



**HAL**  
open science

# Estimation de pose et asservissement de robot par vision omnidirectionnelle

Guillaume Caron

► **To cite this version:**

Guillaume Caron. Estimation de pose et asservissement de robot par vision omnidirectionnelle. Automatique / Robotique. Université de Picardie Jules Verne, 2010. Français. NNT: . tel-00577133

**HAL Id: tel-00577133**

**<https://theses.hal.science/tel-00577133>**

Submitted on 16 Mar 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ DE PICARDIE JULES VERNE  
ÉCOLE DOCTORALE  
SCIENCES ET SANTÉ

# THÈSE

pour obtenir le titre de

**Docteur en Sciences**

de l'Université de Picardie Jules Verne

**Mention : AUTOMATIQUE**

Présentée par

Guillaume CARON

## Estimation de pose et asservissement de robot par vision omnidirectionnelle

Thèse dirigée par El Mustapha MOUADDIB

préparée à l'UPJV au sein du laboratoire  
MODÉLISATION, INFORMATION ET SYSTÈMES

soutenue le 30 novembre 2010

### Jury :

<i>Rapporteurs :</i>	Nicolas ANDREFF	- Université de Franche-Comté
	Michel DEVY	- LAAS/CNRS
<i>Directeur :</i>	El Mustapha MOUADDIB	- Université de Picardie Jules Verne
<i>Président :</i>	Michel DE MATHELIN	- Université de Strasbourg
<i>Examineurs :</i>	Driss ABOUTAJDINE	- Université Mohammed V de Rabat
	Ouiddad LABBANI-IGBIDA	- Université de Picardie Jules Verne
	Eric MARCHAND	- Université de Rennes 1
	Claude PÉGARD	- Université de Picardie Jules Verne



## Remerciements

Ces trois années de préparation de thèse de doctorat ont été jalonnées de discussions, de travaux, de rencontres et de collaborations scientifiques très intéressantes pour moi et je tiens à remercier les personnes m'ayant suivi, accompagné et soutenu tout au long des recherches m'ayant permis d'élaborer cette thèse.

Je pense en premier lieu à El Mustapha Mouaddib, mon directeur de thèse qui a su m'aider à m'orienter vers le bon choix de faire de la recherche et de préparer une thèse, ainsi qu'à me guider durant les trois ans de sa préparation. Au delà du cadre même de la thèse, je pense aussi aux quelques manifestations auxquelles nous avons participé ensemble et avec d'autres collègues comme la fête de la science, des séances de découverte de la robotique pour nos cadets, entre autres, et bien entendu aux quelques discussions scientifiques ou non que nous avons eu autour d'un verre, près du laboratoire ou sur le lieu de conférences et qui sont pour moi de bons souvenirs de cette période. En définitive, cela me fait sourire de me retrouver "collègue" d'El Mustapha après avoir été l'un de ses étudiants à l'IUT Informatique d'Amiens, il y a quelques années...

Assez tôt dans la préparation de la thèse, j'ai émis le souhait auprès d'El Mustapha d'effectuer un séjour de recherche d'un mois dans un autre laboratoire et nous nous sommes orientés vers l'équipe INRIA Lagadic de Rennes, très attirante du point de vue scientifique. Lors de ce séjour en novembre 2008, j'ai travaillé avec Eric Marchand et c'est la seconde personne que j'aimerais remercier car ce séjour s'est si bien passé, tant du point de vue scientifique qu'humainement, que la collaboration s'est poursuivie et a donné lieu à un deuxième séjour d'un mois en novembre 2009. Ce qui m'a beaucoup plu, c'est qu'avec El Mustapha et Eric, nous nous sommes naturellement mis à travailler ensemble, sans être dans le cadre d'un projet bien délimité ou pensé de longue date, mais pour la science, en combinant les compétences de l'équipe de recherche d'Amiens et de celle de Rennes, à savoir la vision omnidirectionnelle et les asservissements et suivis visuels. Cette collaboration a d'ailleurs été de premier ordre dans mes travaux puisque la majorité de cette thèse en est le fruit. Je suis heureux d'avoir été au centre de cette collaboration quasi spontanée et me considère chanceux d'avoir pu bénéficier de l'influence d'El Mustapha et d'Eric.

Je souhaite aussi exprimer ma gratitude à Nicolas Andreff et Michel Devy pour avoir accepté d'être rapporteur de cette thèse et pour les commentaires qu'ils ont fait sur mon travail. Je remercie aussi Michel de Mathelin pour avoir accepté de présider le jury ainsi que les examinateurs de ma thèse, à savoir Driss Aboutajdine, qui a fait un long voyage pour assister à ma soutenance de thèse, Ouidad Labbani, qui avait été trois ans plus tôt la co-encadrante de mon stage de Master, Eric Marchand, que j'ai déjà mentionné, et Claude Pégard, qui a été mon tuteur pour le monitorat IES, entre autres.

Bien entendu, d'autres personnes m'ont marqué pendant ces trois ans au laboratoire MIS et j'aimerais particulièrement remercier Cédric Demonceaux qui a toujours été disponible et avec qui j'ai eu de nombreuses discussions scientifiques parfois, et un peu plus décalées à d'autres moments, voire les deux simultanément ! Je pense

aussi à Claude Pégard, partenaire de choix des journées de démonstrations d'applications robotiques de l'équipe, notamment chez Airbus à Méaulte, qui font partie des bons souvenirs que j'ai de ces trois dernières années. Avec Cédric et Claude, je pense aussi à Pascal Vasseur, Ouiddad Labbani, Djemaa Kachi et Alex Potelle sans qui les séminaires ou les pauses café n'auraient pas eu toute leur saveur.

Mais comment ne pas penser à mes collègues et amis doctorants du laboratoire MIS qui ont partagé mes journées pendant trois ans et les remercier pour l'ambiance dans le bureau, les blagues, les potins et les entre-aides ? Je pense bien sûr à Damien Eynard, ancien collègue de DUT Informatique et partenaire du développement de "notre" logiciel d'étalonnage stéréo, à Sang Ly mon partenaire de tennis pendant deux ans, à Pauline Merveilleux et ses fameuses "Paulinades", à Amina Radgui avec qui j'ai eu de nombreuses discussions scientifiques mais aussi Jean-Charles Bazin, Ali Ghorayeb, Ashutosh Natraj et Jean-François Layerle. Je n'oublie pas non plus ceux de la première heure, toutes équipes confondues comme Pascal Vander Swalmen, avec qui nous avons organisé cette fameuse première Journée des Jeunes Chercheurs du MIS en 2010, ou encore Wissam El Messoussi, Nourredine Hemche et Asli Oncel qui ont énormément facilité mon entrée au laboratoire CREA, devenu MIS par la suite, par leur convivialité, leur gentillesse et leur bonne humeur.

J'aimerais aussi remercier Pierre Détaille qui a toujours été disponible pour réaliser divers supports de caméras, montages électroniques et autres, même si je ne savais pas toujours exactement ce qu'il me fallait...

D'autres enseignants chercheurs ont contribué à l'appréciation que j'ai eue du laboratoire MIS comme Dominique Lecllet, Abdelhamid Rabhi, Jérôme Bosch, Dominique Lazure et Florence Levé. Nous avons travaillé ensemble pour organiser quelques événements, pour les enseignements, pour des projets de Master ou nous croisions simplement à l'occasion de pauses café.

Enfin, je pense aussi aux collègues chercheurs ou doctorants de l'équipe Lagadic comme François Chaumette pour m'avoir accueilli au sein de son équipe et pour les quelques discussions que nous avons eues ou les remarques très intéressantes sur mon travail. J'ai déjà mentionné Eric mais je pense aussi aux autres membres de l'équipe et à tous les doctorants pour leur convivialité, aussi bien dans le travail qu'aux "Happy Hours", une véritable institution.

# Table des matières

<b>1</b>	<b>Introduction</b>	<b>5</b>
1.1	Modélisation . . . . .	5
1.1.1	Projection perspective . . . . .	5
1.1.2	Projection stéréographique . . . . .	7
1.1.3	Stéréoscopie . . . . .	9
1.2	L'estimation de pose . . . . .	13
1.2.1	Estimation linéaire de pose . . . . .	13
1.2.2	Estimation non-linéaire de pose . . . . .	15
1.2.3	Asservissement visuel virtuel . . . . .	16
1.2.4	Estimation de pose relative . . . . .	20
1.2.5	Calcul de pose relative et reconstruction 3D simultanés . . . . .	21
1.3	Choix et représentation des primitives . . . . .	23
1.3.1	Lieux de définition et représentations des primitives . . . . .	23
1.3.2	Appariement, pose, reconstruction : vers une simultanité . . . . .	24
<b>2</b>	<b>Asservissement visuel virtuel géométrique</b>	<b>27</b>
2.1	Introduction . . . . .	27
2.2	Calcul de pose basé points . . . . .	28
2.2.1	AVV sur le plan image . . . . .	30
2.2.2	AVV sur la sphère . . . . .	31
2.2.3	Propagation d'incertitude . . . . .	33
2.2.4	AVV basé points en stéréovision . . . . .	34
2.2.5	Applications, comparaisons et validation . . . . .	35
2.3	Étalonnage de caméras . . . . .	43
2.3.1	Étalonnage d'une caméra omnidirectionnelle . . . . .	43
2.3.2	Étalonnage de banc stéréoscopique . . . . .	45
2.3.3	Stéréovision hybride . . . . .	46
2.3.4	Résultats . . . . .	47
2.4	Suivi et calcul de pose basé modèle 3D filaire . . . . .	57
2.4.1	AVV de modèle 3D dans le plan image . . . . .	57
2.4.2	AVV de modèle 3D en représentation sphérique cartésienne . . . . .	63
2.4.3	Applications et comparaisons . . . . .	64
2.5	Conclusion . . . . .	69
<b>3</b>	<b>Asservissement visuel photométrique</b>	<b>71</b>
3.1	Introduction . . . . .	71
3.2	Formulation du problème . . . . .	72
3.2.1	Asservissement photométrique omnidirectionnel dans le plan image . . . . .	73
3.2.2	AVP sur l'image sphérique en représentation sphérique pure . . . . .	74

3.2.3	AVP sur l'image sphérique en représentation sphérique cartésienne . . . . .	75
3.3	Application sur un robot cartésien . . . . .	77
3.3.1	Validation . . . . .	78
3.3.2	Evaluation du comportement de l'AVP pour différentes poses initiales . . . . .	79
3.3.3	Mouvement ample . . . . .	82
3.4	Application sur un robot mobile . . . . .	82
3.4.1	Formulation . . . . .	82
3.4.2	Application . . . . .	84
3.5	Application au suivi de chemin visuel . . . . .	91
3.5.1	Mise en oeuvre . . . . .	92
3.5.2	Expérimentation . . . . .	92
3.6	Conclusion . . . . .	94
<b>4</b>	<b>Suivi photométrique de primitives géométriques</b>	<b>97</b>
4.1	Introduction . . . . .	97
4.2	Suivi de plan . . . . .	97
4.2.1	Suivi sphérique 2D d'un plan . . . . .	98
4.2.2	Suivi sphérique 3D de plusieurs plans . . . . .	101
4.2.3	Suivi et reconstruction simultanés de plusieurs plans en stéréovision omnidirectionnelle . . . . .	105
4.3	Suivi de droites verticales avec le <i>FOO</i> . . . . .	110
4.3.1	Mise en correspondance et reconstruction simultanées . . . . .	112
4.3.2	Suivi photométrique et reconstruction simultanés . . . . .	120
4.4	Conclusion . . . . .	127
<b>5</b>	<b>Conclusion et perspectives</b>	<b>129</b>
	<b>Bibliographie</b>	<b>133</b>

# Préambule

En robotique, l'autonomie des systèmes passe par une perception avancée de l'environnement. Plusieurs aspects sont à aborder comme les modalités de perception et les traitements des informations perçues. La question principale au sujet des modalités de perception est : quels capteurs utiliser ? Si des capteurs proximétriques, avec une certaine connaissance de l'environnement, suffisent à localiser efficacement le robot et à lui faire atteindre un lieu désiré, des capteurs plus évolués sont nécessaires quand l'environnement est inconnu. La vision artificielle est un moyen commun d'acquérir un grand nombre d'informations sur la structure de l'environnement.

Cependant, avec une seule caméra, même s'il est possible de détecter certaines primitives géométrique, seule la direction de leur position dans la scène peut être calculée. C'est suffisant si la caméra observe un objet connu, défini par son modèle 3D, par exemple. Si la connaissance de l'objet est limitée, c'est-à-dire si le robot sait que l'objet est composé de segments mais n'en connaît pas l'agencement ni les dimensions, une seule image est insuffisante pour retrouver la position de l'objet ou localiser la caméra par rapport à cet objet. Il faut alors apporter de la redondance en utilisant plusieurs caméras sur un banc stéréoscopique ou plusieurs vues d'une même caméra à différentes positions.

Les caméras omnidirectionnelles sont une famille de capteurs visuels engendrant une perception à très large champ de vue. La vision omnidirectionnelle peut être obtenue en plaçant un miroir de révolution, par exemple un miroir sphérique, face à l'objectif de la caméra ou en utilisant un objectif à large champ de vue de type fisheye. Le champ de vue dépend du type d'équipement utilisé et il est possible d'atteindre un champ de vue supérieur à  $180^\circ$  et engendrant une perception à  $360^\circ$  autour de l'axe de la caméra.

Etant donnés les avantages de ces caméras, nous proposons d'utiliser la vision omnidirectionnelle monoculaire et stéréoscopique pour la localisation de robot mobile et la navigation. Le but est de répondre aux deux principales limitations de la vision perspective classique que sont la limitation du champ de vue et le manque de redondance.

Le traitement des informations perçues est un autre aspect important en vue de l'autonomie d'un robot. En effet, les images apportent beaucoup d'informations et permettent d'estimer la position et l'orientation 3D et de réaliser un asservissement de robot en 3D. Ces applications se basent néanmoins sur des traitements de plus bas niveau comme l'extraction de primitives géométriques (points, droites, plans, etc), leur appariement entre images ou encore leur suivi dans une séquence d'images.

Que ce soit en terme de calcul de pose par rapport à un objet connu, de pose relative en environnement inconnu ou enfin de pose relative et de reconstruction simultanées, la représentation des primitives utilisées joue un rôle dans la réussite du calcul. L'utilisation de la vision "non conventionnelle", en l'occurrence ici omnidirectionnelle, rend encore plus pertinente cette question du choix de la représentation et

de la modélisation. Dans le cas de la vision omnidirectionnelle, lorsque les caméras sont centrales, un modèle de projection a été défini et fait intervenir une sphère d'équivalence, qui est un espace unifié de représentation. Dès lors, travailler dans le plan image n'est pas la seule solution puisqu'il est possible de raisonner sur la sphère de projection. Certains travaux utilisent la sphère directement. La justification est principalement théorique : travailler sur la sphère de projection permet de traiter directement les lignes de vue, d'une manière similaire, mais pas identique, qu'avec le plan image en vision perspective. Mais en pratique, on observe que plusieurs travaux définissent des approches dans le plan image omnidirectionnel et que les résultats obtenus sont convaincants. Cependant, qu'une approche sphérique ou plane soit utilisée, elles ne sont jamais confrontées. Dès lors, une comparaison des deux modes de raisonnement est à envisager et ce, pour plusieurs types de primitive. Intuitivement, on est en droit de penser qu'un point, dans le plan image omnidirectionnel ou sur la sphère de projection, reste un point, même si sa dynamique change. Par contre, pour représenter la projection d'une droite 3D, une seule représentation suffit sur la sphère de projection alors qu'il en faut deux dans le plan image omnidirectionnel.

L'étude proposée compare les différentes représentations d'un type de primitive et vise à déduire quelle est la meilleure représentation, c'est-à-dire, sur quel lieu, plan ou sphère, il faut définir la primitive, pour le calcul de pose. Il est donc question, dans ce travail, de formuler le calcul de pose sur ces différents lieux et nous verrons que travailler sur la sphère de projection, non seulement simplifie ces formulations, mais engendre aussi de meilleurs comportements.

Le mémoire se structure en quatre chapitres :

- Le premier chapitre présente le calcul de pose linéaire et non-linéaire de caméra à point de vue unique en utilisant des points, des droites ou des plans. L'accent sera mis sur l'asservissement visuel virtuel qui est utilisé comme méthode d'estimation de pose non-linéaire tout au long de ce travail. Enfin, une discussion sur les primitives, leurs représentations et les lieux d'optimisation définira plus précisément les interrogations abordées ci-dessus et orientera les trois autres chapitres.
- Le deuxième chapitre abordera l'asservissement visuel virtuel géométrique en vision omnidirectionnelle et développera les contributions de ce travail autour du calcul de pose de caméra omnidirectionnelle monoculaire et stéréoscopique et son extension à l'étalonnage de caméras.
- L'asservissement visuel virtuel se basant bien évidemment sur l'asservissement visuel, le troisième chapitre présentera l'asservissement visuel de caméra omnidirectionnelle mais en s'intéressant à la primitive photométrique. Là aussi, l'utilisation de l'image omnidirectionnelle plane ou sphérique influe sur le comportement du système.
- Enfin, le quatrième chapitre réunira les aspects géométriques et photométriques afin de réaliser une reconstruction 3D de la trajectoire de la caméra simultanément au suivi de plans ou de droites verticales.

La conclusion détaillera les contributions suivantes :

- Le chapitre 2 compare l'estimation de pose basé points et modèle 3D filaire

en différentes représentations et montre que dans les conditions les plus difficiles, la représentation sphérique cartésienne de l'image présente un meilleur comportement avec une meilleure stabilité que les autres représentations.

- Toujours au chapitre 2, la généralisation de l'estimation de pose à l'étalonnage de caméras omnidirectionnelles, de bancs stéréos et de bancs stéréos hybrides, est une contribution importante de ce travail avec la réalisation d'un logiciel.
- Au chapitre 3, l'étude de l'asservissement visuel omnidirectionnel photométrique montre que la représentation sphérique cartésienne est la plus efficace quand le mouvement à réaliser par la caméra est important.
- Enfin, le suivi photométrique de plans et le suivi de droites verticales en stéréovision omnidirectionnelle, formulés au chapitre 4, permettent une estimation de mouvement et une reconstruction de plans simultanées et à la bonne échelle, d'une part, et d'améliorer considérablement la précision de reconstruction de droites verticales d'autres part.

La conclusion proposera aussi quelques perspectives de recherche.



# Introduction

## Sommaire

<b>1.1</b>	<b>Modélisation</b>	<b>5</b>
1.1.1	Projection perspective	5
1.1.2	Projection stéréographique	7
1.1.3	Stéréoscopie	9
<b>1.2</b>	<b>L'estimation de pose</b>	<b>13</b>
1.2.1	Estimation linéaire de pose	13
1.2.2	Estimation non-linéaire de pose	15
1.2.3	Asservissement visuel virtuel	16
1.2.4	Estimation de pose relative	20
1.2.5	Calcul de pose relative et reconstruction 3D simultanés	21
<b>1.3</b>	<b>Choix et représentation des primitives</b>	<b>23</b>
1.3.1	Lieux de définition et représentations des primitives	23
1.3.2	Appariement, pose, reconstruction : vers une simultanéité	24

## 1.1 Modélisation

En vision numérique, une image est le fruit de la projection du monde sur les cellules photosensibles d'une caméra, à travers une optique. Le mode de projection dépend de l'optique utilisée et l'image obtenue peut être très différente d'une optique à l'autre.

### 1.1.1 Projection perspective

Les caméras les plus usuelles sont de type "trou d'aiguille" (Figure 1.1(a)), c'est-à-dire que le monde est perçu à travers un trou infiniment petit, qui est généralement représenté par un point, pour obtenir l'image (Figure 1.1(b)). Ce point est le lieu de convergence des rayons lumineux perçus par la caméra : le centre de projection. Ainsi ne sera perçue que la partie de l'environnement incluse dans le volume d'une pyramide définie par la taille de l'image et la distance entre le capteur et le centre de projection : la focale. La base de ce tronc de pyramide se situe à l'infini.

La géométrie de ce type de caméra a largement été étudiée depuis les premières heures de la vision artificielle et même avant. Pour ces caméras, la projection du monde dans l'image est une projection perspective. Ainsi, un point 3D

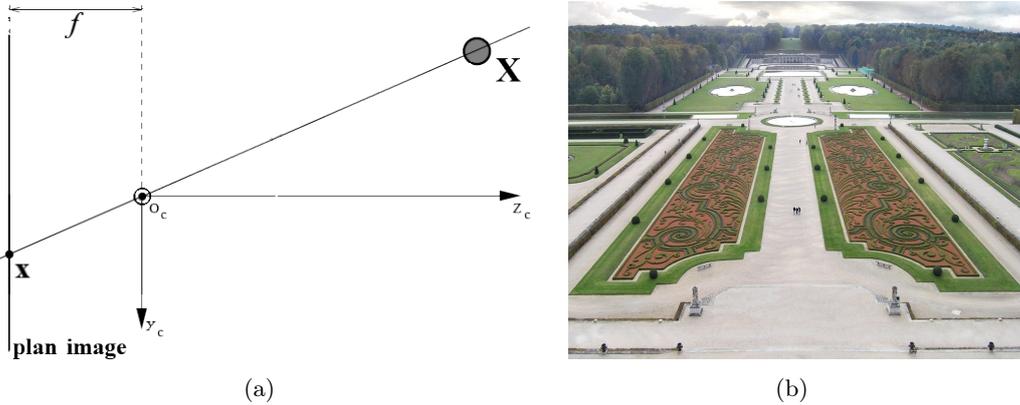


FIGURE 1.1 – Projection perspective. (a) Schéma de la projection perspective d'un point du monde 3D dans l'image 2D. (b) Cette image illustre bien la projection perspective puisque les segments parallèles du jardin du château de Vaux-Le-Vicomte deviennent des droites sécantes dans l'image.

$\mathbf{X} = (X \ Y \ Z \ 1)^\top$ , est projeté dans le plan image en  $\mathbf{x} = (x \ y \ 1)^\top$  par la fonction de projection perspective  $pr(\cdot)$ , tel que  $f$  est la focale de la caméra :

$$\mathbf{x} = pr(\mathbf{X}) \quad \text{avec} \quad \begin{cases} x = f \frac{X}{Z} \\ y = f \frac{Y}{Z} \end{cases} . \quad (1.1)$$

$\mathbf{X}$  est ici exprimé dans le repère caméra  $\mathcal{F}_c$ , défini par le centre  $\mathbf{O}_c$  et les axes  $\mathbf{X}_c$ ,  $\mathbf{Y}_c$  et  $\mathbf{Z}_c$  (Figure 1.1(a)). Son projeté  $\mathbf{x}$  est défini dans le plan image normalisé dont l'origine est le point principal de l'image, c'est-à-dire le point d'intersection de l'axe  $\mathbf{Z}_c$  et du plan image.

Une image est généralement stockée dans la mémoire d'un ordinateur sous forme de tableau, c'est pourquoi l'origine de l'image se situe en haut à gauche et non au centre. Un changement de repère est donc nécessaire pour exprimer un point du plan image normalisé dans le plan image pixelique. Toutefois, un changement d'échelle doit d'abord être appliqué aux points du plan image normalisé pour les exprimer en pixels avant le changement de repère. Les paramètres permettant ce changement d'échelle et le changement de repère sont les paramètres intrinsèques de la caméra :  $\gamma_1 = \{\alpha_u, \alpha_v, u_0, v_0\}$ .  $\alpha_u$  et  $\alpha_v$  sont les paramètres de changement d'échelle et incorporent la focale  $f$  de la caméra et les longueurs des côtés d'un pixel  $k_u, k_v$  :  $\alpha_u = fk_u$  et  $\alpha_v = fk_v$ . En ayant posé ces deux dernières expressions, la focale  $f$  est éliminée de l'équation 1.1, menant à :

$$pr(\mathbf{X}) = \begin{pmatrix} \frac{X}{Z} \\ \frac{Y}{Z} \end{pmatrix} . \quad (1.2)$$

Enfin,  $u_0$  et  $v_0$  sont les coordonnées du point principal dans l'image. Le pixel  $\mathbf{u} = (u \ v \ 1)^\top$  est alors obtenu à partir de  $\mathbf{x}$  par  $\mathbf{u} = \mathbf{K}\mathbf{x}$  avec  $\mathbf{K}$  la matrice des

paramètres intrinsèques :

$$\mathbf{K} = \begin{pmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (1.3)$$

Ceci conduit à la fonction complète de projection d'un point 3D, exprimé dans  $\mathcal{F}_c$ , dans le plan image pixelique :

$$pr_{\gamma_1}(\mathbf{X}) = \mathbf{K}pr(\mathbf{X}). \quad (1.4)$$

Le modèle de projection présenté prend en compte le changement d'échelle, le changement de repère et le fait que les pixels ne sont éventuellement pas carrés mais rectangulaires. Ce modèle peut être enrichi pour prendre en compte : le fait que les pixels sont des parallélogrammes, les distorsions radiales ou tangentielles engendrées par l'optique utilisée, etc.

Remarque : Dans la version idéale du modèle de projection perspective, les lignes de vue s'intersectent toutes en un seul point : le centre de projection. Tout modèle de projection ayant cette caractéristique est dit à projection centrale et fait ainsi partie de la famille des modèles de caméras à point de vue unique.

### 1.1.2 Projection stéréographique

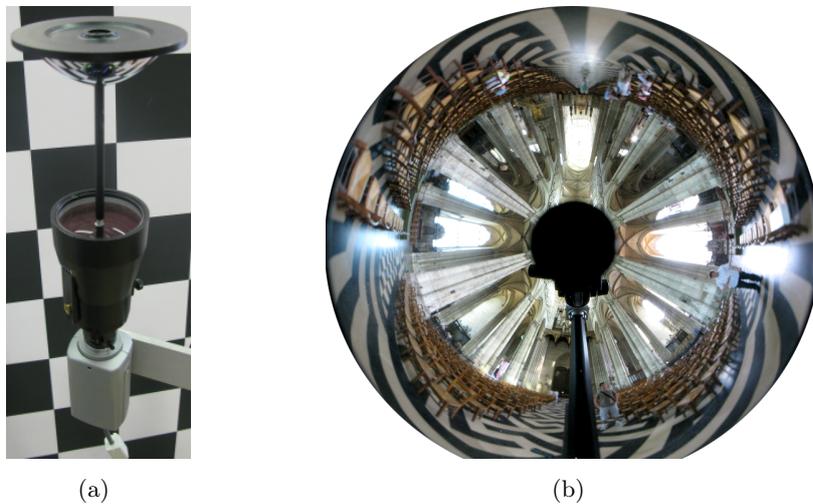


FIGURE 1.2 – Vision omnidirectionnelle. (a) Caméra catadioptrique. (b) Image omnidirectionnelle à l'intérieur de la cathédrale d'Amiens obtenue avec un objectif catadioptrique.

Les caméras omnidirectionnelles, à une seule caméra et sans dispositif mécanique de mouvement, utilisent soit une optique très grand angle de type fisheye, soit une optique catadioptrique (Figure 1.2(a)). Une caméra fisheye a une focale très courte et une caméra catadioptrique possède un miroir de révolution placé devant la caméra pour agrandir le champ de vue. Dans le cas d'une image catadioptrique

(Figure 1.2(b)), l'image est obtenue par la réflexion du monde sur le miroir (catoptrique) qui traverse ensuite l'optique (dioptrique). Seules certaines formes de miroir de révolution permettent de conserver la contrainte de point de vue unique, à savoir les miroirs sphériques, ellipsoïdes et hyperboloïdes avec une caméra perspective et le miroir parabololoïde avec une caméra orthographique [Baker 1999].

Un modèle de projection stéréographique [Geyer 2000, Barreto 2001] a été proposé pour unifier ces différents types de caméras à point de vue unique. Selon ce modèle, le monde est d'abord projeté sur une sphère puis sur le plan image (Figure 1.3).

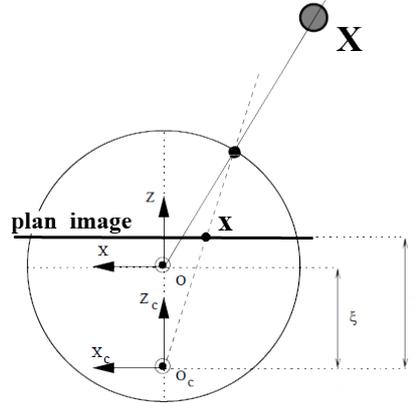


FIGURE 1.3 – Schéma de la projection stéréographique d'un point du monde 3D dans l'image 2D selon le modèle unifié de projection centrale [Barreto 2001].

Selon le modèle de projection unifié [Barreto 2001], un point 3D  $\mathbf{X}$  est d'abord projeté sur la sphère unitaire en  $\mathbf{X}_S$ , par la fonction de projection sphérique  $pr_S$  :

$$\mathbf{X}_S = \begin{pmatrix} X_S \\ Y_S \\ Z_S \end{pmatrix} = pr_S(\mathbf{X}) \quad \text{avec} \quad \begin{cases} X_S = \frac{X}{\rho} \\ Y_S = \frac{Y}{\rho} \\ Z_S = \frac{Z}{\rho} \end{cases} \quad (1.5)$$

où  $\rho = \sqrt{X^2 + Y^2 + Z^2}$ . Ensuite,  $\mathbf{X}_S$  est projeté dans le plan image en  $\mathbf{x}$  par une projection perspective, telle que  $\mathbf{x} = pr(\mathbf{X}_S)$ , grâce à un deuxième centre de projection dans lequel la sphère a pour coordonnées  $(0 \ 0 \ \xi)^T$ . La relation directe entre  $\mathbf{X}$  et  $\mathbf{x}$  définit la fonction de projection  $pr_\xi$  :

$$\mathbf{x} = pr_\xi(\mathbf{X}) \quad \text{avec} \quad \begin{cases} x = \frac{X}{Z + \xi\rho} \\ y = \frac{Y}{Z + \xi\rho} \end{cases} \quad (1.6)$$

$\mathbf{x}$  est un point du plan image normalisé et il est transformé en coordonnées pixeliques  $\mathbf{u}$  par la matrice  $\mathbf{K}$  des paramètres intrinsèques :  $\mathbf{u} = \mathbf{K}\mathbf{x}$ .

La projection de la sphère vers le plan image est inversible, ce qui permet de

retrouver un point sphérique à partir d'un point image :

$$\mathbf{X}_S = pr_{\xi}^{-1}(\mathbf{x}) = \begin{pmatrix} \frac{\xi + \sqrt{1 + (1 - \xi^2)(x^2 + y^2)}}{x^2 + y^2 + 1} x \\ \frac{\xi + \sqrt{1 + (1 - \xi^2)(x^2 + y^2)}}{x^2 + y^2 + 1} y \\ \frac{\xi + \sqrt{1 + (1 - \xi^2)(x^2 + y^2)}}{x^2 + y^2 + 1} - \xi \end{pmatrix}. \quad (1.7)$$

Enfin, en considérant  $\gamma_2 = \{\alpha_u, \alpha_v, u_0, v_0, \xi\}$ , l'ensemble des paramètres intrinsèques du modèle de projection unifié, la fonction de projection d'un point 3D, exprimé dans  $\mathcal{F}_c$ , dans le plan image pixelique s'écrit :

$$pr_{\gamma_2}(\mathbf{X}) = \mathbf{K}pr_{\xi}(\mathbf{X}). \quad (1.8)$$

Ce modèle de projection unifié est valide pour toute caméra à point de vue unique, y compris les caméras perspectives. En effet, il suffit d'annuler  $\xi$  pour retrouver une projection perspective seule.  $\xi$  est d'ailleurs le paramètre dont la valeur dépend de la forme du miroir [Barreto 2001].

Enfin, même si la caméra fisheye n'est pas à point de vue unique, ce modèle de projection est une bonne approximation de la réelle projection et peut être utilisé pour modéliser certaines caméras de ce type [Ying 2004].

### 1.1.3 Stéréoscopie

La stéréovision est généralement réalisée par l'assemblage rigide de deux caméras, dont les champs de vue s'intersectent, sur un même banc. C'est le cas en vision perspective comme en vision omnidirectionnelle (Figure 1.4).

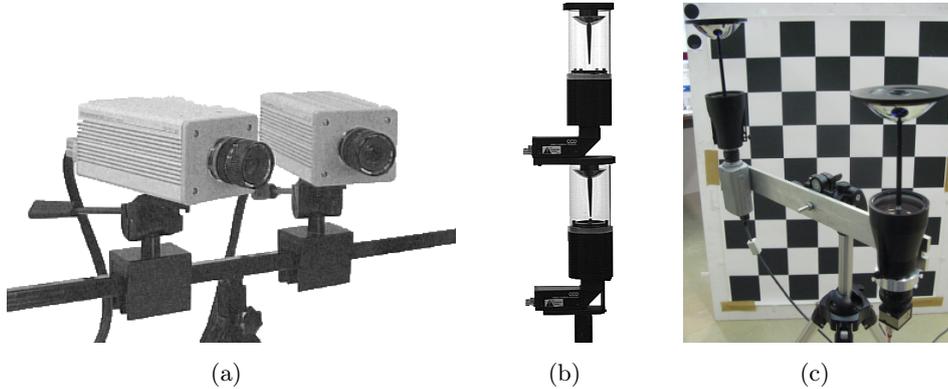


FIGURE 1.4 – Exemples de bancs stéréo. (a) Banc stéréo perspective. (b) Banc stéréo omnidirectionnel vertical [Gluckman 1998]. (c) Banc stéréo omnidirectionnel horizontal. Avec un banc stéréo omnidirectionnel vertical [Gluckman 1998], repris dans de récents travaux [Boutteau 2009], il n'y a pas d'inter-réflexions entre les caméras, contrairement à une configuration horizontale.

On modélise ce type de banc stéréo par une fonction de projection pour chaque caméra, avec les paramètres intrinsèques associés, et la transformation rigide 3D

définissant la position d'une caméra par rapport à la caméra de référence. Cette transformation permet donc de passer du repère de la caméra de référence  $\mathcal{F}_{c_1}$  au repère de l'autre caméra,  $\mathcal{F}_{c_2}$ . En considérant l'orientation relative entre deux caméras d'un banc stéréo  ${}^{c_2}\mathbf{R}_{c_1}$  et la position relative  ${}^{c_2}\mathbf{t}_{c_1}$ , la matrice homogène de transformation rigide  ${}^{c_2}\mathbf{M}_{c_1}$  s'écrit :

$${}^{c_2}\mathbf{M}_{c_1} = \begin{pmatrix} {}^{c_2}\mathbf{R}_{c_1} & {}^{c_2}\mathbf{t}_{c_1} \\ 0 & 1 \end{pmatrix}, \text{ avec } {}^{c_2}\mathbf{t}_{c_1} \in \mathbb{R}^3 \text{ et } {}^{c_2}\mathbf{R}_{c_1} \in \mathbf{SO}(3). \quad (1.9)$$

$\mathbf{SO}(3)$  est le groupe spécial des matrices orthogonales. Quant à la matrice  ${}^{c_2}\mathbf{M}_{c_1}$ , elle appartient au groupe spécial des transformations euclidiennes  $\mathbf{SE}(3)$ .

En vision omnidirectionnelle, cependant, des configurations stéréoscopiques particulières existent avec une seule caméra et plusieurs miroirs. SPHERO [Nayar 1988], un capteur composé d'une caméra et de deux miroirs sphériques coplanaires a ouvert la voie (Figure 1.5(a)). Les intérêts d'utiliser une seule caméra pour la stéréovision omnidirectionnelle sont, d'une part, d'éliminer les problèmes de synchronisation entre les caméras d'un banc stéréo et, d'autre part, d'avoir exactement les mêmes propriétés chromatiques pour les vues provenant des différents miroirs. Cependant, même si dans une telle configuration deux miroirs suffisent, l'un cache des zones de l'environnement à l'autre et vice versa (Figure 1.5(b)). Ce problème d'auto-occultation intervient quand les miroirs sont coplanaires. De plus, la reconstruction de points 3D de l'environnement se situant aux abords de l'axe formé par les deux miroirs souffre d'une incertitude importante, les deux lignes de vues à intersecter étant quasi parallèles.

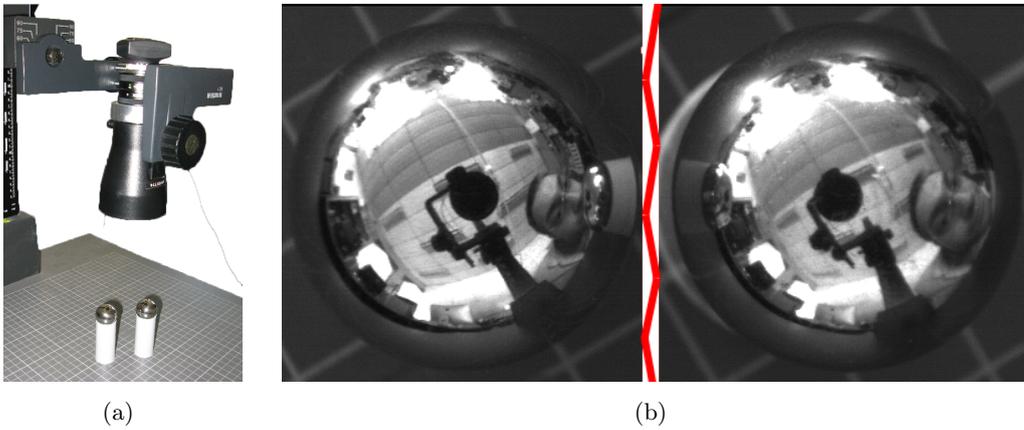


FIGURE 1.5 – SPHEREO [Nayar 1988]. (a) Reproduction d'un système similaire à SPHEREO avec une caméra à objectif quasi-télécentrique et deux miroirs sphériques. (b) Une image obtenue avec ce capteur (la partie centrale de l'image a été supprimée pour rendre la figure compacte). Le bas du visage est visible sur le miroir de droite mais pas sur celui de gauche à cause de l'occultation mutuelle.

Pour contrer le problème d'inter-réflexions dans un système stéréo catadioptrique à une seule caméra, il est possible d'assembler plusieurs miroirs verticalement et face

à face [Jang 2005]. La figure 1.6(a) illustre ce système. Un miroir hyperboloïde, percé en son centre pour laisser passer les rayons lumineux vers la caméra, fait face à un autre miroir hyperboloïde tronqué. La troncature permet d'obtenir un miroir plan sur lequel se reflète l'image du monde, déjà reflétée par le premier miroir. L'agencement de ces trois miroirs permet de garder la contrainte de point de vue unique pour chaque vue du système stéréo. L'image obtenue par ce système est composée de deux anneaux concentriques avec, vers le centre, l'image correspondant au miroir percé et, vers l'extérieur, l'image correspondant au miroir tronqué (Figure 1.6(b)). L'avantage d'un tel système est de voir l'appariement de point facilité puisque deux points correspondant se situent sur la même droite radiale. Cependant, ce système d'une hauteur de 17 cm exacerbe le problème de focalisation, présent en vision catadioptrique monoculaire. La profondeur de champ doit être maximisée en fermant le diaphragme au maximum ce qui engendre un temps d'acquisition très important pour une image, rendant difficile l'utilisation de ce système pour une application en robotique.

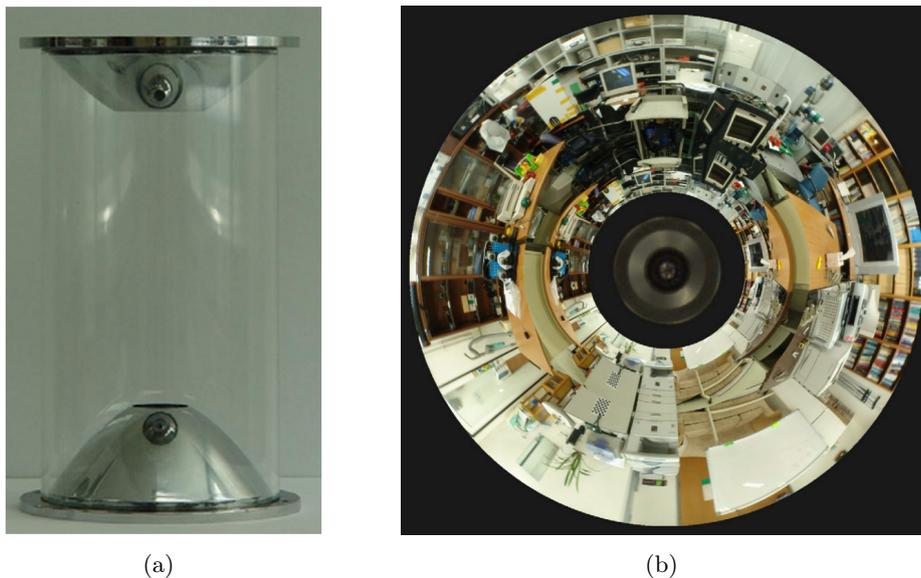


FIGURE 1.6 – Système stéréo catadioptrique à une caméra avec miroirs non-coplanaires [Jang 2005]. (a) Le système de miroirs. (b) Une image obtenue en montnant ce système de miroirs sur une caméra : les deux vues sont concentriques dans l'image.

En utilisant une caméra orthographique et quatre miroirs paraboliques de révolution placés sur un même plan [Mouaddib 2005], les problèmes de point de vue unique, de focalisation, de perception stéréo, quelque soit l'orientation de la zone d'intérêt, sont résolus. En effet, utiliser une caméra orthographique avec un miroir parabolique dans un système de vision catadioptrique permet de conserver une projection centrale, quelque soit l'alignement du miroir par rapport à l'objectif, pourvu que l'axe principal du miroir soit parallèle à l'axe de la caméra. Ainsi, il est possible

de disposer un nombre quelconque de miroirs de taille quelconque mais il s'avère que quatre miroirs paraboliques de même taille et placés en carré (Figure 1.7(a)) donnent les meilleurs résultats en terme de perception stéréo de l'environnement tout autour du capteur et de précision de reconstruction 3D [Dequen 2007]. Ce système présente bien évidemment des réflexions entre les miroirs comme dans le cas de SPHEREO mais les quatre miroirs font que tout point du champ de vue du capteur stéréo reste visible par au moins deux miroirs (Figure 1.7(b)). De plus, dans la partie du champ de vue où un point de l'environnement est visible par trois miroirs ou plus, la reconstruction 3D est contrainte par au moins une ligne de vue supplémentaire ce qui en réduit l'incertitude. Enfin, les miroirs étant de taille identique et disposés à la même distance de l'objectif de la caméra, la profondeur de champ ne pose pas plus de problème qu'en vision catadioptrique monoculaire.

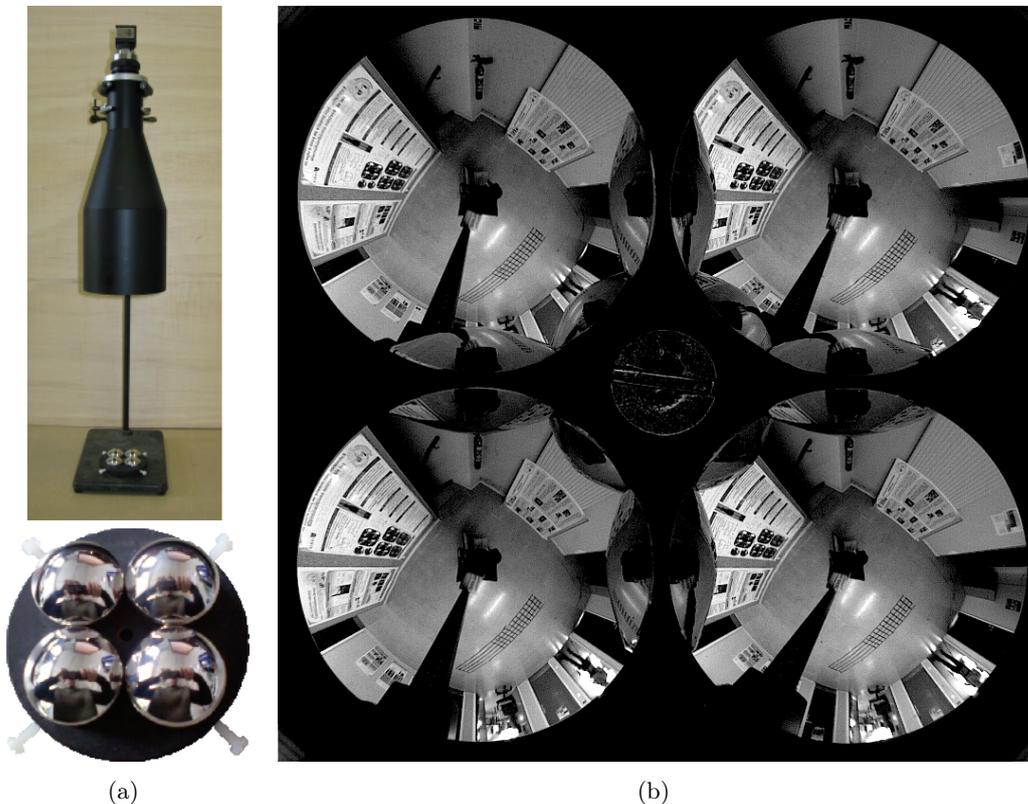


FIGURE 1.7 – Le Four On One (FOO). (a) Système stéréo catadioptrique à une caméra avec quatre miroirs paraboliques de révolution coplanaires [Mouaddib 2005] avec un zoom sur les miroirs. (b) Une image obtenue avec ce capteur. Il existe des zones d'inter-réflexion mais, grâce aux quatre miroirs, tout point 3D du champ de vue du capteur est visible par au moins deux miroirs (cf. les posters sur la gauche).

## 1.2 L'estimation de pose

Le calcul de pose en environnement connu ou partiellement connu consiste à calculer la position et l'orientation de la caméra par rapport au repère monde (ou objet, sans perte de généralité) en utilisant les mesures 2D dans l'image et leur correspondance 3D. Il repose sur la connaissance a priori d'une structure 3D, comme un objet dont les coordonnées des sommets ou des arêtes seraient connues dans le même repère. Le calcul de pose de la caméra par rapport à un objet ou de l'objet par rapport à la caméra est équivalent. Ainsi, la matrice homogène  ${}^c\mathbf{M}_o$  de changement de repère du repère objet  $\mathcal{F}_o$  vers le repère caméra  $\mathcal{F}_c$  s'écrit :

$${}^c\mathbf{M}_o = \begin{pmatrix} {}^c\mathbf{R}_o & {}^c\mathbf{t}_o \\ 0 & 1 \end{pmatrix}. \quad (1.10)$$

Ainsi, connaissant un point du repère objet  ${}^o\mathbf{X} = ({}^oX \ {}^oY \ {}^oZ \ 1)^\top$ , il s'exprime, dans le repère caméra par :

$${}^c\mathbf{X} = \begin{pmatrix} {}^cX \\ {}^cY \\ {}^cZ \\ 1 \end{pmatrix} = {}^c\mathbf{M}_o {}^o\mathbf{X}. \quad (1.11)$$

L'objectif du calcul de pose de caméra est donc de calculer la transformation homogène  ${}^c\mathbf{M}_o$  à partir des primitives 3D de l'objet exprimées dans  $\mathcal{F}_o$  et leur correspondance 2D dans l'image.

### 1.2.1 Estimation linéaire de pose

La plus ancienne méthode pour résoudre ce problème a, probablement, été inventée par Lagrange en 1795 en utilisant trois points. Ce problème a été largement étudié depuis et est généralement appelé "P-3-P" dans la littérature pour "perspective-3-points-problem" ou "P- $n$ -P" pour  $n$  points. Les positions des points 3D sont généralement calculées dans le repère caméra en résolvant un système linéaire dans un premier temps. La transformation 3D permettant de transformer les points du repère objet dans le repère caméra est calculée dans un second temps. De nombreuses variantes existent [Haralick 1991] et sont plus ou moins stables numériquement. Avec trois points, ce problème admet plusieurs solutions et il faut en ajouter pour assurer l'unicité de la solution. A partir de quatre points, l'unicité de la solution est assurée hors de configurations critiques [Thompson 1966, Wrobel 1992]. La pose de la caméra sera ainsi ambiguë si trois des quatre points forment un plan avec le centre de projection de la caméra.

Il apparaît donc intéressant d'utiliser la redondance de points importante pour éviter les cas critiques où la pose obtenue est ambiguë. L'approche RANSAC [Fischler 1981] (RANDOM SAMPLE CONSENSUS) consiste à sélectionner aléatoirement un ensemble minimum de points permettant de calculer la pose et la calcule par une des méthodes linéaires mentionnées précédemment. Cette pose

est appliquée à tous les points de l'objet qui sont ensuite projetés dans l'image. L'ensemble des distances entre ces points et les points détectés forme une mesure d'erreur qui sera minimale pour la pose optimale. Ce processus de sélection aléatoire et de calcul de pose est reproduit plusieurs fois jusqu'à ce que la pose calculée engendre une erreur faible sur un maximum de points.

Une telle approche n'utilise que le nombre minimum de primitives géométriques pour calculer la pose, si bien que dans le cas où toutes les mesures sont bruitées, même si une pose satisfaisante à l'ensemble des primitives est trouvée, elle reste imprécise. Utiliser l'ensemble des primitives pour calculer la pose apparaît alors potentiellement plus précis puisque chaque primitive apporte de l'information. Dans cette optique, d'autres approches linéaires ont été développées pour calculer la position 3D de  $n$  points de l'objet ( $n \geq 4$ ) dans le repère caméra [Quan 1999] pour en déduire la pose  ${}^c\mathbf{M}_o$ . Cette dernière méthode utilise les lignes de vue pour calculer la position des  $n$  points 3D et peut donc être utilisée avec le modèle stéréographique. D'autres approches ont été adaptées à la vision omnidirectionnelle en utilisant le modèle unifié [Paulino 2002] ou en utilisant un modèle de projection générique qui permet de prendre en compte les capteurs non centraux [Gonçalves 2007], dès lors qu'il est possible de les étalonner.

Dans cette thèse, l'estimation linéaire de pose est utilisée pour apporter une valeur initiale à l'optimisation non-linéaire en utilisant quatre points. La méthode retenue [Ameller 2002], consiste à linéariser un système d'équations polynômiales faisant intervenir les distances  $\rho_i$  entre les points 3D de l'objet de référence et l'origine de la caméra, les distances  $d_{ij}$  entre points de l'objet et les angles  $\theta_{ij}$  entre les lignes de vue associées aux points perçus (Figure 1.8) :

$$P_{ij}(\rho_i, \rho_j) = \rho_i^2 + \rho_j^2 - 2 \cos \theta_{ij} \rho_i \rho_j - d_{ij}^2 = 0. \quad (1.12)$$

Pour quatre points, on obtient un système de six équations de la forme de l'équation 1.12 et en considérant le système :

$$\{\rho_i P_{jk}(\rho_j, \rho_k) = 0 \mid i, j < k = 1, 2, 3, 4\}, \quad (1.13)$$

qui contient 24 polynômes de 24 monômes, la matrice  $24 \times 24$ , dont les coefficients sont directement obtenus par les  $\cos \theta_{jk}$  et des  $d_{jk}^2$ , est décomposable par SVD pour obtenir un vecteur du noyau (un vecteur de l'espace nul) composé des 24 monômes. Ce vecteur permet de retrouver simplement les profondeurs  $\rho_i$  des points du modèle et d'en déduire la rotation, puis la translation [Horn 1987] entre  $\mathcal{F}_o$  et  $\mathcal{F}_c$  pour obtenir  ${}^c\mathbf{M}_o$ .

Malgré les approches linéaires permettant de prendre en compte une redondance de primitives, à la manière des moindres carrés, et d'être donc robuste à un bruit gaussien, l'approche RANSAC possède un deuxième intérêt majeur. Elle permet, en effet, la gestion des aberrations ou *outliers*, c'est-à-dire des mauvaises correspondances 2D-3D, ce que les méthodes linéaires classiques à  $n$  points ( $n \geq 4$ ) ne peuvent pas prendre (ou n'ont pas encore pris) en compte. De récents travaux ouvrent la voie vers une possible gestion des outliers dans des approches linéaires en pondérant les

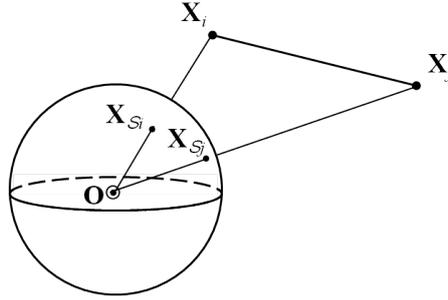


FIGURE 1.8 – Géométrie pour le calcul linéaire de pose de caméra pour chaque paire de correspondance entre les points sphériques et les points 3D.  $\rho_i$  est la distance entre  $\mathbf{O}$  et  $\mathbf{X}_i$  et  $\theta_{ij}$  est l'angle formé par les lignes de vue  $\mathbf{OX}_i$  et  $\mathbf{OX}_j$ .

mesures [Hmam 2010] ou en utilisant le nombre de points minimum (points virtuels de contrôle) obtenus par somme pondérée de tous les points de l'objet [Lepetit 2009]. Cependant, ces méthodes ne présentent la gestion des outliers que comme perspectives ou n'évaluent pas réellement leurs méthodes sur ce point et deviennent de toute façon itératives.

### 1.2.2 Estimation non-linéaire de pose

Le problème de robustesse aux faux appariements a été abordé à maintes reprises mais sous la forme de techniques d'optimisations permettant de minimiser l'erreur du critère :

$$\Delta = \sum_{k=1}^{k_1} (pr_{\gamma_m}({}^c\mathbf{M}_o {}^o\mathbf{P}_k) - \mathbf{p}_k^*)^2 \quad (1.14)$$

où  $\mathbf{p}_k$  représente les coordonnées d'une primitive dans l'image,  ${}^o\mathbf{P}_k$  dans le repère objet et  $pr_{\gamma_m}$  la fonction de projection de la primitive 3D pour les paramètres  $\gamma_m$  du modèle correspondant. Pour la primitive "point",  $\mathbf{u}$  se substitue à  $\mathbf{p}$  et la fonction de projection est  $pr_{\gamma_1}$  si la projection perspective est utilisée ou  $pr_{\gamma_2}$  dans le cas du modèle stéréographique (cf. partie 1.1).

Ce type d'approche se base sur une formulation non-linéaire du problème de calcul de pose contrairement aux approches linéaires. La "linéarisation" d'une fonction non-linéaire est obtenue par un développement de Taylor qui approxime la fonction d'origine en une fonction polynôme dont les coefficients dépendent des dérivées d'ordre 1 à  $n$  de la fonction au voisinage d'un point. Les méthodes d'optimisation bien connues de descente de gradient, Gauss-Newton (GN) ou Levenberg-Marquardt (LM) utilisent la dérivée première de la fonction non-linéaire. En théorie, plus on utilise les ordres élevés de dérivation, plus l'approximation s'approche de la fonction elle-même mais les dérivées d'ordre supérieur à 1 deviennent vite très complexes. Il existe néanmoins des algorithmes utilisant l'ordre 2 [Malis 2004].

De nombreux travaux en vision perspective traitent le problème de calcul non-linéaire de pose en utilisant des méthodes du premier ordre [Lowe 1991,

[DeMenthon 1995, Phong 1995], où les différences principales résident dans la paramétrisation de la rotation (Euler, quaternion, vecteur-angle). En raisonnant sur la vitesse des éléments de la pose, l’asservissement visuel virtuel [Marchand 2002] permet de travailler directement avec les trois angles de rotation (cf. partie 1.2.3).

Certaines de ces méthodes peuvent être directement étendues à la vision omnidirectionnelle. L’asservissement visuel virtuel a déjà fait un pas dans cette direction [Marchand 2007]. Cette méthode permet de bénéficier du travail déjà réalisé en asservissement visuel réel d’une caméra omnidirectionnelle pour le calcul de pose.

### 1.2.3 Asservissement visuel virtuel

L’asservissement visuel virtuel [Marchand 2002] (AVV) est une technique d’optimisation permettant d’estimer la pose d’une caméra. Cette approche raisonne en vitesse de déplacement d’une caméra virtuelle de sa position initiale vers la position de la caméra réelle. Ce principe est le même que l’asservissement visuel, une technique de positionnement de robot par la vision, qui déplace une caméra d’une position initiale vers une position désirée en utilisant des informations visuelles des images. L’AVV bénéficie donc de tout le travail réalisé en asservissement visuel qui en est la base. Ce dernier est donc tout d’abord présenté et est suivi de l’asservissement visuel virtuel.

#### 1.2.3.1 Asservissement visuel

Cette technique de positionnement vise à amener l’effecteur du robot à une pose telle que l’image courante et l’image acquise, au préalable, à la position désirée soient identiques. Pour comparer ces images, on peut utiliser des primitives géométriques comme les points, les droites, les sphères ou les moments car leur utilisation permet de lier analytiquement le mouvement dans l’image au mouvement de la caméra. L’utilisation de l’image elle-même comme référence, c’est-à-dire l’ensemble des intensités des pixels, n’a été abordée que très récemment en vision perspective [Collewet 2008], comme en vision omnidirectionnelle [Caron 2010a]. Ce dernier point constitue une des contributions de cette thèse et sera développé dans le chapitre 3.

Cette tâche robotique de positionnement est exprimée sous la forme d’une régulation d’erreur :

$$\mathbf{e} = \mathbf{s}(\mathbf{r}) - \mathbf{s}^* \quad (1.15)$$

où  $\mathbf{r} = [t_X, t_Y, t_Z, \theta_X, \theta_Y, \theta_Z]$  est une représentation vectorielle de la pose de la caméra.  $\mathbf{s}(\mathbf{r})$  représente le vecteur de primitives image pour la pose courante  $\mathbf{r}$  et  $\mathbf{s}^*$  le vecteur des primitives désirées, c’est-à-dire les primitives image à la pose désirée.  $\mathbf{s}^*$  est donc la consigne de la tâche et est généralement considérée constante.

En imposant une décroissance exponentielle de l’erreur, on a :

$$\dot{\mathbf{e}} = -\lambda \mathbf{e}, \quad (1.16)$$

où  $\lambda$  est le gain de convergence et  $\dot{\mathbf{e}}$  ne dépend que de  $\dot{\mathbf{s}}$ , le mouvement des primitives dans l'image. Ce mouvement est lié à la vitesse de déplacement  $\mathbf{v}$  de la caméra [Espiau 1992] :

$$\dot{\mathbf{s}} = \frac{d\mathbf{s}}{dt} = \frac{\partial \mathbf{s}}{\partial \mathbf{r}} \frac{d\mathbf{r}}{dt} = \frac{\partial \mathbf{s}}{\partial \mathbf{S}} \frac{\partial \mathbf{S}}{\partial \mathbf{r}} \frac{d\mathbf{r}}{dt} = \mathbf{L}_s \mathbf{v}. \quad (1.17)$$

$\mathbf{v} = (v, \omega)$  est le torseur cinématique de la pose avec  $v = (v_X, v_Y, v_Z)$ , la vitesse de translation et  $\omega = (\omega_X, \omega_Y, \omega_Z)$ , la vitesse de rotation.  $\mathbf{L}_s$  est la matrice d'interaction liant le mouvement des primitives dans l'image à la vitesse de la caméra [Chaumette 1990]. La matrice d'interaction est le produit de deux jacobiens (équation 1.17). Le premier dépend du modèle de projection contrairement au second. Pour introduire un nouveau modèle de projection dans un schéma d'asservissement visuel, il faut calculer le jacobien de la primitive image par rapport à la primitive 3D exprimée dans le repère caméra. En vision perspective, les matrices d'interactions ont été élaborées pour les points [Feddema 1989] et celles pour les droites, les segments, les cercles, les sphères, les cylindres et les moments sont formulées dans [Chaumette 1990].

L'asservissement visuel en vision omnidirectionnelle a surtout pris un essor important à partir de l'apparition du modèle de projection unifié. Les matrices d'interactions pour les points [Barreto 2002], les droites [Barreto 2003a, Hadj-Abdelkader 2006], les combinaisons de points, les sphères [Tatsambon Fomena 2008] et les moments sphériques [Tahri 2010] ont été exprimées. Si en vision classique la projection perspective est presque toujours utilisée, la projection stéréographique, définie pour modéliser les caméras omnidirectionnelles, engendre la possibilité de travailler sur une autre surface que l'image plane : la sphère de projection. L'utilisation directe de la sphère présente l'avantage de traiter directement les lignes de vue, de la même manière qu'avec le plan image en vision perspective. Les travaux où les primitives, et donc la matrice d'interaction, sont exprimées sur la sphère [Tatsambon Fomena 2008, Hadj-Abdelkader 2010, Tahri 2010] et non dans le plan image sont à différencier des autres. Il apparaît de plus que les expressions des matrices d'interactions sont plus simples sur la sphère, quelque soit le type de primitive sus-mentionné, que dans le plan image [Tatsambon Fomena 2008].

Une fois la matrice d'interaction exprimée et calculée, la commande en vitesse est déduite à partir des équations 1.17 et 1.16 :

$$\mathbf{v} = -\lambda \mathbf{L}_s^+ \mathbf{e} \quad (1.18)$$

avec  $\mathbf{L}_s^+$ , la pseudo-inverse à gauche de la matrice d'interaction. En pratique, même si l'expression de  $\mathbf{L}_s$  est analytiquement connue, elle est calculée à partir de paramètres approximatifs tels que les paramètres intrinsèques du modèle de projection de la caméra. Ces paramètres sont estimés au préalable dans une étape d'étalonnage à l'issue de laquelle un ensemble de paramètres  $\gamma_m$  est obtenu au sens du maximum de vraisemblance, mais il persiste toujours une erreur résiduelle qui, aussi infime puisse-t-elle être, illustre la connaissance imparfaite de ces paramètres. Dès lors, on ne travaille pas avec la parfaite matrice d'interaction mais avec une estimation  $\hat{\mathbf{L}}_s$  de cette dernière.

### 1.2.3.2 Asservissement visuel virtuel

Le principe de l'asservissement visuel virtuel [Sundareswaran 1998, Marchand 2002] est de virtualiser la caméra. Partant d'une pose initiale, le but est de déplacer la caméra de telle sorte que la projection du modèle d'un objet ou de la scène dans l'image, pour la pose virtuelle, s'ajuste sur l'image réelle courante de l'objet. Ce processus d'asservissement virtuel est similaire à une méthode d'estimation non-linéaire de la pose. Pour le calcul de pose, l'AVV suppose que les paramètres intrinsèques sont connus. Il est cependant possible de relâcher cette connaissance pour étalonner une caméra [Marchand 2001b] ou un système stéréo [Caron 2010a], pour divers modèles de projection, comme il en sera question au chapitre 2.

Le principe est identique à celui de l'asservissement visuel, avec les mêmes matrices d'interaction et le calcul de la vitesse de la caméra est identique. Cependant, pour déplacer la caméra virtuelle, c'est sa pose, définie sur le groupe  $\mathbf{SE}(3)$ , qui est mise à jour à l'itération  $it$  grâce à l'application exponentielle de l'algèbre de Lie de  $\mathfrak{se}(3)$  [Ma 2004] :

$${}^c\mathbf{M}_o^{it} = {}^c\mathbf{M}_o^{it-1} e^{[v]}. \quad (1.19)$$

L'AVV permet de calculer la pose d'un objet défini par un modèle 3D composé de primitives quelconques, pourvu que l'on puisse en définir la matrice d'interaction. Il a ainsi été appliqué au suivi et calcul de pose simultanés d'objet défini par un modèle 3D filaire [Marchand 2001a], c'est-à-dire composé de sommets et d'arêtes, dans le but de faire l'asservissement visuel sur des objets complexes ou de faire de la réalité augmentée (Figure 1.9). Ce suivi simultané au calcul de pose utilisant un modèle filaire a introduit dans le formalisme d'asservissement visuel (virtuel) une nouvelle mesure d'erreur qui est la distance entre un point et une droite. Dans une séquence, le principe est de projeter le modèle 3D dans une image en utilisant les paramètres de pose de l'image précédente et de rechercher, sur la direction normale à une arête projetée, le contour réel dans l'image. Un grand nombre de données est ainsi utilisé pour une mesure d'erreur globalement cohérente pour tout l'objet.

Par contre, si les contours de l'objet ne sont pas très bien marqués ou s'il y a une occultation partielle de l'objet, le suivi peut échouer. Il a donc été rendu robuste grâce à l'utilisation d'un M-Estimateur [Comport 2003] issu des statistiques robustes [Huber 1981]. L'idée est ici de pondérer chaque mesure, *i.e.* chaque distance point-droite, pour minimiser l'impact des mesures erronées et maximiser celle des bonnes mesures en se basant sur l'écart type des erreurs. Le critère à minimiser devient alors :

$$\Delta_{\mathcal{R}} = \sum_{k=1}^{k_1} \Gamma(\mathbf{s}_k(\mathbf{r}) - \mathbf{s}_k^*), \quad (1.20)$$

où  $\Gamma(u)$  est une fonction robuste qui croît de façon subquadratique et est non-décroissante monotone à mesure que  $|u|$  croît. La méthode habituelle d'application de M-Estimateur revient à effectuer les moindres carrés itérés repondérés (Iterative

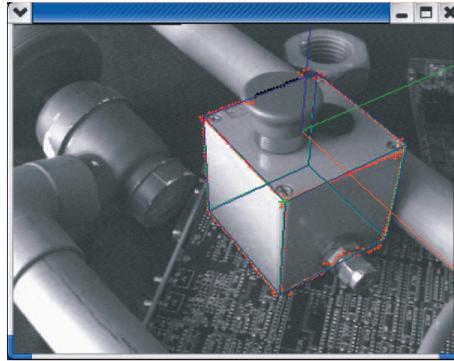


FIGURE 1.9 – Image d'une séquence de suivi d'objet défini par un modèle 3D filaire [Comport 2003]. Les arêtes du modèle sont échantillonnées et les points noirs illustrent une zone où le contour correspondant au modèle n'est pas trouvé. Il est ici occulté par le bouton de l'interrupteur.

Re-Weighted Least Squares : IRLS). L'erreur à réguler prend alors la forme de :

$$\mathbf{e} = \mathbf{D}(\mathbf{s}(\mathbf{r}) - \mathbf{s}^*), \quad (1.21)$$

où  $\mathbf{D}$  est une matrice diagonale de pondération ayant pour coefficients  $\mathbf{D} = \text{diag}(w_1, \dots, w_{k_1})$ . Chaque poids  $w_i$  calculé à chaque itération [Comport 2003] spécifie la confiance à placer dans la primitive.

La loi de commande permettant de déplacer la caméra virtuelle de façon robuste en imposant une décroissance exponentielle découplée de  $\mathbf{e}$  devient alors :

$$\mathbf{v} = -\lambda(\mathbf{D}\mathbf{L}_s)^+ \mathbf{D}(\mathbf{s}(\mathbf{r}) - \mathbf{s}^*). \quad (1.22)$$

L'asservissement visuel virtuel a aussi été utilisé pour le suivi d'objet par modèle 3D en stéréovision perspective [Dionnet 2007] et en vision omnidirectionnelle monoculaire [Marchand 2007]. Dans le premier cas, la difficulté est de réussir à prendre en compte les mesures des deux vues pour faire évoluer l'unique pose du système stéréo tandis que dans le second cas, la difficulté est de reformuler la distance point-droite dans l'image omnidirectionnelle, ce qui nécessite aussi une nouvelle matrice d'interaction, car, dans ce type d'image, la projection d'une droite devient courbe, sauf cas particulier. Ces deux derniers travaux sont détaillés plus avant dans le chapitre 2 dans la section où il est question de suivi et de calcul de pose simultanés d'objet par modèle 3D dans les images de stéréovision omnidirectionnelle, une des contributions de cette thèse. La question du lieu de calcul de l'erreur dans le plan image [Marchand 2007] et sur la sphère y est aussi abordée.

Pour conclure cette partie, on observe que plusieurs travaux d'asservissement visuel, réel ou virtuel, en vision omnidirectionnelle définissent des approches dans le plan image omnidirectionnel et que les résultats obtenus sont fonctionnels, même s'il

apparaît plus correct et même plus simple de travailler sur la sphère d'équivalence. Cependant, qu'une approche sphérique ou planaire soit utilisée, elles ne sont jamais confrontées. Nous proposons de faire une comparaison des deux approches et ce, pour plusieurs types de primitive. Ce sera l'objet du chapitre 2 pour le calcul de pose par AVV et du chapitre 3 pour l'asservissement visuel photométrique.

#### 1.2.4 Estimation de pose relative

Lorsque l'environnement est inconnu, le calcul de pose devient relatif à une position de référence de la caméra, définie par une image. La pose relative est alors estimée entre une image clef et l'image courante. En général, cette pose relative est déterminée grâce à un ensemble de primitives dont les correspondances sont réalisées entre les deux images. Ce genre d'approche passe par le calcul d'une transformation projective dont la transformation rigide, c'est-à-dire la translation et la rotation 3D, est ensuite extraite.

A partir d'ensembles de primitives mis en correspondance dans deux images, il est possible de calculer la matrice de transformation projective liant ces deux ensembles de primitives à partir de la géométrie épipolaire (Figure 1.10(a)). Il s'agit de la matrice fondamentale  $\mathbf{F}$  dans le cas non-calibré et de la matrice essentielle  $\mathbf{E}$  dans l'autre cas [Hartley 2003] (matrices  $3 \times 3$ ). Les matrices fondamentale et essentielle lient deux points, respectivement pixellique et métrique, images d'un même point 3D :

$$\mathbf{u}'^T \mathbf{F} \mathbf{u} = 0 \quad \text{et} \quad \mathbf{x}'^T \mathbf{E} \mathbf{x} = 0 \quad (1.23)$$

avec  $\mathbf{u}'$  et  $\mathbf{x}'$ , désignant respectivement le correspondant de  $\mathbf{u}$  et  $\mathbf{x}$  dans la seconde image. En vision perspective, ces matrices peuvent être calculées par la méthode linéaire des 8 points dont le résultat peut ensuite être raffiné par optimisation des coefficients de la matrice [Hartley 2003]. De la matrice essentielle, qui est liée à la matrice fondamentale par la matrice  $\mathbf{K}$ , composée de paramètres intrinsèques, on peut extraire par décomposition en valeurs singulières (Singular Value Decomposition : SVD) la rotation et la translation entre les deux positions de caméra où les images ont été acquises.

Cependant, ces relations ne sont valides que dans le cas où les primitives utilisées pour calculer la transformation ne sont pas coplanaires. Dans le cas contraire, la matrice fondamentale est dégénérée. Si les primitives appartiennent à un même plan alors la relation projective qui les lie est l'homographie :

$$\mathbf{u}' \propto \mathbf{H}_u \mathbf{u} \quad \text{ou} \quad \mathbf{x}' \propto \mathbf{H}_x \mathbf{x} \quad (1.24)$$

à un facteur d'échelle près (Figure 1.10(b)).  $\mathbf{H}_x$  est définie dans le repère image normalisé et  $\mathbf{H}_u$  dans le pixelique. Ces deux matrices sont liées par la relation  $\mathbf{H}_u = \mathbf{K} \mathbf{H}_x \mathbf{K}^{-1}$ . En agencant en une matrice les coordonnées de points correspondant entre deux vues, la SVD donne une valeur approximative de  $\mathbf{H}$  qui peut ensuite être raffinée par optimisation [Hartley 2003].

En vision omnidirectionnelle, avec le modèle sphérique, ces relations restent valides dans le cas calibré où il est possible de travailler sur la sphère d'équivalence

car la matrice essentielle et l'homographie lient en fait des lignes de vue. Cependant, des travaux ont formulé une matrice fondamentale  $4 \times 4$  dans le plan image omnidirectionnel [Geyer 2002].

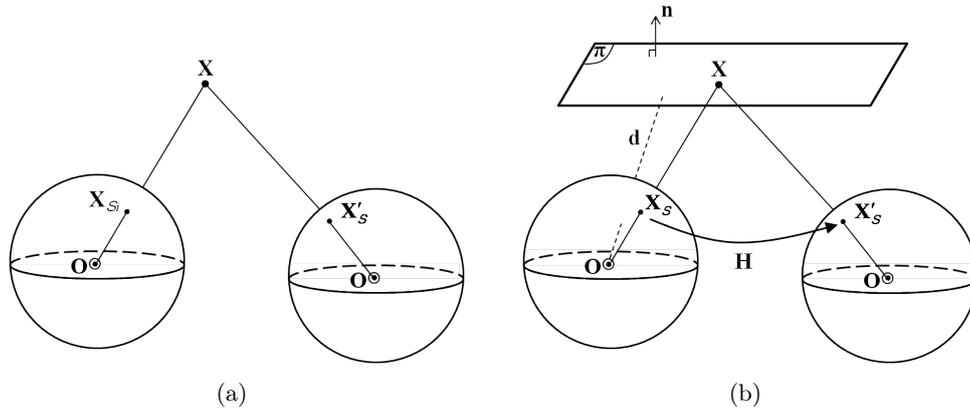


FIGURE 1.10 – Relations de géométrie projective entre deux points correspondants. (a) Relation épipolaire : les points sphériques  $\mathbf{X}_S$  et  $\mathbf{X}'_S$  sont liés par la matrice essentielle sphérique,  $\mathbf{X}'_S{}^T \mathbf{E} \mathbf{X}_S = 0$ . (b) Relation homographique : les points sphériques  $\mathbf{X}_S$  et  $\mathbf{X}'_S$  sont liés par une homographie sphérique,  $\mathbf{X}'_S \propto \mathbf{H} \mathbf{X}_S$ .

Ces méthodes nécessitent de calculer la transformation projective puis d'en extraire la transformation homogène entre les deux caméras. Il existe cependant des approches procédant en deux temps qui estiment la rotation puis la translation [Bazin 2010].

Procéder en plusieurs étapes, en calculant une transformation projective puis en l'éclatant ou en calculant la rotation puis la translation, permet de simplifier les relations géométriques (et les calculs qui en découlent), que la formulation soit linéaire ou non. Cependant, chaque étape produit une approximation d'une solution partielle, ce qui peut mener à des situations où l'accumulation d'approximations engendre une imprécision sur l'estimation finale de la pose relative. Par conséquent, il peut s'avérer intéressant de calculer explicitement la pose relative à partir des primitives de l'image même si ce calcul doit s'accompagner de l'estimation de la structure 3D simultanément.

### 1.2.5 Calcul de pose relative et reconstruction 3D simultanés

Ce type de méthodes consiste à reconstruire l'environnement et la pose relative, qui est directement la transformation rigide 3D, successivement ou simultanément. Ce problème d'estimation de la structure de l'environnement à partir du mouvement [Hartley 2003] (Structure From Motion : SFM) est généralement traité suivant les étapes suivantes :

1. détection de primitives géométriques
2. mise en correspondance des primitives entre vues

3. estimation du mouvement de caméra
4. reconstruction 3D de la structure de l'environnement

Une étape finale d'optimisation des primitives 3D et des positions de caméra (Figure 1.11) peut venir compléter ce processus, qu'il soit local ou sur toute une séquence, comme l'ajustement de faisceaux [Triggs 2000] connu aussi sous le nom Bundle Adjustment : BA.

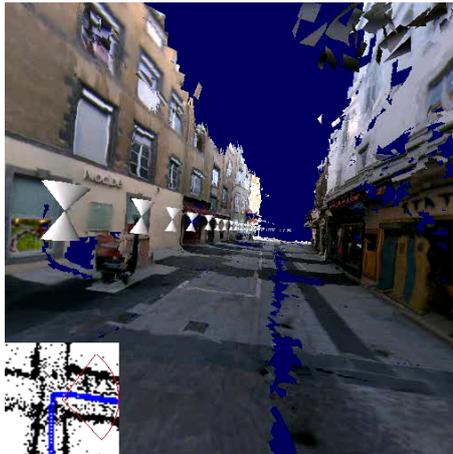


FIGURE 1.11 – Image de reconstruction 3D texturée d'environnement et de position de caméra obtenue par ajustement de faisceaux [Lhuillier 2008]. Ce résultat est obtenu en optimisant la position de chaque caméra et de chaque primitive sur la séquence entière. Une triangulation est effectuée pour obtenir un maillage de l'environnement qui est ensuite texturé à partir des images omnidirectionnelles de la séquence.

Si l'on traite une séquence vidéo dont le mouvement entre chaque image est assez faible, plutôt que de faire la détection et la mise en correspondance entre deux images, faire le suivi des primitives de la première image dans la seconde peut s'avérer plus rapide. Si le suivi des primitives ne se passe que dans l'image, il faudra alors reprendre la chaîne de traitements précédente et la transformation 3D n'est toujours pas directement estimée. Cependant, simultanément au suivi de chaque primitive, il est possible de les reconstruire en 3D pour à la fois obtenir la structure de l'environnement et la transformation homogène entre les caméras. Concrètement, il faut que les primitives image soient assez riches pour pouvoir reconstruire son correspondant 3D. Si elles le sont, la localisation de la caméra et la reconstruction de l'environnement simultanés sont alors réalisables. Le SLAM (Simultaneous Localization And Mapping) réalise en partie ce processus sous le formalisme du filtre de Kalman en estimant chaque primitive 3D (points [Karlsson 2005], droites [Lemaire 2007], plans [Silveira 2008]), les poses de la caméra et l'incertitude de ces estimations, à l'aide de prédictions obtenues par un ou plusieurs autres capteurs comme les odomètres, un laser, des sonars, etc.

Dans ce type de processus, la vision omnidirectionnelle a un avantage : le suivi d'une même primitive pendant une longue période d'une séquence, ce qui est gage de précision. La stéréovision, quant à elle, apporte de la robustesse puisqu'une même primitive est perçue par plusieurs caméras au même instant. Par conséquent, et cela est vrai pour toute méthode de calcul de pose, de pose relative et de reconstruction simultanée, la stéréovision omnidirectionnelle est gage de précision et de robustesse dans tout processus de suivi, pour peu que la géométrie de l'image omnidirectionnelle soit bien exploitée. Ce dernier point est peut être la raison pour laquelle les travaux de suivi, de calcul de pose relative et de reconstruction simultanés restent marginaux en vision ou stéréovision omnidirectionnelle [Kim 2006]. Il existe certes quelques approches utilisant les modèles de projection génériques [Ramalingam 2006] mais le mouvement et la structure sont obtenus par optimisation sur toute la séquence.

Une autre voie consiste à s'intéresser à l'ensemble des intensités d'une région d'une image correspondant à un plan en 3D. Ce type d'approche permet de faire une reconstruction dense du plan observé et d'estimer le mouvement de la caméra. Le suivi et le calcul du mouvement se font simultanément en vision perspective [Benhimane 2007] comme en vision omnidirectionnelle [Mei 2008, Salazar-Garibay 2009]. Le calcul du mouvement peut se faire par homographie, qui englobe le mouvement de la caméra et le plan 3D (normale et distance), la représentation la plus appropriée pour un plan, ou en estimant explicitement la rotation, la translation et le plan 3D. Cette dernière formulation permet, de plus, de réaliser le suivi et la reconstruction de plusieurs plans dans l'image, pour un seul mouvement de caméra, ce qui est particulièrement intéressant en vision omnidirectionnelle [Mei 2008]. En effet, grâce au large champ de vue, plusieurs plans sont généralement visibles en environnement structuré et leur suivi permet une estimation plus précise du mouvement car plus contrainte (Figure 1.12). Cependant, cette estimation reste à l'échelle près en vision monoculaire, c'est pourquoi une extension à la stéréovision omnidirectionnelle sera présentée dans le chapitre 4, pour combler cette lacune, en plus d'une formulation totalement sphérique.

## 1.3 Choix et représentation des primitives

### 1.3.1 Lieux de définition et représentations des primitives

Estimer le mouvement d'une caméra ou reconstruire un environnement en 3D pose le problème du choix de primitive (points, droites, plans, etc) mais aussi de leur représentation. Un point doit-il être représenté en coordonnées cartésiennes ou polaires dans le plan image ? Si la projection est modélisée par la projection stéréographique, faut-il travailler sur la sphère ? Si oui, faut-il utiliser la représentation sphérique cartésienne ou sphérique pure ?

Il est difficile d'avancer une proposition intuitive sur ces problèmes et quelques travaux ont été menés pour répondre à ces interrogations. En vision perspective, là où le cartésien se comporte mal, le polaire se comporte bien et vice-versa. Cela a inspiré des stratégies de combinaison des deux représentations [Corke 2009]. En

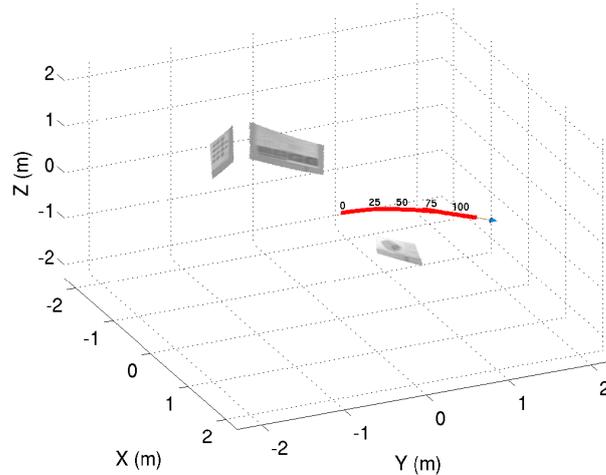


FIGURE 1.12 – Trajectoire estimée par le suivi simultané de trois plans [Mei 2008].

modélisation sphérique, la représentation angulaire offre une formulation découplée pour l’asservissement visuel de robot mobile sur un point [Tatsambon Fomena 2008], ce qui en fait la représentation de prédilection pour cette application mais qu’en est-il pour un asservissement visuel à six degrés de liberté ou un AVV ? Travailler sur la sphère permettant de manipuler directement les lignes de vue, contrairement au plan image, le calcul linéaire de pose est possible par la méthode présentée dans la partie 1.2.1. La formulation, le comportement et la précision de calcul de pose par AVV basé points ou droites en représentation image cartésienne, sphérique cartésienne et sphérique pure sont évalués et confrontés dans le chapitre 2.

Enfin, ces questions de lieux de définition des primitives (plan image ou sphère) et de représentation ne se posent pas uniquement pour les approches géométriques mais aussi pour les approches photométriques. Le suivi dense de plan et l’asservissement visuel photométrique en vision omnidirectionnelle nécessitent le calcul de gradients. Il a été récemment montré que le filtrage d’image omnidirectionnelle donne de meilleurs résultats sur la sphère [Demonceaux 2009]. Cependant, comme il en a brièvement été question dans ce chapitre, une méthode de suivi dense de plans 3D dans les images omnidirectionnelles s’est montrée très performante [Mei 2008]. Mais aucune étude n’a été dédiée à la comparaison entre travailler avec l’image plane ou l’image sphérique. Nous proposons alors d’utiliser et de comparer les trois représentations pour l’asservissement visuel photométrique au chapitre 3 et le suivi photométrique de plan formulé sur l’image sphérique et étendu à la stéréo au chapitre 4.

### 1.3.2 Appariement, pose, reconstruction : vers une simultanéité

Tout au long de ce chapitre, il a été question de mise en correspondance ou de suivi de primitives, de reconstruction 3D, d’estimation de pose, relative ou non. Ces processus sont souvent répartis en étapes qui s’exécutent successivement mais toutes

ces notions sont fortement connexes puisqu'avec la connaissance de l'une d'elles, on peut généralement obtenir les autres. Ainsi, à partir de correspondances 2D-3D, on peut estimer une pose et inversement, à partir d'une pose, on peut projeter des primitives 3D pour trouver leur correspondant 2D dans l'image, ou remonter de l'image à la 3D. Il en va de même pour la pose relative qui, si elle est connue, permet de transférer une primitive d'une image à l'autre, qui inversement, permet d'obtenir la reconstruction 3D, et ainsi de suite.

De ce fait, il est très tentant d'écrire que traiter de mise en correspondance, d'estimation de pose ou de reconstruction 3D est équivalent, voire identique. Le problème est le même mais vu sous un angle différent, comme lorsqu'il est question de suivi ou d'appariement de primitives entre deux images, les deux approches réalisant la même chose mais à des échelles généralement différentes.

Une idée intéressante est alors de détecter des primitives dans une vue et de réaliser simultanément la recherche de correspondances dans une autre vue, l'estimation du mouvement de caméra entre les deux vues et la reconstruction 3D des primitives. Ce sujet est développé dans le chapitre 4.



# Asservissement visuel virtuel géométrique

---

## Sommaire

---

<b>2.1</b>	<b>Introduction</b>	<b>27</b>
<b>2.2</b>	<b>Calcul de pose basé points</b>	<b>28</b>
2.2.1	AVV sur le plan image	30
2.2.2	AVV sur la sphère	31
2.2.3	Propagation d'incertitude	33
2.2.4	AVV basé points en stéréovision	34
2.2.5	Applications, comparaisons et validation	35
<b>2.3</b>	<b>Étalonnage de caméras</b>	<b>43</b>
2.3.1	Étalonnage d'une caméra omnidirectionnelle	43
2.3.2	Étalonnage de banc stéréoscopique	45
2.3.3	Stéréovision hybride	46
2.3.4	Résultats	47
<b>2.4</b>	<b>Suivi et calcul de pose basé modèle 3D filaire</b>	<b>57</b>
2.4.1	AVV de modèle 3D dans le plan image	57
2.4.2	AVV de modèle 3D en représentation sphérique cartésienne	63
2.4.3	Applications et comparaisons	64
<b>2.5</b>	<b>Conclusion</b>	<b>69</b>

---

## 2.1 Introduction

Les problèmes de calcul de pose, basés points ou droites, et d'étalonnage de caméras se ramènent à un problème d'optimisation non-linéaire que l'on propose de résoudre en utilisant un même formalisme : l'asservissement visuel virtuel.

L'asservissement visuel virtuel fait évoluer la pose de la caméra par rapport au monde, ou à un objet de référence, à partir d'une pose initiale. L'évolution de la pose est guidée par le mouvement des primitives dans l'image. Une étape préliminaire d'appariement des primitives est nécessaire pour connaître la correspondance 2D-3D (image-modèle 3D) pour le calcul de pose. Cette correspondance peut soit être réalisée à chaque image, soit être réalisée une seule fois et un suivi des primitives dans les autres images de la séquence permet alors de garder cette correspondance. Nous nous plaçons dans le second cas de figure.

## 2.2 Calcul de pose basé points

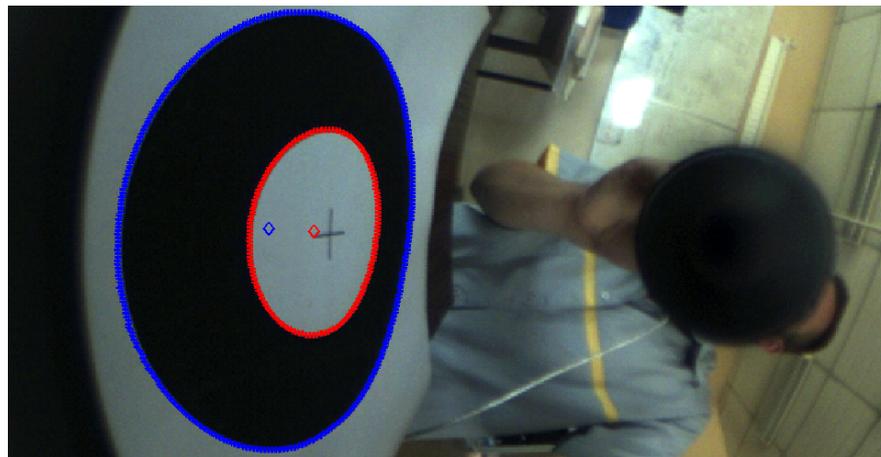
Le suivi de primitive géométrique est généralement réalisé en cherchant la correspondance la plus proche dans la nouvelle image. Pour suivre un point dans une nouvelle image, une recherche est effectuée dans son voisinage afin de trouver le point correspondant. Cette opération est facilitée par l'utilisation de motifs particuliers à suivre comme un disque blanc sur fond noir (Figure 2.1), dont le point central est utilisé comme primitive.



FIGURE 2.1 – Image omnidirectionnelle d'une cible à suivre pour les expérimentations.

Il est aisé de suivre ce type de motif dans une séquence d'images mais considérer que le centre de gravité de l'image du disque est le projeté du point central du disque, comme c'est souvent le cas, est erroné. En effet, la projection d'un cercle 3D (le contour du disque) dans une image omnidirectionnelle donne une quartique plane [Tatsambon Fomena 2008] dont l'origine diffère du centre de gravité de l'image du disque (Figure 2.2(a)). C'est aussi le cas en vision perspective où un cercle 3D se projette en conique dans l'image. Un seul cercle ne permet pas de retrouver le projeté du centre du cercle 3D mais il en faut deux concentriques [Kim 2005]. Cependant, cette différence entre le centre de gravité de l'image du disque et le projeté du centre du disque 3D n'apparaît clairement que lorsque l'image du disque a une surface très importante dans l'image. Inversement, si le disque 3D se projette dans l'image en une petite surface, le centre de gravité et le centre réel de l'image du disque sont presque confondus (Figure 2.2(b)). L'approximation est donc faible et acceptable dans ce dernier cas de figure qui est le nôtre dans les expérimentations réalisées pour l'AVV (Asservissement Visuel Virtuel) basé sur les points. Le centre de gravité

est donc utilisé en tant que primitive “point” et est toujours calculé dans l’image, que ce soit pour l’AVV dans le plan image ou sur la sphère. Dans ces derniers cas, ce point est élevé sur la sphère. Cette façon de procéder permet d’avoir les mêmes données source pour les différentes formulations afin de comparer équitablement le comportement du calcul de pose pour les différentes représentations.



(a)



(b)

FIGURE 2.2 – Comparaison entre le centre de gravité de la forme résultant du projeté du cercle 3D dans l’image (losanges superposés) et le projeté du cercle 3D (la croix au centre de l’anneau) pour (a) un motif de taille importante dans l’image et (b) un motif de taille plus modeste.

Comme il a été introduit dans le premier chapitre (partie 1.2.3), l’asservissement visuel virtuel repose sur l’asservissement visuel réel. Pour le formuler pour un type de primitive, il faut en formuler un calcul d’erreur ainsi que la matrice d’interaction associée qui lie le mouvement des primitives dans l’image à la vitesse de déplacement de la caméra virtuelle. En considérant le point comme primitive, plusieurs lieux de

calcul d'erreur sont envisageables. Il y a naturellement le plan image mais aussi la sphère (représentation cartésienne ou angulaire), dans la modélisation sphérique unifiée pour les caméras centrales. La régulation de l'erreur en ces différents lieux nécessite de formuler une matrice d'interaction pour chaque cas.

### 2.2.1 AVV sur le plan image

L'AVV omnidirectionnel basé point dans le plan image bénéficie des travaux d'asservissement visuel omnidirectionnel [Barreto 2003a, Hadj-Abdelkader 2006]. Le calcul d'erreur se fait de la même façon qu'en vision perspective :

$$\mathbf{e} = \mathbf{x}(\mathbf{r}) - \mathbf{x}^* \quad (2.1)$$

où,  $\mathbf{x}^*$  est le vecteur de points désirés dans l'image,  $\mathbf{r}$  la pose courante de la caméra et  $\mathbf{x}(\mathbf{r})$ , le vecteur des points obtenus par projection dans l'image de la cible composée de points, pour la pose  $\mathbf{r}$  donnée. Le calcul de pose s'effectue avec une caméra calibrée et l'erreur est calculée dans le plan image normalisé. C'est équivalent au calcul dans le plan image pixelique mais plus efficace car la projection du point 3D pour la pose courante  $\mathbf{r}$  dans le plan image normalisé est directement utilisée sans avoir à la transformer dans l'espace pixelique à chaque itération de l'optimisation. Les points de référence issus des mesures dans l'image sont, quant à eux, transformés une seule fois du plan image pixelique vers le plan image normalisé par la relation  $\mathbf{x} = \mathbf{K}^{-1}\mathbf{u}$ .

La variation de cette erreur ne dépend que de  $\mathbf{x}$  ( $\mathbf{x}^*$  étant fixe) et mène à la matrice d'interaction suivante :

$$\dot{\mathbf{x}} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} \frac{\partial \mathbf{X}}{\partial \mathbf{r}} \frac{d\mathbf{r}}{dt} = \mathbf{L}_x \mathbf{v}, \quad (2.2)$$

avec  $\mathbf{X}$  le point 3D dans le repère caméra.  $\mathbf{L}_x$  est le produit de deux jacobiens dont le premier dépend du modèle de projection mais pas le second [Espiau 1992] :

$$\frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \begin{pmatrix} \frac{\rho Z + Y^2 \xi + Z^2 \xi}{(Z + \xi \rho)^2 \rho} & -\frac{X \xi Y}{(Z + \xi \rho)^2 \rho} & -\frac{X(\rho + \xi Z)}{(Z + \xi \rho)^2 \rho} \\ -\frac{X \xi Y}{(Z + \xi \rho)^2 \rho} & \frac{\rho Z + X^2 \xi + Z^2 \xi}{(Z + \xi \rho)^2 \rho} & -\frac{Y(\rho + \xi Z)}{(Z + \xi \rho)^2 \rho} \end{pmatrix} \quad (2.3)$$

et

$$\frac{\partial \mathbf{X}}{\partial \mathbf{r}} = (-\mathbf{I}_{3 \times 3} \quad [\mathbf{X}]_{\times}). \quad (2.4)$$

Ainsi, après quelques manipulations, on obtient la matrice d'interaction associée à un point du plan image omnidirectionnel [Hadj-Abdelkader 2006] :

$$\mathbf{L}_x = \begin{pmatrix} -\frac{1+x^2(1-\xi(\alpha+\xi))+y^2}{\rho(\alpha+\xi)} & \frac{\xi xy}{\rho} & \frac{\alpha x}{\rho} & xy & -\frac{(1+x^2)\alpha-\xi y^2}{\alpha+\xi} & y \\ \frac{\xi xy}{\rho} & -\frac{1+y^2(1-\xi(\alpha+\xi))+x^2}{\rho(\alpha+\xi)} & \frac{\alpha y}{\rho} & \frac{(1+y^2)\alpha-\xi x^2}{\alpha+\xi} & -xy & -x \end{pmatrix}, \quad (2.5)$$

avec  $\alpha = \sqrt{1 + (1 - \xi^2)(x^2 + y^2)}$ .

### 2.2.2 AVV sur la sphère

#### 2.2.2.1 AVV en représentation sphérique pure

A cause de la géométrie particulière des images omnidirectionnelles, le calcul d'erreur dans le plan image n'est pas tout à fait rigoureux. En effet, en vision perspective, que l'on calcule une distance pixelique au centre ou vers l'extérieur de l'image, elle correspond, approximativement, à la même distance réelle, pour de faibles distorsions. Ce n'est pas le cas en vision omnidirectionnelle car les distorsions sont très importantes et une même longueur dans l'espace, à une même distance de la caméra, n'engendre pas, une fois projetée dans l'image, la même longueur pixelique si elle est projetée au centre ou vers l'extérieur de l'image.

Pour corriger cet effet, il faudrait adapter le calcul de la distance en fonction de la zone de l'image avec, par exemple, une pondération de la distance pixelique en fonction de la position du point de référence dans l'image omnidirectionnelle. Une autre façon de faire est de remonter les points sur la sphère du modèle de projection unifié où le calcul de distances est uniforme.

Pour représenter un point sur la surface de la sphère, deux coordonnées d'azimut  $\theta$  et d'élévation  $\phi$  suffisent. On calcule ces deux angles à l'aide de l'élévation d'un point du plan image normalisé (équation 1.7) :

$$\mathbf{S} = \begin{pmatrix} \phi \\ \theta \end{pmatrix} = \begin{pmatrix} \arccos(Z_S) \\ \arctan(Y_S/X_S) \end{pmatrix}. \quad (2.6)$$

L'erreur à réguler pour le calcul de pose sphérique est alors :

$$\mathbf{e} = \mathbf{S}(\mathbf{r}) - \mathbf{S}^*, \quad (2.7)$$

et puisque  $\mathbf{S}^*$  est fixe, la variation de cette erreur ne dépend que de  $\mathbf{S}$  et mène à :

$$\dot{\mathbf{S}} = \frac{\partial \mathbf{S}}{\partial \mathbf{X}} \frac{\partial \mathbf{X}}{\partial \mathbf{r}} \frac{d\mathbf{r}}{dt} = \mathbf{L}_S \mathbf{v}. \quad (2.8)$$

Le détail a été rappelé pour bien illustrer le fait qu'un seul des deux jacobiens, dont le produit mène à la matrice d'interaction, est à ré-exprimer. Le modèle de projection de l'espace vers l'image a bien changé puisque la projection est uniquement sphérique. La matrice d'interaction  $\mathbf{L}_S$  liée à un point sphérique est donc [Tatsambon Fomena 2008] :

$$\mathbf{L}_S = \begin{pmatrix} -\frac{\cos \theta \cos \phi}{\rho} & -\frac{\sin \theta \cos \phi}{\rho} & \frac{\sin \phi}{\rho} & \sin \theta & -\cos \theta & 0 \\ \frac{\sin \theta}{\rho \sin \phi} & -\frac{\cos \theta}{\rho \sin \phi} & 0 & \frac{\cos \theta \cos \phi}{\sin \phi} & \frac{\sin \theta \cos \phi}{\sin \phi} & -1 \end{pmatrix}. \quad (2.9)$$

#### 2.2.2.2 AVV en représentation sphérique cartésienne

Même si la représentation sphérique pure à deux angles est la représentation minimale sur la sphère, il existe une singularité quand  $\mathbf{X}_S = \mathbf{Y}_S = 0$ . Dans ce cas  $\phi = 0$  et  $\theta$  n'est pas défini. Ce cas de figure intervient quand le point en question est au centre de l'image, ce qui ne pose pas de problème avec une caméra catadioptrique

où le centre de l'image est un angle mort mais cette représentation pose problème quand une caméra fisheye est utilisée, puisqu'il est tout à fait possible d'avoir à traiter un point se trouvant au centre de l'image.

La représentation sphérique cartésienne  $\mathbf{X}_S$  est certes redondante mais n'a pas de singularité pour exprimer un point sur la sphère. Le calcul d'erreur peut se faire par le calcul de l'angle entre deux lignes de vue (courante et désirée), par le calcul de la distance géodésique entre deux points sphériques ou par le calcul de la corde entre les deux points. Cependant, le calcul de l'angle passe par l'utilisation d'un arccos dont la dérivation mène à un système présentant une singularité de représentation [Mei 2006]. Une autre piste serait d'utiliser le produit scalaire entre les lignes de vues désirée et courante mais on perd alors le signe de l'erreur. La mesure retenue est finalement celle de la corde entre le point courant et le point désiré car elle permet d'utiliser les méthodes classiques d'optimisation de type Gauss-Newton et les équations et jacobiens qui en découlent sont plus légers que pour la mesure de l'arc de sphère. L'erreur sphérique cartésienne à minimiser est donc :

$$\mathbf{e} = \mathbf{X}_S(\mathbf{r}) - \mathbf{X}_S^*. \quad (2.10)$$

Connaissant la fonction de projection sphérique (équation 1.5), la matrice d'interaction liée à un point sphérique en coordonnées cartésiennes est [Hamel 2002] :

$$\mathbf{L}_{\mathbf{X}_S} = \begin{pmatrix} \frac{1}{\rho}(\mathbf{X}_S \mathbf{X}_S^\top - \mathbf{I}_{3 \times 3}) & [\mathbf{X}_S]_\times \end{pmatrix}. \quad (2.11)$$

### 2.2.2.3 Amélioration du conditionnement de la matrice d'interaction

Un problème de conditionnement du système servant à estimer une transformation 3D ou un mouvement dans l'image, est souvent rencontré. Un mauvais conditionnement est généralement dû à une mauvaise répartition des points, que ce soit dans le plan image ou sur la sphère d'équivalence, et mène à un résultat imprécis. Une étape de pré-conditionnement en centrant et mettant à l'échelle les nuages de points correspondants dans deux images perspectives permet d'améliorer sensiblement l'estimation de matrice fondamentale ou d'homographie [Hartley 2003]. Cependant, si une translation et l'application d'un facteur d'échelle sur un nuage de points dans une image perspective le pré-conditionne très bien, les distorsions importantes des images omnidirectionnelles impliquent l'utilisation de méthodes de pré-conditionnement plus complexes, prenant en compte la résolution non uniforme de l'image.

Effectuer ce pré-conditionnement des points sur la sphère permet d'appliquer une approche isotrope qui reste simple et rapide en temps de calcul. Il est, certes, possible de lever les points sur la sphère, d'appliquer la transformation et de les reprojeter dans le plan image omnidirectionnel pour calculer la transformation. Cependant, il est clair qu'utiliser une approche sphérique, qu'elle soit pure ou cartésienne, est plus efficace, puisqu'il y a au minimum une projection en moins, de la sphère vers le plan image, après pré-conditionnement.

Dans le cas de figure présent, seul un changement d'échelle est appliqué pour pré-conditionner les points sur la sphère. De ce fait, seule la translation de la transformation  ${}^c\mathbf{M}_o$  devra subir l'application du facteur d'échelle inverse après estimation de la transformation 3D.

Une transformation est appliquée sur les lignes de vue de telle sorte qu'elles forment, en moyenne, un angle de  $45^\circ$  avec leur vecteur centroïde. Le nombre de conditionnement est en effet élevé si les lignes de vues des points sur la sphère sont très proches et va en décroissant à mesure qu'elles s'éloignent les unes des autres.

Le vecteur centroïde est le vecteur moyen obtenu à partir des lignes de vues. Le facteur d'échelle à appliquer au nuage de points sphériques est ensuite calculé par le rapport  $\frac{\pi}{4}$  sur l'angle moyen entre le vecteur centroïde et les autres. Ensuite, une nouvelle ligne de vue est obtenue par rotation du vecteur centroïde autour du vecteur issu du produit vectoriel entre le centroïde et la ligne de vue et d'un angle égal au produit du facteur d'échelle par l'angle d'origine entre la ligne de vue et le centroïde. Procéder de cette manière est équivalent à rapprocher virtuellement le nuage de points 3D de la caméra d'un facteur égal au facteur calculé précédemment, sans changer la rotation. Ainsi, après le calcul de pose, le facteur d'échelle inverse est appliqué à la translation estimée pour obtenir la pose à la bonne échelle.

Méthode	type de coordonnées	critère
AVV-PI	plan image normalisé	$\mathbf{x} - \mathbf{x}^*$
AVV-SC	sphérique cartésien ( $\mathbf{X}_S$ )	$\mathbf{X}_S - \mathbf{X}_S^*$
AVV-SP	sphérique pur $(\phi, \theta)$	$\begin{pmatrix} \phi \\ \theta \end{pmatrix} - \begin{pmatrix} \phi^* \\ \theta^* \end{pmatrix}$

TABLE 2.1 – Les différents critères formulés et comparés pour l'AVV de caméra omnidirectionnelle.

### 2.2.3 Propagation d'incertitude

Le calcul de pose ne peut être réussi parfaitement à cause des incertitudes sur l'objet observé, des positions des points image, des paramètres de caméra, etc. Une manière de quantifier l'impact de cette méconnaissance est la propagation d'incertitude. De plus, c'est aussi une méthode théorique pour comparer les différents AVV présentés dans ce chapitre.

Pour propager les erreurs sur les positions de points dans l'image (plane ou sphérique), c'est-à-dire le résidu à convergence du processus d'optimisation, la forme implicite  $f(\mathbf{r}, \mathbf{x}) = 0$  de la propagation d'incertitude est utilisée. Pour l'AVV-PI (Tableau 2.1),  $f$  est la fonction liant un point du plan image normalisé  $\mathbf{x}$  aux paramètres de pose  $\mathbf{r}$ . L'équation de propagation utilisée est obtenue par un développement de

Taylor d'ordre 1 de  $f$  :

$$\left(\frac{\partial f}{\partial \mathbf{r}}\right)_{\tilde{\mathbf{r}}, \tilde{\mathbf{x}}} \Sigma_{\mathbf{r}} \left(\frac{\partial f}{\partial \mathbf{r}}\right)_{\tilde{\mathbf{r}}, \tilde{\mathbf{x}}}^{\top} = \left(\frac{\partial f}{\partial \mathbf{x}}\right)_{\tilde{\mathbf{r}}, \tilde{\mathbf{x}}} \Sigma_{\mathbf{x}} \left(\frac{\partial f}{\partial \mathbf{x}}\right)_{\tilde{\mathbf{r}}, \tilde{\mathbf{x}}}^{\top} + f(\tilde{\mathbf{r}}, \tilde{\mathbf{x}}) f(\tilde{\mathbf{r}}, \tilde{\mathbf{x}})^{\top}. \quad (2.12)$$

Pour les AVV-PI, AVV-SC et AVV-SP,  $\frac{\partial f}{\partial \mathbf{x}} = \mathbf{I}$  et on a :

$$\Sigma_{\mathbf{r}} = \left(\frac{\partial f}{\partial \mathbf{r}}\right)_{\tilde{\mathbf{r}}, \tilde{\mathbf{x}}}^{\dagger} \left(\Sigma_{\mathbf{x}} + f(\tilde{\mathbf{r}}, \tilde{\mathbf{x}}) f(\tilde{\mathbf{r}}, \tilde{\mathbf{x}})^{\top}\right) \left(\left(\frac{\partial f}{\partial \mathbf{r}}\right)_{\tilde{\mathbf{r}}, \tilde{\mathbf{x}}}^{\top}\right)^{\dagger}, \quad (2.13)$$

$()^{\dagger}$  étant la pseudo-inverse à droite.

Pour l'AVV-PI,  $f(\mathbf{r}, \mathbf{x}) = \mathbf{x}^* - pr_{\xi}(\mathbf{X}(\mathbf{r}))$ , ce qui signifie la différence entre un point détecté et la projection du point correspondant de l'objet pour la pose  $\mathbf{r}$ . Pour l'AVV-SC,  $f(\mathbf{r}, \mathbf{X}_{\mathcal{S}}) = \mathbf{X}_{\mathcal{S}}^* - pr_{\mathcal{S}}(\mathbf{X}(\mathbf{r}))$  et pour l'AVV-SP,  $f(\mathbf{r}, \mathbf{S}) = \mathbf{S}^* - cart2sph(pr_{\mathcal{S}}(\mathbf{X}(\mathbf{r})))$ , avec  $cart2sph()$ , la fonction permettant de calculer  $(\phi, \theta)$  à partir de  $\mathbf{X}_{\mathcal{S}}$ .

#### 2.2.4 AVV basé points en stéréovision

En considérant un système stéréoscopique classique à deux caméras, dont les positions relatives sont connues, le critère à minimiser doit prendre en compte les informations visuelles de chaque caméra pour calculer la pose du système stéréo. Pour calculer l'erreur dans le plan image, le critère à minimiser en stéréovision est alors [Dionnet 2007] :

$$\Delta = \sum_{k=1}^{k_1} (pr_{\gamma_{m,1}}({}^{c_1}\mathbf{M}_o {}^o\mathbf{P}_k) - {}^{c_1}\mathbf{s}_k^*)^2 + \sum_{l=1}^{k_2} (pr_{\gamma_{m,2}}({}^{c_2}\mathbf{M}_{c_1} {}^{c_1}\mathbf{M}_o {}^o\mathbf{P}_l) - {}^{c_2}\mathbf{s}_l^*)^2, \quad (2.14)$$

quelque soit le type de primitive  $\mathbf{P}$ .  $\gamma_{m,j}$  est l'ensemble des paramètres intrinsèques du modèle  $m$  pour la caméra  $j$ . Écrire le critère sous cette forme permet de n'avoir qu'une seule pose à estimer, celle de la première caméra qui est la référence du système, sachant que la seconde caméra est définie par rapport à la première (équation 1.9).

Ce critère peut être étendu pour prendre en compte un système stéréo à quatre caméras et même à  $N$  caméras [Caron 2009], connaissant les  $N-1$  poses de caméra,  ${}^{c_j}\mathbf{M}_{c_1}$  par rapport à la caméra 1 de référence :

$$\Delta = \sum_{j=1}^N \sum_{k=1}^{k_j} (pr_{\gamma_{m,j}}({}^{c_j}\mathbf{M}_{c_1} {}^{c_1}\mathbf{M}_o {}^o\mathbf{P}_k) - {}^{c_j}\mathbf{s}_k^*)^2, \quad (2.15)$$

avec  ${}^{c_1}\mathbf{M}_{c_1} = \mathbf{I}_{3 \times 3}$ .

Pour minimiser l'équation 2.15, connaissant les poses relatives de caméras, un changement de repère du torseur cinématique de la pose peut alors être opéré pour l'exprimer dans le repère de chaque caméra. Par conséquent, si la caméra 1 a un

torseur cinématique  $\mathbf{v}_1$  et la caméra  $j$  a un toreur  $\mathbf{v}_j$ , on peut exprimer ce dernier en fonction du premier [Dionnet 2007] :

$$\mathbf{v}_j = {}^{c_j}\mathbf{V}_{c_1}\mathbf{v}_1, j = 2..N. \quad (2.16)$$

${}^{c_j}\mathbf{V}_{c_1}$  est la matrice de changement de repère du toreur cinématique :

$${}^{c_j}\mathbf{V}_{c_1} = \begin{bmatrix} {}^{c_j}\mathbf{R}_{c_1} & [{}^{c_j}\mathbf{t}_{c_1}]_{\times} \\ 0 & {}^{c_j}\mathbf{R}_{c_1} \end{bmatrix}, \quad (2.17)$$

avec  $[{}^{c_j}\mathbf{t}_{c_1}]_{\times}$ , la matrice anti-symétrique de pré-produit vectoriel du vecteur de translation  ${}^{c_j}\mathbf{t}_{c_1}$ . Ainsi, un mouvement de primitive dans l'image de la caméra  $j$  est lié à la vitesse de la caméra 1 par :

$$\dot{\mathbf{s}}_j = \mathbf{L}_j\mathbf{v}_j = \mathbf{L}_j{}^{c_j}\mathbf{V}_{c_1}\mathbf{v}_1 \quad (2.18)$$

et, pour  $N$  caméras, on a :

$$\begin{bmatrix} \dot{\mathbf{s}}_1 \\ \dot{\mathbf{s}}_2 \\ \vdots \\ \dot{\mathbf{s}}_N \end{bmatrix} = \begin{bmatrix} \mathbf{L}_1 \\ \mathbf{L}_2{}^{c_2}\mathbf{V}_{c_1} \\ \vdots \\ \mathbf{L}_N{}^{c_N}\mathbf{V}_{c_1} \end{bmatrix} \mathbf{v}_1. \quad (2.19)$$

Enfin, on obtient la loi de commande virtuelle suivante avec seulement les 6 paramètres de la pose du système stéréo à estimer :

$$\mathbf{v}_1 = -\lambda \begin{bmatrix} \mathbf{L}_1 \\ \mathbf{L}_2{}^{c_2}\mathbf{V}_{c_1} \\ \vdots \\ \mathbf{L}_N{}^{c_N}\mathbf{V}_{c_1} \end{bmatrix}^+ \begin{bmatrix} \mathbf{s}_1(\mathbf{r}_1) - \mathbf{s}_1^* \\ \mathbf{s}_2(\mathbf{r}_2) - \mathbf{s}_2^* \\ \vdots \\ \mathbf{s}_N(\mathbf{r}_N) - \mathbf{s}_N^* \end{bmatrix}. \quad (2.20)$$

La pose  ${}^{c_1}\mathbf{M}_o$  est mise à jour comme lorsqu'une seule caméra est utilisée (équation 1.19) et les poses des  $N - 1$  autre caméras du système stéréo sont ensuite mises à jour en utilisant les  ${}^{c_j}\mathbf{M}_{c_1}$  issus de l'étalonnage :

$${}^{c_j}\mathbf{M}_o = {}^{c_j}\mathbf{M}_{c_1} {}^{c_1}\mathbf{M}_o. \quad (2.21)$$

Ces dernières poses sont utilisées pour le calcul des  $\mathbf{s}_j(\mathbf{r}_j)$ .

Dans le cas d'un AVV basé points, on a  $\mathbf{P} = \mathbf{X}$  et  $\mathbf{s} = \mathbf{x}$ .

### 2.2.5 Applications, comparaisons et validation

Afin de comparer équitablement les trois approches (Tableau 2.1), le seuil de convergence a été calculé pour être équivalent. Ainsi, la tolérance  $T_{IP}$  de stabilité du résidu en AVV-PI correspond à une erreur maximale d'un dixième de pixel  $T_{IP} = 4 \times 2/(10p_x)^2$  au centre de l'image. La tolérance d'un dixième de pixel est reprojétée sur la sphère d'équivalence par l'équation (1.7) pour calculer  $T_{SC}$ . Enfin, cette tolérance est reportée en  $\theta$  et  $\phi$  pour l'AVV-SP en utilisant les fonctions trigonométriques.

### 2.2.5.1 Suivi de cible

**Suivi monocular** Ce premier résultat est obtenu en utilisant une caméra et une cible composée de disques blancs sur fond noir (Figure 2.1). Le centre de ces disques est considéré comme une primitive “point” et le suivi de ces disques est réalisé par proximité entre images successives d’une séquence. Les AVV-PI, AVV-SC et AVV-SP sont appliqués sur une séquence de 400 images monoculaires dans un premier temps. Le suivi et le calcul de pose réussissent sur toute la séquence et donnent tous trois les mêmes résultats (Figure 2.3(d)) avec une parfaite projection des points de la cible et de son repère dans les images (Figure 2.3).

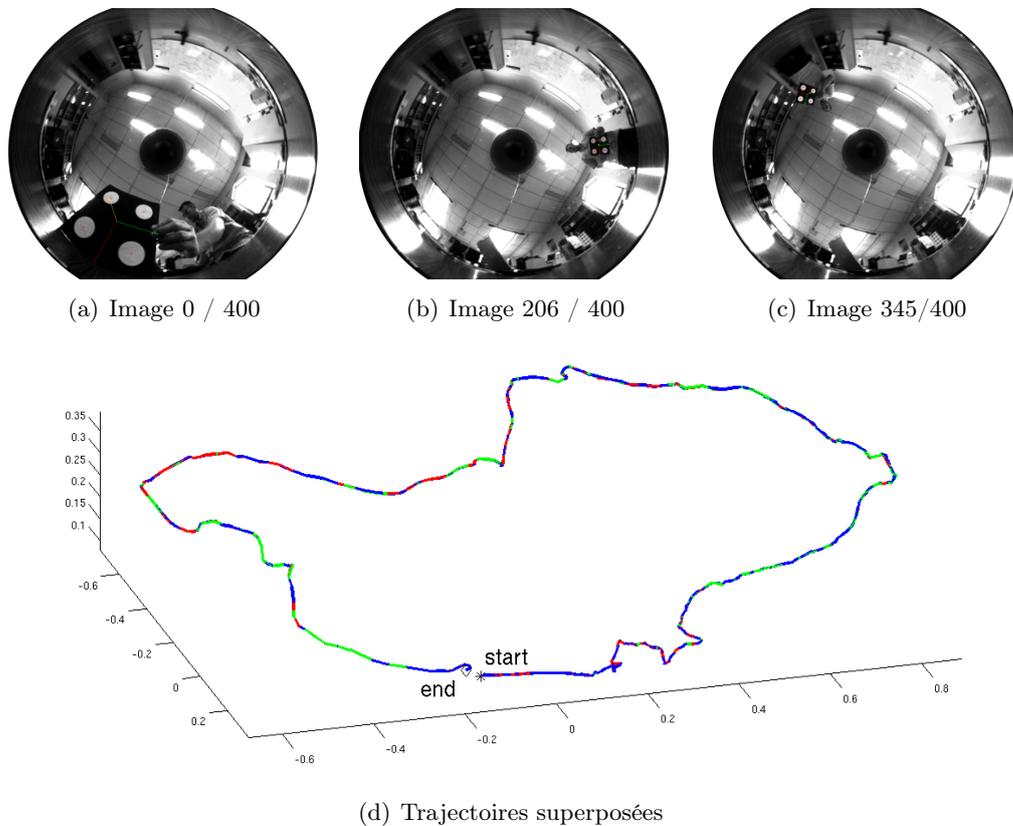
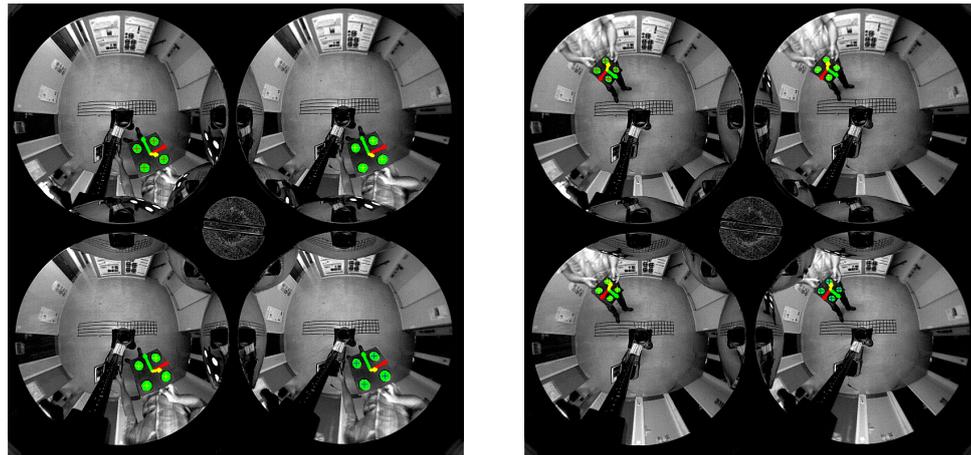


FIGURE 2.3 – Suivi de cible et calcul de pose. La caméra omnidirectionnelle est fixe et la cible connue est déplacée tout autour. (a), (b) et (c) Les quatre points de la cible et son repère sont projetés dans l’image et se superposent parfaitement à l’image de la cible tout au long des 400 images de la séquence. (d) Les trajectoires estimées par les trois méthodes sont identiques.

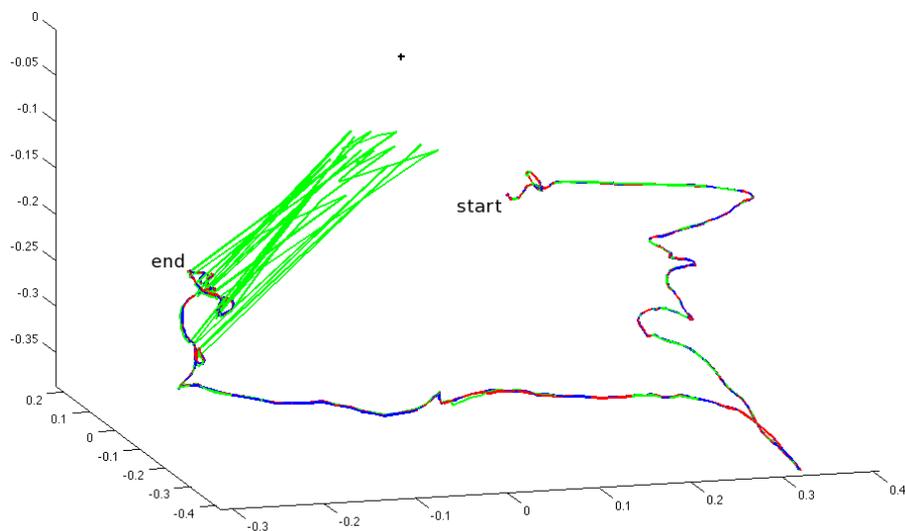
**Suivi stéréo** Les AVV-PI, AVV-SC et AVV-SP sont appliqués sur une séquence de 448 images de stéréovision omnidirectionnelle selon la méthode décrite dans la partie 2.2.4. Le suivi et l’estimation de pose se déroulent parfaitement tout au long de la séquence 2.4 pour les AVV-PI et AVV-SC. L’AVV-SP, quant à lui, souffre de

problèmes de conditionnement en certains points de la trajectoire, particulièrement sur la fin de la séquence, ce qui explique les aberrations visibles sur la trajectoire de la figure 2.4. En effet, le conditionnement atteint des pics à 1100 alors que la moyenne pour la séquence est d'environ 80. Ces problèmes de conditionnement et leur impact sur l'estimation de pose sont abordés plus précisément dans la partie suivante.



(a) Image 0 / 448

(b) Image 300 / 448



(c) Trajectoires superposées

FIGURE 2.4 – Suivi de cible et calcul de pose en stéréovision omnidirectionnelle. Le capteur FOO est fixe et la cible connue est déplacée tout autour. Les informations de chaque miroir sont reportées dans le repère de celui en haut à gauche pour estimer une seule pose. Les quatre points de la cible et son repère sont projetés dans les images et se superposent parfaitement à l'image de la cible tout au long des 448 images de la séquence.

### 2.2.5.2 Localisation externe de robot mobile

Les AVV-PI, AVV-SC et AVV-SP sont appliqués et comparés dans une application de localisation externe de robot mobile.

Le robot mobile embarque une balise composée de quatre diodes infrarouges (Figure 2.5(a)) et est observé par une caméra fisheye (Figure 2.5(c)) équipée d'un filtre infrarouge et positionnée au plafond (Figure 2.5(b)).

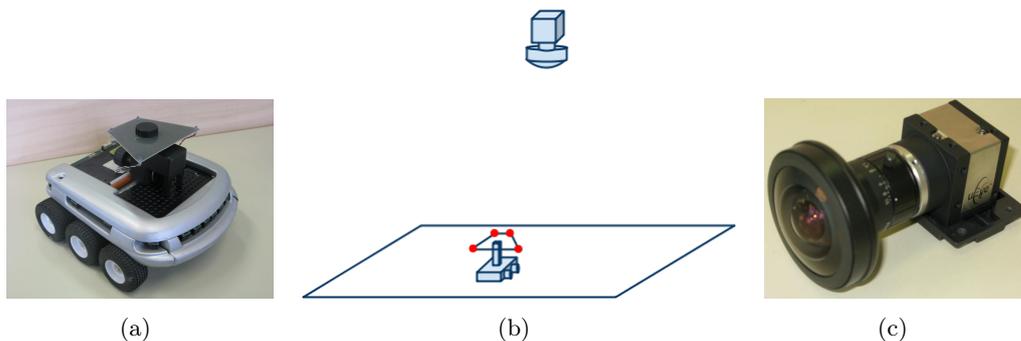


FIGURE 2.5 – Localisation d'un robot en vision omnidirectionnelle. (a) Le robot Koala embarquant une cible de diodes infrarouges. (b) Le système : Le robot mobile embarque une cible suivie dans les images d'une caméra placée en hauteur, directement au plafond ou en haut d'un trépied. (c) Caméra fisheye.

Les diodes infrarouges sont détectées automatiquement et suivies dans les images omnidirectionnelles. La détection automatique se fait par allumage successif des diodes dans un ordre prédéfini et puisqu'avec le filtre infrarouge elles seules sont perçues (Figure 2.6), la détection est grandement simplifiée. Le suivi bénéficie aussi de ces images simples puisque la même technique de suivi par proximité (cf. section 2.2.5.1) peut être appliquée et le centre de gravité de l'image d'une diode considéré comme son centre, pour la correspondance 2D-3D. L'image est cependant pré-traitée avec une binarisation à seuil variable, pour que la surface des diodes dans l'image soit constante, suivie par une érosion pour gérer la visibilité variable des diodes infrarouge ou la fusion de diodes dans l'image. Cette variation est due au cône d'émission infrarouge de la diode.

**Localisation de robot mobile** La balise montée sur le robot est composée de quatre diodes infrarouges placées en trapèze. Les côtés parallèles mesurent respectivement 15 cm et 7 cm et sont distants de 10 cm. Le robot se déplace sur un sol plan. La caméra fisheye est placée au plafond afin de percevoir la surface totale du sol où le robot se déplace. La méthode reste cependant efficace quelque soit l'orientation de la caméra omnidirectionnelle, puisque la pose est estimée en 3D.

Le calcul de pose par AVV est appliqué sur une séquence de 4 minutes 30 comportant 5500 images d'une résolution de  $1280 \times 960$  pixels chacune. Dans cette expérimentation la caméra est placée à 2,50 m du sol et le robot se déplace sur une surface de  $4 \text{ m} \times 3 \text{ m}$ .

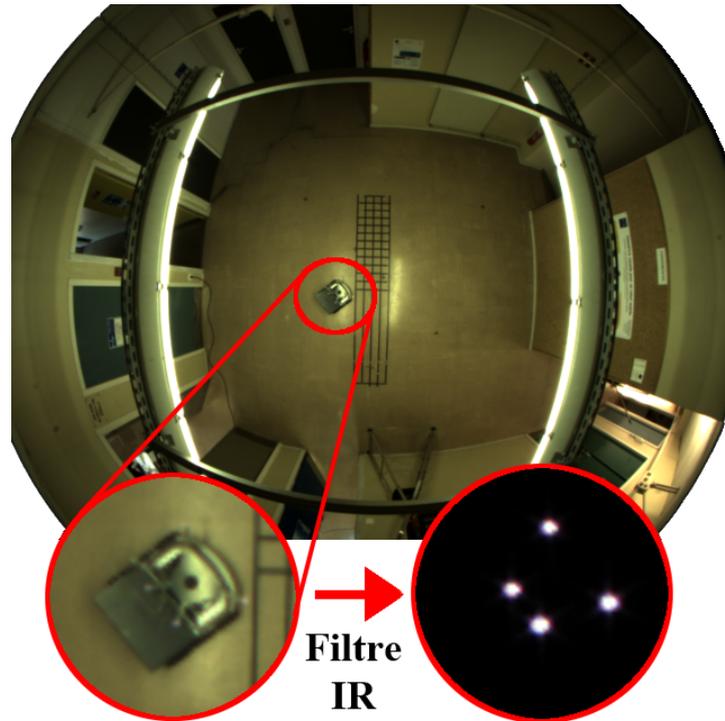


FIGURE 2.6 – Image fisheye avec zoom sur la mire du robot en visible et en infra-rouge.

Après un aller-retour sur une ligne droite, le robot se déplace selon un large rectangle pour un mouvement total d'environ 15 m. Les résultats d'estimation de trajectoire pour les AVV-PI, AVV-SC et AVV-SP (Figure 2.7) sont similaires. Le mouvement réel du robot est totalement planaire. Les écarts types des positions en  $Z$  estimées sont de 3,49 cm pour l'AVV-PI, 3,37 cm pour l'AVV-SC et 4,07 cm pour l'AVV-SP. La méthode AVV-SP donne des résultats moins précis que les deux autres quand le robot passe près de l'axe optique de la caméra. En effet l'estimation de pose par AVV-SP est instable, particulièrement selon l'axe  $Z$  dans cette zone. Ceci est dû à la singularité de la matrice d'interaction quand un point a sa coordonnée  $\phi$  égale à zéro (équation 2.9), ce qui est presque le cas ici.

**Etude du conditionnement** Comparons le conditionnement des matrices d'interactions des AVV-PI, AVV-SC et AVV-SP sans et avec pré-conditionnement pour les deux derniers (Tableau 2.2). Sans pré-conditionnement des points, les AVV-PI et AVV-SC sont mieux conditionnés que l'AVV-SP. Les lignes AVV-SC<sub>p</sub>, AVV-SP<sub>p</sub> du tableau montrent l'impact du pré-conditionnement des points en améliorant près de cinq fois le conditionnement de l'AVV-SC et sept fois l'AVV-SP. Par conséquent, l'AVV-SC<sub>p</sub> a un nombre de conditionnement cinq fois moindre que l'AVV-PI et l'AVV-SP<sub>p</sub> a, quant à lui, un nombre de conditionnement deux fois moindre que l'AVV-PI.

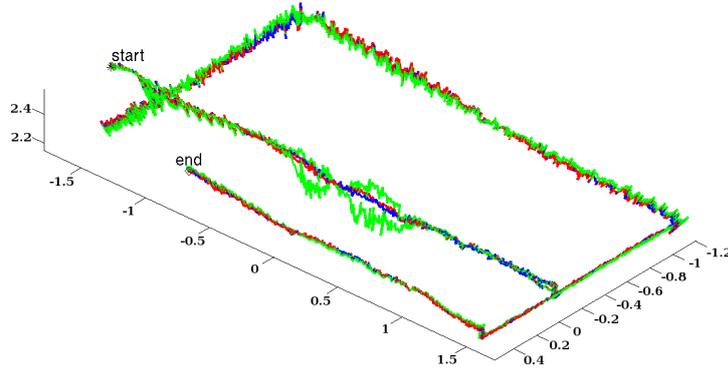


FIGURE 2.7 – Estimation brute de trajectoire par AVV. La trajectoire bleue est estimée par AVV-PI, la rouge par AVV-SC et la verte par AVV-SP. L’unité est le mètre.

Le pré-conditionnement permet de réduire les variations selon l’axe  $Z$  pour l’approche sphérique pure, particulièrement quand le robot s’approche du point principal de l’image. Même si les résultats de l’AVV sphérique pur sont meilleurs avec un pré-conditionnement, l’AVV-SCp mène à des résultats de localisation similaires à ceux de l’AVV-SC (moins de 5 cm de variations 3D en moyenne).

Méthode	moyenne	écart-type
AVV-PI	575,68	305,20
AVV-SC	571,22	310,39
AVV-SP	2076,93	6211,78
AVV-SCp	120,11	92,64
AVV-SPp	312,87	465,79

TABLE 2.2 – Moyennes et écarts types du conditionnement des matrices d’interactions au long d’une séquence dont la trajectoire est présentée en figure 2.7. AVV-SCp et AVV-SPp signifient respectivement AVV-SC pré-conditionné et AVV-SP pré-conditionné.

**Incertitude de localisation** L’incertitude de localisation est calculée en chaque pose par propagation du résidu final à convergence de l’AVV. La figure 2.8 montre la trajectoire estimée dans le plan  $XY$  et les ellipses d’incertitude des AVV-PI, AVV-SC et AVV-SP sont superposées. La surface et l’orientation varient tout au long de la trajectoire mais sans accroissement permanent de la surface puisqu’il n’y a pas d’accumulation d’erreur avec la méthode utilisée. L’axe principal de l’ellipse est toujours orienté vers l’origine du repère, qui correspond à la position de la caméra, ce qui est logique puisque cela correspond à l’incertitude en profondeur qui est généralement toujours la plus importante. Ces observations sont valables pour les AVV-PI, AVV-SC et AVV-SP.

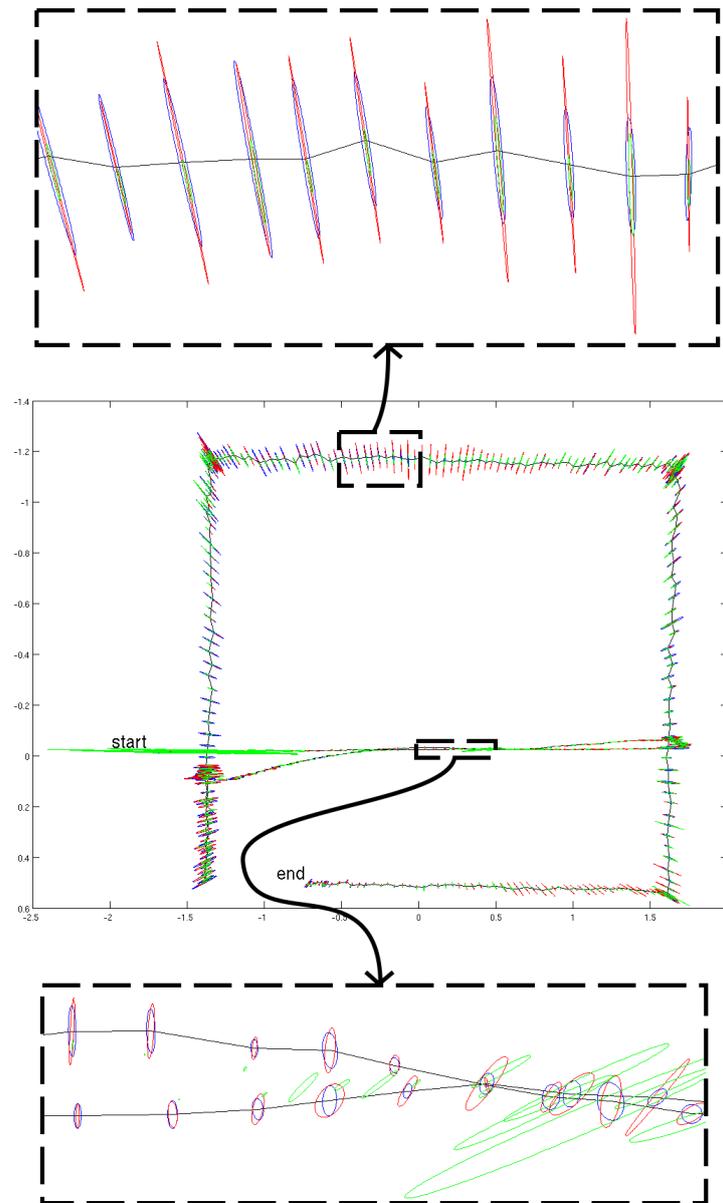


FIGURE 2.8 – Résultats de propagation d'incertitude I. La trajectoire de l'AVV-SC est tracée en noir et les incertitudes en X et Y sont représentées par des ellipses dont les axes font trois fois  $\sigma$  (l'écart-type issu de la matrice de variance-covariance d'incertitude). Les ellipses bleues correspondent à l'AVV-PI, les rouges à l'AVV-SC et les vertes à l'AVV-SP. Deux zones sont grossies, la première (en bas) est proche de la position (0,0) de la caméra et la seconde (en haut) en est éloignée. Les formes d'ellipse dénotent une incertitude plus importante en profondeur qu'en largeur. L'unité est le mètre.

La trajectoire estimée par l'AVV-SP montre une instabilité autour de l'axe op-

tique de la caméra (Figure 2.7), particulièrement selon l'axe  $Z$ . Cela est confirmé par les résultats de la figure 2.9 où les ellipses d'incertitudes des composants YZ couplés sont présentées. La zone d'incertitude maximale de l'AVV-SP se trouve dans un rayon d'une cinquantaine de centimètres autour de l'origine. L'incertitude va ensuite se réduire à mesure que le robot s'éloigne de l'origine. En fait, moins le robot est proche de l'origine, moins  $\phi$  est proche de zéro et la matrice d'interaction est de "moins en moins singulière" pour l'AVV-SP.

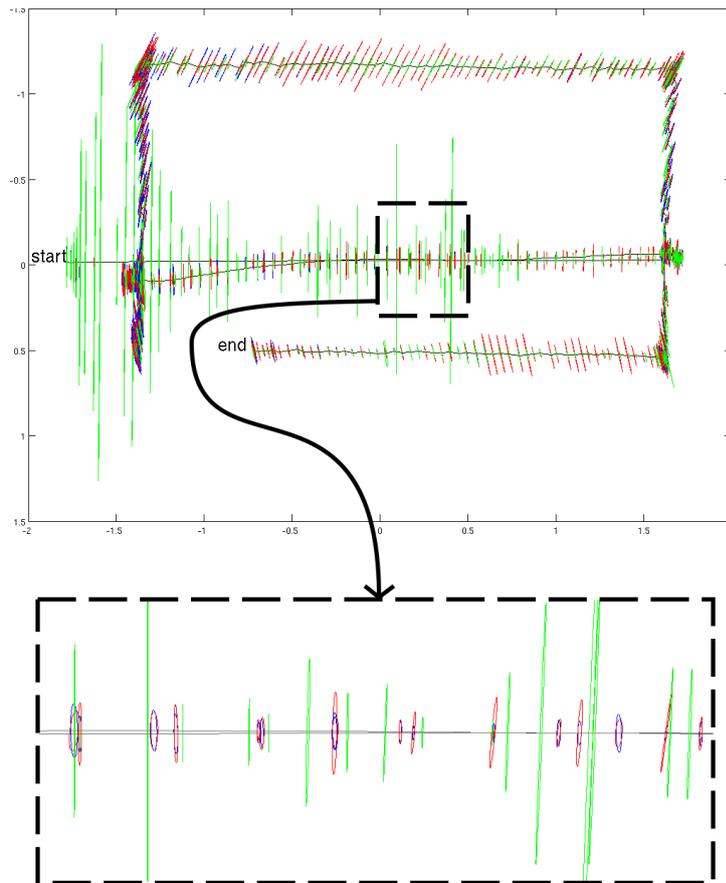


FIGURE 2.9 – Résultats de propagation d'incertitude II. La trajectoire de l'AVV-SC est tracée en noir et les incertitude du bloc YZ de  $\Sigma_r$  est représentée par des ellipses donc les axes font trois fois  $\sigma$  (l'écart-type issu de la matrice de variance-covariance d'incertitude). Les ellipses bleues correspondent à l'AVV-PI, les rouges à l'AVV-SC et les vertes à l'AVV-SP. Pour les ellipses uniquement, l'axe horizontal correspond à l'axe Y et l'axe vertical à l'axe Z, tandis que pour la trajectoire, l'horizontale et la verticale correspondent aux axes X et Y. Une zone est agrandie proche de la position de caméra (0,0). L'unité est le mètre.

**Précision de localisation** Afin de quantifier plus fidèlement la précision de localisation, la méthode d'évaluation décrite dans [Duckett 1998] a été utilisée. La cible composée des quatre diodes infrarouges a été placée dans chaque cellule d'une grille au sol de  $4 \times 10$  cellules et des images ont été acquises pour huit orientations différentes dans chaque cellule. Le principe de cette méthode est de construire une table de contingence et de l'utiliser pour calculer le nombre de chances de succès de la localisation.

L'évaluation de la précision de localisation a été faite avec des cellules de  $10 \text{ cm} \times 10 \text{ cm}$  et 100% des positions ont été estimées dans les bonnes cellules avec les AVV-PI et AVV-SC, menant à un coefficient d'incertitude  $U$  de vraie localisation, sachant la localisation estimée, égal à 1. L'AVV-SCp mène aussi à  $U = 1$ . Cela implique que l'estimation prédit toujours la bonne position du robot avec une précision de 10 cm. Cependant, pour l'AVV-SP,  $U = 0,98$  ce qui implique que la vraie localisation est estimée à 98% pour une précision de 10 cm. Le pré-conditionnement améliore ici la précision d'estimation de l'AVV sphérique pur rendant  $U$  égal à 1.

## 2.3 Étalonnage de caméras

L'étalonnage de caméra est une généralisation du calcul de pose où les paramètres intrinsèques du modèle de projection sont estimés simultanément à la pose. Connaître les paramètres de projection d'une caméra permet de retrouver des informations 3D à partir de mesures image et est nécessaire pour calculer une pose de caméra. Le calcul de pose par AVV peut aussi être étendu à l'étalonnage de caméra, il l'a même déjà été pour les caméras perspectives [Marchand 2001b], ainsi qu'à la stéréoscopie [Caron 2010b]. Sa formulation permet d'aborder l'étalonnage simultané de banc stéréo composé de caméras de différents types et de différents modèles comme, par exemple, perspective et sphérique.

### 2.3.1 Etalonnage d'une caméra omnidirectionnelle

Avec un modèle de projection paramétrique, l'étalonnage de caméra est un processus numérique qui permet de retrouver les valeurs des différents paramètres de ce modèle. L'étalonnage est un problème récurrent en vision omnidirectionnelle [Mei 2006, Barreto 2003b, Scaramuzza 2007] et est généralement traité en utilisant des mires planes composées de points ou de segments de droites dont les coordonnées sont connues dans le repère de la mire. La méthode d'étalonnage présentée ici utilise des images de mires planes composées de points, mais, tout comme le calcul de pose, elle peut être adaptée à tout type de primitive géométrique, dès lors que la matrice d'interaction et la variation des paramètres intrinsèques peuvent être exprimées par rapport à cette primitive.

Pour étendre le calcul de pose par AVV basé points à l'étalonnage, l'erreur à minimiser se calcule dans le plan image pixelique puisque, contrairement au calcul de pose défini dans le plan image, il faut, cette fois-ci estimer aussi les paramètres intrinsèques  $\gamma_m$ ,  $m$  dénotant le modèle de projection à étalonner (1 : perspective et

2 : unifié pour les modèles présentés dans la partie 1.1). Le critère à minimiser est donc très similaire à celui de l'équation 2.1 :

$$\mathbf{e} = \mathbf{u}(\mathbf{r}, \gamma_m) - \mathbf{u}^*. \quad (2.22)$$

Pour l'étalonnage, le mouvement des primitives de l'image est lié à la vitesse de la caméra  $\mathbf{v}$  et à la variation temporelle des paramètres intrinsèques par :

$$\dot{\mathbf{u}} = \frac{\partial \mathbf{u}}{\partial \mathbf{r}} \frac{d\mathbf{r}}{dt} + \frac{\partial \mathbf{u}}{\partial \gamma_m} \frac{d\gamma_m}{dt}. \quad (2.23)$$

Ainsi, pour une image de mire, la loi de commande de l'AVV pour l'étalonnage est :

$$\begin{pmatrix} \dot{\mathbf{v}} \\ \dot{\gamma}_m \end{pmatrix} = -\lambda \left( \mathbf{L}_{\mathbf{u}} \mid \frac{\partial \mathbf{u}}{\partial \gamma_m} \right)^+ (\mathbf{u}(\mathbf{r}, \gamma_m) - \mathbf{u}^*), \quad (2.24)$$

où  $\dot{\gamma}_m$  est la variation des paramètres intrinsèques qui sont mis à jour à l'itération  $it$  par :

$$\gamma_m^{it} = \gamma_m^{it-1} + \dot{\gamma}_m. \quad (2.25)$$

Pour un bon étalonnage, tout le champ de vue du capteur doit être couvert par la mire. Cette mire étant plane et la caméra omnidirectionnelle, il faut utiliser plusieurs images pour bien étalonner la caméra. Ceci permet de plus de maximiser la vraisemblance de l'étalonnage en utilisant des données variées avec différentes positions et orientations de la mire. La loi de commande

$$\begin{pmatrix} \dot{\mathbf{v}}^1 \\ \dot{\mathbf{v}}^2 \\ \vdots \\ \dot{\mathbf{v}}^p \\ \dot{\gamma}_m \end{pmatrix} = -\lambda \begin{pmatrix} \mathbf{L}_{\mathbf{u}}^1 & 0 & \dots & 0 & \frac{\partial \mathbf{u}^1}{\partial \gamma_m} \\ 0 & \mathbf{L}_{\mathbf{u}}^2 & \dots & 0 & \frac{\partial \mathbf{u}^2}{\partial \gamma_m} \\ \vdots & & \ddots & \vdots & \vdots \\ 0 & \dots & \dots & \mathbf{L}_{\mathbf{u}}^p & \frac{\partial \mathbf{u}^p}{\partial \gamma_m} \end{pmatrix}^+ \begin{pmatrix} \mathbf{u}^1 - \mathbf{u}^{1*} \\ \mathbf{u}^2 - \mathbf{u}^{2*} \\ \vdots \\ \mathbf{u}^p - \mathbf{u}^{p*} \end{pmatrix}, \quad (2.26)$$

avec  $\dot{\mathbf{v}}^i$ , le torseur cinématique de la pose de caméra associée à l'image  $i$ ,  $\dot{\gamma}_m$  la variation des paramètres intrinsèques, qui est unique pour l'ensemble des images, et  $\mathbf{u}^{i*}$ , l'ensemble de points détectés dans l'image  $i$ . Il n'y a pas de notation particulière pour préciser le modèle de projection utilisé dans le calcul des matrices  $\mathbf{L}_{\mathbf{u}}^i$ , pour ne pas surcharger l'écriture, mais il va de soi que la fonction de projection qui entre en compte correspond aux paramètres  $\gamma_m$ , soit  $pr_{\gamma_m}()$ .

Avec le modèle perspective comme avec le modèle sphérique, la matrice d'interaction de pose, exprimée dans le plan image pixelique s'écrit en fonction de celle exprimée dans le plan image normalisé [Hutchinson 1996] :

$$\mathbf{L}_{\mathbf{u}} = \begin{pmatrix} \alpha_u & 0 \\ 0 & \alpha_v \end{pmatrix} \mathbf{L}_{\mathbf{x}}, \quad (2.27)$$

où  $\mathbf{L}_{\mathbf{x}}$  fait intervenir le modèle perspective si c'est celui dont on veut estimer les paramètres ou le modèle stéréographique unifié dans le cas contraire.

Pour l'étalonnage d'une caméra perspective, dériver  $\mathbf{u}$  par rapport à  $\gamma_1$  mène à [Marchand 2001b] :

$$\frac{\partial \mathbf{u}}{\partial \gamma_1} = \begin{pmatrix} \frac{\partial u}{\partial \alpha_u} & \frac{\partial u}{\partial \alpha_v} & \frac{\partial u}{\partial u_0} & \frac{\partial u}{\partial v_0} \\ \frac{\partial v}{\partial \alpha_u} & \frac{\partial v}{\partial \alpha_v} & \frac{\partial v}{\partial u_0} & \frac{\partial v}{\partial v_0} \end{pmatrix} = \begin{pmatrix} x & 0 & 1 & 0 \\ 0 & y & 0 & 1 \end{pmatrix}. \quad (2.28)$$

Enfin, pour l'étalonnage d'une caméra centrale, modélisée par la projection stéréographique unifiée (équation 1.8), dériver  $\mathbf{u}$  par rapport à  $\gamma_2$  mène à :

$$\frac{\partial \mathbf{u}}{\partial \gamma_2} = \begin{pmatrix} x & 0 & 1 & 0 & -\frac{\alpha_u \rho x}{Z + \xi \rho} \\ 0 & y & 0 & 1 & -\frac{\alpha_v \rho y}{Z + \xi \rho} \end{pmatrix}. \quad (2.29)$$

### 2.3.2 Etalonnage de banc stéréoscopique

L'étalonnage stéréoscopique peut se faire en estimant successivement les paramètres intrinsèques de chaque caméra, avec une méthode d'étalonnage monoculaire, puis les poses relatives entre les caméras et celle de référence [Boutteau 2009]. Il est aussi possible d'estimer simultanément les paramètres intrinsèques et les poses relatives dans un même problème d'optimisation. C'est ce dernier cas de figure que nous avons retenu car l'optimisation simultanée de tous les paramètres mène à une solution plus vraisemblable pour l'ensemble du système en produisant un ensemble de paramètres ajustant mieux les mesures qu'en cas d'estimation séparée des paramètres intrinsèques et extrinsèques.

Par conséquent, en plus d'estimer la pose du système stéréo associée à chaque  $N$ -uplet d'images de mire,  $N$  jeux de paramètres intrinsèques sont à estimer ainsi que  $N-1$  poses relatives, positionnant les caméras par rapport à la première. Ce qui mène à estimer  $6p + N \text{card}(\gamma_m) + 6(N-1)$  paramètres, avec  $p$   $N$ -uplets d'images. Ainsi, en utilisant six images de la mire pour étalonner un banc stéréo de deux caméras perspectives, 50 paramètres sont à estimer et 52 pour deux caméras modélisées par la projection stéréographique. Enfin, pour étalonner le capteur stéréo-catadioptrique *FOO* [Mouaddib 2005] à quatre miroirs,  $6p + 38$  paramètres sont à estimer pour  $p$  images, soit 74 paramètres avec six poses de mire.

En repartant du critère de l'équation 2.15 à minimiser pour la primitive point, de la loi de commande virtuelle de l'équation 2.20 et en considérant la variation de point image pixelique de l'équation 2.23, on exprime le problème d'estimation de la variation des paramètres à estimer. Ainsi, avec  ${}^{c_j} \dot{\mathbf{u}}_i$ , la variation de l'ensemble des primitives de l'image  $i$  de la caméra  $j$ , la loi de commande virtuelle pour l'étalonnage stéréo avec  $p$   $N$ -uplets d'images est :

$$\begin{aligned} & \left( \mathbf{v}_1^1 \quad \dots \quad \mathbf{v}_1^p \quad \mathbf{v}_{1,2} \quad \dots \quad \mathbf{v}_{1,N} \quad \dot{\gamma}_{m,1} \quad \dots \quad \dot{\gamma}_{m,N} \right)^\top \\ & = \mathbf{L}_s^+ \left( {}^{c_1} \dot{\mathbf{u}}_1 \quad \dots \quad {}^{c_N} \dot{\mathbf{u}}_1 \quad {}^{c_1} \dot{\mathbf{u}}_2 \quad \dots \quad {}^{c_N} \dot{\mathbf{u}}_2 \quad \dots \quad {}^{c_N} \dot{\mathbf{u}}_p \right)^\top, \end{aligned} \quad (2.30)$$

où  $\mathbf{v}_1^i$  est le torseur cinématique de la pose du banc stéréo, pour chaque ensemble de  $N$  images, et  $\mathbf{v}_{1,j}$ , le torseur de pose relative entre la caméra  $j$  et la caméra 1. En considérant  $\mathbf{L}^i$ , la matrice d'interaction de la pose  $i$  du banc stéréo et  $\mathbf{L}_{1,j}^i$ , la

matrice d'interaction de pose relative entre la caméra 1 et la caméra  $j$  pour la pose  $i$ , l'expression de  $\mathbf{L}_S$  est :

$$\mathbf{L}_S = \left( \begin{array}{ccc|ccc|cccc} 0 & \dots & 0 & 0 & \dots & 0 & \frac{\partial {}^{c_1}\mathbf{u}_1}{\partial \gamma_{m,1}} & 0 & \dots & 0 \\ \mathbf{L}^1 & \vdots & \dots & \vdots & \mathbf{L}_{1,2}^1 & \dots & 0 & 0 & \frac{\partial {}^{c_2}\mathbf{u}_1}{\partial \gamma_{m,2}} & \dots & 0 \\ & \vdots & \dots & \vdots & \vdots & \ddots & \vdots & \vdots & \dots & \ddots & \vdots \\ & 0 & \dots & 0 & 0 & \dots & \mathbf{L}_{1,N}^1 & 0 & \dots & 0 & \frac{\partial {}^{c_N}\mathbf{u}_1}{\partial \gamma_{m,N}} \\ \hline \vdots & \dots & \ddots & \vdots \\ \hline 0 & \dots & 0 & 0 & \dots & 0 & \frac{\partial {}^{c_1}\mathbf{u}_p}{\partial \gamma_{m,1}} & 0 & \dots & 0 \\ & \vdots & \dots & 0 & \mathbf{L}^p & \mathbf{L}_{1,2}^p & \dots & 0 & 0 & \frac{\partial {}^{c_2}\mathbf{u}_p}{\partial \gamma_{m,2}} & \dots & 0 \\ & \vdots & \dots & \vdots & \vdots & \ddots & \vdots & \vdots & \dots & \ddots & \vdots \\ & 0 & \dots & 0 & 0 & \dots & \mathbf{L}_{1,N}^p & 0 & \dots & 0 & \frac{\partial {}^{c_N}\mathbf{u}_p}{\partial \gamma_{m,N}} \end{array} \right). \quad (2.31)$$

### 2.3.3 Stéréovision hybride

Les bancs stéréo sont généralement composés de caméras du même type et représentées par un même modèle de projection. Cependant, pour certaines applications [Sturm 2002, Eynard 2010], combiner une caméra omnidirectionnelle et une caméra perspective permet de fusionner le large champ de vue de la première et la résolution plus fine de la seconde. De plus, même si le modèle de projection unifié est valide pour les caméras perspectives, il reste plus complexe que le modèle trou d'aiguille classique. Enfin, utiliser différents modèles pour les caméras centrales est encore pertinent avec l'avantage principal d'appliquer des algorithmes adaptés pour des modèles spécifiques.

En remplaçant  $\gamma_{m,j}$  par  $\gamma_{m_j}$ , le numéro de modèle de projection dépend de la caméra  $j$  et en reprenant l'équation 2.15, on exprime le critère à optimiser pour le calcul de pose et l'étalonnage intrinsèque et extrinsèque simultanément pour tout banc stéréo hybride à  $N$  caméras et jusqu'à  $N$  modèles différents :

$$\Delta = \sum_{j=1}^N \sum_{k=1}^{k_j} (pr_{\gamma_{m_j}}({}^{c_j}\mathbf{M}_{c_1} {}^{c_1}\mathbf{M}_o {}^o\mathbf{X}_k) - {}^{c_j}\mathbf{u}_k^*)^2. \quad (2.32)$$

Le même raisonnement que pour l'étalonnage stéréo de caméras à modèles identiques est fait et conduit à la même loi de commande virtuelle qu'en équation 2.30 et au même jacobien d'étalonnage sauf que  $\gamma_{m_j}$  remplace  $\gamma_{m,j}$  pour prendre en compte des ensembles de paramètres intrinsèques différents, puisque les modèles sont différents. A noter aussi que les matrices d'interactions de pose  $\mathbf{L}^i$  et de pose relative  $\mathbf{L}_{1,j}^i$  sont calculées à partir du produit de jacobiens (équation 2.8) dont le premier est calculé à partir de la projection perspective ou de la projection stéréographique, suivant les modèles utilisés. La combinaison d'autres modèles est possible.

### 2.3.4 Résultats

Cette section évalue les approches présentées pour l'étallonage monoculaire, stéréoscopique et stéréoscopique multi-modèles. Les résultats ont été obtenus avec un logiciel d'étallonage développé conjointement avec Damien Eynard implantant les méthodes de cette partie de la thèse. Les résultats sont comparés à ceux obtenus avec les mêmes jeux d'images et de points avec les Toolbox existantes pour des caméras omnidirectionnelles [Mei 2006] et banc stéréo perspective [Bouguet 2004]. L'évaluation de l'étallonage par AVV de caméras perspectives est fait dans [Marchand 2001b].

#### 2.3.4.1 Etallonage de caméra omnidirectionnelle

Les résultats d'étallonage de caméra omnidirectionnelle monoculaire, modélisée par le modèle sphérique, sont comparés à ceux de la toolbox de C. Mei [Mei 2006].

La grille d'étallonage est un damier de 82 cm  $\times$  105 cm et de 7  $\times$  9 carrés. Un autre type de mire d'étallonage est utilisé et est composé de 36 disques noirs sur fond blanc dans une surface de 80 cm  $\times$  80 cm.

Cette expérimentation montre l'étallonage de caméra catadioptrique composée d'une caméra (Sony DFW-SX910) et d'une optique à miroir parabolique Remote Reality (Figure 2.10(a)). Six images de mire d'étallonage sont utilisées avec une résolution de 1280 $\times$ 960 pixels pour un total de 288 points.

L'étallonage est réalisé sans prendre en compte les distorsions et les résultats sont reportés dans le tableau de la figure 2.10(c). Pour une comparaison équitable, le même ensemble de points image subpixeliques, extrait des coins de la mire d'étallonage en damier, est utilisé. La dernière colonne de ce tableau montre les résultats obtenus avec le second type de mire composé de disques. Pour la comparaison, les deux types de mire apparaissent dans chaque image d'étallonage (Figure 2.10(b)). Les résultats montrent une précision supérieure avec le damier qu'avec les disques, dont le centre de gravité dans l'image est considéré comme la projection du centre 3D, ce qui est une approximation dans une image perspective comme dans une image omnidirectionnelle. Cependant, leur détection est plus facile et plus robuste, quand la qualité d'image est faible ou dans des conditions difficiles en général, que les coins du damier, même si dans le cas présent la détection est faite par la méthode du point selle.

Les résultats d'étallonage par AVV sont très similaires avec ceux obtenus par la toolbox de C. Mei, avec un léger avantage de  $10^{-4}$  pixel d'écart type de l'erreur de reprojection.

Pour évaluer la robustesse de l'étallonage par AVV relativement à l'initialisation, différentes valeurs initiales ont été considérées (Tableau 2.3). L'approche se montre robuste à d'importantes erreurs d'initialisation sur tous les paramètres et converge vers les mêmes valeurs optimales, même avec une erreur initiale très importante.

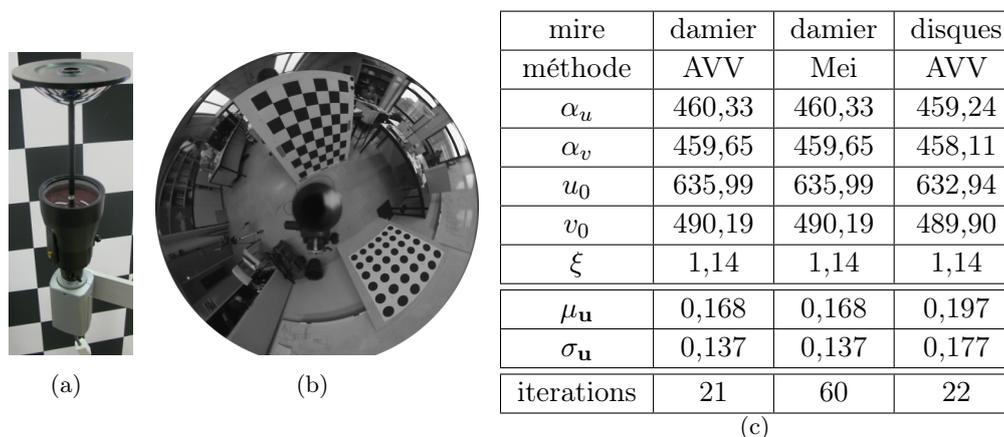


FIGURE 2.10 – (a) Caméra catadioptrique. (b) Une image utilisée pour l'étalonnage (deux types de mire). (c) Résultats d'étalonnage pour les deux types de mire et comparaison avec la toolbox de C. Mei :  $\mu_{\mathbf{u}}$  est l'erreur moyenne de reprojection et  $\sigma_{\mathbf{u}}$  son écart type.

$p_x$	480	0	2500	2500	480	0	2500	0	480	480	480
$p_y$	480	0	2500	0	480	0	2500	2500	480	480	480
$u_0$	640	640	640	640	0	0	0	0	640	640	640
$v_0$	480	480	480	480	0	0	0	0	480	480	480
$\xi$	1	1	1	1	1	1	1	1	0	0,5	2,0
$\bar{\mu}_{\mathbf{u}}$ init	95	174	815	473	601	610	1030	1050	4069	178	98
iter	23	21	24	29	22	23	24	36	D	22	27

TABLE 2.3 – Robustesse à l'initialisation de l'estimation des paramètres intrinsèques. Quand il y a convergence, les paramètres estimés sont ceux de la première colonne de la figure 2.10(c), avec la même erreur résiduelle. " $\bar{\mu}_{\mathbf{u}}$  init" est l'erreur moyenne de reprojection initiale et la dernière ligne présente le nombre d'itérations pour converger (D : Diverge).

### 2.3.4.2 Etalonnage stéréo

**Etalonnage de banc stéréo perspective** Ce banc est constitué de deux caméras perspectives (IDS uEye) avec des focales différentes (8 mm et 6 mm) mais avec une résolution identique de  $752 \times 480$  pixels. Le banc stéréo a une distance entre les deux caméras de 30 cm et il est admis que les axes principaux des deux caméras sont parallèles. Le tableau 2.4 montre les résultats d'étalonnage obtenus et les compare avec ceux estimés par la toolbox de J.-Y. Bouguet [Bouguet 2004]. L'étalonnage est réalisé en utilisant cinq paires d'images de damier (36 points par image) sans estimer les distorsions. Le même ensemble de points est utilisé pour les deux méthodes.

L'écart type de l'erreur de reprojection avec la toolbox de Bouguet est de 0,150 alors qu'avec l'AVV, il est de 0,119.

Les résultats d'étalonnage extrinsèque sont clairement similaires avec une diffé-

	$c_1$		$c_2$	
	AVV	Bouguet	AVV	Bouguet
$\alpha_u$	1118,27	1122,55	1412,82	1418,39
$\alpha_v$	1118,01	1122,14	1408,10	1413,70
$u_0$	414,91	414,23	342,99	342,43
$v_0$	216,24	212,31	253,14	254,70
$t_x$	/	/	-0,304	-0,305
$t_y$	/	/	0,007	-0,006
$t_z$	/	/	0,054	0,056
$\theta_x$	/	/	2,64	2,64
$\theta_y$	/	/	0,64	0,85
$\theta_z$	/	/	2,69	2,70

TABLE 2.4 – Comparaison des paramètres pour l'étalonnage stéréo perspective. Les translations sont en mètre et les rotations en degrés.

rence en translation estimée d'environ 1 mm et valident l'approche basée AVV.

#### Étalonnage de banc stéréo horizontal à deux caméras catadioptriques

Les caméras (Figure 1.4(c)) sont différentes mais les miroirs sont du même modèle et de la même marque. De plus, la résolution utilisée est la même pour les caméras, à savoir 1280×960 pixels.

Les résultats d'étalonnage de ce banc stéréo (Tableau 2.5) montrent une valeur estimée du paramètre  $\xi$  identique pour les deux caméras, ce qui est cohérent car les miroirs sont identiques et de facture précise. En considérant les deux capteurs catadioptriques verticaux, la distance réelle les séparant dans le plan horizontal est d'environ 38,5 cm et la distance verticale est d'environ 9 cm. L'erreur d'estimation de cette distance est de 0,13 cm horizontalement et de 0,05 cm verticalement. Enfin, l'orientation des capteurs est quasi identique, ce qui est bien reflété par les faibles rotations estimées selon les trois axes de  ${}^{c_2}\mathbf{M}_{c_1}$ . Les valeurs initiales des différents paramètres sont issues de l'étalonnage individuel de chaque capteur catadioptrique et la pose relative initiale est calculée à partir des poses de la mire d'une paire d'images.

#### Étalonnage de banc stéréo vertical à deux caméras catadioptriques

Chaque caméra catadioptrique est composée d'une caméra perspective et d'un miroir hyperboloïde (Figure 2.11(b)). L'assemblage d'une telle caméra est soumis à des contraintes pour garantir un point de vue unique. Les images utilisées pour l'étalonnage (Figure 2.11) ont été fournies par R. Boutteau qui a réalisé ce système stéréo vertical et a assemblé précisément chaque caméra catadioptrique en respectant les contraintes d'alignement des axes optiques du miroir et de la caméra perspective et la distance entre la caméra et le foyer du miroir [Boutteau 2010]. Chaque caméra omnidirectionnelle du banc stéréo étant centrale, le modèle de projection stéréogra-

caméra	$c_1$		$c_2$	
$\alpha_u$	458,51	[462,18]	485,14	[484,65]
$\alpha_v$	457,01	[461,00]	484,91	[484,40]
$u_0$	633,48	[635,79]	617,31	[617,01]
$v_0$	498,67	[497,45]	527,12	[523,12]
$\xi$	1,14	[1,14]	1,14	[1,14]
$t_x$	/		-38,33	[-36,82]
$t_y$	/		1,83	[2,20]
$t_z$	/		-8,95	[-9,59]
$\theta_x$	/		1,33	[0,74]
$\theta_y$	/		-2,02	[-1,97]
$\theta_z$	/		-2,20	[-1,74]
$\mu_{\mathbf{u}}$	0,251			
$\sigma_{\mathbf{u}}$	0,212			

TABLE 2.5 – Liste des estimés des paramètres intrinsèques et extrinsèques d’un banc stéréo horizontal à deux caméras catadioptriques par la méthode d’asservissement visuel virtuel.  $t_x$ ,  $t_y$ ,  $t_z$ ,  $\theta_x$ ,  $\theta_y$  et  $\theta_z$  sont les paramètres de pose extraits de  ${}^{c_2}\mathbf{M}_{c_1}$ . Les translations sont exprimées en centimètres et les angles en degrés. Les valeurs entre crochets sont les valeurs initiales de l’étalonnage stéréoscopique, issues de l’étalonnage individuel. Enfin,  $\mu_{\mathbf{u}}$  et  $\sigma_{\mathbf{u}}$ , sont en pixels.

phique unifié est valide.

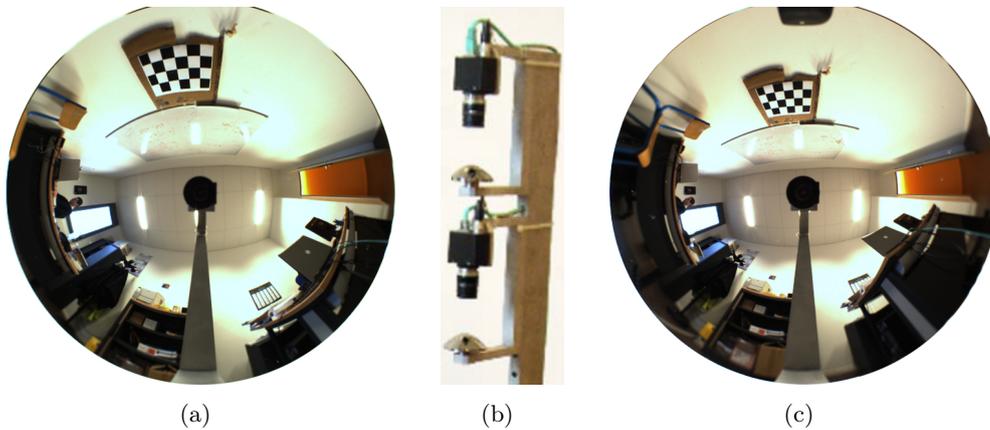


FIGURE 2.11 – Banc stéréo à deux caméras catadioptriques en colonne. L’image du capteur (b) est extraite de la thèse de Boutteau [Boutteau 2010]. (a) et (c) montrent respectivement la vue de la caméra omnidirectionnelle supérieure et celle de la caméra inférieure.

Les coins des carrés de 7 cm de côté du damier sont utilisés pour l’étalonnage à raison de 15 points par image de mire pour un total de 390 points ( $2 \times 13$  images). Le tableau 2.6 présente les résultats d’étalonnage avec un alignement quasi parfait entre

les deux axes des caméras et une translation en  $Z$  proche de la réalité : 22,22 cm estimés contre 22,50 cm mesurés, soit un rapport de l'ordre de 1% de l'écart sur la distance mesurée.

caméra	$c_1$	$c_2$
$\alpha_u$	392,93	386,68
$\alpha_v$	392,86	386,56
$u_0$	685,02	685,02
$v_0$	504,75	504,76
$\xi$	1,00	0,98
$t_x$	/	-0,13
$t_y$	/	-0,00
$t_z$	/	22,22
$\theta_x$	/	0,13
$\theta_y$	/	-0,02
$\theta_z$	/	-0,37
$\mu_{\mathbf{u}}$	0,215	
$\sigma_{\mathbf{u}}$	0,203	

TABLE 2.6 – Liste des estimés des paramètres intrinsèques et extrinsèques d'un banc stéréo vertical à deux caméras catadioptriques par la méthode d'asservissement visuel virtuel.  $t_x$ ,  $t_y$ ,  $t_z$ ,  $\theta_x$ ,  $\theta_y$  et  $\theta_z$  sont les paramètres de pose extraits de  ${}^{c_2}\mathbf{M}_{c_1}$ . Les translations sont exprimées en centimètres et les angles en degrés.  $\mu_{\mathbf{u}}$  et  $\sigma_{\mathbf{u}}$ , sont en pixels.

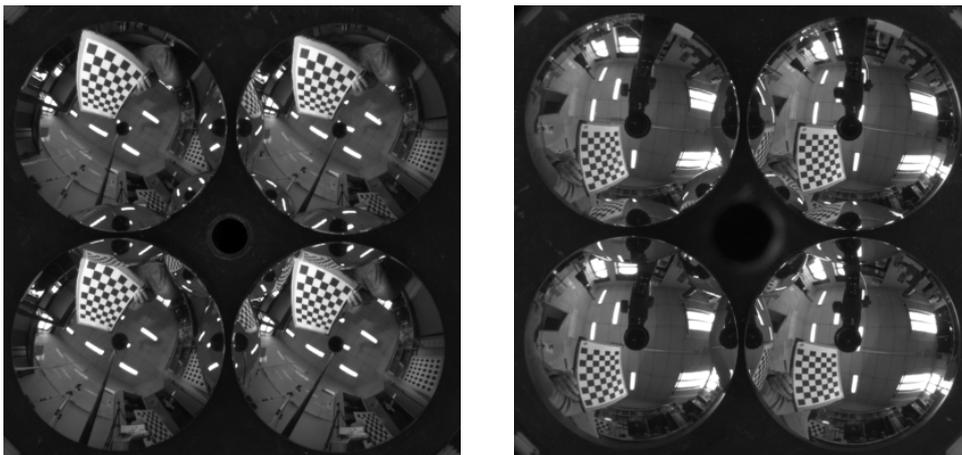
**Étallonage du capteur FOO** L'étallonage stéréoscopique a aussi été appliqué au capteur FOO (*Four On One*) [Mouaddib 2005]. Deux configurations ont été étalonnées. La première comprend les quatre miroirs paraboliques, une caméra et un objectif réellement télécentrique engendrant une projection orthographique (Figure 1.7(a)). Dans cette configuration, le capteur mesure 75 cm. La seconde configuration consiste à remplacer l'objectif réellement télécentrique par celui du dispositif catadioptrique de Remote Reality (Figure 2.10(a)). Avec ce second objectif, la dimension du capteur de stéréovision omnidirectionnelle (Figure 2.12) est de 30 cm, c'est-à-dire la même taille que le capteur catadioptrique monoculaire de la figure 2.10(a). Cependant, l'optique est quasi télécentrique ce qui conduit à une projection qui n'est pas orthographique dont les conséquences sont clairement visibles dans les images (Figure 2.13(b)) comme dans l'estimation des paramètres (Tableau 2.8).

Avec une seule caméra et quatre miroirs, on modélise le capteur comme étant composé de quatre caméras catadioptriques. Les résultats de son étallonage sont visibles dans le tableau 2.7, pour la configuration avec l'objectif réellement télécentrique, et dans le tableau 2.8, pour la seconde configuration. Les images utilisées sont de résolution 2560×1920 pixels (Figure 2.13), ce qui engendre une résolution



FIGURE 2.12 – Capteur FOO (*Four On One*) en utilisant l'optique de l'objectif catadioptrique de Remote Reality (Figure 2.10(a)).

par miroir similaire au banc stéréo à caméras multiples de l'expérience précédente. Les quatre miroirs sont disposés en carré de 3 cm de côté.



(a) Configuration avec objectif réellement télécentrique

(b) Configuration avec objectif quasi-télécentrique

FIGURE 2.13 – Images obtenues avec le capteur FOO dans ses deux configurations. A noter l'effet perspective dans l'image obtenue par la seconde configuration du FOO.

La mire utilisée est toujours un damier mais de dimension  $20 \text{ cm} \times 25 \text{ cm}$ , plus pratique à manipuler, car la faible ligne de base des miroirs nécessite de rapprocher la mire d'étalonnage le plus possible des miroirs pour minimiser l'incertitude sur les points de la mire. Ce second damier possède le même nombre de cellules que le premier.

La précision des paramètres intrinsèques des étalonnages présentés dans les tableaux 2.7 et 2.8 est difficilement quantifiable. Même si, physiquement, une seule

caméra	$c_1$	$c_2$	$c_3$	$c_4$
$\alpha_u$	367,56	388,99	397,39	426,21
$\alpha_v$	367,59	389,21	398,62	431,32
$u_0$	818,14	1737,98	1726,13	810,83
$v_0$	534,87	539,06	1458,94	1447,24
$\xi$	0,680	0,769	0,787	0,847
$r_u (\times 10^{-3})$	1,85	1,97	1,98	1,99
$r_v (\times 10^{-3})$	1,85	1,98	1,97	1,96
$t_x$	/	-2,93	-3,04	-0,28
$t_y$	/	0,13	-3,00	-2,93
$t_z$	/	-0,47	-0,37	0,06
$\theta_x$	/	-0,65	-2,48	-2,90
$\theta_y$	/	2,92	2,10	-0,29
$\theta_z$	/	0,45	0,45	0,02
$\mu_{\mathbf{u}}$	0,918			
$\sigma_{\mathbf{u}}$	0,863			

TABLE 2.7 – Liste des estimés des paramètres intrinsèques et extrinsèques du FOO dans sa première configuration.  $r_u = \frac{\xi}{\alpha_u}$  et  $r_v = \frac{\xi}{\alpha_v}$ . Le reste de la légende est identique à celle du tableau 2.5.

caméra	$c_1$	$c_2$	$c_3$	$c_4$
$\alpha_u$	336,82	369,96	369,97	386,56
$\alpha_v$	333,78	362,63	365,42	381,80
$u_0$	773,98	1611,73	1593,71	768,95
$v_0$	606,51	613,85	1454,82	1448,82
$\xi$	0,746	0,831	0,867	0,923
$r_u (\times 10^{-3})$	2,22	2,25	2,34	2,39
$r_v (\times 10^{-3})$	2,24	2,29	2,37	2,42
$t_x$	/	-3,89	-3,48	-0,07
$t_y$	/	0,07	-3,91	-3,89
$t_z$	/	-0,50	-0,58	-0,36
$\theta_x$	/	-0,77	0,05	0,05
$\theta_y$	/	-0,17	2,35	0,55
$\theta_z$	/	-0,17	-0,28	-0,23
$\mu_{\mathbf{u}}$	1,185			
$\sigma_{\mathbf{u}}$	1,071			

TABLE 2.8 – Liste des estimés des paramètres intrinsèques et extrinsèques du FOO dans sa seconde configuration. Le reste de la légende est identique à celle du tableau 2.7.

caméra est utilisée, la variation des  $\alpha_u$ ,  $\alpha_v$  et  $\xi$  n'est pas aberrante. En effet, ces paramètres sont intrinsèquement liés au miroir auquel ils sont associés et les miroirs

utilisés dans le prototype du FOO ne sont pas parfaitement identiques et présentent des défauts. De plus, ces paramètres sont fortement liés entre eux dans le modèle unifié de projection, de telle sorte qu'en modifiant  $\xi$ , si  $\alpha_u$  et  $\alpha_v$  le sont aussi en compensation, l'étalonnage reste valide. Par conséquent, plutôt que de quantifier les variations des paramètres directement sur  $\alpha_u$ ,  $\alpha_v$  et  $\xi$ , il apparaît plus pertinent d'observer les rapports  $r_u = \frac{\xi}{\alpha_u}$  et  $r_v = \frac{\xi}{\alpha_v}$ . Pour la première configuration, ces rapports (Tableau 2.7) ont une variation moyenne de 6,5% pour  $r_u$  et de 6,1% pour  $r_v$ . Ces variations illustrent les différences physiques entre les miroirs. Pour la seconde configuration, ces rapports (Tableau 2.7) ont une variation moyenne de 4,5% pour  $r_u$  et de 5,0% pour  $r_v$ . Les rapports calculés pour les deux configurations sont du même ordre mais leur écart est dû à l'optique télécentrique utilisée dans un cas et celle quasi-télécentrique de Remote Reality dans l'autre cas.

La précision dans l'estimation des paramètres extrinsèques, *i.e.* les  ${}^{c_j}\mathbf{M}_{c_1}$  pour  $2 \leq j \leq 4$ , est plus aisée à évaluer. Les miroirs étant disposés en carré de 3 cm de côté, l'erreur d'estimation est calculée à partir de la distance entre les miroirs formant un côté et ceux en diagonale, soit six distances. Ces distances sont calculées (Tab 2.9) pour les étalonnages du FOO dans ses deux configurations.

	référence	config 1	config 2
$d(c_1, c_2)$	3,00	2,93	3,89
$d(c_2, c_3)$	3,00	2,87	3,86
$d(c_3, c_4)$	3,00	2,76	3,41
$d(c_4, c_1)$	3,00	2,94	3,89
$d(c_1, c_3)$	4,24	4,27	5,23
$d(c_2, c_4)$	4,24	3,86	5,40
erreur moy	/	0,15	0,87
erreur min	/	0,03	0,41
erreur max	/	0,39	1,16
côté moyen	3,00	2,88	3,76
écart type des côtés	0,0	0,11	0,19

TABLE 2.9 – Evaluation de la précision des paramètres extrinsèques d'étalonnage du FOO pour les deux configurations présentées.  $d(c_i, c_j)$  exprime la distance en centimètres séparant les miroirs  $i$  et  $j$ . Les erreurs sont calculées par rapport aux valeurs de référence.

En les comparant aux véritables distances, une erreur moyenne approximative de 1,5 mm sur les six distances est constatée dans les paramètres extrinsèques estimés de la première configuration, soit une erreur relative de 4%, obtenue par le rapport de l'erreur moyenne sur la distance moyenne entre les miroirs. Pour les estimés de la seconde configuration, l'erreur moyenne est de 8,7 mm, soit une erreur relative de 29%. Cette erreur est due au fait que l'objectif utilisé dans la seconde configuration n'est pas réellement télécentrique, ce qui est facilement visible sur les images par un effet de perspective (Figure 2.13(b)). Cependant, la longueur moyenne estimée des côtés du carré, formé par les miroirs du FOO, illustre une surestimation globale

de la longueur des côtés. La stabilité de l'estimation, illustrée par l'écart type des longueurs estimées des côtés, montre que les positions estimées des miroirs forment un carré, dont les dimensions sont plus importantes. L'optimisation compense, ici, le fait que la projection de la réflexion des miroirs dans l'image ne soit pas orthographique dans la seconde configuration.

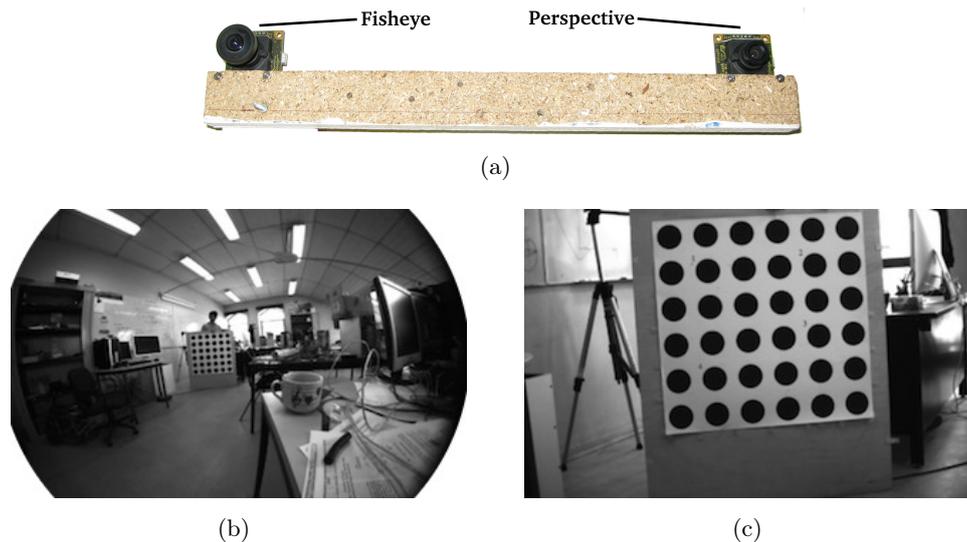


FIGURE 2.14 – Images utilisées pour l'étalonnage du banc stéréo mixte (a) composé d'une caméra fisheye (b) et d'une perspective (c).

**Étalonnage de banc stéréo mixte** Ce banc stéréo mixte (Figure 2.14(a)) est composé d'une caméra modélisée par la projection stéréographique, c'est-à-dire une sphère, et une caméra perspective modélisée par un plan. La mire d'étalonnage est formée de disques de 4,75 cm de rayon et espacés de 4,6 cm (Figure 2.14). Six paires d'images d'étalonnage sont utilisées avec 36 points dans chaque image. Pour cette expérimentation, l'erreur moyenne de reprojection est d'environ 0,170 pixels avec 0,187 d'écart type. Les deux caméras sont montées sur une même barre rigide et de telle sorte que leurs axes principaux soient parallèles, ce qui est estimé (Tableau. 2.10(a)). Le banc stéréo a une distance entre les deux caméras de 30 cm, mesurée manuellement, selon l'axe  $X$  des caméras. L'erreur d'estimation en translation est donc d'environ 6 mm (2%).

Afin d'évaluer la précision de l'étalonnage directement dans les images, des points de l'environnement n'appartenant pas à la mire d'étalonnage sont sélectionnés dans l'image perspective (Figure 2.15(a)). Les coniques épipolaires correspondantes sont obtenues à partir de la matrice essentielle calculée avec les résultats de l'étalonnage et sont tracées sur l'image fisheye (Figure 2.15(b)). En zoomant l'image fisheye sur les points d'intérêt correspondant à ceux sélectionnés dans l'image perspective, il est clair qu'ils reposent sur leur conique épipolaire (Figure 2.15). Cette observation qualitative montre la précision et la vraisemblance des paramètres obtenus par

	fisheye	perspective		fisheye	perspective
$\alpha_u$	482,11	1164,57	$\alpha_u$	512,08	673,12
$\alpha_v$	484,15	1170,25	$\alpha_v$	507,87	679,59
$u_0$	344,92	385,70	$u_0$	325,94	337,34
$v_0$	242,97	218,47	$v_0$	241,15	212,02
$\xi$	1,22	/	$\xi$	1,05	/
$t_x$	/	-0,293	$t_x$	/	-0,075
$t_y$	/	0,006	$t_y$	/	-0,075
$t_z$	/	-0,010	$t_z$	/	0,029
$\theta_x$	/	1,80	$\theta_x$	/	0,27
$\theta_y$	/	-0,69	$\theta_y$	/	3,84
$\theta_z$	/	1,89	$\theta_z$	/	90,28

(a) (b)

TABLE 2.10 – Résultats d'étalonnage de deux bancs stéréo composé d'une caméra fisheye et d'une perspective. Les translations sont en mètre et les rotations en degrés.

étalonnage.

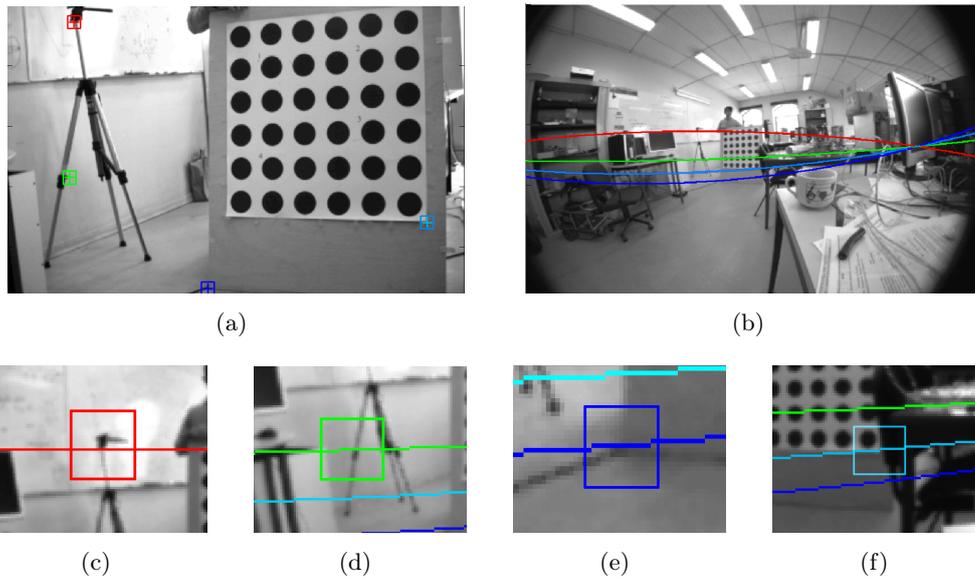


FIGURE 2.15 – Des points d'intérêt sont sélectionnés dans l'image perspective (a) et leurs lignes épipolaires sont calculées dans l'image fisheye (b). (c), (d), (e) et (f) montrent plus précisément la partie des coniques épipolaires correspondant aux points sélectionnés dans l'image perspective.

Une seconde configuration de banc stéréo à caméras fisheye et perspective est utilisée avec une rotation de  $90^\circ$  entre deux caméras Sony XCD-V50 (Figure 2.16) avec des optiques différentes du premier banc hybride.

Ce banc stéréo hybride est calibré à l'aide de sept paires d'images ( $640 \times 480$



FIGURE 2.16 – Capteur stéréo mixte dans une seconde configuration avec une caméra fisheye sur la droite de l’image et une perspective sur la gauche.

pixels) avec 36 points par image menant à 504 points. L’erreur finale de reprojection est de 0,337 pixels avec 0,353 pixels d’écart-type. Le tableau 2.10(b) montre les résultats d’étalonnage avec une rotation estimée de  $90,28^\circ$  autour de l’axe  $Z$ , l’axe optique de la caméra de référence, menant, pour ce paramètre, à un rapport de 0,3% de l’écart de l’estimation sur l’orientation mesurée.

## 2.4 Suivi et calcul de pose basé modèle 3D filaire

Dans la partie 2.2, vous avons développé le calcul de pose par asservissement visuel virtuel en utilisant les points comme primitives en précisant que toute primitive peut être utilisée dans un processus d’asservissement visuel virtuel dès lors qu’il est possible d’en calculer l’interaction avec la pose de caméra. Le suivi et le calcul de pose d’objet défini par un modèle 3D filaire ont été développés de plusieurs manières : vision perspective [Comport 2003], vision omnidirectionnelle [Marchand 2007] et stéréovision perspective [Dionnet 2007]. Cette section présente dans un premier temps la fusion de ces deux derniers travaux pour réaliser le suivi et le calcul de pose de modèle 3D en stéréovision omnidirectionnelle en minimisant la distance algébrique “point-droite”, définie dans le plan image [Caron 2009]. Ensuite, nous avons exprimé le critère à minimiser sur la sphère d’équivalence, permettant ainsi de définir simplement une distance géométrique “point-droite” et aussi d’utiliser toutes les droites tout en évitant les singularités.

### 2.4.1 AVV de modèle 3D dans le plan image

#### 2.4.1.1 Type de primitive

L’objet suivi est défini par un modèle 3D composé d’arêtes (des droites). La primitive considérée est la distance entre un point  $\mathbf{x}$ , détecté dans l’image, et la projection d’une droite, une conique  $\mathbf{c}(\mathbf{r})$  dans le plan image, pour une pose donnée.

Le vecteur de primitives courantes  $\mathbf{s}(\mathbf{r})$  devient alors :

$$\mathbf{s}(\mathbf{r}) = \begin{bmatrix} \vdots \\ s_i(\mathbf{r}) \\ \vdots \end{bmatrix} \quad \text{avec} \quad s_i(\mathbf{r}) = d_a(\mathbf{x}, \mathbf{c}(\mathbf{r})) , \quad (2.33)$$

où  $d_a()$  définit la distance, ou la mesure, algébrique entre un point  $\mathbf{x}$  et la projection  $\mathbf{c}(\mathbf{r})$  d'une droite 3D.

### 2.4.1.2 Projection d'une droite 3D

Plusieurs modélisations de droites 3D existent : combinaison d'un point et d'un vecteur, de deux points, coordonnées de Plucker ou encore l'intersection de deux plans. En considérant cette dernière représentation, le premier plan passe par le centre de la sphère de projection et l'autre est perpendiculaire au premier.

Ces deux plans  $\mathcal{P}_1, \mathcal{P}_2$  sont définis par :

$$\begin{aligned} \mathcal{P}_1 : A_1X + B_1Y + C_1(Z - \xi) &= 0 \\ \mathcal{P}_2 : A_2X + B_2Y + C_2Z + D_2 &= 0 \end{aligned} \quad (2.34)$$

avec  $\xi$  défini dans la partie 1.1.2 et les contraintes suivantes sur les paramètres 3D :

$$\begin{cases} A_1^2 + B_1^2 + C_1^2 = 1 \\ A_2^2 + B_2^2 + C_2^2 = 1 \\ A_1A_2 + B_1B_2 + C_1C_2 = 0 \end{cases} , \quad (2.35)$$

de telle sorte que les deux plans, dont les normales  $\mathbf{N}_1 = (A_1, B_1, C_1)$  et  $\mathbf{N}_2 = (A_2, B_2, C_2)$  sont unitaires, soient orthogonaux.

Selon le modèle de projection stéréographique, la projection d'une droite 3D dans l'image omnidirectionnelle est la projection perspective du cercle défini par l'intersection du plan  $\mathcal{P}_1$  et de la sphère unitaire  $\mathcal{S}$  (Figure 2.17).

Il est ainsi possible de définir l'équation d'une conique à partir des paramètres des plans  $\mathcal{P}_1, \mathcal{P}_2$  et de la sphère  $\mathcal{S}$  [Marchand 2007]. En effet, le cercle 3D précédemment mentionné est caractérisé par le système :

$$\begin{cases} X^2 + Y^2 + (Z - \xi)^2 = 1 \\ A_1X + B_1Y + C_1(Z - \xi) = 0 \end{cases} . \quad (2.36)$$

A partir de la première équation de ce système, on obtient l'équation de la projection perspective de ce cercle dans le plan image [Espiau 1992], qui est une conique :

$$K_0x^2 + K_1y^2 + 2K_2xy + 2K_3x + 2K_4y + K_5 = 0, \quad (2.37)$$

avec

$$\begin{cases} K_0 = 1 + \frac{A_1^2}{C_1^2\xi^2}(\xi^2 - 1) & K_1 = 1 + \frac{B_1^2}{C_1^2\xi^2}(\xi^2 - 1) \\ K_2 = \frac{A_1B_1}{C_1^2\xi^2}(\xi^2 - 1) & K_3 = -\frac{A_1}{C_1\xi^2} \\ K_4 = -\frac{B_1}{C_1\xi^2} & K_5 = -\frac{1}{\xi^2} \end{cases} . \quad (2.38)$$

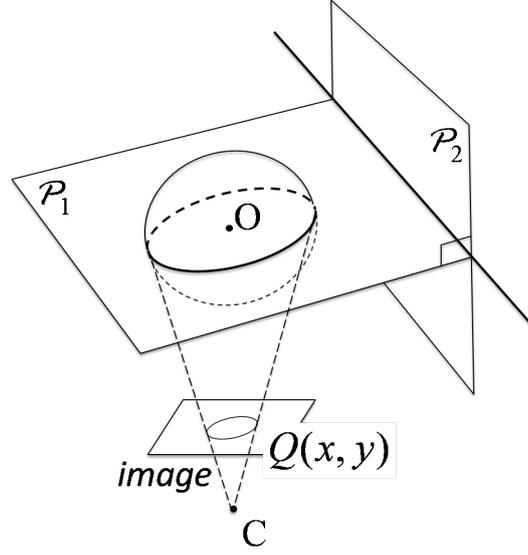


FIGURE 2.17 – Projection d’une droite 3D en grand cercle sur la sphère puis en conique dans le plan image omnidirectionnel.

L’équation de conique 2.37 présente six paramètres pour cinq degrés de liberté. Pour obtenir une représentation minimale, on pose  $a_i = K_i/K_5$ ,  $i = 0..4$  :

$$Q(x, y) = a_0x^2 + a_1y^2 + 2a_2xy + 2a_3x + 2a_4y + 1 \quad (2.39)$$

avec

$$\begin{cases} a_0 = \frac{A_1^2}{C_1^2}(1 - \xi^2) - \xi^2 & a_1 = \frac{B_1^2}{C_1^2}(1 - \xi^2) - \xi^2 \\ a_2 = \frac{A_1 B_1}{C_1^2}(1 - \xi^2) & a_3 = \frac{A_1}{C_1} \\ a_4 = \frac{B_1}{C_1} \end{cases} . \quad (2.40)$$

### 2.4.1.3 Matrice d’interaction pour une droite

La matrice d’interaction doit lier l’évolution des paramètres  $a_i$  à la vitesse de la caméra. Il est possible de ré-écrire  $a_0$ ,  $a_1$  et  $a_2$  en fonction de  $a_3$  et  $a_4$  [Marchand 2007] :

$$\begin{cases} a_0 = a_3^2(1 - \xi^2) - \xi^2 \\ a_1 = a_4^2(1 - \xi^2) - \xi^2 \\ a_2 = a_3 a_4(1 - \xi^2) \end{cases} , \quad (2.41)$$

menant à la variation des paramètres de la conique :

$$\begin{cases} \dot{a}_0 = 2a_3(1 - \xi^2)\dot{a}_3 \\ \dot{a}_1 = 2a_4(1 - \xi^2)\dot{a}_4 \\ \dot{a}_2 = (1 - \xi^2)(a_3\dot{a}_4 + a_4\dot{a}_3) \end{cases} . \quad (2.42)$$

Pour obtenir la matrice d'interaction  $\mathbf{L}_{a_i}$  pour  $a_i$ ,  $i = 0..4$ , il suffit donc de déterminer  $\mathbf{L}_{a_3}$  et  $\mathbf{L}_{a_4}$ . En dérivant  $a_3$  et  $a_4$  de l'équation 2.40, on obtient :

$$\begin{cases} \dot{a}_3 = \frac{\dot{A}_1}{C_1} - \frac{A_1}{C_1^2} \dot{C}_1 = \frac{\dot{A}_1}{C_1} - a_3 \frac{\dot{C}_1}{C_1} \\ \dot{a}_4 = \frac{\dot{B}_1}{C_1} - \frac{B_1}{C_1^2} \dot{C}_1 = \frac{\dot{B}_1}{C_1} - a_4 \frac{\dot{C}_1}{C_1} \end{cases} \quad (2.43)$$

Ensuite, en considérant  $\dot{\mathbf{N}}_1 = (\dot{A}_1, \dot{B}_1, \dot{C}_1)$ , la variation temporelle de la normale au plan  $\mathcal{P}_1$  est donnée par la relation [Andreff 2002] :

$$\dot{\mathbf{N}}_1 = -\frac{1}{D_2} \mathbf{N}_1 \mathbf{N}_2^T v - \mathbf{N}_1 \times \omega, \quad (2.44)$$

menant à :

$$\begin{cases} \mathbf{L}_{A_1} = [-A_1 A & -B_1 A & -C_1 A & 0 & -C_1 & B_1] \\ \mathbf{L}_{B_1} = [-A_1 B & -B_1 B & -C_1 B & C_1 & 0 & -A_1] \\ \mathbf{L}_{C_1} = [-A_1 C & -B_1 C & -C_1 C & -B_1 & A_1 & 0] \end{cases}, \quad (2.45)$$

avec

$$A = -\frac{A_2}{D_2}, \quad B = -\frac{B_2}{D_2}, \quad C = -\frac{C_2}{D_2}. \quad (2.46)$$

Ensuite, en injectant le système 2.45 dans le système 2.43, on obtient :

$$\mathbf{L}_{a_3} = [\alpha a_3 \quad \alpha a_4 \quad \alpha \quad a_3 a_4 \quad -1 - a_3^2 \quad a_4] \quad (2.47)$$

$$\mathbf{L}_{a_4} = [\beta a_3 \quad \beta a_4 \quad \beta \quad 1 + a_4^2 \quad -a_3 a_4 \quad -a_3] \quad (2.48)$$

avec  $\alpha = A - C a_3$  et  $\beta = B - C a_4$ . Et enfin, en substituant 2.47 et 2.48 dans 2.42, on déduit directement l'interaction liée à  $a_0$ ,  $a_1$  et  $a_2$ , puisqu'on a :

$$\mathbf{L}_{a_0} = 2a_3(1 - \xi^2) \mathbf{L}_{a_3} \quad (2.49)$$

$$\mathbf{L}_{a_1} = 2a_4(1 - \xi^2) \mathbf{L}_{a_4} \quad (2.50)$$

$$\mathbf{L}_{a_2} = (1 - \xi^2)(a_3 \mathbf{L}_{a_4} + a_4 \mathbf{L}_{a_3}) \quad (2.51)$$

#### 2.4.1.4 Matrice d'interaction pour la primitive *distance point-conique*

La primitive retenue est la distance entre un point et la projection d'une droite 3D dans le plan image, c'est-à-dire une conique, avec le modèle de projection stéréographique utilisé. La distance utilisée est la mesure algébrique. En considérant un point  $(x, y)$ , sa distance algébrique d'une conique est :

$$d_a = Q(x, y) \quad (2.52)$$

avec  $Q(x, y)$  défini dans l'équation 2.39. La matrice d'interaction  $\mathbf{L}_{d_a}$  liée à cette distance [Marchand 2007], en considérant la variation temporelle de  $d_a$  :

$$\dot{d}_a = \dot{a}_0 x^2 + \dot{a}_1 y^2 + 2\dot{a}_2 xy + 2\dot{a}_3 x + 2\dot{a}_4 y \quad (2.53)$$

qui mène immédiatement à :

$$\mathbf{L}_{d_a} = \begin{bmatrix} x^2 \\ y^2 \\ 2xy \\ 2x \\ 2y \end{bmatrix}^\top \begin{bmatrix} \mathbf{L}_{a_0} \\ \mathbf{L}_{a_1} \\ \mathbf{L}_{a_2} \\ \mathbf{L}_{a_3} \\ \mathbf{L}_{a_4} \end{bmatrix} \quad (2.54)$$

Il y aura donc autant d'ensembles de  $\mathbf{L}_{d_a}$  que de caméras utilisées et seront combinés comme illustré dans l'équation 2.20. Cependant, une quantité non négligeable d'outliers peut apparaître si une partie de l'objet est occultée ou s'il existe des contours plus forts que les arêtes de l'objet lui-même dans l'arrière-plan des images. L'utilisation d'un M-Estimeur permet de rendre le suivi robuste à ces problèmes, comme introduit dans la partie 1.2.3.2 et plus particulièrement avec la loi de commande virtuelle de l'équation 1.22. Le calcul du vecteur de mise à jour de pose dans le suivi stéréo robuste de modèle 3D s'écrit alors :

$$\mathbf{v}_1 = -\lambda \begin{bmatrix} \mathbf{D}_1 \mathbf{L}_{d_{a_1}} \\ \mathbf{D}_2 \mathbf{L}_{d_{a_2}} \quad c_2 \mathbf{V}_{c_1} \\ \vdots \\ \mathbf{D}_N \mathbf{L}_{d_{a_3}} \quad c_N \mathbf{V}_{c_1} \end{bmatrix}^+ \begin{bmatrix} \mathbf{D}_1 \\ \mathbf{D}_2 \\ \vdots \\ \mathbf{D}_N \end{bmatrix} \begin{bmatrix} \mathbf{d}_{a_1} \\ \mathbf{d}_{a_2} \\ \vdots \\ \mathbf{d}_{a_N} \end{bmatrix}, \quad (2.55)$$

avec  $\mathbf{d}_{a_j}$  le vecteur d'erreurs de l'image de la caméra  $j$ . Le calcul des poids des matrices  $\mathbf{D}_j$  se fait individuellement à chaque image.

#### 2.4.1.5 Traitement d'image

Afin de trouver les contours correspondants dans une nouvelle image "proche" de la précédente, l'algorithme des contours mobiles [Bouthemy 1989] est utilisé. L'idée est d'échantillonner le contour initial à un pas régulier et d'utiliser un masque de gradient orienté pour trouver le contour correspondant dans l'image par maximisation de convolution dans un intervalle de recherche 1D de part et d'autre du contour initial, selon la direction normale au contour. L'un des avantages de cette méthode est que la recherche ne s'effectue que pour les contours alignés dans la même direction que le contour parent. Un tableau de 180 masques est généré hors ligne et est indexé en fonction de l'angle du contour. L'efficacité de la convolution permet d'atteindre des performances temps-réel.

Plus précisément, le procédé consiste à chercher le point correspondant  $\mathbf{x}_{t+1}$  dans l'image  $\mathbf{I}^{t+1}$  pour chaque point  $\mathbf{x}_t$  (Figure 2.18). Pour obtenir les  $\mathbf{x}_t$  initiaux, le contour parent est échantillonné régulièrement. Pour ce faire, il est possible d'échantillonner régulièrement la droite, ou l'arête 3D et de projeter ensuite ces points 3D dans l'image. Cependant, il est plus efficace d'échantillonner directement la conique.

Une relation permet de transformer une conique en cercle par décomposition en

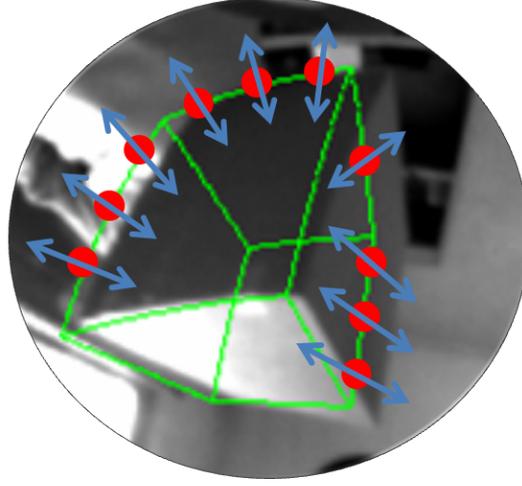


FIGURE 2.18 – Recherche du contour correspondant selon les normales aux échantillons du projeté du modèle 3D par la méthode des contours mobiles [Bouthemy 1989].

valeurs propres de la matrice de conique  $\mathbf{C}$  [Sturm 2007] :

$$\mathbf{C} = \begin{pmatrix} a_0 & a_2 & a_3 \\ a_2 & a_1 & a_4 \\ a_3 & a_4 & 1 \end{pmatrix}, \quad (2.56)$$

qui se décompose donc en :

$$\mathbf{C} = \mathbf{R}\mathbf{\Sigma}\mathbf{R}^T. \quad (2.57)$$

Ensuite, en utilisant  $\mathbf{\Sigma}$  et  $\mathbf{R}$ , la transformation  $\mathbf{T}$  de la conique en cercle unitaire est donnée par :

$$\mathbf{T} = \mathbf{\Sigma}^{-1}\mathbf{R}^{-1}, \quad (2.58)$$

qui applique d'abord une rotation à la conique pour l'ajuster sur le repère 2D ortho-normal standard et normalise ensuite ses axes pour l'ajuster sur le cercle unitaire. Les points extrêmes de l'arête projetée dans l'image sont connus, ce qui permet de définir l'intervalle à échantillonner. C'est donc un arc de cercle qu'il suffit d'échantillonner avec un pas angulaire régulier. La transformation inverse  $\mathbf{T}^{-1}$  est finalement appliquée à chaque échantillon pour le replacer sur la conique.

En chaque échantillon  $\mathbf{x}_t$ , un intervalle de recherche 1D  $\{\mathbf{Q}_q, q \in [-J, J]\}$ , avec  $J$  la borne de l'intervalle de recherche, est déterminé dans la direction  $\delta$  de la normale au contour. Cette normale s'exprime directement à partir des dérivées partielles de  $Q(x, y)$ , tel que :

$$\delta = \begin{pmatrix} \frac{\partial Q}{\partial x} \\ \frac{\partial Q}{\partial y} \end{pmatrix} = \begin{pmatrix} 2a_0x + 2a_2y + 2a_3 \\ 2a_1y + 2a_2x + 2a_4 \end{pmatrix} \quad (2.59)$$

Pour chaque point issu de l'échantillonnage, et pour chaque position entière  $Q_q$ , un critère correspondant à la racine carrée d'un ratio de vraisemblance logarithmique  $\zeta^q$  est calculé [Bouthemy 1989]. Ce dernier est en fait la somme des valeurs absolues de convolutions, calculée en  $\mathbf{x}_t$  et  $Q_q$ , respectivement dans les images  $\mathbf{I}^t$  et  $\mathbf{I}^{t+1}$ , en utilisant un masque  $M_\delta$  pré-déterminé, fonction de l'orientation du gradient du contour. La nouvelle position  $\mathbf{x}_{t+1}$  est ensuite obtenue par :

$$Q_q^* = \arg \max_{q \in [-J, J]} \zeta^q \quad \text{avec} \quad \zeta^q = |\mathbf{I}_{\nu(\mathbf{x}_t)}^t * M_\delta + \mathbf{I}_{\nu(Q_q)}^{t+1} * M_\delta|. \quad (2.60)$$

$\nu(\cdot)$  est le voisinage du pixel considéré. Dans les expérimentations qui suivent, la taille du voisinage est limitée à  $7 \times 7$  pixels afin de garder des performances temps-réel. Pour la même raison, il y a un compromis à faire entre la taille de l'intervalle de recherche et le mouvement maximal de l'objet entre deux images successives d'une séquence. Une distance de recherche de dix pixels permet d'obtenir de bons résultats en présence d'un mouvement à vitesse moyenne entre deux images.

Cette recherche bas niveau produit une liste de points utilisés pour calculer la distance définie à l'équation 2.52 et utilisé dans les équations 2.33 et 2.54.

### 2.4.2 AVV de modèle 3D en représentation sphérique cartésienne

L'inconvénient de minimiser la distance point-droite dans le plan image en vision omnidirectionnelle réside dans le fait que la représentation de la projection d'une droite sous forme de conique présente une singularité [Barreto 2001]. La projection d'une droite 3D formant un plan avec l'axe optique de la caméra engendre ainsi une conique dégénérée, puisque cette projection est en fait un segment de droite rectiligne dans l'image. C'est d'ailleurs le seul cas où une droite 3D se projette en droite (radiale) dans une image omnidirectionnelle. Une solution pourrait être d'utiliser deux représentations, la conique et la droite dans l'image mais, au delà de la gestion simultanée de deux représentations pour un même type de primitive géométrique, la projection d'une droite quasi parallèle à l'axe optique de la caméra engendre une conique très mal conditionnée ce qui va fortement perturber le calcul de la mesure algébrique entre un point et la conique. De plus, il serait difficile de décider quand représenter la projection d'une droite 3D par une conique ou par une droite ou introduire un seuil empirique. Enfin, et cela a été brièvement abordé ci-dessus, utiliser une distance géométrique entre un point et la projection d'une droite, plutôt qu'une mesure algébrique, s'avère plus efficace lors d'un mauvais conditionnement de la conique et est plus pertinent car c'est une véritable distance.

Minimiser la distance point-droite sur la sphère plutôt que dans le plan image permet de résoudre tous ces problèmes à la fois et simplifie de surcroît les expressions. On ne parle plus de distance point-droite ou point-conique mais *point-grand cercle* car une droite 3D se projette en grand cercle sur la sphère par intersection de la sphère et du plan formé par la droite 3D et le centre de la sphère (Figure 2.17), quelque soit son orientation. Cela résout le premier problème de représentation de la projection d'une droite 3D. La projection sphérique d'une droite 3D est définie à

l'équation 2.36 et en ne considérant que les points situés sur la surface de la sphère, la normale  $\mathbf{N}_1$  suffit à paramétrer le grand cercle.

L'interaction entre la normale et le mouvement de la caméra virtuelle passe par l'expression de la variation  $\dot{\mathbf{N}}_1$  de la normale au grand cercle qui a été faite à l'équation 2.44. La variation temporelle de  $\mathbf{N}_1$  conduit à la matrice d'interaction liée à la projection d'une droite 3D sur la sphère :

$$\mathbf{L}_{\mathbf{N}_1} = \begin{bmatrix} \mathbf{L}_{A_1} \\ \mathbf{L}_{B_1} \\ \mathbf{L}_{C_1} \end{bmatrix} = \left[ -\frac{1}{D_2} \mathbf{N}_2 \mathbf{N}_1^T \quad \middle| \quad [\mathbf{N}_1]_{\times} \right] \quad (2.61)$$

#### 2.4.2.1 Matrice d'interaction de la *distance point-grand cercle*

En considérant un point  $\mathbf{X}_S$  sur la sphère, sa distance au grand cercle défini par  $\mathbf{N}_1$ , peut être exprimée par le produit scalaire  $d = \mathbf{N}_1 \cdot \mathbf{X}_S$  :

$$d = A_1 X_S + B_1 Y_S + C_1 Z_S \quad (2.62)$$

Cette distance est intéressante car  $d = 0$  quand le point est sur le grand cercle et va tendre vers  $\pm 1$  en s'éloignant du grand cercle.

On peut alors calculer la matrice d'interaction  $\mathbf{L}_d$  liée à cette distance. En considérant la variation temporelle de  $d$ , on a :

$$\dot{d} = \dot{A}_1 X_S + \dot{B}_1 Y_S + \dot{C}_1 Z_S, \quad (2.63)$$

et on obtient immédiatement  $\mathbf{L}_d$  :

$$\mathbf{L}_d = \begin{pmatrix} X_S \\ Y_S \\ Z_S \end{pmatrix}^T \begin{pmatrix} \mathbf{L}_{A_1} \\ \mathbf{L}_{B_1} \\ \mathbf{L}_{C_1} \end{pmatrix}, \quad (2.64)$$

à remplacer dans la loi de commande virtuelle de l'équation 2.55 pour réaliser le suivi et calcul simultané sphérique.

Remarque : Les points servant à calculer la distance *point-grand cercle* sont aussi obtenus par la méthode des contours mobiles. L'échantillonnage est cependant fait ici sur le grand cercle, ce qui permet de réaliser un échantillonnage régulier sur la sphère directement, contrairement au cas de la conique où une transformation supplémentaire est nécessaire. Les échantillons sont ensuite projetés dans l'image et le reste du processus est identique aux contours mobiles effectués dans le plan image.

### 2.4.3 Applications et comparaisons

L'algorithme de suivi et de calcul de pose de modèle 3D filaire en stéréovision omnidirectionnelle a été appliqué sur des séquences d'images d'une résolution de 1280×960 pixels. Le développement est réalisé en C++ en utilisant, en partie, la librairie ViSP [Marchand 2005b].

La première expérimentation de suivi de boîte vise à la fois à valider la méthode de suivi et calcul de pose basé modèle 3D dans le plan image et à montrer l'utilité du capteur FOO avec quatre miroirs là où deux suffiraient, intuitivement. Les comparaisons entre l'utilisation de deux ou quatre miroirs utilisent d'ailleurs les mêmes séquences d'images afin d'avoir la même caméra, les mêmes conditions d'expérimentation et la même résolution.

Enfin, après avoir évalué la robustesse de l'approche aux occultations mutuelles, le but de la dernière expérimentation est de comparer l'optimisation sphérique à l'optimisation dans le plan image afin de confirmer la formulation plus judicieuse de la première sur la seconde.

#### 2.4.3.1 Suivi de boîte

Dans cette expérimentation, le capteur est fixe et une boîte (30 cm  $\times$  25 cm  $\times$  20 cm) est déplacée à la main dans une large partie du champ de vue du capteur (Figure 2.19), à un intervalle de distances au capteur de 35 cm à 75 cm. Les images sont acquises à une fréquence de dix images par seconde et le temps de traitement moyen sur un PC Pentium IV 3 GHz est de 300 ms par image stéréo. Les étapes de suivi et de calcul de poids pour le M-Estimeur sont aisément parallélisables et l'utilisation d'un processeur multi-cœur permettrait d'atteindre la cadence vidéo. Malgré la présence de gradients perturbateurs dans l'arrière-plan et des variations d'illumination sur l'objet, le suivi réussit tout au long de la séquence de 312 images. Bien entendu, certains points des contours mobiles sont influencés par certains gradients forts de l'arrière-plan mais grâce à l'estimation robuste et la robustesse accrue par la stéréovision, ces outliers sont rejetés et ne perturbent ni le suivi ni l'estimation de la pose.

Le suivi a aussi été appliqué à deux miroirs seulement, ceux avec la baseline la plus importante (les miroirs en haut à gauche et en bas à droite). Ces deux miroirs seraient ceux utilisés dans un capteur stéréo catadioptrique plus classique : un capteur à deux miroirs (ou un banc stéréo à deux caméras). Mais avec deux miroirs seulement, l'objet est perdu à partir de l'image 277 (Figure 2.19(d)). Cela montre la robustesse accrue par la redondance d'information apportée par les deux miroirs supplémentaire et donc l'intérêt de ce capteur.

**Auto-occultations** Une autre expérimentation a été faite pour montrer la robustesse apportée par le FOO par rapport à un capteur stéréo à deux miroirs. Un problème d'auto-occultation peut apparaître quand les miroirs de révolution sont sur un même plan. Ils se reflètent les uns sur les autres. Ce phénomène engendre des zones problématiques dans les images et sont généralement évitées en ignorant les pixels appartenant à ces zones. C'est ce que nous avons fait pour éviter les problèmes de suivi et cette expérimentation a été faite de telle sorte que la position initiale de la boîte soit en dehors de la zone d'inter-réflexions des deux miroirs diagonaux, elle est déplacée dans cette zone puis est replacée aux environs de sa position initiale. Le suivi en utilisant les quatre miroirs réussit (Figures 2.20(a) et 2.20(c)) tandis qu'avec

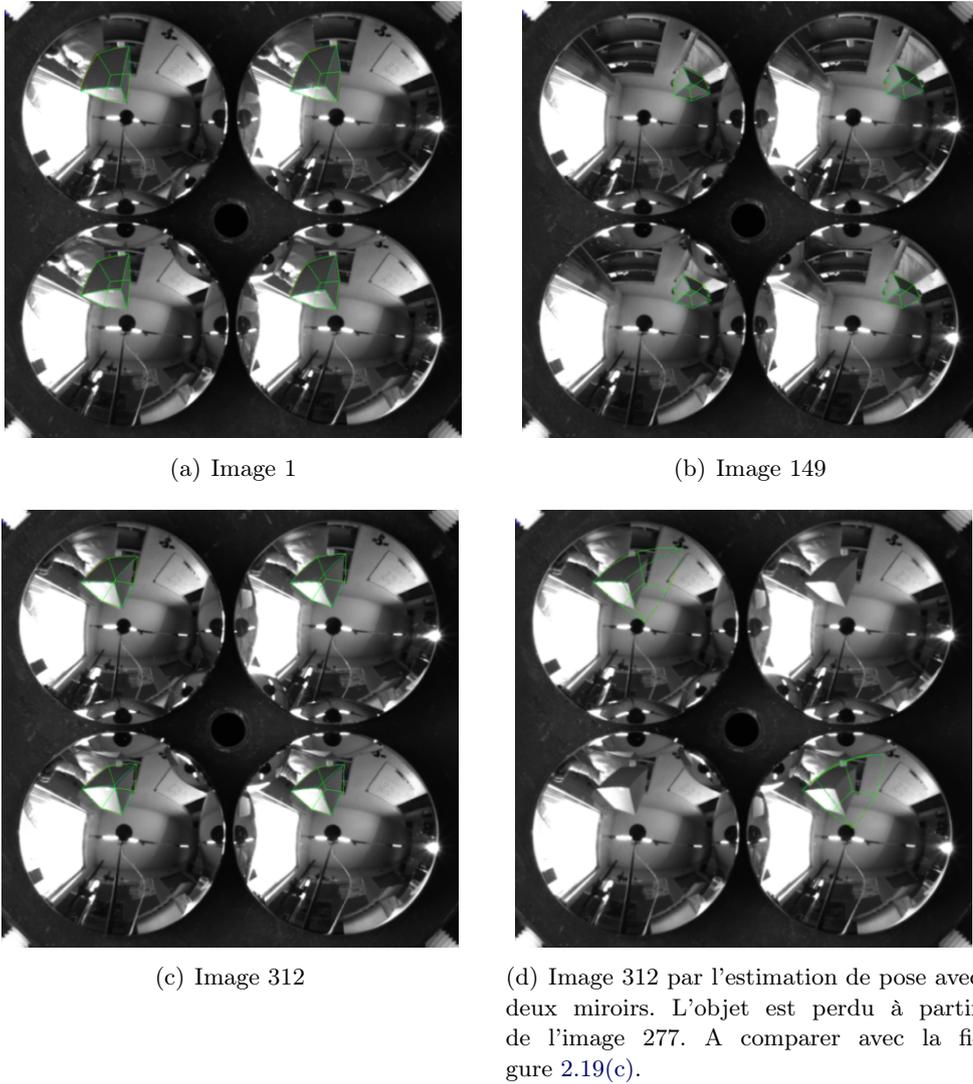


FIGURE 2.19 – Images d'estimation de pose pour la séquence avec la boîte déplacée à la main.

deux miroirs, la boîte est perdue dans la zone d'auto-occultation (Figures 2.20(b) et 2.20(d)).

#### 2.4.3.2 Comparaison des approches sphériques et plan image

La formulation sphérique du calcul de pose par le suivi d'un objet défini par un modèle 3D filaire a un impact aussi bien en vision omnidirectionnelle monoculaire que stéréo. L'impact principal attendu des développements théoriques est un meilleur comportement avec les droites radiales ou quasi radiales. En effet, quand le suivi est réalisé dans le plan image omnidirectionnel et quand une droite suivie est quasi rectiligne dans l'image, la conique qui la caractérise devient mal conditionnée,

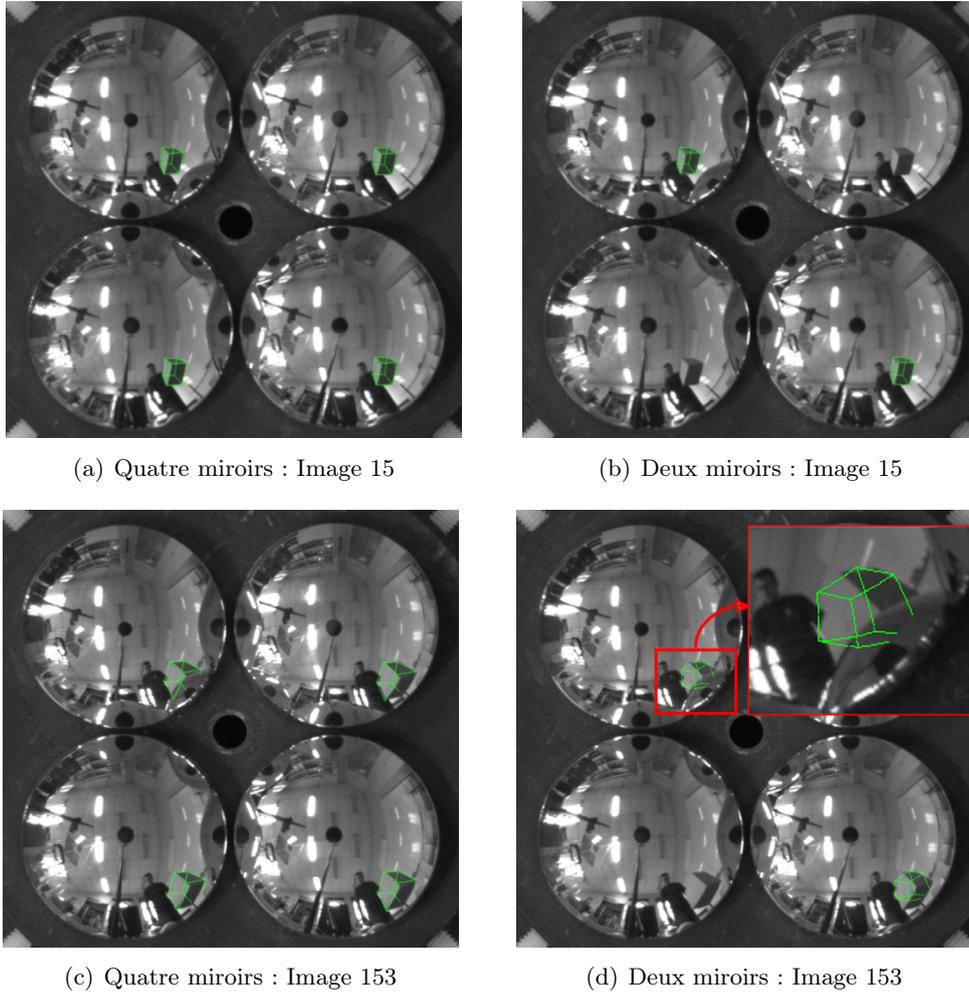


FIGURE 2.20 – Images de suivi et d’estimation de pose simultanés en présence d’auto-occultations. La robustesse apportées par les quatre miroirs permet de gérer efficacement les auto-occultations.

ce qui engendre des distances algébriques importantes même si la distance géométrique réelle est faible. Ceci a pour effet de rejeter les mesures d’erreur entre les échantillons et leur droite correspondante quand celle-ci est radiale (Figure 2.21(a)) alors que la formulation sphérique (Figure 2.21(b)) ne souffre pas de ce problème et seuls les échantillons ne se situant pas sur le contour de l’objet sont rejetés (en bas à droite de la boîte dans l’image). De plus, la distance retenue est géométrique, ce qui fait plus sens qu’une distance algébrique qui n’est qu’une mesure, pas une distance au sens strict.

La stéréo bénéficie de la formulation sphérique de la même manière qu’avec une unique caméra. En effet, là où les quatre miroirs apportent la redondance nécessaire pour la réussite du suivi tout au long des séquences présentées dans les figures 2.19 et 2.20, la robustesse apportée par la formulation sphérique permet la réussite de ce

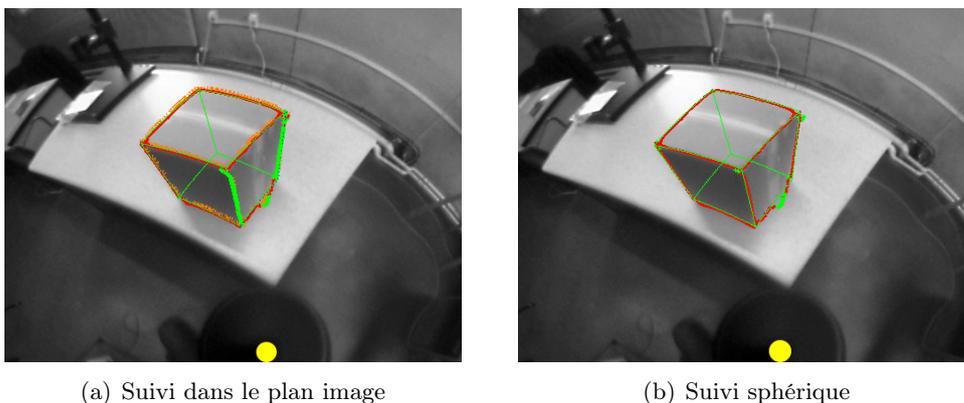


FIGURE 2.21 – Superpositions des échantillons (croix rouges et vertes) et du modèle 3D projeté sur l'image (lignes vertes). Les échantillons en rouge sont considérés comme vraisemblables par l'estimateur robuste alors que ceux en vert sont considérés aberrants. Sur chaque sous-figure, le gros point jaune indique le point principal de l'image et les coniques singulières sont les arêtes alignées avec ce point.

suivi, même avec uniquement deux miroirs (Figure 2.22). Ce dernier étant un échec si le suivi est réalisé dans l'image plane (Figure 2.19).

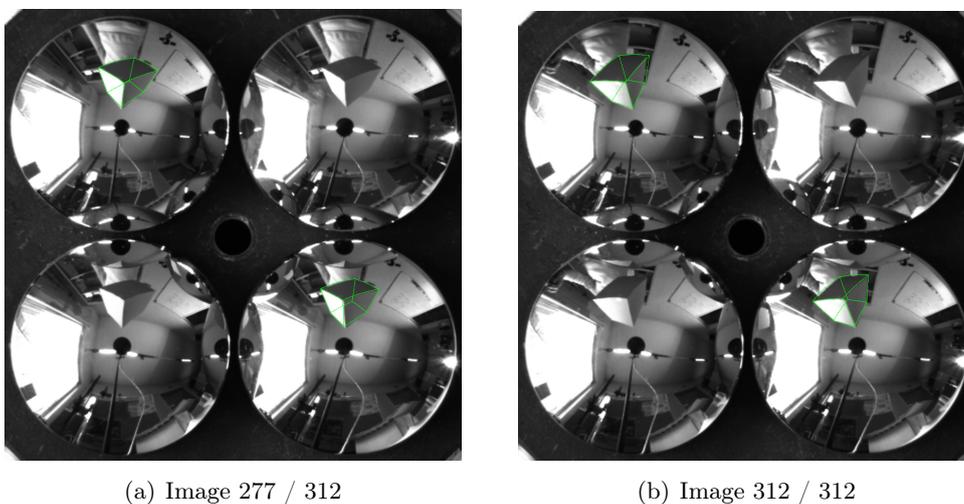


FIGURE 2.22 – Application de l'AVV sphérique sur la même séquence qu'en figure 2.19. Le suivi avec quatre miroirs fonctionnait déjà avec l'AVV dans le plan image et fonctionne parfaitement aussi avec la représentation sphérique. Par contre, en utilisant deux miroirs sur les quatre, le suivi dans le plan image échoue à partir de l'image 277 de cette séquence de 312 images alors qu'avec la formulation sphérique, le suivi est efficace sur toute la séquence.

## 2.5 Conclusion

Dans ce chapitre, nous avons montré qu'il est possible, en utilisant l'AVV, de résoudre le problème d'estimation de pose, de l'étalonnage et du suivi et calcul de pose basé modèle filaire. La formulation a été faite pour les caméras omnidirectionnelles monoculaires et stéréoscopiques. Nous avons également comparé et discuté trois représentations différentes : dans le plan image, sur la sphère en représentation cartésienne et sphérique pure.

Nous avons évalué le comportement et les résultats obtenus par l'estimation de pose basé points à l'aide d'un système de localisation précis, qui sera utilisé comme système d'accès à la vérité terrain dans le chapitre 3.

La généralisation du calcul de pose à l'étalonnage a permis de montrer que le formalisme utilisé permet d'estimer simultanément les paramètres intrinsèques et extrinsèques de plusieurs caméras assemblées sur un banc stéréo rigide. Nous avons aussi formulé et validé l'étalonnage stéréo simultané de caméras faisant intervenir des modèles de projection différents.

Enfin, nous avons formulé l'AVV sphérique basé modèle filaire, ce qui a permis d'éviter les singularités. Les démonstrations de suivi confirment ce qui était attendu des développements théoriques : un meilleur comportement de l'AVV sur la sphère que dans le plan image.



# Asservissement visuel photométrique

---

## Sommaire

<b>3.1</b>	<b>Introduction</b>	<b>71</b>
<b>3.2</b>	<b>Formulation du problème</b>	<b>72</b>
3.2.1	Asservissement photométrique omnidirectionnel dans le plan image	73
3.2.2	AVP sur l'image sphérique en représentation sphérique pure	74
3.2.3	AVP sur l'image sphérique en représentation sphérique cartésienne	75
<b>3.3</b>	<b>Application sur un robot cartésien</b>	<b>77</b>
3.3.1	Validation	78
3.3.2	Evaluation du comportement de l'AVP pour différentes poses initiales	79
3.3.3	Mouvement ample	82
<b>3.4</b>	<b>Application sur un robot mobile</b>	<b>82</b>
3.4.1	Formulation	82
3.4.2	Application	84
<b>3.5</b>	<b>Application au suivi de chemin visuel</b>	<b>91</b>
3.5.1	Mise en oeuvre	92
3.5.2	Expérimentation	92
<b>3.6</b>	<b>Conclusion</b>	<b>94</b>

---

## 3.1 Introduction

L'asservissement visuel contrôle le mouvement d'un robot vers une position désirée en utilisant les images acquises par une caméra et en utilisant généralement des primitives géométriques présentes dans ces images. Cependant, détecter et suivre des primitives géométriques (points, des droites, des cercles, etc) dans les images pose des difficultés [Marchand 2005a] et a fortiori dans les images omnidirectionnelles. Une idée intéressante est alors d'utiliser directement l'image comme un tout plutôt que d'en extraire des primitives. Cette approche aurait alors l'avantage d'éliminer les problèmes de détection et de suivi et, de plus, apporterait beaucoup plus d'informations.

Ce principe a déjà été développé dans des travaux [Nayar 1996, Deguchi 2000] où, pour réduire la dimensionnalité des données de l'image, une décomposition en valeurs propres est faite. La commande est ensuite réalisée dans l'espace des valeurs propres et la matrice d'interaction liée à l'espace propre est apprise hors ligne.

Collewet et Marchand [Collewet 2008] ont proposé d'utiliser directement les intensités des pixels de toute l'image, en entrée du schéma de commande. Une approche d'asservissement visuel direct en utilisant l'homographie a été développée [Benhimane 2007] mais, même si l'intensité de l'image est utilisée à la base de la méthode, une importante étape de traitement d'image est nécessaire afin d'estimer l'homographie utilisée pour construire la loi de commande.

Etendre ce type d'approche [Collewet 2008] à la vision omnidirectionnelle apparaît intéressant afin de profiter d'un plus large champ de vue qu'en vision perspective et ainsi maximiser les chances de percevoir une zone texturée de l'environnement. Cependant, l'extension de l'asservissement visuel photométrique (AVP) à la vision omnidirectionnelle nécessite d'étudier les différentes représentations et problèmes inhérents à la vision omnidirectionnelle. En effet, utiliser les intensités nécessite le calcul des gradients de l'image. Or, en vision omnidirectionnelle, plusieurs travaux [Démonceaux 2009] ont montré l'intérêt d'effectuer le traitement d'images (ici, le calcul de gradient) sur la sphère d'équivalence du modèle de projection unifié.

Ce chapitre vise à formuler l'asservissement visuel omnidirectionnel photométrique sur l'image sphérique, même si la représentation dans le plan image sera aussi formulée pour les besoins de comparaison. Ces trois approches seront ensuite validées et comparées. Les expérimentations d'AVP seront menées sur un robot manipulateur et sur un robot mobile. Une application de suivi de chemin visuel termine le chapitre avant de conclure.

### 3.2 Formulation du problème

En considérant l'asservissement visuel comme un problème d'optimisation minimisant la différence entre les images courante et désirée :

$$\mathbf{e} = \|\mathbf{I}(\mathbf{r}) - \mathbf{I}(\mathbf{r}^*)\|, \quad (3.1)$$

la loi de commande la plus efficace est basée sur une technique de type Levenberg-Marquardt (LM). Il a été montré [Collewet 2008] qu'elle assure une meilleure convergence que d'autres lois de commande. Par conséquent, plutôt qu'une loi de commande de type Gauss-Newton (1.18), nous allons utiliser celle de LM :

$$\mathbf{v} = -\lambda (\mathbf{H} + \mu \text{diag}(\mathbf{H}))^{-1} \mathbf{L}_I^\top (\mathbf{I}(\mathbf{r}) - \mathbf{I}(\mathbf{r}^*)) \quad (3.2)$$

avec  $\mathbf{H} = \mathbf{L}_I^\top \mathbf{L}_I$  où  $\mathbf{L}_I$  est la matrice d'interaction liée à la luminance de l'image  $\mathbf{I}$ . Si  $\mu$  est très élevée, la loi de commande se comporte comme une descente de gradient tandis qu'une valeur très faible pour  $\mu$  mène l'équation (3.2) à se comporter comme l'équation (1.18).

### 3.2.1 Asservissement photométrique omnidirectionnel dans le plan image

La primitive visuelle considérée est la luminance  $I$  de chaque point de l'image. Le vecteur de primitives  $\mathbf{s}(\mathbf{r})$  devient alors :

$$\mathbf{s}(\mathbf{r}) = \mathbf{I}(\mathbf{r}) = (\mathbf{I}_{1\bullet}, \mathbf{I}_{2\bullet}, \dots, \mathbf{I}_{N\bullet}) \quad (3.3)$$

avec  $\mathbf{I}_{i\bullet}$ , la  $i$ -ème ligne de l'image.

#### 3.2.1.1 Contrainte du flot optique

L'idée de base est celle du flot optique. L'intensité du point  $\mathbf{x}$  est considérée constante malgré le mouvement :

$$I(\mathbf{x} + \mathbf{dx}, t + dt) = I(\mathbf{x}, t) \quad (3.4)$$

en admettant  $\mathbf{dx}$  faible. Dans ce cas, l'équation de contrainte du flot optique (Optical Flow Constraint Equation : OFCE) est valide :

$$\nabla I^\top \dot{\mathbf{x}} + I_t = 0 \quad (3.5)$$

avec  $\nabla I$  le gradient spatial de  $I(\mathbf{x}, t)$  et  $I_t = \frac{\partial I(\mathbf{x}, t)}{\partial t}$ .

#### 3.2.1.2 Expression de la matrice d'interaction

Pour l'asservissement visuel omnidirectionnel défini dans le plan image, la connaissance de la matrice d'interaction  $\mathbf{L}_\mathbf{x}$  (équation (2.5)) mène à [Collewet 2008] :

$$I_t = -\nabla I^\top \mathbf{L}_\mathbf{x} \mathbf{v}. \quad (3.6)$$

On pose  $\mathbf{L}_I(\mathbf{x})$ , la matrice d'interaction liée à  $I$  au pixel  $\mathbf{x}$  :

$$\mathbf{L}_I(\mathbf{x}) = -\nabla I^\top \mathbf{L}_\mathbf{x}. \quad (3.7)$$

#### 3.2.1.3 Calcul de gradients

Dans une image perspective, les gradients image  $\nabla I$  sont calculés avec le même voisinage pour tout point de l'image : un carré régulièrement échantillonné. Dans les images omnidirectionnelles, comme indiqué dans le chapitre 2, la géométrie de l'image est différente : la résolution et l'orientation ne sont pas constantes. Par conséquent, le voisinage utilisé pour le calcul de gradient doit être adapté.

En admettant néanmoins que le voisinage autour de  $\mathbf{u}_0 = (u_0 \ v_0 \ 1)^\top$  soit carré et régulièrement échantillonné, l'idée, pour trouver le voisinage autour du pixel  $\mathbf{u} = (u \ v \ 1)^\top$ , est de projeter ce voisinage sur un pôle de la sphère. Ensuite, il doit être déplacé pour être centré au point sphérique  $\mathbf{X}_\mathcal{S} = pr_\xi^{-1}(\mathbf{K}^{-1}\mathbf{u})$ . En considérant  $\mathbf{X}_{\mathcal{S}0}$ , la projection inverse de  $\mathbf{u}_0$ , ce mouvement est défini par une rotation autour de l'axe  $[\mathbf{X}_{\mathcal{S}0} \times \mathbf{X}_\mathcal{S}]$  et d'angle  $\arccos(\mathbf{X}_{\mathcal{S}0}^\top \mathbf{X}_\mathcal{S})$ .

Enfin, le voisinage est projeté sur le plan image et, pour trouver les intensités des pixels, on réalise une interpolation. Par souci d'efficacité en temps de calcul, l'interpolation au plus proche voisin a été utilisée pour estimer ces intensités. De plus, en utilisant ce type d'interpolation, les voisinages adaptés peuvent être calculés hors ligne de telle sorte que le processus de calcul de gradient soit aussi rapide qu'avec un voisinage standard.

La figure 3.1 montre des formes de voisinage à différentes positions dans l'image.

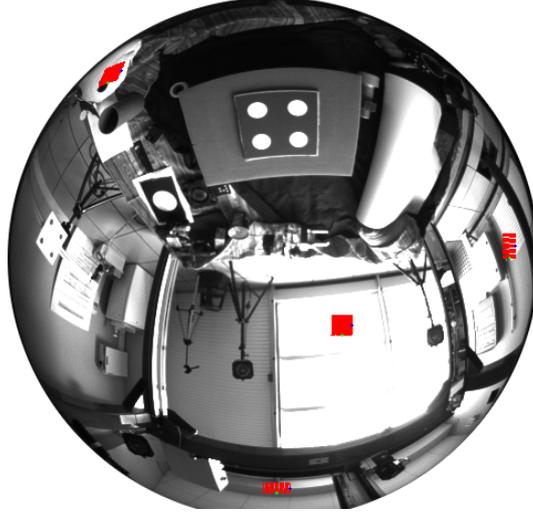


FIGURE 3.1 – Voisinages adaptés exprimés en différentes positions de l'image omnidirectionnelle.

### 3.2.2 AVP sur l'image sphérique en représentation sphérique pure

Travailler avec la sphère d'équivalence implique de travailler avec l'image sphérique  $I_S$ , qui est la projection inverse de l'image  $I$  sur la sphère  $\mathcal{S}$ . En considérant la paramétrisation en azimuth et élévation d'un point sphérique, l'OFCE sphérique pure devient [Daniilidis 2002] :

$$\frac{1}{\sin \phi} \frac{\partial I_S}{\partial \theta} \dot{\mathbf{p}}_\theta + \frac{\partial I_S}{\partial \phi} \dot{\mathbf{p}}_\phi + I_t = 0 \quad (3.8)$$

où  $(\dot{\mathbf{p}}_\theta \ \dot{\mathbf{p}}_\phi \ 0)^\top$  est le vecteur de flot optique tangent à la sphère au point  $(\theta \ \phi)^\top$ , avec  $\mathbf{p} = (\sin \phi \cos \theta \ \sin \phi \sin \theta \ \cos \phi)^\top$ . Cette équation de contrainte est utilisée pour déduire la matrice d'interaction  $\mathbf{L}_{I_S}(\mathbf{S})$ , sachant  $\mathbf{L}_S$  (équation 2.9) :

$$\mathbf{L}_{I_S}(\mathbf{S}) = -\nabla I_S^\top \mathbf{L}_S. \quad (3.9)$$

Calculer les gradients sphériques peut se faire par le calcul des gradients de l'image plane, en utilisant les voisinages adaptés (section 3.2.1.3), et leur transfert

sur la sphère par [Daniilidis 2002] :

$$\begin{cases} \frac{\partial I_S}{\partial \theta} = \frac{\partial I}{\partial u} \frac{\partial u}{\partial \theta} + \frac{\partial I}{\partial v} \frac{\partial v}{\partial \theta} \\ \frac{\partial I_S}{\partial \phi} = \frac{\partial I}{\partial u} \frac{\partial u}{\partial \phi} + \frac{\partial I}{\partial v} \frac{\partial v}{\partial \phi} \end{cases}. \quad (3.10)$$

Cependant, il est possible de calculer les gradients sphériques directement sur l'image sphérique  $I_S$  [Demonceaux 2009]. C'est plus efficace que d'élever les gradients de l'image plane sur la sphère et le filtrage d'image omnidirectionnelle est, de plus, uniquement valide sur la sphère d'équivalence où la convolution est commutative avec un échantillonnage régulier. La méthode consiste à définir des voisinages adaptés à la géométrie sphérique afin de calculer les gradients de manière similaire aux images perspectives.

Demonceaux *et al.* [Demonceaux 2009] ont présenté une métrique géodésique pour formuler les voisinages mais dans le cas présent, c'est une métrique angulaire qui doit être utilisée pour satisfaire l'équation (3.8). Par conséquent, en considérant  $\Delta_{\phi, \theta} = \arccos(|(0 \ 0 \ 1) \cdot pr_{\xi}^{-1}([u_0 + 1 \ v_0 \ 1]^T)|)$ , les deux  $N$ -voisinages  $\phi_{\mathcal{N}}$  et  $\theta_{\mathcal{N}}$  s'expriment par :

$$\begin{cases} \phi_{\mathcal{N}} = \left\{ (\theta, \ \phi + k\Delta_{\phi})^T, -\frac{N}{2} \leq k \leq \frac{N}{2}, k \neq 0 \right\} \\ \theta_{\mathcal{N}} = \left\{ (\theta + k\Delta_{\theta}, \ \phi)^T, -\frac{N}{2} \leq k \leq \frac{N}{2}, k \neq 0 \right\} \end{cases}. \quad (3.11)$$

$N$  définit ici le nombre de voisins autour d'un point pour un voisinage linéaire.

Finalement, les voisinages sont projetés sur le plan image pour retrouver les intensités. L'interpolation au plus proche voisin est utilisée pour rendre le processus rapide et permettre un pré-calcul hors ligne des voisinages sphériques projetés dans l'image pour chaque pixel.

Les résultats des calculs de gradients par élévation des gradients image (Figures 3.2(a) et 3.2(c)) ou directement sur l'image sphérique (Figures 3.2(b) et 3.2(d)) montrent des différences avec des gradients moins lisses pour le calcul direct. Les gradients calculés sur la sphère respectent mieux les orientations radiales et circulaires que ceux élevés du plan image.

### 3.2.3 AVP sur l'image sphérique en représentation sphérique cartésienne

Toujours en travaillant sur  $I_S$ , nous avons montré qu'une représentation cartésienne pouvait être utilisée (section 2.2.2.2). Dans cette représentation, l'OFCE devient alors :

$$\frac{\partial I_S}{\partial X_S} \dot{X}_S + \frac{\partial I_S}{\partial Y_S} \dot{Y}_S + \frac{\partial I_S}{\partial Z_S} \dot{Z}_S + I_t = 0. \quad (3.12)$$

La matrice d'interaction  $\mathbf{L}_{I_S}(\mathbf{X}_S)$  liée à  $I_S$  au point sphérique  $\mathbf{X}_S$  est définie par :

$$\mathbf{L}_{I_S}(\mathbf{X}_S) = -\nabla I_S^T \mathbf{L}_{\mathbf{X}_S}. \quad (3.13)$$

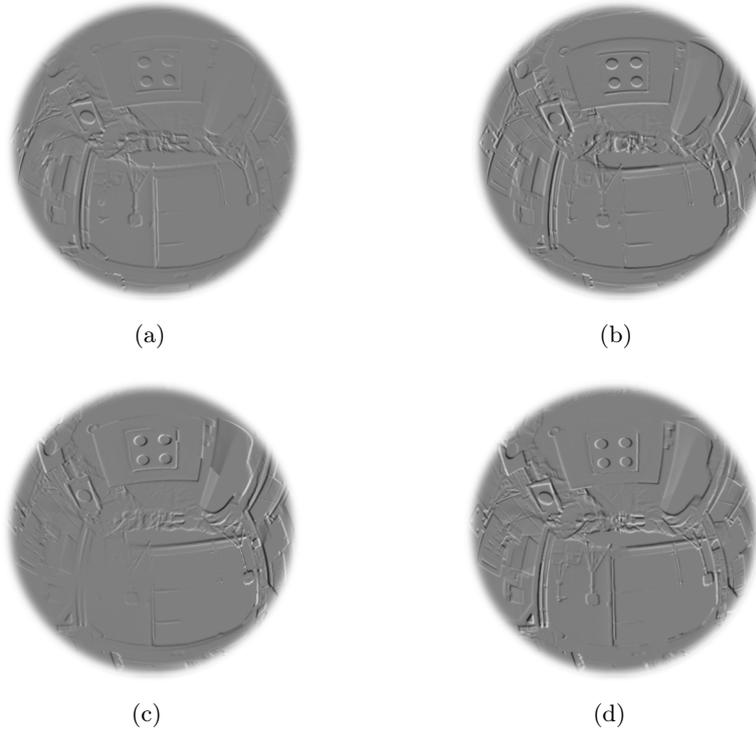


FIGURE 3.2 – Comparaisons des images de gradients sphériques purs lorsqu'ils sont calculés dans le plan image puis élevés ((a) et (c)) et sur l'image sphérique directement ((b) et (d)). L'image source est visible dans la figure 3.1.

$\mathbf{L}_{X_S}$  est donnée par l'équation (2.11) mais les gradients de l'image sur la sphère doivent de nouveau être revisités.

Une première approche consiste à calculer le gradient dans l'image plane (section 3.2.1.3) et de les élever sur la sphère en dérivant l'équation (1.7) :

$$\left\{ \begin{array}{l} \frac{\partial I_S}{\partial X_S} = \frac{\partial I}{\partial u} \frac{\partial u}{\partial X_S} + \frac{\partial I}{\partial v} \frac{\partial v}{\partial X_S} \\ \frac{\partial I_S}{\partial Y_S} = \frac{\partial I}{\partial u} \frac{\partial u}{\partial Y_S} + \frac{\partial I}{\partial v} \frac{\partial v}{\partial Y_S} \\ \frac{\partial I_S}{\partial Z_S} = \frac{\partial I}{\partial u} \frac{\partial u}{\partial Z_S} + \frac{\partial I}{\partial v} \frac{\partial v}{\partial Z_S} \end{array} \right. \quad (3.14)$$

Cette technique n'est pas des plus efficace car pour calculer la forme des voisinages adaptés, il y a déjà deux projections entre le plan image et la sphère. Par conséquent, calculer les gradients dans le plan image et les élever sur la sphère ajoute une projection de plus. Calculer les gradients directement sur la sphère d'équivalence est plus efficace et apporte plus d'information, particulièrement pour les gradients en  $Z_S$  (Figures 3.3(a) et 3.3(b)).

Pour calculer les gradients  $(\frac{\partial I_S}{\partial X_S}, \frac{\partial I_S}{\partial Y_S}, \frac{\partial I_S}{\partial Z_S})$  directement sur la sphère, il faut

définir un pas d'échantillonnage :

$$\Delta_{X_S} = \Delta_{Y_S} = \Delta_{Z_S} = \left\| pr_{\xi}^{-1} \left( \begin{bmatrix} u_0 + 1 \\ v_0 \\ 1 \end{bmatrix} \right) - \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\|. \quad (3.15)$$

A partir d'un point  $(X_S \ Y_S \ Z_S)^\top$ , trois  $N$ -voisins sont définis, un pour chaque axe. Le voisinage sphérique pour la première coordonnée,  $X_{S\mathcal{N}}$ , est défini par :

$$X_{S\mathcal{N}} = \left\{ \frac{(X_S + k\Delta_{X_S} \ Y_S \ Z_S)^\top}{\|(X_S + k\Delta_{X_S} \ Y_S \ Z_S)^\top\|}, \quad -\frac{N}{2} \leq k \leq \frac{N}{2}, k \neq 0 \right\}. \quad (3.16)$$

La procédure est similaire pour  $Y_{S\mathcal{N}}$  et  $Z_{S\mathcal{N}}$ . Une fois les voisinages sphériques créés, ils sont projetés dans le plan image pour retrouver les intensités en utilisant une interpolation au plus proche voisin pour les coordonnées non entières.

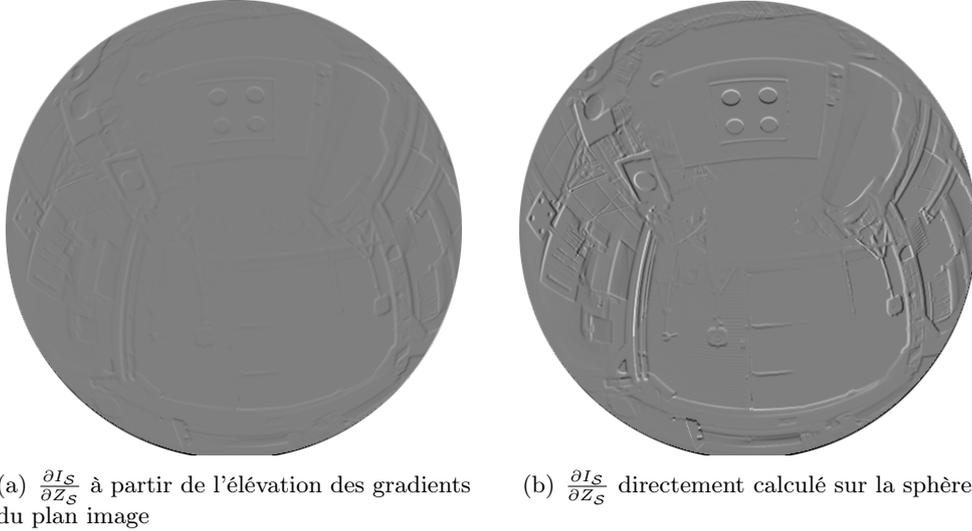


FIGURE 3.3 – Calcul adapté de gradients. (a) et (b) comparent deux manières de calculer les gradients. Le gradient sphérique apporte une information plus détaillée que l'élévation des gradients de l'image.

### 3.3 Application sur un robot cartésien

Cet ensemble d'expérimentations est réalisé sur un robot cartésien (Figure 3.7(a)) muni d'une caméra fisheye (Figure 3.4) montée sur son effecteur. Le développement a été fait en C++ avec la librairie ViSP [Marchand 2005b]. Dans cette section, trois types d'expériences sont présentés :

- Le but de la première expérimentation est de montrer la réussite de l'asservissement visuel omnidirectionnel photométrique, en utilisant les représentations présentées dans ce papier et de les comparer.

Méthode	type de coordonnées	critère
AVP-PI	plan image normalisé	$I(\mathbf{x}) - I(\mathbf{x})^*$
AVP-SC	sphérique cartésien	$I_S(\mathbf{X}_S) - I_S(\mathbf{X}_S)^*$
AVP-SP	sphérique pur	$I_S(\mathbf{S}) - I_S(\mathbf{S})^*$

TABLE 3.1 – Les différents critères comparés pour l’AV photométrique de caméra omnidirectionnelle.

- La seconde manipulation montre le comportement de l’asservissement visuel en utilisant l’AVP-PI, l’AVP-SC et l’AVP-SP (Tableau 3.1) avec une unique position désirée et trois positions et orientations différentes.
- La dernière expérimentation est un asservissement utilisant l’AVP-SC partant d’une position initiale plus éloignée de la position désirée.

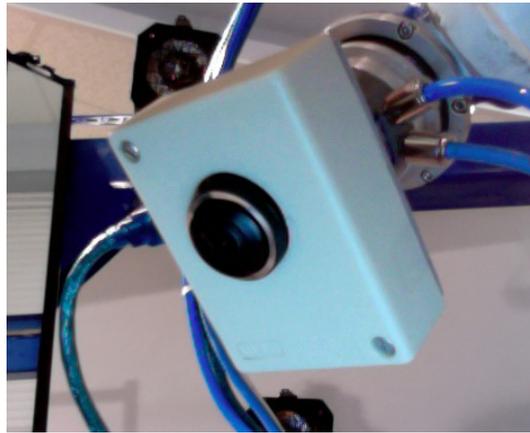


FIGURE 3.4 – Caméra fisheye utilisée pour l’asservissement visuel omnidirectionnel photométrique et montée sur l’effecteur du robot AFMA 6.

Pour toutes les expérimentations, la matrice d’interaction est uniquement calculée à la position désirée. De plus,  $Z$  est un paramètre inconnu et est supposé constant pour tout pixel tout au long du mouvement de la caméra. La réussite des expériences montre que la méthode est robuste à l’estimation grossière de  $Z$ .

### 3.3.1 Validation

Le premier objectif de ce jeu d’expérimentations est de montrer la réussite de l’asservissement visuel photométrique en vision omnidirectionnelle, dans ses différentes formulations. Dans un second temps, la comparaison des différents modes de calcul de gradient pour chaque représentation vise à sélectionner la meilleure manière de calculer les gradients. Cette comparaison se fera en terme de conditionnement de matrice d’interaction et de trajectoire réalisée.

Pour cette expérimentation, la translation de la caméra entre les deux positions est  $\Delta_X = 3,5\text{cm}$ ,  $\Delta_Y = -7,8\text{cm}$ ,  $\Delta_Z = 17,1\text{cm}$ , dans le repère du robot, sans variation d'orientation. Cette manipulation met en avant, qualitativement comme quantitativement, la meilleure approche en terme de distance entre la trajectoire de caméra et la droite liant les positions de départ et d'arrivée et de conditionnement de la matrice d'interaction. La figure 3.5 montre les images désirée et initiale ainsi que leur différence. Les trajectoires de caméra de cette figure sont obtenues avec les AVP-PI, AVP-SC et AVP-SP. Quand il y a convergence, l'erreur de positionnement est inférieure au dixième de millimètre en translation et  $0,02^\circ$  en rotation, et ce pour toutes les expérimentations avec le robot manipulateur.

Pour l'AVP-PI, les gradients de l'image sont calculés classiquement ou en utilisant les voisinages adaptés introduits dans la section 3.2.1.3. Il est clair que le mouvement de la caméra durant le processus d'asservissement n'est que légèrement modifié. Le tableau 3.2 confirme cette similitude avec un nombre de conditionnement quasi égal dans les deux cas.

Pour l'AVP-SC, deux trajectoires sont obtenues quand les gradients calculés dans le plan image sont élevés sur la sphère et la troisième est obtenue quand les gradients sphériques sont directement calculés sur l'image sphérique. Les résultats pour l'AVP-SP sont obtenus de la même manière. Le comportement des AVP-SC et AVP-SP sont clairement meilleurs quand les gradients sont calculés directement sur l'image sphérique et particulièrement pour l'AVP-SP. En fait, dans les deux cas, les gradients sphériques sont différents quand ils sont calculés directement sur l'image sphérique et lorsqu'ils sont calculés dans le plan image puis élevés (Figures 3.3(a) et 3.3(b)). Le tableau 3.2 montre aussi que les AVP-SC et AVP-SP sont mieux conditionnés quand les gradients sont directement calculés sur l'image sphérique.

méthode	conditionnement
AVP-PI gradients classiques	5,46
AVP-PI gradients adaptés	5,53
AVP-SC gradients classiques levés	6,60
AVP-SC gradients adaptés levés	6,74
AVP-SC gradients sphériques cartésiens	<b>4,79</b>
AVP-SP gradients classiques levés	10,07
AVP-SP gradients adaptés levés	10,13
AVP-SP gradients sphériques purs	6,73

TABLE 3.2 – Comparaison du conditionnement des matrices d'interaction, pour les différentes méthodes, à la position désirée.

### 3.3.2 Evaluation du comportement de l'AVP pour différentes poses initiales

A partir d'ici les AVP-PI, AVP-SC et AVP-SP sont comparés en utilisant respectivement le calcul adapté de gradients de l'image, le calcul direct de gradients

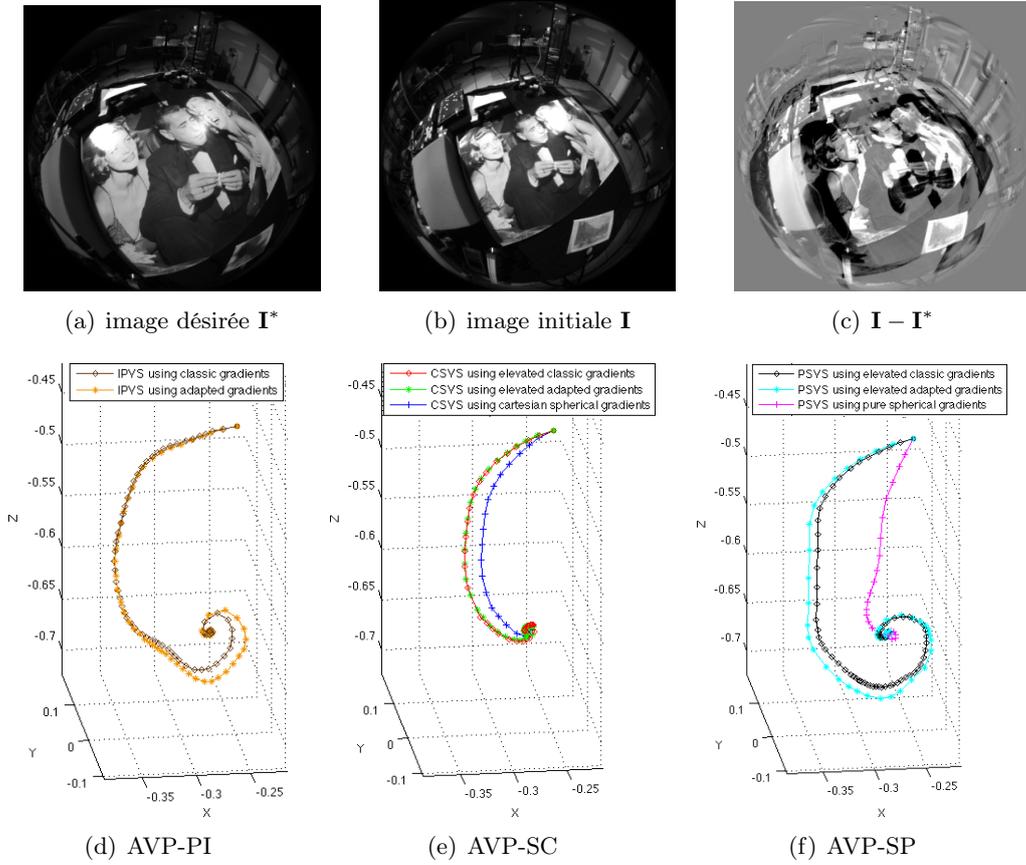


FIGURE 3.5 – Impact du calcul de gradients sur l’asservissement visuel photométrique. Les trois courbes représentent, de gauche à droite, les trajectoires de l’AVP-PI, de l’AVP-SC et de l’AVP-SP. Chaque graphe montre des trajectoires en utilisant la même méthode d’asservissement visuel mais en calculant différemment les gradients. La trajectoire la plus proche d’une droite rectiligne est obtenue en utilisant l’AVP-SC avec le calcul sphérique des gradients. Le meilleur conditionnement de matrice d’interaction est aussi obtenu avec cette méthode : 4,79 (Tableau 3.2).

sphériques cartésiens et le calcul direct de gradients sphériques purs. Ces calculs de gradients sont, en théorie et en pratique, plus cohérents que les autres comme la section précédente l’a présenté.

L’asservissement visuel omnidirectionnel photométrique est réalisé à partir de trois positions initiales différentes vers une unique position désirée. Les erreurs  $\{\Delta_X, \Delta_Y, \Delta_Z, \Delta_{R_X}, \Delta_{R_Y}, \Delta_{R_Z}\}$  entre la position désirée et la position initiale sont respectivement de  $\{0\text{cm}, 0\text{cm}, -8, 18\text{cm}, 0^\circ, 0^\circ, 0^\circ\}$ ,  $\{-8, 66\text{cm}, -0, 13\text{cm}, -4, 50\text{cm}, -2, 37^\circ, 11, 35^\circ, 7, 96^\circ\}$  et  $\{-15, 89\text{cm}, -1, 64\text{cm}, -1, 32\text{cm}, 2, 54^\circ, 14, 56^\circ, 7, 39^\circ\}$ .

La dernière expérience est la situation habituellement la plus difficile, avec une translation selon l’axe  $X$  de la caméra et une rotation selon son axe  $Y$ , ce qui donne

lieu à des ambiguïtés projectives. La fonction de coût conduit alors à des mouvements moins rectilignes et qui peuvent sembler erratiques. La figure 3.6 montre les images initiales de différence et la troisième position initiale engendre les plus importantes différences entre les trois trajectoires. L'évolution du résidu en fonction du temps (partie gauche de la figure 3.6), montre toujours que l'AVP-SC converge plus vite que les autres, même si, pour la troisième position initiale, cet avantage est plus faible.

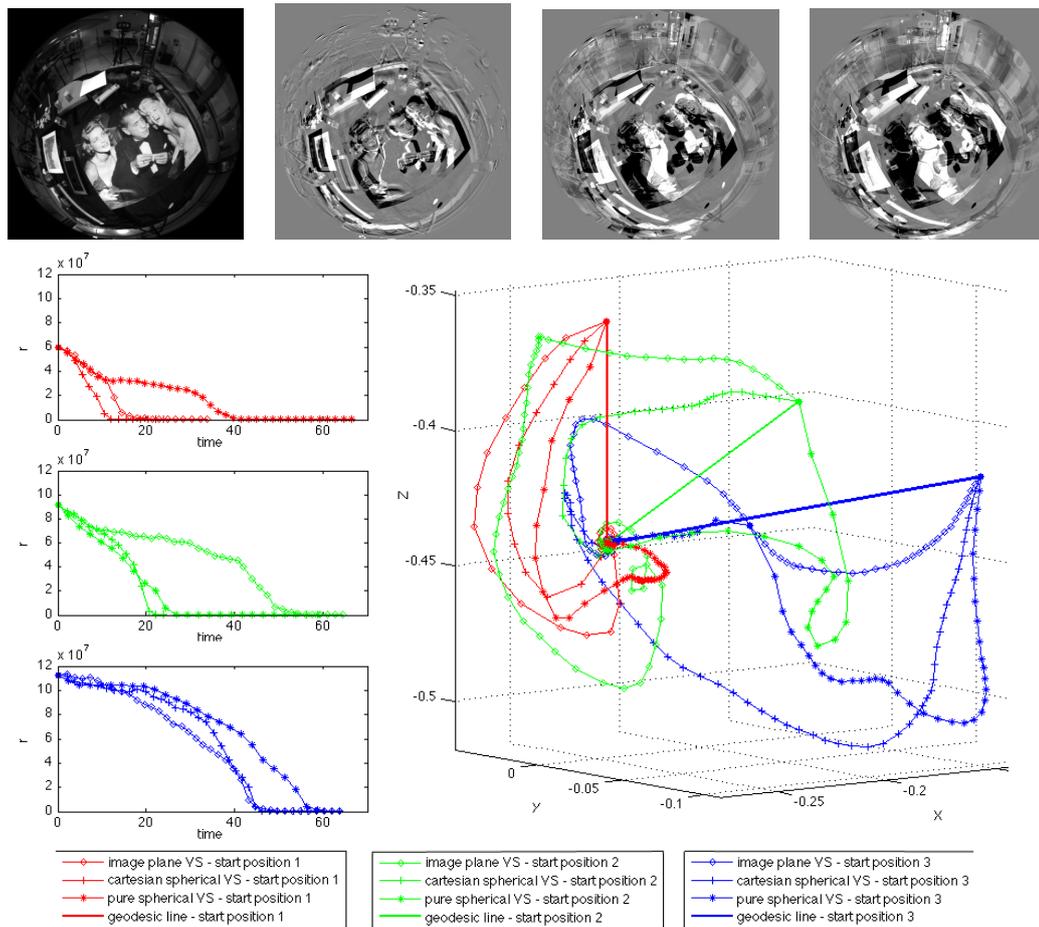


FIGURE 3.6 – Différentes positions initiales pour atteindre la même position désirée. Les images fisheye de différence montrent plusieurs conditions initiales. Sur la partie droite, les trajectoires de caméra pour atteindre la position désirée sont obtenues, pour chaque position initiale, en utilisant l'AVP-PI, l'AVP-SC et l'AVP-SP avec le calcul de gradient adapté (sphérique pour les deux derniers). A gauche, l'évolution du résidu par rapport au temps est toujours à l'avantage de l'AVP-SC.

### 3.3.3 Mouvement ample

Cette dernière expérimentation sur le robot cartésien vise à montrer que ce genre de technique peut être utilisée pour des mouvements relativement importants. Dans cette manipulation, l'écart entre la position initiale et la position désirée est principalement selon l'axe  $Z$  de la caméra. L'AVP-SC a été le seul à converger, ce qui s'explique par le fait qu'il est le seul à avoir une composante en  $Z$  dans sa formulation. Il est donc cohérent qu'il se comporte mieux que les autres quand un mouvement important en  $Z$  est nécessaire pour atteindre la position désirée.

L'erreur initiale de positionnement  $\{\Delta_X, \Delta_Y, \Delta_Z, \Delta_{R_X}, \Delta_{R_Y}, \Delta_{R_Z}\}$  est  $\{-41, 10\text{cm}, 39, 40\text{cm}, -40, 00\text{cm}, 2, 44^\circ, -9, 31^\circ, -10, 05^\circ\}$ . Malgré une différence importante en position et en orientation (Figure 3.7(d)), l'asservissement visuel photométrique réussit. La figure 3.7(e) montre la trajectoire suivie par la caméra dans le repère du robot.

## 3.4 Application sur un robot mobile

### 3.4.1 Formulation

Il s'agit d'un robot mobile (Figure 3.8(a)) non-holonome à deux degrés de liberté de type unicycle. L'objectif est de le commander pour le mener précisément à une pose désirée définie par une image. En positionnant judicieusement la caméra sur le robot et en admettant un alignement parfait de l'axe optique de la caméra sur l'axe de rotation du robot et de l'axe  $Y$  de la caméra sur l'axe longitudinal du robot mobile, seules les vitesses  $v_Y$  et  $\omega_Z$  sont à utiliser (Figure 3.8(b)).

La partie 3.3.3 a montré expérimentalement qu'utiliser l'image sphérique avec une représentation cartésienne faisait converger le système contrairement aux autres. Cette représentation apparaît donc comme toute indiquée pour le cas présent. On définit alors la matrice d'interaction d'un point pour le robot unicycle à partir de l'équation 2.11 :

$$\mathbf{L}_{\mathbf{x}_S} = \begin{pmatrix} \frac{X_S Y_S}{\rho} & Y_S \\ \frac{Y_S^2 - 1}{\rho} & -X_S \\ \frac{Y_S Z_S}{\rho} & 0 \end{pmatrix}. \quad (3.17)$$

Cette matrice est à remplacer dans l'équation 3.13 pour obtenir la matrice d'interaction liée à l'intensité en un point sphérique par rapport au mouvement de la caméra du robot mobile.

Cependant, il a été montré que dans le cas d'un asservissement visuel géométrique sur un point, la représentation sphérique pure est la mieux adaptée pour le contrôle d'un robot mobile de type unicycle [Tatsambon Fomena 2008]. En effet, dans le cas d'une caméra omnidirectionnelle catadioptrique, l'angle mort au centre de l'image fait que cette représentation n'est pas singulière si le point reste visible et, surtout, la matrice d'interaction déduite de l'équation 2.9 montre un



(a) position désirée



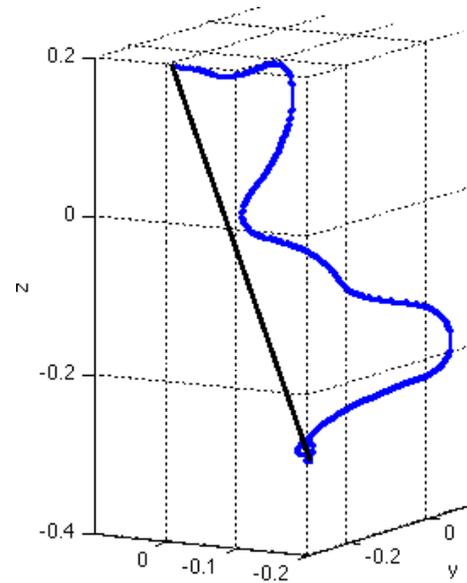
(b) position initiale



(c) image désirée



(d) différence initiale



(e) trajectoire de l'effecteur

FIGURE 3.7 – Expérimentation démarrant d'une position distante de  $\{-41, 10\text{cm}, 39, 40\text{cm}, -40, 00\text{cm}, 2, 44^\circ, -9, 31^\circ, -10, 05^\circ\}$  de la position désirée, avec d'importantes différences entre les images initiale et désirée (cf. (a)(c)). (e) montre en bleu la trajectoire de l'effecteur (noir : la géodésique).

découplage entre le contrôle de la vitesse de translation et la vitesse de rotation [Tatsambon Fomena 2008] :

$$\mathbf{L}_S = \begin{pmatrix} -\frac{\sin \theta \cos \phi}{\rho} & 0 \\ -\frac{\cos \theta}{\rho \sin \phi} & -1 \end{pmatrix}. \quad (3.18)$$

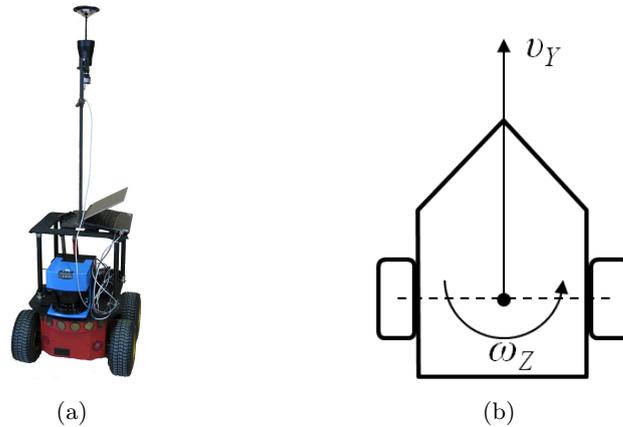


FIGURE 3.8 – (a) Le robot mobile Pioneer 3-AT avec une caméra catadioptrique placée verticalement sur un mat. (b) schéma illustrant les deux degrés de liberté contrôlés.

Cette matrice est à remplacer dans l'équation 3.9 pour obtenir la matrice d'interaction liée à l'intensité en un point sphérique par rapport au mouvement de la caméra.

Enfin, la représentation sphérique pure a l'avantage d'être minimale par rapport à la représentation cartésienne.

### 3.4.2 Application

Les représentations sphériques pure et cartésienne ont été confrontées. La première présentant de clairs avantages théoriques et la seconde se montrant particulièrement efficace en contrôle à six degrés de liberté.

Afin d'évaluer le comportement des AVP-SC et AVP-SP, une même position désirée (Figure 3.9) est à atteindre à partir de quatre positions initiales différentes (Tableau 3.3). Les positions et orientations du robot, utilisées dans cette section pour la mesure d'erreur en positionnement, sont obtenues par la méthode de localisation externe, à l'aide d'une balise infrarouge placée sur le robot, présentée dans la partie 2.2.5.2.

exp	$\Delta_X$	$\Delta_Y$	$\Delta_{R_Z}$
1	-48,64	-0,38	-17,56
2	-47,91	-1,43	9,44
3	48,24	12,99	16,70
4	46,17	-3,82	-4,37

TABLE 3.3 – Erreurs initiales dans le plan de déplacement du robot mobile pour l'AVP. Les écarts  $\Delta_X$  et en  $\Delta_Y$  sont en centimètres et les écarts angulaires  $\Delta_{R_Z}$  en degrés.



FIGURE 3.9 – Positionnement désiré. (a) l'image omnidirectionnelle acquise en cette position et (b) la vue externe du robot mobile en sa position désirée.

Les AVP-SC et AVP-SP réussissent à positionner le robot à l'endroit désiré pour les quatre positions initiales (Figures 3.10, 3.11, 3.12 et 3.13). Cependant, plus l'orientation est différente, plus difficile est la convergence. L'expérimentation 4 (Tableau 3.3) présente ici une faible différence entre les orientations initiale et désirée. L'asservissement converge dans ce cas mais pour une différence d'orientation plus importante, de l'ordre de  $10^\circ$  et  $15^\circ$ , l'asservissement n'a pas permis de converger vers la position désirée. Ces limites dépendent aussi de l'environnement et de l'écart entre l'image initiale et l'image désirée. Dans notre cas, l'environnement dans lequel évolue le robot est faiblement texturé avec un ensemble de portes dont les dimensions varient peu et assez régulièrement espacées, ce qui peut engendrer une convergence vers un minimum local si la rotation entre les deux images est trop importante. Malgré cela, il a été possible de réaliser une expérimentation avec un écart angulaire de près de  $18^\circ$  (expérimentation 1, Tableau 3.3).

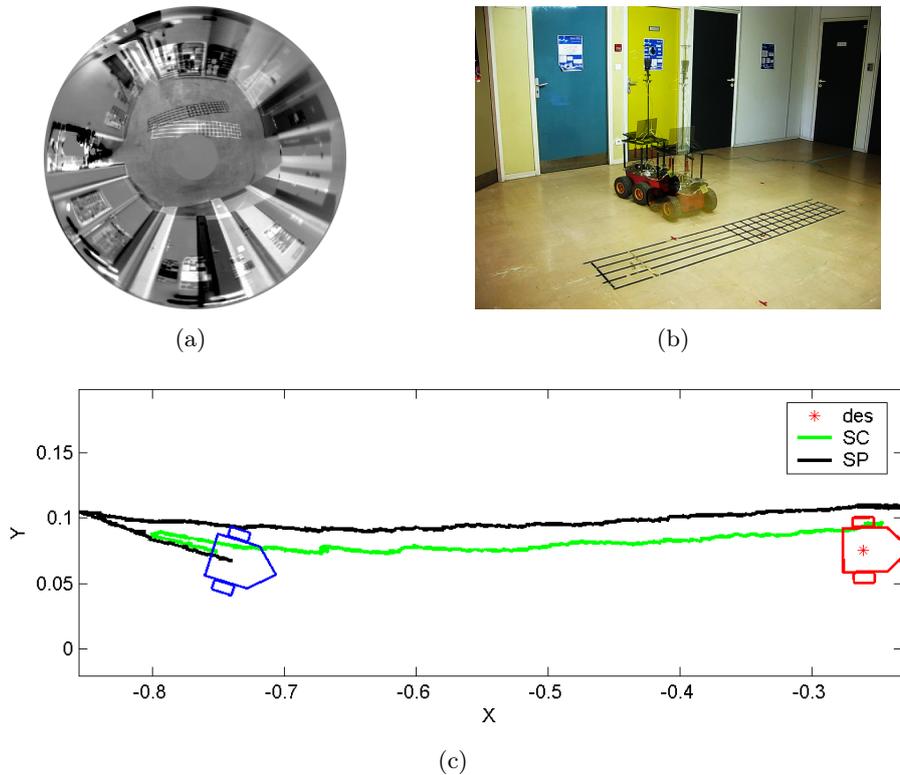


FIGURE 3.10 – Position initiale 1. (a) Différence entre l'image omnidirectionnelle désirée et l'image initiale. (b) Position initiale du robot avec la position désirée incrustée (Figure 3.9(b)). (c) Trajectoires obtenues par AVP-SC et AVP-SP.

Les AVP-SC et AVP-SP convergent vers la position désirée du robot pour les erreurs initiales présentées au tableau 3.3. Contrairement à l'utilisation du robot cartésien à six degrés de liberté, une erreur persiste au positionnement final par rapport à la position réellement désirée (Tableau 3.4). Ces erreurs sont principalement dues au fait que le robot est non-holonome, à la commande simple et aussi au fait que la correspondance des axes de la caméra et du robot n'est que supposée. Enfin, il est considéré que le monde est sphérique autour de la position désirée, c'est-à-dire que le paramètre  $\rho$  de profondeur d'un point 3D est constant en tout point de la sphère.  $\rho$  est fixé à 2,5 m à la position désirée dans les expérimentations de cette partie, ce qui est une approximation grossière, puisque la pièce où se situe le robot n'est pas sphérique. La prise en compte de ces derniers points en travaillant avec une commande adaptée à la non-holonomie et/ou en réalisant un étalonnage extrinsèque entre la caméra et le robot sont des pistes pour améliorer la précision du positionnement par ces techniques.

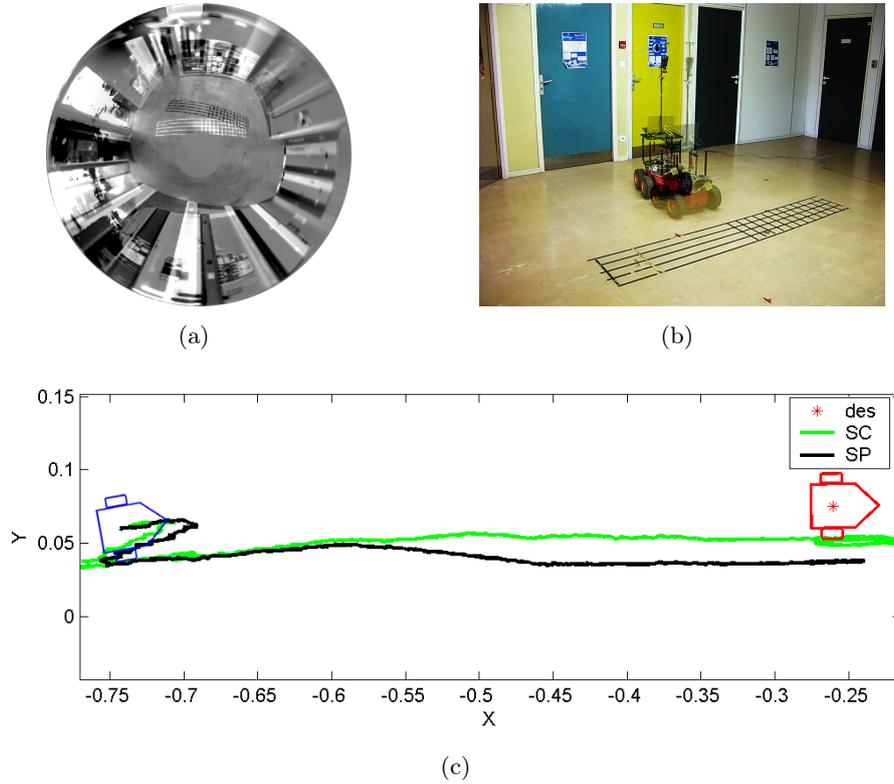


FIGURE 3.11 – Position initiale 2. (a) Différence entre l'image omnidirectionnelle désirée et l'image initiale. (b) Position initiale du robot avec la position désirée incrustée (Figure 3.9(b)). (c) Trajectoires obtenues par AVP-SC et AVP-SP.

méthode	exp	$\Delta_X$	$\Delta_Y$	$\Delta_{R_Z}$
AVP-SC	1	1,11	1,98	0,76
	2	3,74	-2,22	0,64
	3	0,52	3,10	1,16
	4	2,32	0,49	0,96
AVP-SP	1	1,44	3,35	0,79
	2	-0,60	-3,77	-0,15
	3	1,46	3,59	0,76
	4	1,21	-2,13	0,29

TABLE 3.4 – Erreurs finales de positionnement du robot mobile à convergence des AVP-SC et AVP-SP dans le plan de déplacement du robot mobile. Les écarts  $\Delta_X$  et en  $\Delta_Y$  sont en centimètres et les écarts angulaires  $\Delta_{R_Z}$  en degrés.

Même si certaines approximations sont réalisées, du point de vue de la commande ou de l'alignement des axes de la caméra et de ceux du robot, le positionnement reste relativement précis, particulièrement en orientation. Chaque représentation semble apporter son point fort avec une erreur finale en position plus faible en moyenne

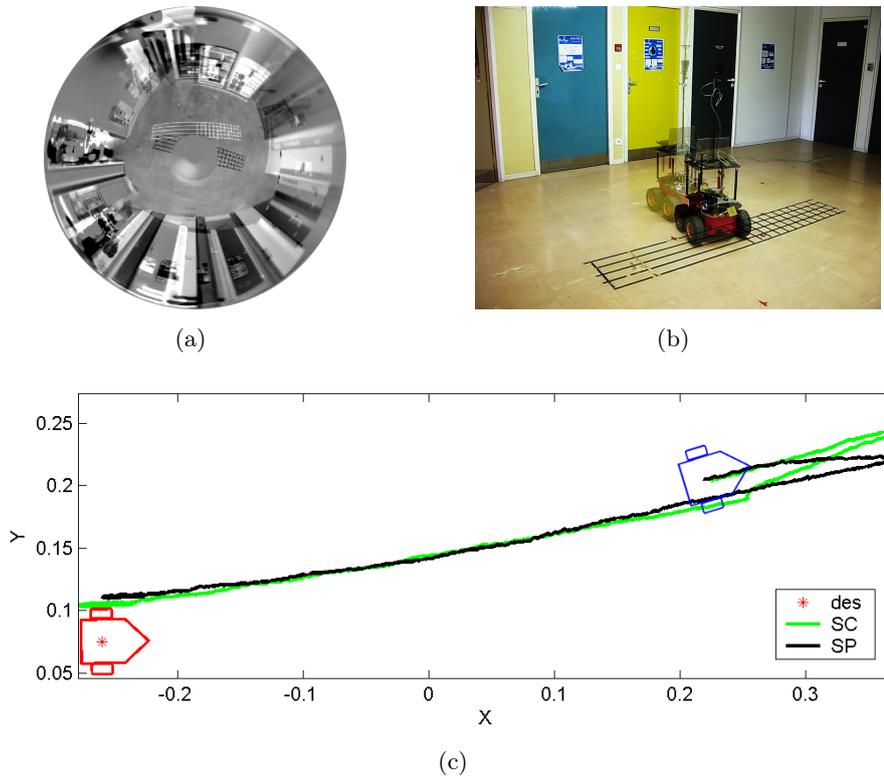


FIGURE 3.12 – Position initiale 3. (a) Différence entre l'image omnidirectionnelle désirée et l'image initiale. (b) Position initiale du robot avec la position désirée incrustée (Figure 3.9(b)). (c) Trajectoires obtenues par AVP-SC et AVP-SP.

pour l'AVP-SC et une erreur finale en orientation plus faible pour l'AVP-SP, dans les expérimentations réalisées (Figure 3.14).

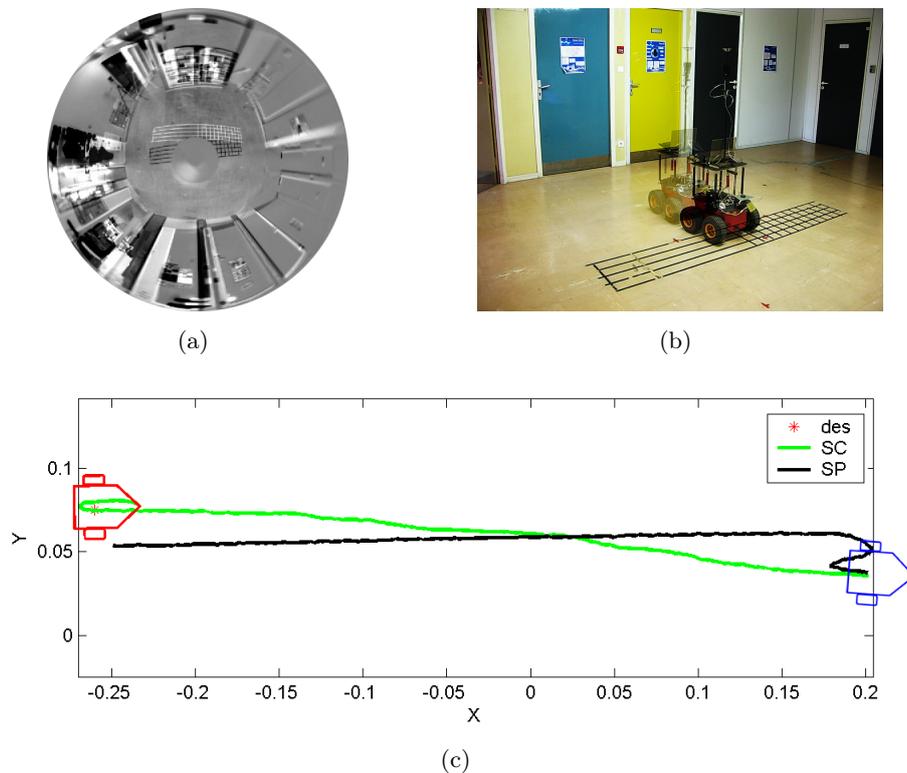


FIGURE 3.13 – Position initiale 4. (a) Différence entre l'image omnidirectionnelle désirée et l'image initiale. (b) Position initiale du robot avec la position désirée incrustée (Figure 3.9(b)). (c) Trajectoires obtenues par AVP-SC et AVP-SP.

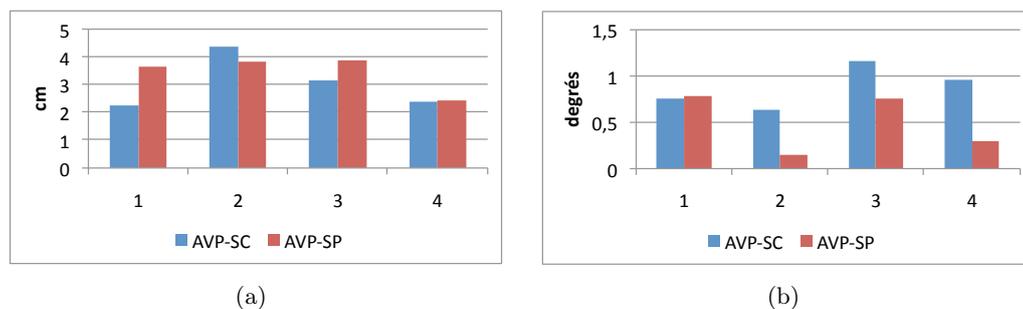


FIGURE 3.14 – Comparaison sous forme de diagrammes en barres des erreurs finales des AVP-SC et AVP-SP pour quatre positions initiales (en abscisses). (a) présente la norme de l'erreur en position et (b) en orientation.

**Trajectoire longue** Pour que l'asservissement visuel photométrique fonctionne, la différence entre les images initiale et désirée doit être relativement faible pour respecter l'hypothèse de l'OFCE (Equations 3.5, 3.8 et 3.12) et garantir la convergence. C'est pourquoi les positions initiales testées jusque là sont relativement proches de la

position désirée, de l'ordre de 50 cm en distance et de  $18^\circ$  maximum en orientation. En augmentant la distance entre les positions initiale et désirée, la différence entre l'image initiale et l'image désirée devient plus importante à mesure que la position initiale est éloignée de la position désirée.

Afin de montrer la robustesse de la méthode quant au respect des hypothèses de base, cette expérience vise à observer la réussite de l'asservissement photométrique pour une distance entre les positions initiale et désirée de 1,20 m. La différence entre les images initiale et désirée est par conséquent assez importante (Figure 3.15(b)). Malgré cette erreur initiale importante, l'AVP-SC permet de positionner précisément le robot à la position désirée avec une erreur finale de 0,5 cm en distance et  $0,3^\circ$  en orientation. Dans les mêmes conditions, l'AVP-SP, quant à lui, échoue. Le mouvement du robot par AVP-SP est identique au mouvement par AVP-SC au départ mais après s'être éloigné de la trajectoire optimale, l'AVP-SP reste bloqué dans un minimum local.

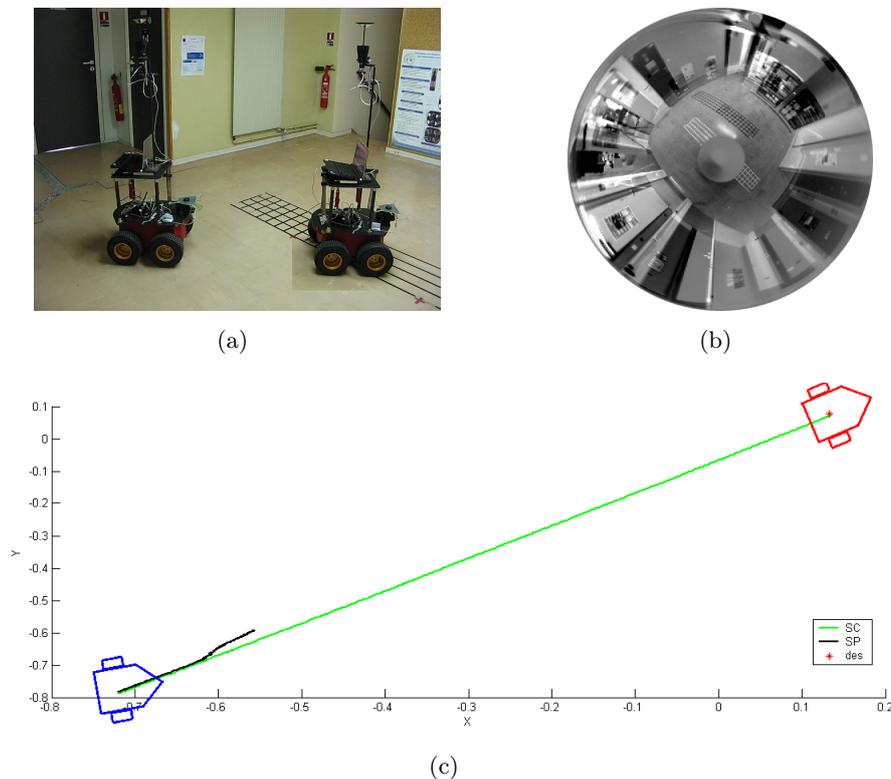


FIGURE 3.15 – Positionnement à partir d'une position initiale éloignée. (a) montre, d'un point de vue extérieur, la position initiale (à gauche) et la position désirée (à droite) du robot mobile. 1,20 m sépare ces deux positions, ce qui engendre une différence importante entre les images acquises à ces positions (b). (c) illustre la trajectoire du robot par AVP-SC, qui converge vers la bonne position, et AVP-SP, qui se perd dans un minimum local. Avec l'AVP-SC, l'erreur finale de positionnement est de 0,5 cm et  $0,3^\circ$ .

### 3.5 Application au suivi de chemin visuel

Le principe du suivi de chemin visuel est de définir des points de passage pour le robot à l'aide d'images acquises en ces points (Figure 3.16). L'idée est ensuite de replacer le robot à la position de départ du chemin et de réaliser des asservissements visuels successifs sur les images des points de passage. Ce type de procédé est intéressant pour atteindre une position finale désirée mais en suivant une trajectoire particulière.

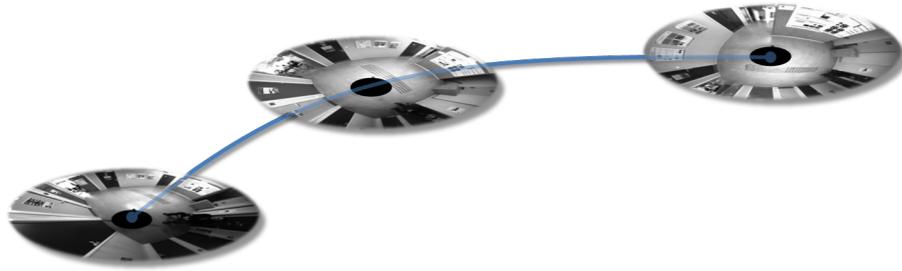


FIGURE 3.16 – Suivi de chemin visuel. Une image est associée à chaque point de passage et des asservissements visuels sont successivement réalisés sur ces images.

Pour le suivi de chemin défini par des images omnidirectionnelles, l'asservissement visuel géométrique 3D [COURBON 2009] peut être utilisé, en optimisant directement la pose de la caméra, ce qui permet de mieux maîtriser la trajectoire du robot pour de grands déplacements.

En considérant des écarts moyens en position et en orientation entre les points de passage, nous proposons d'utiliser l'asservissement visuel photométrique pour le suivi de chemin visuel. Une des difficultés est de déterminer si un point de passage est atteint pour passer au suivant. Il faut donc déterminer quand l'image courante et l'image désirée ont un écart minimal. La valeur du résidu obtenu par la somme des différences au carré (SSD) (Equation 3.1) pourrait être utilisée dans ce but mais d'autres critères existent pour savoir à quel point deux images sont identiques, comme la ZNCC : Zero-mean Normalized Cross Correlation. Cette dernière est connue pour être robuste aux changements d'illumination, contrairement à la SSD.

L'inconvénient de la SSD, c'est que cette mesure n'est pas robuste aux changements d'illumination, même faible. Un seuil fixe sur cette mesure pour savoir si l'image désirée a été atteinte deviendrait caduque dès que l'illumination de la scène change. La ZNCC, quant à elle, est efficace même si l'illumination globale évolue. Par conséquent, si le coefficient de ZNCC entre l'image courante et l'image désirée est supérieur à un seuil déterminé expérimentalement (0,7 dans les expérimentations) et correspondant à un alignement quasi parfait de deux images, l'asservissement est considéré terminé pour l'image désirée associée au point de passage. Le robot cherche alors à atteindre le point de passage suivant.

### 3.5.1 Mise en oeuvre

Pour réaliser le suivi de chemin visuel, nous avons fixé la vitesse longitudinale à une constante, ce qui signifie que seule la vitesse de rotation est à calculer. Ainsi, à partir de la matrice d'interaction d'un point cartésien d'une image sphérique pour un robot mobile non-holonyme (équation 3.17), on déduit directement celle où la vitesse longitudinale est constante :

$$\mathbf{L}_{\mathbf{x}_S} = \begin{pmatrix} Y_S \\ -X_S \\ 0 \end{pmatrix}, \quad (3.19)$$

qui ne dépend plus de la profondeur du point  $\rho$ . Le produit de ce jacobien avec le gradient sphérique cartésien amène à la matrice d'interaction liée à l'intensité en un point sphérique par rapport au mouvement de la caméra du robot mobile.

En formulation sphérique pure, la matrice d'interaction pour le suivi de chemin visuel est :

$$\mathbf{L}_S = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \quad (3.20)$$

qui ne dépend pas non plus de la profondeur du point 3D et qui est, de plus, constante en tout point sur la sphère. La matrice d'interaction photométrique en représentation sphérique pure est ensuite obtenue par un produit entre le gradient sphérique pur et le jacobien de l'équation 3.20.

L'avantage de considérer la vitesse longitudinale constante est d'éliminer le terme de profondeur  $\rho$  de la matrice d'interaction. Elle ne dépend alors que d'informations directement obtenues à partir de l'image : la reprojction des coordonnées d'un pixel sur la sphère et le gradient sphérique.

### 3.5.2 Expérimentation

La mémoire visuelle est constituée d'une séquence d'images établie par apprentissage. Pour cela, le robot est piloté manuellement et on acquiert une image omnidirectionnelle tous les 30 cm ou les 10°. D'autres stratégies d'élaboration de la base d'images peuvent être envisagées, comme par exemple lors de l'application d'une méthode d'exploration de l'environnement du robot [Merveilleux 2010].

L'expérience retenue pour valider l'approche est menée sur une trajectoire courbe de 3,5 m avec une base de vingt images de référence. La figure 3.17 montre la trajectoire réalisée et une partie des images clés. Le calcul des matrices d'interaction se fait ensuite hors-ligne avant de réaliser le suivi de chemin à proprement parler.

Le suivi de chemin, réalisé par asservissement visuel sur chaque image de référence successivement, donne des trajectoires légèrement différentes suivant la représentation sphérique utilisée (Figure 3.18). Les positions et orientations du robot sont obtenues par la méthode de localisation externe présentée dans la partie 2.2.5.2.

Il a été constaté lors des expérimentations que la représentation sphérique pure engendre une correction de l'orientation du robot plus rapide que la représentation

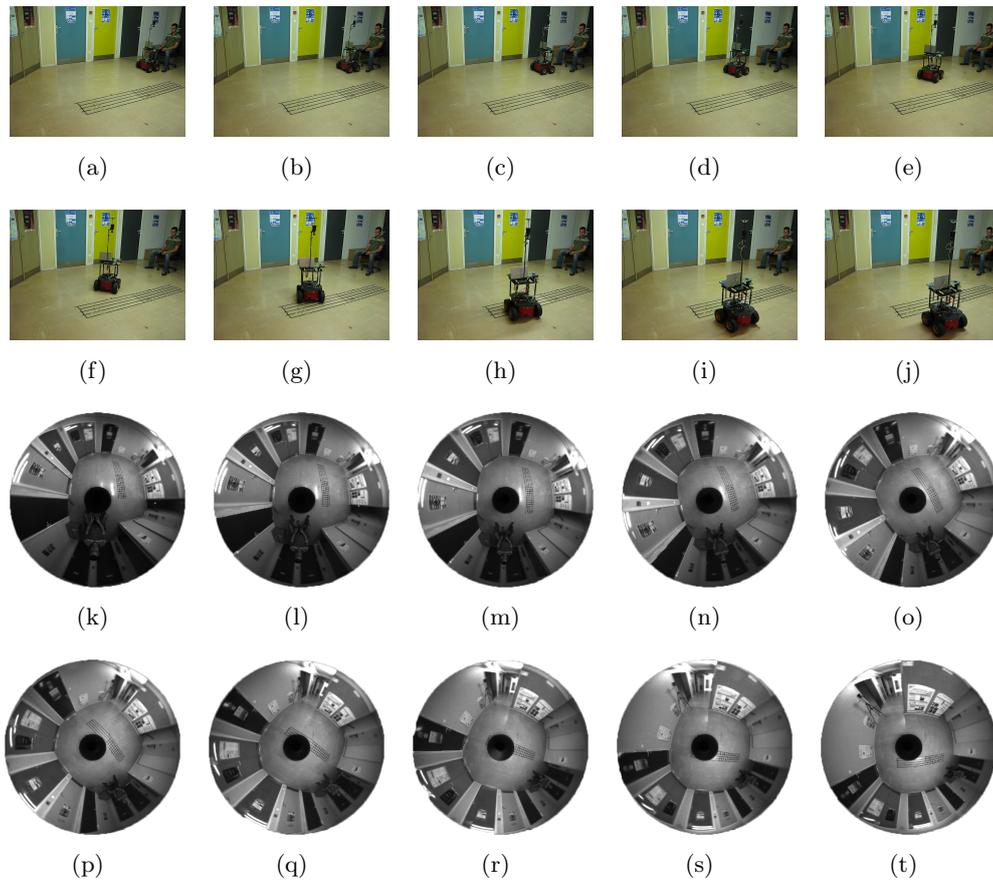


FIGURE 3.17 – Suivi de chemin visuel. (a-j) Vue externe de la trajectoire du robot. (k-t) Images clés acquises aux points de passage du robot.

sphérique cartésienne, ce qui explique que la trajectoire du robot par AVP-SP soit plus éloignée de la trajectoire de référence que celle par AVP-SC. En effet, en atteignant l'orientation désirée avant la position désirée, le robot ne peut plus atteindre cette position par la commande utilisée.

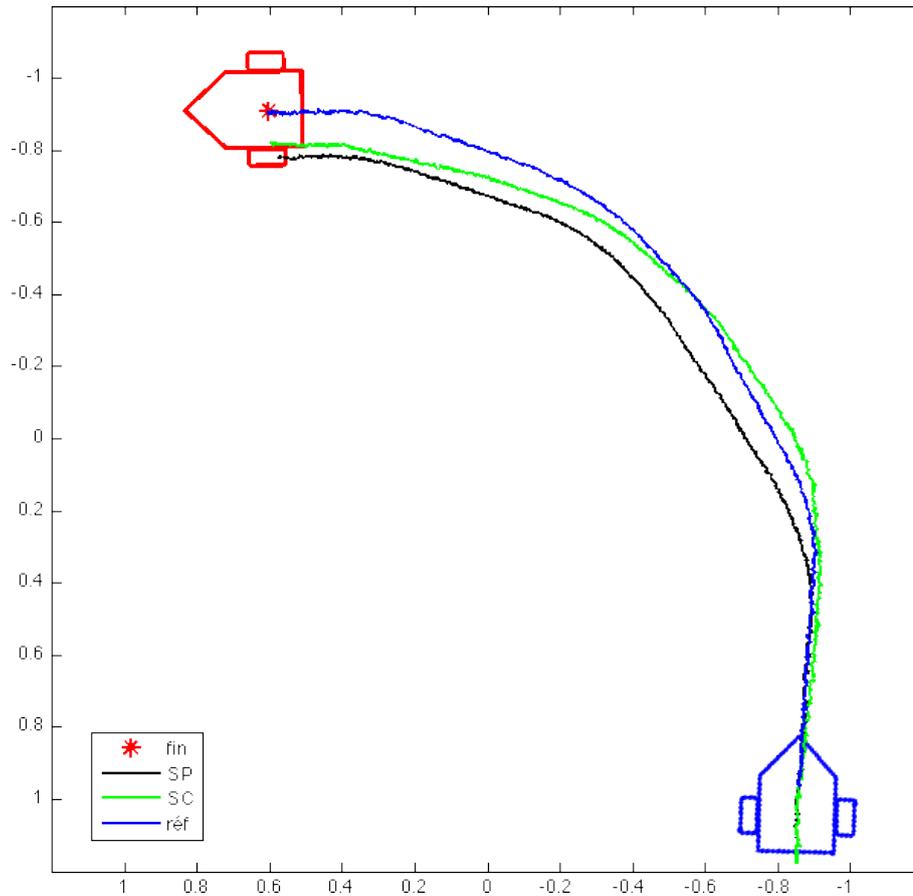


FIGURE 3.18 – Trajectoires réalisées par le robot pour un suivi de chemin visuel par AVP-SC et AVP-SP. L’axe du robot à sa position initiale correspond à la verticale de ce graphe. L’unité est le mètre.

### 3.6 Conclusion

La formulation de l’asservissement visuel photométrique omnidirectionnel soulève de nouvelles questions par rapport à l’asservissement visuel photométrique en vision perspective. En effet, même si une représentation polaire est possible pour une image perspective, la représentation cartésienne est généralement utilisée pour les méthodes reposant sur l’équation de contrainte du flot optique, que ce soit pour un asservissement, du suivi, ou autre. La formulation de l’asservissement photométrique omnidirectionnel dans le plan image qui a été faite dans ce chapitre peut être vue comme une extension directe de la vision perspective.

Cependant, à cause de la géométrie particulière de l’image omnidirectionnelle, il est apparu que réaliser l’asservissement photométrique en utilisant l’image sphérique est plus efficace qu’avec l’image omnidirectionnelle plane. Deux représentations sphériques ont été confrontées : l’une cartésienne, redondante, et l’autre angulaire, minimale mais avec une singularité. Les trois représentations ont été confrontées

---

et il ressort des expériences que la représentation cartésienne est celle qui se comporte le mieux pour des tâches d'asservissement à un, deux et six degrés de liberté. Cette dernière constatation est surtout vérifiée pour des déplacements importants où, à deux et six degrés de liberté, seul l'AVP-SC permet d'atteindre la position désirée. De plus, même dans les cas où l'AVP-SC et l'AVP-SP convergent tous les deux, l'AVP-SC donne généralement des trajectoires plus proches de l'optimal que l'AVP-SP.

Enfin, le champ d'application de l'asservissement photométrique omnidirectionnel est assez large avec la possibilité de réaliser un suivi de chemin visuel. Le travail présenté à ce sujet a juste pour but de valider l'approche mais une extension possible serait d'utiliser une technique globale de recherche d'image dans une base, de telle sorte que le robot mobile puisse se replacer sur sa trajectoire, quelque soit sa position initiale.



# Suivi photométrique de primitives géométriques

---

## Sommaire

---

<b>4.1</b>	<b>Introduction</b>	<b>97</b>
<b>4.2</b>	<b>Suivi de plan</b>	<b>97</b>
4.2.1	Suivi sphérique 2D d'un plan	98
4.2.2	Suivi sphérique 3D de plusieurs plans	101
4.2.3	Suivi et reconstruction simultanés de plusieurs plans en stéréovision omnidirectionnelle	105
<b>4.3</b>	<b>Suivi de droites verticales avec le <i>FOO</i></b>	<b>110</b>
4.3.1	Mise en correspondance et reconstruction simultanées	112
4.3.2	Suivi photométrique et reconstruction simultanés	120
<b>4.4</b>	<b>Conclusion</b>	<b>127</b>

---

## 4.1 Introduction

La plupart des techniques existantes de suivi d'objet dans une séquence d'image peut être divisée en deux principales classes : le suivi basé primitive ou le suivi basé modèle. La première approche se concentre sur le suivi de primitives 2D, telles que des primitives géométriques (point, segments, cercles, etc) ou les contours d'un objet (comme les contours actifs). La seconde approche utilise explicitement un modèle des objets suivis. Ce modèle peut être un modèle 3D menant principalement à un processus d'estimation de pose correspondant à un recalage entre les mesures dans l'image et la projection du modèle 3D (cf. chapitre 2). On peut aussi considérer des modèles 2D qui peuvent être une image de référence, ou un *template*. Dans ce cas, l'objectif est d'estimer le mouvement ou la déformation entre l'image courante et le template de référence. Le suivi différentiel, comme le tracker de Lucas-Kanade (KLT) [Lucas 1981, Hager 1998, Baker 2001, Benhimane 2007], est un exemple de ces techniques.

## 4.2 Suivi de plan

En considérant un modèle 2D, la déformation à estimer peut être une homographie. Ce modèle de mouvement 2D a l'avantage d'être exact pour un plan, quelque

soit le mouvement de la caméra. Ce modèle de mouvement a d'abord été défini pour les caméras à modèle de projection perspective mais est en fait valide pour toute caméra à projection centrale [Mei 2008, Salazar-Garibay 2009], y compris certaines caméras omnidirectionnelles dès lors qu'elles vérifient cette condition.

Le modèle de projection centrale unifié peut donc s'appliquer ici et il a même déjà été utilisé pour le suivi de plan dans les images omnidirectionnelles monoculaires [Mei 2008]. L'approche est similaire à celle proposée dans [Benhimane 2007] même si le recalage entre le template de référence et l'image courante est directement fait dans les images omnidirectionnelles. Quoiqu'il en soit, les gradients de l'image, nécessaires pour lier les intensités de l'image à la variation du mouvement, sont calculés dans le plan image. Cependant, comme il en a été question au chapitre 3, il a été montré que le filtrage d'images omnidirectionnelles doit être fait sur l'image sphérique plutôt que dans le plan image [Demonceaux 2009]. Rappelons ici que la géométrie sphérique, pour ces images, est adaptée pour garder les propriétés du produit de convolution, utilisé pour calculer les gradients, contrairement à la géométrie plane.

Cette partie vise à formuler le suivi simultané de plusieurs régions planes de l'image, contraintes par homographie, directement sur l'image sphérique. Dans un premier temps, le suivi d'un seul plan est présenté avec la représentation minimale. Ensuite, pour le suivi de plusieurs plans, plutôt que d'estimer une homographie sur la sphère pour chaque plan, la pose de la caméra, à un facteur d'échelle près, et les plans 3D sont reconstruits. Enfin, basé sur cette nouvelle formulation sphérique, une méthode de reconstruction complète de plans à la bonne échelle et de suivi simultané de régions planes dans les images stéréoscopiques est proposée.

## 4.2.1 Suivi sphérique 2D d'un plan

### 4.2.1.1 Formulation

**Transformation projective associée à un plan 3D** Que la caméra se déplace par rapport au plan ou que le plan se déplace par rapport à la caméra, les projections correspondantes du plan 3D dans deux images sont liées par une homographie. En considérant la rotation  ${}^c_2\mathbf{R}_{c_1}$  et la translation  ${}^c_2\mathbf{t}_{c_1}$  entre les deux positions de la caméra dans le temps, l'homographie  $\mathbf{H}$  est définie à un facteur d'échelle près [Mei 2008] :

$$\mathbf{H} \propto {}^c_2\mathbf{R}_{c_1} - \frac{{}^c_2\mathbf{t}_{c_1} {}^c_1\mathbf{n}^\top}{{}^c_1d}, \text{ avec } {}^c_2\mathbf{t}_{c_1} \in \mathbb{R}^3 \text{ et } {}^c_2\mathbf{R}_{c_1} \in \mathbf{SO}(3). \quad (4.1)$$

${}^c_1\mathbf{n}$  est la normale au plan exprimée dans le repère de la première caméra et  ${}^c_1d$  la distance orthogonale du centre de la caméra au plan. L'homographie est une matrice  $3 \times 3$  et a neuf paramètres,  $(h_{1,1}..h_{3,3})$  et huit degrés de liberté, c'est pourquoi  $h_{3,3}$  est fixé à 1.

L'homographie  $\mathbf{H}$  lie deux points correspondants  ${}^c_1\mathbf{X}_S$  et  ${}^c_2\mathbf{X}_S$  sur les deux sphères, à un facteur près :

$${}^c_2\mathbf{X}'_S = \alpha {}^c_2\mathbf{X}_S = \mathbf{H} {}^c_1\mathbf{X}_S. \quad (4.2)$$

En fait, une homographie relie deux lignes de vue correspondantes, ce qui est mieux illustré par la contrainte de colinéarité [Mei 2007] :

$${}^{c_2}\mathbf{X}_S \times (\mathbf{H}^{c_1}\mathbf{X}_S) = 0. \quad (4.3)$$

Cette relation est vérifiée pour tout facteur scalaire appliqué à  ${}^{c_1}\mathbf{X}_S$  et/ou à  ${}^{c_2}\mathbf{X}_S$  et puisque l'on travaille sur la sphère d'équivalence, le point sphérique  ${}^{c_2}\mathbf{X}_S$  correspondant à  ${}^{c_1}\mathbf{X}_S$  après avoir été transféré par  $\mathbf{H}$  est :

$${}^{c_2}\mathbf{X}_S = \frac{{}^{c_2}\mathbf{X}'_S}{\|{}^{c_2}\mathbf{X}'_S\|} \quad (4.4)$$

**Suivi sphérique de plan** Nous proposons une approche non-linéaire pour réaliser le suivi de plan. Cette approche vise à calculer l'homographie  $\mathbf{H}$  permettant de transférer les points sphériques de la sphère de la caméra de référence vers la sphère de la caméra courante en minimisant la somme quadratique des différences des intensités des images sphériques. Cela conduit à minimiser le critère :

$$\Delta_S = \sum_{i=1}^k (\mathbf{I}_S(\mathbf{H}\mathbf{X}_{S,i}^*) - \mathbf{I}_S(\mathbf{X}_{S,i}^*))^2 \quad (4.5)$$

où  $\mathbf{X}_S^*$  est un point de la région de référence.

Soit  $\mathbf{h} = (h_{1,1}, h_{1,2}, h_{1,3}, h_{2,1}, h_{2,2}, h_{2,3}, h_{3,1}, h_{3,2})$  une représentation vectorielle de l'homographie. Le vecteur de primitives  $\mathbf{I}_S(\mathbf{h})$  est l'ensemble des intensités de la région d'intérêt de l'image sphérique, obtenue après transfert du template de référence par homographie. L'erreur à réguler est alors :

$$\mathbf{e} = \mathbf{I}_S(\mathbf{h}) - \mathbf{I}_S^* \quad (4.6)$$

En imposant une décroissance exponentielle de cette erreur, sa variation temporelle est :

$$\dot{\mathbf{e}} = -\lambda \mathbf{e} \quad (4.7)$$

qui ne dépend que de  $\dot{\mathbf{I}}_S$ . La variation de l'intensité à un point donné sur la sphère est définie grâce à la l'OFCE sphérique cartésienne (équation 3.12), en se basant sur l'hypothèse d'illumination constante de pixels se correspondant dans le temps :

$$\mathbf{I}_S(\mathbf{X}_S + d\mathbf{X}_S, t + dt) = \mathbf{I}_S(\mathbf{X}_S, t). \quad (4.8)$$

Si  $d\mathbf{X}_S$  est assez faible, l'OFCE sphérique cartésienne est valide :

$$\nabla \mathbf{I}_S^T \dot{\mathbf{X}}_S + \dot{\mathbf{I}}_S = 0 \quad (4.9)$$

avec  $\nabla \mathbf{I}_S$  le gradient sphérique cartésien de  $\mathbf{I}_S(\mathbf{X}_S, t)$  calculé directement sur la sphère comme introduit dans la partie 3.2.3. Par conséquent, la variation de  $\mathbf{I}_S$  s'exprime par :

$$\dot{\mathbf{I}}_S = -\nabla \mathbf{I}_S^T \dot{\mathbf{X}}_S. \quad (4.10)$$

$\dot{\mathbf{X}}_S$  peut être décomposé en :

$$\dot{\mathbf{X}}_S = \frac{d\mathbf{X}_S}{dt} = \frac{\partial \mathbf{X}_S}{\partial \mathbf{h}} \frac{d\mathbf{h}}{dt} = \mathbf{G}_{\mathbf{X}_S \mathbf{h}} \dot{\mathbf{h}}. \quad (4.11)$$

Ainsi, les équations (4.10) et (4.11) mènent au jacobien de l'image lié à  $\mathbf{I}_S$  au point sphérique  $\mathbf{X}_S$  :

$$\mathbf{G}_{\mathbf{I}_S}(\mathbf{X}_S) = -\nabla \mathbf{I}_S^\top \mathbf{G}_{\mathbf{X}_S \mathbf{h}}. \quad (4.12)$$

Ensuite, la variation du vecteur homographie  $\mathbf{h}$  est, selon un schéma d'optimisation de type Gauss-Newton :

$$\dot{\mathbf{h}} = -\lambda \mathbf{G}_{\mathbf{I}_S}(\mathbf{X}_S)^+ (\mathbf{I}_S(\mathbf{h}) - \mathbf{I}_S^*) \quad (4.13)$$

où  $\lambda$  est un gain proportionnel qui règle le temps de convergence de la minimisation et  $\mathbf{G}_{\mathbf{I}_S}(\mathbf{X}_S)^+$ , la pseudo-inverse à gauche de  $\mathbf{G}_{\mathbf{I}_S}(\mathbf{X}_S)$ .

Dans un tel procédé, une méthode d'optimisation de type Levenberg-Marquardt est souvent utilisée car elle permet en général une meilleure et plus rapide convergence :

$$\dot{\mathbf{h}} = -\lambda (\mathbf{Q} + \mu \text{diag}(\mathbf{Q}))^{-1} \mathbf{G}_{\mathbf{I}_S}(\mathbf{X}_S)^\top (\mathbf{I}_S(\mathbf{h}) - \mathbf{I}_S^*) \quad (4.14)$$

avec  $\mathbf{Q} = \mathbf{G}_{\mathbf{I}_S}(\mathbf{X}_S)^\top \mathbf{G}_{\mathbf{I}_S}(\mathbf{X}_S)$ .

$\mathbf{H}$  est initialisée avec la matrice identité pour la première paire d'image et sa valeur précédente est ensuite utilisée comme valeur initiale pour une nouvelle image.  $\mathbf{H}$  est mise à jour grâce à  $\dot{\mathbf{h}}$  par addition terme à terme.

Enfin, un estimateur robuste basé sur les M-Estimeurs [Huber 1981] est mis en place pour rejeter les aberrations et améliorer la robustesse par rapport aux changements d'illumination, réflexions spéculaires et occultations partielles (cf. partie 1.2.3.2).

#### 4.2.1.2 Application : Caméra omnidirectionnelle mobile

Pour évaluer la précision du suivi d'un plan en vision omnidirectionnelle monoculaire en utilisant la formulation sphérique, un robot mobile a été utilisé pour trois expérimentations. Une caméra paracatadioptrique, acquérant des images de  $640 \times 480$  pixels ( $450 \times 450$  utiles), est montée verticalement sur un robot Pioneer 3-AT (Figure 3.8(a)). Une rotation pure (Figure 4.1) et une translation pure (Figure 4.2) ont été réalisées pour évaluer simplement l'efficacité de l'approche. Deux plans de la scène sont suivis indépendamment pour comparer le mouvement estimé.

La figure 4.1(d) présente les rotations extraites des homographies estimées, tout au long de la séquence de rotation à  $360^\circ$ . La discontinuité est volontairement laissée pour montrer que le dernier angle estimé est le même que le premier. L'erreur finale est d'environ  $0,35^\circ$ , pour peu que le robot ait exactement fait un tour complet.

Pour l'expérimentation de translation pure (Figure 4.2), la distance mesurée à la main entre le plan et la position initiale de la caméra est utilisée pour obtenir une trajectoire à la bonne échelle. Le suivi mène à une erreur finale d'estimation de

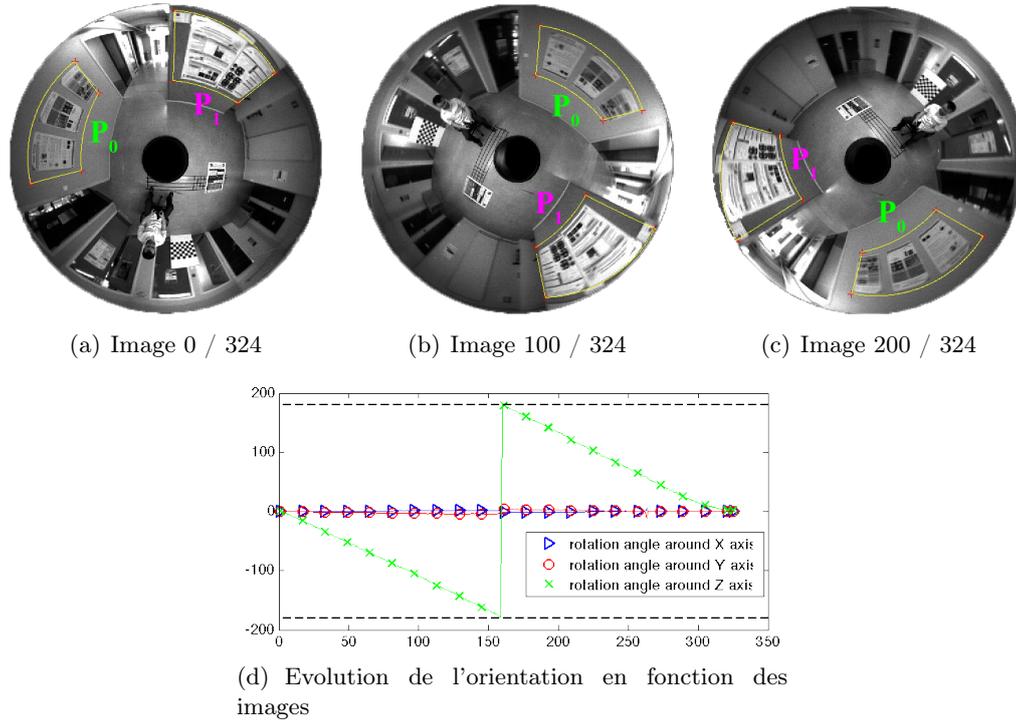


FIGURE 4.1 – Rotation pure. Le suivi sphérique de plan réussit tout au long de la séquence de rotation de  $360^\circ$  de la caméra, pour  $P_0$  et  $P_1$ , les deux plans de l'image. Les plans sont ici suivis indépendamment pour comparer les résultats d'estimation.

position relative de (erreur finale sur distance réelle) : 3,3% pour le plan  $P_0$  et 1,8% pour le plan  $P_1$ .

Finalement, une trajectoire plus complexe est réalisée par le robot (Figure 4.3). Il commence par se déplacer selon une trajectoire courbe qui le ramène proche de sa position initiale pour ensuite continuer en ligne droite. Dans cette séquence, le mouvement des régions planes est très important tout comme les effets de perspective. De plus, chaque plan suivi est successivement occulté (Figure 4.3(b)) et présente quelques réflexions spéculaires. La trajectoire est bien estimée et la méthode montre sa robustesse aux occultations partielles grâce au M-Estimateur. Après un mouvement complet d'environ 11 m, l'erreur entre les positions finales estimée et mesurée est en-dessous de 1%.

## 4.2.2 Suivi sphérique 3D de plusieurs plans

### 4.2.2.1 Suivi sphérique 3D d'un plan

Le suivi d'un plan en estimant directement les coefficients de la matrice de transformation projective  $\mathbf{H}$  mène à l'ensemble minimal de paramètres à estimer. Nous formulons maintenant, le suivi dans l'espace euclidien. Par conséquent, la rotation, la translation de la caméra, la normale et la distance au plan sont explicitement esti-

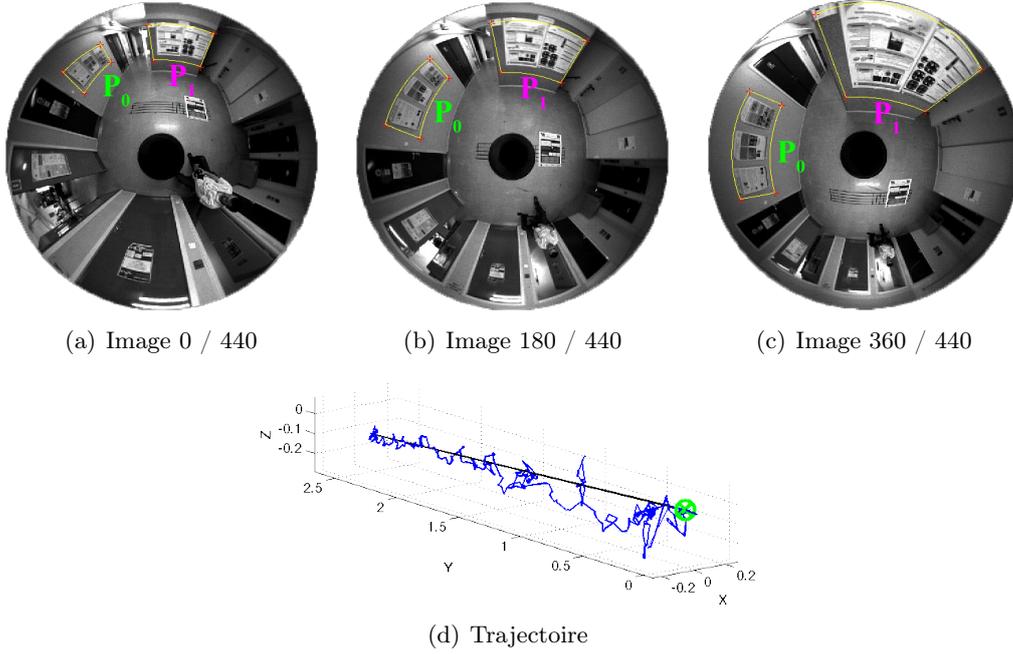


FIGURE 4.2 – Translation pure. Le chemin parcouru par le robot est de 2,70 m principalement selon l'axe  $Y$  de la caméra (l'axe principal du robot). A cause de leur différence de taille dans l'image, le suivi de  $P_0$  engendre une erreur plus importante que le suivi de  $P_1$ . Une fois encore chaque plan est suivi individuellement et séparément pour comparer les résultats d'estimation de mouvement. La trajectoire présentée (d) peut paraître erratique mais, d'une part, les variations diminuent à mesure que la caméra se rapproche du plan observé et, d'autre part, ces variations restent dans un volume maximal de  $10 \text{ cm}^3$ . De plus, aucun filtrage n'est réalisé.

més. Cette formulation permet d'envisager le suivi de plusieurs plans en estimant un unique mouvement de caméra, quand la caméra se déplace dans un environnement structuré, par exemple.

Le critère à minimiser pour le suivi de plan est similaire à celui de l'équation 4.5, à ceci près que les paramètres à optimiser sont la pose relative de la caméra et le plan 3D :

$$\hat{\mathbf{H}} = \min_{\mathbf{h}} \sum_{i=1}^k (\mathbf{I}_S(\mathbf{H}\mathbf{X}_{S,i}^*) - \mathbf{I}_S(\mathbf{X}_{S,i}^*))^2 \quad (4.15)$$

avec  $\mathbf{h} = (\mathbf{n}, d, \mathbf{r})^\top$  une représentation vectorielle de  $\mathbf{H}$  sur  $\mathbf{SE}(3)$ .  $\mathbf{r}$  est une représentation vectorielle de la pose relative et  $\mathbf{H}$  est calculée grâce à ces paramètres et l'équation (4.1).

Le raisonnement est similaire à la formulation présentée dans la partie 4.2.1.1 pour lier la variation d'intensité d'un point de l'image sphérique au mouvement de

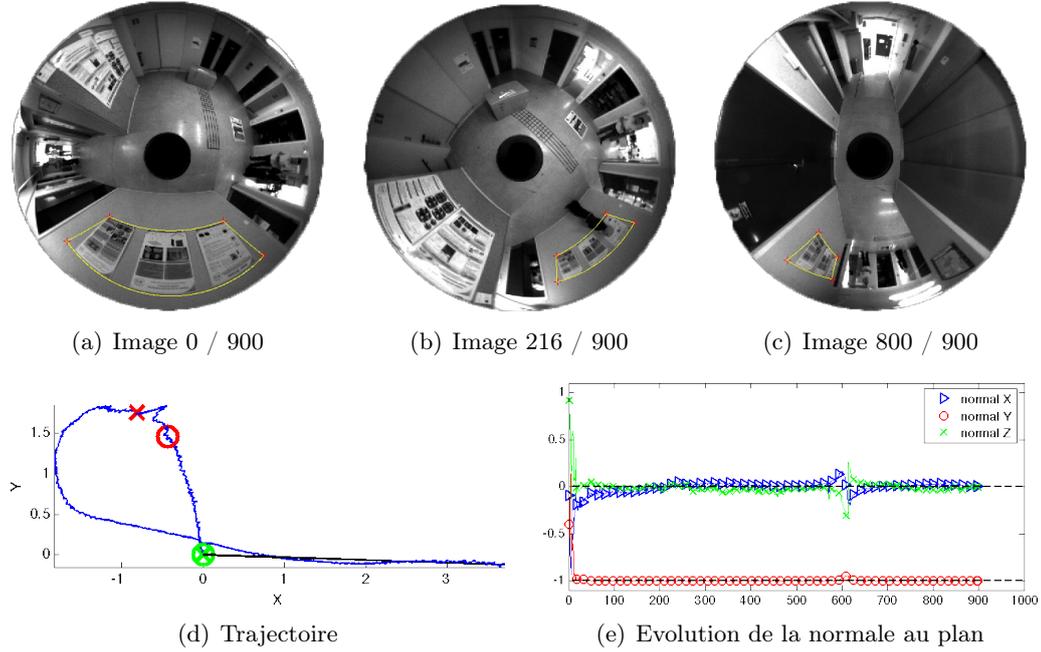


FIGURE 4.3 – Suivi d’un plan et estimation de mouvement. Mouvement général sur 11 m en présence d’occultation partielle du plan d’intérêt. L’écart type des estimation de la coordonnées  $Z$  est d’environ 4 cm.

caméra sauf que le vecteur  $\mathbf{h}$  étant différent, la décomposition de  $\dot{\mathbf{X}}_S$  l’est aussi :

$$\dot{\mathbf{X}}_S = \frac{d\mathbf{X}_S}{dt} = \frac{\partial \mathbf{X}_S}{\partial \mathbf{h}} \frac{d\mathbf{h}}{dt} = \mathbf{G}_{\mathbf{X}_S \mathbf{h}} \dot{\mathbf{h}} = [\mathbf{J}_{\mathbf{X}_S} \quad | \quad \mathbf{L}_{\mathbf{X}_S}] \begin{bmatrix} \dot{\mathbf{n}} \\ \dot{d} \\ \mathbf{v} \end{bmatrix} \quad (4.16)$$

où  $\mathbf{v} = (v, \omega)^\top$  est le torseur cinématique de la pose relative de la caméra. Le jacobien  $\mathbf{J}_{\mathbf{X}_S}$  exprime la variation d’un point sphérique  $\mathbf{X}_S$  par rapport au plan 3D, dont la normale est contrainte, et  $\mathbf{L}_{\mathbf{X}_S}$  est la matrice d’interaction, pour une pose relative, cette fois-ci (Equation 2.11).  $\mathbf{L}_{\mathbf{X}_S}$  dépend principalement des coordonnées d’un point sphérique, mis à part  $\rho$ , qui est la distance entre le point 3D et l’origine du repère caméra courant (Equation. 1.5). Cette distance est obtenue en faisant l’intersection de la ligne de vue du point  $\mathbf{X}_S$  avec le plan 3D.

En considérant la nouvelle variation de  $\mathbf{X}_S$  (Equation (4.16)), on obtient le jacobien image liant  $\mathbf{I}_S$  au point sphérique  $\mathbf{X}_S$  :

$$\mathbf{G}_{\mathbf{I}_S}(\mathbf{X}_S) = -\nabla \mathbf{I}_S^\top \mathbf{G}_{\mathbf{X}_S \mathbf{h}}, \quad (4.17)$$

qui semble être identique à celui de l’équation 4.12 mais  $\mathbf{G}_{\mathbf{X}_S \mathbf{h}}$  est clairement différent (Equation (4.16)).

### 4.2.2.2 Suivi sphérique 3D de plusieurs plans

Pour traiter le suivi de plusieurs plans quand la caméra est mobile, le problème peut être formulé pour calculer une unique pose relative et plusieurs normales et distances. Dans une telle situation, les intensités d'une région plane sont utilisées pour calculer le plan qui lui correspond tandis que les intensités de toutes les régions suivies sont utilisées pour calculer la pose relative. Par conséquent, le jacobien liant la variation des intensités de l'image au mouvement de la caméra et les paramètres des  $p$  plans est :

$$\mathbf{G}_{\mathbf{I}_S} = \begin{bmatrix} -\nabla \mathbf{I}_S^{(1)} \\ -\nabla \mathbf{I}_S^{(2)} \\ \dots \\ -\nabla \mathbf{I}_S^{(p)} \end{bmatrix}^\top \left[ \begin{array}{cccc|c} \mathbf{J}_{\mathbf{x}_S}^{(1)} & 0 & \dots & 0 & \mathbf{L}_{\mathbf{x}_S}^{(1)} \\ 0 & \mathbf{J}_{\mathbf{x}_S}^{(2)} & \dots & \vdots & \mathbf{L}_{\mathbf{x}_S}^{(2)} \\ \vdots & \dots & \ddots & 0 & \vdots \\ 0 & \dots & 0 & \mathbf{J}_{\mathbf{x}_S}^{(p)} & \mathbf{L}_{\mathbf{x}_S}^{(p)} \end{array} \right] \quad (4.18)$$

où  $(i)$  illustre le numéro du plan qui est lié au jacobien et son ensemble de primitives correspondant. Ensuite, les vecteurs d'erreurs correspondant à chaque plan sont empilés et sont reliés, grâce au dernier jacobien à plans multiples, aux vitesses de la caméra et des paramètres des plans :

$$\mathbf{V} = [\dot{\mathbf{n}}^{(1)} \quad \dot{d}^{(1)} \quad \dot{\mathbf{n}}^{(2)} \quad \dot{d}^{(2)} \quad \dots \quad \dot{\mathbf{n}}^{(p)} \quad \dot{d}^{(p)} \quad \mathbf{v}]^\top \quad (4.19)$$

### 4.2.2.3 Mise à jour des paramètres

La variation des paramètres des plans et du vecteur de pose relative est calculée grâce à une optimisation de type Levenberg-Marquardt :

$$\mathbf{V} = -\lambda (\mathbf{Q} + \mu \text{diag}(\mathbf{Q}))^{-1} \mathbf{G}_{\mathbf{I}_S}^\top \begin{bmatrix} \mathbf{I}_S(\mathbf{h}^{(1)}) - \mathbf{I}_S^* \\ \vdots \\ \mathbf{I}_S(\mathbf{h}^{(p)}) - \mathbf{I}_S^* \end{bmatrix} \quad (4.20)$$

avec  $\mathbf{Q} = \mathbf{G}_{\mathbf{I}_S}^\top \mathbf{G}_{\mathbf{I}_S}$  et  $\mathbf{h}^{(i)} = (\mathbf{n}^{(i)}, \mathbf{d}^{(i)}, \mathbf{r})^\top$ .

La pose relative est mise à jour à l'itération  $it$  avec l'application exponentielle de  $\mathbf{se}(3)$  [Ma 2004] par  ${}^{c_2}\mathbf{M}_{c_1}^{it} = {}^{c_2}\mathbf{M}_{c_1}^{it-1} e^{[\mathbf{v}]}$ . Les paramètres des plans sont mis à jour par addition de leur variation temporelle.

### 4.2.2.4 Application : Caméra omnidirectionnelle mobile

Le suivi de plusieurs plans est réalisé en utilisant la même séquence générale que dans la section 4.2.1.2. La figure 4.4(a) montre la première image avec les régions planes sélectionnées et qui sont suivies simultanément dans la séquence d'images. Les deux plans suivis sont visibles dans une partie importante de la séquence mais pas dans le couloir. Pour traiter ce type de problème, quand la position de la caméra est trop proche du plan et si la surface de la région correspondante de l'image est trop petite, ce plan particulier n'est plus suivi.

Le suivi est réalisé avec succès tout au long de la séquence et la trajectoire de la caméra (Figure 4.4(e)), et donc du robot, montre la stabilité de l'estimation, même en présence d'occultations partielles. L'écart type des coordonnées  $Z$  estimées est de 2,64 cm pour la séquence complète, 1,5 cm quand les deux plans sont suivis simultanément et 2,19 cm après avoir arrêté de suivre le plan  $\mathbf{P}_1$ . Pour comparer, le suivi du plan  $\mathbf{P}_0$  seul mène à un écart type de  $Z$  égal à 4 cm pour la séquence complète (une petite instabilité en occultation partielle). Les variations des paramètres estimés des normales, toujours dans le repère de la première position de caméra, illustre aussi la stabilité d'estimation (Figure 4.4(c) et 4.4(d)). Les plans réels sont verticaux, comme l'axe de la caméra et, par conséquent, la composante  $Z$  estimée est nulle. Les composantes  $X$  et  $Y$  dépendent de l'orientation du plan dans le premier repère caméra (Figure 4.3(a)).

### 4.2.3 Suivi et reconstruction simultanés de plusieurs plans en stéréovision omnidirectionnelle

Dans le suivi de régions planes en stéréovision, la région de référence est définie dans la première image de la première caméra  $c_1$ . Avant de suivre le ou les plans dans le temps, la correspondance doit être faite dans les images des autres caméras du banc stéréo au même instant. Cet appariement dense est réalisé simultanément à la reconstruction du plan 3D dans un processus de suivi.

Plus précisément, le capteur *FOO* est utilisé dans les manipulations réalisées pour la validation du suivi de plusieurs plans en stéréovision omnidirectionnelle. Nous considérons ce capteur comme un banc stéréo à quatre caméras, chacune représentée par une sphère du modèle de projection unifié. Grâce au modèle de projection unifié, l'approche de suivi et reconstruction simultanée de plan proposée ici est générique, c'est-à-dire qu'elle peut s'appliquer à tout système à plusieurs caméras centrales étalonnées et si les transformations relatives entre ces caméras sont connues. Utiliser un capteur à une seule caméra et plusieurs miroirs permet, de résoudre les problèmes d'acquisition synchronisée et d'aberration chromatique. La ligne de base du *FOO* est certes faible (3 cm), ce qui nécessite une détection et une correspondance précise pour atteindre une précision raisonnable dans l'estimation 3D, mais c'est aussi un avantage pour cette étape d'initialisation. En effet, les points de vue étant proches les uns des autres, les images sont légèrement différentes, ce qui permet d'appliquer l'approche de suivi de plan pour réaliser l'appariement dense des plans dans les quatre vues.

La pose relative dans un suivi de plan monoculaire, même avec plusieurs plans, est estimée à l'échelle près. Utiliser un banc stéréo étalonné intrinsèquement et extrinsèquement permet de fixer l'échelle et d'avoir les plans et la position relative en mètres.

La figure 4.5 montre les relations géométriques entre points correspondants dans des vues sphériques. En se basant sur ces relations, le suivi de plan stéréo se réalise en deux étapes principales :

1. la recherche des régions dans les trois vues correspondant au template de

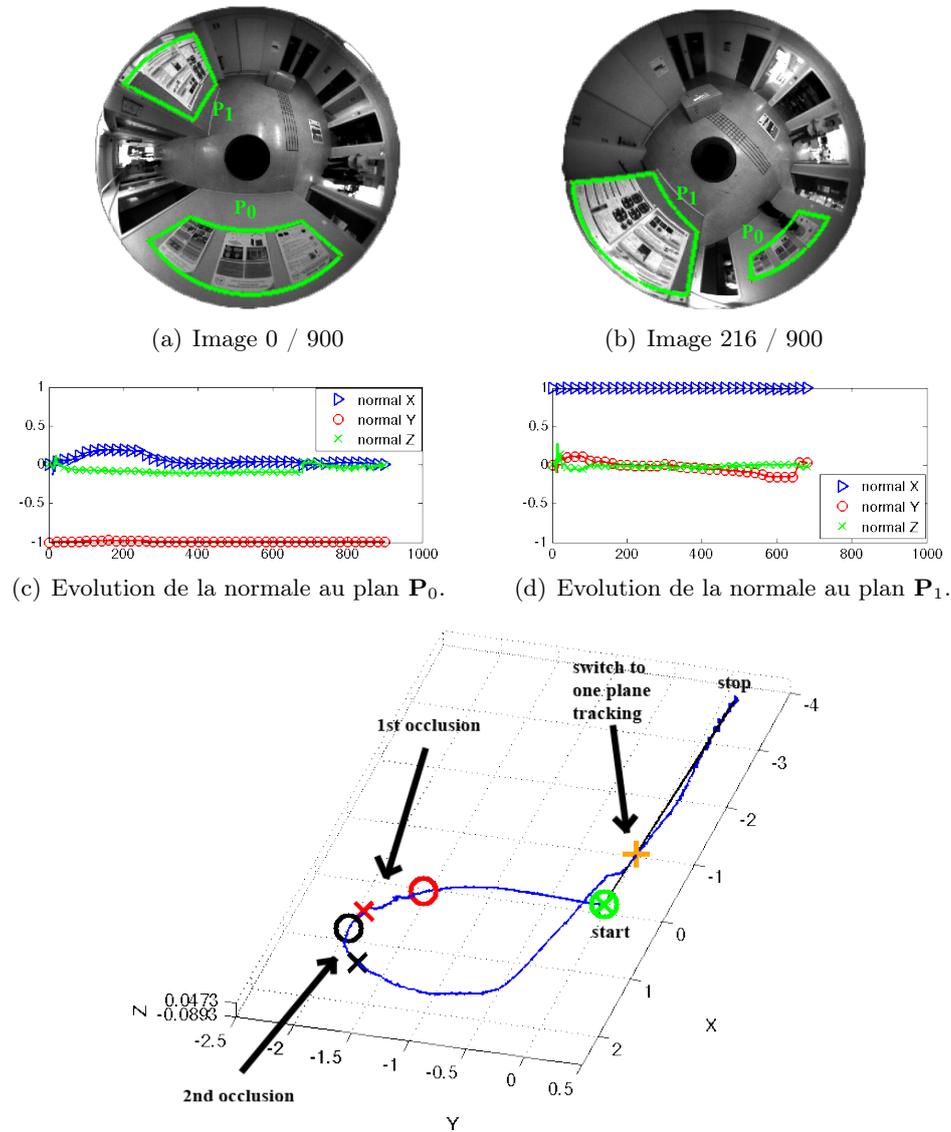


FIGURE 4.4 – Mouvement général sur 11 m en présence d’occultations partielles des plans  $P_0$  et  $P_1$  suivis simultanément dans une séquence de 900 images jusqu’à ce que  $P_1$  ne soit plus visible. Le reste de l’estimation est réalisé en suivant uniquement  $P_0$ , à partir de la croix orange (e). La trajectoire est à la bonne échelle en mètres car la distance à  $P_0$  est mesurée manuellement à partir de la position initiale. L’erreur finale d’estimation de position est sous 1 cm et l’écart type de l’estimation de la coordonnée  $Z$  est d’environ 2,64 cm sur la trajectoire complète.



Enfin, la normale et la distance sont mises à jour par addition de leur variation à leurs précédentes valeurs et les homographies  ${}^{c_j}\mathbf{H}_{c_1}$  sont recalculées après cette mise à jour en utilisant  ${}^{c_j}\mathbf{R}_{c_1}$  et  ${}^{c_j}\mathbf{t}_{c_1}$  issus de l'étalonnage, de façon similaire à l'équation (4.1).

#### 4.2.3.2 Suivi stéréo de plusieurs plans à huit vues

Après l'étape d'initialisation stéréo pour chaque plan, le suivi dans le temps est mené, en estimant la pose relative  ${}^t\mathbf{M}_0$  entre deux poses du banc stéréo. Cette pose relative est exprimée entre la position courante au temps  $t$  et la position de référence au temps 0 de la caméra de référence,  $c_1^0$  et  $c_1^t$ , du banc stéréo. Par conséquent, au temps  $t$ , les primitives image des caméras  $c_2$ ,  $c_3$ ,  $c_4$  sont exprimées par rapport à  $c_1$ . Ensuite, les informations au temps  $t$  sont liées à l'image de la caméra  $c_1$  au temps 0. Les informations des quatre caméras au temps 0 sont toujours utilisées pour contraindre les paramètres du plan 3D.

Ainsi, pour résumer, la région de l'image de la caméra  $c_1^0$ , correspondant au plan d'intérêt, forme les primitives désirées et les images de  $c_2^0$ ,  $c_3^0$ ,  $c_4^0$  sont utilisées pour ajuster  ${}^{c_1}\mathbf{n}$  et  $d$ .  $c_1^t$ ,  $c_2^t$ ,  $c_3^t$ ,  $c_4^t$  sont utilisées pour ajuster à la fois les paramètres des plans 3D et la pose relative du banc stéréo.

En considérant  $\mathbf{r}$ , une représentation vectorielle de la pose  ${}^{c_j}\mathbf{M}_{c_1^0}$ , le jacobien de la pose  $\mathbf{L}_{\mathbf{I}_S}(\mathbf{X}_S)$  liée à  $\mathbf{I}_S$  au point  $\mathbf{X}_S$  est :

$$\mathbf{L}_{\mathbf{I}_{S_j}}(\mathbf{X}_S) = -\nabla \mathbf{I}_{S_j}^T \mathbf{L}_{\mathbf{X}_{S_j}}. \quad (4.23)$$

$\mathbf{L}_{\mathbf{X}_S}$  est définie dans l'équation 2.11.

Cependant, les  $\mathbf{L}_{\mathbf{X}_S}$  des caméras 2, 3 et 4 sont transformées pour être exprimées dans le repère de  $c_1$  [Caron 2009] en utilisant la matrice de changement de repère de torseur cinématique  ${}^{c_j}\mathbf{V}_{c_1}$  entre les torseurs des caméras 1 et  $j$ . Cela mène au jacobien pour le suivi stéréo de plusieurs plans. Par souci de compacité, le jacobien de deux plans suivis par un banc stéréo de deux caméras seulement est présenté mais il est extensible à un nombre quelconque de plans et de caméras (quatre caméras et deux plans dans les résultats). Le jacobien lie les intensités d'images sphériques aux plans 3D et à la pose relative entre les positions courante et de référence du banc stéréo :

$$\mathbf{G} = \left[ \begin{array}{cc|cc} \mathbf{J}_{\mathbf{I}_{S_2^0}}^{(1)}(\mathbf{X}_S) & 0 & 0 & 0 \\ 0 & \mathbf{J}_{\mathbf{I}_{S_2^0}}^{(2)}(\mathbf{X}_S) & 0 & 0 \\ \mathbf{J}_{\mathbf{I}_{S_1^t}}^{(1)}(\mathbf{X}_S) & 0 & \mathbf{L}_{\mathbf{I}_{S_1^t}}^{(1)}(\mathbf{X}_S) & \\ \mathbf{J}_{\mathbf{I}_{S_2^t}}^{(1)}(\mathbf{X}_S) & 0 & \mathbf{L}_{\mathbf{I}_{S_2^t}}^{(1)}(\mathbf{X}_S) {}^{c_2}\mathbf{V}_{c_1} & \\ 0 & \mathbf{J}_{\mathbf{I}_{S_1^t}}^{(2)}(\mathbf{X}_S) & \mathbf{L}_{\mathbf{I}_{S_1^t}}^{(2)}(\mathbf{X}_S) & \\ 0 & \mathbf{J}_{\mathbf{I}_{S_2^t}}^{(2)}(\mathbf{X}_S) & \mathbf{L}_{\mathbf{I}_{S_2^t}}^{(2)}(\mathbf{X}_S) {}^{c_2}\mathbf{V}_{c_1} & \end{array} \right]. \quad (4.24)$$

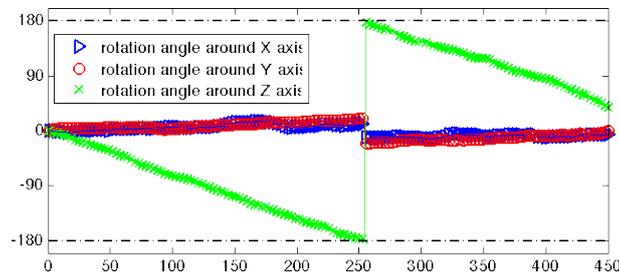
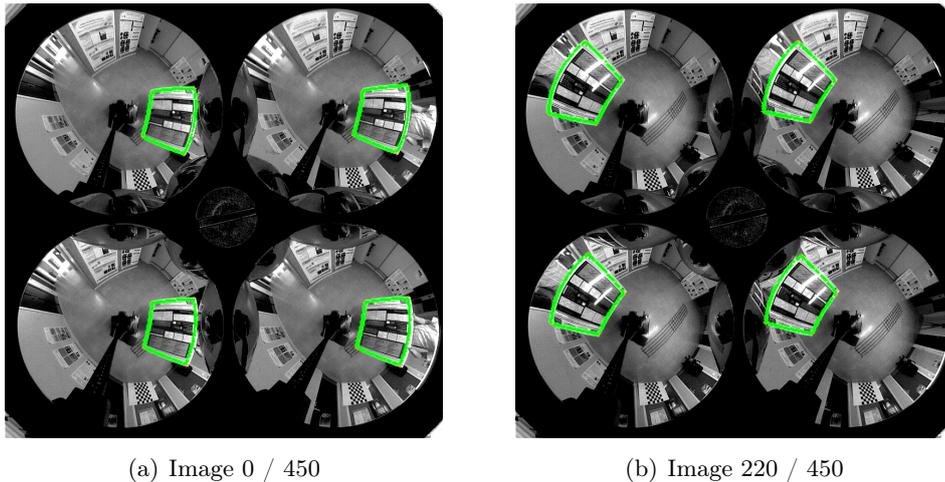
${}^{c_j}\mathbf{H}_{c_1^0}$  est l'homographie, liée au plan correspondant entre  $c_1$  au temps 0 et  $c_j$  au temps  $t$ , calculé en utilisant les estimés de la normale au plan, sa distance et la

rotation et la translation extraites de  ${}^{c_j}M_{c_1} = {}^{c_j}M_{c_1} {}^tM_0$ . Enfin, les jacobiens  $\mathbf{J}_{\mathbf{I}_{S_j^0}}$  sont recalculés à chaque itération puisque les plans 3D sont toujours mis à jour.

#### 4.2.3.3 Application : reconstruction et pose relative par suivi de plan inconnu

Que ce soit dans le cas statique ou mobile, les images sont acquises à une résolution de  $800 \times 800$  pixels, menant à un diamètre de 400 pixels pour chaque miroir quand le *FOO* est utilisé.

**Caméra omnidirectionnelle stéréo statique** Dans cette expérimentation, la caméra est statique et un objet plan est déplacé tout autour de la caméra (Figure 4.6). Le suivi réussit tout au long de la séquence de 450 images, même en présence de réflexions spéculaires sur l'objet suivi (Figure 4.6(b)) menant à des différences locales entre les templates courant et de référence (Figure 4.6(a)). L'objet ne fait pas le tour complet de la caméra, ce qui explique que l'angle de rotation final n'est pas nul (Figure 4.6(c)).



(c) Angles estimés de rotation relative par rapport à l'image initiale.

FIGURE 4.6 – Suivi stéréo d'un plan se déplaçant tout autour du capteur. L'estimation de la rotation de l'objet plan par rapport à la caméra est montré en (c).

**Caméra omnidirectionnelle stéréo mobile** Pour évaluer le comportement de l’approche proposée de suivi de plusieurs plans, le capteur *FOO* est utilisé pour retrouver le mouvement du robot (Fig. 4.7) dans une translation pure (Fig. 4.8).



FIGURE 4.7 – Le robot mobile Pioneer 3-AT avec le capteur stéréo omnidirectionnel *FOO* composé d’une caméra orthographique et de quatre miroirs. L’ensemble est placé verticalement.

L’expérimentation avec une translation pure vise à montrer l’efficacité de la méthode de suivi vis-à-vis d’un mouvement important entre la première (Figure 4.8(a)) et la dernière image (Figure 4.8(b)) de la séquence. 230 images sont acquises pour un déplacement total de 2,40 m. La trajectoire estimée est très proche de celle réalisée réellement (Figure 4.8(c)) avec une erreur finale en position, au plus loin des plans observé, d’environ 10 cm. Ce résultat est obtenu sans ajouter aucune distance mesurée manuellement ou d’autre information. La précision de l’estimation dépend, bien entendu, de la précision de l’étalonnage et de la réussite du suivi qui est efficace, puisqu’il réussit malgré des variations de taille et d’orientation des régions suivies dans les images. Le suivi montre aussi sa robustesse aux variations d’illumination, spécialement pour les spécularités. Enfin, l’estimation est stable, ce qui est illustré par la stabilité des normales aux deux plans (Figure 4.8(d)) à partir du moment où elles ont atteint leur optimum, quand la distance entre les vues courante et initiale est assez importante pour réduire les incertitudes.

### 4.3 Suivi de droites verticales avec le *FOO*

Toujours sous l’hypothèse que le robot évolue en environnement structuré, il est intéressant d’utiliser un sous ensemble des droites de l’environnement pour la localisation : les droites verticales. Ces droites sont de bons marqueurs “naturels” car elles suffisent potentiellement à localiser la caméra, et donc le robot, dans l’environnement, particulièrement lorsque le sol est horizontal. De plus, contrairement au suivi de plan où la détection automatique n’est pas évidente, la détection des droites dans les images est aisée, l’initialisation peut donc être automatique. Détecter, mettre en correspondance et reconstruire les droites verticales de la scène 3D

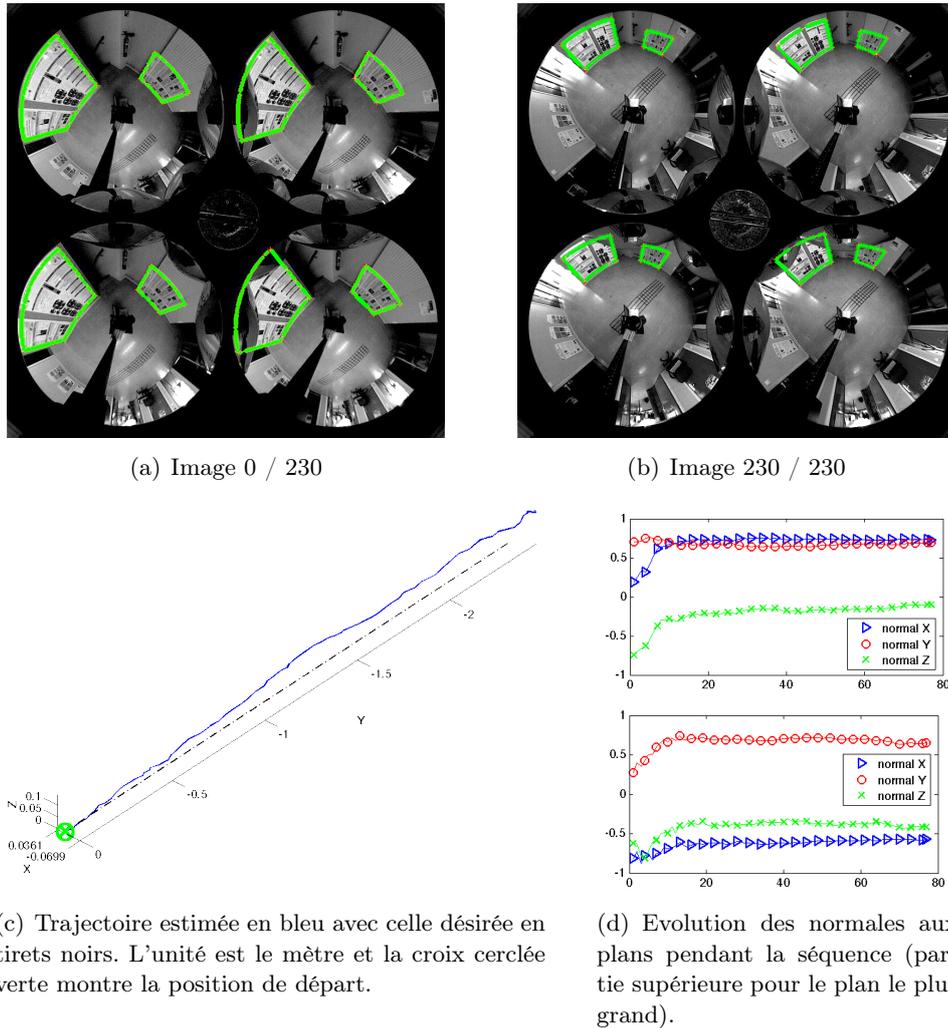


FIGURE 4.8 – Mouvement rectiligne sur 2,40 m. L'erreur finale en position est de 10 cm en 3D menant à un ratio de l'erreur sur la longueur réelle du chemin d'environ 4%. A noter que le second plan, le plus petit dans les images (a) et (b), n'est pas vertical, c'est pourquoi sa composante en  $Z$  n'est pas nulle.

permet de localiser la caméra, et donc le robot, dans l'environnement.

En plaçant le capteur stéréo *FOO* à la verticale, les droites verticales ont la propriété de se projeter en droites radiales sur les miroirs du capteur. Cette affirmation est aussi vraie pour une caméra omnidirectionnelle monoculaire mais l'avantage, ici, c'est que les quatre radiales correspondant à une même droite verticale de la scène s'intersectent en la position de cette droite dans un plan horizontal (Figure 4.9).

L'objectif, dans un premier temps, est alors de détecter et de mettre en correspondance les radiales, issues d'une même droite verticale, dans les images du *FOO*. Cette mise en correspondance permet de faire une première triangulation pour initialiser un processus d'ajustement non-linéaire photométrique.

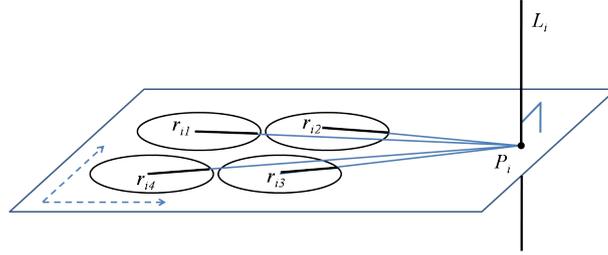


FIGURE 4.9 – Schéma de la géométrie de projection des droites verticales sur les miroirs.

### 4.3.1 Mise en correspondance et reconstruction simultanées

#### 4.3.1.1 Principe

Nous proposons un algorithme procédant par essais. L'idée est de générer une droite verticale, la projeter dans l'image du *FOO* et contrôler si ses quatre projections coïncident avec des radiales détectées. En effet, seules des radiales correspondantes sont issues d'une même droite verticale (Figure 4.9). Par conséquent, même si une verticale générée se projette au même endroit qu'une radiale détectée dans un miroir, sa projection dans les trois autres miroirs ne coïncidera avec trois radiales détectées (une dans chaque miroir) que dans le cas où elles se correspondent. En principe, deux miroirs suffisent pour faire cette vérification mais la redondance permet de confirmer la cohérence géométrique. Le but est donc de trouver la droite verticale qui engendre l'erreur angulaire minimale, entre ses projections et les détectées, dans les miroirs.

Pour ce faire, voici un critère global basé sur l'erreur angulaire. On suppose que le plan image est parallèle au sol (caméra orientée verticalement). Soit  $L_i$ , une droite verticale et  $P_i$  le point d'intersection de  $L_i$  avec le plan image et le sol, puisque ces plans sont parallèles dans la configuration présentée.  $r_{ij}$  est la radiale issue de la projection de  $L_i$  sur le miroir  $j$  et  $\theta_{ij}$  est son angle dans ce miroir (Figure 4.9). Définissons aussi  $\Delta_{ij}$ , la demi-droite partant du centre du miroir  $j$  et portant  $r_{ij}$ . Le critère global  $E$  est une somme d'erreurs angulaires où  $\theta_{ij}$  est l'angle de la radiale réelle  $r_{ij}$  et  $\theta_{kj}$ , l'angle d'une  $r_{kj}$ , projection d'une quelconque droite verticale  $L_k$  :

$$E = \sum_j |\theta_{ij} - \theta_{kj}| \quad (4.25)$$

Avec ce critère, si la droite verticale "choisie" est la bonne,  $E$  sera nulle et positive sinon. L'idée est d'étudier les variations de  $E$  par rapport à la distance au miroir source sur  $\Delta_{ij}$  afin d'exhiber une relation pour le processus d'appariement. Le tout étant de déterminer une méthode efficace de choix de droite verticale.

### 4.3.1.2 Etude des variations des erreurs de projection en fonction de la distance

Commençons par aborder un cas idéal avec une seule droite verticale  $L_i$  qui est parfaitement projetée, en une seule radiale par miroir (Figure 4.10(a)). A partir de la radiale d'un miroir, l'idée est d'échantillonner  $\Delta_{ij}$ , sur une distance allant du robot à la taille maximale de l'environnement. En effet, l'hypothèse principale est que  $L_i$  intersecte  $\Delta_{ij}$  quelque part. Cet échantillonnage donne un ensemble de positions candidates  $P_k$  de la droite verticale de l'environnement. Chaque verticale  $L_k$  passant par ces points est alors projetée sur les trois autres miroirs et autant d'erreurs globales  $E$  sont calculées qu'il y a d'échantillons. La figure 4.10(a) illustre le processus avec quelques échantillons pour plus de clarté.

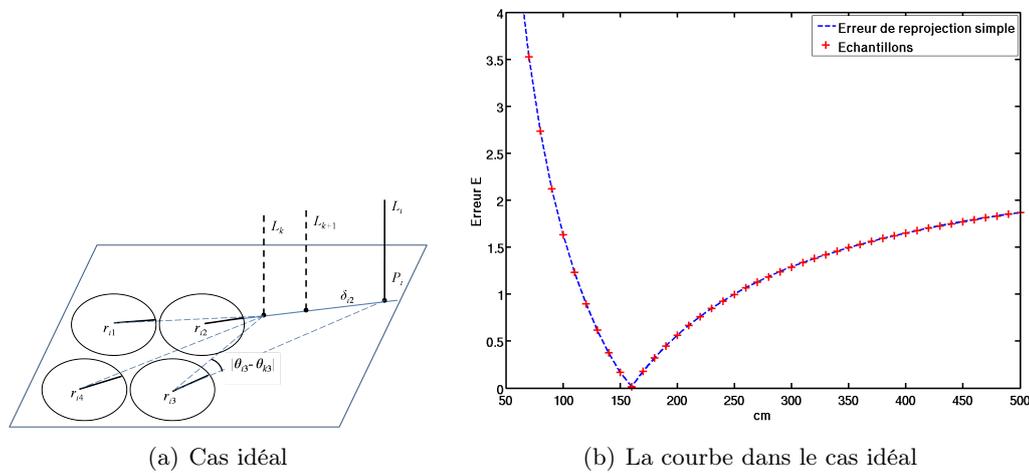


FIGURE 4.10 – Cas idéal : une radiale par miroir.

La figure 4.10(b) montre l'évolution de  $E$  en fonction de la position de  $L_k$ . Commencant proche du robot et s'en éloignant, la courbe d'erreur décroît jusqu'à un minimum et croît après. La courbe a la forme idéale pour trouver la position de l'erreur minimum. Cette position est atteinte au plus près du point réel, même si le pas d'échantillonnage est important. Mais afin d'utiliser le nombre minimal d'échantillons tout en gardant une bonne précision, la courbe autour du minimum est localement interpolée par une fonction quadratique. L'annulation de la dérivée de cette fonction apporte la distance optimale, c'est-à-dire la distance où l'erreur de reprojection est minimale. Cela signifie que reprojeter la ligne verticale passant par ce point sur chaque miroir donne les droites radiales colinéaires aux droites détectées. Pour la simulation présentée en figure 4.10(b), la distance entre le point réel  $P_i$  et le centre du capteur est de 167,70 cm et la distance optimale calculée est de 168,95 cm. Il y a donc une erreur de 0,7% due à l'échantillonnage et à l'approximation, pour un échantillonnage régulier de 10 cm.

Mais les cas réels comportent plus de radiales détectées qui peuvent être bruitées ou incomplètes. Etudions donc ces différents cas de figure pour évaluer la robustesse

de l'appariement aux problèmes de détection.

La figure 4.11(a) présente un cas plus proche de la réalité avec plusieurs radiales détectées. Un ensemble de droites verticales a été aléatoirement généré, puis projeté sur les quatre miroirs. Parmi ces verticales, on retrouve aussi la même droite  $L_i$  que dans le cas idéal afin de comparer les courbes. A cela s'ajoutent d'autres radiales, générées aléatoirement et indépendamment les unes des autres, afin de simuler des radiales "parasites". En suivant donc le même processus que celui qui a été présenté précédemment, quand on projette sur les miroirs une des  $L_k$ , venant d'un échantillon de  $\Delta_{ij}$ , on calcule l'erreur de reprojection par rapport à la radiale détectée la plus proche. La distance optimale résultante est exactement la même que dans le cas idéal, même si, dans certains cas, la courbe (Figure 4.11(b)) est légèrement différente, à cause de radiales proches. Cependant, dans tous les cas, les bonnes radiales sont appariées.

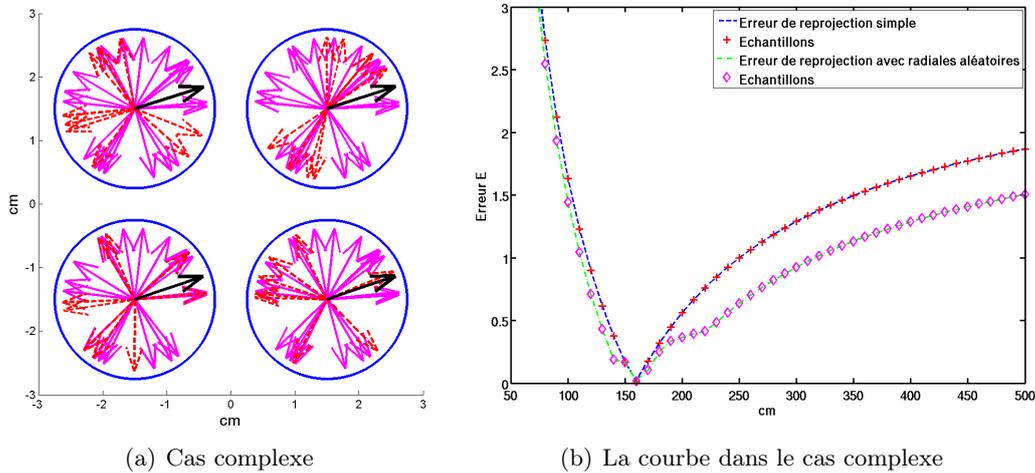


FIGURE 4.11 – Cas complexe : beaucoup de radiales.

Pour s'approcher de plus en plus de la réalité, nous avons aussi ajouté un bruit angulaire aléatoire (entre  $-0,5^\circ$  et  $+0,5^\circ$ ) aux radiales détectées, ce qui mène à un changement de la courbe (Figure 4.12). Dans ce cas, le minimum de la courbe change et il peut être plus proche ou plus loin du capteur. L'erreur de reprojection devient également un peu plus importante. Malgré cela, les bonnes radiales sont mises en correspondance.

Le dernier cas à étudier est le cas de la radiale manquante. Les radiales correspondantes peuvent être détectées dans trois miroirs ou moins et pas dans les autres. Dans ce cas, le risque est de faire de faux appariements. Si la radiale manquante est isolée, l'erreur angulaire de reprojection sera très forte (Figure 4.13(a)), ce qui permet de définir un seuil  $T$  au dessus duquel, un minimum de courbe est considéré trop élevé pour identifier une réelle position de droite verticale. Dans les autres cas, la courbe est monotone, ce qui signifie qu'il n'y a pas de droite verticale possible dans l'intervalle de recherche.

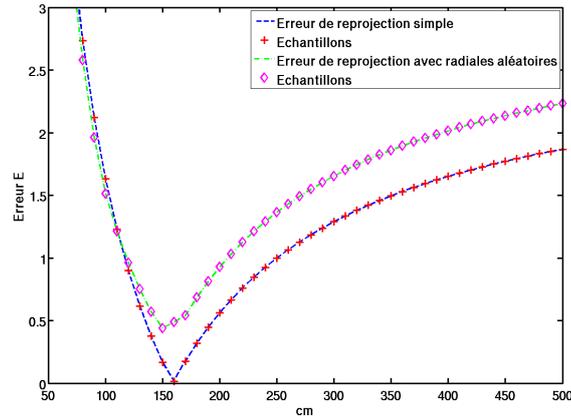
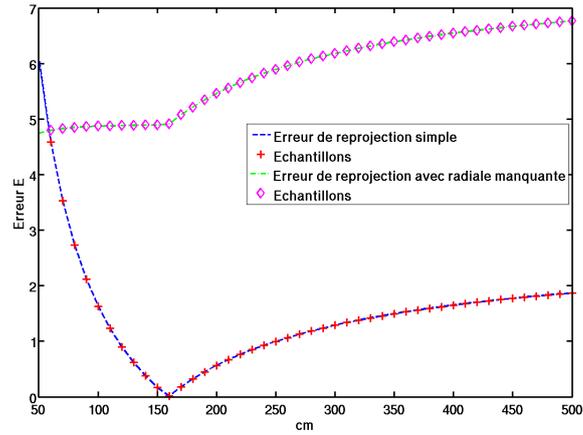


FIGURE 4.12 – La courbe dans le cas bruité.

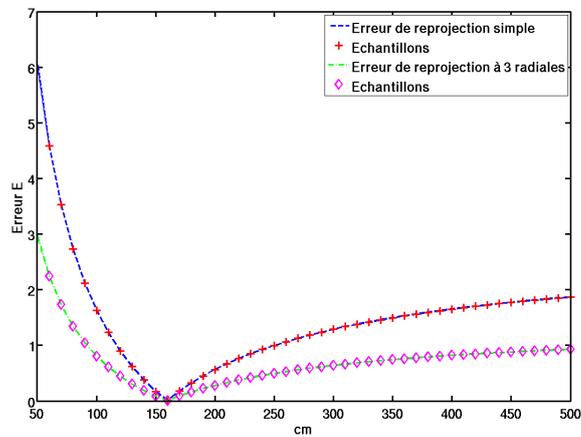
Cependant, puisque nous avons mentionné précédemment qu'utiliser quatre ou même trois droites pour trianguler est plus intéressant que deux, il est possible d'éliminer les radiales du miroir engendrant toujours la plus grande erreur. Pour traiter ce problème, après avoir trouvé le minimum de la courbe pour la distance optimale, la valeur correspondante de  $E$  est testée avec le même seuil  $T$  et si  $E$  lui est supérieure, l'algorithme s'intéresse de plus près aux erreurs angulaires individuelles de chaque  $L_k$  projetées sur les quatre miroirs. Si un miroir a toujours l'erreur angulaire maximale, il est éliminé et chaque erreur angulaire associée à ce miroir est ôtée du calcul de chaque  $E$ . Après cela, une nouvelle courbe des erreurs  $E$  en fonction des distances le long de  $\Delta_{ij}$  est construite mais en ne prenant en compte que trois miroirs. La distance optimale est alors calculée et si l'erreur associée  $E$  est inférieure à  $T$ , les trois radiales correspondantes sont appariées. Par exemple, le cas rejeté de la figure 4.13(a), après avoir été traité pour éliminer le miroir engendrant toujours l'erreur maximale, donne la courbe de la figure 4.13(b). Enfin, si après ce traitement, la courbe a encore une fois un minimum supérieur au seuil  $T$ , ou est monotone, la mise en correspondance n'est pas réalisée. Ceci est possible si plus d'une radiale  $r_{ij}$  est manquante pour une  $L_i$  donnée.

### 4.3.1.3 Triangulation

Après avoir mis en correspondance les radiales, nous voulons retrouver la position  $P_i$  de la droite verticale, c'est-à-dire l'intersection de la droite verticale et du plan image ou du sol. Mais à cause des imperfections du miroir ou des imprécisions dues à l'échantillonnage de l'image, d'importantes incertitudes peuvent apparaître sur une droite verticale loin du capteur. C'est pourquoi utiliser plusieurs droites pour trouver un point d'intersection est intéressant. Nous utilisons les quatre radiales (ou trois, dans le cas où un miroir ait été rejeté) pour trouver  $P_i$ . Par conséquent, nous avons un système surdéterminé qui est résolu par moindres carrés.



(a) La courbe dans le cas d'une radiale manquante



(b) Cas identique (mêmes radiales) mais en éliminant un miroir

FIGURE 4.13 – Le cas le plus complexe avec une radiale manquante.

#### 4.3.1.4 Extraction des droites radiales

De nombreux travaux ont été menés sur ce problème et la façon de procéder est décrite brièvement. Une transformée de Hough est d'abord appliquée pour détecter des segments de droite après une détection de contours par la méthode de Canny. Les segments sont détectés dans toute l'image et filtrés par deux critères : l'appartenance à un des quatre miroirs et l'alignement avec le centre du cercle auquel il appartient. La figure 4.14 montre un exemple de détection de radiales. Les segments sont détectés avec une précision angulaire de  $0,5^\circ$ , ce qui mène à une erreur maximale de 4,4 cm en largeur pour une droite verticale distante de 5 m, si la radiale est parfaitement détectée.

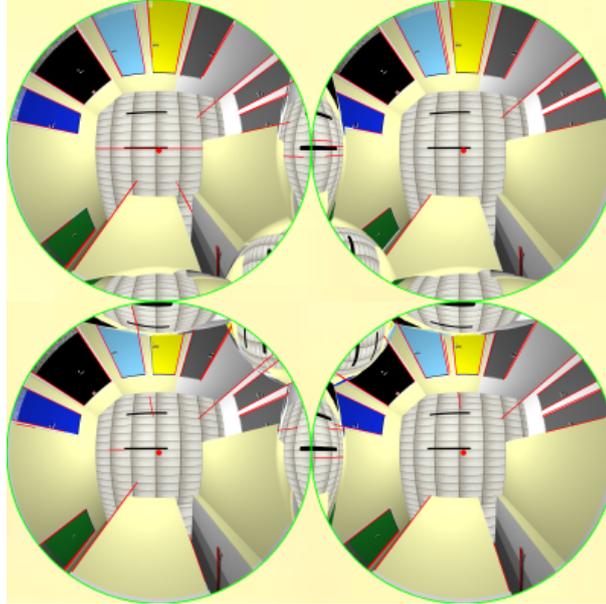


FIGURE 4.14 – Un exemple de radiales détectées sur une image de synthèse.

#### 4.3.1.5 Résultats

La méthode d'appariement a été appliquée sur des images de synthèse afin d'évaluer la robustesse au bruit, avant de travailler avec des images réelles. Notons que les images de synthèse représentent le même environnement que les images réelles puisqu'elles ont été générées avec Pov-Ray, un logiciel de rendu 3D par lancer de rayons, avec une définition précise de l'environnement réel. Dans cette section, le terme "*n*-MEC" est utilisé pour "Mise En Correspondance de *n* radiales".

**Mise en correspondance** Comme le montre la figure 4.15 un grand nombre de droites radiales peut être détecté et apparié. L'intervalle de recherche des  $L_i$  est compris entre 50 cm et 500 cm avec 45 échantillons et un seuil  $T = 0,085$  degrés.

Sur les mêmes images, nous avons appliqué différents bruits gaussiens ( $\sigma = 0,005$ ,  $\sigma = 0,01$  et  $\sigma = 0,05$ ), cinq fois chacun. Le tableau 4.1 présente les résultats d'appariement pour ces quinze images. Quand le bruit est trop important, l'algorithme, gardant les mêmes valeurs que dans le cas parfait, détecte beaucoup de fausses droites, c'est pourquoi, lorsque  $\sigma = 0,05$ , si peu d'appariements ont lieu. Si on adapte les paramètres de détection, les résultats sont identiques au cas non bruité.

Une des images réelles, comportant neuf radiales mises en correspondance, est présentée en figure 4.16(a). L'intervalle de recherche est fixé entre 50 cm et 500 cm avec 45 échantillons et un seuil à  $T = 1,5$  degrés.  $T$  était très bas avec les images de synthèse car les détections sont quasiment parfaites. Mais dans les images réelles, les distorsions, l'échantillonnage, l'illumination, la transformée de Hough, etc, peuvent produire des erreurs de détection.

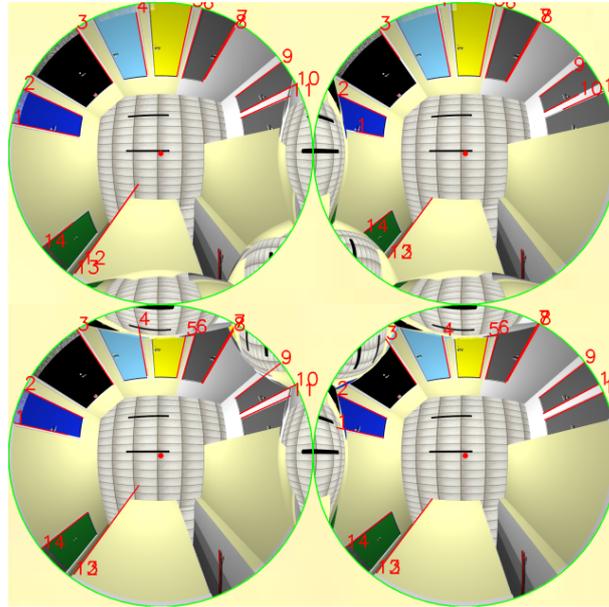
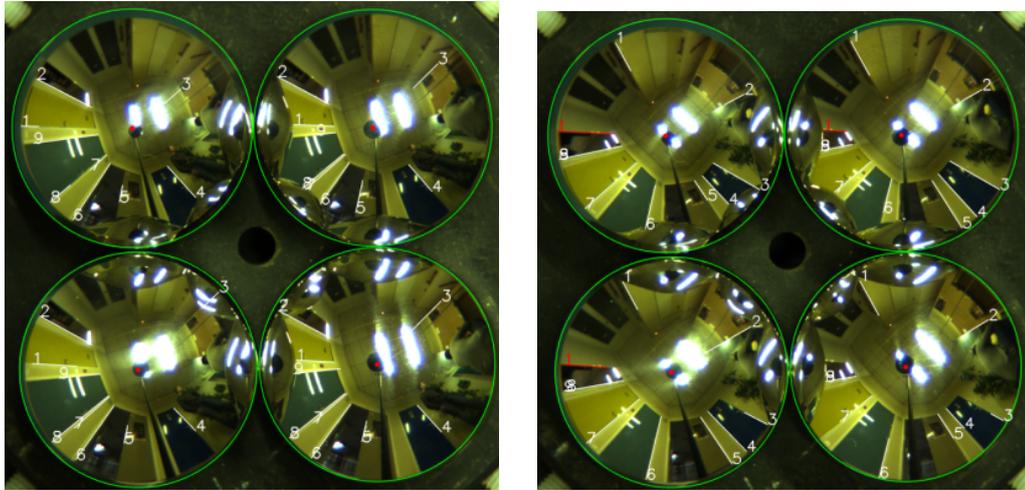


FIGURE 4.15 – Un exemple de radiales mises en correspondance sur des images de synthèse : 81,3% de seize radiales détectées et potentiellement appariables sont mises en correspondance avec 0% de faux appariements.

$\sigma$	nombre de radiales						ratios M & F
0,005	détections	15	16	13	15	15	80,3%
	4-MEC	12	10	12	13	12	
	3-MEC	0	0	0	0	0	
	fausse MEC	0	0	1	0	0	
0,01	détections	15	16	13	15	15	77,1%
	4-MEC	10	12	8	13	11	
	3-MEC	0	0	2	0	1	
	fausse MEC	0	0	0	0	0	
0,05	détections	5	4	5	3	5	42,3%
	4-MEC	3	1	1	0	2	
	3-MEC	0	0	0	2	0	
	fausse MEC	0	0	0	0	0	

TABLE 4.1 – Impact du bruit gaussien sur les images de synthèse. “ratios M & F” signifie “ratios de MEC et fausse MEC”.

Pour valider la méthode de mise en correspondance de droites radiales, vingt images ont été acquises avec le capteur *FOO* en le déplaçant d'environ 25 cm entre chaque prise de vue et en le tournant pour faire une trajectoire courbe afin de varier les points de vue autant que possible. Dans cette expérimentation, nous avons mis en correspondance, en moyenne, 77% des radiales détectées pour seulement 0,8%



(a) Un exemple de radiales appariées sur une image dans les quatre sous-images.

(b) Radiales appariées avec une 3-MEC en rouge (0, 75T).

FIGURE 4.16 – Exemples d'appariement : expérimentations en conditions réelles.

de faux appariements (un seul faux appariement dans cette séquence). Les résultats détaillés sont présentés dans le tableau 4.2.

Image	1	2	3	4	5	6	7	8	9	10	11	...	
Détections	9	9	10	10	10	13	11	11	15	13	13	...	
4-MEC	6	8	6	9	6	8	6	10	9	10	10	...	
3-MEC	0	0	0	0	1	0	0	0	0	0	0	...	
fausses MEC	0	0	0	0	0	0	0	0	0	0	0	...	
	...			12	13	14	15	16	17	18	19	20	moyenne
	...			13	10	8	12	8	10	9	11	9	10
	...			5	9	8	10	8	7	7	9	7	8
	...			1	1	0	1	0	1	0	0	0	8
	...			1	0	0	0	0	0	0	0	0	/

TABLE 4.2 – Appariement de droites par rapport à la détection. Les 3-MEC sont mentionnées et les fausses MEC aussi. Le ratio moyen des bons appariements, incluant les 3-MEC, sur les radiales détectées est de 77% pour cette séquence. Le ratio des fausses MEC, dans cette séquence, et des radiales mises en correspondance est de 0,8%.

**Triangulation** Les résultats de reconstruction à partir d'images de synthèse (Figure 4.18) montrent une estimation de position de droite verticale à l'issue de la mise en correspondance qui peut être très proche de la référence (sur la droite de la reconstruction en rouge de la figure 4.18) mais aussi assez éloignée (en bas et en

haut à gauche de la reconstruction en rouge de la figure 4.18), malgré une bonne correspondance.

La reconstruction surdéterminée par l’approche de type moindres carrés permet de corriger ces reconstructions erronées pour obtenir globalement une reconstruction plus proche de la référence (en pointillés noirs sur la figure 4.18).

En repartant de la reconstruction obtenue par moindres carrés et en la considérant “optimale”, la figure 4.19 montre que la reconstruction est robuste à des erreurs de l’ordre de 5 mm sur la position des miroirs. En effet, les étoiles noires montrent les verticales obtenues par moindres carrés à partir des positions bruitées des miroirs et reste proche de la reconstruction non bruitée. Ensuite, même si l’appariement des radiales est robuste à un bruit de  $0,5^\circ$  sur les radiales, la reconstruction obtenue par la méthode des moindres carrés devient mauvaise avec des erreurs de plus de 2 m pour certaines verticales (en rouge sur la figure 4.19). Enfin, la combinaison du bruit sur les positions des miroirs et du bruit sur les radiales donnent des résultats d’une qualité proche de la reconstruction en présence du seul bruit sur les radiales (en vert sur la figure 4.19).

Sur les images réelles, la reconstruction donne des résultats où l’erreur par rapport aux verticales de référence est assez semblable à celle obtenue avec les images de synthèse et l’application des bruits sur la position des miroirs et sur l’orientation des radiales. L’analyse des résultats se poursuit dans la partie suivante.

### 4.3.2 Suivi photométrique et reconstruction simultanés

Utiliser directement les angles obtenus à partir de la détection des radiales, que ce soit dans un algorithme, comme celui de la partie précédente, ou dans un processus d’optimisation non-linéaire, n’est efficace que dans le cas où elle est parfaite. On entend par “détection parfaite”, la détection exacte de la projection de la même droite verticale de la scène dans toutes les vues. Une détection parfaite permet d’assurer une mise en correspondance fiable et une reconstruction précise. Cependant, la détection d’une même primitive dans deux images différentes n’est que rarement exactement identique et même si la mise en correspondance est robuste à ce problème, la reconstruction de droite verticale qui s’en suivra sera fautive par rapport à la réalité, même si elle est correcte géométriquement. Ceci est principalement dû au fait que ces approches de reconstruction purement géométrique ne garantissent pas une consistance photométrique vis-à-vis des images. Cette constatation a déjà été faite pour le calcul de pose en utilisant des points [Georgel 2008] mais il est clair qu’elle est aussi valide pour la reconstruction de primitives. Dès lors, les droites verticales vont être contraintes pour que leurs radiales correspondantes aient la même apparence photométrique dans les images.

L’ajustement photométrique des radiales utilise les voisinages et consiste à déplacer la droite verticale pour que les voisinages autour de ses projections correspondent le mieux au voisinage de référence.

Le voisinage  $\mathcal{V}$  d’une radiale est défini par l’ensemble des pixels formant un angle inférieur à un seuil  $T_{\mathcal{V}}$ , définissant la largeur du voisinage, avec la radiale d’angle

$\theta^*$  :

$$\mathcal{V} = \left\{ \mathbf{u}, \left| \theta^* - \arctan \frac{v - v_0}{u - u_0} \right| \leq T_{\mathcal{V}} \text{ et } l_{inf} \leq \sqrt{v - v_0^2 + u - u_0^2} \leq l_{sup} \right\}, \quad (4.26)$$

où  $l_{inf}$  et  $l_{sup}$  sont respectivement les bornes inférieure et supérieure de la radiale, par rapport au centre  $u_0, v_0$  (Figure 4.17). Ces bornes peuvent éventuellement être respectivement fixées à 0 et au rayon du cercle englobant l'image d'un miroir pour travailler sur un secteur complet du disque formé par l'image.

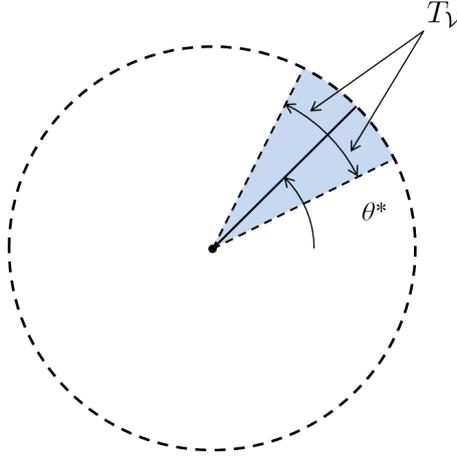


FIGURE 4.17 – Schéma illustrant le template défini par le voisinage autour d'un segment radial. L'ensemble des pixels de la zone en bleu clair forme le template associé à la radiale.

En considérant, sans perte de généralité, le miroir 1 comme miroir de référence, une radiale de ce miroir permet de définir le template de référence sur lequel les templates des projections de la verticale vont s'ajuster. En formulant ce problème sous la forme d'un problème d'optimisation, l'erreur à réguler, pour une radiale dans un miroir, est :

$$\mathbf{e} = \|\mathbf{I}(\mathcal{V}_\theta) - \mathbf{I}(\mathcal{V}^*)\|. \quad (4.27)$$

Puisque l'ajustement des patches est radial, en prenant l'hypothèse d'illumination radiale constante :

$$\mathbf{I} \left( \begin{pmatrix} \theta \\ \rho \end{pmatrix} + \begin{pmatrix} d\theta \\ 0 \end{pmatrix}, t + dt \right) = \mathbf{I} \begin{pmatrix} \theta \\ \rho \end{pmatrix}, \quad (4.28)$$

on déduit l'OFCE radiale :

$$\nabla_{radial} \mathbf{I}^T \dot{\theta} + \mathbf{I}_t = 0. \quad (4.29)$$

De la même manière que l'équation 3.1 combinée à l'équation de contrainte du flot optique 3.5 mène à l'expression de la variation de l'intensité d'un pixel en fonction de la pose de la caméra, l'équation 4.27, en la combinant à l'OFCE radiale 4.29 mène à :

$$\mathbf{I}_t = -\nabla_{radial} \mathbf{I}^T \mathbf{L}_\theta \dot{\mathbf{P}}, \quad (4.30)$$

en considérant  $\dot{\theta} = \mathbf{L}_\theta \dot{\mathbf{P}}$  et  $\mathbf{P}$ , le point d'intersection de la droite verticale avec le plan image (Figure 4.9), c'est-à-dire la position à calculer de la droite verticale.  $\mathbf{L}_\theta$  exprime la variation de l'angle  $\theta$  de la verticale projetée par rapport à la position de la verticale. Si on pose que la largeur et la hauteur des pixels sont identiques, on a :

$$\theta = \arctan \frac{c_j Y}{c_j X} \quad (4.31)$$

avec

$${}^{c_j} \mathbf{P} = \begin{pmatrix} c_j X \\ c_j Y \\ 1 \end{pmatrix} = {}^{c_j} \mathbf{M}_{c_1} {}^{c_1} \mathbf{P} \quad (4.32)$$

tel que :

$${}^{c_j} \mathbf{M}_{c_1} = \begin{pmatrix} {}^{c_j} \mathbf{R}_{c_1} & {}^{c_j} \mathbf{t}_{c_1} \\ 0 & 1 \end{pmatrix}, \text{ avec } {}^{c_j} \mathbf{t}_{c_1} \in \mathbb{R}^2 \text{ et } {}^{c_j} \mathbf{R}_{c_1} \in \mathbf{SO}(2) \quad (4.33)$$

Les matrices de changement de repère  ${}^{c_j} \mathbf{M}_{c_1}$ ,  $1 \leq j \leq 4$  sont considérées connues (obtenues par étalonnage). Ainsi,  $\mathbf{L}_\theta$  s'exprime comme suit :

$$\mathbf{L}_\theta = \begin{pmatrix} \frac{\partial \theta}{\partial X} & \frac{\partial \theta}{\partial Y} \end{pmatrix} = \begin{pmatrix} \frac{-Y}{X^2+Y^2} & \frac{X}{X^2+Y^2} \end{pmatrix}. \quad (4.34)$$

Ce jacobien image  $\mathbf{L}_\theta$  est, certes, valide pour tout point appartenant à la radiale mais aussi pour tout point du voisinage. En effet, en considérant une représentation polaire des points du voisinage avec  $u_0, v_0$  comme origine, chaque point du voisinage est défini par rapport à sa radiale avec un  $\rho$  constant et un angle  $\theta'$  défini avec un écart  $\delta_\theta$  constant par rapport au  $\theta$  de la radiale. Ainsi, un point du voisinage varie géométriquement de la même manière qu'un point de la radiale :

$$\frac{\partial \theta'}{\partial \mathbf{P}} = \frac{\partial \theta'}{\partial \theta} \frac{\partial \theta}{\partial \mathbf{P}} = \frac{\partial(\theta + \delta_\theta)}{\partial \theta} \frac{\partial \theta}{\partial \mathbf{P}} = \left( \frac{\partial \theta}{\partial \theta} + \frac{\partial \delta_\theta}{\partial \theta} \right) \frac{\partial \theta}{\partial \mathbf{P}} = (1 + 0) \frac{\partial \theta}{\partial \mathbf{P}} = \frac{\partial \theta}{\partial \mathbf{P}}. \quad (4.35)$$

Le jacobien  $\mathbf{L}_\theta$  n'est donc à calculer qu'une seule fois pour l'ensemble du patch tandis que le gradient radial  $\nabla_{radial} \mathbf{I}$  doit être calculé en tout point de ce patch. Le principe repose une nouvelle fois sur l'idée de convolution d'un filtre dérivateur avec un voisinage adapté (cf. partie 3.2.2). Avec le modèle de projection simplifié qui est utilisé, le voisinage est directement défini dans le plan image mais selon la représentation polaire des coordonnées centrées en  $u_0, v_0$ . Seul le gradient en  $\theta$  est à calculer et le voisinage "linéaire" radial  $\theta_{\mathcal{N}}$  d'un point de l'image omnidirectionnelle est défini par :

$$\theta_{\mathcal{N}} = \left\{ (\theta + k\Delta_\theta \quad \rho)^\top, -\frac{N}{2} \leq k \leq \frac{N}{2}, k \neq 0 \right\}. \quad (4.36)$$

Enfin, on déduit le jacobien image lié à l'intensité  $\mathbf{I}$  au pixel  $\mathbf{u}$  du template autour de la radiale définie par  $\theta$  :

$$\mathbf{L}_{\mathbf{I}}(\mathbf{u}) = -\nabla_{radial} \mathbf{I}^\top \mathbf{L}_\theta, \quad (4.37)$$

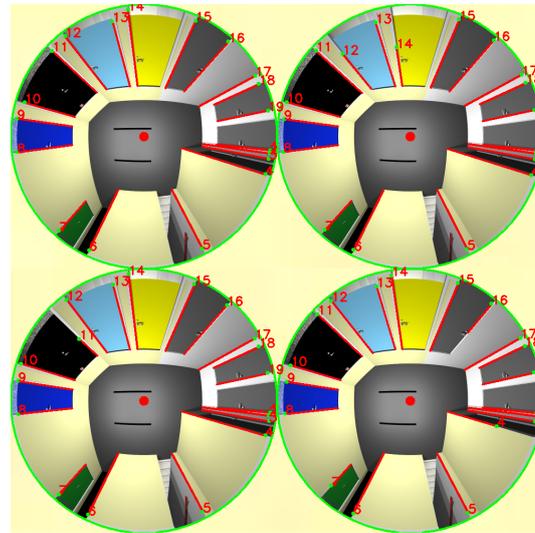
ce qui mène au calcul de la variation de la position  $\mathbf{P}$  de la droite verticale dans le plan horizontal, avec une approche de type Gauss-Newton :

$$\dot{\mathbf{P}} = -\lambda \mathbf{L}_{\mathbf{I}}(\mathbf{u})^+ (\mathbf{I}(\mathcal{V}_\theta) - \mathbf{I}(\mathcal{V}^*)) \quad (4.38)$$

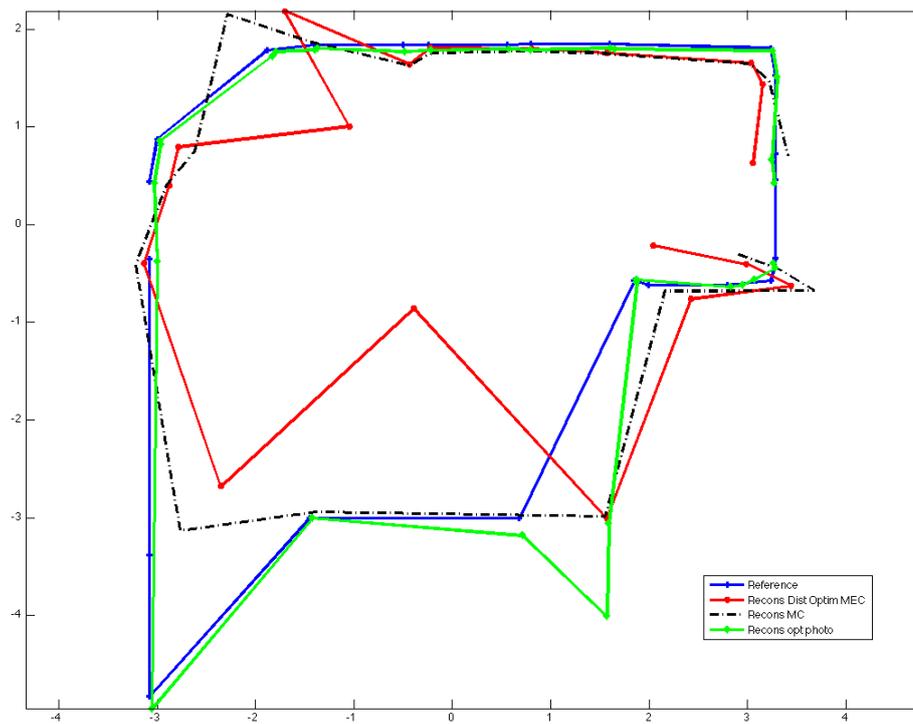
De la même manière qu’il a été évoqué dans la partie 4.2.3, l’empilement des matrices d’interactions liées à la projection de la droite verticale dans différentes vues ainsi que l’empilement des vecteurs d’erreur permettent de fusionner la redondance pour n’estimer qu’une seule position de droite verticale.

**Evaluation sur images de synthèse** Afin d’évaluer l’apport de l’ajustement photométrique des radiales, les reconstructions obtenues par plusieurs méthodes à partir d’une image de synthèse sont comparées. La figure 4.18 montre la reconstruction directement obtenue à partir de la mise en correspondance décrite dans la partie 4.3.1, la reconstruction obtenue par reconstruction surdéterminée par moindres carrés des quatre radiales mises en correspondance, et enfin la reconstruction obtenue par ajustement non-linéaire de la position de la droite verticale minimisant l’erreur photométrique des radiales. Ces reconstructions sont à comparer avec la vérité terrain et il est clair que la précision obtenue par la méthode photométrique est de loin la meilleure.

La précision de la détection des radiales influe grandement sur la précision de reconstruction des positions de droites verticales, tout comme la connaissance des positions relatives des miroirs. Dès qu’une erreur apparaît sur ces éléments, la reconstruction purement géométrique peut encore être correcte géométriquement mais n’a plus rien à voir avec la réalité. La figure 4.19 montre l’impact du bruit de ces éléments sur la reconstruction. A cause du bruit, les radiales ne sont plus réellement la projection de droites verticales réelles et les reconstructions donnent des positions aberrantes, particulièrement en profondeur (le capteur virtuel se situe au centre de l’environnement). Avec l’approche photométrique, la mise en correspondance permet uniquement d’obtenir une position initiale de la droite verticale réelle qui est ensuite optimisée pour minimiser l’erreur photométrique entre le voisinage de la radiale du miroir 1 et les voisinages des projections de la verticale dans les autres miroirs. Ainsi, là où il est impossible de trouver exactement la correspondance dans le cas purement géométrique où le lien avec l’image est perdu, l’ajustement photométrique permet de retrouver les vrais correspondances dans les quatre miroirs et d’obtenir ainsi une bonne reconstruction. Bien entendu, puisque la radiale initiale du miroir 1 a été bruitée, la reconstruction ne peut pas donner exactement la droite verticale d’origine. Une verticale “virtuelle” est en fait obtenue, sur un mur par exemple, mais pas une arête. L’éloignement de cette verticale virtuelle par rapport à la verticale d’origine dépend du bruit appliqué sur la radiale du miroir 1.



(a)



(b)

FIGURE 4.18 – Résultats de reconstruction de positions de droites verticales dans le plan horizontal. (a) L'image de synthèse utilisée ( $1000 \times 1000$  pixels) avec les radiales détectées et mises en correspondance par la méthode présentée dans la partie 4.3.1. (b) Résultats de reconstruction de l'environnement illustrant la précision apportée par l'ajustement photométrique.

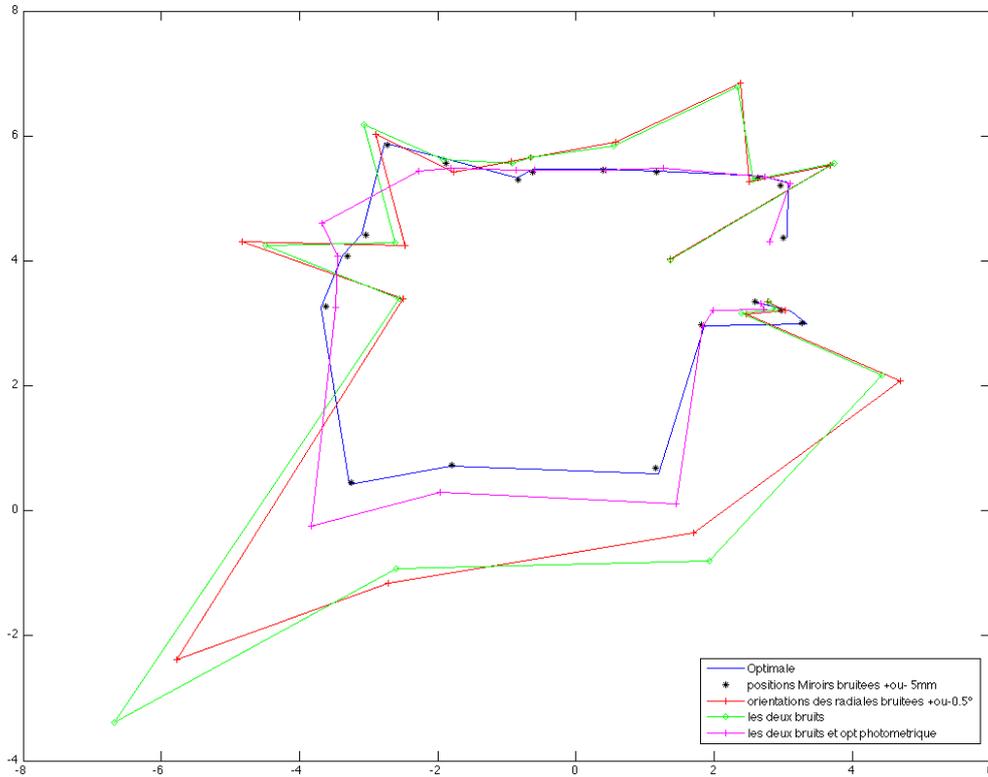


FIGURE 4.19 – Robustesse de la reconstruction en présence d’erreurs sur la connaissance des positions des miroirs entre eux et de l’orientation des radiales détectées.

**Evaluation sur images réelles** La détection et la mise en correspondance des radiales se font dans un premier temps dans l’image réelle (Figure 4.20) pour initialiser la méthode d’ajustement photométrique.

À l’issue de la mise en correspondance, les droites verticales obtenues à la distance optimale sont utilisées comme position initiale pour l’ajustement photométrique. La figure 4.21 présente la reconstruction de l’environnement par moindres carrés réalisée avec les appariements et après ajustement photométrique. Les droites de l’environnement sont à une distance moyenne du capteur FOO de 1,40 m. L’erreur moyenne de reconstruction après ajustement photométrique est de 22 cm menant à un rapport de l’erreur moyenne sur la distance moyenne des verticales de 15%. La reconstruction par moindres carrés engendre elle une erreur moyenne de plus d’1 m menant à un rapport de 76% d’erreur, principalement dû à deux reconstructions aberrantes (verticales 6 et 7, figure 4.21). En fait, à cause de la faible distance entre les miroirs, une faible erreur d’orientation des radiales peut engendrer une intersection de ces dernières dans la direction opposée à la position de la verticale réelle. En éliminant les deux reconstructions aberrantes, le rapport d’erreur moyenne est réduit à 57% mais reste conséquent.

De faibles erreurs d’orientation des radiales peuvent mener à d’importantes erreurs de reconstruction et l’écart entre les angles des radiales avant et après ajustement

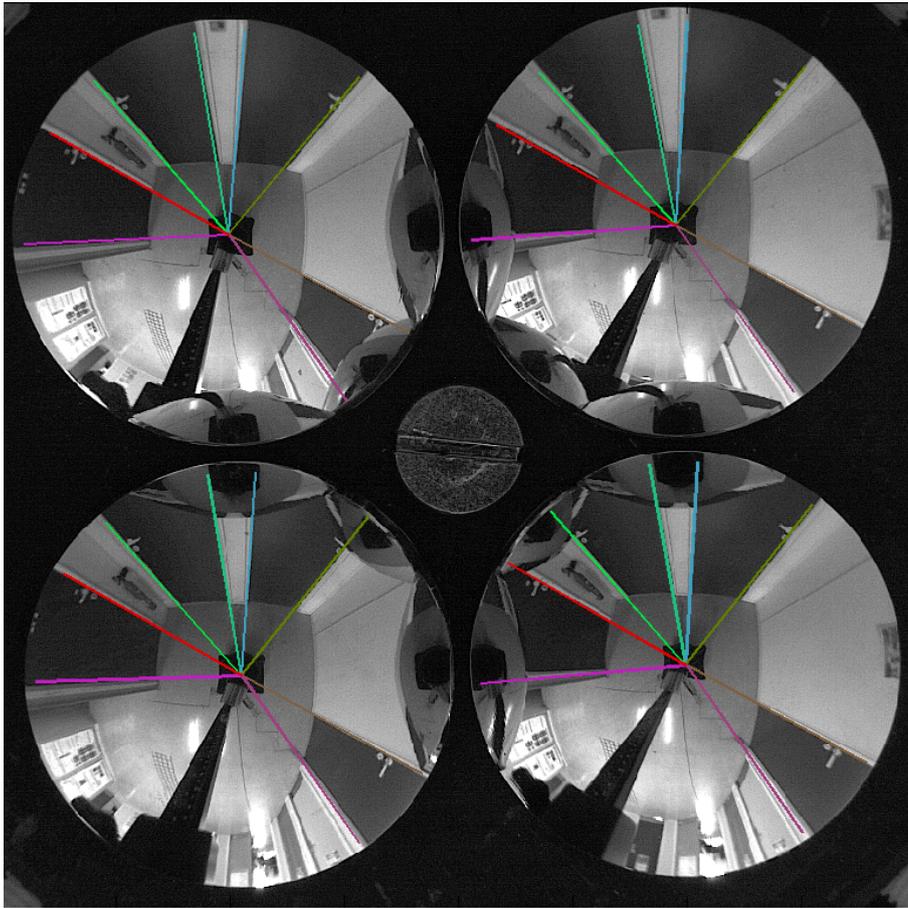


FIGURE 4.20 – Détection et mise en correspondance des radiales dans l'image du FOO.

tement photométrique est de l'ordre de  $0,50^\circ$  (écart type :  $0,73^\circ$  ; écart minimum :  $0,01^\circ$  ; écart maximum :  $3,84^\circ$ ). L'écart maximum entre une radiale avant ajustement et après de  $3,84^\circ$  peut sembler important mais rappelons que même si une radiale est détectée et appariée avec d'autres, ce n'est pas forcément une correspondance réelle : quatre radiales détectées et mises en correspondance peuvent ne pas être exactement la projection d'une même droite verticale, même si visuellement, elles en sont très proches. Ceci montre bien qu'un appariement ou une reconstruction purement géométrique donne un résultat cohérent d'un point de vue géométrique mais qui peut être faux par rapport à la réalité. La contrainte photométrique permet d'atteindre un résultat de reconstruction consistant (Figure 4.21). La reconstruction n'est cependant pas parfaite car l'imprécision de l'étalonnage n'est pas compensée. Une des perspectives de ce travail serait d'autoriser un ajustement photométrique qui ne serait pas uniquement radial.

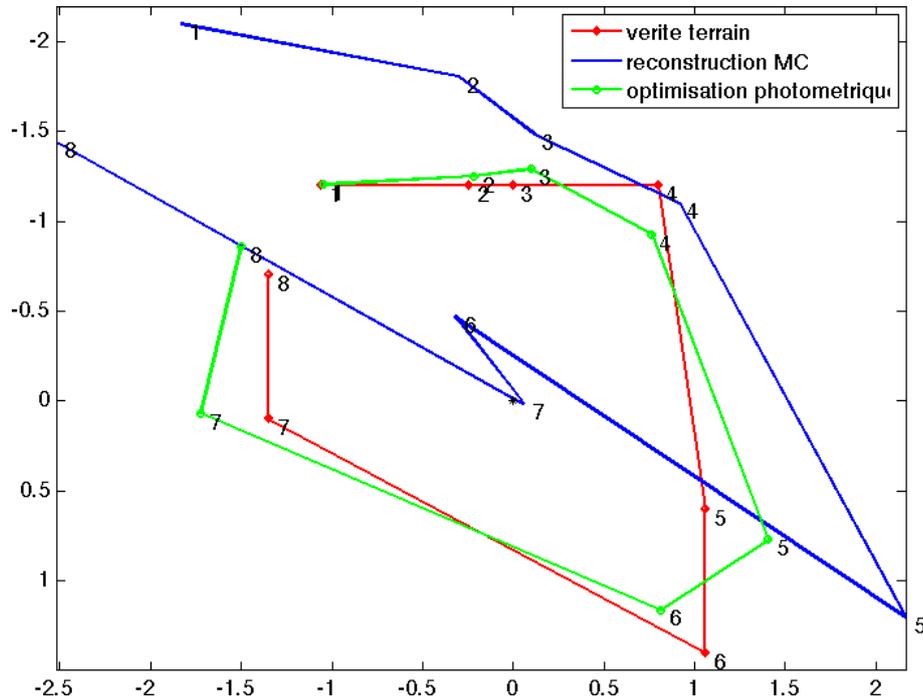


FIGURE 4.21 – Reconstructions de droites verticales par triangulation aux moindres carrés linéaires (bleu) et par ajustement photométrique non-linéaire (vert) superposées à l’environnement réel (rouge).

## 4.4 Conclusion

Le suivi de droites ou de plans se fait généralement en utilisant une primitive géométrique, comme un segment issu d’un contour pour une droite ou un ensemble de points d’intérêt inclus dans une région de l’image pour un plan. Travailler de cette manière nécessite une étape d’extraction de primitive de l’image et, si la primitive géométrique seule est prise en compte, la consistance photométrique est perdue, ce qui mène généralement à des résultats d’estimation de mouvement ou de reconstruction corrects géométriquement mais incohérents vis-à-vis de la réalité.

Dans le cadre du suivi de plan, même si le suivi d’un ensemble de points d’intérêt permet de garder localement une consistance photométrique, les travaux utilisant directement toutes les intensités d’une région de l’image correspondant à un plan dans la scène permettent de garder une consistance photométrique globale pour toute la région. En reformulant ce type d’approche sur l’image omnidirectionnelle sphérique et en l’étendant à la stéréovision omnidirectionnelle, ce chapitre a montré qu’une grande précision est atteinte en suivi et en estimation de mouvement, aussi bien avec une unique caméra, à l’échelle près, qu’avec un système stéréo, à la bonne échelle.

La prochaine étape de ce travail est sans nul doute la détection automatique des régions planaires, ce qui paraît envisageable grâce à la stéréovision.

Le suivi photométrique de plan a permis le suivi d'une primitive géométrique englobant un ensemble d'intensités pixeliques. En considérant un ensemble d'intensités pixeliques avoisinant une primitive, nous avons formulé l'ajustement de droites en stéréovision omnidirectionnelle par leur voisinage photométrique et démontré l'intérêt de ce type d'approche contre des approches purement géométriques. En effet, et cela ne vaut pas uniquement pour les droites, une détection imprécise d'une primitive peut permettre un appariement cohérent dans les images mais une reconstruction qui ne l'est pas, vis-à-vis de la scène réelle. L'ajustement de primitives par leur voisinage photométrique direct permet de retrouver la consistance photométrique perdue dans une approche totalement géométrique et ainsi obtenir une reconstruction plus proche de la réalité.

Les perspectives de ce dernier travail sont nombreuses car le suivi et la reconstruction simultanés de primitives géométriques par ajustement photométrique peuvent être formulés pour tout type de primitive géométrique (points, droites quelconques, cercles, etc) en se basant sur la partie 4.3.2, dès lors qu'il est possible d'en exprimer un voisinage et de lier la dynamique de ses intensités à celle de la primitive 3D.

Enfin, en se basant sur le suivi de plan simultané à l'estimation de mouvement de caméra omnidirectionnelle stéréo, il serait intéressant de réaliser aussi l'estimation de mouvement de caméra simultanément au suivi et à la reconstruction des droites verticales.

# Conclusion et perspectives

---

Ce mémoire de thèse aborde les questions d'estimation de position et d'orientation de caméras omnidirectionnelles conjointement au positionnement de robot guidé par de telles caméras. Le formalisme de l'asservissement visuel, virtuel ou non, permet aisément de traiter l'estimation de pose et le positionnement avec les mêmes bases théoriques. Différentes informations visuelles ont été étudiées pour réaliser ces tâches, de la plus élémentaire, l'intensité pixelique, à des informations de plus haut niveau, à savoir des points, des droites et des plans.

Au premier chapitre, nous avons vu qu'une caméra devenait omnidirectionnelle par l'utilisation d'optiques particulières dont la projection géométrique est unifiée sous la forme d'un modèle stéréographique faisant intervenir une sphère. L'étude de diverses méthodes d'estimation de pose, relative ou non, de suivi et de reconstruction a montré que la paramétrisation des transformations géométriques avait un impact sur le résultat final, tout comme le choix des primitives. En plus du choix des primitives, nous avons émis l'hypothèse que le lieu de définition de la projection d'un point 3D, d'une droite 3D, d'un plan 3D, etc, a aussi un impact sur le résultat de l'estimation. Ce type de questionnement est apparu avec l'avènement du modèle de projection stéréographique pour les caméras puisque le plan image n'est plus le seul lieu où la projection d'une primitive 3D peut être définie.

Sous le formalisme de l'asservissement visuel virtuel, au deuxième chapitre, nous avons étudié la question du lieu de représentation des primitives géométriques point et droite pour le calcul de pose. Le calcul non-linéaire de pose est influencé par le choix du lieu et les comparaisons des formulations dans le plan image, sur la sphère en représentation cartésienne et sur la sphère en représentation sphérique pure ont montré un meilleur conditionnement du problème sur la sphère en représentation cartésienne. Cela engendre une meilleure convergence de l'optimisation et une meilleure stabilité dans l'estimation de pose. A cela s'ajoute que, pour le cas des droites, la formulation sphérique ne comporte pas de singularités, contrairement à la formulation dans le plan image. Ceci permet de prendre en compte toutes les droites, quelque soit leur orientation par rapport à la caméra omnidirectionnelle, et d'avoir un système plus stable, particulièrement aux abords des cas singuliers.

Toujours au second chapitre, le calcul de pose basé points en vision omnidirectionnelle monoculaire et stéréoscopique a été généralisé à l'étalonnage de caméras et aussi étendu à l'étalonnage de bancs stéréos à caméras et modèles de projection différents. Ce travail a permis l'estimation simultanée des paramètres de projection de caméras, identiques ou non, et de leurs positions relatives. Cet étalonnage est né-

cessaire pour appliquer les méthodes de suivi, de calcul de pose et de reconstruction aux images réelles.

Alors qu'au deuxième chapitre la question de représentation et de lieu de définition des primitives était abordée d'un point de vue géométrique, le troisième chapitre a traité ces questions du point de vue photométrique. La modélisation du calcul de pose basé points du deuxième chapitre a servi de base au développement de l'asservissement visuel omnidirectionnel photométrique. Traiter directement les intensités des pixels nécessite de lier la photométrie à la géométrie et il s'avère que travailler avec l'image sphérique engendre un meilleur comportement de l'asservissement visuel que dans le plan image. En comparant les représentations sphériques pure et cartésienne, nous avons montré que le comportement de l'asservissement visuel est meilleur en représentation sphérique cartésienne pour toutes les applications réalisées.

Enfin, en fusionnant les aspects géométriques et photométriques, en représentation sphérique cartésienne, le quatrième chapitre s'est attaché à traiter le suivi de primitive géométrique par ajustement d'une région photométrique appartenant à un plan ou avoisinant une droite en stéréovision omnidirectionnelle. Le suivi de plans a été réalisé simultanément à l'estimation de mouvement du banc stéréo permettant ainsi de reconstruire les plans suivis et la trajectoire du capteur stéréo à la bonne échelle. En utilisant spécifiquement le capteur *FOO* à une seule caméra orthographique et quatre miroirs, une méthode d'appariement, de reconstruction et d'ajustement précis de droites verticales a été proposée et montre qu'incorporer directement le critère photométrique dans la reconstruction d'une primitive de la scène 3D permet d'en améliorer grandement la précision.

Les contributions majeures de cette thèse s'articulent ainsi autour de cinq points :

- L'étude de l'asservissement visuel virtuel basé points en représentation plan image, sphérique cartésienne et sphérique pure n'a pas montré de grandes différences lors d'un mouvement général avec des poses estimées quasiment identiques. Cependant, dans des conditions moins évidentes, comme lorsque la cible est assez petite dans l'image ou proche d'une singularité de représentation, la représentation sphérique cartésienne, mieux conditionnée, a montré un meilleur comportement avec une meilleure stabilité que les autres représentations.
- La formulation sphérique cartésienne du suivi et du calcul de pose basé modèle 3D filaire permet d'éviter la singularité qui apparaît dans le plan image quand une droite 3D forme un plan avec l'axe optique de la caméra. La mesure d'erreur utilisée est de plus géométrique et a donc plus de sens que la mesure algébrique utilisée dans les travaux antérieurs. La formulation sphérique permet un suivi plus robuste aussi bien en vision omnidirectionnelle monoculaire que stéréoscopique.
- L'extension de l'asservissement visuel virtuel à l'étalonnage de caméras omnidirectionnelles, de bancs stéréos et de bancs stéréos hybrides, basée sur l'étalonnage par AVV de caméra perspective, est une contribution importante de ce travail avec la réalisation d'un logiciel. L'étalonnage simultané intrinsèque

et extrinsèque permet d'atteindre une estimation plus vraisemblable des paramètres du modèle de projection qu'un étalonnage séparé.

- L'étude de l'asservissement visuel omnidirectionnel photométrique en représentation plan image, sphérique cartésienne et sphérique pure a montré des différences de comportement même pour de faibles déplacements. La représentation sphérique cartésienne sort particulièrement du lot quand le mouvement à réaliser par la caméra pour atteindre la position désirée est important. Elle est en effet, en générale la seule représentation qui permette de converger.
- Enfin, le suivi photométrique de plans et de droites verticales en stéréovision omnidirectionnelle a permis une estimation de mouvement et une reconstruction de plans simultanées à la bonne échelle, d'une part, et d'améliorer considérablement la précision de reconstruction de droites verticales d'autres part. Ces travaux, faisant un pas vers le tout simultané montrent qu'il est intéressant de continuer à explorer cette voie.

Le travail présenté dans ce document ouvre les quelques pistes de recherche suivantes :

- Intégrer l'estimation de pose à l'ajustement photométrique de radiales, qui deviendrait alors un suivi de radiales, est une première perspective intéressante. En effet, de la même manière que le suivi de plans en fait évoluer les paramètres tout au long d'une séquence, la mise à jour des positions de verticales en utilisant deux image du FOO, donc huit vues, permettrait de les raffiner à mesure que la distance entre le FOO et sa position initiale augmente, diminuant ainsi l'incertitude de reconstruction.
- Une seconde perspective sur le travail des droites radiales serait de l'étendre aux droites quelconques en représentation sphérique. Ceci permettrait de s'affranchir des hypothèses de verticalité du capteur et de l'orthogonalité de son mouvement à son axe.
- L'étalonnage est actuellement réalisé en minimisant une erreur géométrique calculée dans le plan image. Comme il a été abordé au court de ce document, le calcul d'erreur n'est pas uniforme dans le plan image omnidirectionnel. Ainsi, définir l'étalonnage à partir d'un calcul d'erreur sur la sphère du modèle stéréographique, apparaît plus cohérent et présenterait, au minimum, un meilleur conditionnement, permettant de converger vers l'optimum, même en cas d'erreur très importante sur les paramètres initiaux. La généralisation du calcul de pose basé modèle 3D filaire à l'étalonnage est aussi une piste pour rendre l'étalonnage plus robuste, toujours en utilisant la distance *point - conique* ou *point - grand cercle*.
- Enfin, le suivi de chemin visuel par asservissement photométrique gagnerait à intégrer une méthode de recherche d'image dans la base servant à définir le chemin afin de déterminer l'image sur laquelle asservir le robot lorsqu'il n'est pas placé au tout début du chemin.



# Bibliographie

- [Ameller 2002] M.-A. Ameller, L. Quan et B. Triggs. *Le calcul de pose : de nouvelles méthodes matricielles*. In Reconnaissance des Formes et Intelligence Artificielle, Angers, France, January 2002. 14
- [Andreff 2002] N. Andreff, B. Espiau et R. Horaud. *Visual Servoing from Lines*. Int. Journal of Robotics Research, vol. 21, no. 8, pages 669–699, August 2002. 60
- [Baker 1999] S. Baker et S. K. Nayar. *A Theory of Single-Viewpoint Catadioptric Image Formation*. Int. Journal on Computer Vision, vol. 35, no. 2, pages 175–196, 1999. 8
- [Baker 2001] S. Baker et I. Matthews. *Equivalence and Efficiency of Image Alignment Algorithms*. In IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR'01, pages 1090 – 1097, December 2001. 97
- [Barreto 2001] J. P. Barreto et H. Araujo. *Issues on the Geometry of Central Catadioptric Imaging*. In IEEE Int. Conf. on Computer Vision and Pattern Recognition, Hawaii, USA, December 2001. 8, 9, 63
- [Barreto 2002] J.P. Barreto, F. Martin et R. Horaud. *Visual Servoing/Tracking Using Central Catadioptric Images*. In Int. Symposium on Experimental Robotics, Advanced Robotics Series, pages 863–869, 2002. 17
- [Barreto 2003a] J.P. Barreto. *General central projection systems : modeling , calibration and visual servoing*. PhD thesis, University of Coimbra, October 2003. 17, 30
- [Barreto 2003b] J.P. Barreto et H. Araujo. *Paracatadioptric Camera Calibration Using Lines*. In Int. Conf. on Computer Vision, pages 1359–1365, 2003. 43
- [Bazin 2010] J.-C. Bazin, C. Démonceaux, P. Vasseur et I. Kweon. *Motion Estimation by Decoupling Rotation and Translation in Catadioptric Vision*. Computer Vision and Image Understanding, vol. 114, no. 2, pages 254–273, February 2010. 21
- [Benhimane 2007] S. Benhimane et E. Malis. *Homography-based 2D Visual Tracking and Servoing*. Int. Journal of Robotics Research, vol. 26, no. 7, pages 661–676, 2007. 23, 72, 97, 98
- [Bouguet 2004] J.-Y. Bouguet. *Camera Calibration Toolbox for Matlab*, 2004. 47, 48
- [Bouthemy 1989] P. Bouthemy. *A Maximum Likelihood Framework for Determining Moving Edges*. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 11, no. 5, pages 499–511, 1989. 61, 62, 63
- [Boutteau 2009] R. Boutteau, X. Savatier, J.-Y. Ertaud et B. Mazari. *An Omnidirectional Stereoscopic System for Mobile Robot Navigation*. vol. 5, pages 3–17, 2009. 9, 45

- [Boutteau 2010] R. Boutteau. *Reconstruction tridimensionnelle de l'environnement d'un robot mobile, à partir d'informations de vision omnidirectionnelle pour la préparation d'interventions*. PhD thesis, Université de Rouen, April 2010. 49, 50
- [Caron 2009] G. Caron, E. Marchand et E.M. Mouaddib. *3D Model Based Pose Estimation For Omnidirectional Stereovision*. In IEEE/RSJ International Conference on Intelligent RObots and Systems, IROS, St. Louis, Missouri, USA, October 2009. 34, 57, 108
- [Caron 2010a] G. Caron, E. Marchand et E. Mouaddib. *Omnidirectional Photometric Visual Servoing*. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, Taipei, Taiwan, October 2010. 16, 18
- [Caron 2010b] G. Caron, E. Marchand et E.M. Mouaddib. *Single Viewpoint Stereoscopic Sensor Calibration*. In International Symposium on I/V Communications and Mobile Networks, ISIVC, Rabat, Morocco, September 2010. 43
- [Chaumette 1990] F. Chaumette. *La relation vision-commande : théorie et application à des tâches robotiques*. PhD thesis, Université de Rennes 1, Mention informatique, July 1990. 17
- [Collewet 2008] C. Collewet, E. Marchand et F. Chaumette. *Visual servoing set free from image processing*. In IEEE Int. Conf. on Robotics and Automation, pages 81–86, Pasadena, California, May 2008. 16, 72, 73
- [Comport 2003] A.I. Comport, E. Marchand et F. Chaumette. *A real-time tracker for markerless augmented reality*. In ACM/IEEE Int. Symp. on Mixed and Augmented Reality, ISMAR'03, pages 36–45, Tokyo, Japan, October 2003. 18, 19, 57
- [Corke 2009] P. Corke, F. Spindler et F. Chaumette. *Combining Cartesian and cylindrical coordinates in IBVS*. In IEEE Int. Conf. on Intelligent Robots and Systems, IROS, pages 5962–5967, St Louis, USA, October 2009. 23
- [Courbon 2009] J. Courbon. *Navigation de robots mobiles par mémoire sensorielle*. PhD thesis, Université de Clermont-Ferrand, December 2009. 91
- [Daniilidis 2002] K. Daniilidis, A. Makadia et T. Bulow. *Image Processing in Catadioptric Planes : Spatiotemporal Derivatives and Optical Flow Computation*. In OMNIVIS '02 : Proceedings of the Third Workshop on Omnidirectional Vision, page 3, Washington, DC, USA, 2002. IEEE Computer Society. 74, 75
- [Deguchi 2000] K. Deguchi. *A Direct Interpretation of Dynamic Images with Camera and Object Motions for Vision Guided Robot Control*. Int. Journal on Computer Vision, vol. 37, no. 1, pages 7–20, 2000. 72
- [DeMenthon 1995] Daniel F. DeMenthon et Larry S. Davis. *Model-Based Object Pose in 25 Lines of Code*. Int. Journal of Computer Vision, vol. 15, pages 123–141, 1995. 16

- [Demonceaux 2009] C. Demonceaux et P. Vasseur. *Omnidirectional image processing using geodesic metric*. In Int. Conf. on Image Processing, pages 221–224, Cairo, Egypt, November 2009. 24, 72, 75, 98
- [Dequen 2007] G. Dequen, L. Devendeville et E. Mouaddib. *Stochastic Local Search for Omnidirectional Catadioptric Stereovision Design*. In Iberian Conf. on Pattern Recognition and Image Analysis, Girona, Spain, June 2007. 12
- [Dionnet 2007] F. Dionnet et E. Marchand. *Robust stereo tracking for space robotic applications*. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pages 3373–3378, San Diego, California, October 2007. 19, 34, 35, 57
- [Duckett 1998] T. Duckett et U. Nehmzow. *Mobile robot self-localisation and measurement of performance in middle-scale environments*. Robotics and Autonomous Systems, vol. 24, no. 1-2, pages 57–69, 1998. 43
- [Espiau 1992] B. Espiau, F. Chaumette et P. Rives. *A new approach to visual servoing in robotics*. IEEE Trans. on Robotics and Automation, vol. 8, no. 3, pages 313–326, June 1992. 17, 30, 58
- [Eynard 2010] D. Eynard, P. Vasseur, C. Demonceaux et V. Fremont. *UAV Altitude Estimation by Mixed Stereoscopic Vision*. In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, Taipei, Taiwan, October 2010. 46
- [Feddema 1989] J.T. Feddema, C.S.G. Lee et Mitchell O.R. *Automatic selection of images features for visual servoing of a robot manipulator*. In IEEE Int. Conf. on Robotics and Automation, Scottsdale, USA, May 1989. 17
- [Fischler 1981] M.A. Fischler et R.C. Bolles. *Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography*. Commun. ACM, vol. 24, no. 6, pages 381–395, 1981. 13
- [Georgel 2008] P. Georgel, S. Benhimane et N. Navab. *A Unified Approach Combining Photometric and Geometric Information for Pose Estimation*. In British Machine Vision Conf., September 2008. 120
- [Geyer 2000] C. Geyer et K. Daniilidis. *A Unifying Theory for Central Panoramic Systems and Practical Applications*. In European Conf. on Computer Vision, Dublin, Ireland, June 2000. 8
- [Geyer 2002] C. Geyer et K. Daniilidis. *Properties of the Catadioptric Fundamental Matrix*. In European Conf. on Computer Vision, Copenhagen, Denmark, May 2002. 21
- [Gluckman 1998] J. Gluckman, S.K. Nayar et K.J. Thoresz. *Real-time omnidirectional and panoramic stereo*. In DARPA Image Understanding Workshop, Monterey, USA, November 1998. 9
- [Gonçalves 2007] N. Gonçalves et H. Araújo. *Linear solution for the pose estimation of noncentral catadioptric systems*. In Int. Conf. on Computer Vision, pages 1–7, 2007. 14

- [Hadj-Abdelkader 2006] H. Hadj-Abdelkader. *Asservissement visuel en vision omnidirectionnelle*. PhD thesis, Université Blaise Pascal - Clermont II, November 2006. 17, 30
- [Hadj-Abdelkader 2010] H. Hadj-Abdelkader, Y. Mezouar et P. Martinet. Points-based visual servoing with central cameras. March 2010. 17
- [Hager 1998] G. Hager et P. Belhumeur. *Efficient Region Tracking With Parametric Models of Geometry and Illumination*. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 20, no. 10, pages 1025–1039, Octobre 1998. 97
- [Hamel 2002] T. Hamel et R. Mahony. *Visual servoing of an under-actuated dynamic rigid-body system : An image-based approach*. IEEE Trans. on Robotics and Automation, vol. 18, no. 2, pages 187–198, 2002. 32
- [Haralick 1991] R. M. Haralick, C. Lee, K. Ottenberg et Michael Nolle. *Analysis and Solutions of the Three Point Perspective Pose Estimation Problem*. Rapport technique, Hamburg, Germany, Germany, 1991. 13
- [Hartley 2003] R. Hartley et Z. Zisserman. Cambridge University Press, 2003. 20, 21, 32
- [Hmam 2010] H. Hmam et J. Kim. *Optimal non-iterative pose estimation via convex relaxation*. Image and Vision Computing, March 2010. 15
- [Horn 1987] B.K.P. Horn. *Closed-form solution of absolute orientation using unit quaternions*. Journal of Optical Society of America A, vol. 4, no. 4, pages 629–642, 1987. 14
- [Huber 1981] P.-J. Huber. Robust statistics. Wiley, New York, USA, 1981. 18, 100
- [Hutchinson 1996] S. Hutchinson, G. Hager et P. Corke. *A tutorial on visual servo control*. IEEE Trans. on Robotics and Automation, vol. 12, pages 651–670, 1996. 44
- [Jang 2005] G. Jang, S. Kim et I. Kweon. *Single Camera Catadioptric Stereo System*. In OmniVis, workshop of ICCV 2005, Beijing, China, October 2005. 11
- [Karlsson 2005] N. Karlsson, E. Di Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian et M.E. Munich. *The vSLAM algorithm for robust localization and mapping*. In Int. Conf. on Robotics and Automation, Barcelona, Spain, April 2005. 22
- [Kim 2005] J.-S. Kim, P. Gurdjos et I. Kweon. *Geometric and Algebraic Constraints of Projected Concentric Circles and Their Applications to Camera Calibration*. In IEEE Transactions on Pattern Analysis and Machine Intelligence, volume 27, pages 637–642, April 2005. 28
- [Kim 2006] Jungho Kim, Kuk jin Yoon, Jun sik Kim et Inso Kweon. *Visual SLAM by Single-Camera Catadioptric Stereo*, October 2006. 23
- [Lemaire 2007] T. Lemaire et S. Lacroix. *Monocular-vision based SLAM using line segments*. In Int. Conf. on Robotics and Automation, Roma, Italy, April 2007. 22

- [Lepetit 2009] V. Lepetit, F. Moreno-Noguer et P. Fua. *EPnP : An Accurate  $O(n)$  Solution to the PnP Problem*. Int. J. Comput. Vision, vol. 81, no. 2, pages 155–166, 2009. 15
- [Lhuillier 2008] Maxime Lhuillier. *Automatic scene structure and camera motion using a catadioptric system*. Computer Vision and Image Understanding, vol. 109, no. 2, pages 186–203, 2008. 22
- [Lowe 1991] D.G. Lowe. *Fitting Parameterized Three-Dimensional Models to Images*. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 13, pages 441–450, 1991. 16
- [Lucas 1981] B.D. Lucas et T. Kanade. *An Iterative Image Registration Technique with an Application to Stereo Vision*. In Int. Joint Conf. on Artificial Intelligence, IJCAI'81, pages 674–679, 1981. 97
- [Ma 2004] Y. Ma, S. Soatto, J. Košecá et S. Sastry. *An invitation to 3d vision*. Springer, 2004. 18, 104
- [Malis 2004] E. Malis. *Improving vision-based control using efficient second-order minimization techniques*. In IEEE Int. Conf. on Robotics and Automation, New Orleans, USA, April 2004. 15
- [Marchand 2001a] E. Marchand, P. Bouthemy et F. Chaumette. *A 2D-3D model-based approach to real-time visual tracking*. Image and Vision Computing, vol. 19, no. 13, pages 941–955, November 2001. 18
- [Marchand 2001b] E. Marchand et F. Chaumette. *A new formulation for non-linear camera calibration using virtual visual servoing*. Rapport technique 1366, IRISA, January 2001. 18, 43, 45, 47
- [Marchand 2002] E. Marchand et F. Chaumette. *Virtual visual servoing : A framework for real-time augmented reality*. Computer Graphics Forum, vol. 21(3), pages 289–298, September 2002. 16, 18
- [Marchand 2005a] E. Marchand et F. Chaumette. *Feature tracking for visual servoing purposes*. Robotics and Autonomous Systems, vol. 52, no. 1, pages 53–70, July 2005. 71
- [Marchand 2005b] E. Marchand, F. Spindler et F. Chaumette. *ViSP for visual servoing : a generic software platform with a wide class of robot control skills*. IEEE Robotics and Automation Magazine, vol. 12, no. 4, pages 40–52, December 2005. 64, 77
- [Marchand 2007] E. Marchand et F. Chaumette. *Fitting 3D models on central catadioptric images*. In IEEE Int. Conf. on Robotics and Automation, pages 52–58, Roma, Italy, April 2007. 16, 19, 57, 58, 59, 60
- [Mei 2006] C. Mei et P. Rives. *Calibrage non biaise d'un capteur central catadioptrique*. In Congrès Reconnaissance de Formes et Intelligence Artificielle, Janvier 2006. 32, 43, 47
- [Mei 2007] Christopher Mei. *Couplage Vision Omnidirectionnelle et Télémétrie Laser pour la Navigation en Robotique/Laser-Augmented Omnidirectional Vi-*

- sion for 3D Localisation and Mapping*. PhD thesis, INRIA Sophia Antipolis, Project-team ARobAS, 2007. 99
- [Mei 2008] C. Mei, S. Benhimane, E. Malis et P. Rives. *Efficient Homography-Based Tracking and 3-D Reconstruction for Single-Viewpoint Sensors*. IEEE Transactions on Robotics, vol. 24, no. 6, pages 1352–1364, Dec. 2008. 23, 24, 98
- [Merveilleux 2010] P. Merveilleux, E. Mouaddib et O. Labbani-Igbida. *Free space detection...*. In Int. Conf. on Image Processing, Hong Kong, October 2010. 92
- [Mouaddib 2005] E. Mouaddib, R. Sagawa, T. Echigo et Y. Yagi. *Stereo Vision with a Single Camera and Multiple Mirrors*. In IEEE Int. Conf. on Robotics and Automation, Barcelona, Spain, April 2005. 11, 12, 45, 51
- [Nayar 1988] S.K. Nayar. *Sphereo : Determining Depth using Two Specular Spheres and a Single Camera*. In Cambridge Symposium on Advances in Intelligent Robotics Systems, 1988. 10
- [Nayar 1996] S.K. Nayar, S.A. Nene et H. Murase. *Subspace Methods for Robot Vision*. IEEE Trans. on Robotics and Automation, vol. 12, no. 5, pages 750–758, October 1996. 72
- [Paulino 2002] A. Paulino et H. Araujo. *Pose Estimation for Central Catadioptric Systems : An Analytical Approach*. In Int. Conf. on Pattern Recognition, Washington, DC, USA, 2002. IEEE Computer Society. 14
- [Phong 1995] T.Q. Phong, R. Horaud, A. Yassine et P.D. Tao. *Object Pose from 2-D to 3-D Point and Line Correspondences*, 1995. 16
- [Quan 1999] L. Quan et Z. Lan. *Linear N-Point Camera Pose Determination*. IEEE Trans. Pattern Anal. Mach. Intell., vol. 21, no. 8, pages 774–780, 1999. 14
- [Ramalingam 2006] S. Ramalingam, S. Lodha et P. Sturm. *A Generic Structure-from-Motion Framework*. Computer Vision and Image Understanding, vol. 103, no. 3, pages 218–228, September 2006. 23
- [Salazar-Garibay 2009] A. Salazar-Garibay, E. Malis et C. Mei. *Visual Tracking of Planes with an Uncalibrated Central Catadioptric Camera*. In IEEE/RSJ Int. Conf. on Intelligent RObots and Systems, Saint-Louis, USA, October 2009. 23, 98
- [Scaramuzza 2007] D. Scaramuzza et R. Siegwart. *A Practical Toolbox for Calibrating Omnidirectional Cameras*, 2007. 43
- [Silveira 2008] G. Silveira, E. Malis et P. Rives. *Monocular-vision based SLAM using line segments*. IEEE Trans. on Robotics, vol. 24, no. 5, pages 969–979, October 2008. 22
- [Sturm 2002] P. Sturm. *Mixing Catadioptric and Perspective Cameras*. In OMNIVIS '02 : Proceedings of the Third Workshop on Omnidirectional Vision, page 37, Washington, DC, USA, 2002. IEEE Computer Society. 46

- [Sturm 2007] P. Sturm et P. Gargallo. *Conic fitting using the geometric distance*. In Asian Conf. on Computer vision, pages 784–795, Berlin, Heidelberg, 2007. Springer-Verlag. 62
- [Sundareswaran 1998] V. Sundareswaran et R. Behringer. *Visual servoing based augmented reality*. San-Francisco, USA, November 1998. 18
- [Tahri 2010] O. Tahri, Y. Mezouar, F. Chaumette et H. Araujo. Visual servoing and pose estimation with cameras obeying the unified model. March 2010. 17
- [Tatsambon Fomena 2008] R. Tatsambon Fomena. *Asservissement visuel par projection sphérique*. PhD thesis, Université de Rennes 1, Mention traitement du signal et télécommunications, November 2008. 17, 24, 28, 31, 82, 83
- [Thompson 1966] E.H. Thompson. *Space Resection : Failure Cases*. vol. 5, no. 27, pages 201–207, 1966. 13
- [Triggs 2000] B. Triggs, P. McLauchlan, R. Hartley et A. Fitzgibbon. Bundle adjustment a modern synthesis. Springer, 2000. 22
- [Wrobel 1992] B.P. Wrobel. *Minimum Solutions for Orientation*. In IEEE Workshop on Calibration and Orientation Cameras in Computer Vision, Washington D.C., USA, August 1992. 13
- [Ying 2004] X. Ying et Z. Hu. *Can We Consider Central Catadioptric Cameras and Fisheye Cameras within a Unified Imaging Model*. In European Conf. on Computer Vision, pages 442–455, Prague, Czech Republic, May 2004. 9



---

**Résumé :** Ce travail s'inscrit dans le cadre d'un programme de recherche sur la vision omnidirectionnelle, monoculaire et stéréoscopique. L'estimation de position et d'orientation de robot par ce type de vision artificielle repose sur le même formalisme de base que l'asservissement visuel. Cette technique consiste à commander le mouvement d'un robot en utilisant l'information visuelle apportée par une ou plusieurs caméras. Ce mouvement est virtuel dans le cas d'une estimation de pose.

Utiliser le large champ de vue apporté par la vision omnidirectionnelle a généralement une bonne influence sur le comportement du robot mais les informations visuelles, et en particulier leur représentation, ont un impact important sur ce comportement. Un choix inadéquat de représentation d'information peut engendrer l'échec de l'asservissement ou du calcul de pose.

Cette thèse vise à étudier et comparer différentes primitives visuelles et leurs représentations en vision omnidirectionnelle. Même si un modèle de projection stéréographique impliquant une sphère a été défini pour les caméras omnidirectionnelles centrales, un point, une droite ou un plan est presque toujours représenté dans le plan image. Quelques travaux ont certes formulé l'estimation de pose ou l'asservissement visuel à partir de primitives définies sur la sphère mais sans montrer, en pratique, quel est le meilleur lieu de définition : le plan image ou la sphère. C'est une des motivations de cette étude.

Ce travail propose aussi d'utiliser les intensités des pixels d'une image omnidirectionnelle comme primitive visuelle pour l'asservissement de robot, permettant ainsi de s'affranchir du traitement d'image nécessaire à la détection de points, de droites, etc. La validation expérimentale montre une très grande précision de positionnement. Cette constatation a servi de base à l'utilisation de voisinage photométrique inclus dans une région pour le suivi de plan ou avoisinant un segment pour le suivi de droites verticales. Ce dernier point est une des contributions les plus importantes de cette thèse car cela permet de reconstruire une droite verticale par ajustement photométrique des voisinages de ses projections dans les images de stéréovision omnidirectionnelle. Ceci peut, de plus, être étendu à l'estimation de mouvement et à toute primitive géométrique.

**Mots clés :** Vision omnidirectionnelle, stéréoscopie, estimation de pose, asservissement visuel, suivi

---

---

## Robot pose estimation and servoing using omnidirectional vision

**Abstract :** This work is about omnidirectional vision, monocular and stereoscopic. Position and orientation estimation of a robot using this kind of artificial vision relies on the same theoretical framework as visual servoing. This technique consists in the control of a robot motion using visual information brought by one or several cameras. This motion is virtual when dealing with pose estimation.

Using the wide field of view of omnidirectional vision has generally a good impact on the robot behavior but visual informations, and particularly their representation, influence deeply this behavior. A bad choice can lead the servoing or pose estimation to fail.

This thesis aims to study and compare various visual features and their representations in omnidirectional vision. Even if a projection model involving a sphere has been designed for central omnidirectional cameras, a point, a line or a plane are always represented in the image plane. Some works have expressed pose estimation or visual servoing from spherical features but without showing, practically, which surface is the best : image plane or sphere. This is a motivation for this work.

This work also proposes to use pixel intensities of an omnidirectional image as a visual feature for the servoing of a robot, allowing, hence, to withdraw image processing for the detection of points, lines, and so on. Experimental validation shows a great positioning precision. This is the base of the use of photometric neighborhoods included in a region for the precise plane tracking or neighboring a line for the precise tracking of vertical straight lines. This last contribution is one of the most important of this thesis because it allows to reconstruct a vertical line thanks to the photometric fitting of their projections neighborhoods in omnidirectional stereovision images. Furthermore, it can be extended to motion estimation and to every geometrical feature.

**Keywords :** Omnidirectional vision, stereovision, pose estimation, visual servoing, tracking

---