la caméra en imposant :

$$\begin{cases} X_c = 391 \text{ (pixels)} & Y_c = 291 \text{ (pixels)} \\ F_x = 1360 & F_y = 1360 \\ K_d = 0 & \end{cases}$$

X_c et Y_c ont été choisis comme étant le milieu de l'image, F_x et F_y ont été minorés de 10 % par rapport aux valeurs obtenues par la calibration, la distorsion a été négligée.

- D'autre part, une mauvaise calibration du robot par le biais de la matrice M_{pc}. Pour ce faire, une erreur de 5 degrés a été introduite sur chaque axe, et une erreur de 3 cm sur les translations. La matrice M_{pc} devient :

$$\begin{pmatrix} 0.0944 & 0.9924 & -0.0789 & -59 \text{ mm} \\ -0.9917 & 0.0868 & -0.0944 & 33 \text{ mm} \\ -0.0868 & 0.0871 & 0.9924 & 76 \text{ mm} \end{pmatrix}$$

Cette mauvaise calibration a conduit, pour la position finale du porteur, aux résultats suivants :

– Angles d'Euler (en degrés) :

$$\gamma = -0.15 \pm 0.01 \quad \beta = -0.44 \pm 0.02 \quad \alpha = 0.05 \pm 0.02$$

– Translations (en mm) :

$$t_x = 0.30 \pm 0.03 \quad t_y = -0.30 \pm 0.01 \quad t_z = 610.88 \pm 0.12$$

Comme on peut le constater, la précision obtenue ne souffre pas de la mauvaise calibration. En revanche, on peut remarquer que la décroissance de l'erreur normalisée est moins rapide avec un dispositif non calibré (voir la figure 3.23).

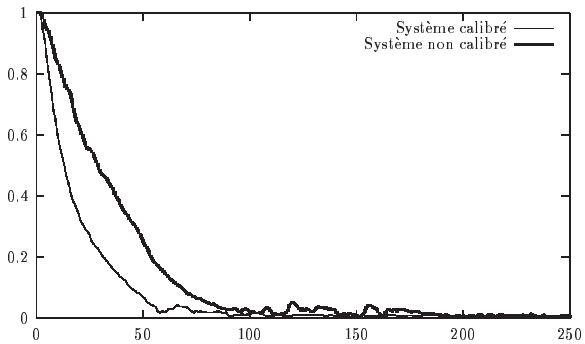


FIG. 3.23 – Résultats expérimentaux — Objet de type b — Influence de la calibration sur l'erreur normalisée —

Influence de la précision de la valeur estimant les profondeurs

Les résultats présentés ont été obtenus sur l'objet non plan de type c, ils concernent l'influence de la précision de l'estimation de la valeur z^*. Pour ce faire, nous avons tout d'abord réalisé une tâche de positionnement amenant le capteur en face de l'objet pour $\hat{z}^* = z^* \approx 65$ cm. Cette tâche n'amène pas de remarques particulières. Nous avons ensuite réalisé la même tâche tout d'abord pour $\hat{z}^* = 30$ cm, puis pour $\hat{z}^* = 200$ cm. Les différentes évolutions des erreurs normalisées et des composantes du torseur cinématique sont représentées sur la figure 3.24.

À la vue de ces figures on retrouve les résultats déjà obtenus en simulation : une forte valeur de $\hat{z}^*$ conduit à des valeurs de commande élevées, une faible valeur à des faibles valeurs des commandes. De même, la décroissance de l'erreur est d'autant plus rapide que la valeur de $\hat{z}^*$ est élevée. Toutefois, on peut remarquer, contrairement aux simulations, l'apparition d'une portion croissante dans le graphe représentant l'erreur normalisée avant une décroissance définitive (voir les figures 3.24a et 3.24e). Ce phénomème reste inexpliqué. En revanche, l'influence de $\hat{z}^*$ sur le chemin parcouru par la caméra confirme les résultats de simulation comme en témoigne la figure 3.25, une forte valeur de $\hat{z}^*$ conduit à un chemin beaucoup plus long.

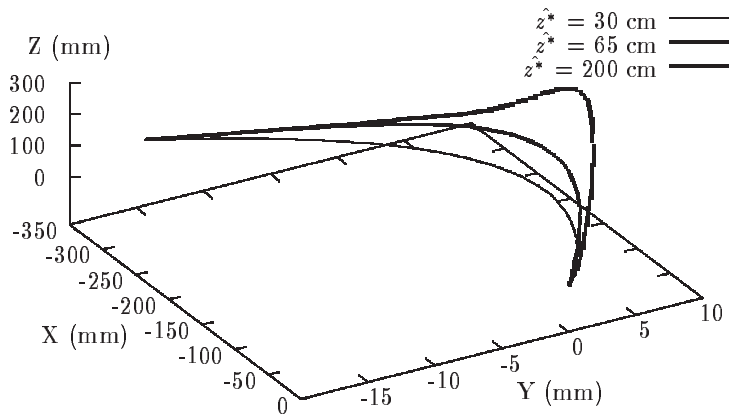
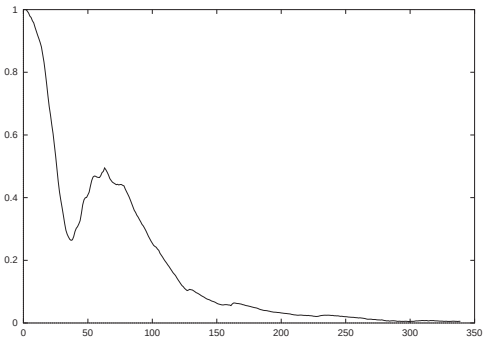
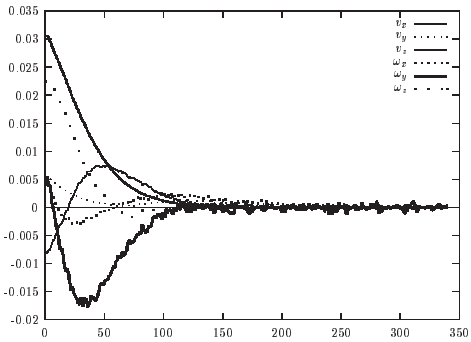


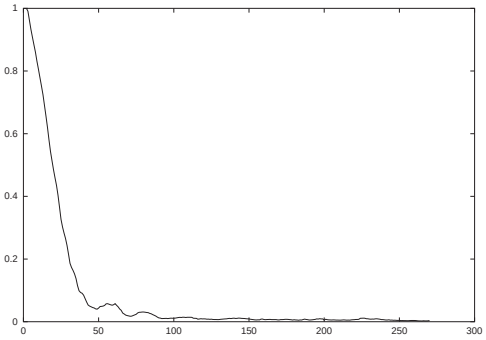
FIG. 3.25 – Résultats expérimentaux — Objet de type c — Influence de $\hat{z}^*$ sur la trajectoire de la caméra



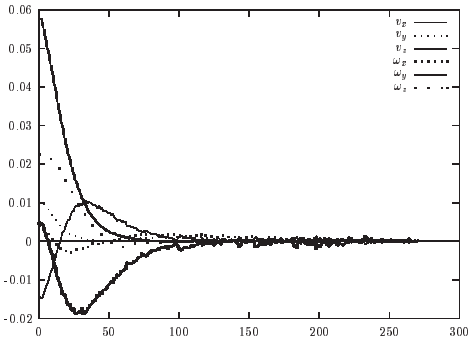
(a) Erreur normalisée pour $\hat{z}^* = 30$ cm



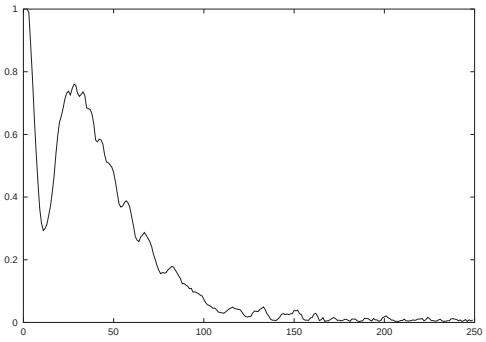
(b) Composantes du torseur cinématique (m/s ou rad/s) pour $\hat{z}^*$ = 30 cm



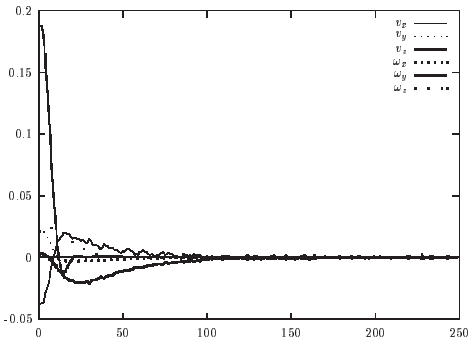
(c) Erreur normalisée pour $\hat{z}^* = 65$ cm



(d) Composantes du torseur cinématique (m/s ou rad/s) pour $\hat{z}^*$ = 65 cm



(e) Erreur normalisée pour $\hat{z}^*$ = 200 cm



(f) Composantes du torseur cinématique (m/s ou rad/s) pour $\hat{z}^*$ = 200 cm

FIG. 3.24 – Résultats expérimentaux — Objet de type c — Influence de $\hat{z}^*$

Par ailleurs, nous avons vérifié, toujours par la méthode de Dementhon, que la précision du positionnement n'est pas affectée par le choix de $\hat{z}^*$. Il s'en suit, dans les cas où aucune approximation n'est disponible, un choix confortable pour la valeur de $\hat{z}^*$.

Influence d'une occlusion partielle de l'objet

Les résultats obtenus jusqu'alors étant satisfaisants, nous avons voulu tester l'influence d'une occlusion partielle de l'objet de type c. L'obtenir lors du mouvement du porteur n'est pas très aisée, aussi avons-nous positionné le capteur de telle sorte que l'objet ne soit pas entièrement visible en position initiale.

Pour de faibles occlusions comme celle rapportées sur la figure 3.26, la convergence de l'algorithme s'obtient sans problème. Signalons que des résultats similaires ont été obtenus quand, en position initiale, le contour de l'objet n'est pas entièrement visible en raison de la non planéité de l'objet.

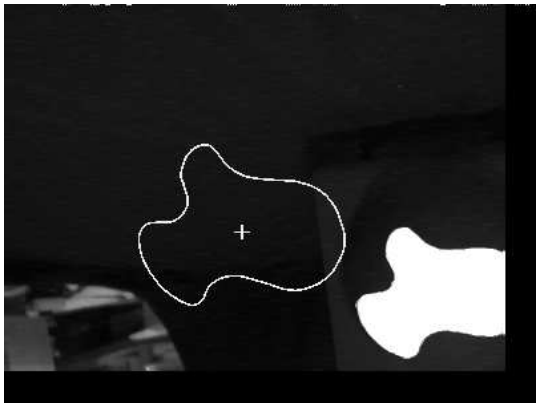
3.5.5 Conclusion

Nous nous sommes intéressés, dans ce début de chapitre, à la réalisation de tâches de positionnement par rapport à des objets, plans on non, dont la forme est complexe.

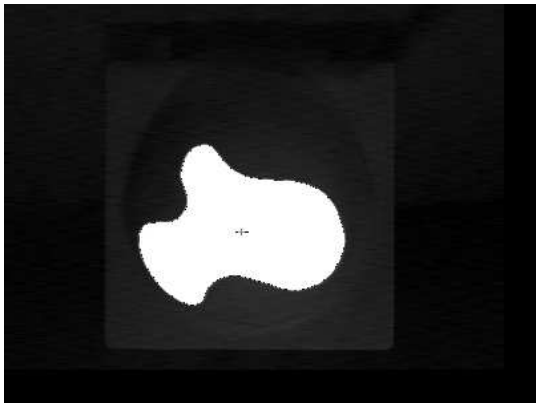
Dans un premier temps, nous avons calculé une approximation de la matrice d'interaction associée à des informations visuelles issues du contour de l'image de l'objet, la signature polaire. Les résultats, obtenus tant en simulations que sur site expérimental, ont validé l'approche utilisée pour ce calcul.

Par ailleurs, ils ont montré que la convergence de la loi de commande était possible sous de très faibles hypothèses. En effet, seuls les contours souhaité et courant sont nécessaires, aucune connaissance précise des dimensions de l'objet ou de sa localisation n'est requise. De plus, les résultats expérimentaux ont montré que cette loi de commande était robuste vis-à-vis de l'estimation réalisée sur les profondeurs, des occlusions partielles et d'un dispositif robot et capteur mal calibré.

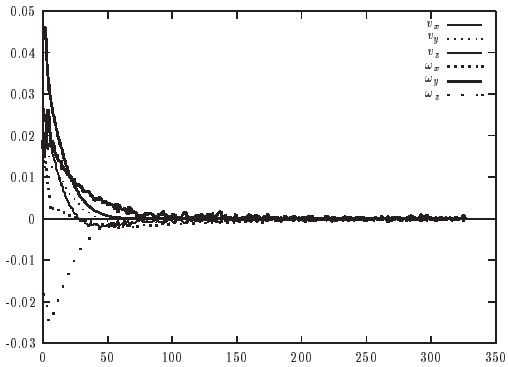
En outre, bien qu'aucune simulation ou expérimentation n'ait été réalisée, l'approche proposée devrait pouvoir appréhender plus généralement le cas des tâches de suivi. De même, elle doit pouvoir traiter le cas d'informations visuelles provenant d'un contour ouvert.



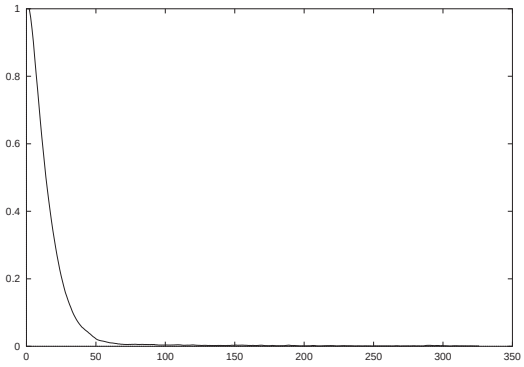
(a) Images actuelle et désirée



(b) Image finale



(c) Composantes du torseur cinématique (m/s ou rad/s)



(d) Erreur normalisée

FIG. 3.26 – Résultats expérimentaux — Objet de type c — Test d'occlusion

Toutefois, on peut déplorer le coût des calculs nécessités par cette méthode et les conséquences occasionnées sur la cadence d'acquisition des images et, par suite, sur les relatives faibles performances dynamiques du porteur.

Après avoir décrit un contour par le biais d'une représentation en coordonnées polaires, nous allons maintenant nous intéresser à la réalisation de tâches de positionnement à l'aide d'informations visuelles issues d'une description de ce contour sous forme paramétrique.

3.6 Asservissement visuel basé sur une représentation paramétrique

Nous avons vu en 3.4.3 que ce type de représentation était très bien adapté à la modélisation de formes quelconques. Aussi, allons-nous, dans le cas général, chercher à obtenir la matrice d'interaction associée à des informations visuelles issues d'une représentation paramétrique d'un contour.

3.6.1 Calcul de la matrice d'interaction

Soit $(\mathcal{C})$ le contour de l'image de l'objet observé, il représente le support d'un arc géométrique γ défini par :

$$\begin{array}{rcl} \mathcal{F}: & I \longrightarrow & \mathbb{R}^2 \\ & u \longrightarrow & \mathcal{F}(u) = \begin{pmatrix} X(u) = f(u, \underline{P}) \\ Y(u) = g(u, \underline{Q}) \end{pmatrix} \end{array}$$

où I est un intervalle de $\mathbb{R}$ décrivant le domaine de variation de la variable u, $\underline{P}$ et $\underline{Q}$ des paramétrages des fonctions f et g. Bien que, théoriquement, cela ne soit pas nécessaire, nous supposons dim $\underline{P}$ = dim $\underline{Q} = m/2$ afin de simplifier les développements ultérieurs (avec m la dimension du vecteur constitué des informations visuelles).

En différenciant les fonctions f et g par rapport au temps, on obtient une expression de la vitesse d'un point de coordonnées $(X(u), Y(u))^T$ dans le capteur :

$$\left\{ \begin{array}{rcl} \dot{X} & = & \dfrac{\partial f}{\partial u}\dot{u} + \dfrac{\partial f}{\partial \underline{P}}\dot{\underline{P}} \\ \dot{Y} & = & \dfrac{\partial g}{\partial u}\dot{u} + \dfrac{\partial g}{\partial \underline{Q}}\dot{\underline{Q}} \end{array} \right. \tag{3.57}$$

On pourrait être tenté, comme on l'a fait en 3.5.1 pour $\dot{\theta}$ dans le cas de la signature polaire, d'éliminer $\dot{u}$ entre les deux équations, puis, de choisir m valeurs distinctes de u conduisant aux couples $(X(u), Y(u))$ et enfin d'inverser le système ainsi obtenu. Mais, sachant que la paramétrisation d'un contour sous une forme paramétrique n'est pas minimale, cette façon de faire aboutirait nécessairement à une matrice à inverser singulière. Toutefois, en réécrivant (3.57) sous la forme :

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \end{pmatrix} = A \begin{pmatrix} \dot{u} \\ \dot{\underline{P}} \\ \dot{\underline{Q}} \end{pmatrix} \tag{3.58}$$

où :

$$A = \begin{pmatrix} \dfrac{\partial f}{\partial u} & \dfrac{\partial f}{\partial \underline{P}} & 0 \\ \dfrac{\partial g}{\partial u} & 0 & \dfrac{\partial g}{\partial \underline{Q}} \end{pmatrix} \tag{3.59}$$

on peut chercher à éliminer $\dot{u}$ d'une autre façon. En effet, on peut, au sens des moindres carrés, résoudre le système ci-dessus par rapport aux variables $\dot{u}$, $\dot{\underline{P}}$ et $\dot{\underline{Q}}$. Pour ce faire, prémultiplions chaque membre de (3.58) par A^T, il vient :

$$A^T \begin{pmatrix} \dot{X} \\ \dot{Y} \end{pmatrix} = A^T A \begin{pmatrix} \dot{u} \\ \dot{\underline{P}} \\ \dot{\underline{Q}} \end{pmatrix} \tag{3.60}$$

Si la matrice carrée $A^T A$ est régulière, on obtient $\dot{u}$, $\dot{\underline{P}}$ et $\dot{\underline{Q}}$ en inversant ce système. Or, on peut montrer que $A^T A$ se met sous la forme suivante :

$$A^T A = \begin{pmatrix} \dfrac{\partial f}{\partial u}^2 + \dfrac{\partial g}{\partial u}^2 & \dfrac{\partial f}{\partial u}\dfrac{\partial f}{\partial \underline{P}} & \dfrac{\partial g}{\partial u}\dfrac{\partial g}{\partial \underline{Q}} \\ \dfrac{\partial f}{\partial u}\dfrac{\partial f}{\partial \underline{P}} & \dfrac{\partial f}{\partial \underline{P}}^2 & 0 \\ \dfrac{\partial g}{\partial u}\dfrac{\partial g}{\partial \underline{Q}} & 0 & \dfrac{\partial g}{\partial \underline{Q}}^2 \end{pmatrix} \tag{3.61}$$

qui est malheureusement aussi une matrice singulière. En effet, en développant le déterminant de cette matrice par rapport aux éléments de la dernière ligne, on peut montrer aisément qu'il s'annule.

Il semble donc qu'il faille, pour chaque point du contour, estimer $\dot{u}$. Cela signifie que, d'une image acquise à l'instant t, on sache, dans l'image acquise à l'instant $t+1$, mettre en correspondance tous les points de ce contour avec

ceux de l'image précédente (tout au moins ceux utiles au calcul de la matrice d'interaction). Ce processus doit être itéré jusqu'à la position désirée du porteur afin d'en déduire les valeurs de u associées à chaque point du contour et par suite les informations visuelles. En pratique, il faut donc réaliser la tâche pour obtenir la consigne visuelle ! De plus, la consigne ainsi obtenue est dépendante de la position initiale du porteur. Cette façon de procéder n'est donc pas viable.

La seule alternative restante est donc de supposer les effets des déformations du contour dues aux variations de $\underline{P}$ ou $\underline{Q}$ prépondérants par rapport à ceux de u, c'est-à-dire négliger les variations du contour dues à u. Pour rendre cette hypothèse la plus valide possible, il convient d'imposer un minimum de contraintes :

- Tout d'abord, comme le calcul de $\underline{P}$ et $\underline{Q}$ sur lesquels sont basées les valeurs des informations visuelles dépend de l'origine de la variable u, il importe, pour chaque image, de conserver une même convention. Une façon simple consiste, à choisir dans un premier temps une origine quelconque, à calculer les valeurs constituant $\underline{Q}$, puis, par exemple, à chercher u_0 tel que $g(u_0, \underline{Q}) = Y_c$ où Y_c est le centre de gravité de l'image de l'objet. On recalcule ensuite les valeurs de $\underline{P}$ et $\underline{Q}$ en considérant cette nouvelle origine (voir dans le paragraphe 3.7.3 un exemple de ce calcul dans le cas des descripteurs de Fourier). De cette façon, on fixe bien une même origine des u mais, par contre, pas la manière dont ils sont distribués le long du contour.

- Par ailleurs, considérons à présent le cas où la caméra est en face de l'objet — supposé plan pour cette occasion — et supposons un mouvement de translation suivant l'axe z. Il est aisé de montrer, dans ces conditions, que l'image subit une déformation d'un facteur d'échelle fonction du déplacement réalisé. Dans ce cas, la mise en correspondance entre les deux images est triviale. En effet, les points se déplacent radialement d'une image à l'autre comme le rapporte la figure 3.27. Par conséquent, pour tenir compte de ce type de déformation qui se gère facilement, nous introduisons la contrainte suivante : à chaque image les points du contour sont rééchantillonnés d'une part, de telle manière à ce que tous les points soient équidistants, d'autre part, de telle manière à ce que le nombre N de points soit constant. Il s'en suit un même angle α, de valeur $2\pi/N$ tel que :

$$\begin{aligned} \cos\alpha &= \frac{\overrightarrow{OM(u_j)}_t \overrightarrow{OM(u_{j+1})}_t}{\|\overrightarrow{OM(u_j)}_t\| \|\overrightarrow{OM(u_{j+1})}_t\|} \\ &= \frac{\overrightarrow{OM(u_j)}_{t+1} \overrightarrow{OM(u_{j+1})}_{t+1}}{\|\overrightarrow{OM(u_j)}_{t+1}\| \|\overrightarrow{OM(u_{j+1})}_{t+1}\|} \end{aligned} \tag{3.62}$$

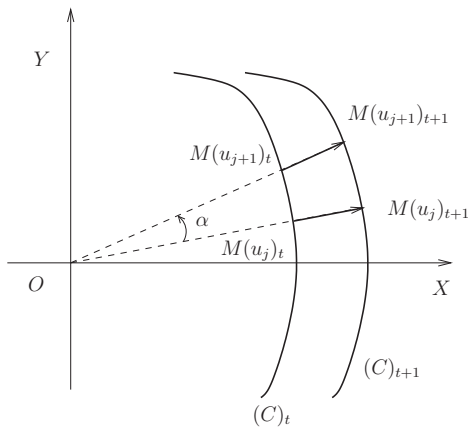


FIG. 3.27 – Transformation d'un contour due à un mouvement de translation dans la direction de l'objet

Ces deux contraintes étant imposées, on peut considérer légitimement l'égalité $\dot{u} = 0$ dans tous les cas de mouvements de translation et dans les cas de rotation d'axe z. Notons cependant que la deuxième contrainte a été établie en supposant l'objet plan et la caméra en face de l'objet considéré.

Dans ces conditions, le système (3.57) se simplifie en :

$$\left\{ \begin{array}{rcl} \dot{X} - \dfrac{\partial f}{\partial \underline{P}} \dot{\underline{P}} & = & 0 \\ \dot{Y} - \dfrac{\partial g}{\partial \underline{Q}} \dot{\underline{Q}} & = & 0 \end{array} \right. \tag{3.63}$$

Ce système est maintenant formé de deux équations indépendantes, on choisit alors $m/2$ valeurs de u conduisant à $m/2$ couples $(X(u), Y(u))$ et on opère ensuite classiquement. Il vient alors immédiatement :

$$L_{\underline{P}}^T = \left(\frac{\partial \underline{f}}{\partial \underline{P}} \right)^{-1} \begin{pmatrix} L_{X_1}^T \\ \vdots \\ L_{X_{m/2}}^T \end{pmatrix} \tag{3.64}$$

et :

$$L_{\underline{Q}}^T = \left(\frac{\partial \underline{g}}{\partial \underline{Q}} \right)^{-1} \begin{pmatrix} L_{Y_1}^T \\ \vdots \\ L_{Y_{m/2}}^T \end{pmatrix} \tag{3.65}$$

dans lesquelles on a posé :

$$\left\{ \begin{array}{rcl} \underline{f} & = & \big(f(u_1), \ldots, f(u_{m/2})\big)^T \\ \underline{g} & = & \big(g(u_1), \ldots, g(u_{m/2})\big)^T \end{array} \right. \tag{3.66}$$

Remarque : dans le cas où les fonctions f et g se décomposent linéairement sur une même base de fonctions par rapport aux paramétrages $\underline{P}$ et $\underline{Q}$ (comme on le verra en 3.7.1 dans le cas des descripteurs de Fourier) on montre aisément que :

$$\frac{\partial \underline{f}}{\partial \underline{P}} = \frac{\partial \underline{g}}{\partial \underline{Q}} \tag{3.67}$$

Le cas général étant traité, nous allons à présent l'appliquer au calcul de la matrice d'interaction dans le cas d'une fonction paramétrique très utilisée en vision, celle basée sur les descripteurs de Fourier.

3.7 L'exemple des descripteurs de Fourier

3.7.1 Calcul de la matrice d'interaction

Ce type de représentation permet de décrire chaque point du contour sous la forme suivante :

$$\begin{pmatrix} X(u) \\ Y(u) \end{pmatrix} = \begin{pmatrix} a_0 \\ c_0 \end{pmatrix} + \sum_{k=1}^{k=h} \begin{pmatrix} a_k & b_k \\ c_k & d_k \end{pmatrix} \begin{pmatrix} \cos ku \\ \sin ku \end{pmatrix} \tag{3.68}$$

on a donc, dans ce cas, $m = 4h + 2$.

Le calcul de la matrice d'interaction s'effectue grâce à l'utilisation de (3.64) et (3.65) qui conduisent à :

$$\left\{ \begin{array}{rcl} L_{\underline{P}}^T & = & M \begin{pmatrix} L_{X_1}^T \\ \vdots \\ L_{X_{m/2}}^T \end{pmatrix} \\ \\ L_{\underline{Q}}^T & = & M \begin{pmatrix} L_{Y_1}^T \\ \vdots \\ L_{Y_{m/2}}^T \end{pmatrix} \end{array} \right. \tag{3.69}$$

avec :

$$M = \begin{pmatrix} 1 & \cos u_1 & \cdots & \cos hu_1 & \sin u_1 & \cdots & \sin hu_1 \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 1 & \cos u_{m/2} & \cdots & \cos hu_{m/2} & \sin u_{m/2} & \cdots & \sin hu_{m/2} \end{pmatrix}^{-1} \tag{3.70}$$

et :

$$\begin{aligned} \underline{P} &= (a_0, \cdots, a_h, b_1, \cdots, b_h)^T \\ \underline{Q} &= (c_0, \cdots, c_h, d_1, \cdots, d_h)^T \end{aligned} \tag{3.71}$$

Par suite, on en déduit la matrice d'interaction recherchée :

$$L_{\underline{s}}^T = \begin{pmatrix} M & 0 \\ 0 & M \end{pmatrix} \begin{pmatrix} L_{X_1}^T \\ \vdots \\ L_{X_{m/2}}^T \\ L_{Y_1}^T \\ \vdots \\ L_{Y_{m/2}}^T \end{pmatrix} \tag{3.72}$$

avec $\underline{s} = \left(\underline{P}, \underline{Q}\right)^T$.

Remarque : on peut vérifier, cette fois, qu'en choisissant $m/2$ valeurs distinctes de u la matrice M^{-1} est toujours régulière.

Étudions à présent plus en détail la matrice $L_{\underline{s}}^T$. Pour ce faire, intéressons nous à une ligne i particulière :

- si $i \leq m/2$, soit p_i cette information visuelle. Connaissant l'expression de $L_{X_j}^T$ en fonction de X_j et Y_j avec $X_j = X(u_j)$ et $Y_j = Y(u_j)$, il vient immédiatement :

$$L_{p_i}^T = \sum_{j=1}^{j=m/2} \begin{pmatrix} -\dfrac{m_{ij}}{z_j} \\ 0 \\ m_{ij}\dfrac{X_j}{z_j} \\ m_{ij}X_jY_j \\ -m_{ij}(1+{X_j}^2) \\ m_{ij}Y_j \end{pmatrix}^T \tag{3.73}$$

- de même, si $m/2 < i \leq m$, notons q_i l'information visuelle étudiée, il vient alors en connaissant l'expression de $L_{Y_j}^T$ en fonction de X_j et Y_j :

$$L_{q_i}^T = \sum_{j=1}^{j=m/2} \begin{pmatrix} 0 \\ -\dfrac{m_{ij}}{z_j} \\ m_{ij}\dfrac{Y_j}{z_j} \\ m_{ij}(1+{Y_j}^2) \\ -m_{ij}X_jY_j \\ -m_{ij}X_j \end{pmatrix}^T \tag{3.74}$$

Par suite, l'utilisation explicite de (3.68) dans les expressions de $L_{p_i}^T$ et $L_{q_i}^T$ conduit aux relations suivantes :

$$\begin{cases} L_{\underline{P}_{i,1}}^T &= -\sum_j m_{i,j}/z_j \\ L_{\underline{P}_{i,2}}^T &= 0 \\ L_{\underline{P}_{i,3}}^T &= a_0 \sum_j m_{i,j}/z_j + \sum_j m_{i,j}\beta_j/z_j \\ L_{\underline{P}_{i,4}}^T &= a_0c_0 \sum_j m_{i,j} + c_0 \sum_j m_{i,j}\beta_j + a_0 \sum_j m_{i,j}\gamma_j + \varphi_i \\ L_{\underline{P}_{i,5}}^T &= -(1+a_0^2) \sum_j m_{i,j} - 2a_0 \sum_j m_{i,j}\beta_j + \psi_i^p \\ L_{\underline{P}_{i,6}}^T &= c_0 \sum_j m_{i,j} + \sum_j m_{i,j}\gamma_j \\ L_{\underline{Q}_{i,1}}^T &= 0 \\ L_{\underline{Q}_{i,2}}^T &= -\sum_j m_{i,j}/z_j \\ L_{\underline{Q}_{i,3}}^T &= c_0 \sum_j m_{i,j}/z_j + \sum_j m_{i,j}\gamma_j/z_j \\ L_{\underline{Q}_{i,4}}^T &= (1+c_0^2) \sum_j m_{i,j} + 2c_0 \sum_j m_{i,j}\gamma_j + \psi_i^q \\ L_{\underline{Q}_{i,5}}^T &= -a_0c_0 \sum_j m_{i,j} - c_0 \sum_j m_{i,j}\beta_j - a_0 \sum_j m_{i,j}\gamma_j - \varphi_i \\ L_{\underline{Q}_{i,6}}^T &= -a_0 \sum_j m_{i,j} - \sum_j m_{i,j}\beta_j \end{cases} \tag{3.75}$$

relations dans lesquelles on a posé :

- $\beta_j = \sum_k a_k \cos ku_j + b_k \sin ku_j$;
- $\gamma_j = \sum_k c_k \cos ku_j + d_k \sin ku_j$;

- $\varphi_i = \sum_j m_{i,j}\beta_j\gamma_j$;
- $\psi_i^p = -\sum_j m_{i,j}\beta_j^2$;
- $\psi_i^q = \sum_j m_{i,j}\gamma_j^2$;

Les relations (3.75) font exhiber les termes $\sum_j m_{i,j}$, $\sum_j m_{i,j}\beta_j$ et $\sum_j m_{i,j}\gamma_j$. Or, on peut vérifer, numériquement ou à l'aide d'un calculateur symbolique, que ces termes se simplifient, suivant les valeurs de i, et quelques soient les valeurs des u_j en :

$$\left\{\begin{array}{rcl} \sum_j m_{i,j} & = & 1, \underbrace{0,\ldots,0}_{m/2-1 fois} \\ \sum_j m_{i,j}\beta_j & = & 0, a_1,\ldots,a_h,b_1,\ldots,b_h \\ \sum_j m_{i,j}\gamma_j & = & 0, c_1,\ldots,c_h,d_1,\ldots,d_h \end{array}\right. \tag{3.76}$$

Par suite, en supposant toutes les profondeurs égales à $\hat{z}^*$, on peut en déduire une expression simplifiée de la matrice d'interaction recherchée :

$$\widehat{L_{\underline{s}}^T}\Big|_{\hat{z}^*} = \begin{pmatrix} -\frac{1}{\hat{z}^*} & 0 & \frac{a_0}{\hat{z}^*} & a_0c_0+\varphi_1 & -1-a_0^2+\psi_1^p & c_0 \\ 0 & 0 & \frac{a_1}{\hat{z}^*} & a_0c_1+c_0a_1+\varphi_2 & -2a_0a_1+\psi_2^p & c_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \frac{a_h}{\hat{z}^*} & a_0c_h+c_0a_h+\varphi_{h+1} & -2a_0a_h+\psi_{h+1}^p & c_h \\ 0 & 0 & \frac{b_1}{\hat{z}^*} & a_0d_1+c_0b_1+\varphi_{h+2} & -2a_0b_1+\psi_{h+2}^p & d_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \frac{b_h}{\hat{z}^*} & a_0d_h+c_0b_h+\varphi_{m/2} & -2a_0b_h+\psi_{m/2}^p & d_h \\ 0 & -\frac{1}{\hat{z}^*} & \frac{c_0}{\hat{z}^*} & 1+c_0^2+\psi_1^q & -a_0c_0-\varphi_1 & -a_0 \\ 0 & 0 & \frac{c_1}{\hat{z}^*} & 2c_0c_1+\psi_2^q & -a_0c_1-c_0a_1-\varphi_2 & -a_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \frac{c_h}{\hat{z}^*} & 2c_0c_h+\psi_{h+1}^q & -a_0c_h-c_0a_h-\varphi_{h+1} & -a_h \\ 0 & 0 & \frac{d_1}{\hat{z}^*} & 2c_0d_1+\psi_{h+2}^q & -a_0d_1-c_0b_1-\varphi_{h+2} & -b_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \frac{d_h}{\hat{z}^*} & 2c_0d_h+\psi_{m/2}^q & -a_0d_h-c_0b_h-\varphi_{m/2} & -b_h \end{pmatrix} \tag{3.77}$$

Tout comme dans le cas de la signature polaire, la forme de cette matrice met clairement en évidence les points suivants :

- À la vue des première et $(h+2)^{\text{ième}}$ lignes de $\widehat{L_{\underline{s}}^T}\Big|_{\hat{z}^*}$ on s'aperçoit que la matrice d'interaction relative à l'information visuelle (a_0, c_0) diffère de celle de l'information visuelle (a_0, c_0), quand elle est considérée en tant que la projection d'un point, uniquement par l'ajout des fonctions φ_1, ψ_1^p et ψ_1^q, c'est-à-dire, sur les termes relatifs aux rotations d'axes x et y.

- Le découplage des mouvements de translation suivant les axes x et y.

- L'invariance de la forme de l'image de l'objet, forme décrite par les informations visuelles (a_k, b_k, c_k, d_k) pour $k \neq 0$, en cas de translation suivant les axes x et/ou y.

De plus, notons que l'on retrouve une nouvelle fois les conséquences de la non vérification de la contrainte de rigidité des informations visuelles par le biais des termes φ_i, ψ_i^p et ψ_i^q.

La matrice d'interaction étant obtenue, nous allons décrire succinctement la façon dont sont obtenues les informations visuelles.

3.7.2 Extraction des informations visuelles

Comme dans le cas de la signature polaire, et pour les mêmes raisons que celles avancées en 3.5.2, la méthode des moindres carrés est utilisée. Les fonctions f et g n'étant dépendantes qu'un seul des deux paramétrages $\underline{P}$ ou $\underline{Q}$, le problème est séparable. Par suite, il nous faut minimiser les deux critères suivants (en conservant les notations introduites en 3.5.2) :

$$\left\{ \begin{array}{l} J_P = \displaystyle\sum_{i=1}^{i=N} \left(X_r(u_i) - \underline{\alpha}_i^T \hat{\underline{P}} \right)^2 \\ J_Q = \displaystyle\sum_{i=1}^{i=N} \left(Y_r(u_i) - \underline{\alpha}_i^T \hat{\underline{Q}} \right)^2 \end{array} \right. \tag{3.78}$$

où $X_r(u_i)$ et $Y_r(u_i)$ sont les coordonnées réellement mesurées en u_i sur le contour, après rééchantillonnage.

La solution est alors donnée par :

$$\left\{\begin{array}{rcl} \underline{\hat{P}} & = & \left(\sum_{i=1}^{i=N} \underline{\alpha}_i \underline{\alpha}_i^T\right)^{-1} \sum_{i=1}^{i=N} \underline{\alpha}_i X_r(u_i) \\ \underline{\hat{Q}} & = & \left(\sum_{i=1}^{i=N} \underline{\alpha}_i \underline{\alpha}_i^T\right)^{-1} \sum_{i=1}^{i=N} \underline{\alpha}_i Y_r(u_i) \end{array}\right. \tag{3.79}$$

Remarque : Les fonctions f et g se décomposant sur une même base de fonctions, le calcul de $\underline{\hat{P}}$ et $\underline{\hat{Q}}$ ne nécessite l'inversion que d'une seule matrice, en l'occurence la matrice $\sum_i \underline{\alpha}_i \underline{\alpha}_i^T$.

3.7.3 Convention concernant l'origine de la variable u

Nous avons vu en 3.6.1 qu'il était nécessaire de choisir une convention sur l'origine de la variable de parcours du contour pour satisfaire, en partie, l'hypothèse $\dot{u} = 0$. Examinons par conséquent l'effet d'une translation u_0 du paramètre u sur le contour considéré :

$$\begin{pmatrix} X(u-u_0) \\ Y(u-u_0) \end{pmatrix} = \begin{pmatrix} a_0 \\ c_0 \end{pmatrix} + \sum_{k=1}^{k=h} \begin{pmatrix} a_k & b_k \\ c_k & d_k \end{pmatrix} \begin{pmatrix} \cos k(u-u_0) \\ \sin k(u-u_0) \end{pmatrix} \tag{3.80}$$

Simplifications faites, on obtient l'écriture des nouvelles informations visuelles $(a_k^{'}, b_k^{'}, c_k^{'}, d_k^{'})$ avec $k \neq 0$ par :

$$\begin{pmatrix} a_k^{'} & b_k^{'} \\ c_k^{'} & d_k^{'} \end{pmatrix} = \begin{pmatrix} a_k & b_k \\ c_k & d_k \end{pmatrix} \begin{pmatrix} \cos ku_0 & \sin ku_0 \\ -\sin ku_0 & \cos ku_0 \end{pmatrix} \tag{3.81}$$

Par conséquent, et conformément à 3.6.1, le processus d'obtention des informations se déroule comme suit. Tout d'abord, après rééchantillonnage, on calcule les informations visuelles par rapport à une origine arbitraire, on dispose ainsi des (a_k, b_k, c_k, d_k). Puis, on cherche u_0 en résolvant, à chaque nouvelle image, $Y(u_0) = c_0$ [3]. Enfin, les informations visuelles réellement utilisées sont obtenues par (3.81).

Ces principaux résultats étant établis, nous allons tester dans le paragraphe suivant le comportement de la loi de commande :

$$T_c = -\lambda \left(L^T \big|_{\underline{s}=\underline{s}^*, z=\hat{z^*}} \right)^+ (\underline{s} - \underline{s}^*) \tag{3.82}$$

[3] Dans le cas des descripteurs de Fourier, on peut montrer que le centre de gravité de l'image a pour coordonnées (a_0, c_0).

avec :

$$\underline{s} = (a_0, \cdots, a_h, b_1, \cdots, b_h, c_0, \cdots, c_h, d_1, \cdots, d_h)^T \tag{3.83}$$

3.7.4 Résultats de simulations

Les exemples de simulations proposés ici sont identiques à ceux présentés en 3.5.3 dans le cas de la signature polaire : l'utilisation des objets plan puis non plan, la réalisation des tâches de positionnement en face ou suivant une orientation différente par rapport à l'objet, le comportement de la loi de commande pour ces différents objets et ces différentes tâches en environnement bruité. De même, l'influence des deux paramètres intervenant dans la loi de commande, à savoir, le nombre d'harmoniques et l'estimation de la profondeur une fois la tâche réalisée, est discutée.

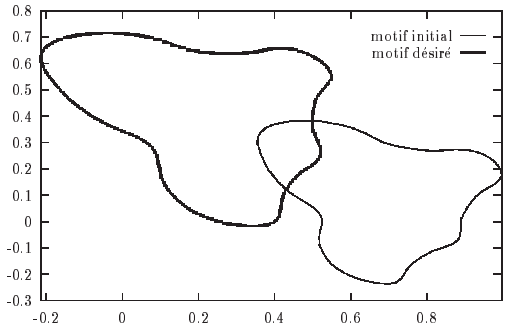
3.7.4.1 Simulations non bruitées

Cas d'un objet plan

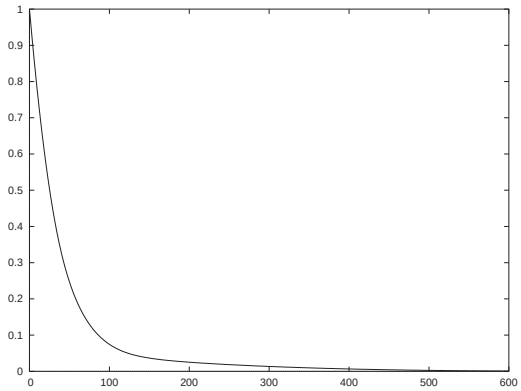
Nous considérons tout d'abord le cas d'un objet plan et supposons qu'à tout instant le contour de l'objet reste entièrement visible. Par ailleurs, nous supposons modéliser le contour grâce à h_m harmoniques. Par contre, la loi de commande n'est supposée n'en utiliser que h_c. Cette façon de procéder permet de travailler avec h_c harmoniques, obtenues à partir de h_m harmoniques, plus représentatives que si elles étaient obtenues à partir de $h_m = h_c$ harmoniques mais elle permet aussi de diminuer la complexité des calculs. Toutefois, comme on le verra en 3.7.4.3, cela peut conduire à certains désavantages.

La figure 3.28a représente le motif initial et le motif désiré, les figures 3.28b et 3.28c représentent respectivement l'erreur normalisée relative aux informations visuelles de dimension $4h_c + 2$ (et non pas $4h_m + 2$) et les composantes du torseur cinématique. Les conditions de simulations étant inchangées par rapport à celles du paragraphe 3.5.3.1 exceptées les valeurs des paramètres suivants :

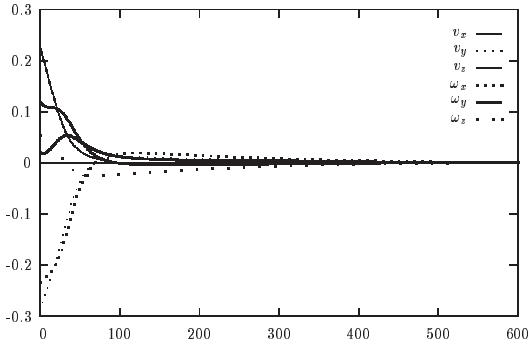
- $h_m = 2$
- $h_c = 1$
- $\lambda = 0.6$



(a) motif initial et motif désiré



(b) Erreur normalisée



(c) Composantes du torseur cinématique (m/s ou rad/s)

FIG. 3.28 – Positionnement en face d'un objet plan

- Nombre de points du contour rééchantillonnés : $N = 300$

Les résultats obtenus sont tout à fait satisfaisants malgré toutes les hypothèses émises et ce, avec un faible nombre d'harmoniques. Par ailleurs, l'autre tâche de positionnement, consistant à une orientation souhaitée à 30 degrés par rapport à l'axe x de la position finale précédente, a été également testée. Les motifs initial et désiré sont représentés sur la figure 3.29a, l'erreur normalisée et les composantes du torseur cinématique sur les figures 3.29b et 3.29c respectivement. Les conditions de simulations étant inchangées exceptée la valeur de h_c qui a été portée à 2. Une nouvelle fois la convergence est atteinte bien que toutes les profondeurs utilisées soient grossièrement approximées par $\hat{z}^*$ et que la deuxième contrainte (cf. 3.6.1) soit inefficace dans ce cas de figure. En fait, comme dans le cas de la signature polaire, seules les images courante et souhaitée sont nécessaires et pratiquement aucune information sur l'objet ou sur sa localisation par rapport à la caméra n'est requise.

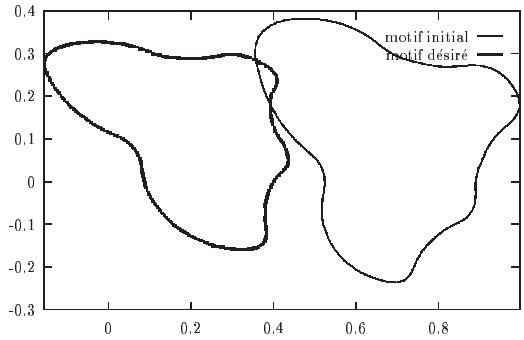
Par ailleurs, concernant une comparaison des deux méthodes et pour la tâche de positionnement en face d'un objet plan (voir les figures 3.4 et 3.28), aucune différence significative de comportement tant au niveau des vitesses de convergence que des amplitudes des composantes du torseur cinématique n'est à noter. En revanche, concernant l'autre tâche (voir les figures 3.5 et 3.29), les amplitudes des commandes sont beaucoup plus importantes dans le cas des descripteurs de Fourier. D'autre part, signalons que les deux méthodes nécessitent un nombre très voisin d'informations visuelles, $4h_m+2 = 10$ pour les descripteurs de Fourier, $2h+3 = 11$ pour la signature polaire.

Le cas d'un objet dont le contour n'est pas une fonction holomorphe n'appelle pas de remarques particulières comme le montre la figure 3.30. Pour cette simulation seul h_c a été modifié, il est passé à 2.

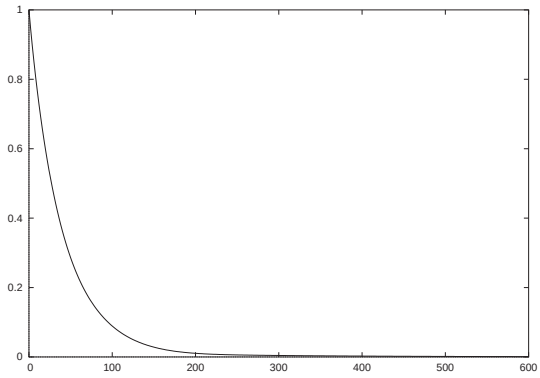
Cas d'un objet non plan

Plusieurs simulations ont été effectuées, tout d'abord sur l'objet représenté sur la figure 3.6 puis sur des objets moins profonds. Certes, certaines d'entre elles ont conduit à des cas de convergence, mais il faut surtout retenir les cas de divergence en cas de changement de la position initiale de la caméra. Il est vrai que la contrainte 2 est, dans ce cas, totalement inefficace [4]. En conséquence,

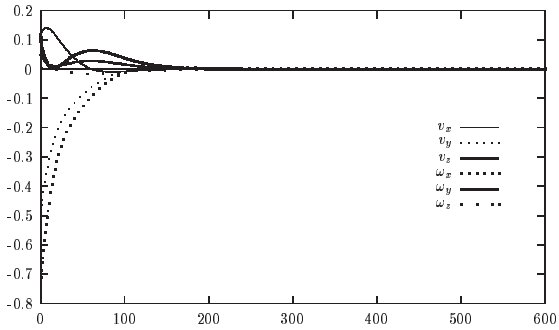
[4] Du reste, il est curieux de constater la convergence de la loi de commande lors de la réalisation d'une tâche de positionnement consistant à amener le porteur non en face de l'objet observé.



(a) motif initial et motif désiré

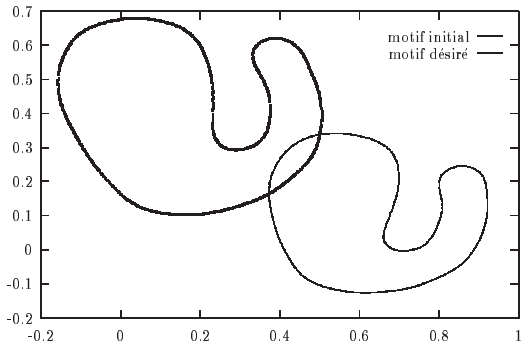


(b) Erreur normalisée

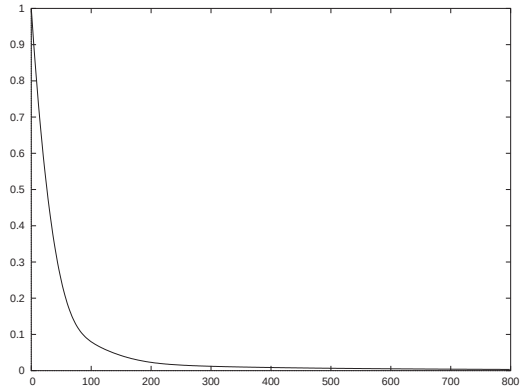


(c) Composantes du torseur cinématique (m/s ou rad/s)

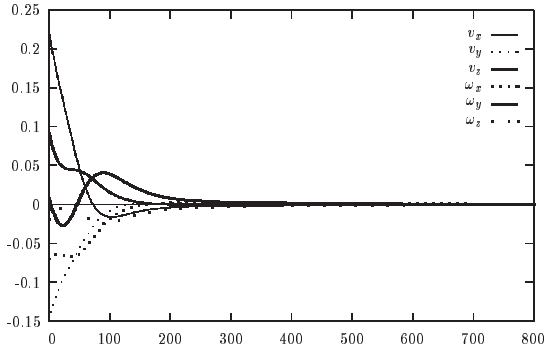
FIG. 3.29 – Positionnement à 30 degrés d'un objet plan



(a) motif initial et motif désiré



(b) Erreur normalisée



(c) Composantes du torseur cinématique (m/s ou rad/s)

FIG. 3.30 – Positionnement en face d'un objet plan dont le contour n'est pas une fonction holomorphe

nous préférons ne pas présenter de résultats ayant abouti à une convergence, ils ne seraient pas représentatifs.

3.7.4.2 Simulations bruitées

Nous considérons, dans ce paragraphe, le cas où à chaque point du contour un bruit uniforme d'amplitude maximum $\pm$ 2 pixels a été ajouté. Les figures 3.31a et 3.31b représentent respectivement le comportement du signal d'erreur normalisé et des composantes du torseur cinématique dans le cas d'un positionnement en face de l'objet, les figures 3.32a et 3.32b représentent aussi le comportement de l'erreur normalisée et des composantes du torseur cinématique mais concernant l'autre tâche. Les conditions de simulations étant inchangées par rapport à 3.7.4.1. Comme dans le cas de la signature polaire, l'introduction de bruit ne modifie pas le comportement de la loi de commande pour les raisons déjà invoquées en 3.5.3.2. En particulier, l'erreur normalisée n'est qu'imperceptiblement affectée par le bruit.

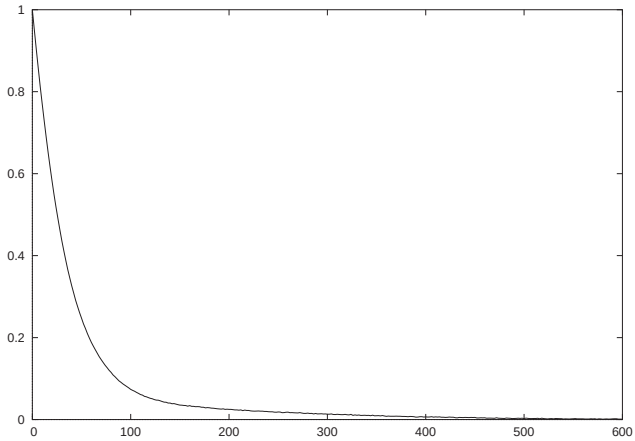
Le comportement de l'algorithme étant connu dans diverses configurations particulières, nous allons discuter maintenant de l'influence du nombre d'harmoniques et de la précision de l'estimation faites sur la loi commande.

3.7.4.3 Influence du nombre d'harmoniques sur le comportement de la loi de commande

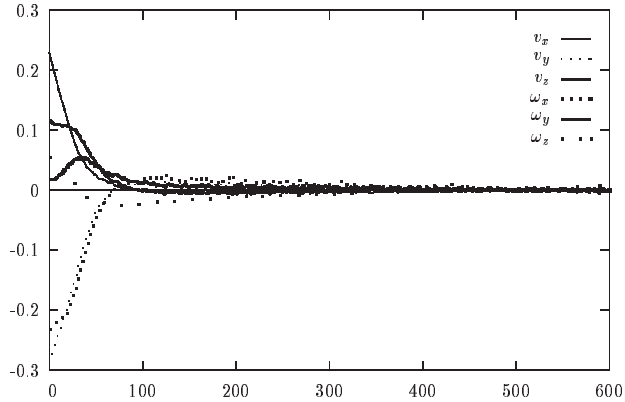
L'analyse faite au paragraphe 3.5.3.4 dans le cas de la signature polaire reste valable en considérant le nombre h_c d'harmoniques. Il doit être suffisamment important pour que la relation $\underline{s}|_{k+1} = \underline{s}|_k + \Delta t L^T_{|\underline{s}} T_c$ soit vérifiée au moins grossièrement.

De plus, comme en 3.5.3.4, le raisonnement effectué sur le rang de la matrice d'interaction implique la contrainte $4h_c + 2 \geq 6$. La plus petite valeur possible pour h_c est donc, théoriquement, $h_c = 1$.

Cette fois, bien que l'objet observé soit strictement identique à celui utilisé dans le cas de la signature polaire, une seule harmonique, voire deux, s'avère suffisante. Cependant, en procédant ainsi, on diminue l'influence du bruit, la complexité des calculs et on augmente la pertinence des informations visuelles, comme expliqué en 3.7.4.1, mais on ne prend en compte qu'un comportement simplifié de la relation liant la variation du contour dans l'image et les mouvements du porteur (c'est pourquoi, dans le cas de l'objet dont le contour n'est pas

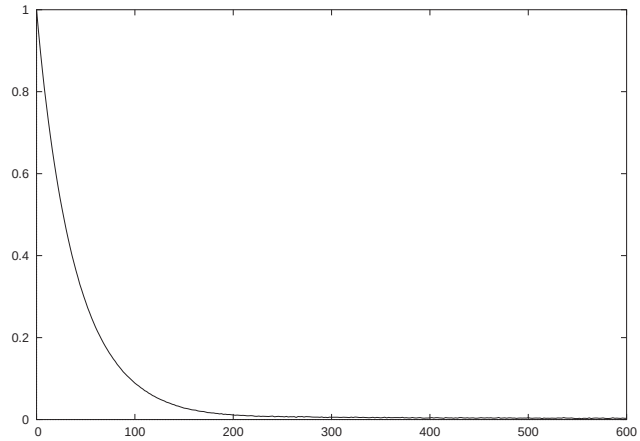


(a) Erreur normalisée

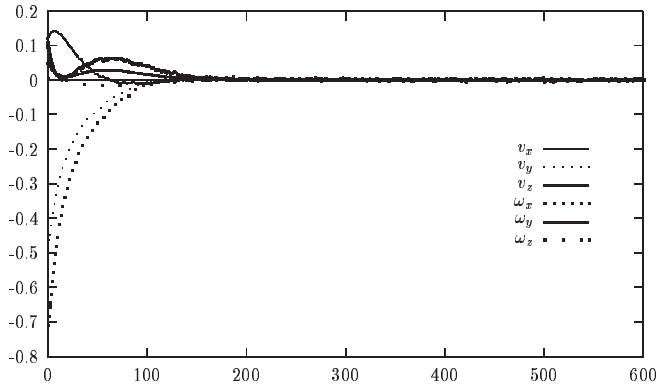


(b) Composantes du torseur cinématique (m/s ou rad/s)

FIG. 3.31 – Positionnement en face d'un objet plan — Informations visuelles bruitées —



(a) Erreur normalisée



(b) Composantes du torseur cinématique (m/s ou rad/s)

FIG. 3.32 – Positionnement à 30 degrés d'un objet plan — Informations visuelles bruitées —

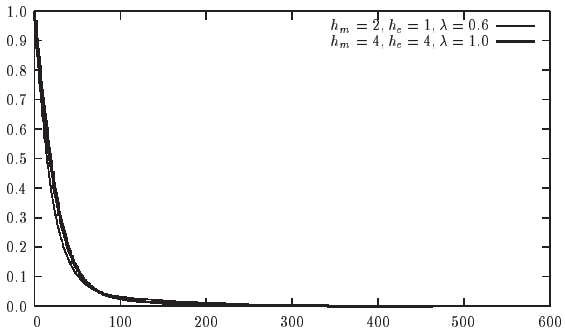
une fonction holomorphe, h_c a été porté à 2, sous peine de divergence). Il s'en suit, comme le montre clairement la figure 3.33b, des valeurs de commande élevée et, par suite, la nécessité d'utiliser des valeurs faibles de λ (dans ce cas on avait $\lambda = 0.6$). Ce comportement s'atténue, tout en assurant une vitesse de convergence équivalente (figure 3.33a), en imposant, par exemple, $h_m = h_c = 4$ comme le met en évidence la figure 3.33c. Cette raison nous a poussé aussi à fixer h_c à 2 dans le cas de la tâche amenant la caméra suivant un angle de 30 degrés par rapport à un positionnement en face de l'objet.

3.7.4.4 Influence de la précision de l'estimation de la profondeur une fois la tâche réalisée sur le comportement de la loi de commande

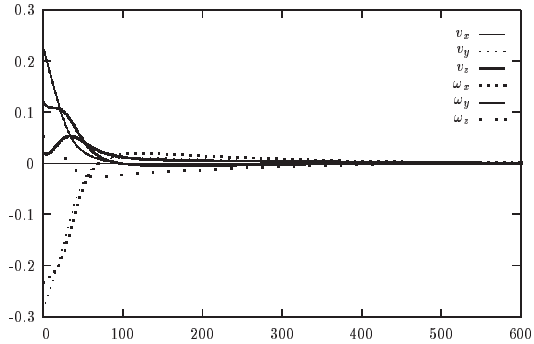
Concernant l'influence de $\hat{z}^*$, les deux méthodes ont un comportement très voisin. Elles sont toutes deux peu sensibles à une mauvaise estimation de cette valeur. Signalons plus précisément, dans le cas des descripteurs de Fourier, une convergence dans l'intervalle [10 cm, 250 cm] concernant les deux tâches de positionnement par rapport à un objet plan et pour $h_m = h_c = 4$ (conformément aux conséquences du paragraphe 3.7.4.3).

De même, la vitesse de convergence est très influencée par $\hat{z}^*$ (figure 3.34a), elle est, dans un premier temps, d'autant plus rapide que z^* est surestimé. Il s'en suit, dans ce cas, des valeurs de commande plus élevées, comme en témoigne la figure 3.35. De plus, à la vue de la figure 3.34b, comme dans le cas de la signature polaire, le chemin parcouru par la caméra est alors beaucoup plus long. Cependant, dans un second temps, la convergence devient quasiment linéaire et finalement le critère de convergence n'est pas atteint plus rapidement pour $\hat{z}^* =$ 50 cm que pour $\hat{z}^* = 250$ cm. En fait, si l'on s'intéresse au torseur cinématique (figure 3.35), on peut remarquer que, contrairement à la signature polaire, les composantes suivant les rotations ne sont que faiblement affectées par la valeur de $\hat{z}^*$ et par suite, la vitesse de convergence aussi. On assiste à un quasi découplage des composantes suivant les translations et les rotations.

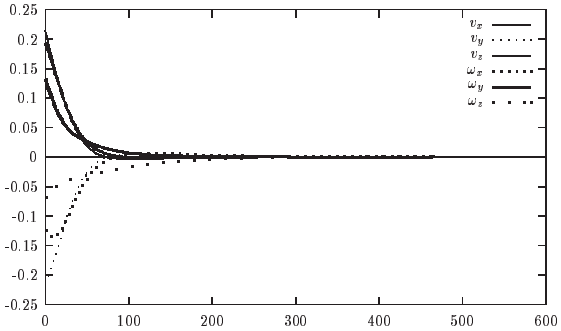
Le comportement de l'algorithme étant connu, nous allons à présent le valider sur la plate-forme expérimentale. Les tests portent sur la validation des résultats obtenus en simulation mais aussi sur l'influence de la calibration du dispositif expérimental ou d'une occlusion partielle de l'objet.



(a) Erreur normalisée

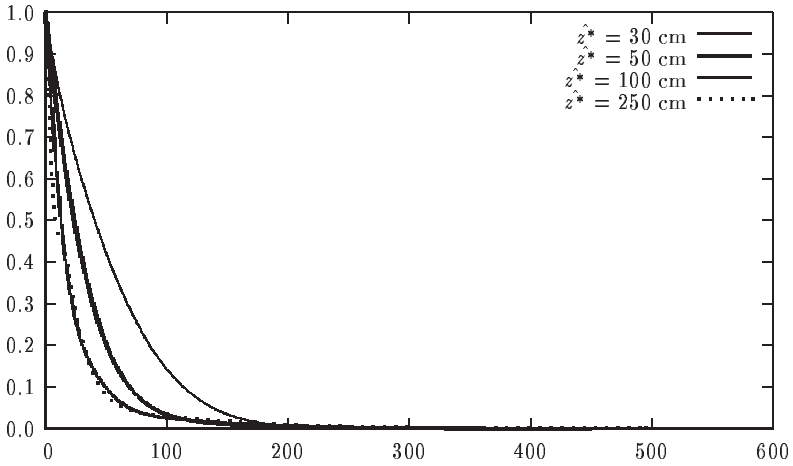


(b) Composantes du torseur cinématique (m/s ou rad/s), $h_m = 2$, $h_c = 1$, $\lambda = 0.6$

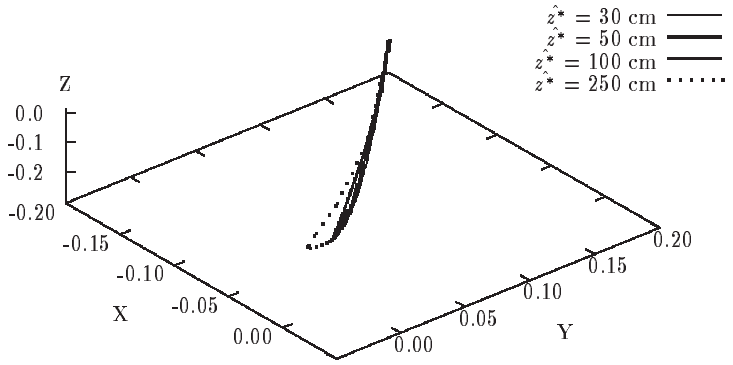


(c) Composantes du torseur cinématique (m/s ou rad/s), $h_m = 4$, $h_c = 4$, $\lambda = 1$

FIG. 3.33 – Positionnement en face d'un objet plan — Influence du nombre d'harmoniques —

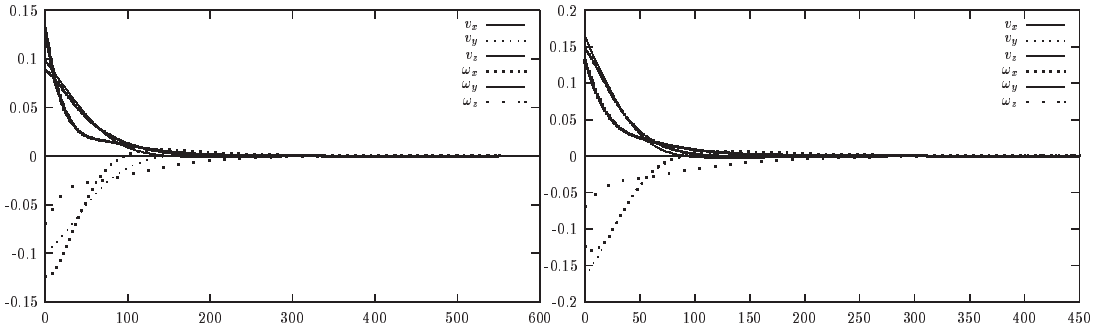


(a) Erreur normalisée



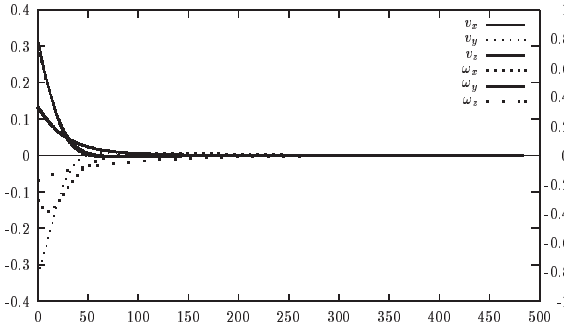
(b) Trajectoire du centre optique de la caméra dans le repère de l'objet

FIG. 3.34 – Positionnement en face d'un objet plan — Influence de $\hat{z}^*$

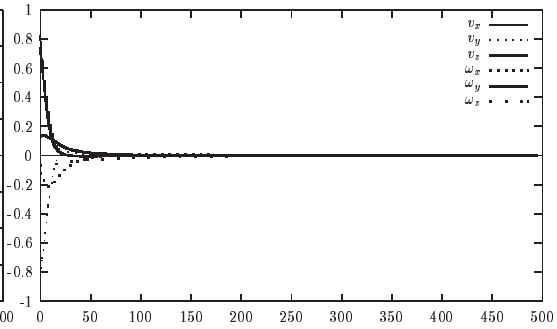


(a) $\hat{z}^* = 30$ cm

(b) $\hat{z}^* = 50$ cm



(c) $\hat{z}^* = 100$ cm



(d) $\hat{z}^* = 250$ cm

FIG. 3.35 – Positionnement en face d'un objet plan — Influence de $\hat{z}^*$ sur les composantes du torseur cinématique (m/s ou rad/s)

3.7.5 Résultats expérimentaux

Avant de présenter plus avant les résultats obtenus, précisons que le déroulement d'une expérimentation est identique à celui utilisé dans le cas de la signature polaire, en particulier la façon de vérifer la bonne réalisation de la tâche (voir le paragraphe 3.5.4.2). De même, les objets utilisés sont identiques à ceux présentés en 3.5.4.3. Toutefois, un quatrième a été ajouté pour pouvoir traiter le cas d'un contour non holomorphe, il est représenté sur la figure 3.36.

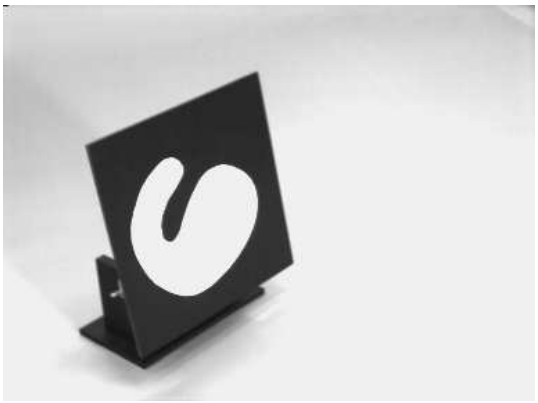


FIG. 3.36 – Objet plan dont le contour n'est pas une fonction holomorphe

3.7.5.1 Cas d'un objet plan

La première expérimentation consiste en la réalisation d'une tâche assurant un positionnement du capteur en face de l'objet plan, l'objet a. Les conditions d'expérimentations et les résultats quantitatifs sont présentés ci-dessous :

- $\Delta t = 120$ ms
- $h_m = 12$
- $h_c = 3$
- $\lambda = 0.3$
- $z^* = 68$ cm
- Nombre de points du contour rééchantillonnés : $N = 500$
- Approximation de z^* : $\hat{z^*} = 65$ cm

- La matrice d'interaction a été maintenue constante et a été calculée en $\underline{s} = \underline{s}^*$. Toutes les profondeurs ont été supposées égales à $\hat{z^*}$.
- Situation initiale :
  - Angles d'Euler (en degrés) :

$$\gamma = 16.93 \pm 0.02 \quad \beta = 16.26 \pm 0.02 \quad \alpha = -4.11 \pm 0.02$$

  - Amplitude de translation par rapport à la situation désirée (en mm) :

$$\Delta_x = -111 \quad \Delta_y = 79 \quad \Delta_z = 162$$

- Situation désirée :
  - Angles d'Euler (en degrés) :

$$\gamma = -0.19 \pm 0.01 \quad \beta = -0.00 \pm 0.01 \quad \alpha = 0.05 \pm 0.02$$

  - Translations (en mm) :

$$t_x = 1.73 \pm 0.02 \quad t_y = -0.09 \pm 0.02 \quad t_z = 611.31 \pm 0.10$$

- Situation obtenue :
  - Angles d'Euler (en degrés) :

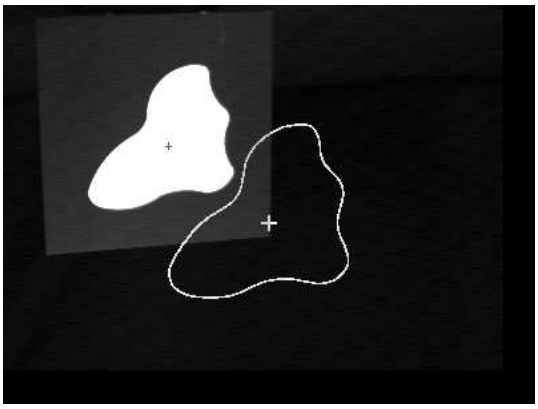
$$\gamma = 0.93 \pm 0.01 \quad \beta = -0.65 \pm 0.01 \quad \alpha = -0.00 \pm 0.02$$

  - Translations (en mm) :

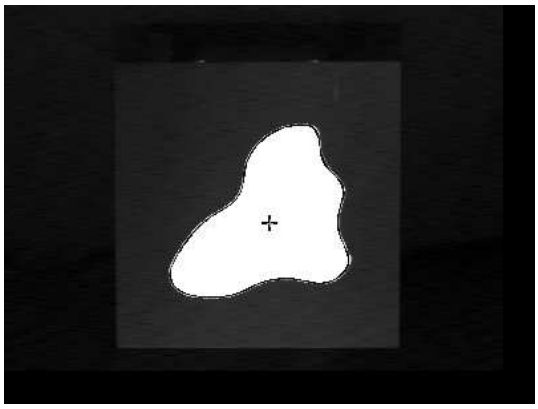
$$t_x = 0.12 \pm 0.02 \quad t_y = 2.15 \pm 0.01 \quad t_z = 609.88 \pm 0.13$$

De plus, la figure 3.37a représente le motif initial et le motif désiré, la figure 3.37b le motif obtenu, les figures 3.37c et 3.37d le comportement des composantes du torseur cinématique et de l'erreur normalisée.

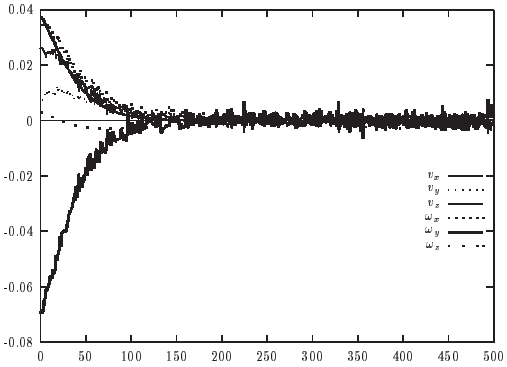
Même si les résultats obtenus restent satisfaisants, on peut toutefois remarquer qu'ils sont moins précis que dans le cas de la réalisation d'une tâche de positionnement basée sur la signature polaire. En fait, en comparant les figures 3.18 et 3.37, on peut remarquer que l'utilisation d'informations visuelles issues des descripteurs de Fourier conduit à des signaux de commande plus bruités ($\approx$ 2.5 fois). Plusieurs phénomènes pourraient expliquer cette observation :



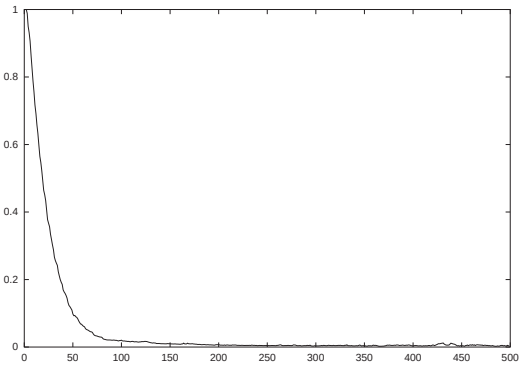
(a) Images actuelle et désirée



(b) Image finale



(c) Composantes du torseur cinématique (m/s ou rad/s)



(d) Erreur normalisée

FIG. 3.37 – Résultats expérimentaux — Objet de type a — Positionnement en face de l'objet

- Le rééchantillonnage des points du contour qui nécessairement introduit du bruit, en particulier — on le comprend bien — quand le nombre de points réels du contour est supérieur à celui imposé. On peut d'ailleurs remarquer sur la figure 3.37c que l'amplitude du bruit est plus importante en fin de mouvement, quand le nombre réel de points est élevé, qu'en début de mouvement, où il est plus faible.
- La conséquence de la non vérification de l'hypothèse $\dot{u} = 0$.
- Un moins bon conditionnement de la matrice d'interaction.

Par ailleurs, comme dans le cas de la signature polaire, des cas de minima locaux ont été rencontrés. Signalons qu'ils disparaissent quand h_m augmente.

La tâche suivante consiste en un positionnement selon une orientation différente de celle précédente. Pour ce faire, à partir de la position finale précédente, le support de l'objet est incliné. L'image correspondante constitue ainsi la nouvelle consigne visuelle. Plus précisément, les différentes situations suivantes ont été mesurées :

- Situation initiale :
  - Angles d'Euler (en degrés) :

$$\gamma = 5.51 \pm 0.01 \quad \beta = 16.11 \pm 0.02 \quad \alpha = -4.05 \pm 0.02$$

  - Amplitude de translation par rapport à la situation désirée (en mm) :

$$\Delta_x = -111 \quad \Delta_y = 79 \quad \Delta_z = 162$$

- Situation désirée :
  - Angles d'Euler (en degrés) :

$$\gamma = -11.55 \pm 0.00 \quad \beta = -0.12 \pm 0.02 \quad \alpha = -4.11 \pm 0.02$$

  - Translations (en mm) :

$$t_x = 0.97 \pm 0.02 \quad t_y = -26.26 \pm 0.02 \quad t_z = 602.78 \pm 0.10$$

- Situation obtenue :
  - Angles d'Euler (en degrés) :

$$\gamma = -13.01 \pm 0.01 \quad \beta = 0.96 \pm 0.01 \quad \alpha = -4.28 \pm 0.02$$

– Translations (en mm) :

$$t_x = 1.79 \pm 0.02 \quad t_y = -28.32 \pm 0.01 \quad t_z = 603.72 \pm 0.15$$

Le comportement de l'algorithme est rapporté sur la figure 3.38, on peut noter en particulier le bruit important présent dans la commande. Une nouvelle fois, on peut constater un positionnement moins précis. Toutefois, la convergence est obtenue bien que l'hypothèse $\dot{u} = 0$ ne soit pas vérifiée puisque la contrainte 2 est inefficace.

Les expérimentations suivantes précisent le comportement de la loi de commande vis-à-vis d'une calibration erronée, d'une imprécision sur l'estimation des profondeurs ou d'une occlusion partielle de l'objet.

Influence de la calibration du dispositif expérimental

Les erreurs introduites sont identiques à celles décrites en 3.5.4.3. Par ailleurs, les résultats donnés ci-après, concernant la position finale du porteur, sont à comparer avec les premiers donnés dans ce paragraphe.

– Angles d'Euler (en degrés) :

$$\gamma = 0.98 \pm 0.01 \quad \beta = -0.48 \pm 0.01 \quad \alpha = -0.4 \pm 0.02$$

– Translations (en mm) :

$$t_x = -0.60 \pm 0.02 \quad t_y = 1.54 \pm 0.01 \quad t_z = 607.73 \pm 0.13$$

La précision n'est pas spécialement affectée par une calibration imprécise, alors que la vitesse de convergence l'est très légèrement comme en témoigne la figure 3.39 où l'erreur normalisée d'un dispositif calibré est comparée à celle d'un dispositif non calibré.

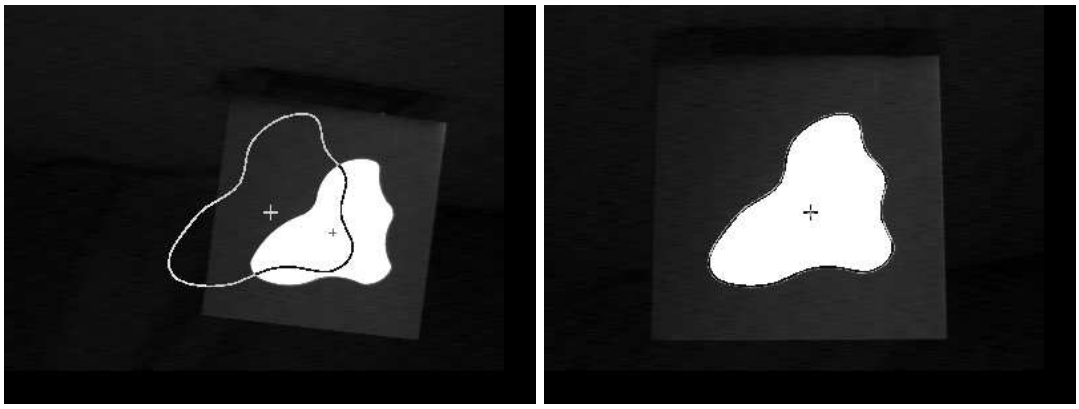
Influence de la précision de la valeur estimant les profondeurs

Pour cette expérimentation, la situation désirée est celle décrite en début de paragraphe, pour $\hat{z}^* = z^*$. Suivant les valeurs de $\hat{z}^*$, les résultats suivants ont été obtenus :

- Position finale du porteur pour $\hat{z}^* = 45$ cm :

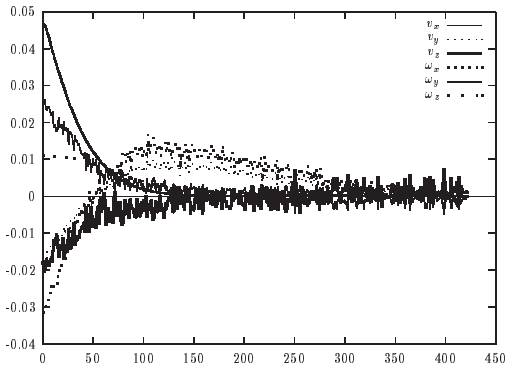
  – Angles d'Euler (en degrés) :

$$\gamma = 6.15 \pm 0.01 \quad \beta = -0.48 \pm 0.02 \quad \alpha = -0.4 \pm 0.02$$

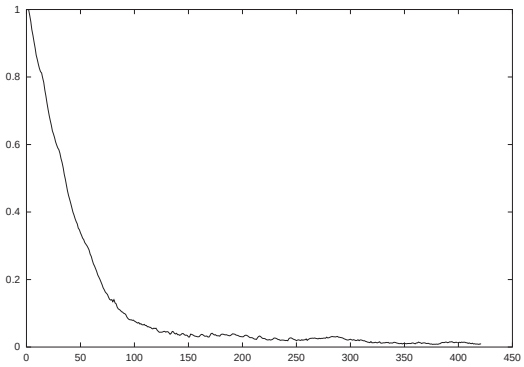


(a) Images actuelle et désirée

(b) Image finale



(c) Composantes du torseur cinématique (m/s ou rad/s)



(d) Erreur normalisée

FIG. 3.38 – Résultats expérimentaux — Objet de type a — Positionnement non en face de l'objet

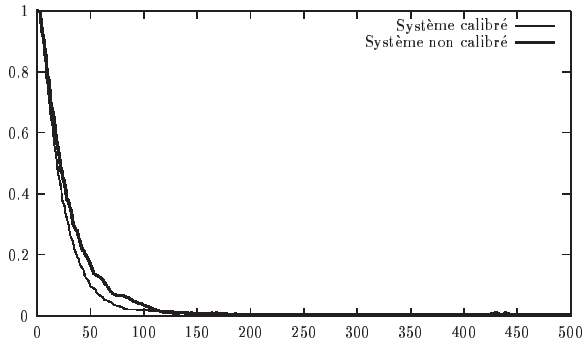


FIG. 3.39 – Résultats expérimentaux — Objet de type a — Influence de la calibration sur l'erreur normalisée —

- Translations (en mm) :

$$t_x = -0.60 \pm 0.03 \quad t_y = 10.56 \pm 0.01 \quad t_z = 607.73 \pm 0.14$$

- Position finale du porteur pour $\hat{z}^* = 95$ cm :
  - Angles d'Euler (en degrés) :

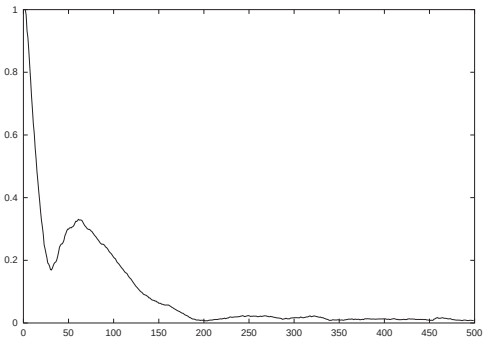
$$\gamma = 2.53 \pm 0.01 \quad \beta = -1.34 \pm 0.02 \quad \alpha = -0.07 \pm 0.02$$

  - Translations (en mm) :

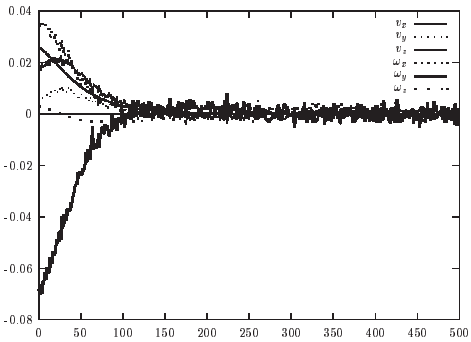
$$t_x = -2.20 \pm 0.03 \quad t_y = 4.76 \pm 0.02 \quad t_z = 609.21 \pm 0.13$$

Comme on peut le constater les résultats sont plutôt décevants, en particulier dans le cas où les profondeurs sont sous-estimées. De plus, les erreurs normalisées observées sur les figures 3.40e et surtout 3.40a ne tendent pas franchement vers une valeur nulle. En effet, comme en témoigne la trajectoire de la caméra représentée sur la figure 3.41, il s'agit de minima locaux mais pour des situations très proches les unes des autres.

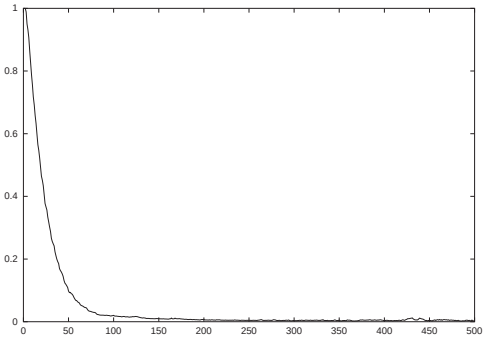
Ces minima locaux n'étant pas apparus en simulation, ces résultats nous permettent de constater que le logiciel utilisé ne modélise qu'imparfaitement la réalité. Par exemple, il ne gère pas l'augmentation (respectivement la diminution) des points du contour en cas de mouvements tendant à s'approcher (respectivement à s'éloigner) de l'objet. Toutefois, on retrouve bien le fait qu'une surestimation de $\hat{z}^*$ conduit à des signaux de commande de forte amplitude (figures 3.40b, d et f).



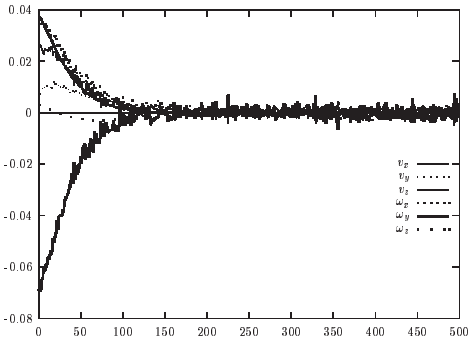
(a) Erreur normalisée pour $\hat{z}^* = 45$ cm



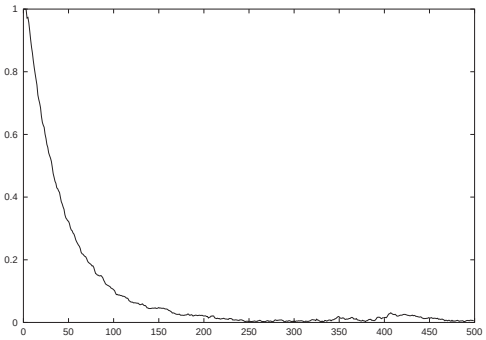
(b) Composantes du torseur cinématique (m/s ou rad/s) pour $\hat{z}^*$ = 45 cm



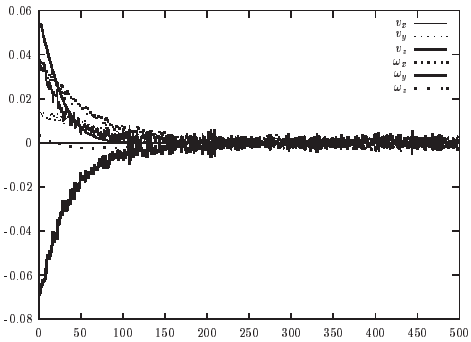
(c) Erreur normalisée pour $\hat{z}^* = 65$ cm



(d) Composantes du torseur cinématique (m/s ou rad/s) pour $\hat{z}^*$ = 65 cm



(e) Erreur normalisée pour $\hat{z}^* = 95$ cm



(f) Composantes du torseur cinématique (m/s ou rad/s) pour $\hat{z}^*$ = 95 cm

FIG. 3.40 – Résultats expérimentaux — Objet de type a — Influence de $\hat{z}^*$ sur l'erreur normalisée

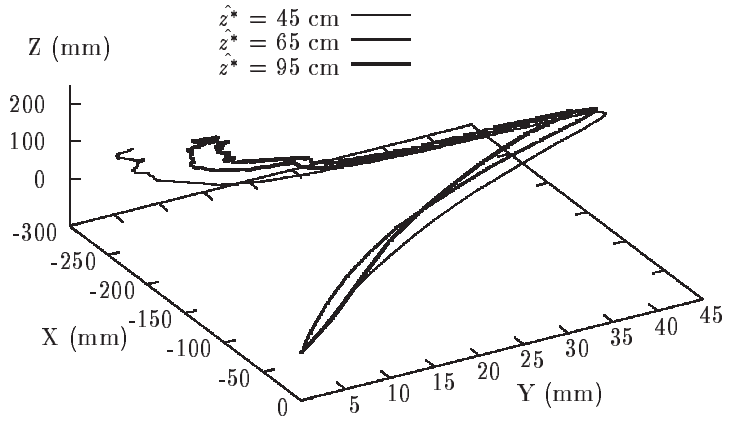


FIG. 3.41 – Résultats expérimentaux — Objet de type a — Influence de $\hat{z}^*$ sur la trajectoire de la caméra

Influence d'une occlusion partielle de l'objet

Les résultats concernant ce test sont plutôt satisfaisants comme le rapporte la figure 3.42. Par contre, notons une fois de plus des imprécisions obtenues dans la situation obtenue :

– Angles d'Euler (en degrés) :

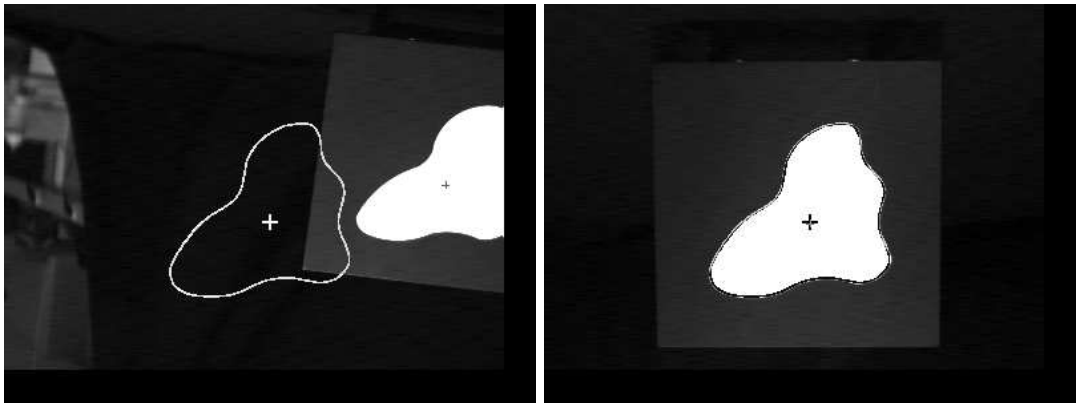
$$\gamma = -1.96 \pm 0.01 \quad \beta = -0.87 \pm 0.01 \quad \alpha = -0.24 \pm 0.03$$

– Translations (en mm) :

$$t_x = -1.48 \pm 0.03 \quad t_y = 2.1 \pm 0.02 \quad t_z = 607.71 \pm 0.12$$

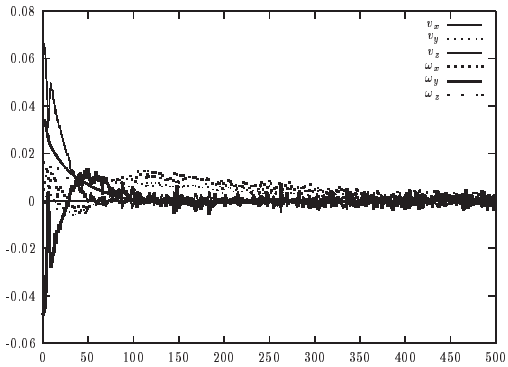
3.7.5.2 Cas d'un objet plan dont le contour est décrit par une fonction non holomorphe

Nous traitons ici le cas particulier d'un objet dont le contour dans l'image n'est pas une fonction holomorphe, cas, rappelons-le, que ne peut pas traiter la

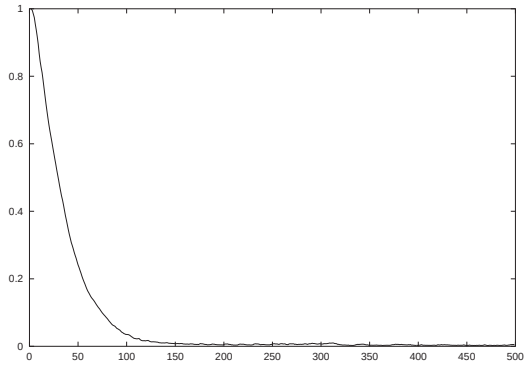


(a) Images actuelle et désirée

(b) Image finale



(c) Composantes du torseur cinématique (m/s ou rad/s)



(d) Erreur normalisée

FIG. 3.42 – Résultats expérimentaux — Objet de type a — Test d'occlusion

méthode basée sur la signature polaire. Les conditions d'expérimentations et les résultats quantitatifs sont présentés ci-dessous :

- $\Delta t = 160$ ms
- $h_m = 15$
- $h_c = 4$
- Situation initiale :
  - Angles d'Euler (en degrés) :

$$\gamma = 3.87 \pm 0.02 \quad \beta = 7.28 \pm 0.02 \quad \alpha = 10.95 \pm 0.05$$

  - Amplitude de translation par rapport à la situation désirée (en mm) :

$$\Delta_x = 260 \quad \Delta_y = 18 \quad \Delta_z = 260$$

- Situation désirée :
  - Angles d'Euler (en degrés) :

$$\gamma = -0.08 \pm 0.02 \quad \beta = -0.07 \pm 0.02 \quad \alpha = 0.01 \pm 0.03$$

  - Translations (en mm) :

$$t_x = 0.93 \pm 0.05 \quad t_y = -0.36 \pm 0.01 \quad t_z = 611.05 \pm 0.22$$

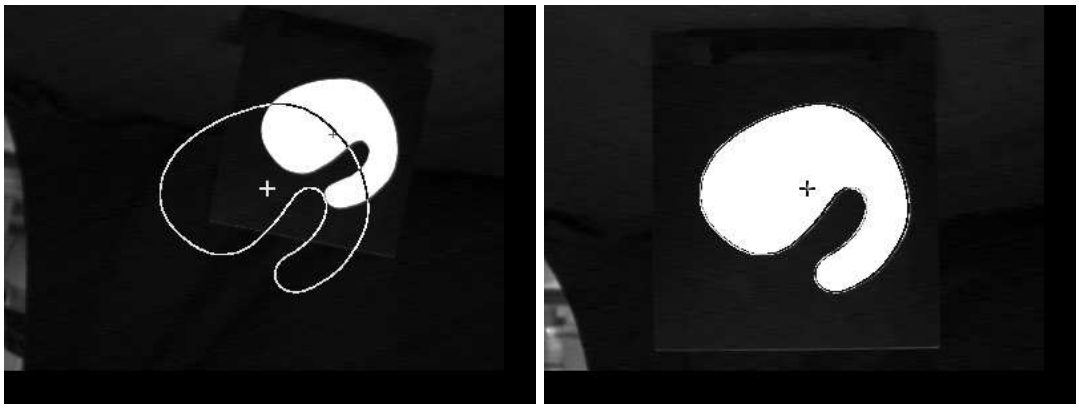
- Situation obtenue :
  - Angles d'Euler (en degrés) :

$$\gamma = -0.80 \pm 0.01 \quad \beta = 0.57 \pm 0.01 \quad \alpha = 0.27 \pm 0.02$$

  - Translations (en mm) :

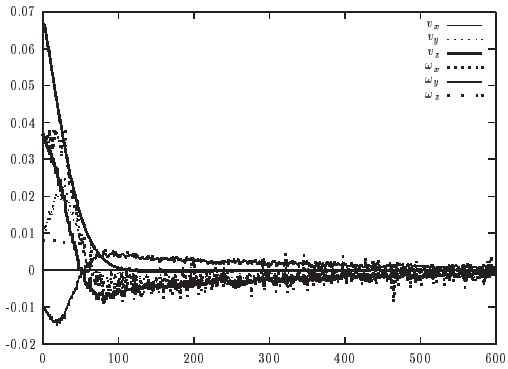
$$t_x = 2.75 \pm 0.02 \quad t_y = -1.86 \pm 0.01 \quad t_z = 610.57 \pm 0.11$$

De plus, les figures 3.43c et 3.43d rapportent respectivement les comportements des composantes du torseur cinématique et de l'erreur normalisée relative aux informations visuelles. La figure 3.43a représente le motif initial et le motif désiré, la figure 3.43b le motif obtenu. La réalisation de cette tâche n'appelle pas de remarques particulières.

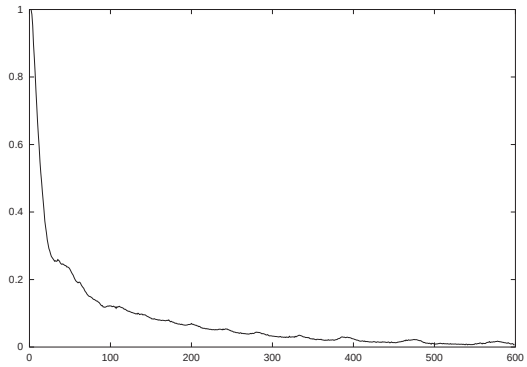


(a) Images actuelle et désirée

(b) Image finale



(c) Composantes du torseur cinématique (m/s ou rad/s)



(d) Erreur normalisée

FIG. 3.43 – Résultats expérimentaux — Objet dont le contour n'est pas représenté par une fonction holomorphe — Positionnement en face de l'objet

3.7.5.3 Cas d'un objet non plan

Comme en simulation, des cas de convergence ont été obtenus, en particulier sur l'objet b, mais, en règle générale, l'algorithme s'est avéré très peu robuste vis-à-vis de ce type d'objet. Des cas fréquents de minima locaux ou de convergence uniquement pour certaines positions initiales ont été rencontrés. La non vérification de la contrainte 2 est sans doute une des raisons majeures de cet échec.

3.7.6 Conclusion

Après avoir proposé une méthode générale de calcul de la matrice d'interaction dans le cas d'informations visuelles issues d'une représentation sous forme paramétrique du contour de l'image de l'objet observé, nous l'avons appliqué au cas des descripteurs de Fourier. Ce calcul repose sur l'hypothèse de validité de la relation $\dot{u} = 0$ où u représente la variable de parcours du contour. Pour rendre cette hypothèse légitime, nous avons introduit deux contraintes sur la façon d'extraire les informations visuelles. Cependant, les résultats obtenus en simulation et sur site expérimental ont montré leur inefficacité dans le cas d'objets non plans. De plus, en ce qui concerne la robustesse de l'algorithme, on a pu noter une sensibilité par rapport à une imprécision sur la valeur de l'estimation des profondeurs une fois la tâche réalisée. Par contre, le cas d'une occlusion partielle a été correctement traité, tout comme celui d'une mauvaise calibration du porteur et de la caméra. De plus, même si dans le cas d'objets plans, la précision du positionnement final du porteur est moins bonne que dans le cas de l'approche basée sur la signature polaire, les résultats sont globalement bons.

Par ailleurs, la convergence de la loi de commande ne requiert aucune hypothèse sur les dimensions de l'objet. En effet, seules sont nécessaires les images courante et souhaitée et une approximation, somme toute assez grossière (à environ 5 cm près), de la distance de la caméra à l'objet une fois la tâche de positionnement réalisée.

3.8 Conclusion

Finalement, nous avons proposé dans ce chapitre deux méthodes permettant de réaliser une tâche de positionnement par rapport à un objet de forme complexe. Une comparaison des ces deux méthodes conclue à un net avantage de

celle basée sur une description polaire du contour. Elle permet d'appréhender des objets non plans et conduit, d'une manière générale, à des positionnements plus précis. D'autre part, elle est plus robuste vis-à-vis d'une mauvaise estimation de la profondeur une fois la tâche réalisée. Cependant, rappelons, qu'à ce jour, la méthode basée sur les descripteurs de Fourier est l'unique méthode permettant d'appréhender des contours non holomorphes. D'autre part, comme on le verra dans le chapitre de conclusion de ce document, elle offre des perspectives de recherche très intéressantes.

À présent que l'on connaît le comportement des deux types de méthodes développées dans ce chapitre et donc, de l'approche asservissement visuel basé sur le contour dans l'image de l'objet observé, on peut vouloir la comparer à une approche basée sur des informations visuelles de type point caractéristique.

Tout d'abord, on peut souligner la faible influence du bruit sur la loi de commande. En effet, le processus d'extraction d'un point caractéristique, au même titre qu'un point du contour, fournit une information bruitée qui, dans le cas d'un asservissement visuel basé sur des primitives points est à utiliser telle quelle, alors que dans le cas d'une approche contour la méthode d'identification du modèle permet de limiter fortement son influence.

D'autre part, nous avons vu aussi que l'approche contour était peu sensible à une occlusion partielle de l'objet. Au contraire, dans le cas de primitives points, elle peut être dramatique. Une alternative pourrait consister à envisager une loi de commande utilisant des informations visuelles redondantes comme dans [Feddema 89a], toutefois, on se heurterait alors à un autre problème : l'appariement des points d'une image à l'autre. En effet, ce problème est compliqué, en particulier lorsque la mise en correspondance doit être effectuée sur des images très différentes comme cela peut être le cas entre l'image consigne et l'image initiale. Rappelons que ce problème est évité dans une approche contour.

De plus, les détecteurs de *points d'intérêts* (que nous détaillerons par ailleurs en 4.3.3) restent sensibles à des changements de point de vue ou de facteur d'échelle dans le sens où un point détecté dans une première image peut ne pas l'être dans une seconde ou de façon insuffisamment précise (voir dans [Schmid 96] une évaluation comparative de ce type d'algorithme). Cela peut être problématique en asservissement visuel. Par contre, et nous en terminerons là, la détection du contour est moins critique. En effet, une détection sommaire est possible puisque l'on ne cherche pas une modélisation précise de ce contour, que ce soit dans l'approche polaire ou paramétrique.

Chapitre 4

Positionnement par rapport à un objet plan de forme inconnue

Dans le chapitre précédent, nous nous sommes confrontés à la réalisation d'une tâche de positionnement par rapport à des objets de forme complexe par asservissement visuel 2D. Le mouvement du capteur est tel qu'il tend à faire coïncider l'image courante de l'objet avec une image désirée de ce même objet. On émet ainsi implicitement l'hypothèse que l'image courante puisse effectivement coïncider avec l'image désirée, acquise à un instant qui peut être très différent de l'instant présent. On suppose, en particulier, que l'objet ne se soit pas déformé depuis l'acquisition de l'image désirée ou, dans le cas où la consigne visuelle est obtenue à l'aide d'un modèle, qu'il ne se déformera pas. Une deuxième hypothèse suppose que la consigne visuelle soit connue. C'est-à-dire, que l'on puisse décrire avec précision l'image de l'objet une fois la tâche réalisée. Comme on l'a vu en introduction de ce document, ce n'est pas toujours possible. Nous avions cité en exemple le cas d'objets pour lesquels une tâche de positionnement est souhaitée et que l'on rencontre dans le domaine des industries agro-alimentaires. De même, on aurait pu citer les domaines de l'agriculture, de l'horticulture ou de l'industrie forestière. Ces objets, plantations, fleurs, arbres ou produits alimentaires appartiennent tous à une même classe, toutefois, ils sont tous différents à l'intérieur de leur classe respective et, de ce fait, il existe autant de consignes visuelles que d'objets de la classe. Dans ce contexte, une unique consigne visuelle n'est pas exprimable, on doit donc la supposer inconnue.

Nous nous proposons, dans ce chapitre, de traiter ce cas de figure pour une tâche particulière de positionnement consistant à amener la caméra en face de l'objet observé. Les dimensions de cet objet, comme sa forme, sont supposées

inconnues. Par contre, nous le supposons plan et immobile. L'idée directrice est de maximiser la surface de l'image de l'objet afin que, sous certaines hypothèses, cette surface soit effectivement maximum quand la caméra est en face de l'objet. Pour ce faire, nous traitons dans un premier temps le cas plus simple d'un objet filiforme en maximisant la taille du segment issu de la projection sur le capteur de cet objet. Nous montrons, ensuite, comment l'approche retenue peut se généraliser à un objet présentant trois points caractéristiques, puis à un objet quelconque. En fin de chapitre, la notion de points caractéristiques sera approfondie et des résultats d'expérimentations, y compris en environnement non calibré, seront présentés. Mais, tout d'abord, rappelons le formalisme de redondance de la fonction de tâche qui nous sera utile tout au long de ce chapitre.

4.1 Le formalisme de redondance de la fonction de tâche

L'approche fonction de tâche, déjà introduite en 3.1, permet d'appréhender la réalisation d'une tâche référencée vision $\underline{e}_1(\underline{r}, t)$ comme la régulation à zéro de la fonction de tâche suivante (en conservant les notations du chapitre 2) :

$$\underline{e}_1(\underline{r}, t) = \widehat{L_{\underline{s}}^T}^+ (\underline{s}(\underline{r}, t) - \underline{s}^*) \tag{4.1}$$

Cependant, quand les degrés de liberté ne sont pas tous contraints par la réalisation de $\underline{e}_1$, on peut, grâce au *formalisme de redondance* décrit dans [Samson 91] et [Samson 88], faire cohabiter une autre tâche dite *secondaire*, $\underline{e}_1$ étant alors la tâche la plus prioritaire dite *principale*. Certaines applications utilisent à profit cette façon de faire [Coste-Manière 95, Santos 95] et [Marchand 96].

La tâche secondaire est représentée par un coût secondaire h_s à minimiser de gradient $\underline{g}_s = \frac{\partial h_s}{\partial \underline{r}}$. Si $\widehat{L_{\underline{s}}^T}$ est de rang plein, Samson a montré dans [Samson 91] et [Samson 88] qu'une fonction de tâche réalisant $\underline{e}_1 = 0$ sous la contrainte de minimisation de h_s s'écrivait :

$$\underline{e} = \widehat{L_{\underline{s}}^T}^+ (\underline{s} - \underline{s}^*) + \alpha \left(\mathbb{I}_n - \widehat{L_{\underline{s}}^T}^+ \widehat{L_{\underline{s}}^T} \right) \underline{g}_s^T \tag{4.2}$$

où :

- α est un scalaire positif ;
- $\mathbb{I}_n - \widehat{L_{\underline{s}}^T}^+ \widehat{L_{\underline{s}}^T}$ est un opérateur de projection dans le noyau de $\widehat{L_{\underline{s}}^T}$.

Le torseur cinématique étant toujours obtenu par :

$$T_c = -\lambda \underline{e} \tag{4.3}$$

Ces principaux rappels étant effectués, nous allons, dans le paragraphe suivant, spécifier plus avant le problème qui nous intéresse dans le cas d'un objet simple, un objet filiforme, et pour la tâche consistant à positionner la caméra en face de cet objet.

4.2 Étude d'un objet simple : le segment

4.2.1 Spécification de la tâche

On se propose d'étudier le positionnement d'une caméra par rapport à un objet filiforme de longueur L à l'aide d'asservissements visuels 2D. La dimension de cet objet est supposée inconnue tout comme sa position dans l'espace.

Dès à présent, nous tenons à signaler que nous ne voulons pas estimer la position et l'orientation 3D de l'objet observé, par exemple par vision dynamique. En effet, comme évoqué en 2.1.3, ces techniques permettent à partir des mesures du mouvement de la caméra et du mouvement dans l'image une reconstruction 3D de l'objet. Toutefois, elles sont sensibles aux erreurs de calibration du porteur comme du capteur et produisent habituellement des estimations trop peu précises pour le problème que l'on se propose de résoudre (voir par exemple [Xie 89] pour des résultats de précision concernant la reconstruction 3D).

L'objet observé est caractérisé par un segment d'extrémités $[m_1, m_2]$ tels que $\underline{Om_1} = (x_1, y_1, z_1)^T$ et $\underline{Om_2} = (x_2, y_2, z_2)^T$, coordonnées exprimées dans le repère de la caméra. Par convention ce repère est tel que son centre O et l'axe des z coïncident respectivement avec le centre optique et avec l'axe optique de la caméra (cf. figure 4.1).

Les points m_1 et m_2 se projettent sur le plan image par transformation perspective respectivement en M_1 et M_2 tel que $\underline{OM_1} = (X_1, Y_1, f)^T$ et $\underline{OM_2} = (X_2, Y_2, f)^T$ suivant :

$$\underline{OM} = \frac{f}{z}\,\underline{Om} \tag{4.4}$$

avec f la distance focale que l'on supposera, comme au chapitre 3, égale à l'unité.

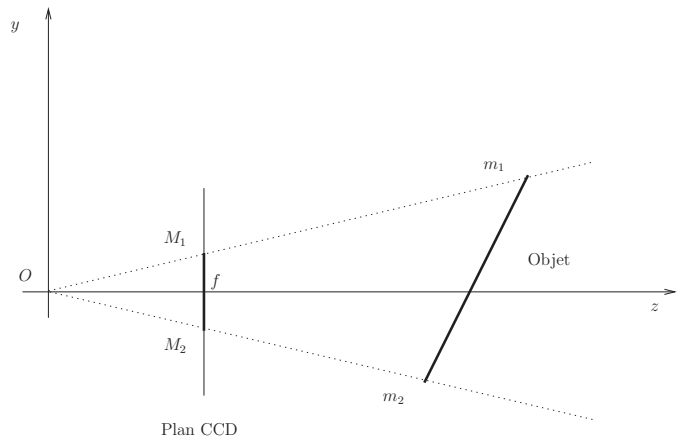


FIG. 4.1 – Projection de l'objet sur le plan image de la caméra

L'image obtenue est caractérisée par le segment $[M_1, M_2]$ de longueur l. On dira que la tâche est réalisée quand l'axe optique de la caméra est orthogonal à $[m_1, m_2]$ et l'image centrée par rapport à l'axe y. On aura alors $z_1 = z_2$ et la caméra sera en face de l'objet considéré.

Bien que l'objet soit très simple, réaliser la tâche de positionnement choisie n'est pas trivial. En effet, supposons la dimension L de l'objet connue et l'image centrée. On peut écrire :

$$\begin{cases} Y_1 &= \dfrac{l}{2} \\ Y_2 &= -\dfrac{l}{2} \end{cases} \tag{4.5}$$

où la consigne visuelle l s'obtient en considérant la caméra en face de l'objet :

$$l = \frac{L}{z^*} \tag{4.6}$$

soit :

$$\begin{cases} Y_1^* &= \dfrac{1}{2}\dfrac{L}{z^*} \\ Y_2^* &= -\dfrac{1}{2}\dfrac{L}{z^*} \end{cases} \tag{4.7}$$

on en déduit les composantes en y dans le repère de la caméra :

$$\begin{cases} y_1 &= \dfrac{L}{2}\dfrac{z_1}{z^*} \\ y_2 &= -\dfrac{L}{2}\dfrac{z_2}{z^*} \end{cases} \tag{4.8}$$

Par ailleurs, la contrainte de dimension de l'objet impose :

$$(y_1 - y_2)^2 + (z_1 - z_2)^2 = L^2 \tag{4.9}$$

d'où, en utilisant (4.8) dans (4.9), il vient :

$$\left(\frac{L}{2z^*}\right)^2 (z_1 + z_2)^2 + (z_1 - z_2)^2 - L^2 = 0 \tag{4.10}$$

Par conséquent, une infinité de position du segment permet de centrer l'image sans pour autant réaliser la tâche. Toutefois, les équations ci-dessus nous indiquent une façon possible de réaliser la tâche de positionnement : centrer l'image puis, scruter d'autres positions (celles vérifiant (4.10)) selon une loi de commande restant à définir mais maintenant l'image centrée. On rencontrera alors la position recherchée assurant $z_1 = z_2 = z^*$. Signalons en outre que, curieusement, la connaissance éventuelle de la dimension de l'objet n'apporte rien.

La bonne réalisation de la tâche se décompose donc en deux parties :

- centrer l'objet dans l'image ; il s'agit en fait de choisir judicieusement les informations visuelles.
- rendre égaux z_1 et z_2. Là, la démarche est plus délicate. *A priori* deux voies sont possibles : (i) déterminer une fonction de tâche secondaire qui s'annule en $z_1 = z_2$, (ii) détecter directement cette égalité grâce à des informations extraites de l'image. Dans les deux cas de figure, les mouvements devront appartenir au noyau de $\widehat{L_{\underline{s}}^T}$ afin de maintenir l'image centrée.

Nous allons détailler dans les paragraphes suivants les différentes voies possibles.

4.2.2 Centrage de l'objet dans l'image

4.2.2.1 Utilisation de quatre informations visuelles

L'approche la plus immédiate est de choisir le vecteur $\underline{s}$ tel que :

$$\underline{s} = (X_1, X_2, Y_1, Y_2)^T \tag{4.11}$$

et de chercher à atteindre le motif $\underline{s}^*$ donné par :

$$\underline{s}^* = (0, 0, Y, -Y)^T \tag{4.12}$$

avec $Y = l_c/2$ où l_c est la taille désirée du segment $[M_1M_2]$ dans l'image.

En se rappelant que la matrice d'interaction dans le cas d'un point de coordonnées (X, Y) s'écrit :

$$L^T_{|\underline{s}} = \begin{pmatrix} -1/z & 0 & X/z & XY & -1-X^2 & Y \\ 0 & -1/z & Y/z & 1+Y^2 & -XY & -X \end{pmatrix} \tag{4.13}$$

on obtient :

$$L^T_{|\underline{s}=\underline{s}^*} = \begin{pmatrix} -1/z_1 & 0 & 0 & 0 & -1 & Y \\ -1/z_2 & 0 & 0 & 0 & -1 & -Y \\ 0 & -1/z_1 & Y/z_1 & 1+Y^2 & 0 & 0 \\ 0 & -1/z_2 & -Y/z_2 & 1+Y^2 & 0 & 0 \end{pmatrix} \tag{4.14}$$

Les profondeurs z_1 et z_2 étant inconnues, on fait dans $L^T_{|\underline{s}=\underline{s}^*}$ l'approximation $z_1 = z_2 = z$ où z est une approximation de z^*, profondeur atteinte une fois la tâche réalisée.

On utilise ensuite (4.2) et (4.3) avec $\alpha = 0$ pour réaliser le centrage souhaité.

4.2.2.2 Utilisation de trois informations visuelles

Une autre façon d'opérer est de choisir :

$$\underline{s} = (X_1, X_2, Y_1 + Y_2)^T \tag{4.15}$$

et :

$$\underline{s}^* = (0, 0, 0)^T \tag{4.16}$$

En utilisant la propriété qui lie un paramétrage $\underline{P\prime}$ à un autre paramétrage $\underline{P}$ relatif à la même primitive géométrique [Chaumette 90], à savoir :

$$L^T(\underline{P\prime}) = \frac{\partial \underline{P\prime}}{\partial \underline{P}} L^T(\underline{P}), \tag{4.17}$$

on obtient la matrice d'interaction souhaitée :

$$L^T_{\substack{X_i=0 \\ z_i=z}} = \begin{pmatrix} -\frac{1}{z} & 0 & 0 & 0 & -1 & Y_1 \\ -\frac{1}{z} & 0 & 0 & 0 & -1 & Y_2 \\ 0 & -\frac{2}{z} & \frac{Y_1+Y_2}{z} & 2+{Y_1}^2+{Y_2}^2 & 0 & 0 \end{pmatrix} \tag{4.18}$$

Si l'on veut comme en 4.2.2.1 imposer une taille l_c à atteindre dans l'image, on introduit un coût secondaire h_s à minimiser :

$$h_s = \frac{1}{2}\left(l - l_c\right)^2 \tag{4.19}$$

La loi de commande décrite par (4.2) nécessite le calcul de $\underline{g}_s$. On a :

$$\underline{g}_s \left|_{\substack{x_i=0 \\ z_i=z}}\right. = (l - l_c)\left(L_{Y_1}\left|_{\substack{x_1=0 \\ z_1=z}}\right. - L_{Y_2}\left|_{\substack{x_2=0 \\ z_2=z}}\right.\right) \tag{4.20}$$

avec :

$$L_{Y_i}\left|_{\substack{x_i=0 \\ z_i=z}}\right. = \begin{pmatrix} 0 & -1/z & Y_i/z & 1+Y_i^2 & 0 & 0 \end{pmatrix}^T \tag{4.21}$$

Simplifications faites, il vient :

$$\underline{g}_s \left|_{\substack{x_i=0 \\ z_i=z}}\right. = l(l - l_c)\begin{pmatrix} 0 & 0 & 1/z & Y_1+Y_2 & 0 & 0 \end{pmatrix}^T \tag{4.22}$$

Le centrage et la tâche secondaire sont ensuite réalisés grâce à (4.2) et (4.3).

Grâce aux méthodes de centrage que nous venons de détailler, nous allons pouvoir, dans les deux cas, élaborer les lois de commande permettant la bonne réalisation de la tâche. À ce stade, il s'agit maintenant d'obtenir $z_1 = z_2$.

4.2.3 Détermination d'une fonction de tâche secondaire

La première étape étant réalisée, on va chercher une fonction de tâche secondaire e_2 qui permet $z_1 = z_2$. Pour cela, on se place dans les conditions du paragraphe 4.2.2.1 où on utilise quatre informations visuelles. Dans ce cas de figure, les seuls mouvements qui maintiennent l'image centrée et constante sont tels qu'ils laissent les informations visuelles invariantes : ce sont donc des mouvements qui appartiennent au noyau de la matrice d'interaction $L^T_{|X_i=0}$ (l'image étant centrée, on a $\underline{s} = \underline{s}^*$). Cependant, les profondeurs étant inconnues, on doit se contenter du noyau de $L^T_{|X_i=0,z_i=z}$. De plus, parmi ces mouvements, seuls nous intéressent ceux qui génèrent un mouvement dans le plan (yOz). Ils sont de la forme :

$$T_s = \left(0, z\omega_x\left(1+Y^2\right), 0, \omega_x, 0, 0\right)^T \tag{4.23}$$

la loi de commande est donc du type :

$$T_c = -\lambda L^{T^+}_{\substack{X_i=0 \\ z_i=z}}\left(\underline{s} - \underline{s}^*\right) + T_s \tag{4.24}$$

où dans (4.23), et donc dans (4.24), ω_x est à déterminer en fonction d'une tâche secondaire e_2.

Connaissant T_c, on en déduit la loi d'évolution de $\underline{s}$:

$$\underline{s}|_{k+1} = \underline{s}|_k + \Delta t L^T T_c \tag{4.25}$$

où Δt est la période d'acquisition des images.

On impose à la fonction de tâche secondaire e_2 recherchée d'être fonction des informations visuelles disponibles et de s'écrire de la façon la plus simple possible :

$$e_2\left(\underline{\epsilon}\right) = \alpha\epsilon_1 + \beta\epsilon_2 + \gamma \tag{4.26}$$

dans lequel on a introduit le vecteur $\underline{\epsilon} = (\epsilon_1, \epsilon_2)$ avec :

$$\left\{ \begin{array}{lcl} \epsilon_1 & = & Y_1 - Y \\ \epsilon_2 & = & Y_2 + Y \end{array} \right. \tag{4.27}$$

On souhaite que cette fonction s'annule en $z_1 = z_2 = z^*$, on impose donc la contrainte suivante :

$$e_2|_{k+1, z_1 = z_2 = z^*} = 0 \tag{4.28}$$

Grâce à (4.26) puis (4.25), on exprime (4.28) uniquement en fonction de grandeurs connues à l'instant k. On en déduit alors γ en fonction de α et de β. En effet, on peut montrer que γ se met sous la forme :

$$\gamma(\alpha, \beta) = \frac{b_1(\alpha, \beta) z^* + b_2(\alpha, \beta)}{b_3(\alpha, \beta) z^*} \tag{4.29}$$

où b_1, b_2 et b_3 ont une forme peu sympathique qu'il est inutile de présenter ici.

Pour éviter que $\gamma(\alpha, \beta)$ dépende de z^* qui est inconnu, on impose $b_2 = 0$. Cette nouvelle contrainte fixe le choix du coefficient β, fonction de grandeurs connues à l'instant k et de α :

$$\beta(\alpha) = \alpha \frac{Y_1^2 - Y_1 Y_2 + \beta_1 Y_1 + \beta_2 Y_2 + \beta_3}{Y_2^2 - Y_1 Y_2 - \beta_2 Y_1 - \beta_1 Y_2 - \beta_3} \tag{4.30}$$

où on a posé :

$$\left\{ \begin{array}{lcl} \beta_1 & = & -Y \dfrac{1 + 2z^2 \left(1 + Y^2\right)^2}{1 + z^2 \left(1 + Y^2\right)^2} \\ \beta_2 & = & \dfrac{Y}{1 + z^2 \left(1 + Y^2\right)^2} \\ \beta_3 & = & 2Y \left(1 + Y^2\right) \dfrac{\omega_x}{\lambda} \end{array} \right. \tag{4.31}$$

$\beta(\alpha)$ étant connu, on peut exprimer maintenant $\gamma(\alpha)$ grâce à (4.29) :

$$\gamma(\alpha) = \alpha\gamma_0 \frac{Y_1 + Y_2 + \gamma_1}{Y_2^2 - Y_1Y_2 - \beta_2 Y_1 - \beta_1 Y_2 - \beta_3} P\left(Y_1, Y_2\right) \tag{4.32}$$

avec :

$$\begin{cases} \gamma_0 & = & -\dfrac{\lambda\Delta t z^2\left(1+Y^2\right)}{2\left(1+z^2\left(1+Y^2\right)^2\right)} \\ \gamma_1 & = & -\dfrac{2\omega_x\left(1+z^2\left(1+Y^2\right)^2\right)}{\lambda z^2\left(1+Y^2\right)} \end{cases} \tag{4.33}$$

et :

$$\begin{aligned} P\left(Y_1, Y_2\right) &= Y_1^3 Y_2 + Y_2^3 Y_1 - 2Y_1^2 Y_2^2 + \beta_2\left(Y_1^3 - Y_2^3\right) \\ &+\beta_1\left(Y_1^2 Y_2 - Y_2^2 Y_1\right) + \gamma_2 Y_1^2 - \beta_3 Y_2^2 + 2Y_1 Y_2 + \gamma_3\left(Y_1 - Y_2\right) + \gamma_4 \end{aligned} \tag{4.34}$$

dans lequel on a posé :

$$\begin{cases} \gamma_2 & = & 2Y\left(1+Y^2\right)\dfrac{\omega_x}{\lambda} - 1 \\ \gamma_3 & = & 2Y\dfrac{1+Y^2+\lambda\Delta t}{\lambda\Delta t} \\ \gamma_4 & = & -4Y^2\dfrac{1+Y^2}{\lambda\Delta t} \end{cases} \tag{4.35}$$

L'utilisation de (4.30) et (4.32) dans (4.26) conduit à :

$$e_2|_{k+1} = \alpha e_2(0)\frac{z_1 - z_2}{z_1 z_2}\left(Y_1^2 - Y_1Y_2 + \beta_1 Y_1 + \beta_2 Y_2 + \beta_3\right) \tag{4.36}$$

avec :

$$e_2(0) = \frac{\lambda\Delta t z}{2Y} \tag{4.37}$$

En imposant :

$$\alpha = \frac{1}{e_2(0)\left(Y_1^2 - Y_1Y_2 + \beta_1 Y_1 + \beta_2 Y_2 + \beta_3\right)} \tag{4.38}$$

on obtient finalement :

$$e_2|_{k+1} = \frac{z_1 - z_2}{z_1 z_2} \tag{4.39}$$

Cette fonction convient parfaitement : elle ne s'annule qu'en $z_1 = z_2$, elle est proportionnelle à l'erreur de positionnement $z_1 - z_2$ et surtout toutes les grandeurs sont connues au moment du calcul de e_2.

On propose donc la loi de commande définie par (4.24) dans laquelle on impose :

$$\omega_x|_k = K_p \, e_2|_k \tag{4.40}$$

où K_p est un gain proportionnel, soit en fonction des coefficients α, β et γ :

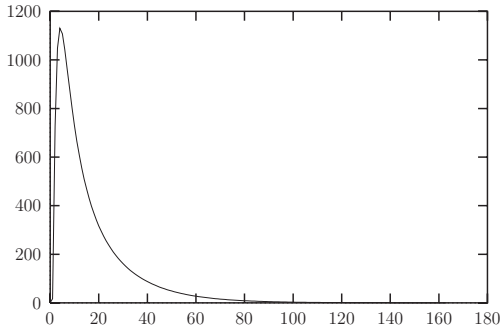
$$\omega_x|_k = K_p \left(\alpha|_{k-1} \, \epsilon_1|_k + \beta|_{k-1} \, \epsilon_2|_k + \gamma|_{k-1} \right) \tag{4.41}$$

Le comportement de cette loi de commande fait l'objet des paragraphes suivants. Cependant, on s'aperçoit d'ores et déjà que pour l'initialiser, il faudra imposer pour la première itération une valeur ω_x non nulle. De plus, conformément à (4.39), on pourra choisir indifféremment son signe puisque la différence des profondeurs est connue par e_2 dès la deuxième itération. Pratiquement, on a choisi une valeur de ω_x suffisamment importante pour assurer une variation significative de $\underline{\epsilon}$.

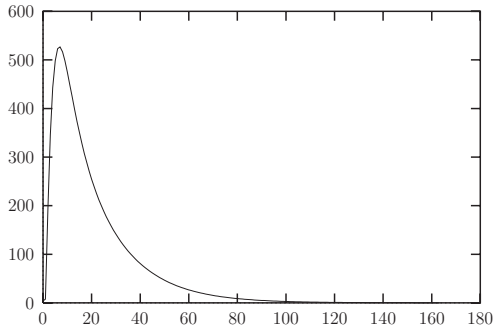
4.2.3.1 Résultats de simulations non bruitées

Les figures 4.2a, 4.2b, 4.2c et 4.2d représentent, une fois le centrage obtenu, les évolutions respectives de $\epsilon_1|_k$, $\epsilon_2|_k$, des composantes du torseur cinématique à l'instant k et de $(z_1 - z_2)|_k$. Elles nous permettent de constater que cette loi de commande permet de réaliser correctement la tâche de positionnement souhaitée et cela sans information sur la situation ou sur la taille de l'objet. Une erreur de position initiale $z_1 - z_2$ de 100 mm (après centrage) est réduite à zéro en une centaine d'itérations.

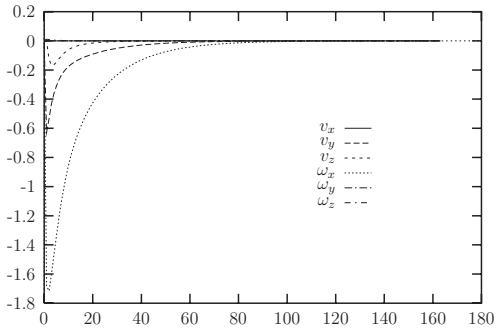
Par ailleurs, on peut remarquer qu'en fait le mouvement réalisé n'appartient pas exactement au noyau de la matrice d'interaction. En effet, les signaux d'erreur ϵ_1 et ϵ_2 augmentent fortement au début du mouvement, il s'en suit bien évidemment des commandes de compensation, en particulier sur v_z, pour maintenir l'image invariante. Par contre, quand $z_1 \to z_2$ l'approximation $z_1 = z_2 = z$ devient valide et les signaux ϵ_1 et ϵ_2 tendent alors vers zéro.



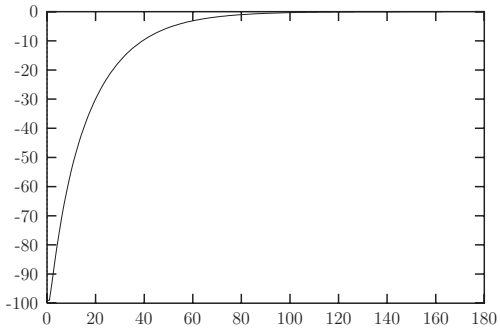
(a) Erreur dans le capteur relative à l'information visuelle Y_1 (μm) en fonction du nombre d'itérations



(b) Erreur dans le capteur relative à l'information visuelle Y_2 (μm) en fonction du nombre d'itérations



(c) Composantes du torseur cinématique (m/s ou rad/s) en fonction du nombre d'itérations



(d) Erreur de positionnement (mm) en fonction du nombre d'itérations

FIG. 4.2 – Utilisation d'une fonction de tâche secondaire — Informations visuelles non bruitées —

4.2.3.2 Résultats de simulations bruitées

Pour tester la robustesse de l'algorithme précédent, on se propose de bruiter les informations visuelles comme suit :

$$\begin{cases} Y_1 &= Y_1 + b_1 \\ Y_2 &= Y_2 + b_2 \end{cases} \tag{4.42}$$

où b_i est un bruit uniforme dans l'intervalle $[-p_y/10, p_y/10]$, p_y étant la hauteur des pixels.

Comme le montrent les courbes 4.3a, 4.3b, 4.3c et 4.3d, la loi de commande est sensible au bruit même si son amplitude est faible. Analysons plus avant ce phénomène.

Pour ce faire, on va tout d'abord se placer dans un cas simple — celui où la consigne est atteinte et le bruit nul — et on va exposer, dans ce cas, une façon plus rapide d'obtenir la fonction e_2.

La consigne étant réalisée, seul le mouvement appartenant à Ker $L^T_{|\underline{s}=\underline{s}^*, z_1=z_2=z}$ existe, le torseur cinématique s'écrit :

$$T_c = \left(0, z\left(1+Y^2\right)\omega_x|_k, 0, \omega_x|_k, 0, 0\right)^T \tag{4.43}$$

Connaissant T_c, on en déduit la loi d'évolution de $\underline{s}$:

$$\underline{s}|_{k+1} = \underline{s}^* + \Delta t L^T_{|\underline{s}=\underline{s}^*} T_c \tag{4.44}$$

par suite, il vient :

$$Y_1|_{k+1} - Y = \frac{(1+Y^2)\Delta t\, \omega_x|_k (z - z_1)}{z_1} \tag{4.45}$$

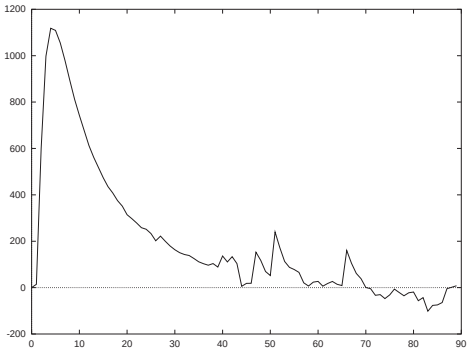
$$Y_2|_{k+1} + Y = \frac{(1+Y^2)\Delta t\, \omega_x|_k (z - z_2)}{z_2} \tag{4.46}$$

La seule façon d'obtenir un terme proportionnel à l'erreur de positionnement $z_1 - z_2$ est de réaliser $\alpha\left(\left(Y_1|_{k+1} - Y\right) - \left(Y_2|_{k+1} + Y\right)\right)$ soit :

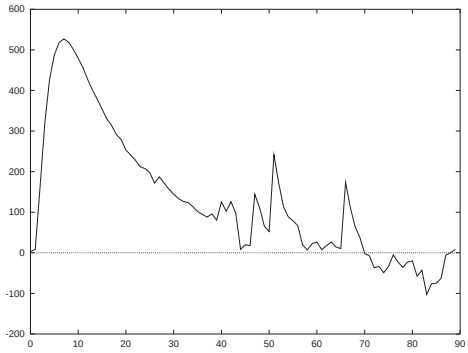
$$\alpha\left(\left(Y_1|_{k+1} - Y\right) - \left(Y_2|_{k+1} + Y\right)\right) = \alpha\frac{z(1+Y^2)\Delta t\, \omega_x|_k (z_1 - z_2)}{z_1 z_2} \tag{4.47}$$

On impose alors :

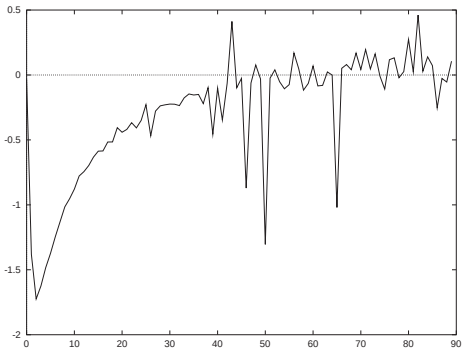
$$\alpha = \frac{1}{z(1+Y^2)\Delta t\, \omega_x|_k} \tag{4.48}$$



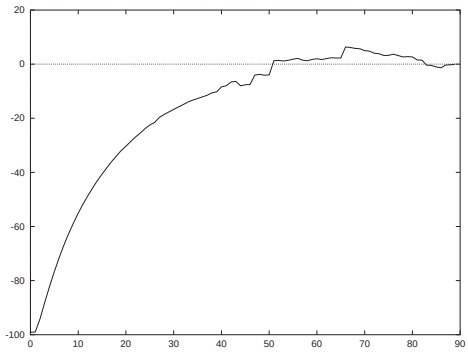
(a) Erreur dans le capteur relative à l'information visuelle Y_1 (μm) en fonction du nombre d'itérations



(b) Erreur dans le capteur relative à l'information visuelle Y_2 (μm) en fonction du nombre d'itérations



(c) Composante ω_x du torseur cinématique (m/s ou rad/s) en fonction du nombre d'itérations



(d) Erreur de positionnement(mm) en fonction du nombre d'itérations

FIG. 4.3 – Utilisation d'une fonction de tâche secondaire — Informations visuelles bruitées —

et finalement on obtient :

$$e_2|_{k+1} = \left.\frac{z_1 - z_2}{z_1 z_2}\right|_k \tag{4.49}$$

soit :

$$\omega_x|_{k+1} = \frac{K_p}{z(1+Y^2)\Delta t\ \omega_x|_k}\left(\left(Y_1|_{k+1} - Y\right) - \left(Y_2|_{k+1} + Y\right)\right) \tag{4.50}$$

Considérons maintenant le cas où les informations visuelles sont bruitées conformément à (4.42). Dans ces conditions (4.47) devient :

$$e_2|_{k+1} = \alpha\frac{z(1+Y^2)\Delta t\ \omega_x|_k\,(z_1 - z_2)}{z_1 z_2} - \alpha(b_2 - b_1) \tag{4.51}$$

et si l'on conserve pour α l'expression donnée en (4.48) :

$$e_2|_{k+1} = \left.\frac{z_1 - z_2}{z_1 z_2}\right|_k - \frac{b_2|_{k+1} - b_1|_{k+1}}{z(1+Y^2)\Delta t\ \omega_x|_k} \tag{4.52}$$

Cette expression se décompose en deux termes, un premier non bruité et un second bruité. À la vue de ce second terme, nous pouvons tirer les conclusions suivantes :

- Au début du mouvement appartenant à Ker $L^T_{|\underline{s}=\underline{s}^*, z_1=z_2=z}$, l'erreur de positionnement $z_1 - z_2$ est importante et, par suite, la commande $\omega_x|_k$ l'est aussi. Ce second terme est donc négligeable devant le premier qui lui, au contraire, est grand.

- Quand on s'approche de $z_1 = z_2$, la commande $\omega_x|_k$ tend vers zéro et cette fois c'est le premier terme qui devient négligeable devant le second. Le bruit est en quelque sorte amplifié. La fonction $e_2|_{k+1}$ et donc la commande $\omega_x|_{k+1}$ est essentiellement du bruit.

Le problème vient du fait que $\lim\limits_{z_1 \to z_2} \alpha = \infty$. Mais, si on imposait une valeur bornée de α, la commande $\omega_x|_k$ étant décroissante, d'après (4.47) le signal d'erreur $e_2|_{k+1}$ décroîtrait alors rapidement vers zéro $\omega_x|_k$ n'étant plus compensée par α.

Par conséquent, cette façon de faire ne peut conduire à un positionnement précis, une autre approche est indispensable. Elle est décrite dans le prochain paragraphe.

4.2.4 Scrutation d'informations visuelles

La façon de centrer l'image à l'aide de quatre informations visuelles limite en fait notre marge de manœuvre puisque non seulement les informations visuelles restent invariantes au cours du mouvement mais l'image du segment aussi.

On se place alors dans les conditions du paragraphe 4.2.2.2 où trois informations visuelles sont utilisées.

Une fois l'image centrée, on applique une loi de commande du type :

$$T_c = -\lambda L^{T+}_{\substack{X_i=0 \\ z_i=z}} \left(\underline{s} - \underline{s}^*\right) + T_s \tag{4.53}$$

avec :

$$T_s = \begin{pmatrix} 0 & z\omega_x \dfrac{2 + {Y_1}^2 + {Y_2}^2}{2} & 0 & \omega_x & 0 & 0 \end{pmatrix}^T (\text{qui } \in \text{Ker } L^T_{\substack{X_i=0 \\ z_i=z}}) \tag{4.54}$$

où ω_x est à déterminer.

Si $\underline{s}$ est proche de $\underline{s}^*$, seul T_s génère un mouvement. Il s'apparente à une pseudo-rotation d'axe x. Intuitivement, on imagine assez facilement que la taille du segment dans l'image $l = Y_1 - Y_2$ peut fournir des informations. On pense en particulier qu'elle passe par un maximum une fois la caméra en face de l'objet, cette hypothèse a été validée grâce à des simulations.

Afin d'élaborer une commande basée sur la taille du segment, on va s'efforcer dans le paragraphe suivant d'en trouver une modélisation.

4.2.4.1 Modélisation de la taille de l'image à l'aide d'informations dynamiques

On peut montrer que lorsque les conditions suivantes sont remplies :

- seuls ω_x et v_y sont non nuls ;
- la cadence d'échantillonnage est suffisamment rapide pour rendre l'approximation $t = k\Delta t$ valide à l'ordre un ;
- le mouvement réalisé appartient au noyau de $L^T_{\substack{X_i=0 \\ z_i=z}}$.

la taille de l'image est donnée [1] par :

$$l(\theta_x) = l(\theta_x|_k) - 2\frac{(1+a_k^2)\,(z_1|_k - z_2|_k)\,(\theta_x|_k - \theta_x)}{z_1|_k + z_2|_k} \tag{4.55}$$

avec $a_k = \dfrac{y_1|_k}{z_1|_k} = -\dfrac{y_2|_k}{z_2|_k}$ (si l'on suppose que la tâche principale maintienne l'image correctement centrée) et $\theta_x|_k = \displaystyle\int_{t=0}^{t=k\Delta t} \omega_x dt$.

Malheureusement, cette expression est en pratique inutilisable $z_1|_k$ et $z_2|_k$ étant inconnus.

Pour obtenir néanmoins une représentation analytique de $l = f(\theta_x)$ on va raisonner, dans le paragraphe suivant, à partir d'informations statiques.

4.2.4.2 Modélisation de la taille de l'image à l'aide d'informations statiques

Soient m_1 et m_2 les extrémités du segment considéré et m_0 un point de l'espace tels que $m_1 = (y^*, z^*)^T$, $m_2 = (-y^*, z^*)^T$ et $m_0 = (0, z_0)^T$ coordonnées exprimées dans le repère de la caméra en position souhaitée. On suppose que le mouvement réalisé lors de la maximisation de la taille de l'image résulte d'une rotation d'axe x passant par m_0 et d'angle θ_x puis d'une translation $\mathcal{T}$ (cf. figure 4.4) . Les coordonnées des points m_1 et m_2 s'expriment alors dans le repère de la caméra par :

$$\left\{ \begin{array}{lcl} m_1\prime & = & \mathcal{R}\,(m_1 - m_0) + \mathcal{T} \\ m_2\prime & = & \mathcal{R}\,(m_2 - m_0) + \mathcal{T} \end{array} \right. \tag{4.56}$$

où :

$$\mathcal{R} = \begin{pmatrix} \cos\theta_x & \sin\theta_x \\ -\sin\theta_x & \cos\theta_x \end{pmatrix}$$

et :

$$\mathcal{T} = \begin{pmatrix} 0 \\ z_0 \end{pmatrix}$$

On en déduit par transformation perspective les coordonnées des points $M_1 = (X_1, Y_1)^T$ et $M_2 = (X_2, Y_2)^T$ images des points m_1 et m_2 :

[1] Le calcul complet est proposé en annexe A.

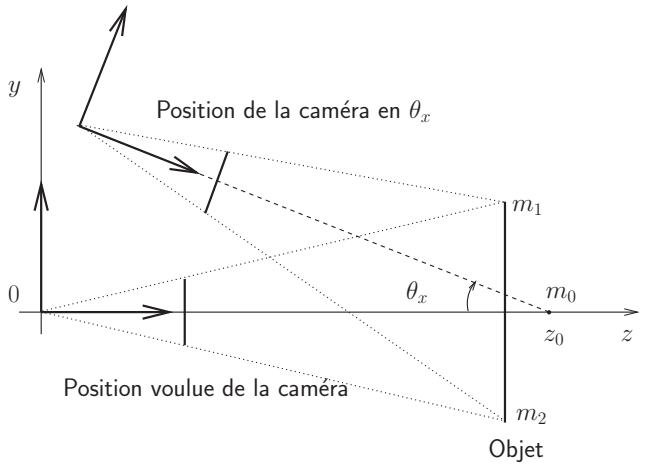


FIG. 4.4 – Modélisation de la taille de l'objet dans l'image en fonction de θ_x

$$\begin{cases} X1 &= 0 \\ Y1 &= -\dfrac{y^* \cos\theta_x + (z^* - z_0)\sin\theta_x}{y^* \sin\theta_x - (z^* - z_0)\cos\theta_x - z_0} \\ X2 &= 0 \\ Y2 &= -\dfrac{y^* \cos\theta_x - (z^* - z_0)\sin\theta_x}{y^* \sin\theta_x + (z^* - z_0)\cos\theta_x + z_0} \end{cases} \tag{4.57}$$

On cherche à maintenir par le biais de z_0 une image centrée, c'est-à-dire à annuler $Y_1 + Y_2$ et en fait, par analogie avec le paragraphe précédent, à réaliser un mouvement appartenant au noyau de la matrice d'interaction $L^T_{X_i=0}$. Deux valeurs de z_0 conviennent :

$$\begin{cases} z_{01} &= \dfrac{1}{2}\dfrac{2z^* \cos\theta_x - z^* + \sqrt{z^{*2} - 4y^{*2}(\cos\theta_x - 1)\cos\theta_x}}{\cos\theta_x - 1} \\ z_{02} &= \dfrac{1}{2}\dfrac{2z^* \cos\theta_x - z^* - \sqrt{z^{*2} - 4y^{*2}(\cos\theta_x - 1)\cos\theta_x}}{\cos\theta_x - 1} \end{cases} \tag{4.58}$$

On lève cette ambiguïté en se placant en $\theta_x = \pi/2$:

$$\begin{cases} z_{01}(\pi/2) &= 0 \\ z_{02}(\pi/2) &= z^* \end{cases} \tag{4.59}$$

on élimine ainsi le cas irréaliste $z_0 = z_{01}$.

Posons maintenant $v = \dfrac{y^*}{z^*}$ et supposons $|\theta_x| \leq \dfrac{\pi}{2}$, il vient après simplification :

$$l_m = 2v \frac{2\cos\theta_x + 1 - \sqrt{1 - 4v^2(\cos\theta_x - 1)\cos\theta_x}}{2v^2(\cos\theta_x - 1) + 1 + \sqrt{1 - 4v^2(\cos\theta_x - 1)\cos\theta_x}} \tag{4.60}$$

On peut vérifier, comme on s'y attend, que $l_m(\theta_x)$ est une fonction 2π périodique, paire, qu'elle s'annule en $\theta_x = \pi/2$ et qu'en $\theta_x = 0$ elle vaut bien $2\dfrac{y^*}{z^*}$ qui est en fait la taille de l'image quand la caméra se situe en face de l'objet.

De plus, pour $|\theta_x| \leq \dfrac{\pi}{2}$ on peut montrer qu'elle admet un unique maximum en $\theta_x = 0$.

Cette fonction, relativement complexe, peut s'écrire plus simplement dans le cas où v est faible par une valeur approchée l_a de l_m :

$$l_a = 2v\cos\theta_x \tag{4.61}$$

La figure 4.5 représente en simulation l'erreur en pixel entre la taille de l'image l et la taille de l'image modélisée respectivement par (4.60) puis par (4.61). Les conditions de simulation étant les suivantes :

- les calculs concernant la matrice d'interaction ont été réalisés pour les vraies valeurs de z_i mais en $\underline{s} = \underline{s}^*$;
- ω_x = -0.05 rad/s ;
- $m_1 = (0.1, 0.7)^T$ et $m_2 = (-0.1, 0.7)^T$ (en mètre).

Pour se donner une idée des ordres de grandeur, on a représenté sur la figure 4.6 les variations de $l(\theta_x)$, $l_m(\theta_x)$ et $l_a(\theta_x)$.

Comme on peut le vérifier l'hypothèse d'un mouvement de pseudo rotation de rayon $z_0 = z_{02}$ variable et de centre m_0 est assez réaliste puisque l'erreur relative maximum est de l'ordre de 0.6 %.

Les résultats obtenus sont donc tout à fait satisfaisants, cependant que ce soit l_m ou l_a ces fonctions dépendent, par le biais de v, de la situation désirée et par conséquent inconnue de l'objet. Une phase d'identification ou d'apprentissage en ligne est donc nécessaire. La mise en œuvre d'une telle identification de l_m peut s'avérer délicate puisque le modèle obtenu est non linéaire. De plus, la fonction

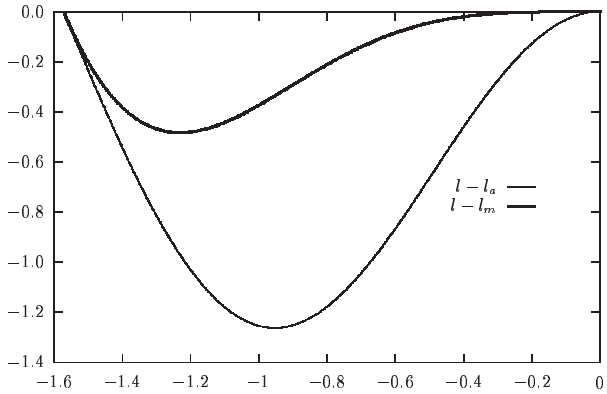


FIG. 4.5 – Erreur de modélisation : $l - l_m$ et $l - l_a$ (pixel) en fonction de θ_x (rad)

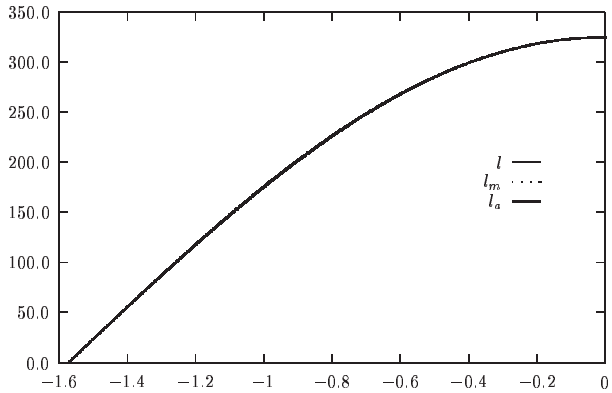


FIG. 4.6 – Variations de l, l_m et l_a (pixel) en fonction de θ_x (rad)

à utiliser est en pratique $l_m(\theta_x - \theta_{xc})$ où θ_{xc} est la valeur angulaire où l_m admet un maximum. C'est en fait cette valeur qu'il est important de connaître mais qui est totalement inconnue. Par conséquent, en l'absence d'estimé initial correct, l'utilisation d'un algorithme d'identification non linéaire, du type Levenberg-Marquardt par exemple, conduirait inévitablement à l'échec. Aussi, dans un but de simplification, allons nous examiner le cas de figure où le point m_0 est supposé fixe. Le passage des points m_i aux points $m_i\prime$ est toujours réalisé par (4.56) mais avec $m_0 = (0, z^*)^T$ et $\mathcal{T} = (t_y, z^*)^T$.

Les points M_1 et M_2 deviennent :

$$\begin{cases} X1 &= 0 \\ Y1 &= -\dfrac{y^* \cos\theta_x - t_y}{y^* \sin\theta_x - z^*} \\ X2 &= 0 \\ Y2 &= -\dfrac{y^* \cos\theta_x + t_y}{y^* \sin\theta_x + z^*} \end{cases} \tag{4.62}$$

On cherche la valeur de t_y qui annule $Y_1 + Y_2$ par analogie avec un mouvement du noyau de $L^T_{X_i=0}$. Il vient :

$$t_y = \frac{y^{*2} \cos\theta_x \sin\theta_x}{z^*} \tag{4.63}$$

que l'on injecte dans $l_m = Y_1 - Y_2$ qui devient :

$$l_m = 2v \cos\theta_x \tag{4.64}$$

On retrouve l'expression obtenue précédemment quand on supposait v faible. En fait, il s'avère que cette expression peut s'utiliser sous d'autres hypothèses moins restrictives que v faible comme le présupposait déjà la figure 4.5 puisque la valeur de v était relativement importante ($v \approx 0.14$) et la modélisation étonnamment correcte.

4.2.4.3 Application à la commande

Au cours du mouvement, dont la forme de la loi de commande est donnée par (4.53), on réalise une identification en ligne de l en supposant l modélisé par l_a que l'on développe en :

$$l_a = c_1 \cos\theta_x + c_2 \sin\theta_x \tag{4.65}$$

On obtient ainsi une équation linéaire par rapport aux paramètres c_1 et c_2 que l'on peut identifier par la méthode des moindres carrés. Toutefois, comme l'acquisition des couples (θ_x, l_a) est réalisée en ligne, on préfère à l'algorithme présenté en 3.5.2 une formulation récursive. D'autre part, pour tenir compte des imperfections du modèle, on pondère exponentiellement les mesures passées (figure 4.7) en introduisant un facteur d'oubli γ avec $\gamma = e^{-\Delta t/\tau}$ où τ est la constante de temps de l'exponentielle [Labarrere 88].

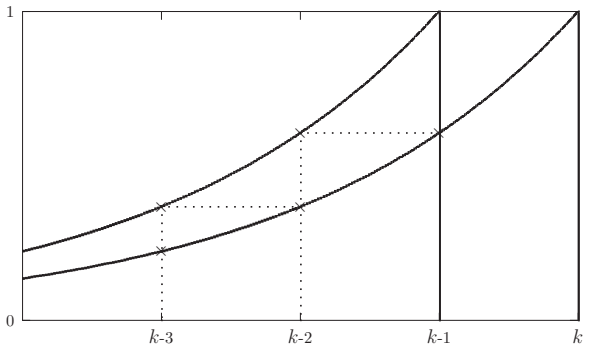


FIG. 4.7 – Pondération exponentielles des échantillons (d'après [Labarrere 88])

On identifie ensuite l_a à $l_{max} \cos(\theta_x - \theta_{xc})$ conformément à (4.61) ou (4.64) pour obtenir la valeur de θ_{xc} recherchée. Par suite, il vient :

$$\theta_{xc} = -\arctan \frac{c_2}{c_1} \tag{4.66}$$

Puis, pour obtenir une décroissance quasi exponentielle de l'erreur angulaire $\theta_x - \theta_{xc}$, on calcule la commande par :

$$\omega_x = K_x(\theta_x - \theta_{xc}) \tag{4.67}$$

où K_x est un scalaire positif permettant d'ajuster la vitesse de décroissance de cette erreur.

Il ne reste plus qu'à déterminer la façon de faire pendant le temps où le résultat de l'apprentissage n'est pas suffisamment correct pour utiliser θ_{xc}. On propose simplement d'imposer ω_x à une valeur ω_0 constante. Cependant, afin de ne pas conduire à un éventuel changement de signe de ω_x une fois les résultats de l'apprentissage corrects, il convient de déterminer le signe à appliquer à ω_0.

Ce problème se résout grâce à la phase initiale décrite dans 4.2.2.2. On suppose α suffisamment petit pour que l'image soit centrée dans le capteur sans que la

consigne l_c soit atteinte. Par suite, on peut montrer en utilisant (4.2), (4.22) puis (4.3) dans lesquelles on aura préalablement remplacé Y_1 par $l/2$ et Y_2 par $-l/2$ que le torseur cinématique devient :

$$\underline{T}_c = \begin{pmatrix} 0 & 0 & \dfrac{-\alpha\lambda l(l-l_c)}{z} & 0 & 0 & 0 \end{pmatrix}^T \tag{4.68}$$

et en estimant la variation des informations visuelles par :

$$\underline{\dot{s}} = L^T_{|X_i=0} T_c \tag{4.69}$$

il vient :

$$\dot{Y}_1 + \dot{Y}_2 = \frac{1}{2}\alpha\lambda \frac{z_1|_k - z_2|_k}{z_1|_k z_2|_k} \frac{l^2(l-l_c)}{z} \tag{4.70}$$

on en déduit alors le signe de ω_0 directement relié à celui de $z_1|_k - z_2|_k$:

$$\text{signe}(\omega_0) = \text{signe}\left(z_1|_k - z_2|_k\right) = \text{signe}\left(\frac{\dot{Y}_1 + \dot{Y}_2}{l - l_c}\right) \tag{4.71}$$

La loi de commande étant complètement définie, on va la tester grâce à des simulations. Elles font l'objet des paragraphes suivants.

4.2.4.4 Simulations non bruitées

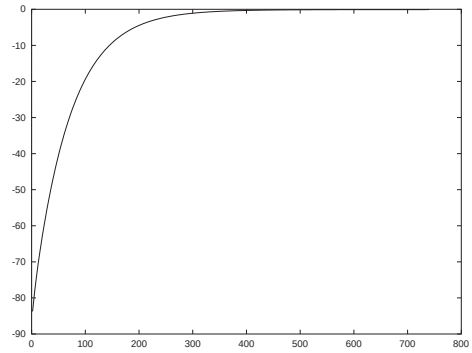
Les figures 4.8a, 4.8b, 4.8c et 4.8d montrent, une fois l'image centrée et ajustée à $l = l_c$, le comportement de l'algorithme proposé. Plus précisément, la figure 4.8a rapporte le comportement de l'erreur de positionnement $z_1 - z_2$, la figure 4.8b celui de ω_x, l'évolution de θ_{xc} est représentée sur la figure 4.8c alors que l'évolution de l'erreur de modélisation est décrite par la figure 4.8d.

Tout d'abord, on s'aperçoit que l'erreur de modélisation est très faible. On remarque aussi que θ_{xc} devient rapidement stable et, finalement, cette bonne modélisation couplée à une bonne identification conduit à une convergence rapide.

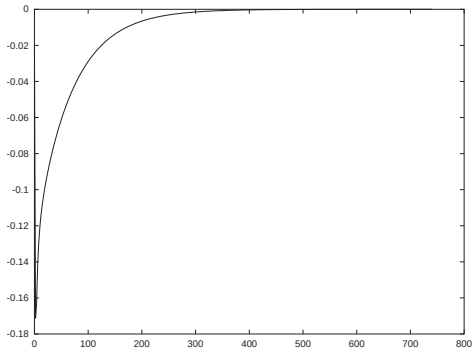
Nous allons donc tester dans le paragraphe suivant la robustesse de l'algorithme vis-à-vis d'une image bruitée.

4.2.4.5 Simulations bruitées

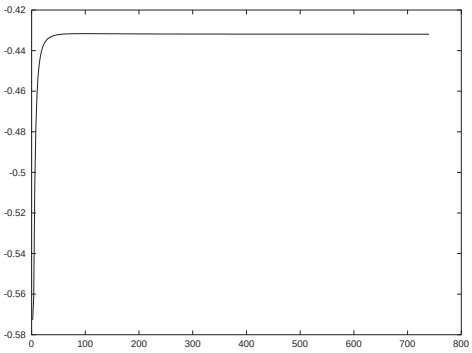
Les informations visuelles sont bruitées conformément à (4.42) mais cette fois dans l'intervalle $[-2p_y, 2p_y]$, c'est-à-dire par un bruit uniforme d'amplitude



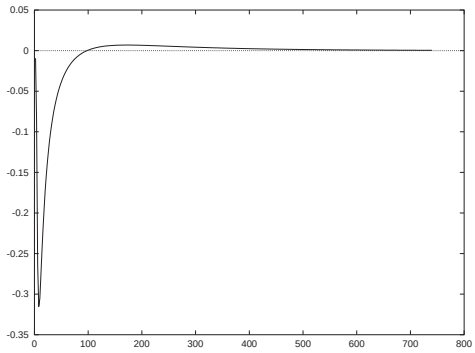
(a) Évolution de l'erreur de positionnement $z_1 - z_2$ (mm) en fonction du nombre d'itérations



(b) Évolution de ω_x (rad/s) en fonction du nombre d'itérations



(c) Évolution de θ_{xc} estimé (rad) en fonction du nombre d'itérations



(d) Évolution de l'erreur de modélisation $l - l_a$ (pixel) en fonction du nombre d'itérations

FIG. 4.8 – Maximisation de la taille de l'objet dans l'image — Informations visuelles non bruitées —

maximum 2 pixels. Pour ces simulations, seule la durée d'apprentissage a été modifiée, elle a été portée à 20 itérations.

À la vue des figures 4.9a, 4.9b et 4.9c on remarque tout d'abord nettement la présence de la période initiale destinée à l'apprentissage, période pendant laquelle ω_x est constant. Par ailleurs, la consigne angulaire θ_{xc} est cette fois un peu plus bruitée mais se stabilise toujours relativement vite. On peut noter aussi la faible influence du bruit sur ω_x ou sur l'erreur de positionnement vis-à-vis des oscillations qu'il aurait pu engendrer. En effet, comme le modèle recopie bien la réalité on peut imposer une forte valeur de γ et ainsi filtrer fortement le bruit.

4.2.4.6 Discussion sur l'erreur de positionnement

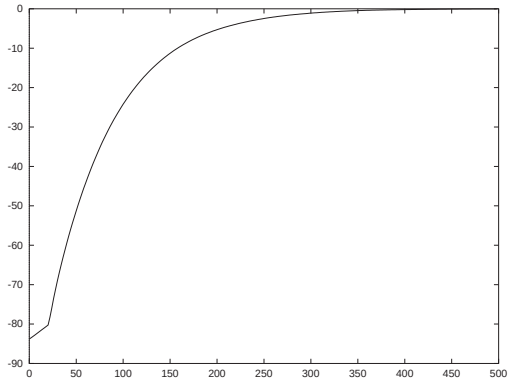
L'algorithme décrit ci-dessus repose sur deux points importants :

- Tout d'abord sur une modélisation correcte de la taille du segment dans l'image. On a vu que la validité du modèle dépendait soit de la situation du porteur par rapport à l'objet par l'intermédiaire de la variable v soit de l'hypothèse selon laquelle la trajectoire du porteur, dans le repère de l'objet, serait une pseudo rotation autour d'un point fixe. Cette dernière hypothèse n'est en pratique pas très exploitable car difficilement vérifiable. On s'efforcera donc de vérifier v faible même si cette hypothèse est plus forte comme on l'a vu lors de simulations.

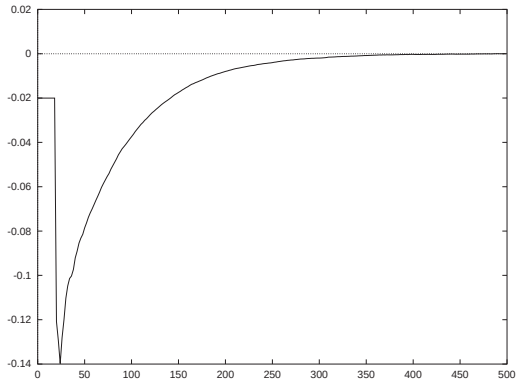
  Si besoin était, n'oublions pas qu'une identification non linéaire pourrait toujours être envisagée avec les restrictions apportées dans le paragraphe 4.2.4.2.

- Il repose aussi sur la qualité de l'identification. Une première contrainte vient immédiatement à l'esprit : disposer d'un nombre suffisant de couples $(\theta_x|_k, l_k)$ puis, une deuxième concernant la répartition de ces points. En effet, pour réaliser un apprentissage global de la courbe $l(\theta_x)$ et non pas un apprentissage local (qui serait trop dépendant du bruit) il faut, dans la mesure du possible, disposer d'autant de couples au début du mouvement qu'à la fin. La conséquence de ces deux contraintes est de devoir limiter ω_x en imposant K_x suffisamment faible. Pour les mêmes raisons on a intérêt à choisir ω_0 faible.

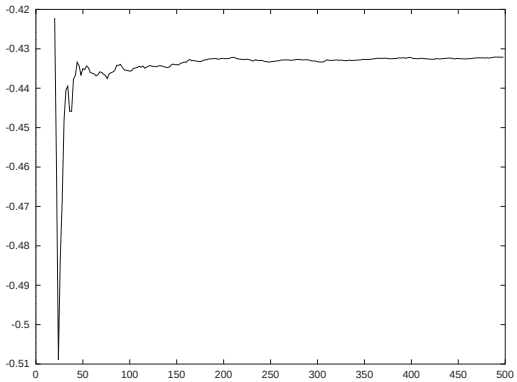
  Par contre, quand l'erreur de positionnement initiale $z_1 - z_2$ est très faible et par suite l'erreur angulaire aussi, les deux contraintes précédentes ne peuvent être que difficilement remplies et une erreur de positionnement relativement importante peut apparaître. Le cas le plus défavorable que



(a) Évolution de l'erreur de positionnement $z_1 - z_2$ en fonction du nombre d'itérations



(b) Évolution de ω_x en fonction du nombre d'itérations



(c) Évolution de θ_{xc} estimé en fonction du nombre d'itérations

FIG. 4.9 – Maximisation de la taille de l'image — Informations visuelles bruitées par un bruit d'amplitude deux pixels —

nous avons rencontré conduit à un comportement rapporté sur la figure 4.10 où une erreur de positionnement initiale faible ($\approx$ 15 mm) devient inférieure à 1.5 mm après asservissement.

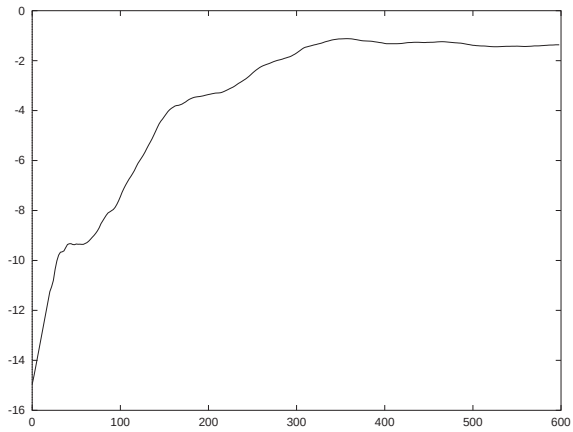


FIG. 4.10 – Évolution de l'erreur de positionnement en fonction du nombre d'itérations dans le cas où l'erreur initiale est faible — Informations visuelles bruitées par un bruit d'amplitude deux pixels —

Après avoir traité le cas des objets de type « segment », nous allons, dans le paragraphe suivant, voir comment il est possible de transposer la démarche suivie et les résultats obtenus au cas d'un objet plan de forme quelconque.

4.3 Étude d'un objet plan de forme quelconque

Dans le paragraphe 4.2, on a réussi à réaliser la tâche de positionnement souhaitée en maximisant un critère — la taille de l'image de l'objet segment — au cours du mouvement. Dans le cas d'un objet décrit dans un espace de dimension deux une transposition naturelle consiste à maximiser la surface S de l'image de l'objet. Pour ce faire, il serait intéressant de disposer d'une modélisation de cette surface. L'obtenir dans le cas général n'est pas simple aussi, allons nous simplifier une nouvelle fois le problème en supposant que l'on puisse extraire de l'image trois points caractéristiques de l'objet. Nous appliquerons ensuite la loi de commande résultante au cas d'un objet de forme complexe présentant de tels points, puis nous discuterons de quelques méthodes permettant leurs extractions. Pour terminer, le cas général d'un objet quelconque sera traité.

4.3.1 Étude d'un objet présentant trois points caractéristiques

En ne raisonnant que sur ces trois points, l'image à traiter devient équivalente à une image beaucoup plus simple comme celle décrite sur la figure 4.11, c'est-à-dire à un triangle de sommets $M_1(X_1, Y_1)$, $M_2(X_2, Y_2)$ et $M_3(X_3, Y_3)$ respectivement images des points $m_1(x_1, y_1, z_1)$, $m_2(x_2, x_2, z_2)$ et $m_3(x_3, y_3, z_3)$. En adoptant la même démarche qu'en 4.2, on va tout d'abord spécifier :

- une tâche principale qui permette l'équivalent du centrage réalisé dans le paragraphe 4.2.2.2 ;
- une tâche secondaire qui conduise à la réalisation de $z_1 = z_2 = z_3$ grâce aux informations fournies par l et h (avec h la hauteur issue du sommet M_3 comme décrit sur la figure 4.11).

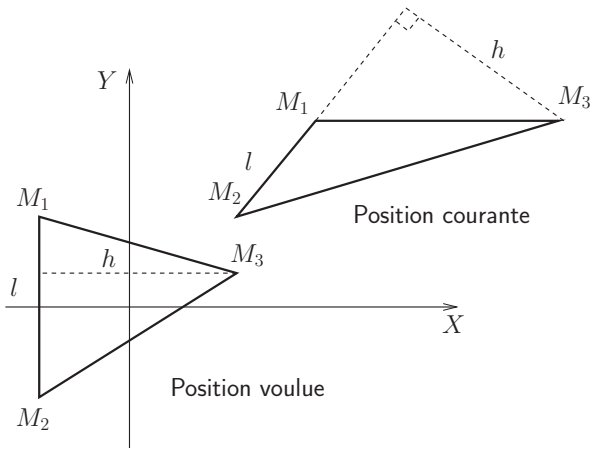


FIG. 4.11 – Triangle formé à l'aide des trois points caractéristiques (M_1, M_2, M_3) — position courante et position voulue —

4.3.1.1 Spécification de la tâche principale

Dans le cas de l'objet segment le centrage de l'image avait une signification bien précise, maintenant on s'aperçoit que plusieurs approches pourraient être utilisées comme chercher à positionner le centre de gravité par exemple. Cependant, si l' on veut pouvoir maximiser l, h ou même directement le produit lh,

on a tout intérêt à rester dans les conditions du paragraphe 4.2.2.2 où l'on sait comment varie l ou h. C'est-à-dire à imposer :

- $Y_1 + Y_2 = 0$ et $X_1 = X_2$ en ce qui concerne la maximisation de l (le cas $X_1 = X_2 = 0$ du paragraphe 4.2.2.2 était un cas particulier du cas $X_1 = X_2$) ;
- $X_1 + X_3 = 0$ en ce qui concerne la maximisation de h ;

ce qui conduit à définir $\underline{s}$ par :

$$\underline{s} = (X_1 - X_2, X_1 + X_3, Y_1 + Y_2)^T \tag{4.72}$$

et la consigne visuelle par :

$$\underline{s}^* = (0, 0, 0)^T \tag{4.73}$$

Les informations visuelles étant choisies on peut exprimer la matrice d'interaction en $\underline{s}^*$ et pour $z_i = z$ $(i = 1 \ldots 3)$:

$$L^T_{\substack{\underline{s}^* \\ z_i = z}} = \begin{pmatrix} 0 & 0 & 0 & 2X_1Y_1 & 0 & 2Y_1 \\ -\dfrac{2}{z} & 0 & 0 & X_1(Y_1 - Y_3) & -2(1 + X_1^2) & Y_1 + Y_3 \\ 0 & -\dfrac{2}{z} & 0 & 2\left(1 + {Y_1}^2\right) & 0 & -2X_1 \end{pmatrix} \tag{4.74}$$

et les deux vecteurs suivants appartenant au noyau de $L^T_{|\underline{s}^*, z_i = z}$:

$$\underline{T}_{sl} = \begin{pmatrix} -z\omega_x X_1 Y_3 & z\omega_x\left(1 + {X_1}^2 + {Y_1}^2\right) & 0 & \omega_x & 0 & -\omega_x X_1 \end{pmatrix}^T \tag{4.75}$$

$$\underline{T}_{sh} = \begin{pmatrix} -z\omega_y\left(1 + {X_1}^2\right) & 0 & 0 & 0 & \omega_y & 0 \end{pmatrix}^T \tag{4.76}$$

On appliquera donc une loi de commande de la forme suivante :

$$T_c = -\lambda L^{T+}_{|\underline{s}^*, z_i = z}(\underline{s} - \underline{s}^*) + T_{sl} + T_{sh} \tag{4.77}$$

dans laquelle ω_x et ω_y sont à déterminer.

4.3.1.2 Modélisation de la surface de l'objet dans l'image

Soient m_1, m_2 et m_3 les trois points caractéristiques de l'objet et m_0 un point de l'espace tels que $m_1 = (-x^*, y^*, z^*)^T$, $m_2 = (-x^*, -y^*, z^*)^T$, $m_3 = (x^*, r^*, z^*)^T$

et $m_0 = (0, 0, z^*)^T$ coordonnées exprimées dans un repère fixe. On suppose que le mouvement réalisé lors de la maximisation de la surface de l'image résulte de trois rotations de centre m_0 et d'une translation $\mathcal{T}$. Les coordonnées des points m_1, m_2 et m_3 s'expriment alors dans le repère de la caméra par :

$$\left\{ \begin{array}{rcl} m_1\prime & = & \mathcal{R}\,(m_1 - m_0) + \mathcal{T} \\ m_2\prime & = & \mathcal{R}\,(m_2 - m_0) + \mathcal{T} \\ m_3\prime & = & \mathcal{R}\,(m_3 - m_0) + \mathcal{T} \end{array} \right. \tag{4.78}$$

où :

$$\mathcal{R} = \begin{pmatrix} \mathrm{C}_{\theta_y}\mathrm{C}_{\theta_z} & \mathrm{S}_{\theta_y}\mathrm{S}_{\theta_z} & \mathrm{S}_{\theta_y} \\ -\mathrm{C}_{\theta_x}\mathrm{S}_{\theta_z} - \mathrm{S}_{\theta_x}\mathrm{S}_{\theta_y}\mathrm{C}_{\theta_z} & \mathrm{C}_{\theta_x}\mathrm{C}_{\theta_z} - \mathrm{S}_{\theta_x}\mathrm{S}_{\theta_y}\mathrm{S}_{\theta_z} & \mathrm{S}_{\theta_x}\mathrm{C}_{\theta_y} \\ \mathrm{S}_{\theta_x}\mathrm{S}_{\theta_z} - \mathrm{C}_{\theta_x}\mathrm{S}_{\theta_y}\mathrm{C}_{\theta_z} & -\mathrm{S}_{\theta_x}\mathrm{C}_{\theta_z} - \mathrm{C}_{\theta_x}\mathrm{S}_{\theta_y}\mathrm{S}_{\theta_z} & \mathrm{C}_{\theta_x}\mathrm{C}_{\theta_y} \end{pmatrix}$$

avec les notations habituelles $\mathrm{C}_\theta = \cos\theta$ et $\mathrm{S}_\theta = \sin\theta$ et :

$$\mathcal{T} = \begin{pmatrix} t_x \\ t_y \\ z \end{pmatrix}$$

On en déduit par transformation perspective les coordonnées des points $M_1 = (X_1, Y_1)^T$, $M_2 = (X_2, Y_2)^T$ et $M_3 = (X_3, Y_3)^T$ images des points m_1, m_2 et m_3. Puis, on cherche à maintenir l'image constante comme le ferait un mouvement appartenant au noyau de la matrice d'interaction. On résout donc, par rapport aux variables t_x, t_y, et θ_z, le système non linéaire suivant correspondant à $\underline{s} = \underline{s}^*$:

$$\left\{ \begin{array}{rcl} X_1 - X_2 & = & 0 \\ X_1 + X_3 & = & 0 \\ Y_1 + Y_2 & = & 0 \end{array} \right. \tag{4.79}$$

Son expression en fonction des inconnues est complexe et ne sera pas donnée ici. De plus, sa résolution analytique n'a malheureusement pas abouti. Toutefois, pour simplifier son expression, on se propose de réaliser un développement en série de Taylor à l'ordre deux par rapport aux variables $u = x^*/z^*$, $v = y^*/z^*$ et $w = r^*/z^*$. De plus, on pose $k_x = t_x/z^*$ et $k_y = t_y/z^*$. Tous calculs faits, la résolution de (4.79) fournit :

$$\left\{ \begin{array}{rcl} k_x & = & 0 \\ k_y & = & u \sin\theta_x \sin\theta_y \\ \theta_z & = & 0 \end{array} \right. \tag{4.80}$$

Il s'en suit :

$$\left\{ \begin{array}{lcl} l = Y_1 - Y_2 & = & 2v\cos\theta_x \\ h = X_3 - X_1 & = & 2u\cos\theta_y \end{array} \right. \tag{4.81}$$

On retrouve l'expression de l obtenue en 4.2.4.2. Par contre, le développement en série réalisé conduit à des expressions beaucoup trop simplifiées. En effet, l (respectivement h) devrait dépendre de θ_y (respectivement θ_x). De ce fait, les simulations de commande correspondantes ont échoué. Ne disposant pas d'une modélisation correcte, on ne peut donc pas au cours d'un même mouvement maximiser l et h simultanément et utiliser (4.77).

Par contre, on peut maximiser séquentiellement l puis h. On applique alors au porteur tout d'abord :

$$T_c = -\lambda L^{T+}_{|\underline{s}^*, z_i = z}(\underline{s} - \underline{s}^*) + T_{sl} \tag{4.82}$$

puis, une fois la longueur l maximisée :

$$T_c = -\lambda L^{T+}_{|\underline{s}^*, z_i = z}(\underline{s} - \underline{s}^*) + T_{sh} \tag{4.83}$$

où dans (4.82), conformément au paragraphe 4.2.4.3, ω_x est obtenue par (4.67). De même, on calcule ω_y par :

$$\omega_y = K_y(\theta_y - \theta_{yc}) \tag{4.84}$$

en supposant la longueur h modélisée par $h_{max}\cos(\theta_y - \theta_{yc})$, K_y jouant un rôle similaire à celui joué par K_x dans (4.67).

Pour que l'algorithme soit complet, il reste à déterminer les signes de ω_x et ω_y nécessaires à la phase d'apprentissage.

On procède pour cela comme en 4.2.4.3. Au cours d'une première phase de mouvement on cherche à atteindre la consigne visuelle tout en imposant une taille l_c de l'image. La fonction de tâche à réguler s'écrit :

$$\underline{e} = L^{T+}_{\substack{\underline{s}=\underline{s}^* \\ z_i = z}}(\underline{s} - \underline{s}^*) + \alpha \left(\mathbb{I}_6 - L^{T+}_{\substack{\underline{s}=\underline{s}^* \\ z_i = z}} L^{T}_{\substack{\underline{s}=\underline{s}^* \\ z_i = z}} \right) \underline{g}_s{}^T \tag{4.85}$$

où $\underline{g}_s{}^T = \left(\dfrac{\partial h_s}{\partial \underline{r}} \right)^T$ avec $h_s = \dfrac{1}{2}(l - l_c)^2$.

Supposons α suffisamment petit pour que $\underline{s} = \underline{s}^*$ et $l \neq l_c$. Alors, on peut montrer que $\dot{\underline{s}} = L^{T}_{\substack{\underline{s}=\underline{s}^* \\ z_i = z}} T_c$ devient :

$$\dot{\underline{s}} = \left(\frac{\alpha\lambda h l(l - l_c)}{2z} \frac{z_1 - z_2}{z_1 z_2}, \quad \frac{\alpha\lambda h l(l - l_c)}{2z} \frac{z_1 - z_3}{z_1 z_3}, \quad \frac{\alpha\lambda l^2(l - l_c)}{2z} \frac{z_1 - z_2}{z_1 z_2} \right)^T \tag{4.86}$$

on a donc accès, de la même façon qu'en 4.2.4.3, aux signes de $z_1 - z_2$ et $z_1 - z_3$ et par suite à ceux de ω_x et ω_y.

L'algorithme complet a été implanté en simulation sur des images bruitées, il a conduit à d'aussi bons résultats que ceux présentés en 4.2.4.5 avec les mêmes limitations qu'en 4.2.4.6.

Aucune représentation graphique de simulation ne sera proposée, elle serait redondante avec celle du paragraphes 4.2.4.5. Par contre, les simulations ayant validé les différentes lois de commande, des résultats expérimentaux seront présentés dans les paragraphes suivants.

4.3.2 Résultats expérimentaux

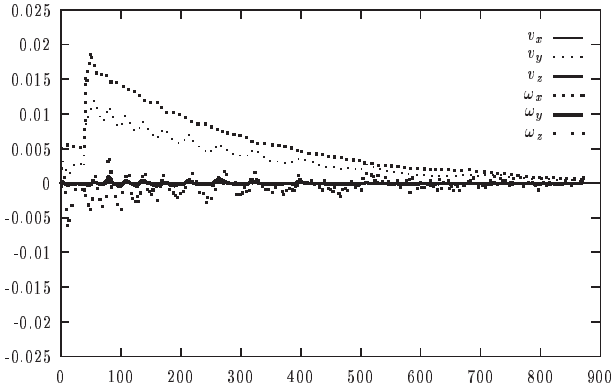
4.3.2.1 Vérification des résultats

Avant de présenter les résultats obtenus, précisons tout d'abord que la méthode de vérification de ces résultats est identique à celle décrite en 3.5.4.2 quand il s'agissait de réaliser une tâche de positionnement grâce à l'utilisation de la signature polaire. En outre, l'objet utilisé (voire la figure 3.16), tant pour réaliser la tâche de positionnement que pour vérifier les résultats est identique. De même, les notations ont été conservées.

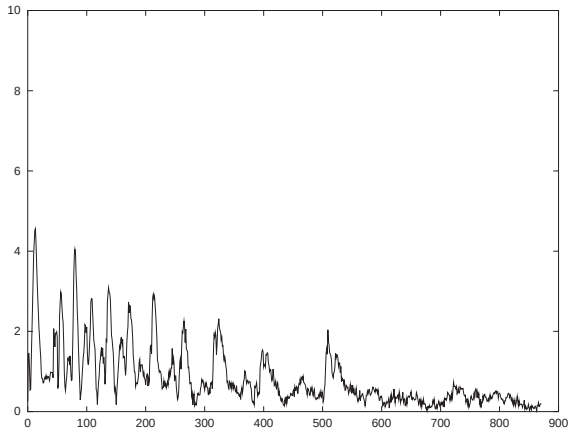
4.3.2.2 Résultats expérimentaux

Les figures 4.12a, 4.13a, 4.13b et 4.12b montrent les comportements respectifs des composantes du torseur cinématique, de la consigne angulaire et de la position angulaire courante, de la taille de l'image réelle et modélisée et de l'erreur dans le capteur (définie comme étant $\|\underline{s}\|$) lors de la maximisation de l. Les figures 4.14a, 4.15a, 4.15b et 4.14b représentent l'évolution des mêmes variables que précédemment mais lors de la maximisation de h. La figure 4.16 résume les différentes phases nécessaires à la réalisation de la tâche de positionnement souhaitée.

Tout d'abord ces résultats expérimentaux confirment les simulations sur images bruitées, en particulier les figures 4.13b et 4.15b valident expérimentalement les résultats théoriques concernant les modélisations de $l = f(\theta_x)$ et $h = f(\theta_y)$. On peut remarquer aussi sur les figures 4.13a et 4.15a l'obtention rapide d'une valeur de consigne angulaire qui s'affine au fur et à mesure de l'apprentissage puis qui

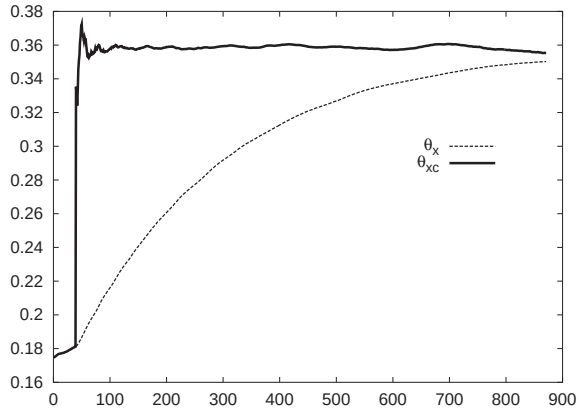


(a) Composantes du torseur cinématique (m/s ou rad/s) en fonction du nombre d'itérations

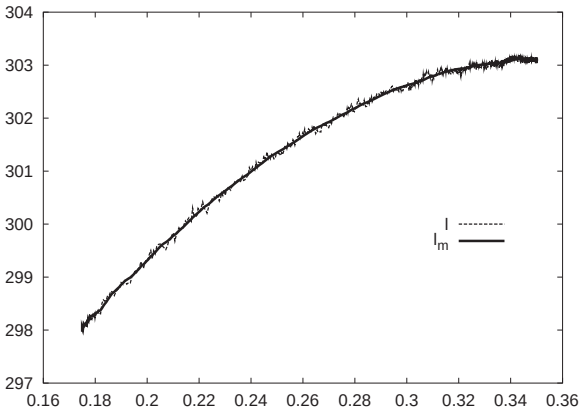


(b) Erreur dans le capteur (pixel) en fonction du nombre d'itérations

FIG. 4.12 – Résultats expérimentaux — cas où trois points de l'objet sont connus — Maximisation de l

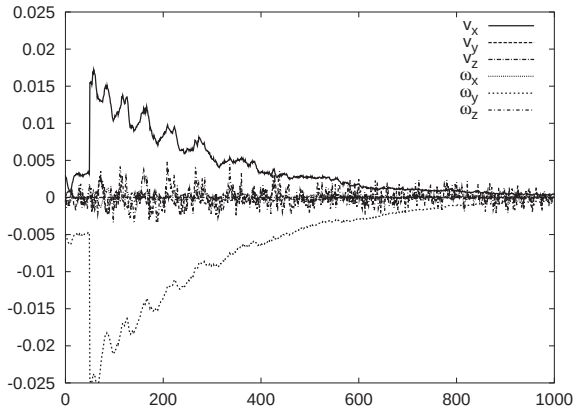


(a) Consigne angulaire (rad.) et position angulaire courante (rad.) en fonction du nombre d'itérations

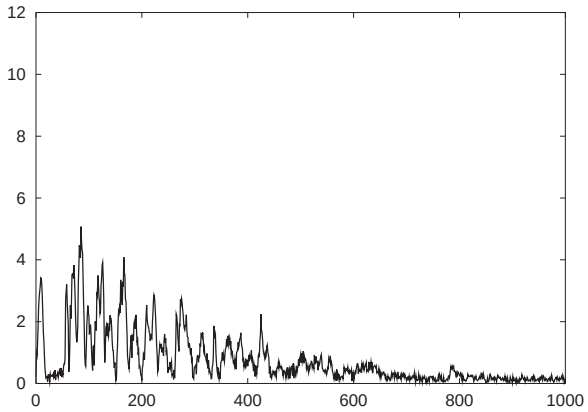


(b) Taille de l'image et taille modélisée (pixel) en fonction de θ_x

FIG. 4.13 – Résultats expérimentaux — cas où trois points de l'objet sont connus — Maximisation de l

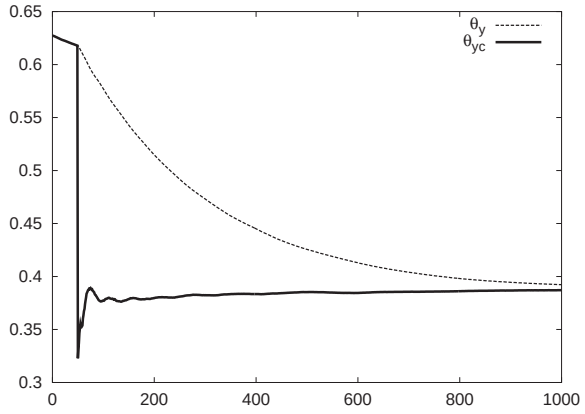


(a) Composantes du torseur cinématique (m/s ou rad/s) en fonction du nombre d'itérations

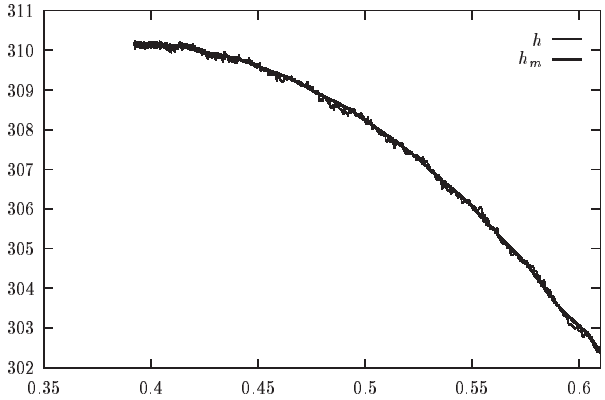


(b) Erreur dans le capteur (pixel) en fonction du nombre d'itérations

FIG. 4.14 – Résultats expérimentaux — cas où trois points de l'objet sont connus — Maximisation de h

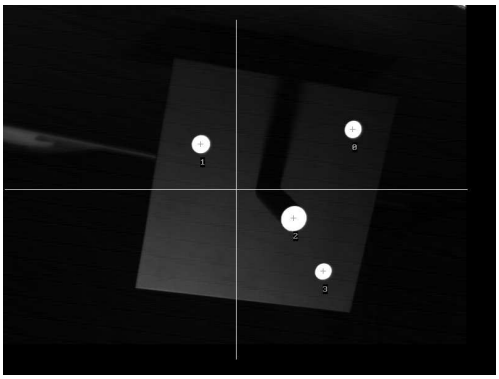


(a) Consigne angulaire (rad.) et position angulaire courante (rad.) en fonction du nombre d'itérations

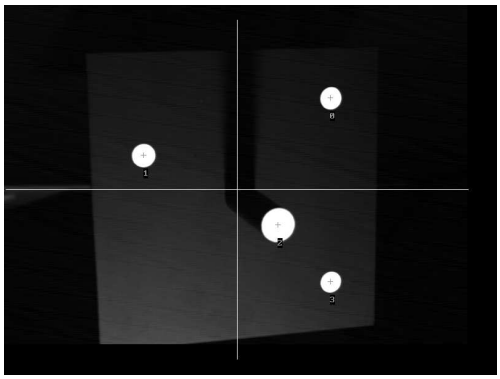


(b) Taille de l'image et taille modélisée (pix.) en fonction de θ_y

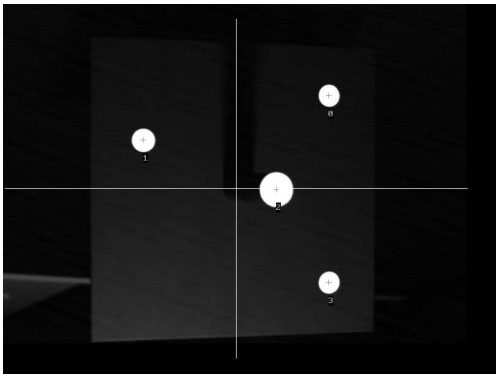
FIG. 4.15 – Résultats expérimentaux — cas où trois points de l'objet sont connus — Maximisation de h



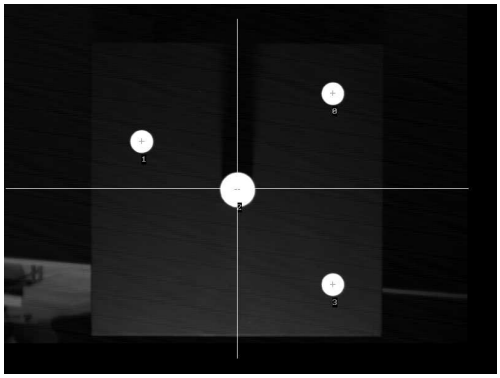
(a) Position initiale



(b) Position après centrage



(c) Position après maximisation de l



(d) Position finale

FIG. 4.16 – Résultats expérimentaux — cas où trois points de l'objet sont connus —

se stabilise. Il est à noter, comme en simulation, la faible influence du bruit sur le calcul de cette valeur.

Par ailleurs, les résultats obtenus par la méthode de Dementhon sont les suivants :

- Position initiale de la caméra par rapport à l'objet :
  - Angles d'Euler (en degrés) :

$$\gamma = 11.53 \pm 0.01 \quad \beta = 13.20 \pm 0.02 \quad \alpha = 7.97 \pm 0.02$$

  - Coordonnées des points caractéristiques (exprimées dans le repère de la caméra en mm) :

$$\begin{aligned} m_1 &= \begin{pmatrix} 77.87 \pm 0.05 & 72.77 \pm 0.02 & 830.88 \pm 0.13 \end{pmatrix}^T \\ m_2 &= \begin{pmatrix} 101.43 \pm 0.03 & -52.53 \pm 0.02 & 805.58 \pm 0.13 \end{pmatrix}^T \\ m_3 &= \begin{pmatrix} -29.78 \pm 0.02 & -38.77 \pm 0.03 & 782.20 \pm 0.14 \end{pmatrix}^T \end{aligned}$$

- Position finale de la caméra par rapport à l'objet :
  - Angles d'Euler (en degrés) :

$$\gamma = 0.20 \pm 0.00 \quad \beta = 0.08 \pm 0.01 \quad \alpha = -0.02 \pm 0.01$$

  - Coordonnées des points caractéristiques (exprimées dans le repère de la caméra en mm) :

$$\begin{aligned} m_1 &= \begin{pmatrix} 65.14 \pm 0.03 & 64.98 \pm 0.02 & 632.21 \pm 0.08 \end{pmatrix}^T \\ m_2 &= \begin{pmatrix} 65.09 \pm 0.03 & -65.00 \pm 0.02 & 631.75 \pm 0.08 \end{pmatrix}^T \\ m_3 &= \begin{pmatrix} -64.89 \pm 0.03 & -32.46 \pm 0.02 & 631.67 \pm 0.09 \end{pmatrix}^T \end{aligned}$$

Si le positionnement était réalisé sans erreur, on devrait obtenir une matrice de rotation égale à l'identité et par conséquent des angles d'Euler tous nuls et, pour les points m_i les valeurs suivantes :

$$\begin{aligned} m_1 &= \begin{pmatrix} 65 & 65 & z^* \end{pmatrix}^T \\ m_2 &= \begin{pmatrix} 65 & -65 & z^* \end{pmatrix}^T \\ m_3 &= \begin{pmatrix} -65 & -32.5 & z^* \end{pmatrix}^T \end{aligned}$$

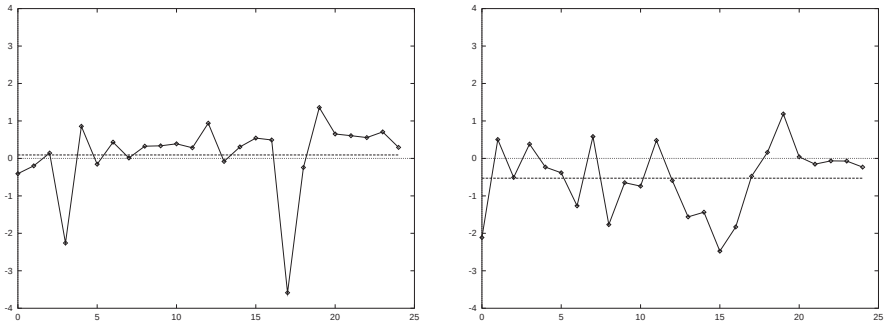
Par conséquent, les résultats obtenus sont tout à fait satisfaisants. Par ailleurs, on s'aperçoit une nouvelle fois que les valeurs des variables u, v et w (respectivement ≈ 0.13, ≈ 0.13 et ≈ 0.06) sont relativement importantes et que les modélisations de l et h sont quand même correctes.

Pour valider ce résultat — obtenu pour une position initiale donnée — nous avons réalisé 25 autres essais en suivant la même procédure mais pour d'autres positions initiales choisies dans le volume tel que $\gamma \in [-12^\circ, 27^\circ]$, $\beta \in [-22^\circ, 35^\circ]$ et $z_3 \in [353 \text{ mm}, 713 \text{ mm}]$. Les résultats sont représentés sur la figure 4.17 où le numéro de l'essai est porté en abscisse et les valeurs moyennes de γ et β sur les 100 images en ordonnée [2]. De plus, des résultats statistiques descriptifs sont résumés dans le tableau 4.1.

Par ailleurs les constantes utilisées dans l'algorithme pour l'ensemble des essais étaient :

- $\Delta t = 40$ ms
- Phase de centrage :
  - $\alpha = 1.5$;
  - $l_c = 300$ pixels ;
  - $z = 60$ cm ;
  - $\lambda = 2$.
- Phase de maximisation de l :
  - $\omega_0 = 0.005$ rad/s ;
  - $\lambda = 3$;
  - facteur d'oubli pour les moindres carrés $= 0.99$;
  - $K_x = 0.1$;
  - nombre d'acquisitions d'apprentissage $= 40$;
  - condition d'arrêt : $|\theta_{xc} - \theta_x| < 5e-3$ et $|\theta_{xc,k} - \theta_{xc,k-1}| < 5e-4$.
- Phase de maximisation de h :
  - $\omega_0 = 0.005$ rad/s ;
  - $\lambda = 5$;

[2] Nous ne présentons pas de résultat concernant l'angle α car l'information visuelle $X_1 - X_2$ a pour consigne zéro ce qui conduit à le rendre nul.



(a) γ en fonction du numéro de l'essai et sa valeur moyenne (degrés)

(b) β en fonction du numéro de l'essai et sa valeur moyenne (degrés)

FIG. 4.17 – Résultats expérimentaux sur 25 essais — cas où trois points de l'objet sont connus —

x	$\overline{m} = \frac{1}{N}\sum_i x_i$	$\sup\lvert x_i\rvert$	σ	$\inf\lvert x_i\rvert$
γ	0.09	3.58	0.99	0.01
β	-0.53	2.47	0.91	0.04

TAB. 4.1 – Résultats expérimentaux sur 25 essais (en degrés) — cas où trois points de l'objet sont connus —

- facteur d'oubli pour les moindres carrés = 0.995 ;
- $K_y = 0.1$;
- nombre d'acquisitions d'apprentissage = 50 ;
- condition d'arrêt : $|\theta_{yc} - \theta_y| < 5e - 3$ et $|\theta_{yc,k} - \theta_{yc,k-1}| < 1e - 4$.

On peut remarquer que, conformément à 4.2.4.6, pour limiter les erreurs de positionnement, une faible valeur de ω_0 a été choisie. Pour la même raison, K_x et K_y sont faibles. En revanche, les mouvements de maximisation laissant théoriquement invariantes les informations visuelles, il ne subsiste en pratique qu'une faible erreur $\underline{s} - \underline{s}^*$ et en conséquence, de fortes valeurs de λ peuvent être utilisées. Par ailleurs, au cours de ces expérimentations on a pu remarquer que le mouvement de maximisation de h (figure 4.14a) comportait de légères oscillations en comparaison avec celui de l (figure 4.12a). Pour limiter ce phénomène, on a augmenté le nombre d'acquisition d'apprentissage (de 40 à 50), la valeur de λ (de 3 à 5) et le facteur d'oubli (de 0.99 à 0.995).

Compte tenu du volume très important dans lequel les positions initiales étaient choisies, les résultats sont globalement bons. Cependant, certaines positions conduisent à des erreurs importantes. Les causes d'erreur ont déjà été détaillées en 4.2.4.6 lors des simulations et se vérifient ici aussi. En effet, on a vu qu'il était important que l'erreur angulaire initiale soit importante une fois la phase de centrage réalisée (et non pas en position initiale avant le début du mouvement de centrage). Malheureusement, à partir d'une configuration initiale caméra / objet donnée, on ne peut pas évaluer cette erreur angulaire après centrage puisque, d'une part l'utilisation d'un algorithme de commande référencée vision ne permet pas de connaître la situation de la caméra une fois le centrage terminé et d'autre part la situation de l'objet est inconnue.

Par contre, au cours du mouvement, quand θ_{xc} devient stable, on connaît l'amplitude angulaire réalisée et, si elle s'avère trop faible, on peut décider de changer le sens de déplacement et imposer une valeur de θ_{xc} de l'ordre d'une dizaine de degrés. On considère ensuite la position courante ainsi obtenue comme la nouvelle position initiale à partir de laquelle pourra débuter, dans des conditions favorables, un nouvel apprentissage.

Par ailleurs, pour un objet donné, on peut expérimentalement définir grossièrement un volume de l'espace pour lequel une fois le centrage réalisé les erreurs angulaires initiales seront suffisantes.

4.3.2.3 Comportement de la loi de commande en environnement non calibré

Dans le but de comparer les performances de la méthode proposée par rapport à une démarche reconstructionniste, nous considérons maintenant le cas où la calibration réalisée est entachée d'erreurs (de la même façon qu'en 3.5.4.3).

Nous avons, dans le cas d'un ensemble robot/caméra calibré, réalisé une tâche de positionnement conformément à l'algorithme proposé et avons obtenu les angles d'Euler suivants :

$$\gamma = 0.12 \quad \beta = -0.49 \quad \alpha = -0.02 \text{ (en degrés)}$$

Nous avons tout d'abord considéré le cas d'une mauvaise identification des paramètres intrinsèques de la caméra en considérant les mêmes erreurs qu'en 3.5.4.3. À partir des mêmes positions du porteur et de l'objet, une fois la tâche de positionnement réalisée, nous avons obtenu cette fois les résultats suivants :

$$\gamma = 0.5 \quad \beta = -0.86 \quad \alpha = -0.06 \text{ (en degrés)}$$

La loi de commande s'avère donc peu sensible vis-à-vis des erreurs de calibration de la caméra.

Considérons maintenant le cas où la matrice M_{pc} a été incorrectement identifiée (toujours comme en 3.5.4.3). Après positionnement de la caméra, suivant la procédure décrite ci-dessus, les angles suivants ont été obtenus :

$$\gamma = -1.46 \quad \beta = 1.71 \quad \alpha = 0.19 \text{ (en degrés)}$$

L'erreur d'orientation est cette fois plus importante. En fait, le biais introduit dans la matrice M_{pc} peut intervenir de deux façons :

- lors du calcul du jacobien du robot ;
- lors de la reconstruction des angles θ_x et θ_y.

Pour analyser le comportement de la loi de commande dans ces deux cas, une matrice incorrectement identifiée a été utilisée pour le calcul du jacobien et par contre, une matrice correctement identifiée a été utilisée pour la reconstruction des angles. Dans ces conditions, les résultats sont les suivants :

$$\gamma = -1.61 \quad \beta = 1.84 \quad \alpha = 0.23 \text{ (en degrés)}$$

Par conséquent, contrairement à ce que l'on imaginait *a priori*, c'est moins la reconstruction des angles qui influe sur l'erreur d'orientation que le jacobien du robot. On peut maintenant comprendre pourquoi cette erreur apparaît. En effet, les mouvements imposés lors des phases de maximisation et que l'on souhaite faire appartenir au noyau de la matrice d'interaction n'y appartiennent pas. La tâche principale introduit alors des mouvements de compensation qui conduisent à une trajectoire différente de celle escomptée.

Finalement, pour apporter un réel plus à la méthode proposée par rapport à des méthodes de reconstruction 3D, il faudrait envisager dans le futur des mouvements réellement en boucle fermée et non plus des mouvements générés par la tâche secondaire qui sont, en fait, réalisés en boucle ouverte. Mais, comme on l'a vu en 4.2.3.2, cela semble difficile à réaliser.

Le comportement de l'algorithme étant maintenant connu, nous allons ci-après approfondir la notion de points caractéristiques, notion sur laquelle il repose, et proposer quelques méthodes d'extraction de ce type de points.

4.3.3 Quelques méthodes d'extraction de points caractéristiques

En fait, l'algorithme proposé ne dépend qu'indirectement de la forme de l'objet dans le sens où le problème réside plutôt dans l'extraction de points caractéristiques, images de points caractéristiques de l'objet, et c'est plutôt ce processus qui dépend directement du type d'objet. Néanmmoins, deux types d'approche émergent :

- Une première approche consiste à segmenter l'image en régions en regroupant des pixels ayant des caractéristiques homogènes d'intensité (voir par exemple [Monga 90] pour un exposé des différentes techniques d'extraction de régions). Cependant, on peut émettre certaines réserves à cette façon de procéder. En effet, il est primordial pour la bonne réalisation de la tâche, qu'un point extrait de l'image soit la projection d'un même point caractéristique de l'objet tout au long du mouvement. En d'autres termes, pour considérer la région obtenue par ce type de segmentation comme une information visuelle valide, il faudra qu'elle puisse être assimilée à un point matériel. Par conséquent, seules des régions de faibles surfaces pourront être retenues.

- Une seconde approche consiste à extraire de l'image des *points d'intérêts*.

Ces points correspondent à un changement bidimensionnel du signal d'intensité comme par exemple les coins, les jonctions mais aussi d'autres points de l'image. Nous conseillons vivement au lecteur la lecture de la thèse de C. Schmid où un état de l'art complet sur ce sujet est présenté [Schmid 96]. Elle propose de classer les méthodes d'extraction existantes en trois grandes catégories :

- Les méthodes basées sur les contours. Elles consistent à extraire ce type de points le plus souvent grâce à la courbure. Citons par exemple [Asada 86] où les changements de courbure sont exploités pour la détection de coins ou de terminaisons, [Kim 88] et [Byun 96] en ce qui concerne les extrema de courbure et [Mokhtarian 86] pour ce qui est des passages à zéro. Cependant, pour pouvoir considérer ce type de point comme une information visuelle valide, il nous faut nous assurer que la propriété choisie (changement, extrema ou zéro de courbure) se conserve par projection perspective. Par exemple, les extrema de courbure utilisés pour la mise en correspondance en stéréovision, comme dans [Kim 88] ou [Byun 96], ne sont conservés que pour une petite région de l'espace [3]. Par ailleurs, on peut montrer, que dans les cas non dégénérés (c'est-à-dire quand l'image de l'objet plan n'est pas une droite), les points de courbure nulle ou points d'inflexion sur l'objet sont aussi des points d'inflexion dans l'image, certains auteurs utilisent cette propriété [Mokhtarian 86]. De la même façon, on peut montrer que les points stationnaires sont conservés par projection perspective. Une démonstration est proposée dans les deux cas en annexe B.
- Les méthodes basées sur l'intensité lumineuse. Elles sont l'objet de nombreux travaux [Moravec 80, Kitchen 82, Dreschler 82, Harris 88, Förstner 94]. Toutefois, pour que ces points puissent être considérés comme des points caractéristiques deux contraintes principales doivent être vérifiées :
  - Comme ces points sont calculés à partir de la mesure de l'intensité lumineuse, il nous faut vérifier l'invariance de cette intensité par transformation perspective. En fait, seuls des objets caractérisés par une *surface lambertienne*[4] vérifient cette propriété. Néanmoins, il existe d'autres types de surface pour lesquelles l'intensité lumineuse dans l'image varie peu en fonction de la position de la caméra (voir les travaux de B.K.P. Horn sur le sujet :

[3]Généralement, en stéréovision, les deux caméras sont suffisamment proches les unes des autres pour que les extrema de courbure correspondent bien dans les deux images au même point sur l'objet.

[4]Ces objets possèdent alors des surfaces complètement mates.

[Horn 74, Horn 77]). Cette contrainte est donc très liée à la nature de l'objet.

  - La deuxième contrainte concerne *la répétabilité* des détecteurs de points d'intérêts. En effet, il est impératif, pour une utilisation en asservissement visuel, que ces algorithmes soient capables de reconnaître les mêmes points d'une image à l'autre en cas de changement des conditions d'acquisition des images. Dans [Schmid 96] une évaluation comparative de plusieurs détecteurs a été effectuée face à une rotation de la caméra, à un changement d'échelle ou de luminosité ou encore de point de vue. Dans tous les cas, le détecteur de Harris modifié (voir [Schmid 96] ou [Schmid 98]) a donné les meilleurs résultats.

- La dernière catégorie de méthodes d'extraction de points d'intérêts concerne celles basées sur un modèle théorique dans l'image, comme par exemple un modèle de coin ou de jonctions de plusieurs lignes. Nous ne détaillerons pas plus ces différents algorithmes dans la mesure où ils sont basés sur des hypothèses concernant la forme de l'objet. Toutefois, nous renvoyons le lecteur intéressé aux travaux suivants : [Rhor 92, Deriche 93, Brand 94].

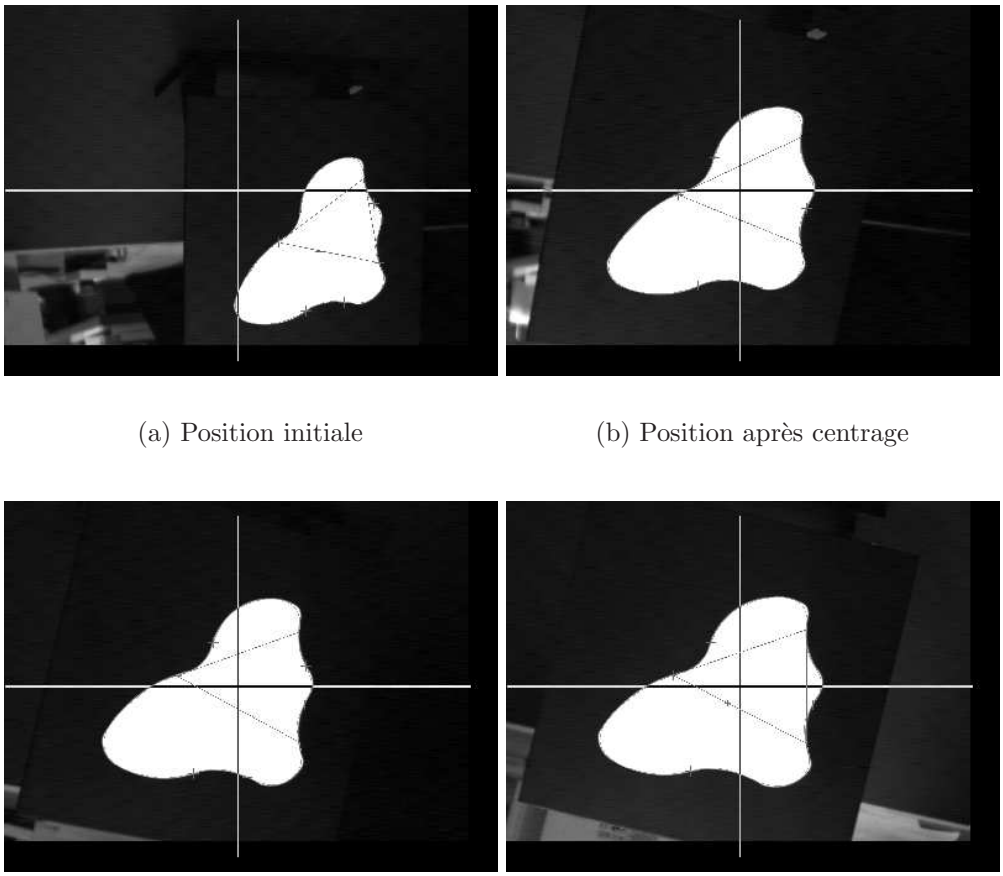
Après avoir décrit quelques approches possibles d'extraction de points caractéristiques, nous allons maintenant illustrer une de ces méthodes pour la réalisation de notre tâche particulière de positionnement sur un objet de forme complexe.

4.3.3.1 Application à un objet plan de forme complexe

Dans l'exemple proposé, l'objet de type a, huit points d'inflexion peuvent être extraits de l'image à partir du contour. Nous en choisissons trois suffisamment éloignés les uns des autres. La figure 4.18 résume les différentes phases de l'algorithme, les figures 4.19, 4.20, 4.21 et 4.22 décrivent son comportement. Afin de montrer le parallèle avec les résultats obtenus en 4.3.2.2 (voir la figure 4.16) les points caractéristiques ont été reliés pour mettre en valeur le triangle ainsi formé. Les résultats suivant ont été obtenus :

- Position initiale de la caméra :
  - Angles d'Euler (en degrés) :

$$\gamma = 5.34 \pm 0.01 \quad \beta = 24.39 \pm 0.02 \quad \alpha = -1.2 \pm 0.01$$



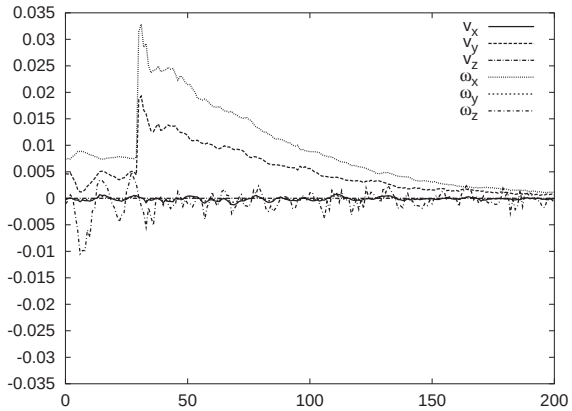
(a) Position initiale

(b) Position après centrage

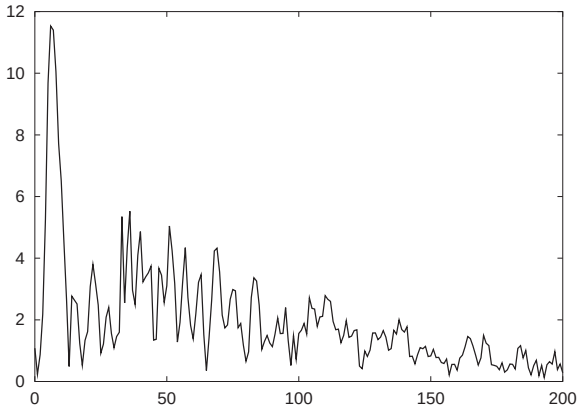
(c) Position après maximisation de l

(d) Position finale

FIG. 4.18 – Résultats exprérimentaux — cas où l'objet présente trois points caractéristiques —

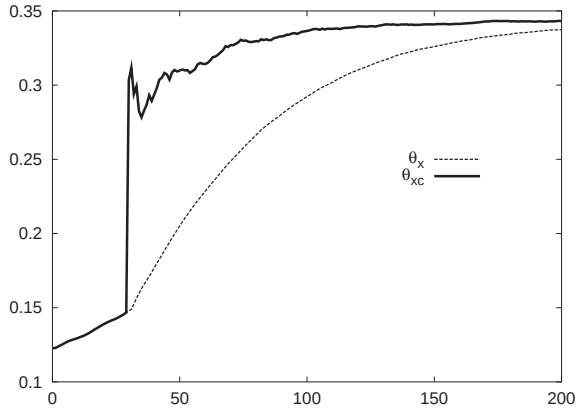


(a) Composantes du torseur cinématique (en m/s ou rad/s) en fonction du nombre d'itérations

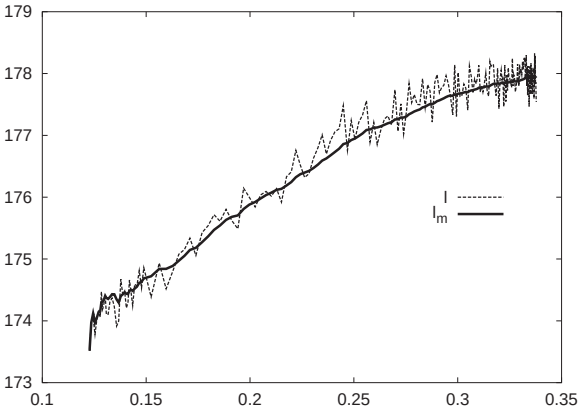


(b) Erreur dans le capteur (en pixel) en fonction du nombre d'itérations

FIG. 4.19 – Résultats expérimentaux — Maximisation de l — cas où l'objet présente trois points caractéristiques

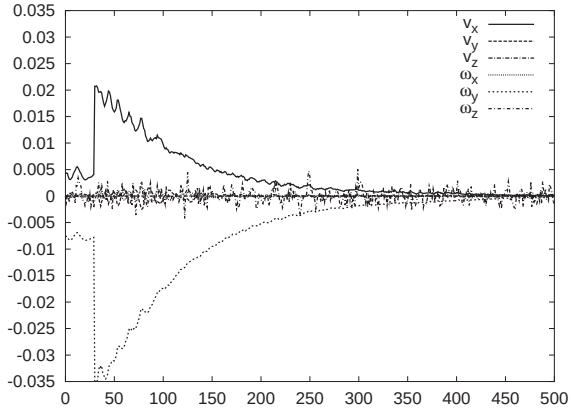


(a) Consigne angulaire et position angulaire désirée (en rad.) en fonction du nombre d'itérations

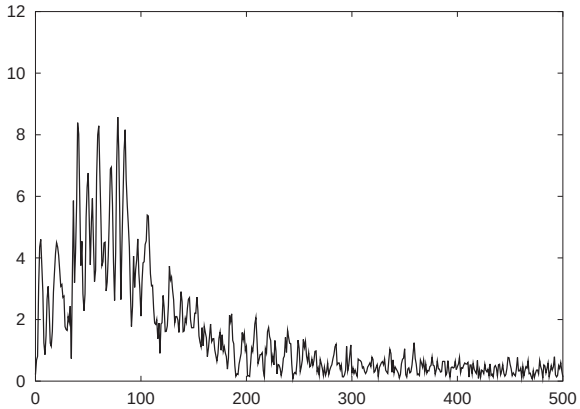


(b) Taille de l'image réelle et désirée en fonction de θ_x (en pixel)

FIG. 4.20 – Résultats expérimentaux — Maximisation de l — cas où l'objet présente trois points caractéristiques

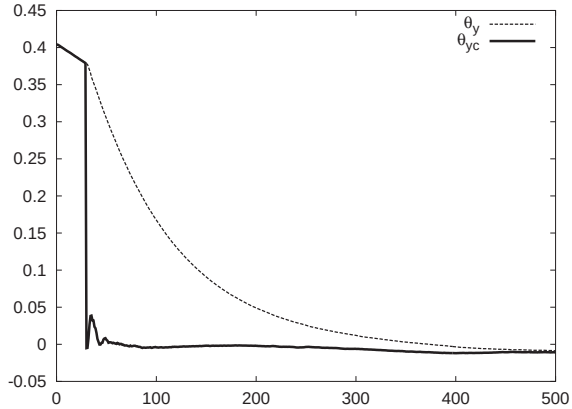


(a) Composantes du torseur cinématique (en m/s ou rad/s) en fonction du nombre d'itérations

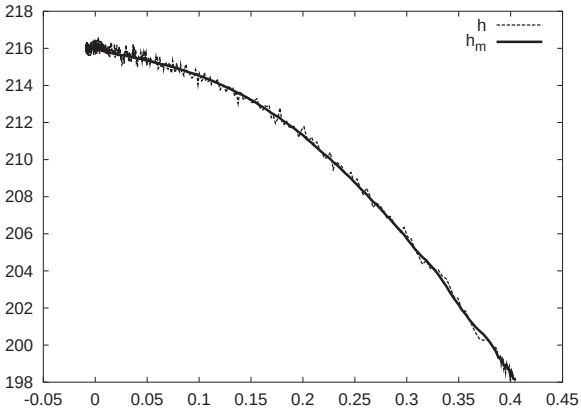


(b) Erreur dans le capteur (en pixel) en fonction du nombre d'itérations

FIG. 4.21 – Résultats expérimentaux — Maximisation de h — cas où l'objet présente trois points caractéristiques



(a) Consigne angulaire et position angulaire désirée (en rad.) en fonction du nombre d'itérations



(b) Taille de l'image réelle et désirée en fonction de θ_y (en pixel)

FIG. 4.22 – Résultats expérimentaux — Maximisation de h — cas où l'objet présente trois points caractéristiques

– Coordonnées des points caractéristiques (exprimées dans le repère de la caméra en mm) :

$$\begin{aligned} m_1 &= \begin{pmatrix} 112.89 \pm 0.04 & 95.46 \pm 0.02 & 720.28 \pm 0.15 \end{pmatrix}^T \\ m_2 &= \begin{pmatrix} 115.20 \pm 0.04 & -34.05 \pm 0.02 & 709.25 \pm 0.15 \end{pmatrix}^T \\ m_3 &= \begin{pmatrix} -4.31 \pm 0.02 & 33.16 \pm 0.04 & 661.06 \pm 0.16 \end{pmatrix}^T \end{aligned}$$

- Position finale de la caméra :

  – Angles d'Euler (en degrés) :

$$\gamma = 0.45 \pm 0.00 \quad \beta = 0.44 \pm 0.01 \quad \alpha = 13.53 \pm 0.01$$

  – Coordonnées des points caractéristiques (exprimées dans le repère de la caméra en mm) :

$$\begin{aligned} m_1 &= \begin{pmatrix} 35.65 \pm 0.02 & 73.84 \pm 0.01 & 557.13 \pm 0.06 \end{pmatrix}^T \\ m_2 &= \begin{pmatrix} 66.07 \pm 0.02 & -52.54 \pm 0.02 & 556.09 \pm 0.06 \end{pmatrix}^T \\ m_3 &= \begin{pmatrix} -75.53 \pm 0.01 & -19.76 \pm 0.14 & 555.60 \pm 0.07 \end{pmatrix}^T \end{aligned}$$

Par conséquent, nous obtenons dans le cas d'un objet de forme complexe de bons résultats même si l'amplitude de variation de l n'est pas très importante (voir la figure 4.20b). Une nouvelle fois on peut s'apercevoir à la vue des figures 4.20a et 4.22a qu'une consigne angulaire stable est disponible rapidement.

Par ailleurs, pour cette expérimentation, quelques valeurs des paramètres ont été changées par rapport à celles fixées en 4.3.2.2. Le bruit introduit par le traitement de l'image étant plus important, le facteur d'oubli a été augmenté ($\gamma = 0.995$). En outre, la durée de ce traitement étant aussi plus importante, Δt est passé à 80 ms. Pour cette raison λ a du être diminué, il a été fixé à 2. Pour terminer, l'objet étant plus large que haut, l_c a été choisi à 200 pixels pour s'assurer que l'image contienne la totalité de l'objet.

4.3.4 Étude d'un objet plan de forme quelconque

On a vu dans le paragraphe 4.3.1.2 qu'il n'était pas possible de trouver une expression analytique de la surface de l'objet dans l'image en fonction de la situation de la caméra, mais que l'on pouvait aboutir à la position souhaitée grâce à deux mouvements successifs. On va chercher à appliquer ici la même procédure, mais sans utiliser de points caractéristiques.

4.3.4.1 Choix des informations visuelles et obtention d'une loi de commande

Considérons dans le capteur une forme *a priori* quelconque et quatre points M_i pour $i = 1, \ldots, 4$ résultant de l'intersection du contour et des axes principaux d'inertie de cette forme (voir la figure 4.23). Si les points M_1 et M_2 d'une part appartenaient à un objet 1 et les points M_3 et M_4 d'autre part appartenaient à un objet 2, on choisirait d'après le paragraphe 4.2.2.2 les informations visuelles suivantes :

$$\left\{ \begin{array}{rcl} X_1 - X_2 & = & 0 \\ Y_1 + Y_2 & = & 0 \end{array} \right. \tag{4.87}$$

et

$$\left\{ \begin{array}{rcl} Y_3 - Y_4 & = & 0 \\ X_3 + X_4 & = & 0 \end{array} \right. \tag{4.88}$$

Dans le cas d'un seul et même objet et du fait que les axes principaux d'inertie sont orthogonaux, la condition $X_1 - X_2 = 0$ entraîne $Y_3 - Y_4 = 0$ et par conséquent on peut choisir pour informations visuelles :

$$\underline{s} = (X_1 - X_2, X_3 + X_4, Y_1 + Y_2)^T \tag{4.89}$$

et :

$$\underline{s}^* = (0, 0, 0)^T \tag{4.90}$$

La matrice d'interaction calculée en $\underline{s}^*$, $Y_4 = Y_3$ et en $z_i = z$ s'écrit :

$$L^T_{\substack{\underline{s}^* \\ Y_4 = Y_3 \\ z_i = z}} = \begin{pmatrix} 0 & 0 & 0 & 2X_1Y_1 & 0 & 2Y_1 \\ -2/z & 0 & 0 & 0 & -2\left(1 + {X_3}^2\right) & 2Y_3 \\ 0 & -2/z & 0 & 2\left(1 + {Y_1}^2\right) & 0 & -2X_1 \end{pmatrix} \tag{4.91}$$

On s'intéresse une nouvelle fois aux vecteurs $\underline{T}_{sl}$ et $\underline{T}_{sh}$ appartenant au noyau de $L^T_{|\underline{s}^*, Y_4 = Y_3, z_i = z}$:

$$\underline{T}_{sl} = \begin{pmatrix} -z\omega_x X_1 Y_3 & z\omega_x \left(1 + {X_1}^2 + {Y_1}^2\right) & 0 & \omega_x & 0 & -\omega_x X_1 \end{pmatrix}^T \tag{4.92}$$

$$\underline{T}_{sh} = \begin{pmatrix} -z\omega_y \left(1 + {X_3}^2\right) & 0 & 0 & 0 & \omega_y & 0 \end{pmatrix}^T \tag{4.93}$$

Ils doivent permettre, comme en 4.3.1.2, au cours des mouvements qu'ils engendrent la maximisation tour à tour de l $(Y_1 - Y_2)$ puis h $(X_3 - X_4)$. Une fois la tâche réalisée, on doit obtenir une image du même type que celle qui est représentée sur la figure 4.23. Remarquons que le centre de gravité C_g de l'image de l'objet ne coïncide pas, dans le cas général, avec le centre du capteur.

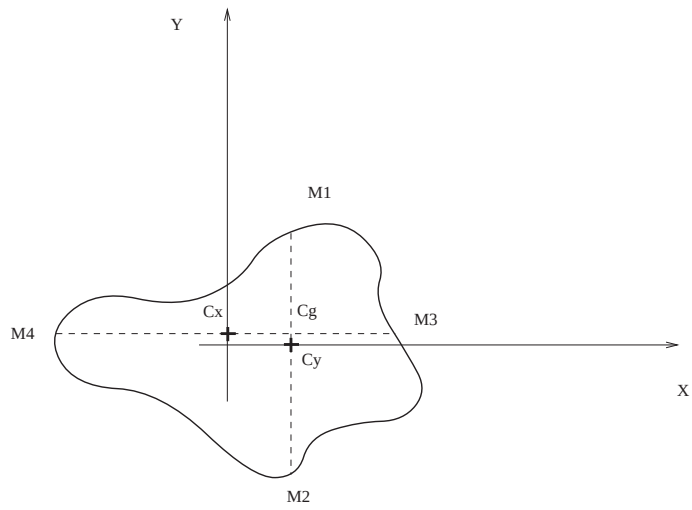


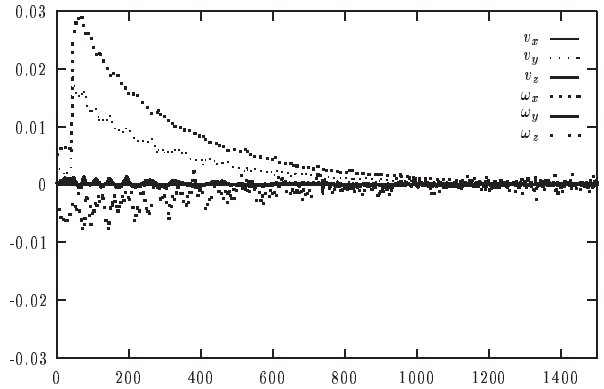
FIG. 4.23 – Objet plan de forme quelconque — Motif désiré —

4.3.4.2 Résultats expérimentaux

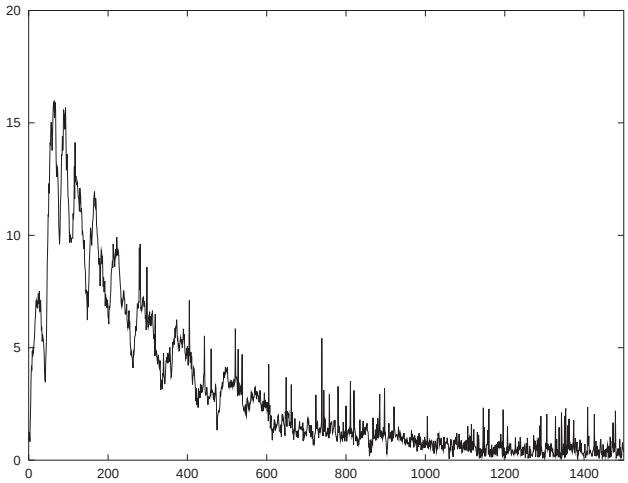
Les figures 4.24a, 4.25a, 4.25b et 4.24b montrent les comportements respectifs des composantes du torseur cinématique, de la consigne angulaire et de la position angulaire courante, de la taille de l'image réelle et modélisée et de l'erreur dans le capteur lors de la maximisation de l. Les figures 4.26a, 4.27a, 4.27b et 4.26b représentent l'évolution des mêmes variables que précédemment mais lors de la maximisation de h. La figure 4.28 résume les différentes phases nécessaires à la réalisation de la tâche de positionnement souhaitée.

Le comportement de la loi de commande est grossièrement identique à celle du paragraphe 4.3.2.2. Cependant, lors de la maximisation de l, l'erreur dans le capteur (figure 4.24b) est cette fois beaucoup plus importante et, de plus ne commence à tendre vers zéro que plus tardivement. Cela se comprend aisément. En effet, les informations visuelles sont construites à partir des coordonnées (X_i, Y_i) de points qui ne représentent pas, à chaque itération, les images des mêmes points physiques puisque ces informations dépendent de la forme de l'image de l'objet. Par conséquent, les mouvements du noyau de $L^T_{s^*,Y_4=Y_3,z_i=z}$ ne laissent pas les informations visuelles invariantes [5] et, il s'en suit une erreur dans l'image importante. Cette erreur conduit à des commandes de compensation provenant de la tâche principale et donc à une trajectoire différente de celle attendue.

[5]Les informations visuelles seraient invariantes si, au cours du mouvement, les axes d'inertie avaient une orientation constante.

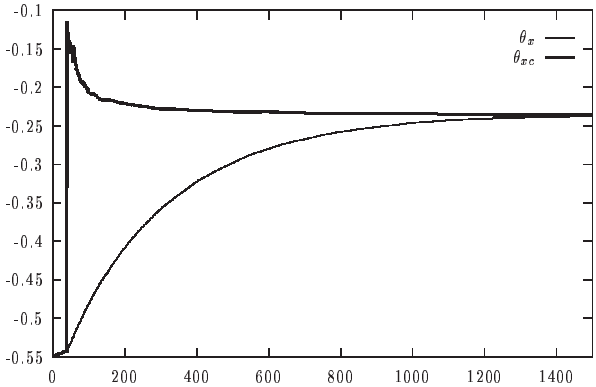


(a) Composantes du torseur cinématique (m/s ou rad/s) en fonction du nombre d'itérations

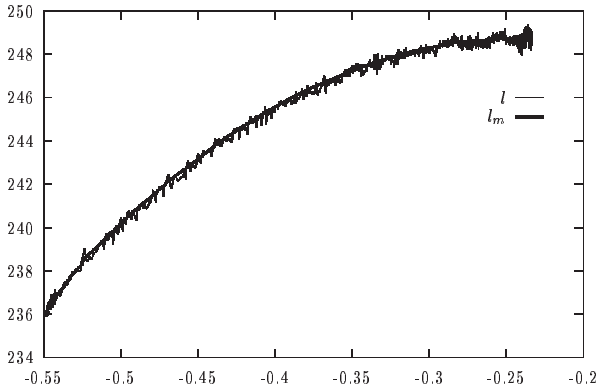


(b) Erreur dans le capteur (pix.) en fonction du nombre d'itérations

FIG. 4.24 – Résultats expérimentaux — cas d'un objet quelconque — Maximisation de l

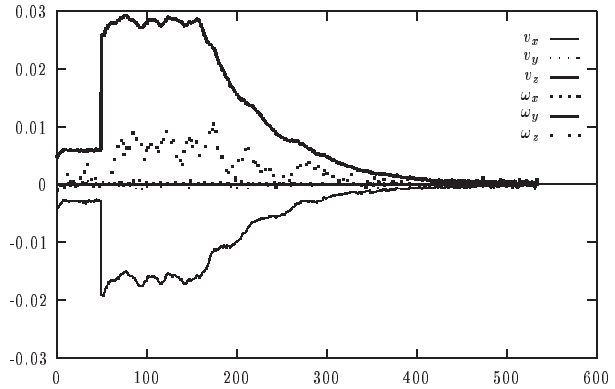


(a) Consigne angulaire (rad.) et position angulaire courante (rad.) en fonction du nombre d'itérations

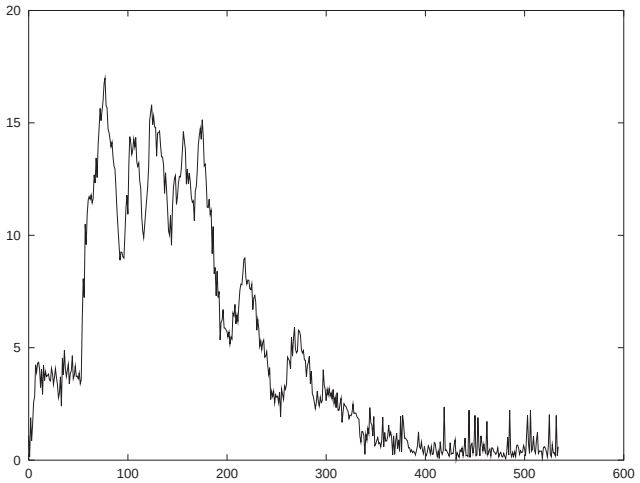


(b) Taille de l'image et taille modélisée (pix.) en fonction de θ_x

FIG. 4.25 – Résultats expérimentaux — cas d'un objet quelconque — Maximisation de l

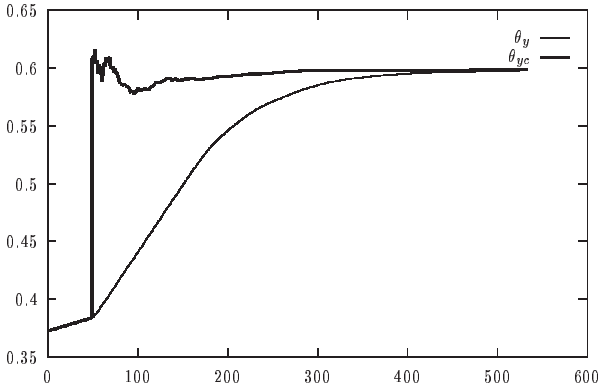


(a) Composantes du torseur cinématique (m/s ou rad/s) en fonction du nombre d'itérations

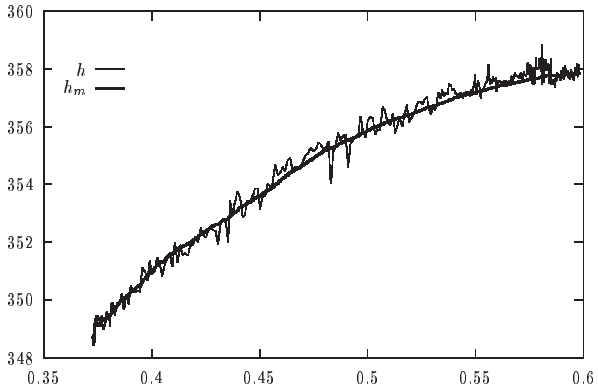


(b) Erreur dans le capteur (pix.) en fonction du nombre d'itérations

FIG. 4.26 – Résultats expérimentaux — cas d'un objet quelconque — Maximisation de h

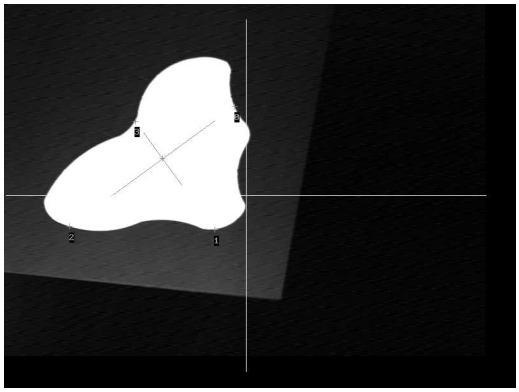


(a) Consigne angulaire (rad.) et position angulaire courante (rad.) en fonction du nombre d'itérations

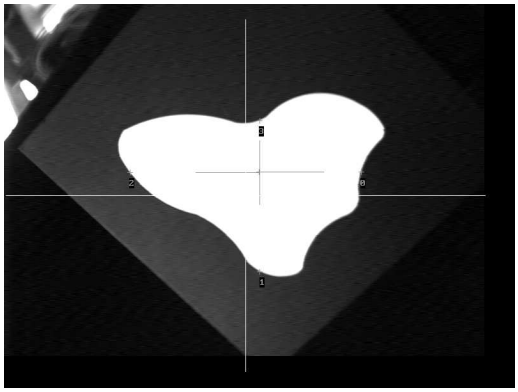


(b) Taille de l'image et taille modélisée (pix.) en fonction de θ_y

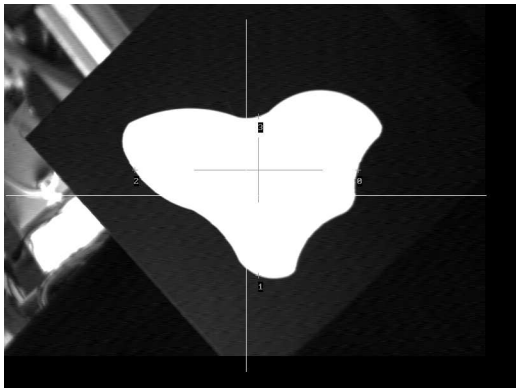
FIG. 4.27 – Résultats expérimentaux — cas d'un objet quelconque — Maximisation de h



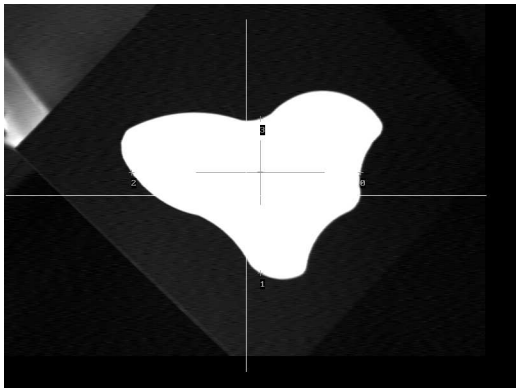
(a) Position initiale



(b) Position après centrage



(c) Position après maximisation de l



(d) Position finale

FIG. 4.28 – Résultats expérimentaux — cas d'un objet quelconque —

Dans le cas de la maximisation de h apparaît le même phénomène. Il est clairement mis en évidence sur la figure 4.26a où des composantes non nulles suivant v_y, ω_x et ω_z existent contrairement à (4.93).

Pour quantifier plus précisément la qualité du positionnement obtenu, on a procédé comme suit. Tout d'abord, un objet plan constitué de quatre points marqués a été placé au même endroit que l'objet quelconque pour localiser la position de la caméra par rapport à l'objet suivant la même procédure qu'en 3.5.4.2. Puis, une fois la tâche de positionnement réalisée, l'objet quelconque a été remplacé par l'objet marqué pour localiser une nouvelle fois la caméra par rapport à l'objet. Les résultats obtenus sont les suivants :

- Position initiale de la caméra par rapport à l'objet :
  - Angles d'Euler (en degrés) :

$$\gamma = 16.13 \pm 0.01 \quad \beta = 9.05 \pm 0.02 \quad \alpha = 16.56 \pm 0.03$$

- Position finale de la caméra par rapport à l'objet :
  - Angles d'Euler (en degrés) :

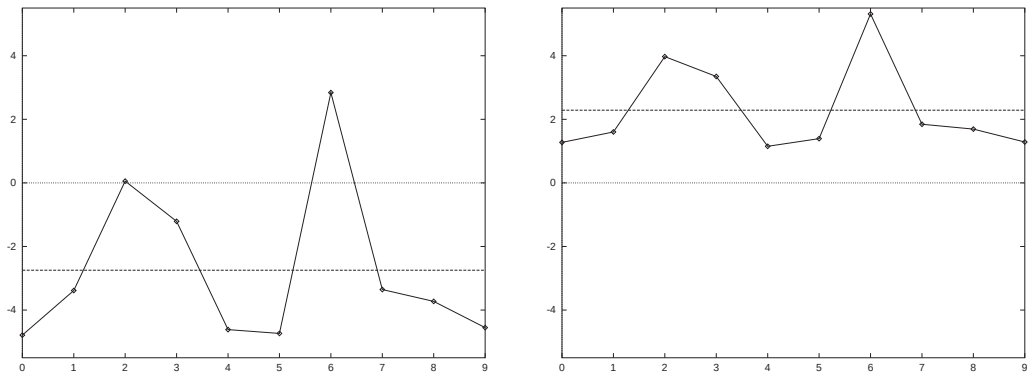
$$\gamma = 0.05 \pm 0.01 \quad \beta = 3.9 \pm 0.01 \quad \alpha = 45.10 \pm 0.03$$

Nous avons réalisé ensuite une série de 10 essais pour différentes positions initiales. Les résultats sont consignés d'une part dans le tableau 4.2 et d'autre part sur la figure 4.29 où le numéro de l'essai est porté en abscisse et les valeurs moyennes de γ et β sur les 100 images en ordonnée. Ces résultats nous confirment que le positionnement effectué est moins précis que dans le cas où trois points caractéristiques de l'objet sont disponibles dans l'image. Cependant, nous tenons à signaler qu'aucune hypothèse n'est requise quant à la géométrie de l'objet où à sa position relative par rapport à la caméra si ce ne sont les restrictions, décrites en 4.2.4.6, influant sur l'erreur de positionnement.

Par conséquent, dans le cas où l'on souhaite positionner précisément une caméra en face d'un objet plan de forme complexe, on privilégiera l'approche extraction de trois points caractéristiques physiques où, comme on l'a vu en 4.3.2.2, des performances importantes en terme d'erreur de positionnement peuvent être atteintes.

x	$\overline{m} = \frac{1}{N}\sum_i x_i$	$\sup\lvert x_i\rvert$	σ	$\inf\lvert x_i\rvert$
γ	-2.75	4.79	3.28	0.05
β	2.29	5.31	1.35	1.15

TAB. 4.2 – Résultats expérimentaux sur 10 essais (en degrés) — cas d'un objet quelconque —



(a) γ en fonction du numéro de l'essai et sa valeur moyenne (rad)

(b) β en fonction du numéro de l'essai et sa valeur moyenne (rad)

FIG. 4.29 – Résultats expérimentaux sur 10 essais — cas d'un objet quelconque —

4.4 Conclusion

Nous avons présenté dans ce chapitre la commande d'un robot par asservissement visuel 2D à l'aide d'informations très simples et cela dans le cas où l'objet observé est plan, de forme quelconque et immobile. Une attention particulière a été portée au cas où trois points caractéristiques peuvent être extraits de l'image. Une discussion concernant les erreurs de positionnement rencontrées a conclu qu'elles pouvaient être limitées dans la mesure où un pré-positionnement relativement grossier de la caméra était possible.

De plus, nous avons approfondi la notion de points caractéristiques et avons montré par la même occasion que des informations visuelles issues de l'image du contour de l'objet étaient valides et réalistes, comme l'ont montré les résultats expérimentaux.

Pour terminer, nous avons étudié le cas d'un objet de forme quelconque, pour lequel aucune hypothèse n'est requise. Les résultats expérimentaux ont montré que la réalisation de la tâche de positionnement souhaitée était possible, toutefois avec une erreur de positionnement plus importante.

Nous regrettons cependant de n'avoir pu réaliser simultanément les phases de maximisation et pensons qu'il serait très intéressant dans le futur de s'y pencher une nouvelle fois. Peut-être faudrait-il songer à une modélisation locale de la surface $S(\theta_x, \theta_y)$ quitte à ce que l'algorithme soit plus sensible aux bruits de mesure des informations visuelles.

D'un autre point de vue, il serait intéressant d'étudier le cas d'un objet comportant un nombre de points caractéristiques supérieur à 3. D'une part, pour limiter l'influence du bruit de mesure sur l'algorithme et, d'autre part, pour réaliser la tâche considérée par rapport à un objet non plan. L'objectif serait alors de se positionner par rapport à un plan moyen passant le plus près possible de l'ensemble des points caractéristiques. Rappelons qu'actuellement nous avons supposé l'objet plan, non pas pour des raisons d'obstacle à la convergence des lois de commande, mais par souci de réalisme vis-à-vis du problème posé. En effet, il serait curieux de spécifier la tâche en imposant le parallélisme du plan image avec celui formé par 3 points quelconques de l'objet.

Conclusion et perspectives

Partis du constat que le domaine applicatif des asservissements visuels 2D était limité par la connaissance précise d'informations concernant la géométrie de l'objet observé, nous avons orienté nos travaux dans deux directions que nous rappelons ci-dessous.

Nous avons developpé au chapitre 3 une approche, basée sur des informations visuelles issues du contour de l'image de l'objet, permettant de s'affranchir d'un modèle 3D de cet objet, comme c'était habituellement le cas. Pour ce faire, nous avons dissocié la primitive 2D de sa primitive 3D. Cette façon de procéder nous a permis de calculer une approximation de la matrice d'interaction dans le cas d'un objet de forme complexe. Ce calcul est basé sur la minimisation d'une erreur de modélisation du contour. Cette approche a ensuite été appliquée avec succès aux cas de la signature polaire et des fonctions paramétriques, en portant une attention particulière aux descripteurs de Fourier. Les résultats sont satisfaisants, même s'ils sont plus nuancés concernant ce dernier type de représentation.

Concernant une comparaison de l'approche asservissement visuel basé contour et de l'approche basée sur des primitives points, nous avons rapporté en 3.8 certains avantages à l'utilisation de notre approche. Rappelons les brièvement. Tout d'abord, on peut noter la faible influence du bruit sur les informations visuelles, due pour beaucoup à la méthode d'identification du modèle du contour. D'autre part, nous avons montré qu'une telle approche était peu sensible aux occlusions partielles de l'objet, contrairement à une approche de type point. De plus, nous avons souligné la complexité du problème d'appariement nécessaire entre l'image de consigne et l'image initiale, problème évité par notre approche. Pour terminer, concernant le processus d'extraction des primitives points, nous avons signalé en 3.8 que les algorithmes employés étaient encore sensibles aux modifications de l'image introduites par le mouvement de la caméra alors qu'une détection de contour, même sommaire, est suffisante que ce soit dans l'approche polaire ou paramétrique puisque l'on ne cherche pas une modélisation précise de ce contour.

L'utilisation de la signature polaire a permis, non seulement, de réaliser des tâches de positionnement de façon très précise par rapport à des objets plans ou non et de formes complexes mais, aussi, sans connaissance de la forme de cet objet ou d'une localisation précise. En fait, seule est requise une image souhaitée de l'objet considéré et la distance approximative (à approximativement $\pm$ 35 cm) à laquelle elle a été acquise. De plus, les résultats expérimentaux ont montré que cette loi de commande était robuste vis-à-vis de l'estimation réalisée sur les profondeurs mais aussi des occlusions partielles ou d'un dispositif robot et capteur mal calibré.

En outre, bien qu'aucune simulation ou expérimentation n'ait été réalisée, l'approche proposée devrait pouvoir appréhender plus généralement le cas des tâches de suivi. De même, elle doit pouvoir traiter le cas d'informations visuelles provenant d'un contour ouvert.

Par ailleurs, d'un point de vue théorique, il est intéressant de remarquer l'influence du nombre d'harmoniques, et par conséquent du nombre d'informations visuelles, sur le phénomème des minima locaux. En effet, les résultats expérimentaux ont montré qu'en augmentant leur nombre, la convergence était obtenue vers le minimum global désiré. De plus, cette observation tend à prouver que les informations visuelles choisies ne sont pas forcément pertinentes dans le sens où une convergence à coup sûr n'est garantie que pour un nombre élevé d'harmoniques. Une information visuelle à elle seule ne renferme donc pas beaucoup d'informations concernant la forme du contour.

Toutefois, augmenter le nombre d'harmoniques conduit nécessairement à augmenter aussi la durée des calculs et, par la même occasion, l'intervalle de temps entre chaque acquisition d'image. L'utilisation d'un capteur comme celui développé à l'université de Gênes par le professeur Sandini [Tistarelli 93] serait le bien venu. En effet, il fournit directement les coordonnées polaires du point observé !

Dans le même ordre d'idée, il serait beaucoup plus judicieux, dans le futur, de combiner le calcul des informations visuelles issues du contour avec celui de sa détection. Une approche particulièrement bien adaptée serait d'utiliser les modèles déformables tels que ceux décrits dans la thèse de B. Bascle [Bascle 94]. En effet, ces travaux traitent le cas particulier, qui nous occupe aussi, d'une séquence d'images. Elle y développe, par exemple, des modèles déformables paramétriques. Notre approche basée sur les descripteurs de Fourier doit s'y intégrer sans problème particulier. De même, le développement d'un modèle déformable « polaire » devrait aboutir.

Pour terminer ces quelques perspectives de recherche concernant ce chapitre,

il en est une, particulièrement intéressante, concernant l'utilisation d'un modèle de contour basé sur des fonctions paramétriques rationnelles. Nous avons vu en 3.4.3.2 que ce type de fonction permet, suivant le degré des fonctions polynômes, de représenter soit des fonctions B-spline rationnelles, soit des courbes de Bézier rationnelles, ou soit encore des fonctions que l'on a appelées des descripteurs de Fourier rationnels. Cette capacité à modéliser des formes quelconques est déjà en soi un atout mais elles jouissent, de plus, de la propriété d'invariance de forme par transformation perspective. En effet, la projection perspective d'une courbe rationnelle reste une courbe rationnelle. Par conséquent, ce type de représentation conduit à une matrice d'interaction non approximée. Cependant, comme nous l'avons vu dans le cas des descripteurs de Fourier, la principale hypothèse sur laquelle repose le calcul de la matrice d'interaction n'est pas toujours valide. Finalement, quel peut-être alors l'intérêt de poursuivre dans cette direction ?

Supposons que l'on puisse, d'une manière ou d'une autre assurer effectivement l'égalite $\dot{u} = 0$; par exemple, en marquant tous les points du contour. La matrice d'interaction obtenue serait alors parfaitement valide. Disposant, en outre, d'une description paramétrique de ce contour, il est alors possible de calculer analytiquement d'autres informations visuelles caractérisant la globalité de l'image. Citons par exemple les plus connus : les coordonnées du centre de gravité, la surface, les moments centrés d'ordre $p + q$. Notons $\underline{P}'$ ce paramétrage, $\underline{P}$ celui obtenu grâce à la description paramétrique du contour et calculons la matrice d'interaction relative aux informations visuelles décrites par $\underline{P}'$:

$$L^T_{\underline{P}'} = \frac{\partial \underline{P}'}{\partial \underline{r}} = \frac{\partial \underline{P}'}{\partial \underline{P}} \frac{\partial \underline{P}}{\partial \underline{r}} = \frac{\partial \underline{P}'}{\partial \underline{P}} L^T_{\underline{P}}$$

Cette expression fait intervenir, d'une part, la matrice $L^T_{\underline{P}}$, qui est connue, et, d'autre part, le terme $\partial \underline{P}'/\partial \underline{P}$ qui s'obtient de façon triviale. Le calcul de cette nouvelle matrice ne pose donc pas, pour l'instant, de gros problèmes. Cependant, il reste à exprimer chaque information visuelle $\underline{P}_i$ en fonction de $\underline{P}'$ et à les injecter dans $L^T_{\underline{P}}$. En revanche, cette étape est beaucoup plus délicate, les relations liant $\underline{P}$ et $\underline{P}'$ étant vraisemblablement non linéaires. Toutefois, l'enjeu est de taille, on obtiendrait ainsi une matrice d'interaction non approximée relative à des informations visuelles extractables très facilement à la cadence vidéo et caractérisant globalement l'image. On s'affranchirait ainsi totalement de la géométrie de l'objet. Pour terminer, il faudra s'assurer que ces nouvelles informations visuelles sont pertinentes. Il reste donc fort à faire avant de savoir si cette approche conduira à des résultats très intéressants ou si, au contraire, elle devra être abandonnée.

D'un tout autre point de vue, nous avons décrit, dans le chapitre 4, une façon de réaliser la tâche particulière de positionnement consistant à amener le

plan image parallèlement à celui de l'objet. L'originalité de ces travaux est de supposer la consigne visuelle, la situation de l'objet observé et ses dimensions inconnues. Par contre, nous supposons cet objet plan et immobile. Nous avons traité dans un premier temps le cas d'un objet filiforme et avons ensuite montré comment l'approche retenue peut se généraliser à un objet présentant trois points caractéristiques, puis à un objet quelconque, pour lequel aucune hypothèse n'est requise.

Des résultats, plus particulièrement bons dans le cas de trois points caractéristiques, ont pu être obtenus dès qu'un prépositionnement grossier de la caméra par rapport à l'objet est possible. Toutefois, l'approche proposée a montré des limites en terme de précision de positionnement en cas d'un robot mal calibré. Signalons, cependant, que cette précision n'est pas afffectée par une mauvaise calibration de la caméra.

Pour terminer ce chapitre, nous avons débattu de la notion de points caractéristiques et avons présenté le cas particulier des points d'inflexion lorsque seul le contour permet d'exhiber de tels points. D'une façon générale, l'extraction des points caractéristiques, à la cadence vidéo et sur objet réel, devra être optimisée dans le futur. En effet, dans le cas particulier de la réalisation d'une tâche par asservissement visuel, on dispose d'une estimation de la vitesse du capteur et cette donnée peut aider grandement le processus d'extraction des informations visuelles, en particulier lors de la phase d'appariement de ces informations entre les images précédente et courante.

Concernant l'algorithme utilisé, ne pourrait-on pas tenter de maximiser la surface de l'objet ? On a vu qu'une modélisation globale de l'image conduisait à des calculs analytiques trop complexes pour aboutir. Par contre, une modélisation plus locale est possible. Certes, cette façon de faire serait plus sensible au bruit de mesure des informations visuelles mais elle permettrait de réaliser la tâche plus rapidement et peut-être, selon les applications, avec suffisamment de précision.

De plus, comme nous l'avons signalé en 4.4, il serait intéressant d'étudier le cas d'un objet comportant un nombre de points caractéristiques supérieur à 3. D'une part, pour limiter l'influence du bruit de mesure sur l'algorithme et, d'autre part, pour réaliser la tâche considérée par rapport à un objet non plan en cherchant à positionner le capteur par rapport au plan moyen passant le plus près possible de l'ensemble des points caractéristiques.

Toujours dans le cas où l'image souhaitée est inconnue, on peut être tenté d'utiliser une modélisation du contour basée sur sa signature polaire. L'idée est issue de l'observation suivante : la caméra étant positionnée parallèlement à un

objet plan, un mouvement de translation dans la direction de l'objet transforme l'image d'un facteur d'échelle fonction de l'amplitude de ce déplacement. En normalisant la signature polaire, les informations visuelles deviennent alors invariantes par ce déplacement et, par conséquent, leur dérivée par rapport au temps est nulle. Le vecteur formé de la dérivée de ces informations pourrait constituer alors les nouvelles informations visuelles, dynamiques cette fois. Il ¡¡ resterait ¿¿ ensuite à obtenir la matrice d'interaction afin de réaliser la tâche de positionnement souhaitée. La seule hypothèse concernant l'objet serait de supposer l'image de son contour décrit par une fonction holomorphe. En revanche, le gain apporté par cette nouvelle approche serait considérable. En effet, elle serait, d'une part plus robuste aux erreurs de calibration du dispositif expérimental et, d'autre part, la durée d'exécution de la tâche serait diminuée puisque l'on éviterait les 3 phases séquentielles actuelles que sont le centrage, la maximisation de l puis de h.

Annexe A

Modélisation de la longueur d'un segment dans l'image à l'aide d'informations visuelles dynamiques

Tout d'abord, cette annexe établit la relation de récurrence liant la position à l'instant $k\Delta t$ d'un point de l'espace, dans le repère de la caméra, à sa position à l'instant $(k+1)\Delta t$ et ce lorsque la caméra est soumise à une commande résultant de l'application du torseur cinématique T_c. Ensuite, par transformation perspective, on obtient les coordonnées dans l'image du point considéré et par voie de conséquence la longueur d'un segment. Toutefois, précisons que ce résultat est établi dans le cas particulier où seul un mouvement appartenant au noyau de la matrice d'interaction est considéré (voir le paragraphe 4.2.4)

A.1 Obtention de la relation de récurrence

Soit $\underline{x}(t)$ les coordonnées du point observé par la caméra à l'instant t dans son repère. Elle est soumise pendant Δt à une vitesse de rotation $\underline{V}$ et à une vitesse de translation $\underline{\Omega}$ constantes telles que $T_c = \left(\underline{V}^T, \underline{\Omega}^T\right)^T$. On a alors la relation suivante :

$$\dot{\underline{x}}(t) = -\underline{V} - \underline{\Omega} \wedge \underline{x}(t) \tag{A.1}$$

où :

$$\underline{V} = (v_x, v_y, v_z)^T \tag{A.2}$$

$$\underline{\Omega} = (\omega_x, \omega_y, \omega_z)^T \tag{A.3}$$

On peut réécrire (A.1) de la façon suivante :

$$\dot{\underline{x}}(t) + \Omega \underline{x}(t) = -\underline{V} \tag{A.4}$$

avec :

$$\Omega = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix} \tag{A.5}$$

Appliquons la transformée de Laplace à (A.4), il vient :

$$\underline{x}(p) = (p\mathbb{I}_3 + \Omega)^{-1} \underline{\nu}(p) \tag{A.6}$$

où on a posé :

$$\underline{\nu}(p) = \underline{x}(t_k) - \frac{\underline{V}(t_k)}{p} \tag{A.7}$$

et $t_k = k\Delta t$.

Simplifications faites, on obtient :

$$\underline{x}(p) = \frac{1}{p\,(p^2+\omega^2)} \begin{pmatrix} p^2+\omega_x^2 & \omega_z p + \omega_x\omega_y & -\omega_y p + \omega_x\omega_z \\ -\omega_z p + \omega_x\omega_y & p^2+\omega_y^2 & \omega_x p + \omega_y\omega_z \\ \omega_y p + \omega_x\omega_z & -\omega_x p + \omega_y\omega_z & p^2+\omega_z^2 \end{pmatrix} \underline{\nu}(p) \tag{A.8}$$

avec $\omega = \|\underline{\Omega}\|$.

On obtient alors, par transformation inverse, la relation recherchée :

$$\underline{x}\left((k+1)\Delta t\right) = Q_x \underline{x}(k\Delta t) + Q_v \underline{V}(k\Delta t) \tag{A.9}$$

où on a posé :

$$Q_x = \begin{pmatrix} \frac{\omega_x^2+\ \omega_y^2+\omega_z^2\ \ C_\omega}{\omega^2} & \frac{\omega_z\omega S_\omega+\omega_x\omega_y(1-C_\omega)}{\omega^2} & \frac{-\omega_y\omega S_\omega+\omega_x\omega_z(1-C_\omega)}{\omega^2} \\ \frac{-\omega_z\omega S_\omega+\omega_x\omega_y(1-C_\omega)}{\omega^2} & \frac{\omega_y^2+(\omega_x^2+\omega_z^2)C_\omega}{\omega^2} & \frac{\omega_x\omega S_\omega+\omega_y\omega_z(1-C_\omega)}{\omega^2} \\ \frac{\omega_y\omega S_\omega+\omega_x\omega_z(1-C_\omega)}{\omega^2} & \frac{-\omega_x\omega S_\omega+\omega_y\omega_z(1-C_\omega)}{\omega^2} & \frac{\omega_z^2+\ \omega_x^2+\omega_y^2\ \ C_\omega}{\omega^2} \end{pmatrix} \tag{A.10}$$

$$Q_v = \begin{pmatrix} -\frac{\omega_x^2\omega\Delta t+\ \omega_y^2+\omega_z^2\ \ C_\omega}{\omega^3} & \frac{\omega_x\omega_y(S_\omega-\omega\Delta t)-\omega_z\omega(1-C_\omega)}{\omega^3} & \frac{\omega_x\omega_z(S_\omega-\omega\Delta t)+\omega_y\omega(1-C_\omega)}{\omega^3} \\ \frac{\omega_x\omega_y(S_\omega-\omega\Delta t)+\omega_z\omega(1-C_\omega)}{\omega^3} & -\frac{\omega_y^2\omega\Delta t+(\omega_x^2+\omega_z^2)C_\omega}{\omega^3} & \frac{\omega_y\omega_z(S_\omega-\omega\Delta t)-\omega_x\omega(1-C_\omega)}{\omega^3} \\ \frac{\omega_x\omega_z(S_\omega-\omega\Delta t)-\omega_y\omega(1-C_\omega)}{\omega^3} & \frac{\omega_y\omega_z(S_\omega-\omega\Delta t)+\omega_x\omega(1-C_\omega)}{\omega^3} & -\frac{\omega_z^2\omega\Delta t+\ \omega_x^2+\omega_y^2\ \ C_\omega}{\omega^3} \end{pmatrix} \tag{A.11}$$

dans lesquelles on a introduit les notations suivantes $C_\omega = \cos\omega\Delta t$ et $S_\omega = \sin\omega\Delta t$.

A.2 Modélisation de la taille d'un segment de l'image

Pour ce faire, conformément au paragraphe 4.2.4 et à l'équation (4.54) en particulier, on restreint la résolution de (A.1) au cas où seuls ω_x et v_y sont non nuls. On obtient alors grâce à (A.9) :

$$\begin{cases} x(t) &= x(t_k) \\ y(t) &= y(t_k)\cos(\Delta\theta_k) + z(t_k)\sin(\Delta\theta_k) - v_y(k)\dfrac{\sin(\Delta\theta_k)}{\omega_x} \\ z(t) &= -y(t_k)\sin(\Delta\theta_k) + z(t_k)\cos(\Delta\theta_k) + v_y(k)\dfrac{1-\cos(\Delta\theta_k)}{\omega_x} \end{cases} \tag{A.12}$$

où on a posé :

$$\Delta\theta_k = \omega_x(t - t_k) \tag{A.13}$$

On en déduit la position des points $m_1 = (x_1, y_1, z_1)^T$ et $m_2 = (x_2, y_2, z_2)^T$, extrémités du segment observé, et par suite, par transformation perspective la taille de l'image : $l = Y_1 - Y_2 = \frac{y_1}{z_1} - \frac{y_2}{z_2}$. Cette expression est, en fait, très complexe et on la simplifie, d'une part, en supposant que $t = t_k$ à l'ordre un et, d'autre part, en imposant, pour réaliser un mouvement appartenant au noyau de $L^T_{\substack{X_i=0\\ z_i=z}}$, la relation suivante entre v_y et ω_x :

$$v_y(k) = 2\omega_x|_k \frac{z_1|_k z_2|_k (1 + a_k^2)}{z_1|_k + z_2|_k} \tag{A.14}$$

avec $a_k = y_1|_k / z_1|_k = -y_2|_k / z_2|_k$ (si l'on suppose que la tâche principale maintient l'image correctement centrée).

Après simplification, il vient :

$$l(t) = Y_1 - Y_2 = l(t_k) - 2\omega_x|_k \frac{(1 + a_k^2)\,(z_1|_k - z_2|_k)\,(t_k - t)}{z_1|_k + z_2|_k} \tag{A.15}$$

Pour $t_k \leq t < t_k + \Delta t$, ω_x est constant d'où :

$$\frac{\partial\theta_x}{\partial t} = \omega_x|_k \Rightarrow \theta_x - \theta_x|_k = \omega_x|_k (t - t_k) \tag{A.16}$$

et finalement, sur l'intervalle $[k\Delta t, (k+1)\Delta t[$, on en déduit l'expression recherchée :

$$l(\theta_x) = l(\theta_x|_k) - 2\frac{(1 + a_k^2)\,(z_1|_k - z_2|_k)\,(\theta_x|_k - \theta_x)}{z_1|_k + z_2|_k} \tag{A.17}$$

Annexe B

Conservation de certaines propriétés propres à un objet plan par transformation perspective

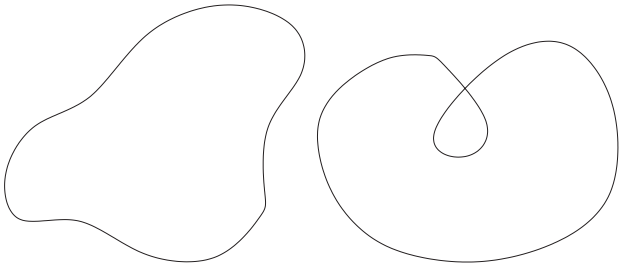
Après avoir rappelé quelques définitions de géométrie (voir [Lelong-Ferrand 77] pour un exposé plus complet), nous vérifions que certaines propriétés, propres à des points particuliers du contour de l'objet, sont conservées par transformation perspective. Deux types de points sont examinés : dans un premier temps les points stationnaires, puis les points d'inflexion. Signalons, cependant, que cette étude se limite au cas des objets plans.

B.1 Définitions

Définition 1 *Un arc γ est dit* ***simple*** *si tous les points de son support sont simples, c'est-à-dire si toute paramétrisation de γ est injective.*

La figure B.1a montre le support d'un exemple d'arc simple, la figure B.1b un contre exemple.

Définition 2 *Soit un arc γ, défini par une paramétrisation $f : I \longrightarrow E$ où I est*



(a) (b)

FIG. B.1 – (a) Arc simple. (b) Arc non simple

un intervalle de $\mathbb{R}$ et E un espace vectoriel de dimension trois tel que :

$$\begin{array}{rcl} f: & I \longrightarrow & E \\ & u \longrightarrow & f(u) = \left\{ \begin{array}{c} x(u) \\ y(u) \\ z \end{array} \right. \end{array}$$

a) Un point simple $M_0 = f(u_0)$ du support de γ est dit ***ordinaire*** *(ou* ***régulier****) si on a $f'(u_0) \neq 0$; il est dit* ***stationnaire*** *si on a $f'(u_0) = 0$.*

b) L'arc γ est dit régulier si on a $f'(u) \neq 0$ pour tout $u \in I$.

Définition 3 *Soit un arc γ, défini par une paramétrisation $f : I \longrightarrow E$ où I est un intervalle de $\mathbb{R}$ et E un espace vectoriel de dimension trois tel que :*

$$\begin{array}{rcl} f: & I \longrightarrow & E \\ & u \longrightarrow & f(u) = \left\{ \begin{array}{c} x(u) \\ y(u) \\ z \end{array} \right. \end{array}$$

et soit $M_0 = f(u_0)$ un point ordinaire du support de γ.

M_0 est dit ***point d'inflexion*** *si et seulement si le vecteur $f'(u_0)$ est colinéaire au vecteur $f''(u_0)$. En conséquence, si M_0 est un point d'inflexion du support de γ alors il admet une courbure nulle en ce point.*

B.2 Étude des points stationnaires

Considérons à présent, d'une part, un objet plan dont le contour $(\mathcal{C})$ est représenté, dans un repère fixe, par le support d'un arc γ supposé simple et, d'autre part, une caméra dont la situation par rapport au même repère fixe est représentée par une matrice homogène. Cette matrice est formée d'une sous matrice correspondant à une matrice de rotation et d'un vecteur correspondant à une translation. On peut montrer que l'inverse d'une telle matrice, que l'on notera $\mathcal{M}$, est de la forme :

$$\mathcal{M} = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{B.1}$$

Par suite, on exprime les coordonnées $\underline{x}_c(u)$ d'un point quelconque de $(\mathcal{C})$ dans le repère de la caméra par :

$$\underline{x}_c(u) = \mathcal{M}\underline{x}(u) \tag{B.2}$$

où $\underline{x}(u) = (x(u), y(u), z, 1)^T$.

On en déduit alors, par transformation perspective, l'expression bien connue des coordonnées $X(u)$ et $Y(u)$ du point observé exprimées dans le capteur :

$$\begin{cases} X(u) & = & \dfrac{m_{11}x(u) + m_{12}y(u) + m_{13}z + m_{14}}{m_{31}x(u) + m_{32}y(u) + m_{33}z + m_{34}} \\ Y(u) & = & \dfrac{m_{21}x(u) + m_{22}y(u) + m_{23}z + m_{24}}{m_{31}x(u) + m_{32}y(u) + m_{33}z + m_{34}} \end{cases} \tag{B.3}$$

Par ailleurs, soit M un point stationnaire de $(\mathcal{C})$. Par définition, M vérifie :

$$\begin{cases} \dfrac{\partial x(u)}{\partial u} = 0 \\ \dfrac{\partial y(u)}{\partial u} = 0 \end{cases} \tag{B.4}$$

Examinons à présent les conditions sur M pour que son image soit un point stationnaire. D'après (B.3), il vient :

$$\begin{cases} \dfrac{\partial X(u)}{\partial u} = A\dfrac{\partial x(u)}{\partial u} + B\dfrac{\partial y(u)}{\partial u} = 0 \\ \dfrac{\partial Y(u)}{\partial u} = C\dfrac{\partial x(u)}{\partial u} + D\dfrac{\partial y(u)}{\partial u} = 0 \end{cases} \tag{B.5}$$

où on a posé :

$$\begin{cases} A &= (m_{11}m_{33} - m_{13}m_{31})\, z \;+\; (m_{11}m_{32} - m_{31}m_{12})\, y(u) + \\ & \quad (m_{11}m_{34} - m_{14}m_{31}) \\ B &= (m_{12}m_{33} - m_{13}m_{32})\, z \;+\; (m_{31}m_{12} - m_{11}m_{32})\, x(u) + \\ & \quad (m_{12}m_{34} - m_{14}m_{32}) \\ C &= (m_{21}m_{33} - m_{23}m_{31})\, z \;+\; (m_{21}m_{32} - m_{22}m_{31})\, y(u) + \\ & \quad (m_{21}m_{34} - m_{24}m_{31}) \\ D &= (m_{22}m_{33} - m_{23}m_{32})\, z \;+\; (m_{22}m_{31} - m_{21}m_{32})\, x(u) + \\ & \quad (m_{22}m_{34} - m_{24}m_{32}) \end{cases} \tag{B.6}$$

Par conséquent, il apparaît clairement la propriété suivante : la projection dans le capteur d'un point stationnaire est un point stationnaire.

Examinons maintenant la réciproque : soit un point ordinaire $(x(u), y(u), z)^T$ tel que :

$$\begin{cases} \dfrac{\partial X(u)}{\partial u} = 0 \\ \dfrac{\partial Y(u)}{\partial u} = 0 \end{cases} \tag{B.7}$$

D'après (B.5), les coefficients A, B, C et D doivent donc être tous nuls. Ce qui conduit à résoudre le système :

$$\begin{cases} (m_{11}m_{33} - m_{13}m_{31})\, z \;+\; (m_{11}m_{32} - m_{31}m_{12})\, y(u) \;+ \\ \qquad\qquad (m_{11}m_{34} - m_{14}m_{31}) \;=\; 0 \quad \text{(B.2.1)} \\ (m_{12}m_{33} - m_{13}m_{32})\, z \;+\; (m_{31}m_{12} - m_{11}m_{32})\, x(u) \;+ \\ \qquad\qquad (m_{12}m_{34} - m_{14}m_{32}) \;=\; 0 \quad \text{(B.2.2)} \\ (m_{21}m_{33} - m_{23}m_{31})\, z \;+\; (m_{21}m_{32} - m_{22}m_{31})\, y(u) \;+ \\ \qquad\qquad (m_{21}m_{34} - m_{24}m_{31}) \;=\; 0 \quad \text{(B.2.3)} \\ (m_{22}m_{33} - m_{23}m_{32})\, z \;+\; (m_{22}m_{31} - m_{21}m_{32})\, x(u) \;+ \\ \qquad\qquad (m_{22}m_{34} - m_{24}m_{32}) \;=\; 0 \quad \text{(B.2.4)} \end{cases}$$

Pour ce faire, exprimons $x(u)$ et $y(u)$ à l'aide de (B.2.1) et (B.2.2) :

$$\begin{cases} x(u) &= -\dfrac{(m_{12}m_{34} - m_{14}m_{32}) + (m_{12}m_{33} - m_{13}m_{32})z}{m_{31}m_{12} - m_{11}m_{32}} \\ y(u) &= -\dfrac{(m_{11}m_{34} - m_{14}m_{31}) + (m_{11}m_{33} - m_{13}m_{31})z}{m_{11}m_{32} - m_{31}m_{12}} \end{cases} \tag{B.9}$$

que l'on injecte dans (B.2.3) et (B.2.4). On obtient alors deux nouvelles équations.

Du reste, on peut montrer aisément qu'elles sont liées. En n'en conservant qu'une, on exprime z en fonction des seuls $m_{i,j}$, que l'on peut injecter dans les

valeurs de $x(u)$ et de $y(u)$ données par (B.9). Par conséquent, on connait à présent les expressions de $x(u)$, $y(u)$ et z uniquement en fonction des $m_{i,j}$. On s'aperçoit alors qu'elles annulent le dénominateur de $X(u)$ et $Y(u)$ donné par (B.3). La solution du problème réciproque conduit donc au cas où l'image du point M est rejetée à l'infini.

Dans le cas où l'image de l'objet n'est pas rejetée à l'infini, c'est-à-dire en pratique dans le cas où la caméra observe l'objet, nous pouvons donc énoncer la propriété suivante :

Propriété 3 *Un point de l'image est stationnaire si, et seulement si, il est la projection d'un point stationnaire du contour de l'objet.*

B.3 Étude des points d'inflexion

Calculons la courbure $k(u)$ en un point M ordinaire de l'objet :

$$k(u) = \frac{\dfrac{\partial x}{\partial u}\dfrac{\partial^2 y}{\partial u^2} - \dfrac{\partial y}{\partial u}\dfrac{\partial^2 x}{\partial u^2}}{\left(\left(\dfrac{\partial x}{\partial u}\right)^2 + \left(\dfrac{\partial y}{\partial u}\right)^2\right)^{3/2}} \tag{B.10}$$

Les points d'inflexion sont donc obtenus en résolvant par rapport à u l'équation suivante :

$$\frac{\partial x}{\partial u}\frac{\partial^2 y}{\partial u^2} - \frac{\partial y}{\partial u}\frac{\partial^2 x}{\partial u^2} = 0 \tag{B.11}$$

Exprimons alors, dans le capteur, la courbure $\mathcal{K}(u)$:

$$\mathcal{K}(u) = \frac{\dfrac{\partial X}{\partial u}\dfrac{\partial^2 Y}{\partial u^2} - \dfrac{\partial Y}{\partial u}\dfrac{\partial^2 X}{\partial u^2}}{\left(\left(\dfrac{\partial X}{\partial u}\right)^2 + \left(\dfrac{\partial Y}{\partial u}\right)^2\right)^{3/2}} \tag{B.12}$$

Si on injecte (B.3) dans (B.12) on obtient une expression très complexe qu'il est inutile de présenter ici. Par contre, seul le numérateur $\mathcal{N}\left(D_{\mathcal{K}}(u)\right)$ nous intéresse, et on peut montrer, toutes simplifications faites [1], qu'il se met sous une forme

[1] Les calculs ont été réalisés grâce au logiciel de calcul symbolique Maple.

nettement plus sympathique :

$$\mathcal{N}\left(D_{\mathcal{K}}(u)\right) = k(u)\left(z\Delta + \sum_{i=1}^{i=3} m_{i4}\Delta_{i3}\right) \tag{B.13}$$

où Δ est le déterminant de $\mathcal{M}$ et Δ_{i3} le cofacteur du coefficient m_{i3}.

Comme dans le paragraphe précédent, $\mathcal{N}\left(D_{\mathcal{K}}(u)\right)$ peut s'annuler sans que le point M soit un point d'inflexion mais, dans ce cas, la courbure est nulle pour tout point de l'image. Ce cas de figure correspond par conséquent à un autre cas dégénéré (celui où l'image de l'objet se réduit à un segment) et nous pouvons donc énoncer une deuxième propriété :

Propriété 4 *Dans les cas non dégénérés (i.e. l'image de l'objet ne se réduit pas à un segment), un point de l'image est un point d'inflexion si, et seulement si, il est la projection d'un point d'inflexion du contour de l'objet.*

Remarque

Comme $\mathcal{M}$ est constitué d'une matrice de rotation et d'une dernière ligne nulle sauf pour un coefficient, (B.13) peut se simplifier. Cette simplification est en fait inutile puisque cette expression n'est que le numérateur de la courbure et ne peut donc pas fournir d'autres renseignements sur les extrema de courbure par exemple.

Bibliographie

[Agin 85] G. J. Agin, « Calibration and use of a light stripe range sensor mounted on the hand of a robot ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'85*, p. 680–685, St Louis, Missouri, USA, March 1985.

[Allen 91] P. K. Allen, B. Yoshimi et A. Timcenko, « Real-time visual servoing ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'91*, p. 851–856, Sacremento, California, USA, April 1991.

[Allen 93] P. K. Allen, B. Yoshimi, A. Timcenko et P. Michelman, Hand-eye coordination for robotic tracking and grasping. Dans K. Hashimoto, Ed., *Visual Servoing*, vol. 7, p. 33–70. World Scientific, Singapore, 1993.

[An 96] M.-H. An et C.-N. Lee, « Stereo vision based on algebraic curves ». Dans *Int. Conf. on Pattern Recognition, ICPR'96*, p. 476–482, 1996.

[Andersson 88] R. L. Andersson, *A robot ping-pong player : Experiment in real-time intelligent control.* The MIT Press, London, England, 1988.

[Asada 86] H. Asada et M. Brady, « The curvature primal sketch ». *IEEE Trans. on Pattern Analysis and Machine Intelligent*, vol. 8, N° 1, p. 44–54, 1986.

[Ayache 87] N. Ayache et F. Lustman, « Fast and reliable passive stereovision using three cameras ». Dans *International Workshop on Industrial Applications of Machine Vision and Machine Intelligence*, Tokyo, Japan, February 1987.

[Ayache 89] N. Ayache, *Vision stéréoscopique et perception multisensorielle.* Science informatique. InterÉditions, Paris, 1989.

[Bascle 94] B. Bascle, « *Contributions et applications des modèles déformables en vision par ordinateur* ». Thèse de doctorat, Université de Nice — Sophia Antipolis, Nice, Juillet 1994.

[Bensalah 96] F. Bensalah, « *Estimation du mouvement par vision active* ». Thèse de doctorat, Université de Rennes I, Rennes, Juillet 1996.

[Berry 97] F. Berry, P. Martinet et J. Gallice, « Trajectory generation by visual servoing ». Dans *IEEE Int. Conf. on Intelligent Robots and Systems, IROS'97*, vol. 2, p. 1065–1071, 7-11 September 1997.

[Besançon 88] J. E. Besançon, « *Vision par ordinateur en deux et trois dimensions* », Chapitre B9, Système d'identification de pièces industrielles et de reconnaissance de l'écriture. Eyrolles, Paris, 1988.

[Besl 88] P. J. Besl, « Geometric modeling and computer vision ». *Proceedings of the IEEE*, vol. 76, N° 8, p. 936–958, August 1988.

[Bien 93] Z. Bien, W. Jang et J. Park, Characterisation and use of feature-Jacobian matrix for visual servoing. Dans K. Hashimoto, Ed., *Visual Servoing*, vol. 7, p. 317–363. World Scientific, Singapore, 1993.

[Billingsley 97] J. Billingsley et M. Schönfisch, « The successful development of a vision guidance system for agriculture ». *Computers and Electronics in Agriculture*, vol. 16, p. 147–163, 1997.

[Brand 94] P. Brand et R. Mohr, « Accuracy in image measure ». Dans *SPIE Conference on videometrics III*, vol. 2350, p. 218–228, 1994.

[Byun 96] J. E. Byun et T. Nagata, « Determining the 3-D pose of a flexible object by stereo matching of curvature representations ». *Pattern Recognition*, vol. 29, N° 8, p. 1297–1307, 1996.

[Böhm 84] W. Böhm, G. Farin et J. Kahmann, « A survey of curve and surface methods in CAGD ». *Computer Aided Geometric Design*, vol. 1, N° 1, p. 1–60, July 1984.

[Castano 94] A. Castano et S. Hutchinson, « Visual compliance : task-directed visual servo control ». *IEEE Trans. on Robotics and Automation*, vol. 10, N° 3, p. 334–341, June 1994.

[Chanet 98] J. P. Chanet et T. Humbert, « Automated reading of pig tattooed identifiers by artificial vision ». Dans *First International Workshop on Future Automation for Meat Processing*, p. 331–334, Frankfurt, Germany, 9-10 June 1998.

[Chaumette 90] F. Chaumette, « *La relation vision-commande : théorie et application à des tâches robotiques* ». Thèse de doctorat, Université de Rennes I, Rennes, Juillet 1990.

[Chen 87] C. H. Chen et A. C. Kak, « Modeling and calibration of a structured light scanner for 3-D robot vision ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'87*, p. 807–815, Raleigh, North Carolina, USA, March-April 1987.

[Chuang 97] D.-M. Chuang, S.-C. Wu et M.-K. Hor, « Adaptive Fuzzy Visual servoing in robot control ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'97*, p. 811–816, Albuquerque, New Mexico, USA, April 1997.

[Coiffet 86] P. Coiffet, *La robotique : principes et applications.* Hermes, Paris, 1986. Traité des nouvelles technologies.

[Collewet 99] G. Collewet, C. Collewet et P. Marchal, « Système de localisation tridimensionnelle pour le guidage d'un robot de traite ». *Accepté dans : Journal Européen des Systèmes Automatisés*, Mars 1999.

[Coste-Manière 95] E. Coste-Manière, P. Couvignou et P. Khosla, « Visual servoing in the task-function framework : a contour following task ». *Journal of Intelligent and Robotic Systems*, vol. 12, p. 1–21, 1995.

[Crowley 95] J. L. Crowley, M. Mesrabi et F. Chaumette, « Comparaison of kinematic and visual servoing for fixation ». Dans *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'95*, vol. 1, p. 335–341, Pittsburgh, USA, Août 1995.

[Crétual 97] A. Crétual et F. Chaumette, « Positioning a camera parallel to a plane using dynamic visual servoing ». Dans *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'97*, p. 43–48, Grenoble, Septembre 1997.

[Crétual 98] A. Crétual et F. Chaumette, « Image-based visual servoing by integration of dynamic measurements ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'98*, Louvain, Belgique, Mai 1998.

[Daucher 97] N. Daucher, M. Dhome, J. T. Lapresté et G. Rives, « Speed Command of a robotic system by monocular pose estimate ». Dans *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'97*, p. 55–62, Grenoble, Septembre 1997.

[Debain 96] C. Debain, « *Lois de commande pour le contrôle et la mobilité des machines agricoles* ». Thèse de doctorat, Université Blaise Pascal — Clermont II, Clermont-Ferrand, Septembre 1996.

[Deguchi 96] K. Deguchi et T. Noguchi, « Visual servoing using eigenspace method and dynamic calculation of interaction matrices ». Dans *Int. Conf. on Pattern Recognition, ICPR'96*, p. 302–306, 1996.

[Dementhon 95] D. F. Dementhon et L. S. Davis, « Model-based object pose in 25 lines of code ». *International Journal of Computer Vision*, vol. 15, p. 123–141, 1995.

[Deriche 93] R. Deriche et T. Blaszka, « Recovering and characterizing image features using an efficient model based approach ». Dans *Conference on Computer Vision and Pattern Recognition*, p. 530–535, 1993.

[Dias 98] J. Dias, C. Paredes, I. Fonseca, H. Araújo, J. Batista et A. T. Almeida, « Simulating pursuit with machine experiments with robots and artificial vision ». *IEEE Trans. on Robotics and Automation*, vol. 14, N° 1, p. 1–18, February 1998.

[Domingo 96] J. Domingo et J. Pelechano, « Measurement and storage of a network of jacobians as a method for the visual positioning of a robot arm ». *Journal of Intelligent and Robotic Systems*, vol. 16, p. 407–422, 1996.

[Dreschler 82] L. Dreschler et H.-H. Nagel, « Volumetric model and 3D trajectory of a moving car derived from monocular tv frame sequences of a street scene ». *Computer Vision, graphics, and image processing*, vol. 20, p. 199–228, 1982.

[Espiau 93] B. Espiau, « Effect of camera calibration errors on visual servoing in robotics ». Dans *Third Int. Symp. on Experimental Robotics, ISER'93*, Kyoto, Japan, October 1993.

[Faugeras 88] O. D. Faugeras, « Quelques pas vers la vision artificielle en trois dimensions ». *Technique et Science Informatiques*, vol. 7, N° 6, p. 547–590, 1988.

[Feddema 89a] J. T. Feddema, C. S. G. Lee et O. R. Mitchell, « Automatic selection of image features for visual servoing of a robot manipulator ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'89*, p. 832–837, Washington, May 1989. IEEE Computer Society Press. Scottsdale, Arizona, USA.

[Feddema 89b] J. T. Feddema et O. R. Mitchell, « Vision-guided servoing with feature-based trajectory generation ». *IEEE Trans. on Robotics and Automation*, vol. 5, N° 5, p. 691–700, October 1989.

[Fischler 81] M. A. Fischler et R. C. Bolles, « Random Sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography ». *Communications of the ACM*, vol. 24, p. 381–395, 1981.

[Förstner 94] W. Förstner, « A framework for low level feature extraction ». Dans *3rd European Conference on Computer Vision*, 1994.

[Gardiner 65] M. Gardiner, « The superellipse : A curve that lies between the ellipse and the rectangle ». *Sci. Amer.*, vol. 213, p. 222–234, 1965.

[Giles 97] D. K. Giles et D. C. Slaughter, « Precision band spraying with machine-vision guidance and adjustable yaw nozzles ». *Transactions of the* asae, vol. 40, N° 1, p. 29–36, 1997.

[Grosso 96] E. Grosso, G. Metta, A. Oddera et G. Sandini, « Robust visual servoing in 3-D reaching tasks ». *IEEE Trans. on Robotics and Automation*, vol. 12, N° 5, p. 732–742, October 1996.

[Hager 95] G. D. Hager, W. C. Chang et A. S. Morse, « Robot hand-eye coordination based on stereo vision ». *IEEE Control Systems Magazine*, vol. 15, N° 1, p. 30–39, February 1995.

[Hager 97] G. D. Hager, « A modular system for robust positioning using feedback from stereo vision ». *IEEE Trans. on Robotics and Automation*, vol. 13, N° 4, p. 582–595, August 1997.

[Han 93] S. Han, D. B. Goldgof et K. W. Bowyer, « Using hyperquadrics for shape recovery from range data3D ». Dans *IEEE Int. Conf. on Computer Vision, ICCV'93*, p. 492–496, Berlin, Germany, June 1993.

[Hanson 88] A. J. Hanson, « Hyperquadrics : Smoothly deformable shapes with convex polyhedral bounds ». *Computer Vision, graphics, and image processing*, vol. 44, p. 191–210, 1988.

[Harris 88] C. Harris et M. Stephens, « A combined corner and edge detector ». Dans *Fourth Alvey Vision Conference*, p. 147–151, Manchester, United Kingdom, August 1988.

[Hashimoto 91] H. Hashimoto, T. T. Kimoto, T. Ebine et H. Kimura, « Manipulator control with image-based visual servo ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'91*, p. 2267–2272, Sacremento, California, USA, April 1991.

[Hashimoto 96] K. Hashimoto, T. Ebine et H. Kimura, « Visual servoing with hand-eye manipulator — Optimal control approach ». *IEEE Trans. on Robotics and Automation*, vol. 12, N° 5, p. 766–774, October 1996.

[Hashimoto 97] K. Hashimoto et T. Noritsugu, « Visual servoing of nonholonomic cart ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'97*, p. 1719–1724, Albuquerque, New Mexico, USA, April 1997.

[Hill 79] J. Hill et W. T. Park, « Real time control of a robot with a mobile camera ». Dans *9th Int. Symp. on Industrial Robots*, p. 233–246, March 1979.

[Horn 74] B. K. P. Horn, « Determining lightness from an image ». *Computer Vision, graphics, and image processing*, vol. 3, N° 1, p. 277–299, 1974.

[Horn 77] B. K. P. Horn, « Image intensity understanding ». *Artificial Intelligence*, vol. 8, N° 2, p. 201–231, 1977.

[Hosoda 94] K. Hosoda et M. Asada, « Versatile visual servoing without knowledge of true jacobian ». Dans *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'94*, p. 186–193, Munich, Allemagne, September 1994.

[Iglésias 92] G. Iglésias, « *Modélisation surfacique et dessins de courbes sur surfaces* ». Thèse de doctorat, Université de Paris XI, Orsay, Septembre 1992.

[Janabi-Sharifi 97] F. Janabi-Sharifi et W. J. Wilson, « Automatic selection of image features for visual servoing ». *IEEE Trans. on Robotics and Automation*, vol. 13, N° 6, p. 890–903, December 1997.

[Jang 91] W. Jang et Z. Bien, « Feature-based visual servoing of an eye-in-hand robot with improved tracking performance ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'91*, p. 2254–2260, Sacramento, California, USA, April 1991.

[Jägersand 97] M. Jägersand, O. Fuentes et R. Nelson, « Experimental evaluation of uncalibrated visual servoing for precision manipulation ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'97*, p. 2874–2880, Albuquerque, New Mexico, USA, April 1997.

[Kelly 96] R. Kelly, « Robust asymptotically stable visual servoing of planar robots ». *IEEE Trans. on Robotics and Automation*, vol. 12, N° 5, p. 759–766, October 1996.

[Khadraoui 96] D. Khadraoui, G. Motyl, P. Martinet, J. Gallice et F. Chaumette, « Visual servoing in robotics scheme using a camera/Laser-Stripe Sensor ». *IEEE Trans. on Robotics and Automation*, vol. 12, N° 5, p. 743–750, October 1996.

[Kim 88] N. H. Kim et A. C. Bovik, « A contour-based stereo matching algorithm using disparity continuity ». *Pattern Recognition*, vol. 21, N° 5, p. 505–514, 1988.

[Kinase 91] A. Kinase et H. Watake, « Robot for masspropagation ». Dans Y. Hashimoto et W. Day, Eds., *Mathematical and control applications in agriculture and horticulture*, p. 225–230, Matsuyama, Japan, 30 September - 3 October 1991. Pergamon Press.

[Kitchen 82] L. Kitchen et R. Rosenfeld, « Gray-level corner detection ». *Pattern Recognition letters*, vol. 1, p. 95–102, 1982.

[Koivo 91] A. J. Koivo et N. Houshangi, « Real-time vision feedback for servoing robotic manipulator with self-tuning controller ». *IEEE Trans. on Systems, Man, and Cybernetics*, vol. 21, N° 1, p. 134–142, February 1991.

[Kondo 91] N. Kondo, « Study on grape harvesting robot ». Dans Y. Hashimoto et W. Day, Eds., *Mathematical and control applications in agriculture and horticulture*, p. 243–251, Matsuyama, Japan, 30 September - 3 October 1991. Pergamon Press.

[Kondo 96] N. Kondo, Y. Nishitsuji, P. P. Ling et K. C. Ting, « Visual feedback guided robotic cherry tomato harvesting ». *Transactions of the* asae, vol. 39, N° 6, p. 2331–2338, 1996.

[Kuhl 82] F. P. Kuhl, « Elliptic Fourier features of a closed contour ». *Computer Vision, graphics, and image processing*, vol. 18, p. 236–258, 1982.

[Labarrere 88] M. Labarrere, J. P. Krief et B. Gimonet, *Le filtrage et ses applications*. Cepadues-Édition, Toulouse, 1988. 3eÉdition.

[Legeard 99] D. Legeard, « *Extraction de caractéristiques pour l'évaluation automatique de la qualité de jambons frais par vision numérique* ». Thèse de doctorat, Université de Rennes I, Rennes, Avril 1999.

[Lelong-Ferrand 77] J. Lelong-Ferrand et J. M. Arnaudies, *Cours de mathématiques, Tome 3, Géométrie et cinématique*. Dunod, Paris, 1977. 2eÉdition.

[Li 96] L. Li et S. D. Ma, « 3D pose estimation from an n-degree planar curved feature in two perspective views ». Dans *Int. Conf. on Pattern Recognition, ICPR'96*, p. 374–377, 1996.

[Liao 94] K. Liao, M. R. Paulsen et J. F. Reid, « Real-time detection of colour and surface defects of maize kernels using machine vision ». *Journal of Agricultural Engineering Research*, vol. 59, p. 263–271, 1994.

[Lim 97] F. L. Lim, G. A. W. West et S. Venkatesh, « Use of log polar space for foveation and feature recognition ». *IEE Proceedings - Vision Image and Signal Processing*, vol. 144, N° 6, p. 323–331, December 1997.

[Lowe 91] D. G. Lowe, « Fitting parameterized three-dimensional models to images ». *IEEE Trans. on Pattern Analysis and Machine Intelligent*, vol. 60, N° 5, p. 441–450, May 1991.

[Ma 93] S. D. Ma, « Conics-based stereo, motion estimation, and pose determination ». *International Journal of Computer Vision*, vol. 10, p. 7–25, 1993.

[Malis 97] E. Malis, F. Chaumette et S. Boudet, « Positioning a coarse calibrated camera with respect to an unknown object by 2D 1/2 visual servoing ». Dans *5th IFAC Symposium on Robot Control, SYROCO'97*, vol. 2, p. 517–523, Nantes, September 1997.

[Malis 98] E. Malis, F. Chaumette et S. Boudet, « Positioning a coarse calibrated camera with respect to an unknown object by 2D 1/2 visual servoing ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'98*, Louvain, Belgique, May 1998.

[Marchand 96] E. Marchand, F. Chaumette et A. Rizzo, « Using the task function approach to avoid robot joint limits and kinematic singularities in visual servoing ». Dans *IEEE Int. Conf. on Intelligent Robots and Systems, IROS'96*, vol. 3, p. 1083–1090, Osaka, Japan, November 1996.

[Marefat 97] M. M. Marefat, L. Wu et C. C. Yang, « Gaze stabilization in active vision — I — Vergence error extraction ». *Pattern Recognition*, vol. 30, N° 11, p. 1829–1842, 1997.

[Martinet 96] P. Martinet, F. Berry et J. Gallice, « Use of first derivative of geometric features in visual servoing ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'96*, p. 3413–3419, Minneapolis, Minnesota, USA, April 1996.

[Martinet 97] P. Martinet, C. Debain, M. Berducat, P. Bonton et J. Gallice, « Asservissement visuel pour le guidage d'engins agricoles ». Dans *Journées Automatique Agriculture et Agro-alimentaire, J3A'97*, p. 253–263, Clermont-Ferrand, Octobre 1997.

[Marty 93] C. Marty, C. Cassagnes et P. Marin, « *La pratique de la commande numérique des machines-outils* », Chapitre 11, Principes de construction et utilisation des courbes et surfaces à pôles. Lavoisier, Paris, 1993.

[Marty-Mahé 97a] P. Marty-Mahé et P. Marchal, « Évaluation de la qualité de carcasses de dinde par vision numérique : détection des défauts d'aspect ». *Comptes rendus de l'académie d'agriculture de France*, vol. 83, N° 2, p. 128–136, 1997.

[Marty-Mahé 97b] P. Marty-Mahé et P. Marchal, « Évaluation de la qualité de carcasses de dinde par vision numérique : évaluation des caractéristiques morphologiques ». *Comptes rendus de l'académie d'agriculture de France*, vol. 83, N° 2, p. 121–127, 1997.

[McFarlane 97] N. J. B. McFarlane, B. Tisseyre, C. Sinfort, R. D. Tillett et F. Sévila, « Image analysis for pruning of long wood grape vines ». *Journal of Agricultural Engineering Research*, vol. 66, N° 2, p. 111–119, February 1997.

[Meyers 90] J. B. Meyers, S. E. Prussia, C. N. Thai, T. L. Sadosky et D. T. Campbell, « Visual inspection of agricultural products moving along sorting conveyors ». *Transactions of the* asae, vol. 33, N° 2, p. 367–372, 1990.

[Mitchell 84] O. R. Mitchell et T. A. Grogan, « Global and partial shape discrimination for computer vision ». *Optical Engineering*, vol. 23, N° 5, p. 484–491, September-October 1984.

[Mitiche 96] A. Mitiche et P. Bouthemy, « Computation and analysis of image motion : a synopsis of current problems and methods ». *International Journal of Computer Vision*, vol. 16, N° 1, p. 29–55, 1996.

[Mitsuda 97] T. Mitsuda, N. Maru, K. Fujikawa et F. Miyazaki, « Binocular visual servoing based on linear time-invariant mapping ». *Advanced Robotics*, vol. 11, N° 5, p. 429–443, 1997.

[Mokhtarian 86] F. Mokhtarian et A. Mackworth, « Scale-based description and recognition of planar curves and two-dimensional shapes ». *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, N° 1, p. 34–43, January 1986.

[Monga 90] O. Monga, « Segmentation d'images : où en sommes nous ? ». Rapport de recherche N° 1216, INRIA, Avril 1990.

[Moravec 80] H. P. Moravec, « *Obstacle avoidance and navigation in the real world by a seeing robot rover* ». Ph. D. Thesis, Stanford artificial intelligence laboratory, 1980. Technical report memo 340.

[Nayar 96] S. K. Nayar, S. A. Nene et H. Murase, « Subspace methods for robot vision ». *IEEE Trans. on Robotics and Automation*, vol. 12, N° 5, p. 750–758, October 1996.

[Nordlund 96] P. Nordlund et T. Uhlin, « Closing the loop : detection and pursuit of a moving object by a moving observer ». *Image and Vision Computing*, vol. 14, p. 265–275, 1996.

[Ozanian 95] T. Ozanian, « Approaches for stereo matching — A review ». *Modelling, Identification and Control*, vol. 16, N° 2, p. 65–94, 1995.

[Papanikolopoulos 91] N. Papanikolopoulos, P. K. Khosla et T. Kanade, « Vision and control techniques for robotic visual tracking ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'91*, p. 857–864, Sacramento, California, USA, April 1991.

[Papanikolopoulos 95] N. P. Papanikolopoulos, « Selection of features and evaluation of visual measurements during robotic visual servoing tasks ». *Journal of Intelligent and Robotic Systems*, vol. 13, p. 279–304, 1995.

[Persoon 86] E. Persoon et K.-S. Fu, « Shape discrimination using Fourier descriptors ». *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, N° 3, p. 388–397, May 1986.

[Pla 97] F. Pla, J. M. Sanchiz, J. A. Marchant et R. Brivot, « Building perspective models to guide a row crop navigation vehicle ». *Image and Vision Computing*, vol. 15, p. 465–473, 1997.

[Rabatel 97] G. Rabatel, J. M. Roger et F. Pierrot, « Utilisation combinée de capteurs inertiels et vision pour le contrôle d'un bras hexa de désherbage automatique ». Dans *Journées Automatique Agriculture et Agroalimentaire, J3A'97*, p. 216–223, Clermont-Ferrand, Octobre 1997.

[Raffy 99] G. Raffy, « *Vision numérique couleur appliquée à l'évaluation de la qualité de carcasses de dinde* ». Thèse de doctorat, Université Blaise Parcal - Clermont II, Clermont-Ferrand, Mars 1999.

[Rhor 92] K. Rhor, « Recognizing corners by fitting parametric ». *International Journal of Computer Vision*, vol. 9, N° 3, p. 213–230, 1992.

[Rives 97] P. Rives et J.-J. Borrelly, « Visual servoing techniques applied to an underwater vehicle ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'97*, p. 1851–1856, Albuquerque, New Mexico, USA, April 1997.

[Rizzi 93] A. A. Rizzi et D. E. Koditschek, A dynamic sensor for robot juggling. Dans K. Hashimoto, Ed., *Visual Servoing*, vol. 7, p. 229–256. World Scientific, Singapore, 1993.

[Rosen 76] C. Rosen, « Machine intelligence research applied to industrial automation ». Technical report, SRI International, 1976. Sixth report.

[Russ 95] J. C. Russ, *The image processing handbook.* CRC Press, London, 1995. 2nd Edition.

[Samson 88] C. Samson, B. Espiau et M. Le Borgne, « Robot redundancy : an automatic control approach ». Dans *NATO Advanced Research Workshop on Robots with Redundancy*, June 1988. Salo, Italia.

[Samson 91] C. Samson, B. Espiau et M. Le Borgne, *Robot control : The task function approach.* Clarendon Press, Oxford, England, 1991.

[Sanchiz 96] J. M. Sanchiz, F. Pla, J. A. Marchant et R. Brivot, « Structure from motion techniques applied to crop field mapping ». *Image and Vision Computing*, vol. 14, p. 353–363, 1996.

[Sanderson 80] A. C. Sanderson et L. E. Weiss, « Image-based visual servo control using relational graph error signals ». Dans *Int. Conf. on Cybernetics and Society*, p. 1074–1077, Cambridge, MA, USA, October 1980.

[Santos 95] A. Santos, B. Espiau, P. Rives, D. Simon et V. Rigaud, « Sensor-based control of holonomic autonomous underwater vehicles ». Rapport de recherche N° 2609, INRIA, Juillet 1995.

[Santos-Victor 97] J. Santos-Victor et G. Sandini, « Visual behaviors for docking ». *Computer Vision and Image Understanding*, vol. 67, N° 3, p. 223–238, September 1997.

[Sarig 93] Y. Sarig, « Robotics of Fruit harvesting : A state-of-the-art Review ». *Journal of Agricultural Engineering Research*, vol. 54, N° 4, p. 265–280, April 1993.

[Schmid 96] C. Schmid, « *Appariement d'images par invariants locaux de niveaux de gris : application à l'indexation d'une base d'objets* ». Thèse de doctorat, Institut national polytechnique de Grenoble, Grenoble, Juillet 1996.

[Schmid 98] C. Schmid, R. Mohr et C. Bauckhage, « Comparing and evaluating interest points ». Dans *6th International Conference on Computer Vision, ICCV'98*, Bombay, India, January 1998.

[Shirai 73] Y. Shirai et H. Inoue, « Guiding a robot by visual feedback in assembling tasks ». *Pattern Recognition*, vol. 5, p. 99–108, 1973.

[Simonton 91] W. Simonton, « Automated plant handling and processing in a robotic workcell ». Dans Y. Hashimoto et W. Day, Eds., *Mathematical and control applications in agriculture and horticulture*, p. 213–218, Matsuyama, Japan, 30 September - 3 October 1991. Pergamon Press.

[Smith 97a] C. E. Smith, S. A. Brandt et N. P. Papanikolopoulos, « Eye-in-hand robotic tasks in uncalibrated environments ». *IEEE Trans. on Robotics and Automation*, vol. 13, N° 6, p. 903–914, December 1997.

[Smith 97b] C. E. Smith et N. P. Papanikolopoulos, « Grasping of static and moving objects using a vision-based control approach ». *Journal of Intelligent and Robotic Systems*, vol. 19, p. 237–270, 1997.

[Spindler 97] F. Spindler et F. Chaumette, « Gaze control using human eye movements ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'97*, vol. 3, p. 2258–2263, Albuquerque, New Mexico, USA, April 1997.

[Srinivasa 97] N. Srinivasa et R. Sharma, « SOIM : a self-organizing invertible map with applications in active vision ». *IEEE Trans. on Neural Networks*, vol. 8, N° 3, p. 758–773, May 1997.

[Staib 92] L. H. Staib et J. S. Duncan, « Boundary Finding with parametrically deformable models ». *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 14, N° 11, p. 1061–1075, November 1992.

[Suh 93] I. H. Suh et T. W. Kim, Visual servoing of robot manipulators by fuzzy membership function based neural networks. Dans K. Hashimoto, Ed., *Visual Servoing*, vol. 7, p. 285–315. World Scientific, Singapore, 1993.

[Tani 77] K. Tani, M. Abe, K. Tanie et T. Ohno, « High precision manipulators with visual sense ». Dans *7th Int. Symp. on industrial robotics*, p. 561–568, Tokyo, Japan, October 1977.

[Tao 95] Y. Tao, C. T. Morrow, P. H. Heinemann et H. J. Sommer, « Fourier-based separation technique for shape grading of patatoes using machine vision ». *Transactions of the ASAE*, vol. 38, p. 949–957, 1995.

[Tillett 93] N. D. Tillett, « Robotic manipulators in horticulture : A review ». *Journal of Agricultural Engineering Research*, vol. 55, N° 2, p. 89–105, June 1993.

[Tillett 95] N. D. Tillett, W. He et R. D. Tillett, « Development of a vision guided robot manipulator for packing horticultural produce ». *Journal of Agricultural Engineering Research*, vol. 61, p. 145–154, 1995.

[Tillett 98] N. D. Tillett, T. Hague et J. A. Marchant, « A robotic system for plant-scale husbandry ». *Journal of Agricultural Engineering Research*, vol. 69, N° 2, p. 169–178, February 1998.

[Tistarelli 93] M. Tistarelli et G. Sandini, « On the advantages of polar and log-polar mapping for direct estimation of time-to-impact from optical flow ». *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 15, N° 4, p. 401–410, April 1993.

[Weiss 87] L. E. Weiss, A. C. Sanderson et C. P. Neuman, « Dynamic sensor-based control of robots with visual feedback ». *IEEE Journal of Robotics and Automation*, vol. RA-3, N° 5, p. 404–417, October 1987.

[Wells 96] G. Wells, C. Venaille et C. Torras, « Promising research vision-based robot positioning using neural networks ». *Image and Vision Computing*, vol. 14, p. 715–732, 1996.

[Wilson 96] W. J. Wilson, C. C. W. Hulls et G. S. Bell, « Relative end-effector control using cartesian position based visual servoing ». *IEEE Trans. on Robotics and Automation*, vol. 12, N° 5, p. 684–696, Octobre 1996.

[Wu 97] Q. M. J. Wu et K. Stanley, « Modular Neural-Visual servoing using a neural-fuzzy decision network ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'97*, p. 3238–3243, Albuquerque, New Mexico, USA, April 1997.

[Wunsch 97a] P. Wunsch et G. Hirzinger, « Real-time visual tracking of 3-D objects with dynamic handling of occlusion ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'97*, p. 2868–2873, Albuquerque, New Mexico, USA, April 1997.

[Wunsch 97b] P. Wunsch, S. Winkler et G. Hirzinger, « Real-time pose estimation of 3-D objects from camera images using neural networks ». Dans *IEEE Int. Conf. on Robotics and Automation, ICRA'97*, p. 3232–3237, Albuquerque, New Mexico, USA, April 1997.

[Xie 89] M. Xie, « *Contribution à la vision dynamique : reconstruction d'objets 3D polyédriques par une caméra mobile* ». Thèse de doctorat, Université de Rennes I, Rennes, Juin 1989.

[Yuan 89] J. S.-C. Yuan, « A general photogrammetric method for determining object position and orientation ». *IEEE Trans. on Robotics and Automation*, vol. 5, N° 2, p. 129–142, April 1989.

Contributions à l'élargissement du champ applicatif des asservissements visuels 2D

Ce document traite de la réalisation de tâches de positionnement par asservissement visuel 2D. On considère tout d'abord, le cas où l'image désirée est complexe, puis le cas où elle est inconnue. Dans un premier temps, nous nous intéressons au calcul de la matrice d'interaction relative à des informations visuelles issues du contour d'un objet observé dans l'image et montrons comment il est possible d'en obtenir une approximation. Pour ce faire, nous reformulons l'approche habituelle en un problème de minimisation d'une erreur de modélisation du contour. Nous illustrons cette nouvelle façon de procéder sur deux exemples en réalisant différentes tâches de positionnement. Le premier est basé sur le calcul de la signature polaire du contour de l'image de l'objet, le second sur une description paramétrique de ce contour. Des résultats de simulation et d'expérimentation valident les développements théoriques. En particulier, la robustesse de la loi de commande est testée vis-à-vis d'une mauvaise calibration du dispositif expérimental, des paramètres intervenants dans la commande, ou encore d'une occlusion partielle de l'objet. D'un tout autre point de vue, nous nous intéressons ensuite au cas de figure où la consigne visuelle est inconnue — c'est-à-dire le cas où l'image finale souhaitée n'est pas connue précisément — et ce dans le contexte d'une tâche particulière de positionnement consistant à amener le capteur en face de l'objet considéré. Par contre, nous supposons cet objet plan et immobile. Pour simplifier le problème, nous traitons tout d'abord le cas d'un objet filiforme et montrons comment cette approche se généralise à un objet plan de forme quelconque dès que trois points caractéristiques existent. Une approche, ne faisant cette fois aucune hypothèse, est proposée à la fin du document. Des résultats de simulations et d'expérimentations valident les lois de commande proposées.

Mots-clés : Asservissement visuel 2D, informations visuelles complexes, signature polaire, fonctions paramétriques, descripteurs de Fourier, consigne visuelle inconnue, tâche de positionnement, approche critère, points caractéristiques.

Contributions to the expansion of the application domain of 2D visual servoing

This document deals with the way to achieve positioning tasks by 2D visual servoing. We consider the case when the desired image is complex and then when it is unknown. First, we are interested in the computation of the interaction matrix with regards to the visual features extracted from the image contour of the observed object and we show how an approximation can be obtained. In order to do this, we modify the usual approach to get a problem of minimization of a contour modelling error. We illustrate this new way to proceed with two examples concerning different positioning tasks. The first one is based on the polar description of the contour while the second uses a parametric description. Experimental results validate the proposed algorithm. Particularly, the robustness of the control law is tested with regards to a coarse calibrated system, to parameters which have an effect on the control law, and to partial occlusion of the object. In a second part, we are interested in the case where the desired visual features are not exactly known. We consider a particular positioning task consisting in moving the camera in front of the object. This object is assumed to be planar and motionless. To simplify this problem, we treat the case of a threadlike object and then we show how our approach can be generalized to an object with three particular points. Another approach, without any hypothetis is proposed at the end of this document. Experimental results validate the proposed control law.

Keywords : 2D visual servoing, complex visual features, polar signature, parametric functions, Fourier descriptors, unknown desired visual features, positioning task, criteria approach, characteristic points.