

Chapitre 1

Asservissement visuel

La vision représente pour les humains environ 70% des données issues des perceptions sensorielles externes [23] : d'une richesse incroyable, c'est aussi, par conséquent, une source d'erreurs inépuisable. La vision artificielle ne déroge pas à ses deux principes, d'où l'importance du traitement du signal d'une part (dans le but de recueillir et interpréter dans cette grande variété d'informations ce qui est pertinent pour une tâche prescrite) et d'une stratégie d'asservissement d'autre part (afin de stabiliser une tâche malgré les erreurs et incertitudes sur les données).

Nous allons donc dans un premier temps présenter le mode d'acquisition des images, puis nous continuerons sur les modèles de perception utilisés. Une fois que nous aurons détaillé la manière dont une scène est projetée sur une image, nous pourrons dans une seconde section distinguer plusieurs configurations et exposer différentes stratégies d'asservissement pouvant être utilisées. Une troisième section sera consacrée à la construction des lois de commandes, dont nous fournirons quelques exemples. La quatrième section de ce chapitre présentera les spécificités d'une utilisation de l'asservissement visuel dans le contexte des manipulateurs parallèles à câbles : après avoir présenté les principaux travaux ayant été effectués dans ce domaine, nous pourrons indiquer les angles d'études que nous avons privilégiés, les problématiques qui en émergent, et les pistes de résolution que nous avons empruntées.

1.1 Modèles de capteurs et projections

1.1.1 De l'oeil à la caméra

Sans pour autant chercher à imiter la vision humaine, la vision artificielle s'en inspire néanmoins fortement pour ce qui concerne l'acquisition d'une image. Ainsi, pour une caméra, le diaphragme joue le rôle de l'iris et de la pupille et détermine la quantité de lumière qui pourra être enregistrée sur un intervalle de temps donné. Par la suite, la lentille joue un rôle équivalent à la cornée et au cristallin, en faisant converger les rayons lumineux vers la rétine sur laquelle sont disséminés environ 125 millions de photorécepteurs : c'est au niveau de ces derniers que l'acquisition est véritablement effectuée. On distingue parmi les photorécepteurs :

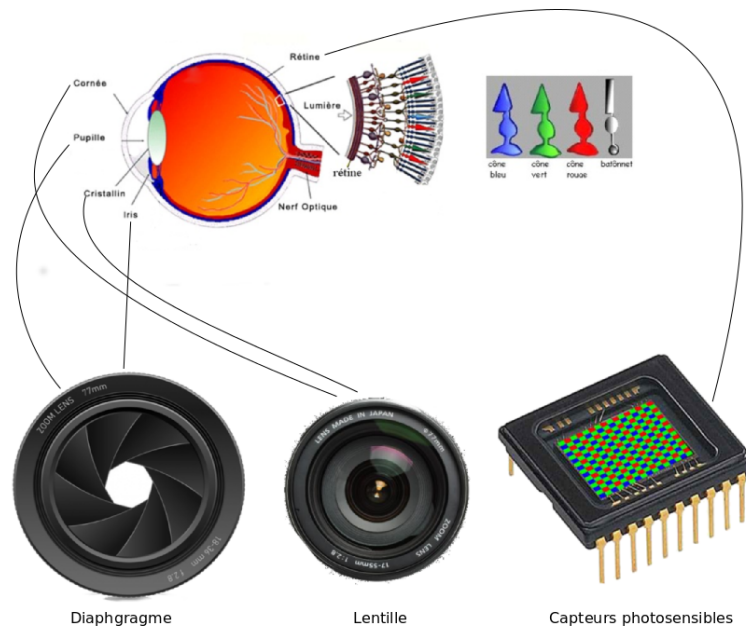


Figure 1.1: Oeil humain vs vision artificielle

- les cônes, généralement impliqués dans la vision diurne. Les cônes présentent 3 types de pigments leur permettant de réagir à des longueurs d'ondes spécifiques (qui ne recouvrent pas exactement le triplet RGB traditionnellement utilisé en traitement d'image)
- les bâtonnets, impliqués dans la vision nocturne. Ne possédant qu'un seul type de pigments, ils ne peuvent pas discriminer les longueurs d'ondes. En revanche, ils sont en moyenne 1000 fois plus sensibles à la lumière que les cônes, et leur population correspond à 20 fois celle des cônes.

Afin de simuler l'activité des photorécepteurs de l'oeil humain, les dispositifs technologiques les plus récents adoptent des stratégies basées sur l'utilisation de filtres placés en amont des capteurs photosensibles, permettant ainsi une acquisition en séquence (un même récepteur recevra successivement les réponses des filtres correspondants aux longueurs d'ondes distinguées) ou simultanée (les réponses sont envoyées sur des capteurs photosensibles dédiés).

L'information contenue dans les données ainsi recueillies est riche et multiple : elle peut-être de nature colorimétrique, géométrique, elle permet de caractériser des déplacements, des déformations. Toutefois, ce qui est vu n'est jamais qu'une représentation de ce qui est observé : il est dès lors fondamental d'exploiter les données recueillies de manière à tendre vers une représentation exploitable de la scène initiale projetée sur les capteurs.

Une première option serait de tenter de reconstruire la scène le plus fidèlement possible. On pourra dans ce cas privilégier l'utilisation de caméras stéréos (Fig.1.2a) [4], ou encore de capteurs RGB-D (Fig.1.2b) [25]. A l'aide de caméras stéréos, nous pouvons percevoir la scène selon plusieurs perspectives, tout comme avec la vision binoculaire. En croisant les données obtenues à partir des différents

points de vue, il est possible d'approcher une représentation tri-dimensionnelle de la scène acquise au sein de l'image. Quant aux capteurs RGB-D, ils complètent les données acquises grâce à une ou plusieurs caméras avec de dispositifs permettant de recueillir une information sur la profondeur. La fusion des données géométriques et colorimétriques obtenues par les caméras classiques et de la localisation tridimensionnelle de points permet une reconstruction 3D de la scène observable. En multipliant les points de vue (soit par le mouvement, soit en utilisant plusieurs dispositifs), ou en exploitant un modèle connu de la scène observée, on sera en mesure d'obtenir une représentation fidèle d'un environnement. Ce sont toutefois des dispositifs onéreux, qui peuvent sembler inutilement intrusifs selon le contexte d'utilisation, et requérant bien souvent une puissance de calcul démesurée par rapport aux informations dont on a besoin.



(a) Jean-Luc Godard expérimentant avec la stéréographie pour tourner *Adieu au langage 3D* (b) La Kinect de @Microsoft qui a permis la démocratisation de l'utilisation des capteurs RGB-D

Figure 1.2: Exemples de caméras stéréo et de capteurs RGB-D.

Une seconde option est de déterminer les transformations géométriques d'objets dans le plan image générées par une modification soit d'une partie de la scène (objet en mouvement), soit de la pose de la caméra elle-même, voire des deux simultanément. Ceci permet entre autres de s'abstraire en partie au moins de la représentation tridimensionnelle de la scène projetée sur l'image, en utilisant par exemple la géométrie plückerienne [2], des descripteurs locaux ou globaux [3], des représentations fréquentielles (Fourier [6], ondelettes [24]), \dots (on pourra retrouver les principales techniques dans [19]).

Le choix d'un type de capteur n'est donc pas anodin, il dépend de paramètres aussi divers que la quantité et la densité des informations, les connaissances préalables que l'on peut avoir d'une scène ou d'un objet, du type d'informations disponibles dans l'images et pertinentes par rapport au contexte (scènes homogènes ou texturées, mobiles ou statiques, diurnes ou nocturnes, \dots).

Nous partirons du principe que nous utilisons une seule caméra simple, ce choix étant justifié dans la dernière section de ce chapitre. L'objectif est à présent de définir le modèle de représentation que nous avons utilisé, soit la manière dont la scène est projetée dans l'image.

1.1.2 De la scène à l'image

Modèle de projection

Nous avons privilégié le modèle *pinhole* qui offre une approximation fiable des caméras perspectives (telles que celles que nous avons utilisées) tout en restant formellement simple [15] [16], sous les hypothèses du respect des conditions de Gauss (angles de faibles incidences), et – c'est notre cas – d'une absence de distorsion de la caméra.

Soient $\mathbf{P} = (X, Y, Z)$ les coordonnées 3-D d'un point dans l'espace et $\mathbf{p} = (x_m, y_m)$ ses coordonnées métriques dans l'image. Le modèle de projection perspective consiste en une projection centrale de centre \mathcal{C} , portée par l'axe \mathbf{z}_c représentant l'axe optique de la caméra. On appelle *plan image* le plan se trouvant à distance focale f de la caméra, soit $Z = f$. On définit également un point de référence $\mathbf{c}(x_c, y_c)$ dans le plan image comme étant le point d'intersection de l'axe optique et du plan image (Fig.1.3).

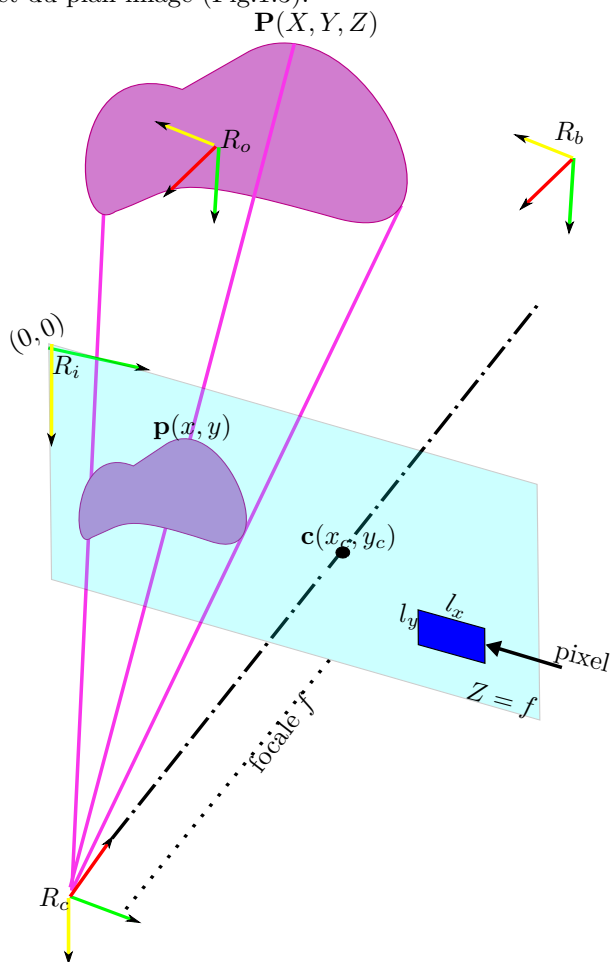


Figure 1.3: Modèle de projection *pinhole*

En plus du *référentiel-base* \mathcal{R}_b et du référentiel de l'organe terminal \mathcal{R}_e (cf. Section ??) s'ajoutent un *référentiel-caméra* noté \mathcal{R}_c , un *référentiel-objet* \mathcal{R}_o ,

ainsi qu'un *référentiel-image* \mathcal{R}_i . Nous noterons également l_x et l_y respectivement la longueur et la hauteur d'un pixel dont nous aurons besoin pour la suite.

A partir des coordonnées d'un point $\mathbf{P}(X, Y, Z)$ exprimées dans le référentiel de la caméra, les coordonnées projetées sur le plan images sont déduites de la manière suivante :

$$x_m = f \frac{X}{Z}, y_m = f \frac{Y}{Z} \quad (1.1)$$

ou, sous forme matricielle :

$$Z \begin{bmatrix} x_m \\ y_m \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (1.2)$$

soit $\tilde{\mathbf{p}} = \mathbf{A}\mathbf{P}$, avec $\tilde{\mathbf{p}} = Z\mathbf{p}$.

Un point dans l'image est généralement représenté par ses coordonnées pixelliques (l'origine du référentiel étant supposée localisée en haut à gauche de l'image).

La conversion entre les coordonnées métriques (x_m, y_m) et les coordonnées pixelliques (x_p, y_p) , se fait selon la relation suivante :

$$\begin{cases} x_p = x_c + x_m/l_x \\ y_p = y_c + y_m/l_y \end{cases} \quad (1.3)$$

ce qui donne sous une forme matricielle :

$$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{l_x} & 0 & x_c \\ 0 & \frac{1}{l_y} & y_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (1.4)$$

soit $\mathbf{p}_p = \mathbf{B}\mathbf{p}$.

On obtient la relation suivante entre les coordonnées métriques 3-D du point et les coordonnées pixelliques dans l'image :

$$Z \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & 0 & x_c \\ 0 & \alpha_y & y_c \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (1.5)$$

avec $\alpha_x = f/l_x$ et $\alpha_y = f/l_y$. Les paramètres $(\alpha_x, \alpha_y, x_c, y_c)$ de la matrice $\mathbf{K} = \mathbf{A}\mathbf{B}$ ainsi construite sont généralement obtenus par calibration [5], [27], [26]. La matrice \mathbf{K} nous permet de faire le lien entre les informations géométriques disponibles dans l'image et la projection de la scène observée. En particulier, nous pouvons à présent définir les coordonnées normalisées ainsi :

$$\begin{cases} x = X/Z \\ y = Y/Z \end{cases} \quad (1.6)$$

indépendantes de la focale, et pouvant être déduites des coordonnées pixelliques en utilisant la matrice \mathbf{K} .

Enfin, les coordonnées 3D du point étant exprimées dans le repère caméra, nous voulons pouvoir les traduire dans le référentiel propre de l'objet.

Changements de repère

Soient à présent ${}^c\mathbf{P} = ({}^cX, {}^cY, {}^cZ)$ les coordonnées du point \mathbf{P} exprimées dans le référentiel \mathcal{R}_c et ${}^o\mathbf{P} = ({}^oX, {}^oY, {}^oZ)$ ses coordonnées dans le référentiel \mathcal{R}_o . Le passage d'un référentiel à l'autre est effectué en utilisant la relation ${}^c\mathbf{P} = {}^c\mathbf{t}_o + {}^c\mathbf{R}_o {}^o\mathbf{P}$, avec ${}^c\mathbf{t}_o$ le vecteur 3×1 de translation entre les deux référentiels, ${}^c\mathbf{R}_o$ la matrice 3×3 de rotation correspondant à la rotation autour des axes du référentiel. En utilisant les coordonnées homogènes $\tilde{\mathbf{P}}$ de \mathbf{P} , il est possible de factoriser cette transformation en une seule opération matricielle :

$${}^c\tilde{\mathbf{P}} = {}^c\mathbf{M}_o {}^o\tilde{\mathbf{P}} \quad (1.7)$$

soit :

$$\begin{bmatrix} {}^cX \\ {}^cY \\ {}^cZ \\ 1 \end{bmatrix} = \begin{bmatrix} {}^c\mathbf{R}_o & {}^c\mathbf{t}_o \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} {}^oX \\ {}^oY \\ {}^oZ \\ 1 \end{bmatrix} \quad (1.8)$$

On appelle ${}^c\mathbf{M}_o$ la *matrice homogène de transformation* du référentiel \mathcal{R}_o au référentiel \mathcal{R}_c , dont l'inverse s'exprime simplement sous la forme :

$${}^o\mathbf{M}_c = \begin{bmatrix} {}^c\mathbf{R}_o^T & -{}^c\mathbf{R}_o^T {}^c\mathbf{t}_o \\ \mathbf{0} & 1 \end{bmatrix} \quad (1.9)$$

Nous sommes dès lors en mesure d'établir une relation entre la scène observée dans son référentiel propre, dans le référentiel de la caméra (et par extension dans chacun des référentiels du manipulateur) et sa projection dans l'image, exprimée en coordonnées pixelliques, métriques et normalisées. Le lecteur remarquera cependant que, s'il est possible directement, connaissant \mathbf{K} et ${}^o\mathbf{M}_c$, de déduire à partir des propriétés de la scène les caractéristiques de sa projection, l'opération inverse est dépendante dans ce modèle d'une estimation pour chaque point de sa profondeur (coordonnée cZ), ce dont nous aurons à tenir compte lorsqu'il s'agira de déterminer le type de mesure que nous utiliserons pour déduire les consignes données au manipulateur.

1.2 Configurations et stratégies d'asservissement visuel

1.2.1 Positionnement de la caméra

Le contexte étant celui d'un manipulateur dont on veut asservir la saisie et le déplacement d'un objet, plusieurs stratégies s'offrent à nous. Une première décision à prendre concerne par exemple le positionnement de la caméra. On distingue ainsi une configuration déportée (Fig.1.4a) d'une configuration embarquée (Fig. 1.4b).

Dans le cas d'une caméra déportée, on connaît les transformations permettant de passer du référentiel-base \mathcal{R}_b au référentiel-caméra \mathcal{R}_c , et du référentiel-caméra au référentiel-objet \mathcal{R}_o . Seul l'organe terminal bouge dans la scène observée. Dans le cas d'une caméra embarquée, la transformation permettant de passer du référentiel de l'organe terminal \mathcal{R}_e au référentiel-caméra \mathcal{R}_c est connue. Toutefois, l'ensemble de la scène est affecté lors d'un déplacement de l'organe terminal (et donc de la caméra).

1.2. CONFIGURATIONS ET STRATÉGIES D'ASSERVISSEMENT VISUEL7

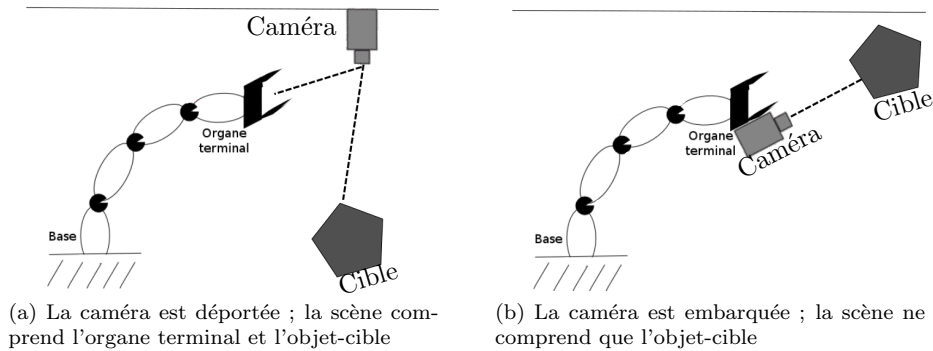


Figure 1.4: Positionnements de caméra

1.2.2 Asservissements en position, basés images et hybrides

De manière générale, une stratégie d'asservissement visuel peut être décomposée de la manière suivante :

1. on sélectionne dans un premier temps un ensemble \mathbf{s} de paramètres à partir desquels on peut estimer soit directement la pose de l'organe terminal, soit le déplacement nécessaire à l'obtention de cette pose. L'ensemble des paramètres dépend d'une part de la valeur des primitives extraites de l'image et d'autre part d'un ensemble de connaissances *a priori*, telles que les propriétés de la caméra, ou encore un modèle de la cible.
2. on définit ensuite la valeur désirée \mathbf{s}^* des paramètres, correspondant à l'état que l'on souhaite atteindre.
3. à partir de données extraites de l'image, on calcule la valeur courante des paramètres
4. on calcule alors une erreur $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$, dont on souhaite qu'elle converge vers 0.
5. on établit enfin une relation entre la variation $\dot{\mathbf{e}}$ de l'erreur et la commande qui sera ensuite envoyée au manipulateur.
6. la commande est effectuée, et l'on boucle les étapes 3 à 6 jusqu'à obtention d'une valeur seuil de l'écart entre les valeurs courantes des paramètres et les valeurs désirées.

On distinguera ici deux grandes classes de stratégies d'asservissement : la première est basée sur l'estimation des paramètres de pose de l'organe terminal (PBVS : position-based visual servo), tandis que la seconde utilise directement les primitives extraites de l'image (IBPS : image-based visual servo). Il est également possible de mixer ces deux approches, ce que l'on appellera un asservissement hybride.

Dans le cas de l'asservissement en position, une connaissance du modèle 3-D de la cible est nécessaire (on parle d'ailleurs parfois d'asservissement 3-D). De l'écart entre les valeurs courantes d'un ensemble de primitives et leurs valeurs

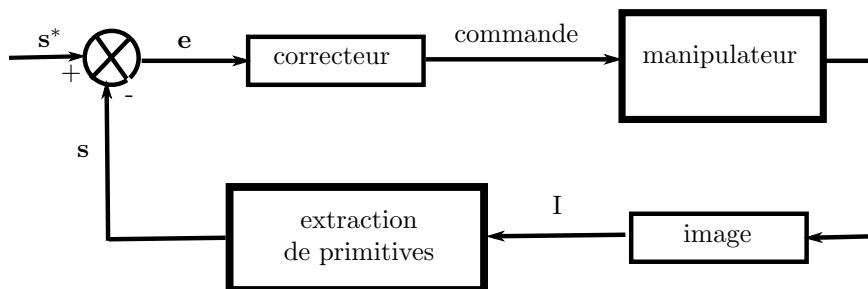


Figure 1.5: Schéma des étapes de l'asservissement visuel

estimées pour une pose désirée, on déduira les paramètres de pose de l'organe terminal. Cela revient à faire varier un référentiel courant de l'organe terminal \mathcal{R}_e vers un référentiel correspondant à la pose prescrite \mathcal{R}_e^* , soit la détermination d'une matrice homogène de transformation ${}^e\mathbf{M}_e$. Comme nous l'avons vu, cette matrice est composée d'une translation et d'une rotation, qui peuvent être décrites à l'aide de 6 paramètres. Dans ce cas, le vecteur \mathbf{s} est constitué de mesures permettant de construire cette matrice de transformation de repère. On pourra d'ailleurs noter que dans une configuration avec caméra embarquée, le problème est équivalent à la détermination d'une matrice de transformation ${}^c\mathbf{M}_c$ entre les référentiels courants et désirés de la caméra.

L'asservissement basé-image repose quant à lui sur l'exploitation direct des informations extraites de l'image pour établir la commande qui sera envoyée au manipulateur. Il ne nécessite pas de connaissance *a priori* du modèle 3-D de l'objet, mais exploite des primitives pour lesquelles on possède un modèle de comportement lorsqu'elles évoluent dans l'espace. On parlera dès lors d'asservissement 2-D, ou d'asservissement 2-D 1/2 lorsque une connaissance du modèle 3-D est néanmoins utilisée (approche hybride).

Quelque-soit la situation, l'objectif de l'asservissement visuel est de traduire un ensemble de mesures effectuées dans une série d'image en loi de commande qui sera envoyée au manipulateur.

1.2.3 Construction d'une loi de commande

Nous reprenons ici les bases de construction d'une loi de commande [14]. Le lecteur les retrouvera développées par exemple dans [7] et [8]. Nous nous contentons ici d'exposer les principes généraux d'outils que nous avons utilisés tout au long de nos travaux.

Soient \mathbf{s} un ensemble de mesures réalisées au sein de l'image sur des primitives choisies, et \mathbf{s}^* les valeurs désirées de ces mesures. On construit le vecteur $\mathbf{e} = \mathbf{s} - \mathbf{s}^*$ représentant l'erreur de mesure, soit la différence entre les valeurs courantes et les valeurs désirées.

On choisit ici de construire une loi de commande en vitesse. Soit le torseur cinématique $\mathbf{v}_c = (\nu_c, \omega_c)$, avec ν_c et ω_c les vitesses instantanées respectivement linéaires et angulaires. Il est nécessaire dans un premier temps de déterminer les variations $\dot{\mathbf{s}}$ des mesures pour un déplacement donné \mathbf{v}_c de la caméra. Cette relation, lorsqu'elle existe, prend la forme algébrique d'une matrice $\mathbf{L}_s \in \mathcal{M}_{k \times 6}$ que l'on appelle *matrice d'interaction* [13], k représentant le nombre de mesures

1.3. CONTEXTE DE L'ÉTUDE ET CHOIX DE STRATÉGIE POUR L'ASSERVISSEMENT VISUEL 9

effectuées. A partir de tous ces éléments nous pouvons établir :

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v}_c \quad (1.10)$$

Notre cible étant fixe, nous avons $\dot{\mathbf{e}} = \dot{\mathbf{s}}$. Nous pouvons dès lors immédiatement déduire de (Equ.1.10) la relation suivante :

$$\dot{\mathbf{e}} = \mathbf{L}_e \mathbf{v}_c \quad (1.11)$$

Nous souhaitons que notre erreur \mathbf{e} de mesure décroisse de manière exponentielle, soit : $\dot{\mathbf{e}} = -\lambda \mathbf{e}$, λ étant appelé le *gain*, permettant de régler la vitesse de convergence. La loi de commande suivante peut alors être construite :

$$\mathbf{v}_c = -\lambda \mathbf{L}_e^+ \mathbf{e} \quad (1.12)$$

où $\mathbf{L}_e^+ \in \mathcal{M}_{6 \times k}$ représente la matrice pseudo-inverse de Moore-Penrose, soit $\mathbf{L}_e^+ = (\mathbf{L}_e^T \mathbf{L}_e)^{-1} \mathbf{L}_e^T$. Lorsque $k = 6$ et que $\det \mathbf{L}_e \neq 0$, on pourra choisir d'utiliser la loi $\mathbf{v}_c = -\lambda \mathbf{L}_e^{-1} \mathbf{e}$.

En pratique, il est difficile de connaître avec exactitude \mathbf{L}_e , qui peut être dépendante de la pose de la caméra. On utilisera dès lors une approximation de la matrice d'interaction, la loi de commande finale devenant :

$$\mathbf{v}_c = -\lambda \widehat{\mathbf{L}_e^+} \mathbf{e} \quad (1.13)$$

Toutes les fois où cela est possible, on préférera utiliser des primitives dont la matrice d'interaction correspondante exhibe des propriétés algébriques intéressantes (triangulaire, inversible, ...). Le choix préalable des primitives obéit donc à un double impératif :

- on attend qu'elles soient aisément identifiables dans l'image et que les mesures puissent être réalisées de manière robuste
- les caractéristiques de la matrice d'interaction qu'elles complètent doivent comporter des qualités algébriques qui permettront d'améliorer les calculs tout autant que de ne pas propager les erreurs de mesures sur une primitive particulière aux autres mesures.

1.3 Contexte de l'étude et choix de stratégie pour l'asservissement visuel

Nous avons vu que l'utilisation d'un robot parallèle à câbles nous permet d'évoluer dans un espace de travail conséquent. En particulier, le robot **Marionet-Assist** que nous avons utilisé pour nos expérimentations évolue dans un cube de $4m \times 3m \times 3m$, et le volume atteignable par un robot de type **Marionet-Crane** peut aller jusqu'à $100m \times 35m \times 35m$. Dans ces circonstances, l'utilisation d'une caméra embarquée a été privilégiée pour deux raisons :

- le champs de vision d'une seule caméra fixe déportée pourra se révéler insuffisant pour couvrir l'ensemble de l'espace de travail
- le déploiement d'un robot à câble pouvant être effectué dans des milieux nocifs, ou difficilement atteignables (suite par exemple à une catastrophe naturelle), l'utilisation d'une caméra déportée est souvent difficilement envisageable.

Ajoutons que le robot **Marionet-Assist** a été pensé pour être utilisé dans des pièces de vie de personnes à la mobilité fragilisée. Dans ce cadre, une caméra déportée peut paraître inutilement intrusive. Le champs de vision d’une caméra embarquée étant limité, cela nous a semblé, conformément à cette application, un choix plus respectueux de l’intimité des personnes.

De la même manière, les environnements d’utilisation des robots à câbles de la famille **Marionet** sont généralement dynamiques. A ce titre, nous avons privilégié une stratégie d’asservissement basé-image, à nouveau pour deux raisons distinctes :

- elle ne nécessite pas une connaissance complète du modèle 3-D de la cible
- l’asservissement en position repose souvent sur l’utilisation de plusieurs caméras (ou de capteurs de type RGB-D) pour garantir une mesure fiable des primitives permettant une reconstruction du modèle 3-D de la cible. Or, pour des raisons toujours liées au contexte d’application des robots de la famille **Marionet**, cela ne correspond pas aux choix de capteurs que nous avons effectués

Enfin, il était envisageable dans cette étude d’utiliser les caméras soit comme capteur intéroceptif, soit comme capteur extéroceptif. Le choix d’une observation des articulations d’un robot parallèle à câbles a été effectué par [1] pour des robots parallèles classiques, et plus récemment pour des robots parallèles à câble [11], [10]. Ces études correspondant à ce jour au principal travail effectué dans le domaine de l’asservissement visuel des robots parallèles à câbles, nous reviendrons dès lors dessus à l’occasion de la section suivante, au sein de laquelle nous définirons nos problématiques et pistes de travail. Nous pouvons relever toutefois que la démarche développée dans leurs recherches consiste à améliorer le positionnement de l’organe terminal, ce qui suppose que la position de la cible est connue avec suffisamment de précision. Il s’agit donc d’un environnement contrôlé. Or, le contexte d’application de nos robots implique au contraire une incertitude non-négligeable sur les propriétés de la cible. Dès lors, la caméra est tout autant utilisée pour la construction d’une loi de commande que pour la localisation de l’objet et l’actualisation de ses propriétés lorsque celles-ci diffèrent d’une première estimation. Le choix d’utilisation des caméras comme capteurs extéroceptifs s’est donc imposé dans notre approche.

Il nous reste en introduction à présenter les avantages et les spécificités d’une utilisation de l’asservissement visuel dans le contexte particulier des robots parallèles à câbles. La section suivante introduira les pistes d’amélioration de contrôle que nous avons repérées, les problématiques auxquelles nous avons été confrontées et la manière dont nous proposons de les résoudre. Nous évoquerons également de quelle manière les spécificités des robots parallèles nous permettent d’assouplir les contraintes de l’asservissement tout en garantissant un contrôle optimal.

1.4 Asservissement visuel des robots parallèles à câbles

Nous avons vu que plusieurs caractéristiques des robots parallèles à câbles peuvent altérer la qualité de la manipulation. Ainsi, la complexité des modèles

géométriques et cinématiques par rapport aux robots séries et aux robots parallèles rigides menace d'une part la précision des opérations de manipulation lors de la résolution numérique, et d'autre part l'effectuation en temps réel. De plus, la nécessaire vérification de l'équilibre statique pour toutes les configurations de câbles en tension strictement positive impacte également la contrainte de réalisation en temps réel d'une tâche ; surtout, il est tout autant difficile de prévoir quels câbles seront en tension lors d'un déplacement prescrit, que de garantir la constance du caractère positif ou nul de la tension d'un câble sur l'intégralité d'un déplacement. Enfin, les incertitudes mécaniques – telles par exemple le diamètre réel de la couche d'enroulement d'un câble autour des tambours – ont également une influence sur la précision du manipulateur.

Il est donc nécessaire, afin de garantir l'efficacité d'une tâche de manipulation, de simplifier la résolution numérique des modèles, de développer une stratégie de contrôle de la tension dans les différents câbles, et de corriger enfin la trajectoire du manipulateur lorsque les incertitudes et erreurs perturbent celle-ci.

L'utilisation de capteurs supplémentaires a été proposée à de nombreuses reprises afin de résoudre un ou plusieurs des problèmes énoncés. Dans le contexte des robots parallèles à jambes rigides, plusieurs suggestions ont été faites :

- mesurer directement la pose de l'organe terminal à l'aide de systèmes de lasers et miroirs [18], [17] ou d'une caméra déportée [9], [21] : cela permet tout autant une simplification des modèles qu'une correction des erreurs et imprécisions.
- utiliser des capteurs proprioceptifs afin d'obtenir une représentation fidèle de l'état des variables articulaires [20], [22], ce qui permet également une simplification des modèles pouvant aboutir à une exécution en temps réel.
- un contrôle référencé vision basé sur l'exploitation de mesures des directions des jambes du robot [1] : dans cette dernière méthode, une stratégie IBVS est utilisée pour définir un contrôle cinématique. Elle permet de s'affranchir du calcul du MGD, ce qui la rend *a priori* robuste aux erreurs de modèles et de calibrations.

En particulier, cette dernière méthode a été déclinée récemment pour le contrôle de robots parallèles à câbles. Elle fait suite à une première approche [12] consistant à mettre dans un premier temps en place un asservissement visuel 3-D (PBVS) sur l'organe terminal, puis à élaborer un schéma de contrôle dynamique utilisant la vision pour estimer la pose et la vitesse de la plate-forme. Malgré des résultats prometteurs obtenus en simulation, il reste à valider cette approche sur un prototype réel.

Les mêmes auteurs ont donc par la suite proposé une approche analogue au suivi de jambes des robots parallèles classiques [10], en exploitant cette fois-ci les directions de départ des câbles, complétant ainsi un système de mesures utilisant des capteurs de forces (afin d'estimer la tension dans les câbles), ainsi que plusieurs caméras déportées pour estimer la pose de la plate-forme. Le schéma d'ensemble utilise donc un premier ensemble de quatre caméras filmant un motif sur l'organe terminal pour proposer un asservissement 3-D donnant le torseur cinématique de l'organe terminal, puis quatre caméras stéréos donnant la direction tangente aux départs de câbles, permettant ainsi de simplifier le contrôle cinématique du robot. A nouveau, les résultats présentés en simulation nous

paraissent tout-à-fait prometteurs, mais doivent encore à notre connaissance être validés expérimentalement.

Nous avons toutefois choisi de ne pas poursuivre dans la même voie pour les trois raisons suivantes :

- pour des raisons d'intrusivité et de coût propres aux conditions d'applications des prototypes sur lesquels nous avons travaillé, le dispositif de mesure nous a paru trop conséquent ; nous avons de plus privilégié des configurations embarquées comme mentionné précédemment.
- bien qu'ils fassent durant leurs travaux l'hypothèse de câbles non-élastiques, les auteurs supposent une précision des mesures de tension qui nous semble difficilement atteignable pour des robots de cette envergure. Il ne s'agissait évidemment pas du sujet de leur étude, mais nous pensons néanmoins que certaines situations que nous avons rencontrées pourraient affecter la qualité des résultats annoncés.
- enfin, le schéma de contrôle repose sur l'estimation et l'inversion de la matrice d'interaction d'une part, et sur l'estimation de la Jacobienne inverse d'autre part, se présentant donc sous la forme :

$$\dot{\mathbf{q}} = -\lambda \widehat{\mathbf{J}}^{-1} \widehat{\mathbf{L}}_e^+ \mathbf{e} \quad (1.14)$$

Nous pensons qu'il est possible d'unifier ces deux étapes à la condition d'exécuter un contrôle cinématique à l'aide d'une caméra embarquée, soit une seule matrice :

$$\dot{\mathbf{q}} = -\lambda \widehat{\mathbf{K}} \mathbf{e} \quad (1.15)$$

ne nécessitant pas plus de connaissance précise de la pose X que des valeurs des variables articulaires ρ .

Bien que nous ne puissions garantir que cette simplification du schéma de contrôle puisse s'appliquer à toutes les configurations de robots à câbles, elle s'avère efficace dans notre contexte, particulièrement pour les robots $N - 1$. Sa mise en place implique toutefois certaines conditions qui peuvent se révéler incompatibles avec un ensemble d'applications, ce que nous exposerons lors du chapitre qui sera consacré à son étude.

Le travail qui est présenté ici a pour objectif d'améliorer la qualité de manipulation d'un robot parallèle à câbles. Pour cela, nous avons distingué quatre critères qui nous permettront d'évaluer notre contribution, à savoir :

1. la **stabilité** de la plate-forme lors d'un déplacement
2. la **précision** de la pose finale
3. la **robustesse** aux imprécisions et erreurs de modèle, aux erreurs et imprécisions des estimations des paramètres de contrôle
4. la **simplicité** du schéma de contrôle

Dans ce but, nos travaux seront présentés de la manière suivante :

- dans un premier temps, nous présenterons une stratégie de contrôle des câbles en tension, qui nous permettra d'améliorer la stabilité des mouvements, ainsi que la précision des déplacements.

1.4. ASSERVISSEMENT VISUEL DES ROBOTS PARALLÈLES À CÂBLES¹³

- une fois cette stabilité assurée, nous montrerons que l'utilisation d'un asservissement visuel améliore la précision du positionnement de l'organe terminal, mais également la robustesse du contrôle
- nous utiliserons ensuite les spécificités des architectures parallèles pour développer une loi de contrôle simplifiée, ce qui nous permettra de gagner ici en robustesse et en simplicité de calcul.
- enfin, un chapitre présentera plusieurs simulations et expérimentations sur notre prototype. Ceci nous permettra de valider nos différentes approches, mais également d'en présenter les limites, ce qui nous permettra en conclusion de faire le point sur les perspectives s'ouvrant en continuité de ces travaux.

Bibliography

- [1] Nicolas Andreff, Tej Dallej, and Philippe Martinet. Image-based visual servoing of a Gough-Stewart parallel manipulator using leg observations. *Int. J. Rob. Res.*, 26(7):677–687, July 2007.
- [2] Nicolas Andreff, Bernard Espiau, and Radu Horaud. Visual Servoing from Lines. Research Report RR-4226, INRIA, 2001.
- [3] La Tuan Anh and Jae-Bok Song. Object tracking and visual servoing using features computed from local feature descriptor. In *Control Automation and Systems (ICCAS), 2010 International Conference on*, pages 1044–1048, Oct 2010.
- [4] V. Brandou, E. Malis, Patrick Rives, A-G Allais, and M. Perrier. Active stereovision using invariant visual servoing. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 2326–2331, Oct 2006.
- [5] Duane C. Brown. Close-range camera calibration. *PHOTOGRAMMETRIC ENGINEERING*, 37(8):855–866, 1971.
- [6] V. Chari, A Sharma, A Namboodiri, and C.V. Jawahar. Frequency domain visual servoing using planar contours. In *Computer Vision, Graphics Image Processing, 2008. ICVGIP '08. Sixth Indian Conference on*, pages 87–94, Dec 2008.
- [7] François Chaumette and S. Hutchinson. Visual servo control, Part I: Basic approaches. *IEEE Robotics and Automation Magazine*, 13(4):82–90, 2006.
- [8] François Chaumette and S. Hutchinson. Visual servo control, Part II: Advanced approaches. *IEEE Robotics and Automation Magazine*, 14(1):109–118, 2007.
- [9] T. Dallej, N. Andreff, Y. Mezouar, and P. Martinet. 3d pose visual servoing relieves parallel robot control from joint sensing. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 4291–4296, Oct 2006.
- [10] T. Dallej, M. Gouttefarde, N. Andreff, R. Dahmouche, and P. Martinet. Vision-based modeling and control of large-dimension cable-driven parallel robots. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 1581–1586, Oct 2012.

- [11] T. Dallej, M. Gouttefarde, N. Andreff, M. Michelin, and P. Martinet. Towards vision-based control of cable-driven parallel robots. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 2855–2860, Sept 2011.
- [12] Tej Dallej, Marc Gouttefarde, Nicolas Andreff, Micaël Michelin, and Philippe Martinet. Towards vision-based control of cable-driven parallel robots. In *IROS*, pages 2855–2860. IEEE, 2011.
- [13] B. Espiau, F. Chaumette, and Patrick Rives. A new approach to visual servoing in robotics. *Robotics and Automation, IEEE Transactions on*, 8(3):313–326, Jun 1992.
- [14] P.J. Esrom. Robot control (the task function approach), by c. samson, m. le borgne and b. espiau oxford university press, oxford, 1991, xvii + 364, references and index (£50.00). *Robotica*, 9:447–448, 12 1991.
- [15] Olivier Faugeras. *Three-dimensional Computer Vision: A Geometric Viewpoint*. MIT Press, Cambridge, MA, USA, 1993.
- [16] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.
- [17] Theo A. G. Heeren and Frans E. Veldpaus. An optical system to measure the end effector position for on-line control purposes. *Int. J. Rob. Res.*, 11(1):53–63, February 1992.
- [18] W.F. Marantette. Machine tool position measurement employing multiple laser distance measurements, February 7 1995. US Patent 5,387,969.
- [19] E. Marchand and F. Chaumette. Feature tracking for visual servoing purposes. *Robotics and Autonomous Systems*, 52(1):53–70, July 2005.
- [20] Jean-Pierre Merlet. Closed-form resolution of the direct kinematics of parallel manipulators using extra sensors data. In *Proceedings of the 1993 IEEE International Conference on Robotics and Automation, Atlanta, Georgia, USA, May 1993*, pages 200–204, 1993.
- [21] Flavien Paccot, Philippe Lemoine, Nicolas Andreff, Damien Chablat, and Philippe Martinet. A vision-based computed torque control for parallel kinematic machines. *CoRR*, abs/0810.2666, 2008.
- [22] V. Parenti-Castelli and R. Di Gregorio. A new algorithm based on two extra sensors for real-time computation of the actual configuration of the generalized stewart-gough manipulator. *ASME J. of Mechanical Design*, 122(3):294–298, september 2000.
- [23] D. Purves. *Neuroscience*. Sinauer Associates Incorporated, 2004.
- [24] Luis E. Ramos-Velasco. Visual servoing using discrete wavelet transform. In *World Automation Congress (WAC), 2012*, pages 1–6, June 2012.

- [25] I Siradjuddin, L. Behera, T.M. McGinnity, and S. Coleman. A position based visual tracking system for a 7 dof robot manipulator using a kinect camera. In *Neural Networks (IJCNN), The 2012 International Joint Conference on*, pages 1–7, June 2012.
- [26] G. P. Stein. Lens distortion calibration using point correspondences. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, CVPR '97, pages 602–, Washington, DC, USA, 1997. IEEE Computer Society.
- [27] Roger Y. Tsai. An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Miami Beach, FL*, pages 364–374, <http://www-cgi.cs.cmu.edu/afs/cs.cmu.edu/user/rgw/www/TsaiCode.html>, 1986.