

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mouloud MAMMERY, Tizi-Ouzou



Faculté de Génie Electrique et d'Informatique
Département d'Automatique

Mémoire de Fin d'Etudes

En vue de l'obtention du diplôme

De Master en Automatique

Thème

Commande Référencée
Vision

Proposé et dirigé par :

M^r DIAF Moussa

Présenté par :

YAHY Taous
NOUFEL Taous

Soutenu le : 14 / 07 /2011

Promotion 2011

Ce travail a été préparé au laboratoire Robotique et Vision de l'U-M-M-T-O

Remerciements



Au terme de ce travail, nous tenons à exprimer notre profonde gratitude à Monsieur Moussa DIAF, Professeur à l'U-M-M-T-O, de nous avoir proposé ce thème et guidé tout le long de ce travail.

Nous tenons à remercier également Ahcène TRIKI doctorant à l'U-M-M-T-O pour son aide.

Nous remercions aussi bien vivement les membres de jury qui nous feront l'honneur d'évaluer ce travail, ainsi que tous les enseignants qui ont contribué à notre formation.



D *Dédicace*



A mes très chers parents;

*Aucun acte ou expression ne pourra exprimer mes
sentiments envers vous.*

A mes frères et sœurs

*Aucun mot ne pourra exprimer ma gratitude envers vous
Pour tout l'amour et le soutien que vous m'avez offert, je
vous dis MERCI.*

A ma nièce IMANE

A toute ma famille;

*A mon binôme YAHY Taous que je remercie pour tous ses
efforts.*

A mes très chères amies;

*Pour tous les instants inoubliables que j'ai passés avec vous,
je vous remercie.*

A tous ceux qui m'aiment.

Taous .N



Dédicace

A mes très chers parents ;

*Aucun acte ou expression ne pourra exprimer mes
sentiments envers vous.*

A mes frères et sœurs bien aimés (es)

A mon neveu YANIS

A toute ma famille;

*A mon binôme NOUFEL Taous que je remercie pour tous ses
efforts.*

A tous mes chers amis;

*Pour tous les instants inoubliables que j'ai passés avec vous,
je vous remercie.*

A tous ceux qui m'aiment.

Taous.Y

Sommaire

Introduction générale	1
 Premier chapitre : Généralités sur l’asservissement visuel	
1.1. Introduction.....	3
1.2. Historique.....	4
1.3. Asservissement visuel.....	7
1.3.1. Définition.....	8
1.3.2. Principe.....	8
1.4. Types de tâche des asservissements visuels.....	9
1.5. L’approche Look and Move.....	11
1.5.1. L’approche static Look and Move.....	12
1.5.2. L’approche dynamique Look and Move.....	13
1.6. Classification des asservissements visuels.....	14
1.6.1. Selon la position du capteur visuel.....	14
1.6.2. Selon le type de la commande.....	16
1.6.2.1. Asservissement visuel séquentie.....	16
1.6.2.2. Asservissement visuel cinématique.....	16
1.6.2.3. Asservissement visuel dynamique.....	17
1.6.3. Selon le type de mesure.....	18
1.6.3.1. Asservissement visuel 3D.....	18
1.6.3.2. Asservissement visuel 2D.....	20

Sommaire

1.6.3.3. Asservissement visuel 2D1/2.....	21
1.6.2.4. Asservissement visuel d2D/dt.....	22
1.7. Le choix des informations visuelles.....	24
1.8. Approche fonction de tâche en asservissement visuel.....	24
1.9. Obtention de la consigne s^*	25
1.10. Obtention des images.....	26
1.10.1. Changement de repère.....	28
1.10.2. Projection perspective.....	28
1.10.3. Passage aux coordonnées pixelliques.....	29
1.11. Calibrage capteur/robot (camera /pince).....	31
1.11.1. Formulation de problème de calibrage.....	32
1.11.2. Solution au problème de calibrage.....	32
1.12. Notion de base de la modélisation d'une caméra.....	34
1.12.1. Définition de l'attitude d'un repère.....	34
1.12.2. Définition des repères.....	34
1.12.3. Axe et angle de rotation équivalent à une rotation donnée.....	36
1.13. Conclusion.....	37

Deuxième chapitre : Les techniques d'asservissement visuel

2.1. Introduction.....	38
2.2. Asservissement visuel 2D.....	39
2.2.1. Notion de jacobien d'image.....	40
2.2.2. Modélisation du jacobien d'image.....	42
2.2.3. Choix de la matrice C.....	48

Sommaire

2.2.4. Les lois de commandes.....	49
2.3. Asservissement visuel 3D.....	50
2.3.1. L'estimation de la pose.....	51
2.3.2. Comportement de la loi de commande.....	52
2.3.3. Intérêt et inconvénients.....	53
2.4. Comparaison entre l'asservissement visuel 2D et 3D.....	54
2.5. Asservissement visuel 2D1/2.....	57
2.5.1. Modélisation d'un asservissement visuel 2D1/2	61
2.6. Asservissement visuel 2D3/4	62
2.7. Comparaison entre AV2D1/2 et AV2D3/4.....	63
2.8. Asservissement visuels dynamique(d2D/dt).....	64
2.9. Conclusion	66

Troisième chapitre : La commande prédictive visuelle (CPV)

3.1 Introduction.....	67
3.2. Commande prédictive.....	67
3.2.1. Principe de la commande prédictive.....	71
3.3. Stratégie de la commande prédictive.....	72
3.4 Réglage des algorithmes de MPC.....	73
3.5 Obtention du modèle.....	75
3.6. Avantages et inconvénients de la Commande prédictive.....	77
3.7. Commande prédictive visuelle.....	78

3.8. Conclusion.....	82
----------------------	----

Quatrième chapitre : Application au suivi d'une cible

4.1. Introduction.....	83
4.2. Application au suivi d'un objet.....	84
4.2.1. Description physique de l'application.....	84
4.2.2. Description fonctionnelle de l'application.....	85
4.3. Réalisation.....	86
4.4. Conclusion.....	88
Conclusion générale.....	89

Introduction générale

De nos jours, l'une des préoccupations du monde moderne est de concevoir des machines destinées à soulager l'homme dans divers secteurs d'activité. Ce besoin d'automatisation permettra d'éviter à l'opérateur humain des tâches difficiles et d'avoir une meilleure rentabilité et qualité du produit. La volonté des recherches de reproduire les capacités humaines de perception et d'action dans les systèmes robotisés a conduit à l'intégration de données issues de capteurs de vision, divers capteurs peuvent être utilisés afin de permettre aux systèmes robotiques de pouvoir appréhender leur environnement, les plus couramment utilisés sont les capteurs à ultrason, les télémètres laser, les capteurs d'effort et les caméras. Les derniers développements dans le domaine de la technologie des capteurs de vision et du traitement d'image permettent aujourd'hui l'utilisation effective des données visuelles dans la boucle de commande des robots. Ceci permet de tenir compte des incertitudes et/ou des changements dans l'environnement (par exemple, pour compenser de petites erreurs de positionnement ou pour saisir des objets convoyés sur un tapis roulant, etc.). Concernant les aspects liés à la vision, il est maintenant possible, par des approches de vision active, de contrôler le mouvement d'une caméra dans le but d'améliorer la reconnaissance, la localisation ou l'inspection de l'environnement puis les informations qui sont fournies par les capteurs de vision sous forme d'images qui subissent différents traitements afin d'extraire les informations utiles pour l'accomplissement de la tâche. Ces traitements sont en générale la détection de contour, seuillage, la segmentation et l'extraction des indices images. Pour permettre aux robots de réaliser d'une manière autonome, des tâches dans des environnements partiellement connus, ou complètement inconnus. Des études récentes ont montré que l'adjonction de la faculté de

vision dans la boucle de commande, permet d'améliorer de manière significative leurs performances et aide à développer de nouvelles applications. Ce domaine de recherche a pris le nom d'*Asservissement visuel*.

D'un point de vue méthodologie, l'asservissement visuel consiste à intégrer directement dans la boucle de commande des robots, des informations extraites des images fournies par des caméras afin de réaliser l'action souhaitée, cela permet un élargissement important du domaine d'application de la robotique, et une amélioration considérable de la précision obtenue.

Ce mémoire a pour but d'étudier théoriquement l'asservissement visuel et de réaliser une tâche de suivie de cible, il se divise en quatre chapitres, qui portent, respectivement, sur les points suivants:

- Nous dresserons, dans le premier chapitre, des généralités sur l'asservissement visuel où on décrira brièvement ces différentes tâches ainsi que la classification des asservissements visuels, on étudiera ensuite, l'intégration de la vision en robotique où on traitera le calibrage ainsi que quelques notions de base de la modélisation d'une caméra.
- Dans le deuxième chapitre, nous détaillerons les différentes techniques d'asservissement visuel, toute en faisant des comparaisons entre elles.
- Dans le troisième chapitre, nous traitons en premier lieu la commande prédictive, en suite, nous l'appliquons à l'asservissement visuel d'où, la commande prédictive visuelle.
- Enfin, le quatrième et dernier chapitre, sera consacré à l'implémentation de l'asservissement visuel où nous avons choisis de réaliser la tâche de suivi de cible.

Nous terminerons avec une conclusion générale, cette dernière sera un résumé sur les différents points abordés dans les chapitres qui composent ce mémoire .

Premier chapitre

Généralités sur l'asservissement visuel

1.1 Introduction

L'asservissement visuel est une jonction entre la perception de l'environnement et une action à mener dans cet environnement par un robot. C'est l'intersection de la robotique, l'automatique et la vision par ordinateur. De la perception via un capteur, naît une action ou mouvement du robot. Du point de vue commande, ceci constitue donc l'asservissement visuel et du point de vue de la perception, il s'agit de vision active. La perception et l'action ont été considérées comme complètement indépendantes depuis longtemps. Cependant, depuis quelques années, les deux communautés se sont rapprochées et considèrent ces deux domaines comme étroitement liés. Ainsi, dans ce chapitre, le processus d'utilisation de la vision dans la commande des robots connu sous le nom d'asservissement visuel est présenté. Pour ce faire, un historique et un état de l'art ainsi que des définitions et des principes sont présentés. Nous montrons aussi les différentes configurations et des techniques de l'asservissement visuel.

Nous terminons par une étude sur l'intégration de la vision en robotique. L'asservissement visuel consiste donc à contrôler un système en utilisant des informations extraites des images fournies par une ou plusieurs caméras.

1.2. Historique

Historiquement, les premières expérimentations combinaient les informations visuelles aux mouvements d'un robot dans une approche en boucle ouverte présentant successivement les opérations « regarde » (looking), puis « bouge » (moving). La précision du système ainsi réalisé est liée à la précision du capteur visuel, à celle du robot, ainsi qu'à celle de sa commande. Pour améliorer la précision des trois briques qui le composent, une alternative à cette escalade dans la recherche qualitative des sous-éléments du système a été créée en introduisant une boucle de commande par retour ou visuel-feedback. C'est ainsi que le terme d'asservissement visuel, *visuel servoing* en anglais, a été introduit par la première fois par Hill et Park en 1979 pour distinguer leur approche d'expérimentation antérieures dites « blocks world » dans les quelles les systèmes alternaient les étapes « prises d'image » puis « bouge » [1]. Notons que les premiers travaux rapportés sur l'asservissement visuel remontent à 1980. Les progrès dans l'art de commander un robot par vision ont, depuis quelque années, le domaine a connu un regain d'intérêt, comme le prouve l'abondante littérature qui est consacrée [2][3]. Vers 1970 et à cause des limitations techniques et technologiques, les premiers travaux ne correspondaient pas exactement à la définition donnée de commande qui consiste à commander le déplacement d'un système dynamique grâce à des informations visuelles issues d'un ou plusieurs capteurs visuels. Ils s'apparentent plutôt à des approches du types « regarde » puis « bouge ». Rapidement, des progrès sont venus atteindre des vitesses d'asservissement importantes pour poursuivre des cibles et saisir des objets en mouvement, souder des jointures. L'une des premières références à ce sujet est celle de Shirai et Inoue en 1973 [4] qui ont décrit un système possédant

une boucle de retour par la vision pour corriger la position d'un robot. Le robot saisit un prisme et le range dans une boîte via un asservissement visuel primitif. Les déterminations de la position et de l'orientation de la boîte sont réalisées par détection de contours. La caméra est fixe et le temps d'exécution d'un cycle était évaluée dans l'ordre de 10 secondes. En 1979, Hill et Park ont décrit l'asservissement visuel du robot *Unimate* [1]. Le traitement d'image de type binaire, pour garantir une cadence suffisante, fournissait une information de position 2D, ainsi qu'une estimation grossière de la profondeur tirée des distances entre des caractéristiques images connues. Leurs expériences ont démontré la faisabilité du guidage plan et 3D d'un robot en mouvement. Dans les années 80, la commande référencée capteur a été développée par C. Samson et B. Espiau et son application utilisant une caméra a été réalisée par F. Chaumette fin des années 80. Depuis, un certain nombre de chercheurs se sont focalisés sur le problème de l'asservissement visuel. En 1980, Gilber et son équipe décrivent un système d'asservissement d'une caméra de poursuite de fusées [5]. Leur système utilise un matériel de traitement d'image dédié à l'identification de la cible, et à sa poursuite à la cadence de 60 Hz. En 1981, Geschke présente l'asservissement d'un bras manipulateur par stéréovision [6]. Le traitement d'image fonctionne à 10Hz, et les boucles d'asservissement à la même vitesse. Dans sa thèse, en 1984, Weiss propose l'utilisation d'une commande adaptative pour réaliser un asservissement stationnaire fondé sur l'image [7]. Son approche permet d'estimer la relation non linéaire et non stationnaire qui existe entre la posture du robot et les caractéristiques images. En 1986, Dzialo et Schalkoff lancent l'idée de la commande d'une caméra montée sur une tête robotique pour des applications de poursuite de cibles [8]. En 1988, Kabuka présente un système qui utilise les projections d'objets dans une image binaire [9]. Il applique une transformation de Fourier sur les axes verticaux et horizontaux pour centrer une cible dans le plan caméra et en calculer l'orientation. La caméra est asservie sur une plate-forme à deux axes par un ordinateur IBM PC/XT et met 30 secondes pour être centrée sur la cible. Son approche est décrite de manière générale, mais seules des expérimentations dans le plan ont été menées. En 1990, Venkatesan et Archibal ont proposé un système pour asservir

un robot à cinq degrés de liberté à partir d'information d'un système à deux lasers montés sur une tête [7]. L'extraction en temps réel de primitives visuelles pertinentes et précises est une des contraintes strictement nécessaires à l'élaboration d'un schéma d'asservissement visuel. Cette contrainte est levée peu à peu avec l'évolution de la technologie des processeurs. Par rapport à ce temps réel, une application a été implémentée par Montagne et al. en 1994 pour estimer l'écart relatif entre deux véhicules qui se suivent. D'autres travaux se sont intéressés à plusieurs autres techniques comme par exemple, le suivi d'objets dont la vitesse est supposée constante mais inconnue, le suivi d'un objet plan à partir d'un estimateur de mouvement, la mise en œuvre de différentes techniques de filtrage de Kalman pour détecter, estimer et compenser les ruptures de modèles de mouvement, la modélisation du mouvement de l'objet comme une perturbation déterministe, l'utilisation d'une loi de commande optimale pour assurer le suivi d'un objet, le suivi d'objet basé sur un modèle d'état à vitesse constante etc. Par ailleurs, des primitives visuelles dynamiques, autres que géométriques, ont été utilisées [10][11]. D'autres auteurs se sont intéressés aux problèmes de singularité, de robustesse et de stabilité liés à l'asservissement visuel 2D. Ainsi, H. Michel montre l'existence de cas de singularités [12]. Dans leurs travaux, Marchand et al, proposent l'utilisation de tâches secondaires pour l'évitement de butées et de singularités et Espiau analyse l'effet des erreurs de calibrage de caméra sur l'asservissement visuel et propose une étude de cas particulier de stabilité [13]. En 1998, F. Chaumette [14] présente une étude sur les problèmes de stabilité et de convergence en asservissement visuel. D'autres auteurs se sont intéressés à la mise en œuvre de techniques de commande avancées appliquées au cas particulier de l'asservissement visuel. Dans le contrôle prédictif, on peut citer les travaux de J.A. Gangloff et al [15] qui propose l'application de technique contrôle prédictif généralisé. En ce qui concerne la commande optimale, K. Hashimoto développe une approche de commande optimale LQ et traite le cas où le nombre de primitives visuelles est redondant [16]. Dans leurs travaux, N. Papanikolopoulos et al., mettent en œuvre une technique de contrôle optimal de type LQG appliquée dans le cas d'objets déformables ainsi qu'une technique de commande adaptative [17]. P. Corke, dans

ses travaux, utilise un retour d'état avec placement de pôles [18]. Dans sa thèse, D. Khadraoui a appliqué des techniques de placement de pôles et de commande H_∞ pour le contrôle latéral de robots mobiles [19].

1.3. Asservissement visuel

Comme cela est vu précédemment, on peut remarquer que l'emploi de la vision en robotique s'est développé de manière progressive et en fonction des progrès récents dans le domaine des capteurs de vision et de traitement d'images. Actuellement, les systèmes de vision sont intégrés avec les systèmes de programmation du robot. Les possibilités vont d'un simple traitement d'images binaires aux systèmes d'extraction de contours et de primitives. Ainsi, le contrôle visuel est la fusion des résultats dans de nombreux domaines de recherche. L'acquisition de données localisées dans l'espace permet d'obtenir des informations spécifiques à l'environnement pour pouvoir gérer les relations entre le robot et son environnement. Les organes de perception sont des capteurs dits capteurs *proprioceptifs* lorsqu'ils mesurent l'état interne du robot comme la position et la vitesse des articulations et *extéroceptifs* lorsqu'ils recueillent des informations sur l'environnement pour la détection de présence, la mesure de distance, la vision artificielle etc. Tout mouvement du robot est effectué par pilotage des actionneurs. A partir de la fonction de la perception et des ordres de l'utilisateur, la commande permet d'engendrer les mouvements du robot. Quelle que soit la performance de la loi de commande utilisée, il est resté difficile de connaître exactement la position du robot mobile ou la situation de l'effecteur d'un robot manipulateur. Même dans le cas où le robot évolue dans un univers parfaitement connu, les capteurs proprioceptifs ne permettent pas de le situer précisément dans cet univers. Il paraît alors important d'utiliser des capteurs extéroceptifs capables de fournir des informations, soit sur l'environnement du robot, soit sur la situation du robot dans cet environnement. Le champ d'application des robots s'en trouve d'ailleurs grandement élargi. Ainsi, l'autonomie et l'efficacité d'un robot vont être augmentées grâce à l'emploi des capteurs extéroceptifs qui sont de plusieurs espèces parmi lesquels on peut citer les capteurs tactiles qui préviennent d'un contact avec un élément extérieur, les capteurs de proximité qui détectent la présence d'un objet situé à une certaine

distance inférieure à un mètre, les capteurs télémétriques pour des objets situés à une distance plus importante et les capteurs visuels qui permettent de percevoir l'environnement.

1.3.1. Définition

L'asservissement visuel est une technique de commande qui consiste à commander le déplacement d'un système dynamique grâce à des informations visuelles issues d'un ou plusieurs capteurs visuels. Il est aussi défini comme étant une technique qui consiste à contrôler un système en utilisant des informations extraites des images fournies par une ou plusieurs caméras. Ceci permet de rendre les robots plus performants et de développer de nouvelles tâches robotiques. En effet, en absence de capteur donnant la position par rapport aux objets environnants, le robot ne peut réaliser que des tâches préprogrammées et ceci, dans un environnement invariant. Par contre, avec l'intégration de capteurs de vision, des tâches non connues au préalable peuvent être envisagées même dans un environnement variant. La mise en œuvre de cette technique implique au moins trois sous-systèmes, à savoir, le capteur visuel qui est souvent une (des) caméra (s), mais peut également être tous type de dispositif imageur (scanner, IRM, caméra linéaire, etc.), un dispositif d'acquisition et de traitement, autrement dit un ordinateur chargé du traitement d'images et de la commande et un sous-système mécanique actionné représentant le robot qui est chargé de l'exécution de cette commande. Le robot génère ses mouvements de façon à ce que sa cible visuelle atteigne une certaine configuration dans l'image qu'il perçoit.

1.3.2. Principe

Il s'agit d'utiliser les informations visuelles issues d'une (des) caméra (s) pour contrôler les mouvements d'un robot. En fonction de la tâche que l'on souhaite accomplir, il est tout d'abord nécessaire de sélectionner les informations visuelles qui seront aptes à la réaliser de manière optimale. Il faut ensuite exprimer cette tâche sous forme de consigne s^* à atteindre par les informations

visuelles courantes s . Une loi de commande appropriée est alors à élaborer pour calculer, à partir de la différence (s^*-s), la consigne à envoyer au robot. Ce dernier réagira à ces commandes selon sa dynamique et l'image donnée par la caméra sera donc modifiée en conséquence. A partir des informations visuelles extraites de la nouvelle image, on peut alors calculer une nouvelle commande et ce, jusqu'à ce que l'on ait accompli la tâche désirée (fig.1.1.).

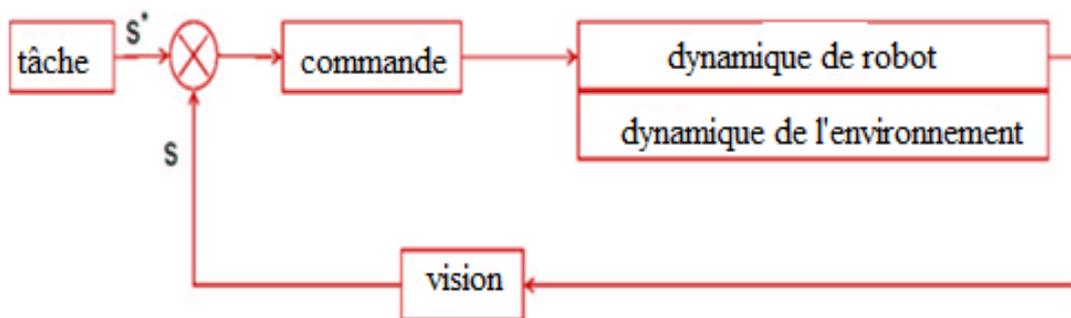
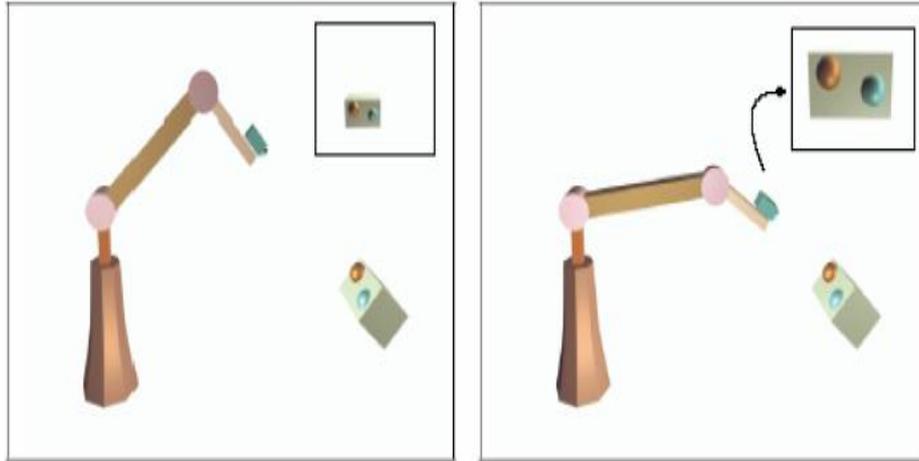


Fig.1.1 Principe de la commande référencée vision

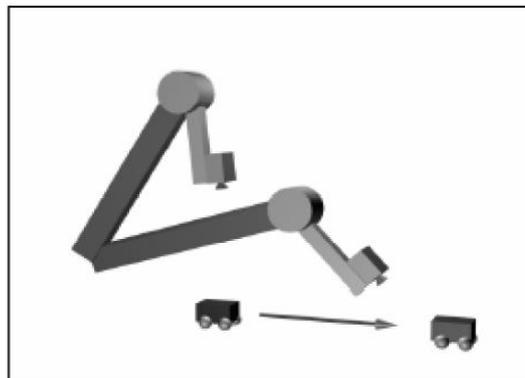
L'utilisation de la caméra comme capteur possède plusieurs avantages. En effet, la mesure est passive, il n'existe aucune interaction entre la caméra et l'environnement, les caractéristiques de la mesure sont proches à l'œil humain, la caméra est un capteur très précis, rapide et possède une grande résolution, la mesure utilisant dans la vision n'est pas coûteuse et la portée est de 50cm à 10 mètres.

1.4. Types de tâche des asservissements visuels

Avant d'effectuer la commande de robot, il est généralement nécessaire de déterminer les tâches que l'on désire lui faire accomplir. Celles-ci sont classées suivant trois grandes catégories : le positionnement, le suivi de cible et la navigation. Le but de positionnement est d'amener le robot à partir d'une position initiale à une position désirée à l'aide de la caméra. Ce type de tâches trouve de nombreuses applications dans le secteur industriel (fig.1.2).

**Fig.1.2** Positionnement

Dans le cadre d'une tâche de suivi, on considère que la cible est mobile. Le problème ne revient plus simplement à amener le robot dans une situation d'équilibre par rapport à une cible mobile, mais à estimer le mouvement de la cible de façon à pouvoir maintenir l'interaction entre la cible et le capteur (fig.1.3).

**Fig.1.3** Suivi de cible

Quant à la navigation, l'objectif est de répondre au problème fondamental qui consiste à « A partir d'une situation A, aller dans une situation B suivant un chemin ». Pour cela, le robot doit réaliser quatre actions à savoir modéliser et

interpréter l'environnement, se localiser dans cet environnement, planifier des déplacements et assurer la bonne exécution des déplacements planifiés (fig.1.4).

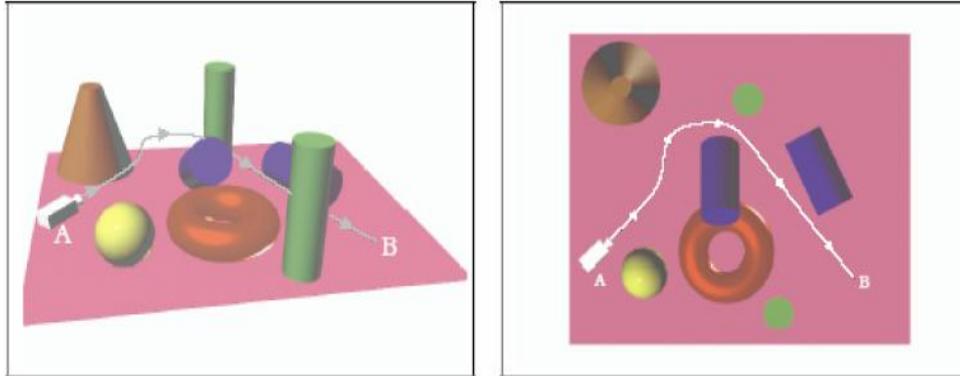


Fig.1.4 Navigation

1.5. L'approche «Look and Move»

Lors de l'acquisition de l'image, on extrait l'information permettant d'estimer la différence entre la position courante et la position désirée. Dans le cas du technique look and move, il s'agit d'un enchaînement de deux étapes. Dans la première (look), on réalise l'acquisition d'une image, dans la seconde étape (move), on calcule et on envoie les signaux de commande du robot permettant de réguler à zéro cette différence. Dans le cas où cet enchaînement est réalisé une seule fois, on parle de look and move statique. Par contre, lorsqu'il est réalisé plusieurs fois on parle de look and move dynamique. Cette approche utilise deux boucles principales de retour pour réaliser l'asservissement visuel. La boucle de retour de bas niveau et la boucle de retour de haut niveau, la première est composée des capteurs de position des articulations du robot (capteurs proprioceptifs). Elle permet généralement d'asservir les différentes articulations du robot en position et stabilise le système tandis que la seconde utilise les capteurs visuels (capteurs extéroceptifs) pour indiquer une commande de positionnement à la boucle interne (fig.1.5).

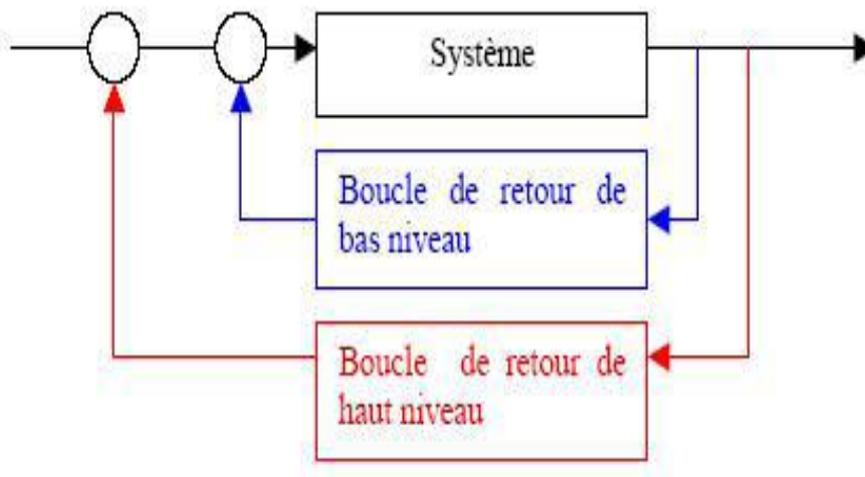


Fig.1.5 Approche “look and move”

1.5.1. L'approche « Static Look and Move »

Cette approche consiste en un enchaînement séquentiel des deux étapes. Dans la première, l'image acquise par une caméra est traitée afin d'extraire les informations nécessaires à l'estimation de la situation courante entre la caméra et l'environnement. Dans la seconde, la loi de commande calcule un déplacement en fonction de l'erreur qui existe entre la position à atteindre p et l'effecteur du robot. Ensuite, cette loi de commande envoie au robot les consignes, dans l'espace articulaire, qui lui permettront de rejoindre la cible.

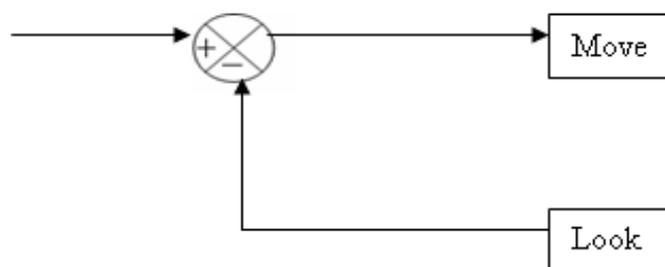


Fig.1.6 Approche static look and move

La simplicité de cette approche a permis son utilisation dès les années 1970 dans les premières applications robotiques utilisant la vision. L'inconvénient fondamental de cette méthode est qu'il faut attendre la fin du mouvement du robot avant de pouvoir acquérir une nouvelle image. Elle est caractérisée par de faibles performances au niveau temps. Elle ne permet pas de contrôler le trajet de déplacement (pas d'évitement d'obstacle possible, par exemple), mais uniquement les positions finales. Il n'est pas possible, non plus, d'utiliser cette méthode dans des scènes non statiques.

1.5.2. L'approche « Dynamic Look and Move »

Cette approche assez semblable à la précédente approche. La différence est que les étapes «look» et «move» ne s'effectuent plus de manière séquentielle mais simultanément, bien que ce soit souvent à des cadences différentes. Cette méthode s'utilise dans des scènes où l'objet est en mouvement (fig.1.7).

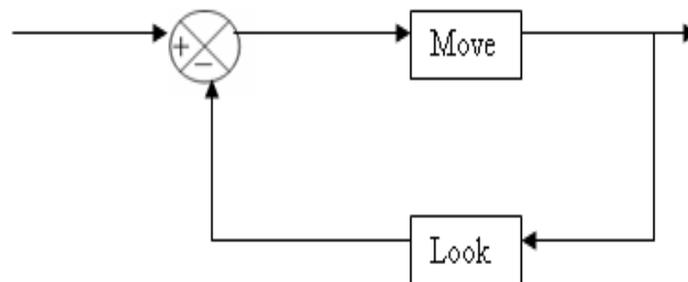


Fig.1.7 Approche «Dynamic look and move»

Théoriquement, une seule boucle devrait être suffisante pour asservir la position de la caméra à la situation voulue. Cependant, des erreurs relatives à l'extraction d'information dans les images, à l'interprétation des résultats ou bien à des modèles trop approximatifs peuvent nécessiter l'exécution de plusieurs boucles d'asservissement ce qu'on appelle «*Dynamic Look and move* ». Ces erreurs

sont dues à l'extraction de primitives dans les images, la modélisation de la caméra et la modélisation du robot.

1.6. Classification des asservissements visuels

On peut classer les systèmes d'asservissement visuel en trois types ; selon la position de capteur visuel, le type de mesure et le type de la commande.

1.6.1. Selon la position de capteur visuel

On distingue trois types de positionnement selon l'application souhaitée. La première consiste à utiliser une caméra fixe qui observe la scène les informations visuelles permettent de commander le robot mais le déplacement de celui-ci n'interagit pas sur la position de la caméra. L'avantage de cette configuration est de voir la scène de l'extérieur mais son inconvénient c'est quelle ne propose qu'une vue unique sur la scène considérée (fig.1.8).

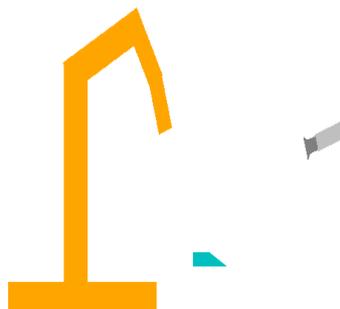


Fig.1.8 Caméra déportée

La seconde possibilité consiste à embarquer la caméra sur le manipulateur, donc évoluer dans la scène avec l'organe qu'elle contrôle dont le mouvement définit l'exécution d'une tâche comme la possibilité d'extraire, de la caméra, les données nécessaires à la régulation de la tâche envisagée. Cette configuration est dite «Eye in Hand». L'information visuelle permet non seulement la mesure de l'attitude ou de changement d'attitude d'un objet situé dans l'espace de travail

du robot, mais, également, la mesure de l'attitude ou de changement d'attitude de l'organe terminal (fig.1.9).

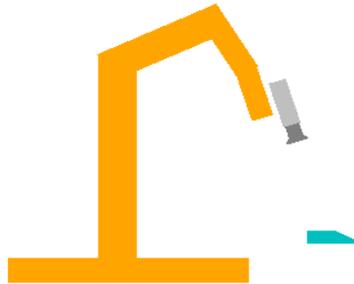


Fig.1.9 Caméra embarquée

La troisième possibilité rassemble les avantages des deux précédentes en embarquant la caméra sur un second manipulateur. La caméra observe alors la scène de l'extérieur tout ayant la possibilité de se mouvoir (fig.1.10).

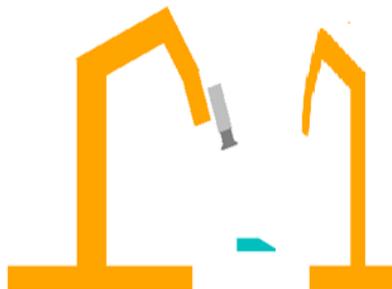


Fig.1.10 Caméra sur un second manipulateur

1.6.2. Selon le type de la commande

Les systèmes de commande, utilisés dans la vision, sont généralement classés suivant le niveau auquel la vision intervient dans la structure de la boucle de contrôle du robot. On parle alors d'asservissement visuel séquentiel, d'asservissement visuel cinématique, d'asservissement visuel cinématique et d'asservissement visuel dynamique.

1.6.2.1. Asservissement visuel séquentiel

La commande séquentielle qui effectue une approche de type «Static Look and Move» est un asservissement en boucle ouverte, qui consiste à acquérir et analyser un ensemble d'images issues du capteur, puis en générer une trajectoire à atteindre par le robot en boucle ouverte selon le but recherché. Le déplacement du robot s'effectue en aveugle et si des erreurs viennent entacher le processus de planification de trajectoire, le robot n'atteindra pas son but. Ce type de commande est utilisé pour des capteurs permettant de faibles fréquences d'acquisition et peut être utilisé pour des missions de replanification de tâches nécessitant des déplacements lents en utilisant la vision. Cependant, l'absence de retour visuel rend la convergence de l'asservissement très sensible aux incertitudes de modélisation du robot et du capteur ainsi qu'aux méthodes de reconstruction de pose. Les commandes séquentielles n'ont qu'un avantage, la simplicité. Par contre elles ne sont ni précises, ni robustes aux perturbations.

1.6.2.2. Asservissement visuel cinématique

La commande cinématique qui effectue une approche de type « Dynamic Look and Move » est un asservissement en boucle fermée où le contrôleur visuel intervient dans une boucle de haut niveau et fourni un état de consigne en vitesse assurant une convergence asymptotique qui est transformée par le générateur de trajectoires en consigne angulaire transmise au contrôleur interne bas niveau du robot qui les réalise.

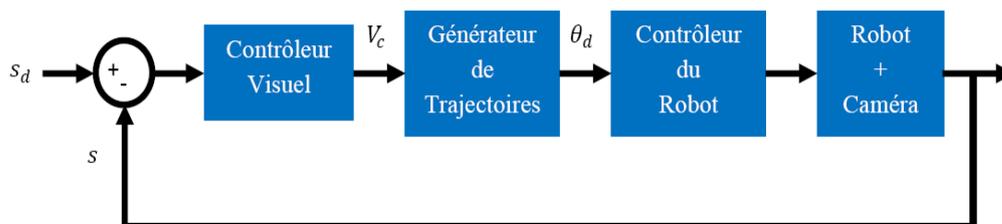


Fig.1.11 Asservissement visuel cinématique

Le robot est considéré comme un simple système de positionnement dont la dynamique est approximée par un gain, et les performances du système dépendent essentiellement de la synthèse du contrôleur visuel et du retard introduit par celui-ci dans la boucle de commande. Ce type de commande dit «indirecte» est utilisé pour des asservissements visuels lents (leur période d'échantillonnage est grande). Pour plusieurs raisons, la plupart des systèmes d'asservissement visuel ont adopté cette architecture :

- La faible vitesse d'acquisition des capteurs de vision augmente la complexité du contrôle de l'effecteur final du robot du à des non linéarités.
- L'utilisation d'un retour interne permet un meilleur contrôle.
- Les systèmes robotique sont généralement livrés avec une interface permettant une commande en vitesse ou en position.
- Une architecture « Look and Move » permet de séparer les singularités du modèle cinématique des singularités du contrôleur visuel, assurant ainsi un découplage entre la commande interne et la commande visuel.

1.6.2.3 Asservissement visuel dynamique

La commande dynamique est aussi une alternative de l'approche «Look and Move» d'asservissement en boucle fermée, où cette fois-ci le contrôleur visuel se substitue au contrôleur interne du robot en envoyant directement les consignes en vitesses ou en couples aux variateurs des moteurs. La conception du contrôleur visuel tient en compte d'un comportement plus réaliste de la dynamique du robot et parfois même de celui du capteur visuel et /ou des actionneurs pour améliorer les performances dynamiques du système (fig.1.13).

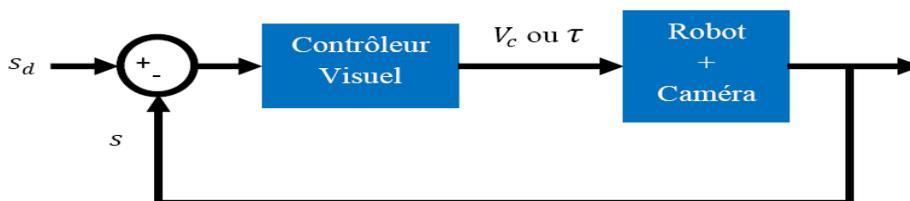


Fig.1.13 Asservissement visuel dynamique.

La commande dynamique a pour objectif de maximiser les performances dynamiques de la boucle de vision. Pour cela il est nécessaire de tenir compte de tous les effets dynamiques (retards, modes oscillants mécaniques, constantes du temps électriques, etc.) dans la synthèse de la loi de commande. Ce type de commande dit « directe » est utilisé pour des asservissements visuels rapides (leur période d'échantillonnage est petite). L'asservissement visuel direct tente de remédier à ces problèmes en supprimant la boucle d'asservissement interne du robot (asservissement en position ou en vitesse). Le contrôle est entièrement effectué par la boucle visuelle qui calcule les valeurs à donner aux articulations du robot.

1.6.3. Selon le type de mesure

1.6.3.1. Asservissement visuel 3D

Ce type d'asservissement utilise l'information visuelle exprimée dans l'espace cartésien 3D en entrée de la loi de commande. Ces informations sont obtenues à partir de la pose relative de l'objet par rapport à la caméra, cette dernière définit la position et l'orientation d'un repère lié à l'objet par rapport à un repère lié à la caméra elle peut être calculée par plusieurs méthodes, la plus utilisée est basée sur le modèle géométrique de l'objet. Le principe de l'asservissement visuel 3D est de contrôler la caméra dans l'espace cartésien en construisant la fonction de tâche à partir de l'estimation de déplacement entre deux images. L'avantage est de spécifier la trajectoire de la caméra dans l'espace où elle est le plus facilement décrite, ce qui permet aussi d'assurer une trajectoire réalisable de la caméra. L'inconvénient est qu'il nécessite une étape de la reconstruction 3D qui requiert des connaissances a priori de la géométrie de la scène et aussi manque de robustesse aux erreurs d'étalonnage du système de vision, en effet, de petites erreurs dans l'image peut entraîner une large erreur dans l'estimation de la pose. Cette erreur peut entraîner une instabilité de la commande ou un échec dans le positionnement. Le traitement d'image se limite

alors à une détection des contours des objets et à un calcul du centre de gravité des points de contour afin de déterminer le centre de chaque objet (fig.1.14).

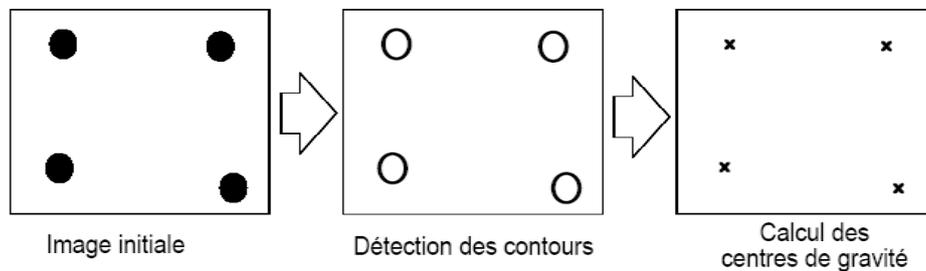


Fig.1.14 Traitement d'image d'une cible

Utiliser une caméra dans un environnement inconnu ne suffit pas pour l'estimation de la pose mais aussi. En effet, il faut considérer que la cible désirée est connue, utiliser deux caméras pour observer la même scène, et mettre en correspondance les images obtenues (fig.1.15).

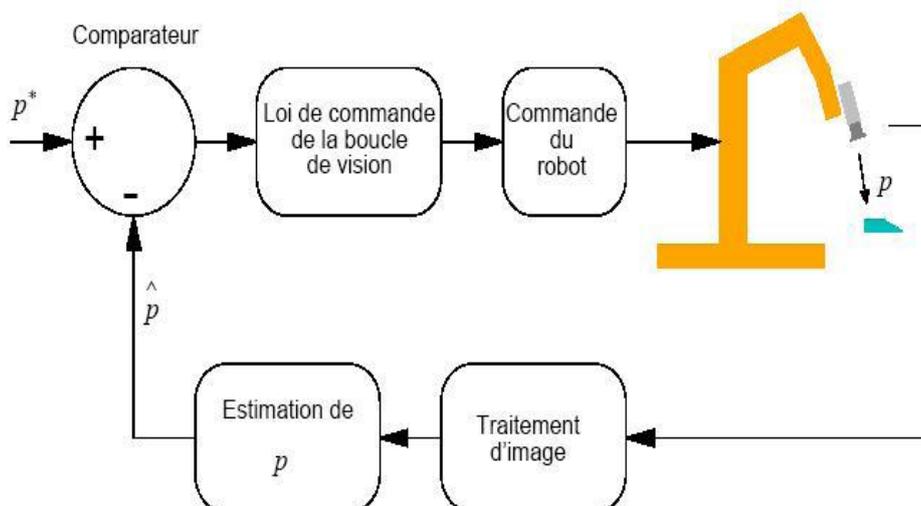


Fig.1.15 Principe d'asservissement visuel 3D

1.6.3.2. Asservissement visuel 2D

En asservissement 2D, les consignes ne sont plus exprimées sous la forme de situation entre la caméra et la scène, mais sous forme d'un motif à atteindre dans l'image. Ces motifs sont le plus souvent des formes géométriques élémentaires (point, segment de droite, cercle, sphère, portion d'ellipse...etc.). La plupart des travaux qui traitent de l'asservissement 2D utilisent des primitives constituées de points. Ces derniers peuvent être situés à l'intersection de segments dans l'image ou encore être extraits du centre de gravité de la projection des disques coplanaires. Ceux-ci nécessitent un temps de traitement d'image faible. 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 délai d'application de la commande est réduit, ce qui contribue à la stabilité du système. Un des principaux avantages de l'asservissement visuel 2D est qu'il n'est pas nécessaire de connaître a priori le modèle de l'objet. Une approximation de la profondeur de l'objet par rapport à la caméra est généralement suffisante pour réaliser le schéma de commande. Ce dernier est en pratique robuste aux bruits de mesure et aux erreurs de calibration de système dans le cas où le déplacement à réaliser entre la position désirée de robot et sa position initiale est peu important, et son inconvénient est qu'elle n'assure pas une trajectoire réalisable de la caméra dans l'espace (fig.1.17).

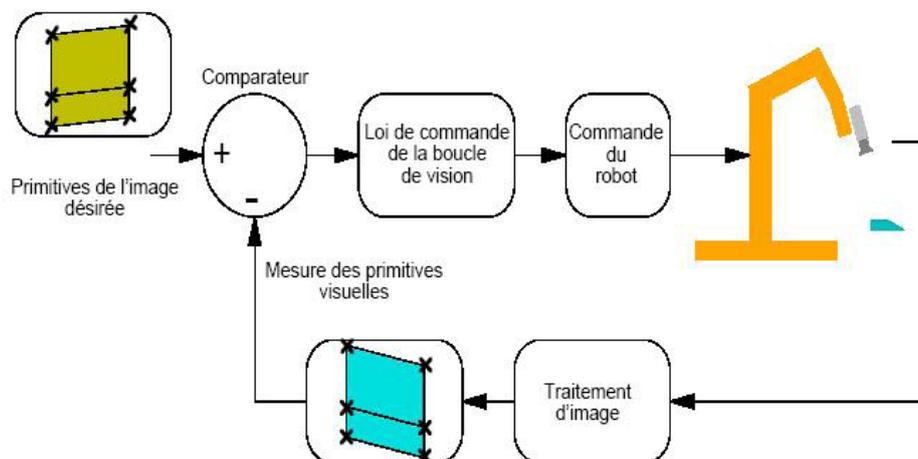


Fig.1.16 : Asservissement visuel 2D

1.6.3.3. Asservissement visuel 2D1/2

Pour profiter des avantages de l'asservissement visuel 2D (contrôle de l'image de l'objet) et 3D (control de la trajectoire du robot) toute en évitant leurs limites respectives, il est possible d'utiliser une méthode hybride qui combine à la fois des informations visuelles, contrairement à l'asservissement visuel 3D, cette estimation ne nécessite pas de connaître le modèle géométrique de l'objet, mais reste insensible aux bruits dans l'image. Les contraintes imposées sont donc l'estimation dans la scène dites des primitives (motifs), ainsi que la possibilité de les extraire et de les suivre grâce aux traitements d'images à une cadence suffisamment élevée pour conserver la robustesse et la stabilité des lois de commande. Cette approche est basée sur l'estimation de l'homographie (méthode d'estimation de la profondeur «z» entre caméra et objet). Elle permet de calculer le déplacement en rotation Ω que la caméra doit effectuer pour atteindre sa position spécifiée ainsi que la direction de son déplacement en translation (torseur cinématique). L'homographie fournit également le rapport d/d^* le rapport entre les distances courantes et désirées de la caméra à l'objet. On peut ainsi séparer les boucles d'asservissement en rotation et en translation de la caméra (fig.1.17), ce qui permet :

- Un fort découplage de la loi de commande.
- Un contrôle partiel dans l'image permettant de conserver l'objet en permanence dans le champ de vision de la caméra.
- Une étude de la stabilité et du domaine de convergence de la loi de commande.

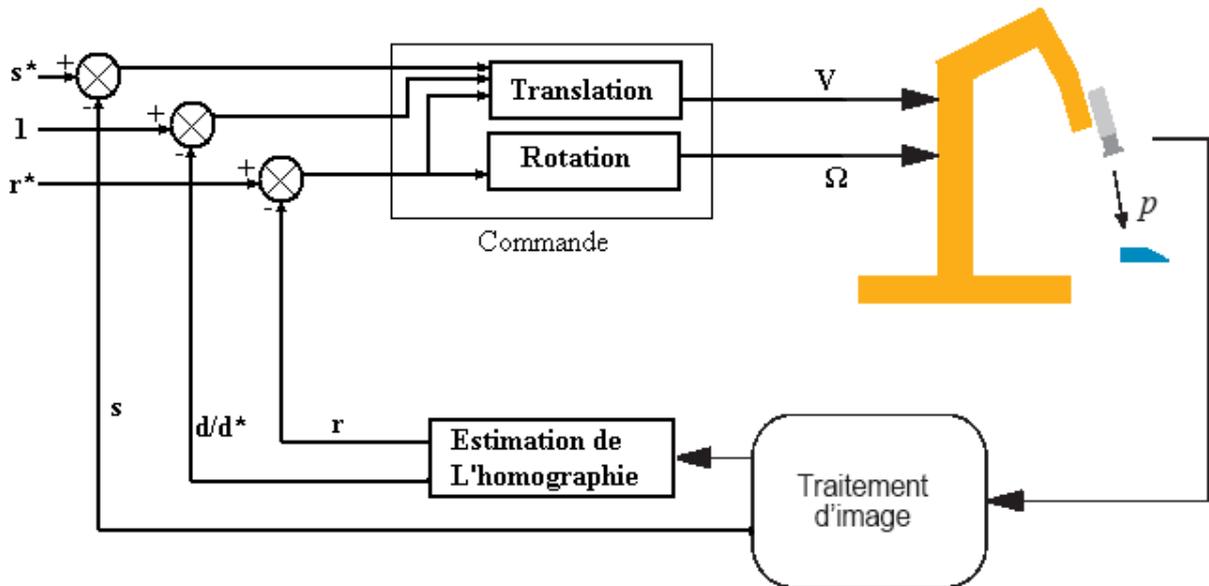


Fig.1.17 : Asservissement visuel 2D1/2

1.6.3.4. Asservissement visuel d2D/dt

Les mesures extraites des séquences d'images sont dynamiques. Les créateurs de cette méthode ne remontent plus cette fois aux primitives visuelles géométriques, mais proposent de contraindre directement les mesures dynamiques de à ce qu'elles atteignent des valeurs souhaitées. Dans ce type d'asservissement, la grandeur asservie est une vitesse relative entre la caméra et la cible. Elle est définie dans le plan image par un champ de vitesse des points. Le principe de la commande consiste alors à contrôler les mouvements de la caméra de telle sorte que le mouvement 2D mesuré atteigne un champ de vitesse désiré d'où l'application d'asservissement visuel d2D/dt. La différence entre asservissement visuel 2D et l'asservissement d2D/dt c'est que 2D nécessite le passage par d'extraction des primitives alors que dans le cas des asservissements visuel d2D/dt, on souhaite seulement atteindre un champ de vitesse désiré. Ce type d'asservissement est à un stade émergent. Actuellement, les traitements d'images qu'il nécessite sont encore très longs et fournissent des informations très bruitées. Contrairement au cas de mesures géométriques, la mesure \underline{s} ainsi que sa valeur désirée s^* correspondent désormais à des paramètres P_i du modèle de mouvement 2D dans l'image. On voit alors que la loi

de commande ne contraint plus seulement le torseur cinématique V de la caméra mais aussi son accélération Γ (fig.1.18).

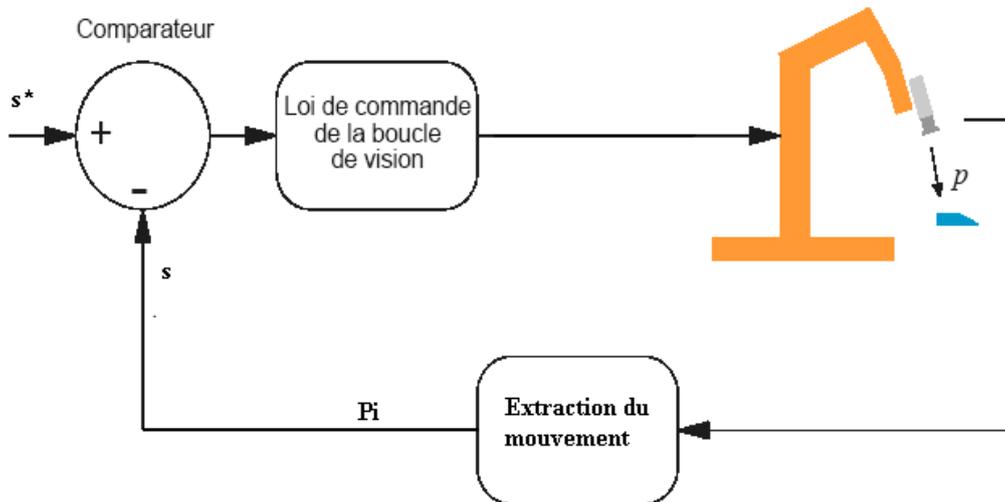


Fig.1.18 Asservissement visuel d2D/dt

1.7. 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 et les informations visuelles de nature géométrique. Les informations visuelles dynamiques [20][21][22] 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 exclusivement géométriques, le cas de figure typique étant celui où l'objet observé est inconnu. Dans le cas informations visuelles de nature géométrique, c'est la primitive point qui reste encore la plus couramment employée [23]. Mais on peut aussi utiliser des droites [24][25], des segments, des sphères, des ellipses [24]. Le choix des informations visuelles doit tenir compte de deux conditions essentielles: Tout d'abord, une condition évidemment nécessaire est qu'il existe des algorithmes de traitement d'image capable de mesurer cette information, et ce à un rythme le plus proche possible du rythme d'échantillonnage de la commande de robot. La seconde condition est qu'il soit possible de calculer de manière explicite le jacobien de cette information

visuelle. Le besoin d'extraire des informations fiables des images acquises implique l'utilisation d'un traitement d'image qui peut se révéler coûteux en temps de calcul. Cela vient alors en contradiction avec la nécessité de contrôler les mouvements du robot à une cadence élevée pour garantir les plus grandes réactivités et stabilité possibles. C'est pourquoi, la plus part des études en *asservissement visuel* ont, dans un premier temps, n'utilisaient que des objets simples permettant 'alléger l'algorithme d'extraction des primitives visuelles utilisées.

1.8. Approche fonction de tâche en asservissement visuel

Une tâche robotique peut généralement être exprimée comme la régulation. La réalisation d'une fonction de tâche à l'aide de l'asservissement visuel nécessite la sélection d'informations adéquates. Cela revient à définir une fonction de tâche ayant les propriétés assurant la capacité de réaliser la tâche suivie. L'élaboration d'une loi de commande en boucle fermée repose sur ces informations. Elle a pour but de réguler cette tâche. Une fois que la fonction de tâche est définie, on doit concevoir une loi de commande afin de la réguler à zéro. les fonctions de tâches classiques sont :

Dans l'espace articulaire, une fonction de tâche permettant une régulation en position peut s'écrire :

$$e(q,t) = C(s(q,t) - s^*(t)) \quad (1.1)$$

où $s(q,t)$ représente l'ensemble des informations visuelles pour réaliser la tâche, $s^*(t)$ décrit la trajectoire idéale à suivre dans l'espace articulaire et C est une matrice de découplage permettant de prendre en compte un nombre informations supérieur au nombre de degrés de liberté à commander. Ce type de fonction est utilisé dans les asservissements visuels 2D car le contrôle s'effectue dans l'image [24].

Dans l'espace cartésien, en suivant le même principe et pour un contrôle en situation, en obtient :

$$e(q, t) = r(s(q, t)) - r(s^*) \quad (1.2)$$

où la position r de l'effecteur de robot est estimée à partir des informations visuelles. Ce type de fonction est utilisé dans les asservissements visuels 3D car le contrôle s'effectue dans l'espace cartésien [26].

1.9. Obtention de la consigne s^* [27]

Une des difficultés majeures de la reconstruction de fonction de tâche est la définition de la trajectoire désirée s^* . En effet il est nécessaire de construire une fonction de tâche de telle manière qu'elle soit nulle si les informations visuelles courantes coïncident avec les informations visuelles de référence. Une première méthode pour définir les informations visuelles désirées est basé sur leur modélisation en utilisant des connaissances a priori sur la scène observée. Cette méthode demande, entre autre, une connaissance parfaite sur la géométrie de système de vision pour que s^* puisse physiquement exister. Une technique pour éviter cette phase de modélisation consiste à effectuer un apprentissage expérimental de l'image de référence. La définition de la consigne $s^*(t)$ est un problème primordial pour la reconstruction de la fonction de tâche. Les méthodes que l'on retrouve le plus souvent dans la littérature considèrent une consigne fixe, soit $s^*(t)=s^*$. Ce choix nécessite l'utilisation de lois de commande stabilisant globalement le système afin d'assurer la convergence de la fonction de tâche. En autre, en présence d'erreur de modélisation et lorsque le déplacement à effectuer est important, les trajectoires obtenues peuvent être surprenantes et bien éloignées de celles entendues. Par conséquent, il est difficile de contrôler les trajectoires réalisées et d'y intégrer des contraintes. La consigne fixe est généralement obtenue, soit en fixant *a priori* la pose de la caméra, soit par apprentissage. Dans le premier cas, lorsque des informations 2D sont nécessaires, elles sont obtenues en utilisant un modèle précis de la scène et de la caméra en appliquant les équations de projection perspective. Les informations 3D sont, quant à elles, fixées par l'utilisateur et obtenues dans

divers repères en utilisant les matrices de passage connues. Cette méthode nécessite une connaissance parfaite du système de vision et de l'objet d'intérêt pour obtenir une consigne qui soit physiquement réalisable et que pour celle-ci corresponde bien à l'objectif fixé. Dans le second cas, la méthode consiste à effectuer un apprentissage expérimental de la consigne. Le robot est d'abord amené dans la configuration désirée et l'image correspondante est acquise. Ensuite le vecteur s^* est obtenu en appliquant les méthodes utilisées pour le calcul de l'état courant s . Cette méthode a l'avantage sur la première de fournir une consigne qui correspond effectivement à l'objectif fixé même en présence d'éventuelles erreurs de modélisation. De ce fait, cette méthode est plus adaptée aux problèmes posés en asservissement visuel.

1.10. Obtention des images

Dans cette partie nous nous intéressons à la modélisation du capteur de vision utilisé, la caméra. En effet pour pouvoir tirer partie au mieux des mesures fournies par la caméra, il est nécessaire de modéliser le mécanisme d'acquisition des données, afin d'établir une relation entre celle-ci et le monde réel. Autrement dit, dans le cas d'une caméra, nous cherchons à connaître la relation entre l'image obtenue et la scène observée. Notons que nous nous contentons ici de présenter les aspects de modélisation géométrique. Le modèle de la caméra le plus couramment utilisé est le modèle sténopé, reposant sur la projection perspective. Il s'agit d'une modélisation simple et linéaire du processus de formation des images au sein d'une caméra. Il suffit d'exprimer les relations de passage du repère monde au repère caméra, d'exprimer la projection du repère caméra dans le plan image et d'appliquer la transformation affine qui conduit aux coordonnées de l'image. Le modèle sténopé permet de lier les coordonnées d'un point dans l'espace 3D à ces coordonnées en pixels dans l'image, selon le modèle de projection illustré sur la figure suivante (fig.1.19):

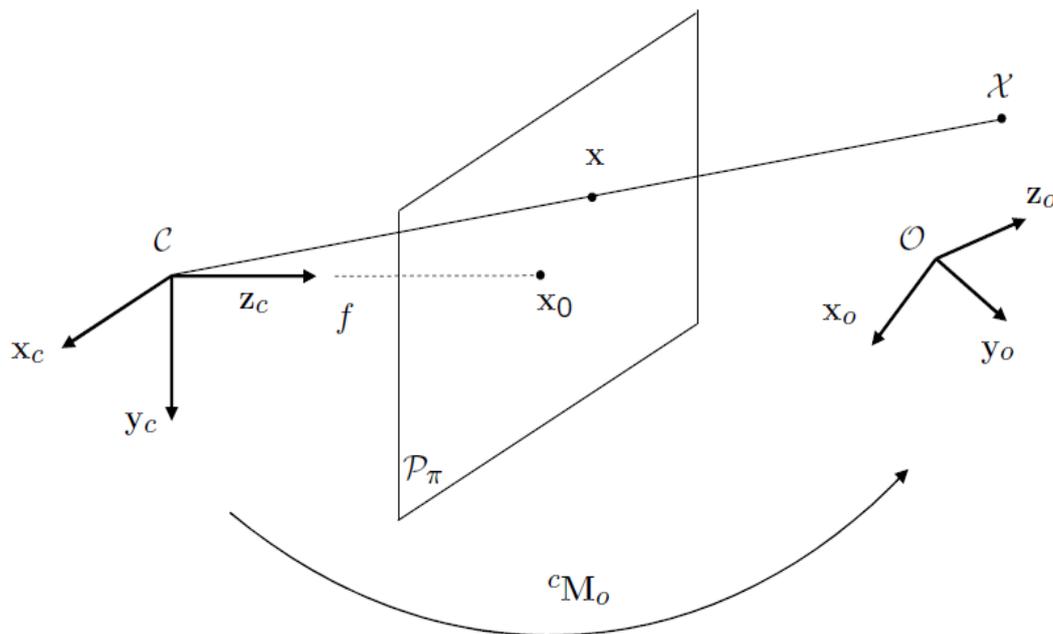


Fig.1.19 Modèle de projection

Il peut se décomposer en trois étapes : Il s'agit tout d'abord d'exprimer les coordonnées de point 3D considéré dans le repère lié à la caméra à l'aide des paramètres extrinsèques de celle-ci. Ensuite le modèle projection perspective de la caméra permet d'exprimer les coordonnées de l'image. Enfin, on obtient les coordonnées préexiliques du point image à l'aide des paramètres intrinsèques de la caméra.

1.10.1. Changement de repère

Notons $F_c=(C,x_c,y_c,z_c)$ le repère de la caméra dont l'origine se situe au centre de projection C . Soit X un point dans l'espace 3D de coordonnées homogènes $P_0=(X,Y,Z,1)$ dans un repère fixe $F_0=(O, x_o, y_o, z_o)$ lié à la scène. La première étape, de changement de repère, consiste à estimer les coordonnées de point X dans le repère F_c (voir figure). Pour cela, la connaissance de l'orientation et la position de la caméra par rapport au repère de l'objet est indispensable. On note cR_o et

${}^c t_o$ la matrice de rotation et le vecteur de translation respectivement permettant de passer du repère F_o au repère F_c . On note ${}^c M_o$:

$${}^c M_o = \begin{bmatrix} {}^c R_o & {}^c t_o \\ 0 & 1 \end{bmatrix}$$

La matrice homogène de changement de repère de F_o vers F_c .

Notons $P_c = (X_c, Y_c, Z_c, 1)$ les coordonnées homogènes de point X exprimées dans le repère de la caméra. On a :

$$P_c = {}^c M_o P_o$$

1.10.2. Projection perspective

L'étape suivante consiste à projeter le point X sur le plan image de la caméra. Le plan défini par les axes (C, x_c) et (C, y_c) est parallèle au plan image $P\pi$. La distance minimale du centre de projection C au plan image notée f est appelé distance focale. L'axe (C, z_c) coupe le plan image $P\pi$ en un point x_o , appelé point principale, de coordonnées $(0, 0, f)$ dans F_c . En absence de phénomène de distorsion, ce point \mathbf{X} se projette sur le plan image $P\pi$ en un point x de coordonnées $p = (x_c, y_c, z_c)$ dans F_c telles que :

$$x_c = f \frac{X_c}{Z_c}, \quad y_c = f \frac{Y_c}{Z_c}, \quad z_c = f$$

Puisque tous les points de plan image $P\pi$ ont comme coordonnées selon l'axe z_c la distance focale f , il est plus commode de travailler avec des coordonnées normalisées, en ramenant la dernière composante à 1. On considère donc les coordonnées normalisées $p' = (x, y, 1)$ telles que :

$$P' = \frac{1}{f} p$$

Notons que l'opération de projection s'écrit sous la forme matricielle suivante :

$$Z_c p' = A p_c$$

avec

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

1.10.3. Passage aux coordonnées pixelliques

La dernière étape consiste maintenant à exprimer les coordonnées en pixels. Il faut alors tenir compte de la position du point principal sur le plan image qui est rarement au centre, ainsi que la taille des pixels qui n'est pas forcément la même selon chaque axe (fig1.20).

Les coordonnées pixelliques du point x , noter u et v , s'expriment alors de la façon suivante :

$$u = u_0 + \frac{x_c}{l_x} ; \quad v = v_0 + \frac{y_c}{l_y}$$

où (u_0, v_0) sont les coordonnées pixelliques du point principal, et l_x et l_y la taille des pixels respectivement selon les directions x_c et y_c .

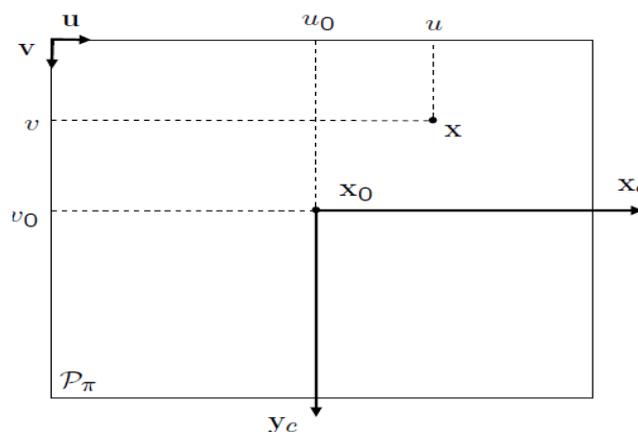


Fig.1.20 Coordonnées pixelliques

Notons qu'on peut à nouveau exprimer cette relation sous forme matricielle :

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = K' p$$

avec :

$$K' = \begin{pmatrix} 1/l_x & 0 & u_0/f \\ 0 & 1/l_y & v_0/f \\ 0 & 0 & 1/f \end{pmatrix}$$

En passant aux coordonnées normalisées p' , on voit que l'on peut faire entrer le paramètre f dans la matrice K' . Cela permet de rassembler tous les paramètres intrinsèques de la caméra au sein d'une seule matrice notée usuellement K . On obtient alors :

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = K p'$$

avec :

$$K = \begin{pmatrix} f/l_x & 0 & u_0 \\ 0 & f/l_y & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

1.11. Calibrage capteur/robot (camera /pince)

La connaissance de la relation géométrique entre la caméra et l'effecteur va nous permettre d'exprimer ces mesures dans le repère de l'effecteur. En fin, ces mesures pourraient être exprimées par rapport à la base du robot si l'on connaît

le modèle géométrique direct de ce dernier. Dans l'asservissement visuel, on cherche à commander le mouvement 3D de la caméra à partir des informations visuelles. La connaissance de la relation géométrique caméra-effecteur va nous permettre d'obtenir précisément le mouvement correspondant du robot. En traitement d'image, l'opération de calibration de caméra revient à modéliser le processus de formation des images, c'est-à-dire trouver la relation entre les coordonnées spatiales d'un point de l'espace avec le point associé dans l'image prise par la caméra. Nous appelons *calibrage caméra/pince* le processus avec lequel on détermine la relation géométrique entre la caméra et l'effecteur du robot ou sa dernière articulation, autrement dit, il consiste à estimer la transformation rigide (rotation et translation) entre un repère attaché à la caméra et un repère attaché à la pince.

1.11.1. Formulation de problème de calibrage [28]

Le problème de calibrage caméra / pince consiste à déterminer la transformation rigide (rotation et translation) entre la caméra montée sur la pince et la pince elle-même. En d'autres termes, on cherche à déterminer la relation rigide entre le repère de la caméra et repère de la pince. En termes d'équations, c'est la recherche d'une matrice homogène de passage X entre la base de l'effecteur (représenté par le repère R_e) et la base de la caméra (représenté par le repère R_c) (fig.1.21).

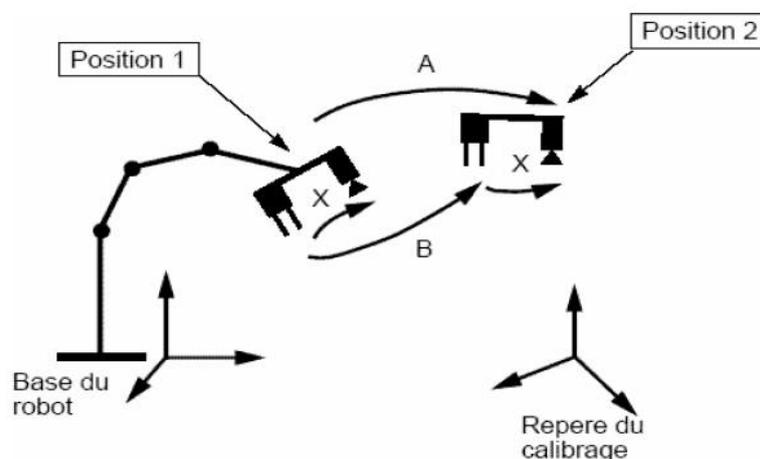


Fig.1.21 Formulation de problème de calibrage

Ici, X représente une rotation R_x et une translation P_x tel que : $X = \begin{bmatrix} R_x & P_x \\ 0 & 1 \end{bmatrix}$

1.11.2. Solution au problème de calibrage [29]

Il existe une multitude de solution possible pour le problème de calibrage ; faisant même objet de thèses, cependant, presque toutes ces méthodes reposent sur le même principe, ce dernier est exposé ci-dessous. On place l'ensemble effecteur-caméra en deux positions dans l'espace. Nous désignant par A (respectivement B) la matrice homogène de passage de R_c^1 (de la position 1) à R_c^2 (de la position 2), (respectivement de R_e^1 à R_e^2). Ainsi, la relation matricielle qui doit être résolu est la suivante :

$$XA = BX \quad \Rightarrow \quad \begin{bmatrix} R_x & P_x \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_A & P_A \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_B & P_B \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_x & P_x \\ 0 & 1 \end{bmatrix} \quad (*)$$

En représentant A (respectivement B) par une matrice de rotation R_A (respectivement R_B) et une matrice de translation P_A (respectivement P_B), la relation précédant se met sous la forme :

$$\begin{cases} R_B R_x = R_x R_A & (a) \\ (R - I) P = R_x P_A - P_B & (b) \end{cases}$$

Une fois la relation estimée, l'estimation de la translation devient un problème trivial : il suffit de résoudre au sens des moindres carrés le système linéaire donné par l'équation (b), quant à la résolution de l'équation (a), elle est loin d'être triviale, mais les propriétés algébriques et géométriques des matrices de rotation ($R^{-1} = R^T$) nous permettent d'aboutir à l'équation suivante :

$$R_B = R_x R_A R_x^T$$

Puisque la matrice R_X est orthogonale (propriété des matrices de rotation), les matrices R_A et R_B ont les mêmes valeurs propres puisqu'elles sont semblables (propriété des matrices).

On sait qu'une matrice de rotation possède (1) comme une de ses valeurs propres; soit n_A (resp n_B) le vecteur propre associé à la valeur propre (1) de R_A (resp R_B). En multipliant à droite l'équation (*) par n_A , on obtient :

$$R_B R_X n_A = R_X R_A n_A = R_X n_A$$

On en déduit que le vecteur $R_X n_A$ est le vecteur propre de R_B associé à la valeur propre 1.

$$n_B = R_X n_A \quad (**)$$

La solution pour la rotation de la caméra est donnée en résolvant le système linéaire (**) puisque le vecteur n_A et n_B ont été extraits de R_A et R_B .

1.12. Notion de base de la modélisation d'une caméra

Quelques notions fondamentales propres aux asservissements par vision sont décrites dans cette section. Elles sont communes à tous les asservissements de l'attitude l'organe terminal du robot manipulateur à partir de mesures fournies par une caméra.

1.12.1. Définition de l'attitude d'un repère [30][27]

En robotique, il est souvent nécessaire de définir la position relative d'un corps par rapport à un autre. Lorsqu'il s'agit d'une position relative suivant 6 degrés de liberté, on parle alors d'attitude (pose en anglais). Une attitude est définie mathématiquement par une translation et une rotation qui permettent de passer d'un repère à un autre. L'attitude peut être également définie de

manière unique par un vecteur de 6 coordonnées, on parle alors de coordonnées opérationnelles. Parmi ces 6 coordonnées, trois définissent la translation et les trois autres définissent la rotation. S'il n'y a pas d'ambiguïté pour la définition de la translation, il n'en est pas de même pour la rotation. Il existe, en effet, plusieurs conventions permettant de décomposer cette rotation en 3 rotations élémentaires.

1.12.2. Définition des repères [29]

On appelle R_c , le repère liée à la caméra. Par convention, R_c est défini comme suit (fig.1.22) :

- Son origine est au centre optique de la caméra.
- L'axe z_c est confondu avec l'axe optique et est dirigé par la scène.
- L'axe x_c est verticale dans l'image et dirigé vers le haut.
- L'axe y_c est horizontal et dirigé tel que R_c soit direct.

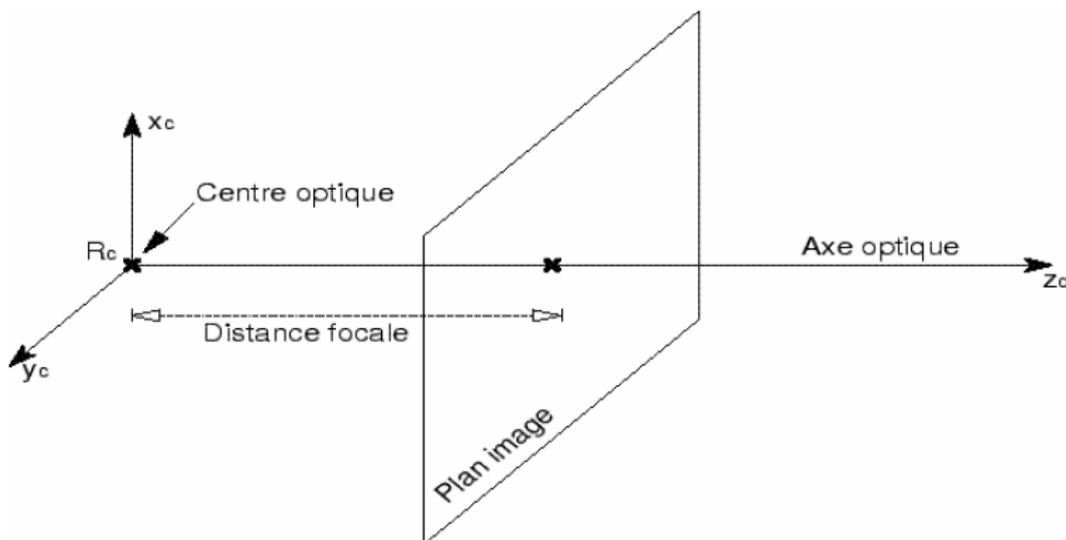


Fig.1.22 Définition d'un repère caméra

On désigne par R_0 un repère lié à un objet situé dans l'espace de travail du robot. La matrice homogène ${}^c T_0$ définit l'attitude de R_0 par rapport à R_c . Il est souvent utile d'utiliser une image virtuelle symétrique à l'image réelle par rapport au centre de la caméra, et ceci afin d'éviter d'inverser l'image en utilisant

une distance focale f négative. Dans un asservissement par vision d'un robot, la grandeur à asservir est souvent l'attitude de R_o (ou d'un autre repère lié à l'objet) par rapport à un autre repère R_t lié à l'organe terminale dans le cas eye to hand, dans ce cas l'estimation des transformations cT_t et cT_o permet de déterminer la position de l'objet par rapport à l'organe terminale (fig.1.23).

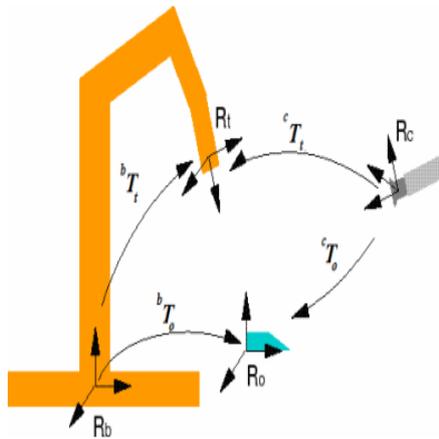


Fig.1.23 Description de la configuration eye to hand

Dans le cas d'une caméra embarquée, la recherche de cT_t constitue le problème de calibrage.

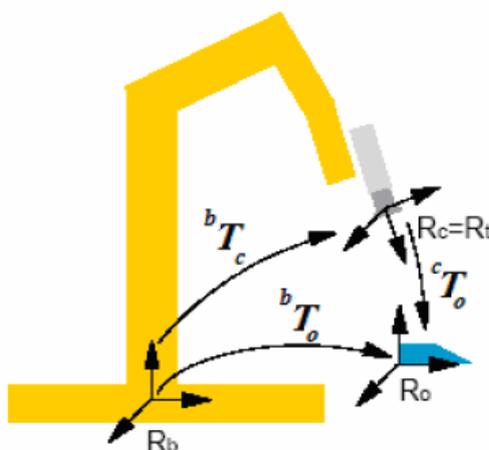


Fig.1.24 Description de la configuration eye in hand

1.12.3. Axe et angle de rotation équivalent à une rotation donnée

On est souvent amené à devoir convertir une matrice homogène en six coordonnées opérationnelles. Soit R une matrice de rotation quelconque dont on voudrait extraire sa représentation minimale θ_u représentant un vecteur de rotation d'angle θ autour d'un axe u .

$$R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}$$

La représentation θ_u s'obtient de manière unique à partir des coefficients r_{ij} ($i=1, 2, 3 ; j=1, 2, 3$) de la matrice de rotation R à l'aide de l'équation suivante :

$$\theta_u = \frac{\theta}{2\sin\theta} \begin{pmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{pmatrix}$$

où :

$$\theta = \arccos \left(\frac{r_{11} + r_{22} + r_{33} + 1}{2} \right)$$

D'après la relation suivante [27]:

$$[u]_x \sin \theta = \frac{R - R^T}{2}$$

avec $[u]_x$ la matrice antisymétrique de produit vectoriel du vecteur u défini par :

$$\mathbf{u} = \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix} \quad \Longrightarrow \quad [\mathbf{u}]_x = \begin{pmatrix} 0 & -u_x & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{pmatrix}$$

Le vecteur \mathbf{u} est alors défini par [32]

$$u_x = \text{sign}(r_{32} - r_{23}) \sqrt{\frac{r_{11} - \cos\theta}{1 - \cos\theta}}$$

$$u_y = \text{sign}(r_{13} - r_{31}) \sqrt{\frac{r_{22} - \cos\theta}{1 - \cos\theta}}$$

$$u_z = \text{sign}(r_{21} - r_{12}) \sqrt{\frac{r_{33} - \cos\theta}{1 - \cos\theta}}$$

1.13. Conclusion

Dans ce chapitre, on a introduit des généralités sur l'asservissement visuel qui permettent à un robot de prendre conscience de son environnement et d'évoluer à travers les informations acquises par le capteur visuel qui est la caméra, présenter les fondements théoriques sur lesquels reposent les asservissements visuels. Comme nous a permis aussi de rappeler les principaux problèmes inhérents à telle ou telle approche. Ces problèmes sont maintenant clairement identifiés et des solutions sont régulièrement proposées en vue de les résoudre.

Dans le chapitre qui suit, on évoquera d'une manière détaillée les différentes techniques d'asservissement visuel.

Deuxième chapitre

Les techniques d'asservissement visuel

2.1 Introduction

L'une des premières problématiques en asservissement visuel, est de sélectionner, de construire et de combiner les informations visuelles qui permettent d'obtenir un comportement le plus proche possible de celui souhaité. Le choix des informations 2D ou 3D impose des lois de commande déférentes. En effet, deux espaces de contrôle sont possibles : soit l'espace cartésien (celui du robot), soit celui du capteur (celui de l'image d'une caméra). Nous parlerons dans le premier cas d'asservissements 3D, et d'asservissements 2D dans le second cas. Quand il s'agit d'employer les deux types d'informations, nous parlerons d'asservissements hybrides etc.

Les techniques d'asservissement visuel consistent à utiliser les informations fournies par un capteur de vision pour contrôler les mouvements d'un système dynamique, ce système pouvant être réel dans le cadre de la robotique, ou bien virtuel dans le cadre de l'animation d'entités artificielles ou de

la réalité augmentée. De manière très classique, on peut classer les différentes techniques d'asservissement visuel en fonction des mesures utilisées en entrée du module de commande. Nous décrivons donc les techniques d'asservissement visuel 3D et 2D, qui ont fait l'objet de travaux depuis de longues années, mais aussi 2D1/2, $d2D/dt$ et 2D3/4, qui sont des approches introduites récemment.

2.2. Asservissement visuel 2D

Comme nous l'avons vu auparavant, il existe différents types d'asservissement visuel. Nous décrivons dans cette partie une méthode d'asservissement visuel, dite 2D, dans laquelle les raisonnements sont réalisés directement au niveau du capteur (la caméra) dont l'objectif est d'atteindre un motif dans l'image et non pas de contrôler une situation entre la caméra et l'objet. Ainsi on supprime la phase de reconstruction d'un modèle 3D et on élimine du même coup les erreurs sur l'estimée de la situation. Notons que grâce à cette approche on peut supprimer les erreurs dues au modèle de la caméra. Seule l'extraction de primitives dans l'image est une source d'erreurs. Les lois de commande consistent alors à contrôler le mouvement de la caméra afin que les mesures dans l'image ($s(t)$) atteigne une valeur désirée s^* , voir suivre une trajectoire spécifier $s^*(t)$. Ce domaine porte tout d'abord sur l'élaboration d'une méthode de calcul explicite de la matrice d'interaction associée aux informations visuelles géométriques. Cette matrice notée J , joue un rôle essentiel dans l'élaboration des diverses lois de commande possible. Elle est également appelée Jacobien de L'image.

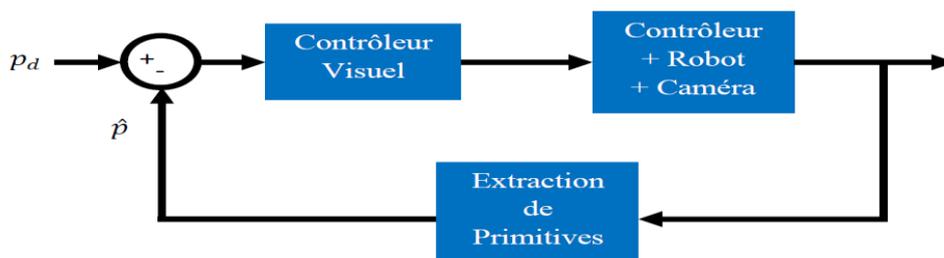


Fig.2.1 Asservissement visuel 2D

2.2.1. Notion de jacobien d'image [33]

Le choix des informations visuelles et l'obtention de la relation les liants aux mouvements du porteur sont deux points fondamentaux de cette approche. Cette relation, obtenue par dérivation de s par rapport à r , est en fait une matrice appelée, suivant les auteurs, matrice d'interaction, jacobien de la tâche ou jacobien de l'image. La notion du jacobien d'image est une entité qui quantifie la relation différentielle entre les variations des informations visuelles fournies par un ou plusieurs capteurs et les mouvements du robot engendrant ces variations. Une information visuelle s , également appelée signal capteur, ne dépend que de la situation entre la caméra et son environnement. Ainsi on considère que les modifications de la nature géométrique sont susceptibles de faire varier la valeur d'une information visuelle. On sait que la caméra est indépendante du robot qui se trouve dans le champ visuel de cette caméra. Les déplacements de la pince du robot ou des objets embarqués sur cette pince sont réalisés à l'aide de différents axes constituant le robot. La situation de la pince ne dépend que de la valeur des coordonnées articulaires q de plus, si la caméra est elle-même mobile, alors s peut s'écrire :

$$s = s(q, t) \quad (2.1)$$

où le paramètre temps ' t ' représente la contribution du mouvement de la caméra.

La différentielle de s permet de relier les variations visuelles dans l'image aux mouvements de la caméra et de la scène. A partir de l'équation (1), on obtient :

$$\dot{s} = \frac{\partial s}{\partial q} \dot{q} + \frac{\partial s}{\partial t} \quad (2.2)$$

Le terme $\frac{\partial s}{\partial q}$ peut se décomposer sous la forme :

$$\frac{\partial s}{\partial q} = \frac{\partial s}{\partial r} \frac{\partial r}{\partial q} \quad (2.3)$$

où l'on reconnaît le classique jacobien du robot $\frac{\partial r}{\partial q}$ si l'on choisit pour r la situation de la pince.

Cette décomposition permet d'exhiber un terme ne dépendant que de la configuration du robot et en aucun cas de la tâche : le jacobien de robot et un terme ne dépendant que de la tâche choisie, représentée par s , et en un cas de la configuration du robot : le terme $\frac{\partial s}{\partial r}$ que l'on peut appeler jacobien de tâche. Il est donc nécessaire de caractériser le jacobien de la tâche pour conserver une approche générale applicable à tous robots notamment quelque soit le nombre de degrés de liberté cette caractérisation permet :

- La réalisation de la commande de robot en boucle fermée à partir des valeurs des informations visuelles (commande référencée capteur).
- L'estimation de la vitesse 3D des objets à partir de la vitesse de leurs primitives 2D fournies par la caméra.
- La quantification de la qualité d'observation des objets mobiles.

Considérant maintenant une composante s_j de s . Nous savons que la différentielle de s_j , est une application linéaire dans \mathbb{R}^3 . Par conséquent, la différentielle de s_j en r n'est autre qu'un torseur cinématique. (Un torseur cinématique d'un repère R_0 par rapport à un repère R_c est défini par la vectrice vitesse de translation de l'origine de R_0 par rapport à R_c ainsi que la vectrice vitesse de rotation de R_0 par rapport à R_c . Classiquement, un torseur T est aussi défini par son vecteur Ω et son champ antisymétrique V . La connaissance du vecteur Ω et du champ v en un point O détermine le champ en tout point M par :

$$V(M) = V(O) + \Omega * \overrightarrow{O(M)}$$

Nous pouvons écrire en point r la relation fondamentale :

$$\dot{s}_j = H_j * T \quad (2.4)$$

Où T est un torseur cinématique représentant la vitesse relative de la scène par rapport à la caméra. $*$ est le produit de torseur et H_j est un torseur dont l'expression dépend à la fois de l'environnement et du capteur lui-même. Il caractérise complètement les interactions entre le capteur et son environnement et nous l'appellerons torseur d'interaction.

La relation (2.4) peut s'écrire sous la forme matricielle suivante :

$$\dot{s}_j = J_j T \quad (2.5)$$

où

$$J_j = H_j \begin{bmatrix} 0 & I_3 \\ I_3 & 0 \end{bmatrix}$$

J_j est la représentation matricielle du torseur d'interaction H_j exprimé dans un repère et en points donnés. La représentation matricielle de l'ensemble $\{H_1, \dots, H_n\}$ associé à $s = (s_1, \dots, s_n)^T$ est appelé jacobien d'image ou une matrice d'interaction et noté J . avec cette notation, nous écrivons :

$$\dot{s} = J T \quad (2.6)$$

2.2.2. Modélisation du jacobien d'image [33]

L'estimation de jacobien peut être réalisée d'une façon analytique si l'on connaît le modèle de l'objet poursuivi [19][34]. Elle peut être également réalisée d'une manière expérimentale, dans ce cas on imprime au robot des mouvements incrémentaux connus autour d'une position nominale et on mesure dans l'image l'effet de ces mouvements sur les signaux 2D. Ensuite la valeur numérique du jacobien à la position nominale sera déterminée en résolvant un système linéaire en les coefficients de ce jacobien. Supposant que la caméra soit immobile et que la pince soit animée d'une vitesse de translation V_0 et d'une vitesse de rotation Ω_0 voire la figure suivante. Le torseur cinématique appliqué à la pince du robot s'écrit donc:

$$T_0 = \begin{bmatrix} V_0 \\ \Omega_0 \end{bmatrix}$$

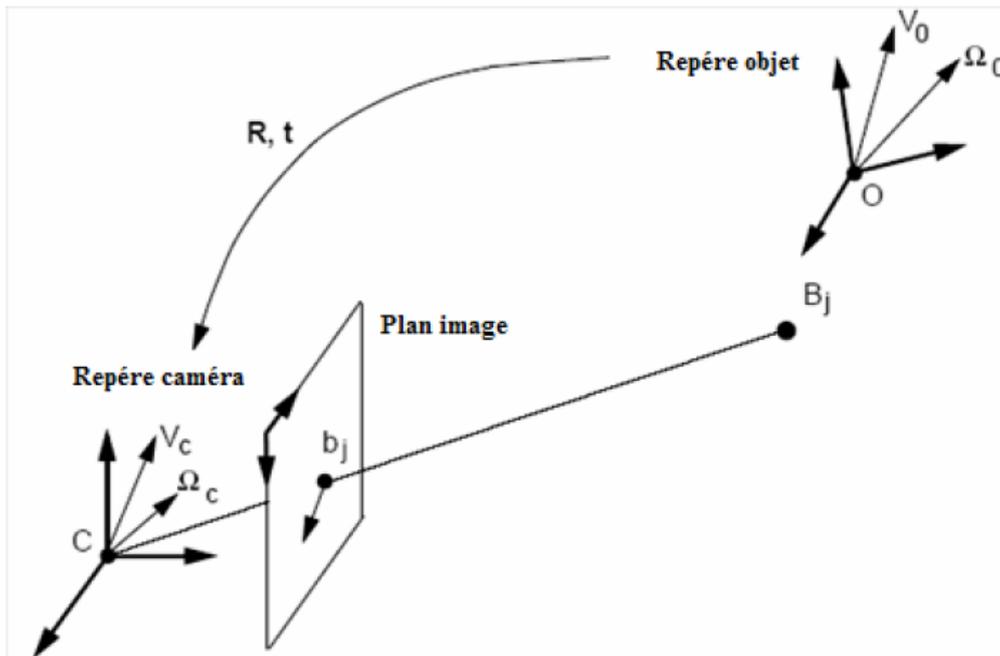


Fig.2.2 La relation entre le torseur cinématique appliquée à la pince du robot et la vitesse 2D de l'image d'un point \$b_j\$

Si l'on note par \$T_c\$ le mouvement de la caméra exprimé dans son repère qui représente le même mouvement que \$T_0\$, on obtient la relation suivante :

$$\begin{bmatrix} V_c \\ \Omega_c \end{bmatrix} = \theta \begin{bmatrix} V_0 \\ \Omega_0 \end{bmatrix} \tag{2.7}$$

Avec

$$\theta = \begin{bmatrix} R & -RS(-R^T \cdot t) \\ 0 & R \end{bmatrix} \tag{2.8}$$

Dans cette équation, R et t représentent la matrice de rotation et de translation rigide entre la caméra et la pince, cette transformation permet le changement de coordonnées du repère pince vers le repère caméra ; $S(a)$ représente la matrice antisymétrique associée au vecteur tridimensionnel a .

Dans ce qui suit nous allons donner la forme explicite de jacobien lorsque la primitive visuelle est un point :

On considère un point de la pince de robot désigné par B_j (voir figure). Les coordonnées de ce point (x_j, y_j, z_j) sont exprimées dans le repère de la caméra. La projection de ce point sur le plan image a pour coordonnées :

$$u_j = \alpha_u \frac{x_j}{z_j} + u_c \quad (2.9)$$

$$v_j = \alpha_v \frac{y_j}{z_j} + v_c \quad (2.10)$$

où α_u , α_v , u_c et v_c sont les paramètres intrinsèques de la caméra associée au modèle sténopé.

En différentiant les deux équations (2.9) et (2.10), on obtient les variations dans l'image des coordonnées u_j et v_j de b_j par rapport à la vitesse de point B_j .

$$\begin{pmatrix} \dot{u}_j \\ \dot{v}_j \end{pmatrix} = \begin{bmatrix} \alpha_u & 0 \\ 0 & \alpha_v \end{bmatrix} \begin{pmatrix} \frac{1}{z_j} & 0 & \frac{-x_j}{z_j^2} \\ 0 & \frac{1}{z_j} & \frac{-y_j}{z_j^2} \end{pmatrix} \begin{pmatrix} \dot{x}_j \\ \dot{y}_j \\ \dot{z}_j \end{pmatrix} \quad (2.11)$$

En appliquant la loi des mouvements rigides, on peut écrire la relation qui donne la vitesse $V_{B_j} = (\dot{x}_j \ \dot{y}_j \ \dot{z}_j)^T$ du point B_j en fonction des éléments de réduction du torseur T_c , soit :

$$V_{B_j} = V_c + \Omega_c * \overrightarrow{CB_j}$$

En remplaçant cette équation dans l'équation (2.11), on obtient :

$$\begin{pmatrix} \dot{u}_j \\ \dot{v}_j \end{pmatrix} = \begin{bmatrix} \alpha_u & 0 \\ 0 & \alpha_v \end{bmatrix} \begin{pmatrix} \frac{1}{z_j} & 0 & \frac{-x_j}{z_j^2} & \frac{-x_j y_j}{z_j^2} & \frac{1+x_j^2}{z_j^2} & \frac{-y_j}{z_j} \\ 0 & \frac{1}{z_j} & \frac{-y_j}{z_j^2} & -\left(\frac{1+y_j^2}{z_j^2}\right) & \frac{x_j y_j}{z_j^2} & \frac{x_j}{z_j} \end{pmatrix} \begin{pmatrix} V_c \\ \Omega_c \end{pmatrix} \quad (2.12)$$

En combinant les deux équations (2.12) et (2.7), nous obtenons :

$$\begin{pmatrix} \dot{u}_j \\ \dot{v}_j \end{pmatrix} = J(B_j) \begin{pmatrix} V_0 \\ \Omega_0 \end{pmatrix} \quad (2.13)$$

Dans cette formule, le jacobien $J(B_j)$, associé au point B_j de la pince est donné par :

$$J(B_j) = \begin{bmatrix} \alpha_u & 0 \\ 0 & \alpha_v \end{bmatrix} \begin{pmatrix} \frac{1}{z_j} & 0 & \frac{-x_j}{z_j^2} & \frac{-x_j y_j}{z_j^2} & \frac{1+x_j^2}{z_j^2} & \frac{-y_j}{z_j} \\ 0 & \frac{1}{z_j} & \frac{-y_j}{z_j^2} & -\left(\frac{1+y_j^2}{z_j^2}\right) & \frac{x_j y_j}{z_j^2} & \frac{x_j}{z_j} \end{pmatrix} \begin{bmatrix} R & -RS(-R^T.t) \\ 0 & R \end{bmatrix} \quad (2.14)$$

Supposons maintenant que l'on veut déterminer le jacobien qui relie le torseur cinématique T_0 et les variations des coordonnées normalisées X_j et Y_j de b_j qui sont donnés par :

$$X_j = \frac{x_j}{z_j} = \frac{u_j - u_c}{\alpha_u} \quad (2.15)$$

$$Y_j = \frac{y_j}{z_j} = \frac{v_j - v_c}{\alpha_u} \quad (2.16)$$

En différentiant ces deux équations, nous obtenons :

$$\begin{pmatrix} \dot{X}_j \\ \dot{Y}_j \end{pmatrix} = \begin{bmatrix} \frac{1}{\alpha_u} & 0 \\ 0 & \frac{1}{\alpha_v} \end{bmatrix} \begin{pmatrix} \dot{u}_j \\ \dot{v}_j \end{pmatrix} \quad (2.17)$$

En utilisant l'équation (2.13), on peut écrire :

$$\begin{pmatrix} \dot{X}_j \\ \dot{Y}_j \end{pmatrix} = \begin{bmatrix} \frac{1}{\alpha_u} & 0 \\ 0 & \frac{1}{\alpha_v} \end{bmatrix} J(B_j) \begin{pmatrix} V_0 \\ \Omega_0 \end{pmatrix}$$

On en déduit l'expression du Jacobien associé aux coordonnées normalisées du point b_j :

$$J_N(B_j) = \begin{pmatrix} \frac{1}{z_j} & 0 & \frac{-x_j}{z_j^2} & \frac{-x_j y_j}{z_j^2} & \frac{1+x_j^2}{z_j^2} & \frac{-y_j}{z_j} \\ 0 & \frac{1}{z_j} & \frac{-y_j}{z_j^2} & -\left(\frac{1+y_j^2}{z_j^2}\right) & \frac{x_j y_j}{z_j^2} & \frac{x_j}{z_j} \end{pmatrix} \begin{bmatrix} R & -RS(-R^T \cdot t) \\ 0 & R \end{bmatrix}$$

En utilisant les deux équations (2.15) et (2.16), on peut réécrire le jacobien $J_N(B_j)$ sous la forme suivante :

$$J_N(B_j) = \begin{pmatrix} \frac{1}{z_j} & 0 & \frac{-X_j}{z_j} & -X_j Y_j & 1 + X_j^2 & -Y_j \\ 0 & \frac{1}{z_j} & \frac{-Y_j}{z_j} & -(1 + Y_j^2) & X_j Y_j & X_j \end{pmatrix} \begin{bmatrix} R & -RS(-R^T \cdot t) \\ 0 & R \end{bmatrix} \quad (2.18)$$

D'où la matrice d'interaction est :

$$J = \begin{pmatrix} \frac{-1}{z} & 0 & \frac{x}{z} & xy & -(1+x^2) & y \\ 0 & \frac{-1}{z} & \frac{y}{z} & 1+y^2 & -xy & -x \end{pmatrix} \quad (2.19)$$

Une question est de savoir comment mettre à jour cette matrice, notée J , dans la boucle d'asservissement. Deux cas de figure sont possibles: J est calculée hors-ligne et J est calculée en-ligne.

- **Calcul hors-ligne**

Dans ce cas, elle n'est calculée qu'une seule fois avant l'asservissement, en prenant comme indices visuels au point d'équilibre souhaité, c'est à dire les consignes s^* . Cette approche allège les calculs dans la boucle d'asservissement, tout en préservant sa convergence [34].

- **Calcul en-ligne**

Nous considérons dans ce cas qu'elle est calculée à chaque itération en fonction des indices visuels. Le torseur dépendant également de l'information 3D (la profondeur z) [35]. Une estimation de z peut se faire si on dispose d'un modèle 3D de l'objet [34].

Ces deux possibilités présentent chacune des avantages, mais aussi des inconvénients. D'une part, le calcul de J en-ligne contraint fortement la trajectoire que doivent suivre dans l'image les informations visuelles pour atteindre leur but. Par exemple, si l'on choisit des points, ce choix implique que ceux-ci doivent aller simultanément en ligne droite vers leur position désirée (fig.2.3(a)). Cependant, ce comportement souhaitable dans l'image peut parfois entraîner des mouvements de la caméra inadéquats (correspondant à une singularité), voire même impossible à réaliser (fig.2.3(b)). La caméra peut ainsi atteindre un minimum local de l'erreur dans l'image. D'une autre part, le deuxième choix ne contraint quasiment pas la trajectoire dans l'image. Du coup, et même si nous ne pouvons pas l'expliquer, il s'avère en pratique que la commande est alors beaucoup moins sujette aux problèmes des mouvements irréalisables et des minima locaux. Par contre, il est tout à fait possible que l'objet sorte du champ de vue de la caméra au cours du positionnement.

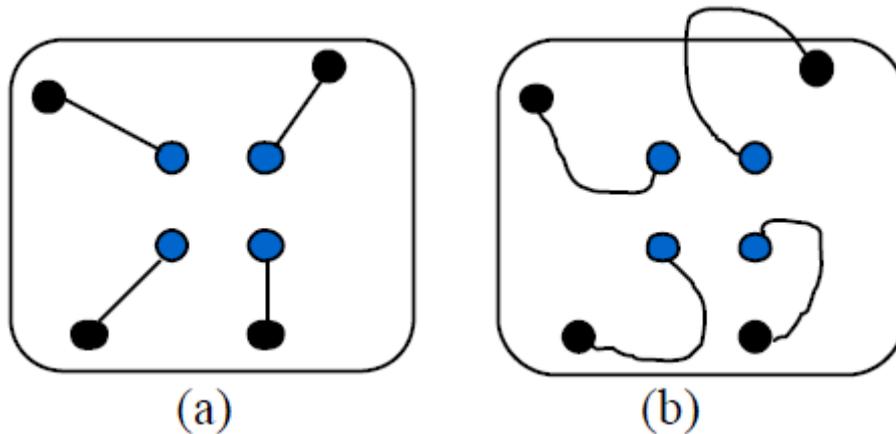


Fig.2.3 trajectoire des points de l'objet dans l'image

2.2.3. Choix de la matrice C

Dans l'asservissement visuel 3D la matrice C est imposée par le choix de la fonction de tâche, mais dans l'asservissement visuel 2D c'est le choix de la matrice C qui détermine la fonction de tâche :

✓ Une solution simple largement adaptée est de choisir $C=I_m$ avec $m=2p$. Afin d'avoir une fonction de tâche admissible, il faudrait choisir $m = 6$ (donc 3 points), mais ce faisant, comme nous l'avons déjà dit, il est possible de rencontrer une singularité de la matrice d'interaction ou d'atteindre un minimum global non désiré de la norme de la fonction de tâche. Si $m > 6$ (donc pour $p > 3$ points), la dimension de vecteur de sortie ou de tâche est supérieur à celle du vecteur de commande (la vectrice vitesse de la caméra est de dimension 6) et on sort du cadre défini par l'admissibilité de la fonction de tâche. En effet, le nombre d'informations visuelles que l'on désire contrôler est supérieur au degré de liberté disponibles et il existe alors des trajectoires irréalisables, c'est-à-dire pour lesquelles il n'existe pas de mouvements de la caméra correspondant. Si une telle trajectoire est nécessaire pour atteindre la convergence, le robot ne pourra alors pas la réaliser et le système atteindra un minimum local de la norme de la fonction de tâche.

✓ Une solution qui permet de tenir compte de la redondance d'informations est de choisir $C = J^+$ c'est-à-dire la pseudo-inverse de la matrice d'interaction.

✓ Une autre solution qui permet de tenir compte de la redondance d'informations est de choisir $C = J^T$. Ce choix ne nécessite pas l'inversion de la matrice d'interaction.

2.2.4. Les lois de commandes

D'une manière générale, l'approche fonction de tâche introduite en commande référencée capteur s'applique parfaitement au cas du capteur vision comme l'a montré Chaumette dans [24]. 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 la 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 ou permet de satisfaire d'autres critères (intégrant bien évidemment la dynamique du porteur[32].

- Dans le cas contraire, une commande cinématique est élaborée. On désire, par exemple que la fonction de tâche se comporte comme un système découplé de premier ordre à savoir $\dot{e} = -\lambda e$ (avec λ est un gain positive et $e = s^* - s$). cela signifie que l'on souhaite que le système converge de manière exponentielle vers son état final. C'est ce qu'on appelle la commande par fonction de tâche. La plus grande part des travaux en asservissement visuel 2D utilise une telle loi [23][28]. Cependant, des variantes de cette approche où, soient des informations visuelles dynamiques peuvent être appréhendées, soit fonctions visuelles fonction de temps. D'un autre point de vue, certains auteurs abordent l'élaboration de loi de commande sous le regard de l'Automatique: Une commande anticipative avec adaptation paramétrique du modèle de système à commander ; une commande de type PI, ou par placement de pôles ; plus généralement, des commandes optimales basées sur la minimisation d'un critère temps-énergie peuvent être mise en œuvre. 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. Toutes fois, certains auteurs mettent en œuvre des observateurs, généralement basés sur le filtre de Kalman, pour l'estimer et ainsi le prendre en compte dans la loi de commande.

2.3. Asservissement visuel 3D

La première approche proposée par Sanderson et Weiss en 1980 est basée sur le contrôle d'un robot dans son espace cartésien. Dans un asservissement 3D, la référence est exprimée sous la forme d'une pose à atteindre. Cette pose définit la position et l'orientation d'un repère lié à l'objet par rapport à un repère lié à la caméra. La référence peut être également exprimée sous forme de primitives géométriques 3D [34]. Le comportement dans le domaine 3D d'asservissement en cas de convergence du système est généralement satisfaisant du point de vue de la trajectoire de la caméra. Cependant, le calcul des informations 3D à utiliser dans la boucle de commande (pose ou primitives 3D) nécessite la connaissance du modèle 3D de l'objet (fig.2.4).

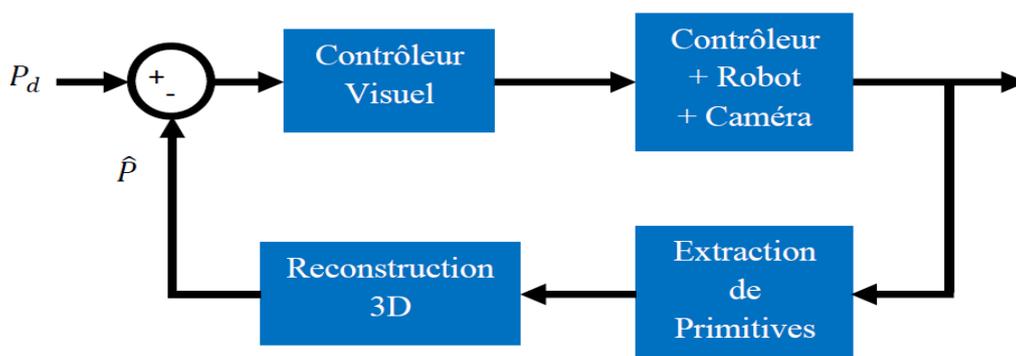


Fig.2 4 Asservissement visuel 3D

Parmi les avantages liés à l'utilisation d'une telle approche, on peut citer celui de l'espace de contrôle qui est équivalent à l'espace cartésien du robot, et qui permet de simplifier la définition de la tâche robotique. Parmi les

inconvenients les plus cités, on peut noter le problème lié à la modélisation de l'objet, celui lié à la reconstruction et calibrage de la caméra (paramètres intrinsèques et extrinsèques), le temps nécessaire à l'obtention de la mesure de la pose, le délai introduit par ce même calcul, les cas singuliers liés à l'algorithme de reconstruction ou à la représentation des primitives dans l'image, l'influence de la précision de détection des primitives sur le calcul de la pose, pour ne citer les plus importants et les plus discutés.

2.3.2. L'estimation de la pose

La reconstruction 3D, c'est-à-dire une estimation, à partir des primitives visuelles de la scène, de la position relative entre la caméra et un ou plusieurs objets de la scène. Pour ce faire, utiliser une seule caméra ne suffit pas. Deux stratégies peuvent être envisagées. Soit on utilise plusieurs caméras, et la mise en correspondance des images vues par chaque caméra permet d'estimer la pose de la caméra par rapport à l'objet observé. Soit on ne considère qu'une seule caméra, mais alors les informations de l'image sont insuffisante pour définir complètement la pose, et il est nécessaire de tenir compte d'information supplémentaire (s) (par exemple des données d'autres capteurs ou une connaissance a priori d'un modèle (3D ou 2D) de l'objet et les paramètres intrinsèques de la caméra). Il existe une vaste littérature traitant des méthodes de reconstruction 3D [page26bourgadez] qui nous ne détaillons pas ici. On citer comme exemple celle calculer à partir de données de vision. L'estimation de la pose d'objet par vision monoculaire a retenu l'attention de plusieurs chercheurs. En effet depuis les années 80, différentes méthodes de reconstruction ont vu le jour basées sur l'utilisation de primitives visuelles géométriques (point , droite, segment, contour elliptique, ...) ou dynamiques (flot optique). Plus récemment, c'est l'utilisation de primitives de type texture qui reliait l'attention de certains chercheurs. Parmi les travaux effectués sur la reconstruction par vision monoculaire, on peut distinguer deux catégories :

- ✓ La première s'appuie sur l'utilisation de méthodes itératives de minimisation ou d'optimisation de fonction de coût (la pose).

- ✓ La seconde catégorie s'attache à établir des solutions analytiques au problème posé.

2.3.3. Comportement de la loi de commande [35][36]

Une fois les informations visuelles 3D sont obtenues grâce à un calcul de pose, il est alors possible de décrire la tâche à réaliser en terme d'une fonction de tâche $e(s(q), s^*)$ à réguler, fonction des informations visuelles courantes s et des informations visuelles désirées s^* intervenant dans le processus de calcul de pose. Il devient également possible de calculer la matrice d'interaction L_e . La matrice d'interaction 3D ($L_e = \partial e / \partial r$) qui ne dépend que de la configuration de la caméra par rapport à l'objet et des informations visuelles choisies. Comme cette matrice dépend des informations visuelles choisies, on peut la réécrire en fonction de s . Elle vient alors :

$$L_e = \frac{\partial e}{\partial s} \frac{\partial s}{\partial r} \quad (2.19)$$

Cependant, dans le cas général où des erreurs de modélisation ou de calibration existent, il est extrêmement difficile de prouver la convergence de la loi de commande. Pour s'en persuader revenons à l'expression de L_e donnée. En effet, très souvent le terme $\partial e / \partial s$ peut ne pas être obtenu sous forme analytique, il s'obtient sous forme numérique par l'algorithme de calcul de pose et ne permet pas donc d'exhiber des conditions particulières de convergence. Par ailleurs, la matrice $\partial e / \partial s$ peut dans certains cas être très mal conditionnée. Citons à titre d'exemple le cas où s est constituée de quatre points coplanaires, bien que, théoriquement, il n'existe qu'une seule valeur de pose conduisant aux valeurs courantes s , différentes valeurs de pose permettant d'obtenir des valeurs quasiment identiques pour s . De faibles différences sur les valeurs de s conduisent dans ce cas à de grandes variations de r . Néanmoins, des travaux ont pu garantir la stabilité de la loi de commande en présence d'erreurs de calibration de la caméra [37]. Ces travaux garantissent également

que l'objet reste entièrement visible durant la réalisation de la tâche. Si ces erreurs de calibration peuvent intervenir dans le calcul de L_e , elles n'influencent pas en revanche sur la précision finale de r lorsque r^* est obtenu par apprentissage. En effet, même si les estimations $\hat{s}, \hat{s}^*, \hat{r}, \hat{r}^*$ sont dans ce cas biaisées, la loi de commande amènera \hat{r} en \hat{r}^* correspondant bien à r^* .

2.3.4. Intérêt et inconvénients

Le choix d'informations visuelles s représentant la pose permet d'obtenir des matrices d'interactions creuses. Avec la relation $L_{\theta u}$ donnée dans [27]

$$\mathbf{L}_{\theta u} = \mathbf{I}_3 - \frac{\theta}{2}[\mathbf{u}]_{\times} + \left(1 - \frac{\text{sinc}(\theta)}{\text{sinc}^2(\frac{\theta}{2})}\right)[\mathbf{u}]_{\times}^2.$$

On a donc un découplage entre les translations et les rotations. Dans ce cas, on peut séparer le problème en deux et s'intéresser aux translations indépendamment des rotations. De plus, une loi de commande assurant une bonne trajectoire des composantes de s d'assurer également une bonne trajectoire de la caméra dans l'espace 3D, puisque s représente directement la pose. Estimer la pose à partir des données de vision permet aussi d'intégrer simplement ces mesures dans des schémas de commande existant, sans concevoir de loi de commande spécifique au cas de l'utilisation de la caméra. Cependant, cette technique présente deux inconvénients majeurs. D'une part, la phase préalable d'estimation de pose n'est pas triviale, elle nécessite une très bonne calibration des caméras, et pose des problèmes de mise en correspondance, de précision, ...etc. d'autre part, selon le choix que l'on a fait pour s , on peut soit assurer que la caméra se déplace en ligne droite dans l'espace 3D, mais alors rien n'assure que la cible reste dans le champ de vision, soit assurer que l'origine de la cible observée décrit une ligne droite dans le plan image, mais alors la trajectoire 3D de la caméra n'est plus une ligne droite. Ainsi, selon l'application que l'on souhaite réaliser, on préférera l'une ou l'autre approche. Un autre inconvénient de ce type est la nécessité d'une étape de reconstruction 3D permettant de fournir une mesure d'altitude, cette dernière peut par exemple être obtenue par triangulation en utilisant un système stéréoscopique calibré, ou par des techniques de reconstruction de pose en cas

d'utilisation d'un capteur monoculaire. Il résulte après convergence de l'asservissement lorsque le régime permanent est atteint, un biais entre la projection du robot et l'objet d'intérêt à atteindre, ceci constitue le problème majeur de l'asservissement visuel 3D.

2.4. Comparaison entre l'asservissement visuel 2D et 3D [27]

Les systèmes d'asservissement visuel ont été étudiés depuis une vingtaine d'années et sont désormais en phase de validation afin de les rendre enfin opérationnels sur des sites industriels. Ces systèmes présentent des caractéristiques différentes et le choix d'une technique plutôt qu'une autre n'est pas toujours facile. Il est donc intéressant de pouvoir effectuer une « comparaison quantitative » entre les différentes techniques. Ceci est possible seulement entre des caractéristiques communes mesurables, comme par exemple la sensibilité au bruit de mesure. Dans certains cas, on ne peut qu'effectuer une « comparaison qualitative » comme dans le cas du comportement dynamique de l'asservissement. Nous définissons cinq critères de comparaison et nous les appliquons aux techniques 2D et 3D :

- **Information** nécessaire pour la modélisation. On peut distinguer l'information mesurée dans l'image de celle a priori nécessaire pour la mise en œuvre de l'asservissement (modèle de la cible, profondeur des points). Il est évident que le nombre d'informations a priori nécessaire est directement lié à la portabilité du schéma d'asservissement. Vu la nature différente de ces informations, on ne peut qu'effectuer une comparaison qualitative. En ce qui concerne les informations mesurées, les deux techniques d'asservissement visuel 2D et 3D nécessitent au moins quatre points mis en correspondance dans les deux images. L'asservissement visuel 3D nécessite la connaissance de la structure de l'objet observé tandis que, dans l'asservissement visuel 2D, on peut suppléer la connaissance de la profondeur par une approximation ou une estimation en ligne.

- **Performance :** Nous concédèrons comme principal indice de la performance la cadence. Afin de faire abstraction de l'architecture du système de vision, de l'architecture du contrôleur du robot et du robot lui-même, seul le délai introduit par le contrôleur visuel doit être mesuré. Il est clair que d'autres indices de performance peuvent être utilisés tel que le rapport entre le temps de convergence et l'énergie dépensée. Le problème du retard introduit dans l'asservissement visuel. Toutefois, un étage d'estimation supplémentaire est introduit dans l'asservissement visuel 3D, ou dans le cas de l'estimation en ligne de la profondeur pour l'asservissement visuel 2D.

- **Sensibilité** au bruit de mesure. Une comparaison quantitative est possible si toutes les techniques utilisent les mêmes informations visuelles. Normalement, le bruit de mesure devrait avoir une influence seulement sur le comportement du système à la convergence. Toutes fois, quand des techniques de reconstruction sont utilisées, il peut avoir également une influence sur la stabilité du système. Les systèmes contrôlant directement dans l'image sont donc théoriquement moins sensibles aux erreurs de mesure. Une technique d'estimation basée sur la minimisation de critères non linéaire peut diminuer la sensibilité de l'asservissement visuel 3D, mais elle diminue également sa performance.

- **Comportement** dynamique : La trajectoire de l'effecteur dans l'espace et la trajectoire des informations visuelles dans l'image peuvent être comparées qualitativement. Une mesure quantitative peut être établie sur la probabilité d'assurer la visibilité de la tâche tout au long de l'asservissement. Dans le cas de l'asservissement visuel 2D, les trajectoires de l'effecteur du robot peuvent être assez surprenantes car son mouvement est contrôlé directement dans l'image. De plus, la présence d'éventuelles singularités de la matrice d'interaction peut poser des problèmes d'admissibilité de la tâche (minima locaux). Par opposition, dans le cas de l'asservissement visuel 3D, les trajectoires des informations visuelles dans l'image peuvent être assez surprenantes car le robot est contrôlé directement dans l'espace cartésien. Dans le premier cas, le problème de la visibilité de la tâche est moins délicat car on a un contrôle

`direct` des informations visuelles dans l'image. Dans le deuxième cas, même en absence d'erreur de calibration, il est difficile d'assurer que l'objet ne sorte pas du champ de vu de la caméra.

- **Robustesse** aux erreurs de calibration: Une comparaison quantitative peut s'effectuer en ce qui concerne les erreurs les paramètres intrinsèques de la caméra et sur les translations caméra / effecteur . Les erreurs dues aux caractéristiques géométriques propres à chaque technique d'asservissement peuvent être comparées qualitativement. Mais à part les cas très simples, il est très difficile de produire des résultats formels sur la robustesse de ces systèmes vis-à-vis des erreurs de calibration. Ceci est dû dans l'asservissement visuel 2D au couplage entre la commande en translation et en rotation, et à l'intervention des erreurs de calibration de manière non triviale dans la matrice d'interaction, ce qui rend l'étude analytique de la boucle fermée très compliquée même dans le cas simple où seulement trois points sont utilisés. Dans le cas de l'asservissement visuel 3D, même s'il présente un découplage de la rotation par rapport à la translation, la difficulté de la démonstration est due d'une part à l'absence de contrôle des informations visuelles observées (problème de visibilité de la tâche), et d'autre part, même si un contrôle indirect dans l'image est introduit, à l'impossibilité de modéliser la matrice d'erreur entre la fonction de tâche mesurée et la fonction de tâche réelle.

Chaque critère de comparaison a un point différent selon la tâche, le matériel disponible, etc. Sur la base de ces observations, un schéma d'asservissement visuel qui puisse réunir les points positifs de l'asservissement visuel 2D et 3D devrait avoir les caractéristiques suivantes :

- ✓ Permettre un contrôle direct dans l'image afin d'assurer la visibilité de la tâche.
- ✓ Permettre le découplage de la loi de commande en rotation de celle en translation.
- ✓ Utiliser un nombre minimum d'informations a priori sur la structure de l'objet observé.

2.5. Asservissement visuel 2D^{1/2}

L'étude, menée dans [38] a porté sur l'élaboration d'une nouvelle technique d'asservissement visuel conservant les avantages respectifs des approches classiques 2D et 3D, et palliant leurs inconvénients. Pour ce faire, nous utilisons dans la fonction de tâche à réguler des mesures 2D provenant directement de l'image et des mesures 3D obtenues par estimation de l'homographie qui relie l'image de points coplanaires. La combinaison de ces informations permet d'aboutir à des solutions aux propriétés particulièrement intéressantes, évitant notamment les problèmes parfois rencontrés en asservissement visuel 2D pour des positions initiales de la caméra très éloignées de sa position désirée (singularités de la matrice d'interaction, problèmes de minima locaux ou de sortie du champ de vue de la caméra). On obtient en effet un fort découplage de la loi de commande entre la boucle d'asservissement en translation et la boucle d'asservissement en rotation, un contrôle partiel dans l'espace cartésien et dans l'image, et une étude assez aisée de la stabilité et du domaine de convergence de la loi de commande. De plus, une loi de commande adaptative, réglant de manière automatique les gains de l'asservissement, a été mise en œuvre. Cette loi de commande permet de conserver en permanence l'objet dans le champ de vision de la caméra. Un autre avantage majeur des techniques d'asservissement visuel 2D^{1/2} par rapport à l'ensemble de celles développées auparavant est que la connaissance d'un modèle géométrique 3D de l'objet n'est plus nécessaire. Cela lui ouvre un champ d'application très important à partir du moment où le motif à atteindre est préalablement déterminé, lors d'une phase d'apprentissage par exemple.

Dans un asservissement visuel hybride, le vecteur de mesure s est composé à la fois d'informations tridimensionnelles et de primitives extraites directement de l'image. L'asservissement visuel hybride peut alors être vu comme un compromis entre les asservissements visuels 2D et 3D. En effet, il permet de contrôler à la fois la trajectoire de la caméra dans son espace de travail et la trajectoire de certaines primitives dans l'espace image. Le premier schéma

d'asservissement visuel hybride a été proposé par Malis et al dans [39]. Il utilise une reconstruction partielle de la pose sans connaissance a priori du modèle 3D de l'objet par l'intermédiaire de l'estimation de la matrice d'homographie relative à un plan de référence. La fonction de tâche, dans ce cas, est composée de deux parties. La première partie contient les coordonnées images d'un point et une grandeur associée au ratio des profondeurs de celui-ci pour les positions courante et désirée de la caméra. La seconde composante représente la rotation (axe / angle) entre les positions courante et désirée de la caméra. La matrice d'interaction associée à cette fonction de tâche a une forme simple et est partiellement découplée, ce qui permet d'aboutir à des résultats formels sur la stabilité en présence d'erreurs de modélisation. Une approche similaire à celle décrite dans [39]. Le choix des primitives géométriques 2D utilisées dans la définition de la fonction de tâche assure dans ce cas la visibilité de l'objet. Cependant, la forme analytique de l'interaction associée n'a plus les propriétés intéressantes obtenues dans [39] et l'analyse de stabilité est alors beaucoup plus difficile. Citons finalement les travaux de Cervera et al. Ils proposent de combiner les coordonnées dans l'image de points à leurs profondeurs. Les trajectoires 3D de la caméra sont dans ce cas satisfaisante et comparables à celles obtenues avec un asservissement visuel 3D. Cette approche est basée sur l'estimation de l'homographie qui relie l'image de trois points. L'homographie permet de connaître des mouvements locaux dans une image, en utilisant uniquement deux images consécutives.

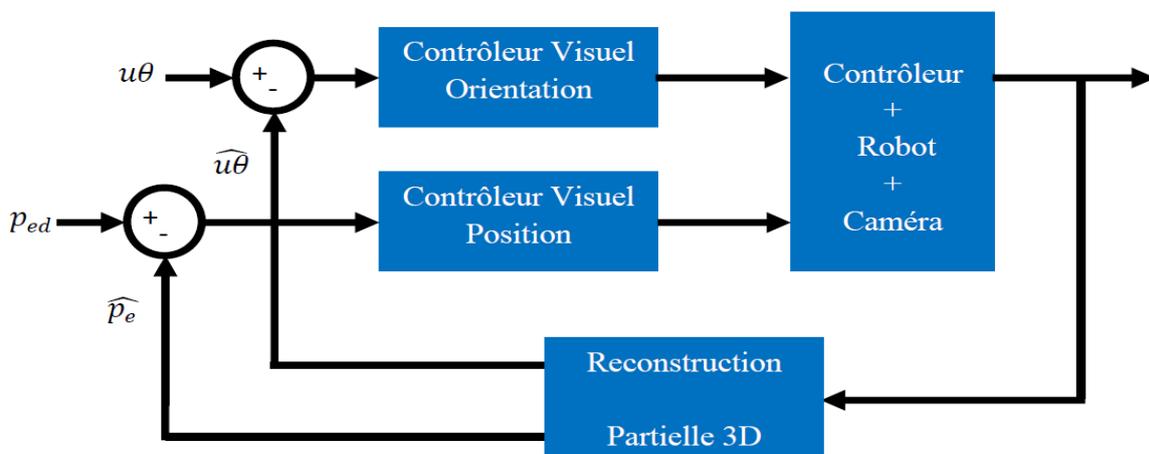


Fig.2.5 Asservissement visuel 2D1/2

L'asservissement visuel 2D1/2 utilise une combinaison d'informations exprimées pour certaines d'entre elles dans l'image et pour d'autre dans le repère caméra ; avec cette technique, il est possible de contraindre la trajectoire d'un point de la cible à se projeter suivant une droite dans l'image. L'objectif est un meilleur contrôle de la position de la cible dans l'image lors de régime transitoire pour des écarts important entre la position de référence (éviter entre autre qu'un des points ne sorte de l'image).

- **Adaptation des gains**

Ce type de commande a pour objectif d'améliorer la robustesse de la stabilité aux erreurs de calibration. L'adaptation est basée sur l'analyse du système en boucle fermée et n'apporte pas de connaissances supplémentaires sur les paramètres géométrique du système qui, eux constants.

- **Adaptation des paramètres géométriques du système**

Ce type de commande a pour objectif d'améliorer non seulement la stabilité, mais aussi le comportement dynamique (et donc les performances) du système, par la mise à jour à chaque itération des paramètres géométriques du système.

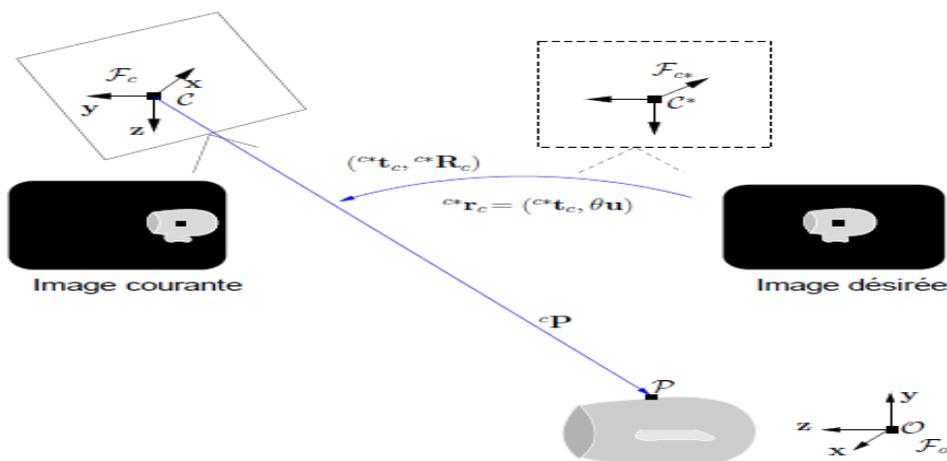


Fig.2.9 Contrôle dans les espaces 2D et 3D : coordonnées de l'image d'un point de l'objet et orientation de la caméra

Dans ce schéma, l'orientation relative cR_c est obtenue après estimation de la pose partielle de la caméra en utilisant les images courantes et désirée (voir figure). Contrairement à l'asservissement visuel 3D, cette estimation ne nécessite pas de connaître le modèle géométrique de l'objet.

Cependant, comme l'asservissement visuel 3D, ce schéma reste sensible aux bruits dans l'image à cause de cette estimation. La matrice d'interaction associée à s , présentant le découplage entre le contrôle de l'orientation et les vitesses de translation de la caméra, est dans [39] :

$$L_s = \begin{bmatrix} L_v & L_{v,w} \\ 0 & L_w \end{bmatrix}$$

avec :

$$L_v = \frac{1}{p_z p_z^*} \begin{pmatrix} -1 & 0 & p_x \\ 0 & -1 & p_y \\ 0 & 0 & -1 \end{pmatrix}$$

et

$$L_{v,w} = \begin{pmatrix} p_x p_y & -(1 + p_x^2) & p_y \\ 1 + p_y^2 & -p_x p_y & -p_x \\ -p_y & p_x & 0 \end{pmatrix}$$

Cette solution a été exploitée dans l'asservissement visuel 2D1/2 où une commande utilisant une matrice d'interaction découplée, ne présentant aucune singularité dans tout l'espace de travail et tolérant un étalonnage grossier du système de vision, a été proposée par Malis en 2002, cette approche a été appliquée sur des systèmes de vision centrale catadioptrique par Hadj-Abdelkader en 2005 et 2007.[41]

2.5.2. Modélisation d'un asservissement visuel 2D1/2

La commande 2D1/2 utilise deux contrôles, celui de la rotation et celui de la translation.

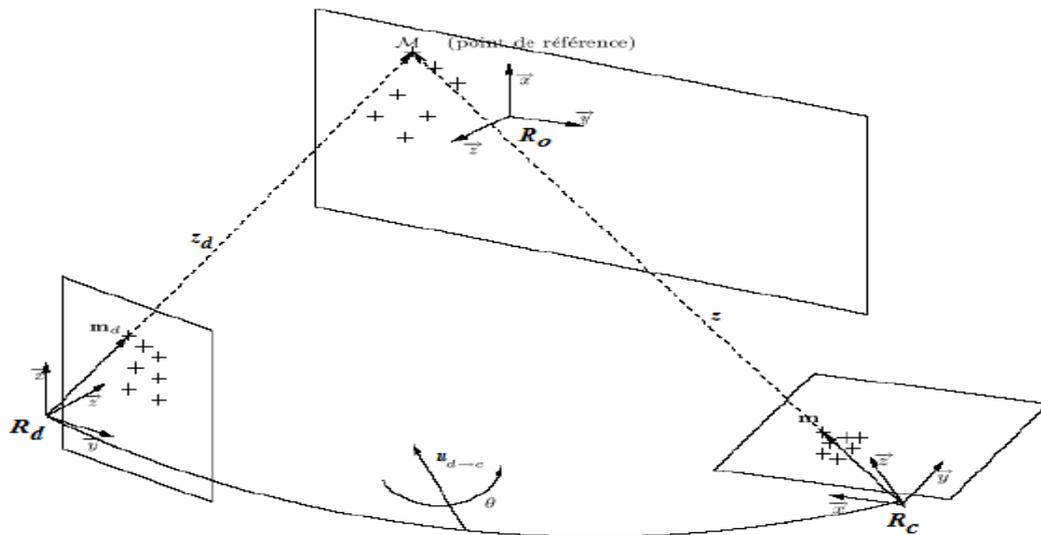


Fig.2.10 Modélisation d'un asservissement visuel 2D1/2

Afin de contrôler l'orientation de la caméra, on utilise la rotation dR_c entre le repère désiré R_d et le repère courant R_c (dR_c doit se ramener à une identité) pour cela, on choisit la représentation vectorielle θ_u de la rotation dR_c . La position de la caméra peut être contrôlée à partir d'information dans l'image et dans le repère cartésien. Pour garder la cible dans le champ de vision de caméra, on utilise trois informations indépendantes, à savoir les deux coordonnées visuelles d'un point de la cible dans l'image, et le rapport $\left(\frac{z}{z_d}\right)$; avec z et z_d la distance courante et désirée de ce point de la cible par rapport à la caméra.

- La fonction de tâche d'un asservissement visuel 2D1/2 est définie par :

$$e = \left[x - x^* \quad y - y^* \quad \ln \left(\frac{z}{z_d} \right) \quad \theta_u \right]^T$$

où $(x, y)^T$ et $(x^*, y^*)^T$ sont respectivement les coordonnées métriques courantes et désirées d'un point caractéristique dans l'image, $\left(\frac{z}{z_d}\right)$ est le rapport entre les distances courantes et désirée de ce point par rapport à la caméra, θ_u la représentation minimale de la rotation à réaliser. Dans l'asservissement visuel 2D1/2, on peut séparer les boucles d'asservissement en rotation et en translation de la caméra, ce qui permet :

- ✓ Un fort découplage de la loi de commande ;
- ✓ Un contrôle partiel dans l'image permettant de conserver l'objet en permanence dans le champ de vision de la caméra ;
- ✓ Une étude de la stabilité et du domaine de convergence de la loi de commande. Ce qui a permis d'obtenir des conditions suffisantes de stabilité prenant en compte les erreurs de calibration et les erreurs d'estimation de l'homographie.

L'avantage majeur de cette approche est que la connaissance d'un modèle géométrique 3D de l'objet n'est plus nécessaire, ce qui lui donne un domaine d'application très important à partir du moment où le motif à atteindre est préalablement déterminé, par exemple lors d'une phase d'apprentissage. La seule information 3D utilisée dans la commande est la profondeur désirée approximative d'un point de l'objet, soit un besoin d'information a priori bien moindre qu'en asservissement visuel 3D et 2D. L'utilisation de droites comme information visuelles pour estimer l'homographie est également possible dans le cas d'un objet planaire. Par contre, l'utilisation de primitives plus complexe (telles des ellipses par exemple) semble plus délicat.

2.6. Asservissement visuel 2D3/4 [27]

On a vu précédemment qu'il est possible de contrôler le mouvement du robot à la fois dans l'espace cartésien et à la fois dans l'image. Nous allons voir à présent qu'il est également possible de le contrôler dans un espace qui n'est plus image (car on utilise directement les coordonnées images des points), mais qui n'est pas non plus purement l'espace cartésien car la reconstruction

euclidienne n'est obtenue qu'à un facteur d'échelle près. On rappelle en effet qu'à partir de la matrice essentielle ou de celle d'homographie, on peut estimer la rotation de la caméra et sa translation. Toutefois, exactement comme dans le cas de l'asservissement visuel 3D, si on contrôle la translation de la caméra entre la position courante et désirée, on n'assure aucun contrôle dans l'image, on a donc préféré deux autres solutions :

✓ Si le contrôle direct de la translation de la caméra entre la position courante et désirée est nécessaire dans la tâche qu'on doit réaliser (parce que, par exemple, on souhaite aller en ligne droite d'un point à un autre), on contrôle deux degrés de liberté en rotation en utilisant un point de référence afin de garder dans l'image, le troisième degré de rotation, à savoir la rotation autour de l'axe optique, est estimé directement.

✓ Si le contrôle de la trajectoire de la caméra n'est pas indispensable, nous proposons une nouvelle méthode qui contrôle un vecteur proportionnel aux coordonnées cartésiennes d'un point de référence de la cible. Ce schéma de contrôle, intermédiaire entre l'asservissement visuel 2D1/2 et l'asservissement visuel 3D, est appelé asservissement visuel 2D3/4 [27].

2.7. Comparaison entre AV2D1/2 et AV2D3/4

Dans le cas d'AV2D3/4, on impose une convergence en ligne droite d'un point dans le repère de la caméra, mais aussi dans sa projection correspondante dans l'image. Le comportement dans l'image est le même que celui de AV2D1/2. Par conséquent, on peut attendre théoriquement à des performances très proches des deux méthodes d'asservissement. Toutefois, il est souligné que l'estimation de l'état est plus sensible au bruit avec la méthode 2D3/4 car la reconstruction partielle est plus sensible aux erreurs de mesure qu'une utilisation directe des coordonnées des points dans l'image. De plus, des erreurs sur la calibration et l'estimation de la profondeur se répercutent de manière plus évidente sur les trajectoires dans

l'image avec la méthode 2D 3/4. Même si on peut assurer la convergence du point de référence comme dans le cas de la méthode 2D1/2, la condition de visibilité de la tâche est donc en pratique plus difficile à satisfaire si on n'utilise pas une commande robuste.

2.8. Asservissement visuels dynamique (d2D/dt)

Les méthodes d'asservissement visuel 2D1/2, mises au point par Chamautte et son équipe [30], reposent sur l'utilisation d'informations visuelles géométriques (coordonnées de points, paramètres représentant l'image d'une droite, etc). Les contraintes imposées sont donc l'existence dans le scène de primitives, ainsi que la possibilité de les extraire et de les suivre grâce au traitement d'image à une cadence suffisamment élevée pour conserver la robustesse et la stabilité des lois de commande. Dans l'asservissement visuel d2D/dt, les mesures extraites des séquences d'images sont dynamique. Les créatures de cette méthode ne remontent plus cette fois aux primitives visuelles géométriques, mais proposent de contraindre directement les mesures dynamiques de manière à ce qu'elles atteignent des valeurs souhaitées. Plus pragmatiquement, le principe de la commande consiste alors à contrôler le mouvement de la caméra de telle sorte que le mouvement 2D mesuré atteigne un champ de vitesse désiré, d'où l'appellation d'asservissement d 2D/dt. Comme pour l'asservissement visuel 2D, il s'agit de contraindre les mouvements de la caméra de manière à ce que les informations visuelles acquises par celle-ci atteignent des valeurs souhaitées. La différence réside dans le fait que dans le cas 2D, il est nécessaire de passer par une étape d'extraction des primitives, alors que dans le cas d2D/dt, on souhaite seulement atteindre un champ de vitesse désiré.

Dans ce type d'asservissement, la grandeur asservie est une vitesse relative entre la caméra et la cible. La référence est définie dans le plan image par un champ de vitesse des points.

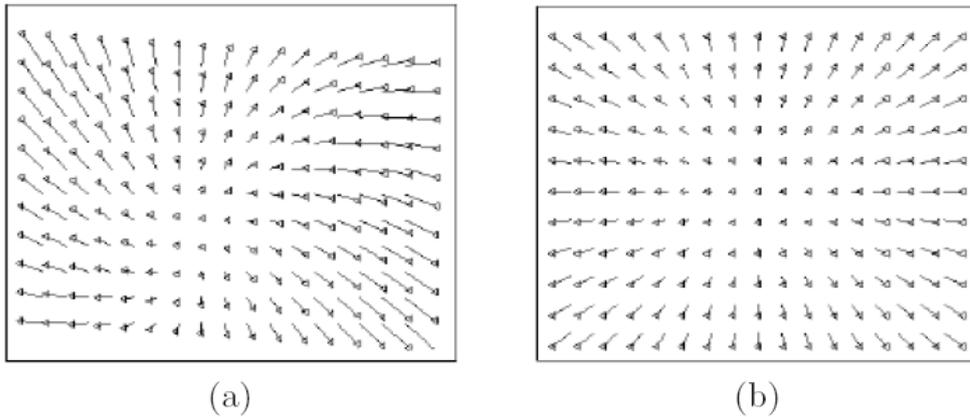


Fig 2.11 Champ de vitesse courant (a) et champ de vitesse désiré (b)

Il est important de noter que nous faisons ici la distinction entre : l'utilisation des informations visuelles de nature géométrique obtenues par le biais d'informations visuelles de nature dynamique (c'est-à-dire résultant d'une analyse du mouvement dans l'image) [31][32][33] et l'utilisation d'un schéma ou les informations visuelles utilisées dans la loi de commande sont dynamiques; Dans le [31], il a poursuivi les travaux en asservissement visuel basés sur l'utilisation dans la boucle de commande d'informations issues d'une analyse du mouvement 2D, et plus précisément des paramètres d'un modèle de ce mouvement estimé sur une séquence d'images segmentées ou non. À partir de ces informations, il est possible de remonter à des données visuelles géométriques par simple intégration des paramètres de mouvement estimés (si une localisation initiale est disponible) On peut alors utiliser directement les lois de commande qui ont été développées en asservissement visuel 2D. Cette approche a été validée sur une tâche de poursuite d'un élément quelconque par le contrôle de l'orientation de la caméra, ainsi que sur une tâche de stabilisation dynamique d'images acquises par une caméra embarquée sur un robot sous-marin. Une seconde approche consiste à définir l'objectif à atteindre sous la forme d'un champ de mouvement 2D désiré. Dans ce cas, les dérivées des paramètres considérés ne dépendent plus linéairement de la vitesse de la caméra, mais à la fois de l'accélération (de manière linéaire) et de la vitesse (de manière quadratique). Deux cas

peuvent se présenter selon que la matrice liant la dérivée de l'information visuelle à l'accélération de la caméra est de rang plein ou non. En cas de rang plein, la commande est réalisée en accélération, la fonction quadratique en vitesse étant considérée comme une mesure de l'état du système. Dans le cas contraire, la discrétisation d'une partie du vecteur d'accélération permet de définir une nouvelle relation d'interaction, conduisant à une commande effectuée en accélération sur certains axes et en vitesse sur les autres. Cette approche a été utilisée dans la réalisation d'une tâche de poursuite d'un objet mobile, ayant pour principe de maintenir nul le champ de mouvement apparent de l'objet d'intérêt. Que le modèle de mouvement considéré soit constant ou quadratique, la matrice d'interaction est de rang plein, conduisant à une commande en accélération [34].

2.9. Conclusion

Les systèmes d'asservissement visuel sont désormais opérationnels sur les sites industriels. Ses systèmes présentent des caractéristiques différentes et le choix d'une technique plutôt qu'une autre n'est pas toujours facile.

L'objectif n'étant pas de déterminer la meilleure technique mais la technique la plus adaptée, pour une application donnée suivant cinq critères : Les informations nécessaires pour la modélisation, les performances, la sensibilité aux bruits de mesure, le comportement dynamique et la robustesse aux erreurs de calibrage.

Troisième chapitre

La commande prédictive visuelle

3.1 Introduction

La commande prédictive ou Model Predictive Control (MPC) en anglais fait partie des techniques de commandes avancées régulièrement utilisée dans le domaine de l'Automatique. Depuis les années 1970, elle a fait l'objet de nombreux travaux de recherche et elle est actuellement l'une des commandes les plus utilisées en industrie dans des domaines très divers (raffinerie, chimie, alimentaire, etc.).

3.2. Commande prédictive

La raison de succès de la commande prédictive peut être attribuée au fait qu'elle peut être la façon la plus générale de poser le problème de la commande de procédés dans le domaine temporel. En effet, sa formulation intègre des

concepts tirés de la commande optimale , la commande stochastique , la commande de système à temps morts la commande avec modèle interne la commande multivariabe et prend en compte les références futures lorsqu'elles sont disponibles. La commande prédictive, ne désigne pas une stratégie de commande spécifique mais un ensemble de méthodes de l'automatique qui utilise explicitement un modèle du processus à commander afin d'obtenir le signal de commande par la minimisation d'une fonction de coût. Ces méthodes donnent des correcteurs linéaires qui ont pratiquement tous la même structure et qui se basent tous plus ou moins sur les idées suivantes :

- L'utilisation d'un modèle pour prédire les sorties du procédé à des instants futurs (notion d'horizon de prédiction)
- Calcul de la séquence des commandes qui minimise une fonction de coût dans le future (notion d'horizon de commande) .
- A chaque instant d'échantillonnage l'horizon de prédiction est déplacé vers le future, et seule la première des commandes calculées est effectivement appliquer au système (notion d'horizon fuyant).

Le problème de commande, régulation ou poursuite de trajectoire, est formulé en un problème d'optimisation non linéaire sous contraintes. En effet, l'atout majeur de cette stratégie de commande est la possibilité de prendre en compte explicitement les contraintes dans la synthèse de la loi de commande. La commande prédictive se base sur la connaissance *a priori* du procédé par l'intermédiaire d'un modèle. Les étapes spécifiques à toutes les lois de commande prédictive peuvent être classifiées comme suit :

- Elaboration du modèle du système sur lequel est basée la prédiction de la sortie ;
- Spécification de la trajectoire que doit suivre la sortie ;
- Minimisation d'un critère quadratique à horizon fini élaborant une séquence de commandes futures ;
- Application du premier élément de la séquence de commande au système et au modèle ;

Le modèle constitue la connaissance a priori du procédé à commander. Écrit sous forme d'équations mathématiques (fonction de transfert, représentation d'état, réponse impulsionnelle, etc.), il permet de prédire le comportement dynamique du procédé, notée y_p en fonction des commandes passées et futures. La trajectoire de référence, notée y_r définit les performances souhaitées du système commandé. Elle peut être constante ou variable. Le critère d'optimisation également appelé fonction de coût, est fonction de l'erreur entre la trajectoire de référence et la sortie prédite sur un horizon de prédiction fini noté N_p . La sortie prédite dépend de la séquence de commandes futures, notée U et composée de N_c commandes. La formulation mathématique de la fonction coût est, dans le cas discret multi-variable, donnée par :

$$J(u) = \sum_{j=k+1}^{k+N_p} (y_r(j) - y_p(j))^T Q (y_r(j) - y_p(j)) \quad (3.1)$$

où Q est une matrice symétrique définie positive.

A cette fonction de coût s'ajoutent les contraintes inévitables dans le contexte expérimental ou industriel. Ces contraintes peuvent être de deux types : les contraintes de fonctionnement et les contraintes physiques. Pour les contraintes de fonctionnement, la limitation des actionneurs en amplitude et/ou vitesse sont données par:

$$u_{\min} \leq u_k \leq u_{\max} \quad \text{et} \quad \Delta u_{\min} \leq u_k - u_{k-1} \leq \Delta u_{\max} \quad (3.2)$$

Pour les contraintes physiques, on a une limitation de certaines variables et pour la commande DMC le critère à minimiser est de la forme :

$$J(t) = \sum_{i=1}^{NP} (y_r(t+i) - y_p(t+i))^2 + \lambda \sum_{i=1}^{Nc} \Delta u(t+i) \quad (3.3)$$

avec λ est le facteur de pondération.

Le vecteur Δu et le critère $J(t)$ s'obtient par méthode des moindres carrés.

Le principe de la commande prédictive (fig.3.1) est de déterminer, à chaque période d'échantillonnage, la séquence de commandes optimales $\{u_k, u_{k+1}, \dots, u_{k+Np-1}\}$ qui minimise la fonction coût sur l'horizon de prédiction N_p .

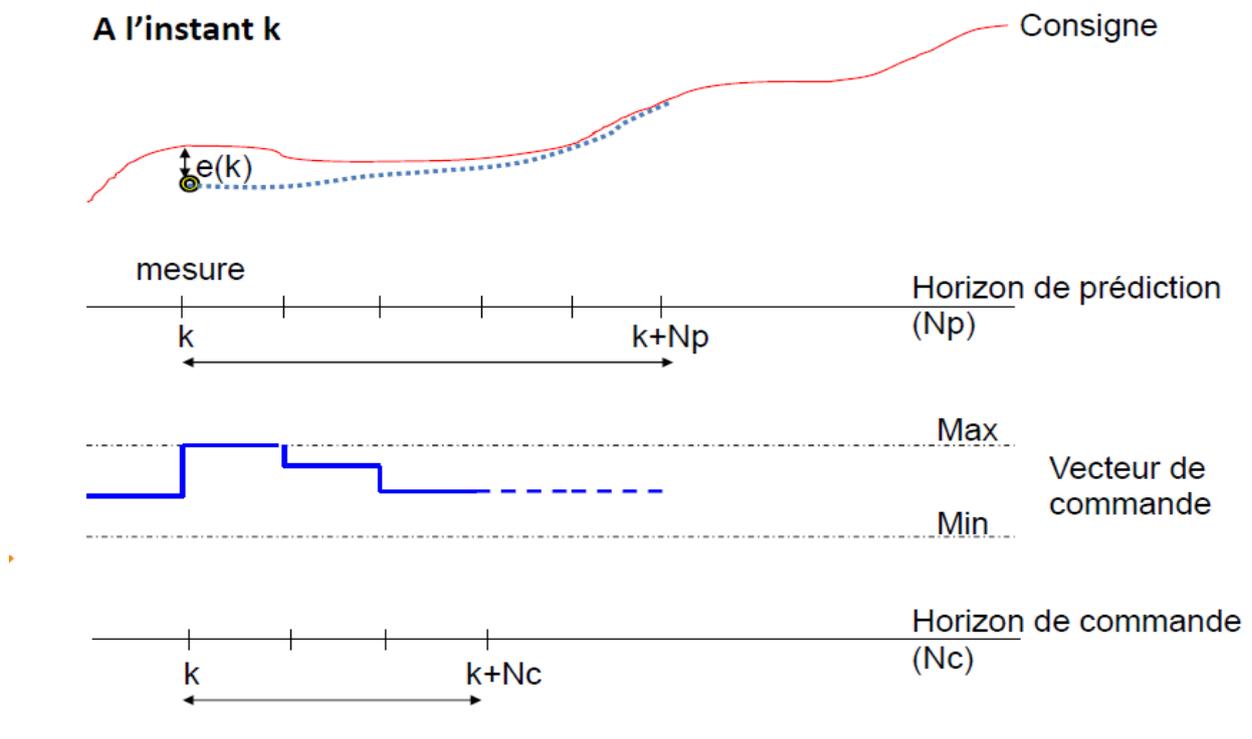


Fig.3.1 MPC à l'instant k

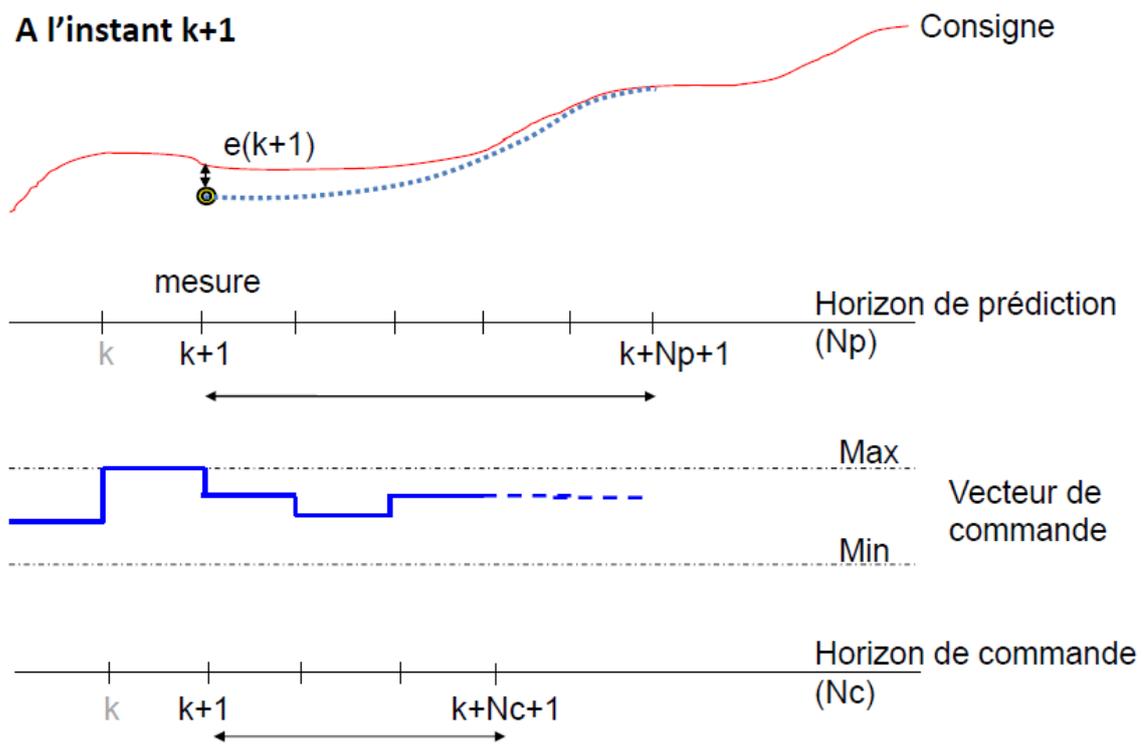


Fig.3.1.b MPC à $k+1$

Seule la première composante de cette séquence de commandes est réellement appliquée au système. A l'itération suivante, l'horizon de prédiction glisse d'un pas et la procédure d'optimisation est répétée. D'où son nom de commande optimale à horizon glissant.

3.2.1. Principe de la commande prédictive [41]

Le but de la commande prédictive basée sur le modèle MPC consiste à prédire la sortie future du système $y(t+j)$ et de la conduire afin qu'elle soit plus proche d'une référence $y_r(t+j)$ comme le montre la figure 3.1 (a et b). Cet objectif est atteint en s'appuyant sur la stratégie suivante :

1. Les sorties futures sur un horizon déterminé de taille Np , appelé l'horizon de prédiction, sont prédites à chaque instant k à l'aide du modèle du

processus. Ces prédictions sont notées $y(k+j/k)$, avec $j=1,\dots,N_p$, pour indiquer la valeur de la sortie à l'instant $k+j$ calculée à l'instant k . Elle dépendent des valeurs connues jusqu'à l'instant k (entrées et sorties passées) et des commandes futures $u(k+j/k)$, $j = 0,\dots,N_c-1$, qui sont celles à envoyer au système et qui doivent être calculées ;

2. L'ensemble des commandes futures est calculé en optimisant un critère déterminé pour garder le processus aussi proche que possible de la trajectoire de référence $r(k+j)$. Ce critère prend généralement la forme d'une fonction quadratique des erreurs entre le signal de sortie prédit et les consignes futures. L'énergie de commande est incluse dans la fonction de coût dans la plupart des cas. Une solution explicite peut être obtenue si le critère est quadratique, le modèle linéaire et s'il n'y a pas de contrainte, sinon une méthode d'optimisation itérative doit être utilisée. Des hypothèses sont également faites sur la structure de la loi de commande future comme par exemple qu'elle sera constante à partir d'un instant donné (horizon de commande).

3. La commande $u(k)$ est envoyée au système et les autres valeurs de commande sont oubliées, parcequ'à l'instant d'échantillonnage suivant la mesure $y(k+1)$ est déjà connue et que l'étape n°1 est répétée avec cette nouvelle valeur, toutes les séquences étant mises à jour. Alors on calcule la commande $u(k+1/k+1)$, qui est différente de $u(k+1/k)$ en principe à cause des nouvelles informations prises en compte.

3.3 . Stratégie de la commande prédictive

La stratégie de la commande prédictive est très similaire à la stratégie utilisée pour la conduite automobile. Le conducteur connaît la trajectoire de référence désirée (le tracé de la route) sur un horizon de commande fini (celui de son champ visuel), et en prenant en compte les caractéristiques de la voiture (modèle mental du comportement du véhicule), il décide quelles actions (accélérer, freiner ou tourner le volant) il faut réaliser afin de suivre la trajectoire désirée. Seule la première action de conduite est exécutée à chaque instant, et la procédure est répétée à nouveau pour les prochaines actions.

Noter que lorsqu'on recourt à des schémas de commande classique, comme les PID, les commandes sont calculées à partir des erreurs passées et de l'erreur courante. Si l'on étant l'analogie de la conduite automobile, la technique PID serait alors équivalente à conduire une voiture en utilisant seulement ce qui se passe juste devant la voiture (accès à la mesure courante) et le rétroviseur (mesures passées). Mais l'analogie est un peu sévère, car la commande MPC a besoin en réalité de beaucoup plus d'informations que le PID (la trajectoire à suivre par le système, et un modèle de ce système). Et si un point situé dans le futur sur la trajectoire de référence est utilisé comme consigne pour le PID, alors les différences entre les deux stratégies sont moins grandes.

L'algorithme d'optimisation est une partie fondamentale de la stratégie car c'est lui qui fournit les commandes. Si la fonction de coût est quadratique, son minimum peut être obtenu comme une fonction (linéaire) explicite des entrées et sorties passées et des consignes futures. La taille du problème d'optimisation dépend du nombre de variables et de horizons de prédiction et est en général relativement modeste (calcul de la pseudo-inverse d'une matrice par exemple). En présence de contraintes par contre (contrainte d'inégalité sur les commandes ou les sorties), la solution s'obtient numériquement à l'aide d'algorithmes beaucoup plus coûteux en temps de calcul.

3.4. Réglage des algorithmes de MPC [42]

La définition d'un critère quadratique a montré que l'utilisateur doit fixer quatre paramètres de réglage. Ce choix est délicat pour une personne non spécialisée car il n'existe pas de relations empiriques permettant de relier ces paramètres à des indicateurs classiques en automatique, tels que marge de stabilité ou bande passante. D'une façon ou d'une autre, les algorithmes de régulation prédictive comportent les paramètres de réglages suivants :

- L'horizon de prédiction N_p qui est, la longueur de l'intervalle de temps futur sur lequel on cherche à minimiser l'écart entre la prédiction et la sortie du procédé.

- L'horizon de commande N_c qui est le nombre de variations de la variable manipulée que l'on autorise dans le futur avant qu'elle ne soit fixée à une valeur constante.

- Le facteur de pondération des variations de l'entrée λ , qui augmente dans le critère à minimiser le poids des fluctuations de la variable manipulée et pour les algorithmes qui utilisent un modèle de convolution tronqué, la longueur de la réponse indicielle ou impulsionnelle L .

Il est nécessaire de bien choisir la période d'échantillonnage puisque le régulateur prédictif est du type discret. Le seul aspect par lequel le choix de la période d'échantillonnage peut être lié à la synthèse du régulateur prédictif est celui de la longueur du modèle de prédiction. Soit t_r le temps de réponse. L doit être choisi de sorte que $L * T_e = t_r$. Des valeurs de 20 à 70 sont les recommandations typiques pour L . Elle doit être suffisamment grande pour éviter les problèmes de troncature lors de la prédiction des valeurs futures du système. N_p représente le nombre de prédictions qui sont utilisées dans l'optimisation du critère. La valeur minimale que peut prendre N_p est égale au retard+1 c'est-à-dire $N_p \geq r+1$. Il est choisi entre deux valeurs : N_1 et N_2 . N_1 qui est l'horizon minimal de prédiction est choisi de telle sorte que le produit $N_1 T_e$ soit égal au retard pur du système. N_2 qui est l'horizon maximal de prédiction est choisi de telle manière à avoir le produit $N_2 T_e$ limité par la valeur du temps de réponse souhaité. Augmenter la prédiction au-delà du temps de réponse n'apporte aucune information supplémentaire et complexifie la résolution. L'horizon de commande N_c est le nombre d'actions futures qui sont calculées par la séquence d'optimisation pour réduire les erreurs de prédiction. Sa valeur influence fortement sur l'effet de calcul nécessaire. L'horizon de prédiction sur la commande doit être augmenté en fonction de la complexité du système à piloter. Pour les systèmes stables simples (SISO), N_c peut être choisi égal à 1, parce que dans ce cas le comportement du système en boucle fermée suit le comportement du système en boucle ouverte, en restant stable. Pour les systèmes instables, N_c doit être choisi (strictement) supérieur au nombre des

pôles instables. Augmenter N_c permet d'accélérer le système au prix d'une sollicitation plus importante de l'action et d'une plus forte sensibilité au bruit de mesure. Plus N_c est petit, plus le contrôle est robuste, en particulier vis-à-vis des erreurs de modèle. La marge de phase diminue pour des valeurs importantes de N_c . Pour le choix de λ qui est le facteur de pondération de la commande, il est comme l'équilibre de la balance. Si $\lambda=0$, on minimise uniquement la différence entre la consigne et la sortie prédite ; d'autre part si λ est très élevé, on pondère excessivement la commande qui n'est plus assez dynamique pour obtenir le ralliement à la consigne. λ est calculer de l'équation suivante : $\lambda = \text{trace}(A^T A)$.

3.5. Obtention du modèle [42]

Si l'on était capable de modéliser parfaitement les processus et les signaux, la commande serait quasiment réduite à un calcul en boucle ouverte ; à l'opposé, la boucle fermée a précisément été inventée pour piloter essentiellement avec comme seule information des mesures intervient explicite de la connaissance du processus. Entre ces deux approches extrêmes, la commande par modèle tente trouver un moyen terme qui apporterait la performance recherchée, tout en ne nécessitant pas une connaissance parfaite du processus. Le choix de la structure de modèle dépend du processus et des spécifications de la commande. Il doit être capable de réaliser une prédiction véritable de comportement futur du processus, sous l'effet d'une action supposée connue dans l'avenir. Ce modèle peut être de connaissance ou de représentation ; sa forme est ouverte. Il peut être de connaissance expérimentales, mathématiques ou simplement logique, à basse de règles ou même être constitué d'une base de données pratiques plus ou moins brute.

Il est possible à partir d'un modèle de réponse impulsionnelle donné de calculer la fonction de transfert équivalente d'un système, le calcul est toutefois plus compliqué et fait appel aux techniques d'identification paramétrique. Dans certain cas, des fonctions de transferts de Laplace continue sont obtenus à partir de modèles en temps discret afin de permettre ensuite de réaliser des

transformations variées selon des différentes périodes d'échantillonnage en temps discret.

Toutes les méthodes d'identification utilisées dans la commande prédictive basée sur le modèle minimisant la somme des carrés des erreurs entre les mesures et les prédictions afin d'obtenir un paramètre qui minimise l'équation:

$$\min \sum \|y_r - y_p\|^2 \quad (3.3)$$

La loi de commande prédictive implique la connaissance du comportement futur du système prédit à l'aide d'un modèle du processus. Ainsi l'élément central de la commande MPC est le modèle du système. Ce point fort peut devenir aussi son point faible, selon la qualité du modèle. Trouver le bon modèle (le plus simple possible, mais malgré tout suffisamment significatif et adapté aux besoins, en offrant des prédictions suffisamment précises) implique une connaissance appropriée du système. Les modèles résultent souvent d'une phase d'identification, qui peut se faire en utilisant les lois de la physique, de la chimie ou encore de façon expérimentale en effectuant diverses expériences sur le système. Le modèle doit être capable de prédire le comportement du système en réponse à une sollicitation donnée. Des techniques existent également, qui utilisent un modèle variant dans le temps sur l'horizon de prédiction, d'autres conçues sur un modèle à base de techniques floues. Prendre en compte les parties non-linéaires des systèmes par une modélisation floue et commander de façon prédictive ce modèle peut conduire à une amélioration des performances. Grâce à la simplicité du modèle, corrélée avec la souplesse du correcteur prédictif, cette vision devient intéressante en milieu industriel. Certes, les systèmes industriels sont rarement linéaires, mais dans la pratique la représentation choisie est souvent un modèle linéaire, induisant en l'absence de contraintes une structure linéaire de la loi de commande prédictive. Ainsi, de cette manière, l'optimisation et l'analyse hors ligne du comportement en boucle fermée sont beaucoup plus faciles. En dernier lieu, si la modélisation linéaire s'avère insuffisante, une mise en œuvre via un modèle non-linéaire peut s'envisager. Ces modèles servant à la prédiction sont classiquement des modèles à temps discret, dès lors que la commande prédictive est plutôt

implémentée sous forme discrète sur ordinateur. Malgré tout, des techniques de synthèse à temps continu existent.

3.6 Avantages et inconvénients de la commande prédictive

La commande prédictive présente un certain nombre d'avantages, par rapport aux autres méthodes:

- Son principe très intuitif et le réglage relativement facile de ses paramètres la rendent accessible aux personnes avec des connaissances limitées en automatique ;
- Elle peut être utilisée pour commander une grande variété de processus, ceux avec des dynamiques simples à ceux plus complexes, par exemple les systèmes à grand retard, à phase non minimale ou instable ;
- Le cas multivariable se traite facilement ;
- Elle est capable intrinsèquement de compenser les retards et les temps morts ;
- Le correcteur obtenu est une loi de commande linéaire facile à implémenter et qui requiert peu de temps de calcul ;
- Le traitement de contraintes sur le système à commander peut être inclus systématiquement dans la définition du correcteur ;
- Elle est très utile lorsque les consignes ou trajectoires à suivre sont connues à l'avance, ce qui est le cas par exemple dans la commande de robots industriels ou dans certains processus industriels.
- Elle peut s'appliquer aux systèmes lents et rapides, avec ou sans retard ;
- Elle est assez robuste aux erreurs de modèle ;
- Elle permet de prendre en compte facilement les perturbations mesurables pour faire une action de tendance.

Quelques désavantages existent aussi. L'un d'eux est que, bien que le correcteur obtenu soit simple à programmer, son obtention est beaucoup plus complexe que pour les correcteurs classiques. Si la dynamique du procédé ne change pas, le calcul du correcteur se fait à l'avance, hors-ligne, mais dans le

cas d'une commande adaptative, un nombre important de calcul doit être mené à chaque instant d'échantillonnage. Lorsque des contraintes sont considérées, le nombre des calculs nécessaire est encore plus grand. Bien que, avec l'énorme puissance des ordinateurs actuels, ce problème tende à s'estomper, il subsiste lorsque l'application nécessite des fréquences d'échantillonnage très élevées, notamment dans les applications de recherche en asservissement visuel rapides de robots. De plus, dans l'industrie, les systèmes de contrôle et de commande sur lesquels fonctionnent les correcteurs ont habituellement d'autres tâches à accomplir, comme la gestion d'alarmes, l'enregistrement de données, les communications avec les autres systèmes et les opérateurs, ce qui réduit les ressources disponibles pour le correcteur. Le plus grand des inconvénients est le besoin d'un modèle approprié du système. L'algorithme de calcul du correcteur prédictif se base sur une connaissance a priori de ce modèle, il est donc évident que les bénéfices obtenus avec la commande prédictive sont affectés par les écarts qui peuvent exister entre le processus réel et le modèle utilisé.

3.7. Commande prédictive visuelle [43][44][45][46]

Les systèmes utilisés en asservissement visuelle sont des systèmes robotiques soumis à des contraintes (limitations des actionneurs, butée mécaniques, ...). De plus, on a vu que l'asservissement visuelle consiste à minimiser une erreur dans le plan image. La commande prédictive semble donc bien adaptée pour traiter ce problème. Son principe, étendu à l'asservissement visuelle, commande prédictive visuelle (Visual Predictive Control (VPC) en Anglais), consiste à déterminer une séquence de commandes qui minimise l'erreur définie entre les primitives de référence ($image_{réf}$) et les primitives extraites à partir de l'image courante. La prédiction de l'image sur l'horizon de prédiction est assurée par le modèle global (caméra et robot). On utilise alors la structure bien connue de commande à modèle interne (fig.3.2).

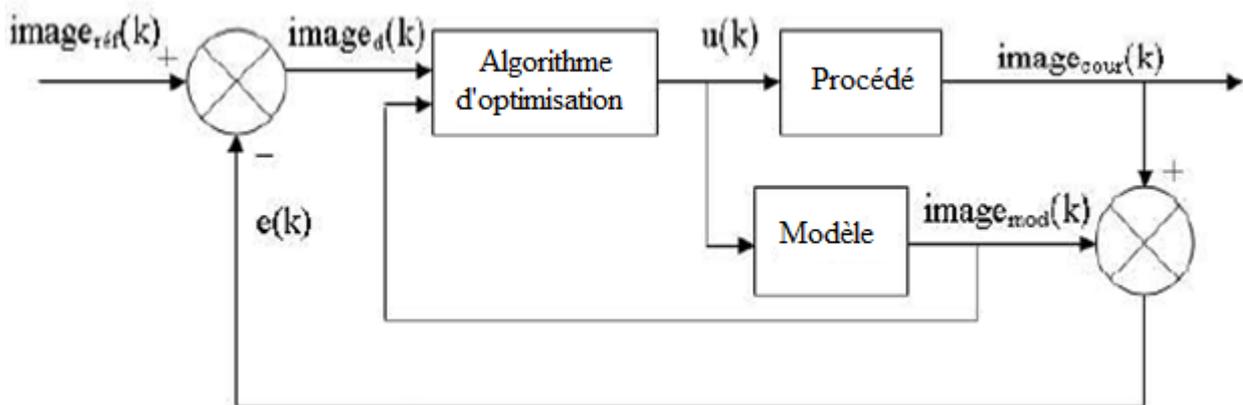


Fig.3.2 Schéma de commande à modèle interne

D'après ce schéma de commande, on peut écrire :

$$\text{image}_d(k) = \text{image}_{\text{réf}}(k) - e(k)$$

$$\text{image}_d(k) = \text{image}_{\text{réf}}(k) - (\text{image}_{\text{cour}}(k) - \text{image}_{\text{mod}}(k))$$

$$\text{image}_d(k) - \text{image}_{\text{mod}}(k) = \text{image}_{\text{réf}}(k) - \text{image}_{\text{cour}}(k)$$

Le signal $e(k)$ représente les erreurs de modélisation et les perturbations. Lors d'applications réelles, ce signal est souvent filtré. Imposer à l'image courante de suivre l'image de référence revient à faire tendre vers zéro la quantité $(\text{image}_d(k) - \text{image}_{\text{mod}}(k))$. La CPV peut alors s'écrire sous forme d'un problème d'optimisation non linéaire sous contraintes comme suit :

$$\min_u J = \sum_{j=k+1}^{k+Np} (\text{erreur}(j))^T Q \text{erreur}(j) \quad (3.4)$$

Où nous avons :

- ✓ $\text{erreur}(j) = \text{image}_d(j) - \text{image}_{\text{mod}}(j)$.
- ✓ $\text{image}_{\text{mod}}(j)$: image prédite par le modèle non linéaire du système caméra plus robot.
- ✓ $u = \{u_k, u_{k+1}, \dots, u_{k+Np-1}\}$ vecteur de commandes pouvant être soumis aux contraintes définies précédemment (3.2).

Le problème d'optimisation (3.4) peut être résolu par des routines d'optimisation qui existe dans les bibliothèques logicielles (Matlab). Par exemple sequential quadratic program (SQP) qui est la plus précise ou la fonction `fmincon` disponible dans la boîte à outil d'optimisation de Matlab. Il peut aussi être transformé en un problème pénalisé (méthode des pénalités) et résolu ensuite par un algorithme d'optimisation non linéaire sans contraintes. Dans un souci d'applicabilité temps réel, l'algorithme d'optimisation utilisé doit pouvoir satisfaire, à tout instant, les contraintes et ceci avec un temps de calcul le plus petit possible. Une modification d'algorithme existant reconnu pour son efficacité et sa robustesse a été proposée par Caucial ; Dufour et Touré dans commande prédictive non linéaire sous contrainte, une condition de faisabilité en 2004 pour satisfaire ces exigences. Un des avantages principaux de VPC est la capacité pour manier des contraintes explicitement dans le problème de l'optimisation. Trois genres de contraintes sont distingués:

- Contraintes sur l'état du système robotique. Pourrait être typiquement une contrainte mécanique telle que limite de l'espace de travail si on prend la pose de la caméra comme étant une variable d'état.

$$\mathbf{X}_{\min} \leq \mathbf{X}_k \leq \mathbf{X}_{\max} \quad (3.5)$$

- Contraintes de la visibilité pour assurer que les primitives visuelles restent dans l'image ou représentent des régions défendues dans l'image. Le dernier peut être très utile à négocier avec action d'évitement d'obstacle ou occlusion de l'image.

$$\mathbf{S}_{\min} \leq \mathbf{S}_m \leq \mathbf{S}_{\max} \quad (3.6)$$

où \mathbf{S}_m est la mesure des primitives dans l'image de référence.

- Contraintes de fonctionnement telles que limitations de l'actionneur en amplitude et/ou vitesse,

$$\mathbf{u}_{\min} \leq \mathbf{u}_k \leq \mathbf{u}_{\max} \quad (3.7)$$

Ces contraintes sont ajoutées au problème (3.4)

Le modèle du système global robot-caméra est un élément crucial de la stratégie VPC. Il permet de prédire l'évolution des primitives visuelles et contrôler ces variations sur l'horizon N_p . Il peut être comparé à une boîte noire d'où les entrées sont les commandes du système robotique et les productions sont la prédiction des primitives visuelles. La figure suivante illustre les différents pas des configurations des primitives prédites par le modèle global (fig.3.3).

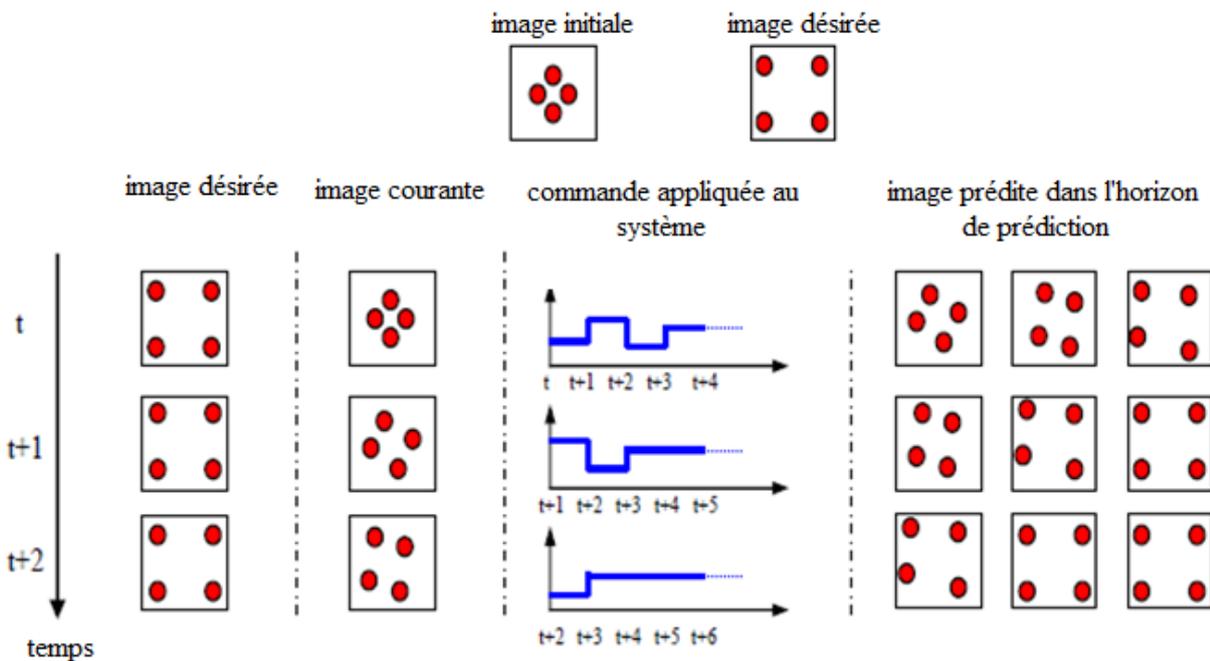


Fig.3.3 Différents pas des configurations des primitives prédites par le modèle global

- L'objectif de commande : faire tendre les primitives courantes vers les primitives désirées. Cela revient à minimiser une erreur dans le plan image (2D).
- Par quel moyen ?
Soit en utilisant une matrice d'interaction reliant les variations des primitives aux variations du vecteur de commande. Cela revient à utiliser un modèle, ou la résolution du problème de l'optimisation sous contraintes dans ce cas la matrice de l'interaction n'est pas nécessaire, par

conséquent, difficultés de l'inversion et les problèmes de la singularité sont effacés.

- Satisfaire des contraintes de visibilité (2D) et des contraintes mécaniques (3D), cela revient à définir un domaine de contraintes.

3.8. Conclusion

Ce chapitre a présenté le principe de la commande prédictive en insistant sur ses particularités qui en font une méthode de la commande particulièrement efficace et très répandu dans la communauté automatique, ainsi qu'à son application à l'asservissement visuel (commande prédictive visuelle) qui est une nouvelle architecture de commande, elle augmente la cadence de la boucle de vision et adaptable à tout type de configuration. Elle assure la gestion naturelle des contraintes et se révèle robuste vis à vis d'erreurs de modélisation, erreurs quasi inévitables sur la caméra. De plus, la possibilité de prendre en compte explicitement les contraintes dans la synthèse de la loi de commande présente un réel avantage pour des applications réelles.

Quatrième chapitre

Application au suivi d'une cible

4.1. Introduction

La plupart des robots manipulateurs travaillent dans des environnements qui leur sont dédiés. Ces robots, que l'on trouve principalement sur des unités de production, ont des lois de commande prédéfinies pour la tâche qu'ils ont à effectuer. Leurs trajectoires ainsi que leurs actions peuvent être cycliques, répétitives ou souvent préenregistrées.

Ces dernières années, on dote les robots de différents capteurs leur permettant d'interagir avec l'environnement. Ainsi, l'utilisation d'une ou plusieurs caméras permet aux robots de réaliser des tâches complexes et de s'adapter aux exigences de l'environnement. Le traitement des informations acquises par la caméra permettra de formuler une loi de commande pour orienter le robot suivant les changements de l'environnement. Il est, par exemple, possible d'asservir visuellement pour guider le bras d'un robot vers une cible connue sans connaître au préalable la position de celle-ci ou pour réaliser une tâche

d'évitement d'obstacle. Pour ce faire, il est nécessaire de concevoir et réaliser une application qui gère les images et leurs traitements ainsi que la commande du robot y afférent.

Dans ce chapitre, il s'agit de mettre en application cette idée en prenant comme cible un objet de forme très simple pour vérifier le principe du suivi. Cette opération est extensible à tout autre objet quelle que soit sa complexité pourvu que les traitements d'images qui interviennent arrivent à réduire et à présenter cet objet sous une forme simplifiée. Cette application est faite sur la base du travail de Magister en Automatique de Monsieur A. Triki [47].

4.2. Application au suivi d'un objet

4.2.1. Description physique de l'application

L'objectif de ce chapitre consiste donc en la localisation d'un objet bien précis d'une image et effectuer son suivi, autrement dit, il s'agit de commander les mouvements d'un bras manipulateur (fig.4.1) pour que son organe terminal (pince) suit les mouvements d'un objet dont l'image de couleur unie se déplaçant horizontalement sur un fond d'une autre couleur contrastée. Pour suivre cet objet, une caméra est fixée à l'organe terminal du robot. Ce robot est connecté à travers une carte d'E/S PCI-6503 à un PC auquel est reliée aussi la caméra. Le robot est commandé via une carte réalisée au sein du laboratoire de recherche qui nous a accueillies telle qu'elle est décrite dans [47].

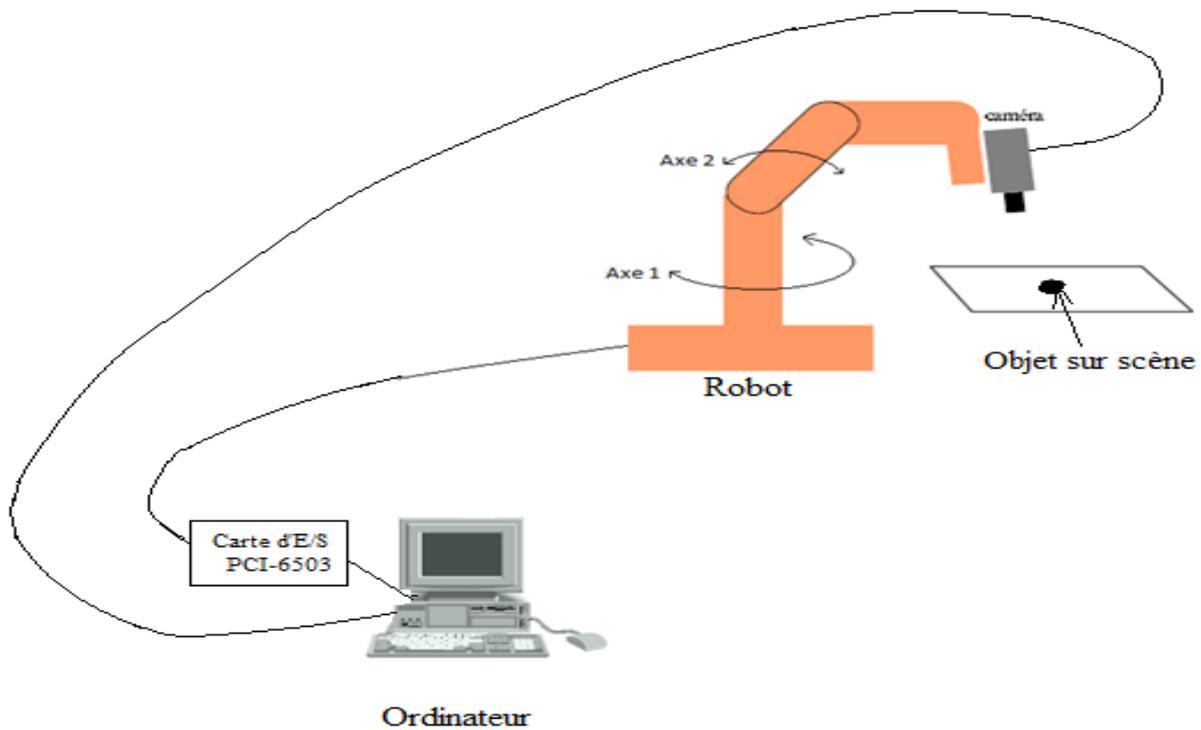


Fig.4.1 Schéma de principe de suivi d'objet

4.2.2. Description fonctionnelle de l'application

Pour réaliser notre application, nous avons d'abord procédé à l'acquisition de l'image à l'aide d'une caméra montée sur l'organe terminal du robot. Les images de dimension 160×120 sont transmises à l'ordinateur. Cette image est binarisée de sorte que la surface représentant l'objet soit en noir et le reste de la surface en blanc. Pour ce faire, nous calculons un seuil de niveau de gris s qui représente la frontière entre les niveaux de gris de l'objet et ceux du fond sur lequel est posé cet objet. Ensuite, pour chaque pixel $p(i, j)$ de niveau de gris $img(i, j)$, on applique la transformation (4.1) suivante :

$$imb(i, j) = \begin{cases} 1 & \text{si } img(i, j) > s \\ 0 & \text{si } img(i, j) < s \end{cases} \quad (4.1)$$

où $imb(i, j)$ est l'image binaire obtenue, $img(i, j)$, l'image en niveau de gris après conversion de l'image en niveaux de gris si celle-ci est en couleur et s , le seuil.

Le centre de gravité $G(x_g, y_g)$ est calculé à partir du contour de l'objet pour situer cet objet dans l'image. Les coordonnées du centre de gravité de l'objet sont comparées aux coordonnées de centre de l'image globale ou pixel central (x_c, y_c) pour commander le robot. Ainsi,

- Si $x_g > x_c$, l'objet est à droite,
- Si $x_g < x_c$, l'objet est à gauche,
- Si $y_g > y_c$, l'objet vers le « bas »,
- Si $y_g < y_c$, l'objet est vers le « haut ».

Il s'agit ensuite, pour le robot, de se déplacer pour ramener les coordonnées (x_g, y_g) pour les faire coïncider avec les coordonnées (x_c, y_c) du centre de gravité de l'objet. Ceci est effectué par la commande qui consiste à élaborer l'entrée à envoyer au robot. On peut définir ainsi e_1 , e_2 , e_3 et e_4 pour réaliser cette commande.

Ainsi :

- si l'objet est à gauche ($x_g < x_c$), mettre $e_1=1$ et $e_2=0$ pour déplacer la caméra à gauche,
- si l'objet est à droite ($x_g > x_c$), mettre $e_1=0$ et $e_2=1$ pour déplacer la caméra à droite,
- si l'objet est en haut ($y_g > y_c$), mettre $e_3=1$ et $e_4=0$ pour déplacer la caméra vers le haut,
- si l'objet est en bas ($y_g < y_c$), mettre $e_3=0$ et $e_4=1$ pour déplacer la caméra vers le bas,
- si $x_g=x_c$ mettre $e_1 = 0$ et $e_2 = 0$.
- si $y_g=y_c$ mettre $e_3 = 0$ et $e_4 = 0$.

4.2.3. Réalisation

La réalisation de la procédure est effectuée, nous avons suivi l'algorithme de la figure 4.2.

1. Acquisition de l'image couleur de dimension 160x120.
2. Conversion de l'image couleur acquise en une image niveau de gris.
3. Binarisation de l'image (seuillage).
4. Calcul de coordonnées de centre de gravité.
5. Déterminer la position de l'objet par rapport au centre de la caméra.
6. Génération des commandes au robot.

Fig.4.2 : Algorithme de la procédure de suivi d'un objet en mouvement

Ce travail est réalisé sous Matlab-Simulink en suivant le schéma bloc de la figure 4.3.

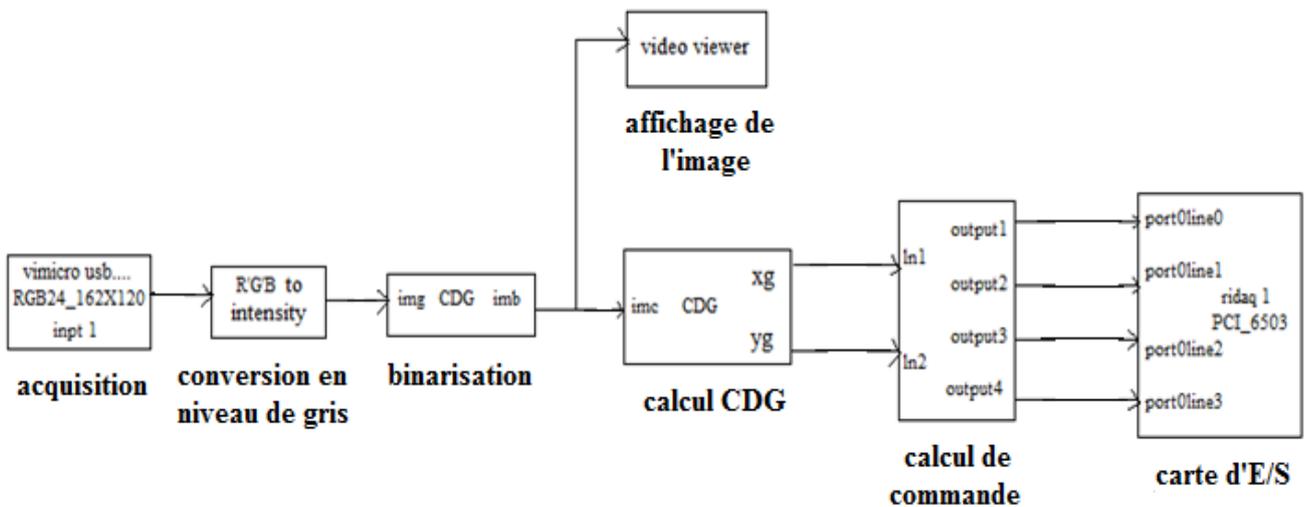


Fig.4.3 Schéma de simulation

Dans ce schéma, le bloc « acquisition » utilise le bloc Simulink (From Video Device) de la bibliothèque Image Acquisition toolbox pour acquérir des images couleurs de dimension 160x120 de la caméra. Le bloc « conversion en niveau de gris » utilise le bloc Color Space Conversion de la bibliothèque « Video and Image Processing Blockset » pour convertir les images couleurs acquises en images au niveau de gris. Le bloc binarisation utilise le bloc « Embed de Matlab Function » de la bibliothèque Simulink dont lequel nous avons programmé une fonction qui permet de binariser des images de niveau gris. Le bloc calcul « CDG » utilise le bloc « Embedde Matlab Function » de la bibliothèque Simulink dans laquelle nous avons programmé une fonction qui permet de calculer le centre de gravité de l'objet dans l'image binaire. Le bloc affichage de l'image utilise le bloc « Video Viewer » de la bibliothèque « Video and Image Processing Blockset » pour afficher les images binaires. Le bloc de calcul de commande réalisé avec un bloc « Sub System » comportant des comparateurs permet de comparer les valeurs de coordonnées du centre de gravité de l'objet aux valeurs de coordonnées du centre de l'image et de mettre à 1 ou à 0 les commandes de robot e_1 , e_2 , e_3 et e_4 . Le bloc carte d'E/S utilise le bloc « Digital Output » de « Data Acquisition Toolbox » pour envoyer les valeurs de commande e_1 , e_2 , e_3 et e_4 en sortie de la carte d'E/S PCI-6503.

3.3. Conclusion

Ce chapitre est consacré à la réalisation d'un asservissement visuel permettant de suivre un objet situé dans une image. Pour la commande visuelle du bras manipulateur dont le but de suivre l'objet dans l'image, les traitements doivent être les plus rapides possibles. Ainsi, nous avons opté pour la localisation basée sur le centre de gravité de l'objet après un traitement de l'image acquise (binarisation et calcul de contours). Nous avons suivi une démarche rapide à l'aide de l'environnement de simulation Matlab qui permet d'arriver rapidement à l'évaluation de techniques, notamment grâce à sa « Toolbox Image ».

Conclusion générale

La vision est devenue un outil essentiel dans un système robotique, elle permet au robot d'accomplir des tâches nouvelles afin d'évoluer et de s'adapter à son environnement et aux changements pouvant survenir. Néanmoins, l'utilisation d'informations visuelles va inévitablement rendre la commande des robots plus complexe. C'est ainsi que des techniques ont été mises au point pour répondre au mieux à ces problèmes. L'asservissement visuel exploite toutes les informations possibles dans l'image, ce qui permet d'atteindre des niveaux de précision plus élevés, tout en étant efficace en temps de calcul. Ce dernier aspect est également appréciable puisque l'exécution en temps réel est toujours importante pour les systèmes robotiques.

Ce mémoire a porté sur l'étude de l'asservissement visuel, dans le premier chapitre on a rappelé ces généralités sur ce dernier. On a mis l'accent sur les travaux effectués dans ce domaine depuis 1970, par la suite, on a défini l'asservissement visuel, son principe, ces différentes tâches qui sont le positionnement, le suivi de cible et la navigation, puis on a vu sa classification selon la position de capteur visuel, selon le type de commande et selon la grandeur asservie. On a traité aussi un point très important concernant la nature des informations visuelles qui seront aptes pour la tâche que l'on souhaite accomplir, puis, on s'est intéressé sur l'intégration de la vision en robotique où on a cherché à connaître la relation entre l'image obtenue par le capteur visuel et la scène observée, on a constaté qu'il suffit d'exprimer les relations de passage du repère monde au repère caméra, d'exprimer la projection du repère caméra dans le plan image et d'appliquer la transformation qui conduit aux coordonnées de l'image, on a traité aussi le calibrage ainsi que quelques notions de base de la modélisation d'une caméra. Le second chapitre,

Conclusion générale

a été dédié aux techniques d'asservissement visuel, où on a détaillé les asservissements 2D qui a pour objectif d'atteindre un motif dans l'image, on a traité la notion de jacobien de cette dernière, le choix de la matrice C pour déterminer la fonction de tâche et les lois de commande existantes pour la réguler à zéro. Dans l'asservissement visuel 3D, on a vu l'estimation de la pose, ces intérêts et inconvénients. Une autre technique d'asservissement visuel conservant les avantages respectifs des approches classiques 2D et 3D, et palliant leurs inconvénients qui est 2D1/2. Un schéma de contrôle, intermédiaire entre l'asservissement visuel 2D1/2 et 3D est appelé 2D3/4. Si la loi de commande consiste à contrôler les mouvements de la caméra de telle sorte que le mouvement 2D mesuré atteigne un champ de vitesse désiré on parle alors d'asservissement visuel $d2D/dt$. Dans le troisième chapitre, une stratégie de commande avancée est proposée pour effectuer des tâches d'asservissement visuel. La technique utilisée se base sur une commande prédictive appliquée à l'asservissement visuel car elle est la mieux adaptée pour prendre en compte les contraintes existantes dans ce dernier, elle devient donc la commande prédictive visuelle. Dans le dernier chapitre, nous nous sommes intéressés au suivi d'objet situant dans une image à partir d'une boucle d'asservissement visuel. L'approche que nous avons choisie est originale dans le sens où elle permet de générer les mouvements du bras de robot à partir de coordonnées de l'objet dans l'image. En effet, en premier lieu on a déterminé la position de l'objet dans l'image acquise, ensuite, on a élaboré une loi de commande au robot pour suivre la cible.

Bibliographies:

- [1] J. Hill et W. T. Park: Real time control of a robot with a mobile camera, ISIRI, USA, 1979.
- [2] Hager. G, Hutchinson. S et Croke. P, Visual Servo Control, IEEE international conference on robotics and automation, Minneapolis Hilton and Towers, Minnesota, Tutorial TT3. 1996.
- [3] Hashimoto. K, Visual Servoing: Real-time Control of Robot Manipulator Based on Visual Sensory Feedback. Vol. 7 of Robotics and Automated System, ISBN 981-02-1364-6, World Scientific Publishing Co. 1993.
- [4] Y. Shirai et H. Inoue, Guiding a robot by visual feedback in assembling tasks, pattern recognition, vol. 5, pp. 99-108, 1973.
- [5] A. Gilbert, M. Giles, G. Flachs, R. Rogers et H. Yee, A real-time video tracking system, IEEE Transaction on pattern analysis and machine intelligence, vol. 2, pp. 47-56, 1980.
- [6] C. Geschke, A robot task using visual tracking, Robotics today, pp. 39-43, 1981.
- [7] L. Weiss, Dynamic visual servo control of robots: an adaptive image-based approach: Carnegie-Mellon University, 1984.
- [8] K.A. Dzialo et R.J. Schalkoff, control implication in tracking moving objects using time-varying perspective imagery, IEEE Transaction on Industrial Electric, vol, IE-33, pp. 247-153, 1986.
- [9] M.Kabuka, J. Desoto et J. Miranda, Robot vision tracking system, IEEE Transaction on Industial Electronic, vol. 35, pp. 40-51, 1988.
- [10] Martinet, P.Betty et Gallice, J.1996a, Use of first derivative of geometric features in visual servoing. Proceedings of the IEEE International Conference

Bibliographies

- on Robotics and Automatisation. Vol.4ICRA. Minneapolis, Minnesota, USA. Pp. 3413-3419. 1996.
- [11] Salganicoff.M, Metta. G, Oddera. A and Sandini. G, A direct approach to vision guided manipulation. Proceedings of the international Conference on Advanced Robotics. ICAR'. Tokyo, Japan, pp. 133-138.1993.
- [12] Michel. H et Rives. P, Singularities in the determination of the situation of a robot effector from the perspective view of 3 points. Technical Report interne de recherche INRIA.1993
- [13] Marchand. E, Chaumette.F et Rizzo. A, Using the task function approach to avoid joint limits and kinematics singularities in visual servoing, Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Vol. 3. IROS'96, Osaka, Japan. Pp. 1083-1090. 1996.
- [14] Chaumette. F, 1998b, Potential problems of stability and convergence in image based position-based visual servoing, The conference of Vision and Control, LNCIS. 1998.
- [15] Gangloff. J et al. d.o.f high speed dynamic visual servoing using GPC controllers, Proceedings of the IEEE International Conference on Robotics and Automatisation. Leuven, Belgique . 1998.
- [16] Hashimoto. K, Kimoto. T et Kimura. H, Image based dynamic Visual for a hand-eye manipulator, Proceedings of the MTNS'91. Kobe, Japan. Pp. 609-614. 1991.
- [17] Papanikolopoulos. N, Khosla. Et Kanade. T, Visual tracking of a moving target by a camera mounted on a robot : a combination of control and vision. IEEE Transaction on Robotics and Automatisation. pp. 14-35. 1993.
- [18] Corke. P, visual control of robots, High Performance Visual Servoing, Research Studies Press, ISBN 0-86380-207-9. 1996.
- [19] Khadraoui. D, La commande Référencée Vision pour le guidage de véhicules, PhD thesis, université Blaise-Pascal, Clermont-Ferrand, France. 1996.

Bibliographies

- [20] A. Crétual et F. Chaumette, Positioning a camera parallel to a plane using dynamic visual servoing . Dans *IEEE/RSJ Int. Conférence.on Intelligent Robots and Systems, IROS'97*, p. 43–48, Grenoble, Septembre1997.
- [21] A. Crétual et F. Chaumette, Image-based visual servoing by integration of dynamic measurements . Dans *IEEE Int. Conférence on Robotics and Automation, ICRA'98*, Louvain, Belgique, Mai 1998.
- [22] J. Santos-Victor et G. Sandini, Visual behaviors for docking . *Computer Vision and Image Understanding*, vol. 67, No 3, p. 223–238, September 1997.
- [23] 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.
- [24] F. Chaumette, *La relation vision-commande : théorie et application à des tâches robotiques*. Thèse de doctorat, Université de Rennes I, Rennes, Juillet 1990.
- [25] 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.
- [26] W.J. Wilson, C.C.W. Hull et G.S. Bell, Relation end effector control using cartesian position-based visual servoing _ *IEEE Transactions on Robotique and Automation*. Vol. 12, n^o5, p 684-696, Octobre 1996.
- [27] E. Malis: *Contribution à la modélisation et à la commande en asservissement visuel*, thèse de doctorat. Université de Rennes 1, mention : Traitement de Signal et Télécommunication, Novembre 1998.
- [28] E. Colle et F. Chaumette, Perception de l'environnement en robotique, Paris, Hermès, 1998.
- [29] F. Dornaika, Contribution à l'intégration Vision/Robotique : Calibrage, Localisation et Asservissement. Thèse de l'INP Grenoble, Septembre 1995.
- [30] J. Gangloff, M. De Martinin et Abba, 6 DOF high speed dynamic visual servoing using GPC controllers_ dans *Proceeding of the IEEE International Conference on Robotics and Automation (ICRA'98,Louvain)*, p 1008-1013, 1998.

Bibliographies

- [31] W. Khalil, E. Dombre, *Modélisation, identification et commande des robots*, Hermès, Paris, deuxième édition, 1999.
- [33] J. Gangloff : *Asservissement visuel*, notes de cours. Novembre 2004.
- [34] SANDERSON, A. et WEISS, L. "Image-based visual servo control using relational graph error signals". Dans *IEEE International Conference on Cybernetics and Society*. Cambridge, Massachusetts, pp. 1074–1077. Octobre 1980.
- [35] Horaud, R., B. Conio, O. Leboulleux et B. Lacolle . An analytic solution for the perspective 4-point problem. *Computer Vision Graphics and Image Processing*48(2), 277–278. 1989.
- [36] Chaumette, F. (1998*b*) . Potential problems of stability and convergence in imagebased and position-based visual servoing . Dans : *The Confluence of Vision and Control* (D. Kriegman, G. Hager and A.S. Morse, Eds.). pp. 66–78. LNCIS Series, No 237, Springer-Verlag. 1998.
- [37] Zanne, P., G. Morel et F. Plestan . A robust 3d vision based control and planning. Dans : *IEEE Int. Conf. on Robotics and Automation*. New Orleans, USA. 2004
- [38] E.Malis, F. Chaumette et S. Boudet, Positioning a coarse-calibrated camera with respect to an unknown planar object by 2D1/2 visual servoing, presented at IEEE international conference on Robotics and Automation ICRA'98, Lueven, Belgique, 1998.
- [35] F.Huynh : Manipulation par la vision pour des tâches de surveillance et de saisie d'objet fixe ou mobile. PhD. Université Paul Sabatier, 1999.
- [32] 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, Avril 1997.
- [35] A. Cretual : *Asservissement visuel à partir d'informations de mouvement dans l'image*, thèse de doctorat. Université de Rennes 1, mention : Traitement de Signal et Télécommunication, Novembre 1998.
- [36] R. Ginhoux : *Compensation des systèmes physiologiques en chirurgie robotisée par commande prédictive*, thèse de doctorat. Université de Louis Pasteur , Strasbourg , spécialité : Robotique médicale, Décembre 2003

Bibliographies

- [37] F. Martinet : *Asservissement visuel*, Habilitation à diriger des recherches. Université Blaise Pascal Clermont-Ferrand. Janvier 1999.
- [38] T. Dallaj : *Contributions par un modèle générique pour l'asservissement visuel des robots parallèles par observation des éléments cinématiques*, thèse de doctorat. Université de Blaise-Pascal Clermont 2, spécialité: Vision pour la Robotique. Décembre 2007.
- [39] E.Malis, F. Chaumette et S. Boudet, Positioning a coarse-calibrated camera with respect to an unknown planar object by 2D1/2 visual servoing, presented at IEEE international conference on Robotics and Automation ICRA'98, Lueven, Belgique, 1998.
- [40] H. Hadj Abdekaer, Y. Mezouar, P. Martinet et F. Chaumette: *Asservissement visuel en vision omnidirectionnelle à partir de droites*, article 2005.
- [41] R. Ginhoux : *Compensation des systèmes physiologiques en chirurgie robotisée par commande prédictive*, thèse de doctorat. Université de Louis Pasteur , Strasbourg , spécialité : Robotique médicale, Décembre 2003.
- [42] M.Chalal, T.Chabane et A.Hamdi : *Application de la commande prédictive basée sur le modèle pour la commande du niveau dans un bac de stockage*, projet de fin d'étude d'ingénieur d'état en automatique. Université de Mouloud Mammeri de Tizi-Ouzou, 2003.
- [43] E. Caurtial: *Commande prédictive et asservissement visuel*, Institut Prisme, journée de pole « contrôle ». Décembre 2009.
- [44] R. Challal : *Asservissement visuel sur HOAP - 3*, rapport de master 2. Université de Montpellier 2, spécialité : Robotique et Automatique. Septembre 2010.
- [45] G. Allibert, E. Courtial et Y. Touré : *Visual Predictive Control for Manipulators with Catadioptric Camera*. Université Orléans, 2010.
- [46] G. Allibert, E. Courtial et F.Chamaute: *Visual Servoing via Nonlinear Predictive Control*: Manuscrit de l'auteur, publié dans "Visual Servoing via Advanced Numerical Methods LNCIS 401, Springer-Verlag (Ed.)" 2010.
- [47] A.Triki : *Contribution à l'étude et réalisation d'un asservissement comprenant une table tournante, un robot et un système de vision artificielle*, mémoire de magister, option : Automatique des systèmes continus et productique. Université Mouloud MAMAERI de Tizi-Ouzou. Décembre 2010.