

# THESE DE DOCTORAT

présentée par

**Jonathan FABRIZIO**

pour obtenir le titre de  
Docteur de l'Université Pierre & Marie Curie  
Spécialité : Informatique

## **Localisation d'obstacles coopératifs par systèmes de vision classiques et panoramiques.**

A soutenir le 15 décembre 2004 devant le jury composé de :

M. Jean SEQUEIRA	Rapporteur	LXAO	LSIS
M. El Mustapha MOUADDIB	Rapporteur	CREA	UPJV
M. Danièle FOURNIER	Examineur	ESPCI	UPMC
M. Peter STURM	Examineur	MOVI	INRIA
M. Maurice MILGRAM	Examineur	LISIF	UPMC
M. Jean DEVARS	Examineur	LISIF	UPMC



# Table des matières

Table des figures	v
Liste des tableaux	ix
Introduction Générale	1
Etat de l'art	9
<b>I Etude des Caméras à Capteur Plan</b>	<b>15</b>
<b>1 Principe Général du Système</b>	<b>17</b>
1.1 Introduction . . . . .	19
1.2 Solution et choix technologiques . . . . .	20
1.3 Etude de la balise . . . . .	21
1.3.1 Eléments constitutifs de la balise . . . . .	21
1.3.2 Masquage de la balise . . . . .	22
1.3.3 Codage de l'information . . . . .	22
1.3.4 Localisation dans l'espace . . . . .	23
1.3.5 Dimensions de la balise . . . . .	25
1.4 Principe général du système . . . . .	26
1.4.1 Correction des distorsions optiques . . . . .	28
1.4.2 Binarisation et détection de contours . . . . .	34
1.4.3 Etiquetage . . . . .	34
1.4.4 Extraction des balises potentielles . . . . .	35
1.4.5 Recherche des disques et localisation dans l'espace de la balise	36
1.4.6 Décodage de l'information portée par la balise . . . . .	36
1.5 Extension de la problématique . . . . .	37

1.5.1	Extension du champ de vision . . . . .	37
1.5.2	Extension de la détection aux panneaux de signalisation . . .	38
1.6	Résultats expérimentaux . . . . .	39
1.7	Conclusion . . . . .	40
<b>2</b>	<b>Localisation Monoculaire : Le Problème P-N-P</b>	<b>43</b>
2.1	Introduction et état de l'art . . . . .	44
2.2	Solution proposée . . . . .	51
2.3	Exemples . . . . .	53
2.3.1	Problème P-3-P pour des points alignés . . . . .	53
2.3.2	Problème P-4-P pour des points coplanaires . . . . .	54
2.3.3	Problème P-5-P pour des points coplanaires . . . . .	56
2.3.4	Problème P-5-P pour des points non coplanaires . . . . .	58
2.4	Cas considéré . . . . .	60
2.4.1	Expression de la solution . . . . .	61
2.4.2	Influence de l'erreur de localisation des points sur l'image . .	62
2.5	Résultats expérimentaux . . . . .	65
2.6	Conclusion . . . . .	65
<b>II</b>	<b>Etude des Capteurs Catadioptriques Panoramiques</b>	<b>69</b>
<b>3</b>	<b>Calibrage des Capteurs Catadioptriques</b>	<b>71</b>
3.1	Introduction . . . . .	73
3.2	Présentation des capteurs catadioptriques panoramiques . . . . .	73
3.3	Etat de l'art . . . . .	75
3.3.1	Méthodes proposées par Sing Bing Kang . . . . .	76
3.3.2	Méthode de Mičušík et Pajdla . . . . .	79
3.3.3	Méthodes proposées par Christopher Geyer et Kostas Daniilidis . . . . .	79
3.3.4	Méthode de Xianghua Ying et Zhanyi Hu . . . . .	82
3.3.5	Méthode de Cyril Cauchois . . . . .	83
3.3.6	Méthode de l'équipe de Carnegie Mellon University . . . . .	84
3.3.7	Bilan . . . . .	85
3.4	Notre Calibrage . . . . .	86
3.4.1	Rappel de la méthode des deux plans . . . . .	87

3.4.2	Principe du calibrage . . . . .	89
3.4.3	Récapitulatif . . . . .	92
3.4.4	Extraction des contours du miroir . . . . .	92
3.4.5	Validation et résultats . . . . .	94
3.4.6	Amélioration du calibrage par des miroirs adaptés . . . . .	98
3.5	Conclusion . . . . .	104
<b>4</b>	<b>Le Problème P-N-P</b>	<b>105</b>
4.1	Introduction . . . . .	106
4.2	Solution du problème Perspective-N-Points . . . . .	106
4.3	Exemple . . . . .	108
4.3.1	Cas de 4 points coplanaires . . . . .	108
4.4	Cas particulier des capteurs respectant la contrainte du foyer unique	110
4.5	Résultats expérimentaux . . . . .	111
4.5.1	Résultats sur images de synthèse . . . . .	111
4.5.2	Résultats sur images réelles . . . . .	112
4.6	Conclusion . . . . .	113
<b>III</b>	<b>Stéréovision Caméra Panoramique/Caméra plan</b>	<b>115</b>
<b>5</b>	<b>Stéréovision panoramique/plan</b>	<b>117</b>
5.1	Introduction . . . . .	118
5.2	Principe de la méthode . . . . .	121
5.3	Etapes de l'algorithme . . . . .	121
5.3.1	Recherche de pixels candidats . . . . .	122
5.3.2	Sélection des pixels de la courbe épipolaire . . . . .	122
5.3.3	Appariement et localisation dans l'espace . . . . .	123
5.4	Efficacité de l'algorithme . . . . .	124
5.5	Résultats expérimentaux . . . . .	125
5.5.1	Couple panoramique/plan . . . . .	125
5.5.2	Couple panoramique/panoramique . . . . .	131
5.6	Conclusion . . . . .	132
	<b>Conclusion</b>	<b>141</b>

**Bibliographie**

**144**

# Table des figures

1	Evolution de l'insécurité routière au cours des 30 dernières années. . . . .	3
1.1	Situation routière typique que le système doit analyser. . . . .	19
1.2	Vue du réflecteur (Grossissement X100). . . . .	21
1.3	Courbes de rétro-réflexion et de rétro-diffusion du matériau. . . . .	22
1.4	Masquage de la balise. . . . .	23
1.5	Le codage porté par la balise. . . . .	24
1.6	La balise. . . . .	24
1.7	Ordonnancement des différentes étapes du processus de détection et de localisation. . . . .	27
1.8	Ordonnancement corrigé des étapes du processus de détection et de localisation. . . . .	27
1.9	Distorsions géométriques de l'image. . . . .	28
1.10	Le modèle <i>Pinhole</i> . . . . .	29
1.11	Le plan image. . . . .	30
1.12	La mire de calibrage. . . . .	33
1.13	Résultats de la localisation sur des scènes de parking. . . . .	39
1.14	Estimation des volumes englobants des véhicules. . . . .	40
2.1	Le problème P-3-P. . . . .	46
2.2	Solution du problème P-3-P en utilisant le modèle <i>weak perspective</i> . . . . .	48
2.3	Résultat de la résolution du problème perspective-N-points sur des images réelles. . . . .	66
3.1	Capteur catadioptrique panoramique. . . . .	74
3.2	Image prise par un capteur catadioptrique panoramique d'une si- tuation routière. . . . .	75

3.3	Réflexion des rayons lumineux sur un capteur catadioptrique respectant la contrainte du foyer unique. . . . .	76
3.4	Définitions des trois repères lié au capteur et à l'espace. . . . .	77
3.5	Réflexion d'un point $P$ de l'espace sur le plan image. . . . .	78
3.6	Projection de deux ensembles de lignes parallèles de l'espace en deux ensembles de cercles sur l'image. . . . .	81
3.7	Projection de 3 lignes espace en trois arcs de cercle puis extrapolation des arcs en sphère pour en tirer $f$ et le centre. . . . .	82
3.8	Mire de calibrage à deux plans qui se fixe sur le capteur SYCLOP. . . . .	84
3.9	Mire de calibrage qui se fixe sur le cône du capteur SYCLOP. . . . .	85
3.10	Calcul des deux homographies $H_1$ et $H_2$ entre le plan image et les deux plans de calibrage $P_1$ et $P_2$ . . . . .	87
3.11	Projection d'un ensemble de points de l'image sur les deux plans de calibrage. . . . .	88
3.12	Les deux plans de calibrage sur le miroir et les motifs de calibrage dans l'image. . . . .	89
3.13	Calcul des homographies entre le plan image et les plans de calibrage. . . . .	90
3.14	Projection d'un ensemble de points de l'image sur les plans de calibrage. . . . .	91
3.15	A gauche la moyenne des résultats obtenus par soustractions successives, à droite une coupe du résultat. . . . .	93
3.16	Résultat de l'extraction des ellipses. . . . .	93
3.17	Moyenne des résultats obtenus par soustractions successives d'images tirées d'une séquence routière. . . . .	94
3.18	Image de synthèse de la mire de calibrage vue par notre capteur virtuel. . . . .	95
3.19	Projection des contours des motifs avec les paramètres idéaux et en utilisant les simplifications apportées par le respect de la contrainte du foyer unique. . . . .	96
3.20	Projection des contours des motifs avec les paramètres trouvés par le calibrage : les contours des motifs sont correctement reprojétés. . . . .	97
3.21	Le miroir a subi une translation de 4 mm dans le plan x-y. Reprojection des motifs de calibrage, à gauche avec les paramètres idéaux du capteur et à droite avec les paramètres du calibrage. . . . .	98



3.22	Image d'une mire par notre capteur . . . . .	99
3.23	Les points de contrôle sont reprojétés sur le sol, à gauche avec les paramètres idéaux du capteur et en utilisant les simplifications issues du respect de la contrainte du foyer unique et à droite en utilisant les paramètres donnés par le calibrage. . . . .	100
3.24	Miroir auto-calibrant avec une aiguille. . . . .	101
3.25	Miroir auto-calibrant sans aiguille. . . . .	101
3.26	Miroir auto-calibrant sans aiguille mais avec une bonne précision. . . . .	102
4.1	Réflexion des rayons lumineux sur le miroir . . . . .	107
4.2	Projection des 4 points de l'espace sur le miroir . . . . .	108
4.3	Projection des 4 points de l'espace sur le miroir en utilisant la contrainte du foyer unique. . . . .	110
4.4	La plateforme de simulation. . . . .	112
4.5	Prises de vue différentes d'une même scène. . . . .	113
4.6	La transformation qui lie les deux prises de vue est estimée. . . . .	114
4.7	L'ensemble des points sélectionnés est repositionné dans l'espace. . . . .	114
5.1	Masques 3x1 et 1x3 pour la recherche de crêtes. . . . .	123
5.2	Masques 4x1 et 1x4 pour la recherche de crêtes. . . . .	123
5.3	Intersection du plan épipolaire avec le miroir. . . . .	130
5.4	Intersection du plan épipolaire avec le miroir. . . . .	131
5.5	Courbe épipolaire sur le plan image obtenue analytiquement. . . . .	132
5.6	Courbe épipolaire sur le plan image obtenue analytiquement. . . . .	132
5.7	Carte des scores . . . . .	133
5.8	Courbe épipolaire estimée par notre méthode. . . . .	133
5.9	La transformation entre les deux capteurs est estimée. . . . .	134
5.10	Sélection d'un point dans l'image panoramique . . . . .	135
5.11	Table des scores. . . . .	135
5.12	Courbe épipolaire obtenue sur l'image de la caméra classique. . . . .	136
5.13	Sélection d'un point sur l'image de la caméra classique. . . . .	136
5.14	Table des scores. . . . .	137
5.15	Courbe épipolaire estimée sur l'image de la caméra panoramique. . . . .	137
5.16	Sélection d'un point dans l'image panoramique. . . . .	138
5.17	Courbe épipolaire estimée dans l'image du second capteur. . . . .	138

5.18 Table des scores. . . . . 139

# Liste des tableaux

1.1	Largeur minimale d'un motif dans l'image. . . . .	26
1.2	Ouverture du champ de la caméra. . . . .	26
3.1	Comparaison des différentes méthodes de calibrage . . . . .	103



# Introduction Générale

La voiture tue. Elle tue trop et trop souvent. Elle est pourtant un instrument destiné à simplifier notre vie mais trop fréquemment, elle la détruit.

En 2003, la sécurité routière [Rou]<sup>[1]</sup> dénombre en France 90 220 accidents corporels (accidents comportant au moins une victime), avec pas moins de 115 929 blessés dont 19 207 gravement (plus de six jours d'hospitalisation) et 5 731 tués (décès survenus dans les 6 jours après l'accident) ; ce nombre s'élève à 6 058 si l'on compte ceux survenus dans les 30 jours après l'accident, comme le font la plupart des pays européens. Cela fait en tout 6,35 morts pour 100 accidents corporels.

Ce bilan humain, seul, suffit à justifier un investissement important (politique, financier...) et met en lumière la nécessité de développer des systèmes d'assistance à la conduite et des systèmes de navigation autonome pour réduire au maximum les risques liés à la circulation routière. Pourtant le bilan ne s'arrête pas au coût humain déjà trop lourd. Le coût financier des accidents de la route est aussi colossal : d'abord les coûts marchands directs : les coûts médicaux et sociaux (premiers secours, soins...), les coûts matériels (dégâts matériels aux véhicules, à la voie publique, remorquage...) et les frais généraux (police, justice, assurance...); ensuite les coûts marchands indirects (perte de productivité lié au handicap, au décès). Enfin il y a les coûts non marchands (préjudice moral, prétiun mortis - en cas de décès et prétiun doloris - en cas de blessure...). La mise à jour par un groupe de travail du Commissariat Général au Plan (CGP) de l'étude intitulée "Prix de la vie humaine, application à l'évaluation du coût économique de l'insécurité routière" (étude remise en 1992 au CGP, au ministère de l'Équipement, du Logement et des Transports par le directeur de recherche à l'École Nationale des Ponts et Chaussées) permet d'évaluer le coût de l'insécurité routière : en 2003, on estime

---

[1] La Sécurité Routière. <http://www.securiteroutiere.equipement.gouv.fr>.

qu'un "tué" coûte 1 099 042 euros, un blessé grave 164 856 euros, un blessé léger 24 179 euros. A cela s'ajoute 5 842 euros de dégâts matériels. Ce qui fait pour 2003, un coût total de l'insécurité routière estimé à 24,2 milliards d'euros !

Face à ce constat, les constructeurs réagissent, proposant une série de systèmes pour éviter la perte de contrôle du véhicule (ABS...) ou pour réduire les risques de blessure en cas d'accident (sécurité passive : ceinture de sécurité, airbag...). Ces systèmes ont permis et contribuent encore aujourd'hui au recul de l'insécurité routière mais seulement dans une certaine proportion. Ces systèmes ont leurs limites : l'airbag, par exemple, peut être dangereux dans bien des cas. S'il se déclenche alors que le conducteur a son visage proche du volant (surtout s'il porte des lunettes), il peut provoquer plusieurs blessures et brûlures. De même, si le conducteur a un objet dans la main lors du déclenchement (téléphone portable...), cet objet est projeté à 300km/h. Il peut aussi provoquer une surdité passagère due à la détonation. Pire, si l'airbag, pour une raison ou pour une autre, ne se déclenche pas au moment de l'accident, celui-ci peut devenir instable et se déclencher à tout moment, rendant dangereuse l'extraction des victimes par les services de secours. Il peut aussi exploser en cas de découpe de la carrosserie par les services de secours. Un sondage du CNSR (Conseil National de la Sécurité Routière [dLSR]<sup>[1]</sup>) montre que 86% de personnes se déclarent beaucoup ou assez intéressées par les problèmes de sécurité routière. Ce nombre monte à 92% si l'on ne considère que les personnes utilisant la voiture au moins une fois par semaine. Les constructeurs le savent bien et font de leurs systèmes de sécurité, des arguments de vente. Mais ces systèmes de sécurité passive ont aussi l'inconvénient d'entraîner chez les utilisateurs un sentiment de sécurité qui peut induire un comportement à risque.

Le gouvernement réagit aussi, par des séries de mesures. La plus récente et la plus marquante étant la mise en service des radars automatiques. Bien que ces derniers aient été vivement contestés, il a été enregistré en novembre 2003, mois de leur mise en circulation, une baisse de 18% du nombre de tués.

Le résultat de ces efforts conjoints est une baisse continue de l'insécurité routière, et ce, malgré une augmentation constante de la circulation (figure 1). L'année 2003 en particulier, enregistre par rapport à 2002, une baisse de 14,5% des accidents corporels, une baisse de 15% des blessés légers et de 20,3% des blessés graves. Le nombre de tués est en recul de 20,9%. Le tout malgré une augmentation de

---

[1] Centre National de la Sécurité Routière. <http://www.securiteroutiere.gouv.fr/cnsr>.

1,6% de la circulation. Ces baisses sont dues, en grande partie, aux récentes mesures adoptées par le gouvernement. L'apport de l'amélioration de la qualité des véhicules et de leurs systèmes de sécurité est plus difficilement quantifiable car il est plus distillé au cours du temps - mais il est au moins aussi important.

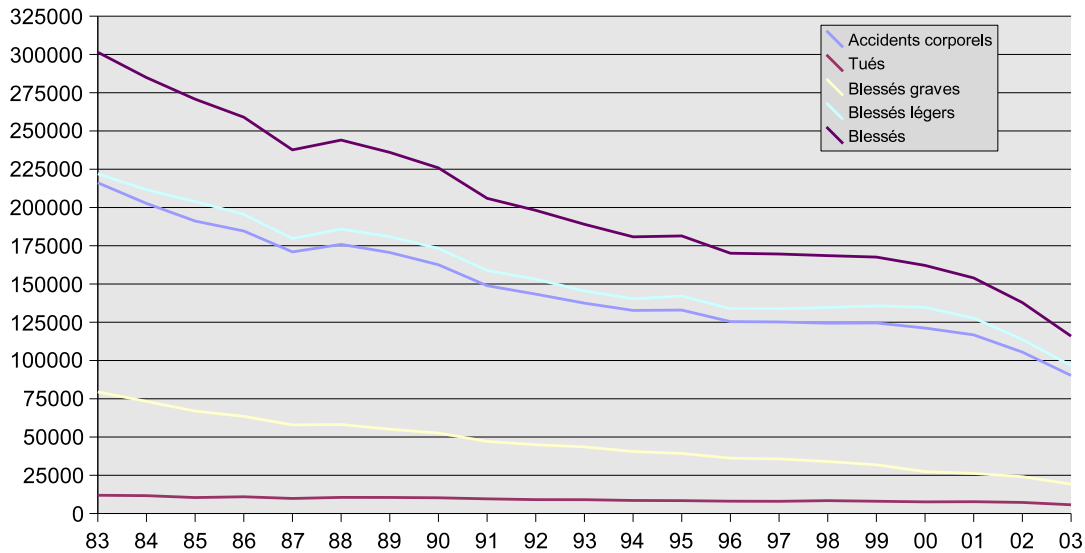


FIG. 1: Evolution de l'insécurité routière au cours des 30 dernières années (en abscisse les années, en ordonnée le nombre de personnes). L'insécurité routière en France ne cesse de diminuer, toutefois elle demeure encore trop élevée.

Malgré tous ces efforts le bilan reste inadmissible. En cause le comportement humain : l'erreur humaine est dans la grande majorité des cas la cause des accidents (surtout dans un pays où l'état des routes comme des véhicules est relativement bon). Il est donc urgent de développer des systèmes d'assistance à la conduite, venant en renfort du conducteur et pourquoi pas venant se substituer à lui.

## Cadre et Objectifs

Les travaux présentés dans cette thèse ont été réalisés au sein de l'équipe PARC (Perception, Automatique et Réseaux Connexionistes) du LISIF (Laboratoire des Instruments et des Systèmes d'Ile de France) à l'université Pierre et Marie Curie

(Paris VI).

Ils s'intègrent dans le projet national ARCOS (Action de Recherche pour une COnduite Sécurisée), projet d'une durée de 3 ans, qui s'inscrit dans le cadre des actions fédératives du PREDIT (Programme de Recherche et d'Innovation dans les Transports Terrestres). Il a comme objectif, à terme, une réduction des accidents de la route de 30%. Dans ce projet, quatre axes sont explorés :

- gérer les interdistances entre véhicules,
- prévenir les collisions sur obstacles fixes, arrêtés ou lents,
- prévenir les sorties de route,
- alerter les véhicules en amont d'incidents.

Ce projet représente un investissement de quinze millions d'euros et regroupe soixante partenaires (laboratoires et industriels).

Dans ce cadre, nos recherches portent sur l'apport d'un système de détection d'obstacles coopératifs dans un système d'aide à la conduite. Les obstacles sont dits coopératifs car, par un ajout d'informations, ils facilitent leur détection - dans notre cas, il s'agit d'informations visuelles. Ils peuvent aussi, par ce même biais, transmettre des informations. Pour ne pas gêner les utilisateurs, nous avons choisi des marquages visibles uniquement dans le proche infrarouge.

Dans un premier temps, le système doit détecter les obstacles coopératifs situés à l'avant du véhicule. Ces obstacles peuvent être fixes (poteaux...) ou mobiles (les autres véhicules).

Sur obstacle mobile, le système peut permettre :

- d'une part, de forcer le respect des distances de sécurité (En 2003, on estime que 24,5% des véhicules roulent avec un temps intervéhiculaire inférieur à 2 secondes et 5,7% avec un temps intervéhiculaire inférieur à une seconde, En circulation dense, ces nombres montent respectivement à 57,8% et 14,1%),
- d'autre part, de freiner à la place du conducteur si celui-ci ne réagit pas assez vite. (L'Observatoire National Interministériel de la Sécurité Routière dénombre en 2003, 9 628 accident corporels par collision arrière et 2 245 accidents corporels par collisions en chaîne).

Sur obstacle fixe, le système peut permettre le freinage ou simplement le signalement au conducteur d'un danger imminent (En 2003, l'ONISR dénombre 13 665 accidents corporels sur obstacles fixes). Il est évident que tous les obstacles ne



peuvent être rendus coopératifs mais une partie significative peut l'être.

Par la suite, le système est étendu à une détection tout autour du véhicule. Cette détection se fait en utilisant un capteur catadioptrique panoramique. Pour cela, une étude approfondie est réalisée sur ce type de capteurs pour pouvoir les utiliser correctement.

Nous montrons brièvement que ce système peut être utilisé pour la détection de la signalisation routière.

L'approche coopérative peut paraître irréalisable tant il semble difficile de marquer l'ensemble des véhicules du parc français. C'est en réalité possible si le coût du marquage est suffisamment bas. On peut prévoir, par exemple pour les véhicules, de rendre les plaques d'immatriculation "*coopératives*".

L'avantage majeur cette approche est la rapidité de création et la robustesse du système. Toutefois, plusieurs problèmes subsistent notamment le problème des véhicules étrangers (non marqués). Il faut donc garder à l'esprit l'idée qu'un système de détection d'obstacles coopératifs ne peut être utilisé qu'en renfort d'un système de détection plus général. Le système coopératif permettant de statuer sur les cas ambigus du système non coopératif et le système non coopératif évitant une non détection en cas d'obstacle non coopératif. Même dans un environnement contraint (une voie de circulation réservée aux véhicules dotés du système par exemple), il est toujours possible de rencontrer un obstacle non coopératif (objet tombé d'un autre véhicule...).

## Organisation du mémoire

Le mémoire commence par un rapide tour d'horizon des systèmes d'aide à la conduite par approche coopérative.

Dans le chapitre 1, nous présentons les données du problème et expliquons le principe général du système. Nous énonçons les étapes du processus de détection et de localisation d'obstacles. Certaines de ces étapes sont détaillées, notamment les étapes de correction des distorsions optiques, d'étiquetage des régions, de localisation dans l'image et de décodage des informations portées par l'obstacle. D'autres étapes sont simplement évoquées et font l'objet de chapitres ultérieurs, notamment l'étape de localisation dans l'espace. A la fin du chapitre, nous mon-

trons des résultats obtenus par ce système et nous ouvrons la problématique à la détection de la signalisation routière. Nous proposons aussi d'utiliser un capteur catadioptrique panoramique pour accroître le champ de vision.

Dans le chapitre 2, nous nous penchons particulièrement sur le problème de la localisation monoculaire en abordant le problème perspective-N-points. Après un état de l'art, une méthode systématique est proposée pour trouver une solution analytique très simple à ce problème. Elle fonctionne dans le cas de trois points alignés, et de quatre points coplanaires. Pour cinq points et plus elle fonctionne dans le cas général. Cette méthode est appliquée à plusieurs exemples : pour 3 points alignés, 4 points coplanaires et 5 points coplanaires et non coplanaires. Enfin, la solution que nous utilisons pour notre système est démontrée et illustrée par des exemples.

Dans le chapitre 3, nous introduisons les capteurs catadioptriques (particulièrement les capteurs catadioptriques panoramiques qui sont composés d'une caméra et d'un miroir de révolution leur permettant un champ de vue à  $360^\circ$ ). Ils répondent à la plupart des contraintes de perception pour la robotique mobile. C'est pour cette raison que nous avons souhaité les étudier dans le cadre de la conduite assistée. Pour pouvoir utiliser ces capteurs, il faut les calibrer et estimer un certain nombre de paramètres importants. Nous proposons donc dans ce chapitre, après un bilan des méthodes existantes, une méthode universelle (indépendante du profil du miroir) pour calibrer ces capteurs (estimation des paramètres intrinsèques et position relative du miroir face à la caméra) et nous montrons son apport à la fois sur des images de synthèse et des images réelles.

Dans le chapitre 4, nous reprenons le problème de localisation monoculaire et particulièrement le problème perspective-N-points mais en utilisant des capteurs catadioptriques. La méthode présentée précédemment, pour des caméras classiques, est étendue aux capteurs panoramiques (avec ou sans foyer de projection unique).

Dans le chapitre 5, nous montrons comment utiliser conjointement une caméra classique et une caméra panoramique pour venir en renfort du système de détection d'obstacles coopératifs. Nous posons les bases de la stéréovision utilisant des couples de capteurs non homogènes. Le problème de la détermination de la géométrie

épipolaire d'un couple de stéréovision non homogène y est abordé et une méthode pour estimer les courbes épipolaires (que les capteurs aient un foyer de projection unique ou non) est proposée. Cette méthode est vérifiée en simulation sur un cas comportant un capteur catadioptrique muni d'un miroir à profil hyperbolique et d'une caméra plan (cas où la géométrie épipolaire est démontrée analytiquement afin de vérifier le résultat obtenu par notre algorithme). Enfin cette méthode est illustrée sur des images réelles issues de couples caméra panoramique/caméra plan et aussi caméra panoramique/caméra panoramique. Dans tous les exemples, les paramètres extrinsèques de chaque capteur sont retrouvés à l'aide des solutions présentées aux chapitres 2 et 4. Toutefois le système de stéréovision n'est pas opérationnel faute d'un algorithme d'appariement robuste.

Le mémoire se termine par une conclusion passant en revue les différents problèmes abordés et analysant les possibilités envisageables pour améliorer le système.



# Etat de l'art

Relativement à l'investissement et à la quantité de travaux portant sur l'assistance à la conduite et à la navigation autonome, peu de recherches ont visé le développement de systèmes coopératifs.

Voyons des projets utilisant l'approche coopérative et des études sur les technologies envisageables pour réaliser des systèmes coopératifs efficaces.

## Le projet "Praxitèle"

Le projet Praxitèle [ADP96]<sup>[1]</sup> regroupe, entre autres, l'INRIA, l'INRETS, Renault, EDF et Dassault. Il a pour objectif de mettre à disposition d'utilisateurs des petits véhicules afin de leur permettre de réaliser des trajets courts. Une fois le trajet effectué, le véhicule doit revenir au point de départ. Un système de conduite en convoi a été étudié. Il permet à un unique chauffeur dans le premier véhicule de guider un train de véhicules. Pour ce faire, l'arrière de chaque véhicule est marqué par trois ensembles de LED. A l'aide d'une caméra, équipée d'un filtre IR polarisant (réduisant l'apport de la lumière ambiante), chaque véhicule de la chaîne localise le véhicule qui le précède.

---

[1] S. Abdou, P. Daviet, and M. Parent. Platooning for vehicles and automatic parking by scheduling robotics actions. *International Symposium on Robotics and Manufacturing, WAC*, 1996.

## Le projet “Promote Chauffeur”

Ce projet vise la création de convois automatiques de poids lourds sur autoroutes. Il regroupe plusieurs industriels.

Le système repose, entre autre, sur une technologie semblable à celle de Praxitèle. Une mire formée par des diodes infrarouge à l'arrière du camion est localisée par le poids lourd suivant à l'aide de deux caméras. L'une ou l'autre des caméras est utilisée suivant la distance au véhicule qui précède, chaque caméra ayant une focale différente.

## Le projet PATH

Des recherches sur l'aide à la conduite par approche coopérative ont été menées dans le cadre du projet PATH (Partners for Advanced Transit and Highways). Elles diffèrent des approches précédemment citées. Leur but est la localisation du véhicule par rapport à la route [TP96]<sup>[1]</sup>. Ce ne sont donc pas les obstacles qui sont coopératifs dans ce projet mais la route qui, à l'aide de plots magnétiques implantés dans le sol, facilite le positionnement du véhicule. Celui-ci est équipé de trois capteurs à l'avant. Par la différence de champs magnétiques perçus par chaque capteur, celui-ci se localise.

Des expérimentations d'asservissement sur ces plots ont été réalisées dans le cadre du NAHSC [GPTZ96]<sup>[2]</sup> (National Automated Highway System Consortium). Le codage d'informations, par ces plots magnétiques, est aussi envisagé (courbure de la route...). Mais le code étant fixé une fois pour toutes, à la pose, il est limité. Pour rendre flexible ce code, il faut que les plots portent un identifiant et que la correspondance identifiant/information soit transmise par un autre canal de communication.

Plusieurs applications sont envisagées [Int98]<sup>[3]</sup> : guidage de chasse-neige, contrôle

- 
- [1] Han-Shue Tan and Satyajit Patwardhan. Magnetic sensor for automatic steering control. *Intellimotion*, 5(3):10–11, 1996.
  - [2] Jürgen Guldner, Satyajit Patwardhan, Han-Shue Tan, and Wei-Bin Zhang. Magnetic markers on i-15 test track for nahsc demonstration. *Intellimotion*, 5(4):02–05, 1996.
  - [3] *Intellimotion*, 7(4), 1998.

latéral et longitudinal pour les poids lourds... Dans le cas des chasse-neige, deux rangées de six capteurs sont utilisées, une à l'avant et une à l'arrière. Seulement trois capteurs sont actifs en même temps. Un capteur additionnel est ajouté sur le coté, à l'avant et à l'arrière, car la conduite des chasse-neige se fait décalée par rapport à la route. Dans le cas des poids lourds, trois rangées de cinq capteurs sont utilisées : une à l'avant, une à l'arrière et une à l'arrière de la remorque.

L'application de ce système d'aide au guidage de chasse neige a été plusieurs fois testée [LYR99, ZTS<sup>+</sup>99, NE02]<sup>[4,5,6]</sup>. Cette technologie est bien adaptée pour ce type d'application car elle est robuste aux mauvaises conditions climatiques. Dans [NE02]<sup>[6]</sup>, les plots ne sont pas de simples aimants implantés dans le sol mais des éléments fabriqués par 3M qui peuvent être inclus dans le revêtement du sol ou fixés par un adhésif.

Pour accroître la précision de localisation, des tests ont été conduits pour coupler le positionnement sur route à l'aide de plots, avec l'utilisation du GPS [FR02]<sup>[7]</sup>. Enfin, au LISIF en partenariat avec l'ESPCI, un système proche est à l'étude. Deux différences majeures sont à souligner :

- les plots sont remplacés par des peintures magnétiques (incluant de la poudre de ferrite), ce qui facilite l'application sur les routes et qui en diminue le coût,
- les éléments magnétiques sont reprogrammables, il est donc possible de coder facilement des informations utiles concernant la route et de les faire évoluer en cas de besoin. Il serait de même envisageable d'utiliser le champ magnétique rémanent pour estimer la distance intervéhiculaire avec le véhicule qui précède.

---

[4] Ty A. Lasky, Kin S. Yen, and Bahram Ravani. The advanced snowplow driver assistance system. *in Proceedings of the Sixth World Congress on Intelligent Transportation System*, November 1999.

[5] Wei-Bin Zhang, Han-Shue Tan, Aaron Steinfeld, Benedicte Bougler, Dan Empey, Kun Zhou, and Masayoshi Tomizuka. Implementing advanced vehicle control and safety systems (avcss) for highway maintenance operation. *in Proceedings of the Sixth World Congress on Intelligent Transportation System*, November 1999.

[6] Marthand Nookala and Brad Estochen. Minnesota, usa intelligent vehicle initiative. *Intelligent Vehicle*, June 2002.

[7] Jay Farrell and UC Riverside. Developing magnetic and gps-aided inertial navigation for path's vehicle guidance system. *Intellimotion*, 9(4), 2002.

## Les travaux du LASMEA

Proche de ce que nous étudions, nous pouvons citer les travaux menés au LASMEA par Marmoiton [Mar00]<sup>[1]</sup>. Dans ses travaux, le véhicule à détecter et à localiser est muni de trois phares : deux feux arrières (gauche et droit) ainsi qu'un feu sur le toit. Le véhicule qui suit est muni d'une caméra (portant un filtre IR pour augmenter les contrastes) et localise les trois feux puis en déduit la position du véhicule dans l'espace.

## Technologies envisageables pour les systèmes coopératifs

Dans [MTH<sup>+</sup>99]<sup>[2]</sup>, Misener et al. proposent des éléments passifs pour améliorer les capacités des capteurs sans rendre le système dépendant d'un balisage particulier. Ils proposent d'ajouter des réflecteurs passifs pour accroître l'écho des obstacles et ainsi, faciliter leur détection. Ils font aussi le point sur les technologies possibles dans le cadre de systèmes coopératifs (LED, Peinture fluorescente, ...). Griffiths et al. étudient dans [GLM<sup>+</sup>01]<sup>[3]</sup> quatre technologies pouvant être utilisées dans le cadre de systèmes coopératifs :

- Des plaques d'immatriculation permettant l'augmentation de l'écho radar. Ce système fonctionne si le signal arrive avec un angle faible sur la plaque. Au delà de 22 °, on ne constate plus d'efficacité particulière.
- Les réflecteurs à base de coins de cubes. Ces réflecteurs ont la propriété d'être rétro-réfléchissants. On peut les utiliser pour marquer des obstacles.

---

[1] François Marmoiton. *Détection et suivi par vision monoculaire d'obstacles mobiles coopératifs à partir d'un véhicule expérimental automobile*. PhD thesis, 2000.

[2] James A. Misener, Chuck Thorpe, Ron Hearne, Lee Johnson, and Andrew c. Segal. Enhancing drivers-assist sensors: Background and concepts for sensors-friendly vehicles and roadways. *in The 6th World Congress on Intelligent Transportation Systems*, November 1999.

[3] Paul Griffiths, Dirk Langer, James A. Misener, Mel Siegel, and Chuck Thorpe. Sensor-friendly vehicle and roadway systems. *IEEE Instrumentation and Measurement Technology Conference, Budapest, Hungary*, May 2001.



- Les peintures fluorescentes pour le marquage au sol ou le marquage d'obstacles. A l'aide d'un laser, il est possible de forcer les pigments fluorescents à émettre de la lumière. Il est possible de fabriquer des peintures pour qu'elles émettent dans le proche infrarouge. Enfin, il est envisageable de moduler la source lumineuse pour faciliter la détection.
- L'utilisation de diodes modulées permet une transmission d'informations aux véhicules qui suivent. Des tests montrent que le signal est perçu jusqu'à 60 m avec un angle de vue de  $25^\circ$ .

## Bilan

L'approche coopérative est peu employée. Pourtant elle permet d'obtenir rapidement des résultats encourageants, que ce soit pour la détection d'obstacles, ou pour la détection de route. Bien que l'approche coopérative puisse être contraignante par certains cotés, elle ne doit pas être mise à l'écart.

Après cet état de l'art, voyons comment notre système fonctionne.



## Première partie

# Etude des Caméras à Capteur Plan



# Chapitre 1

## Principe Général du Système

### Sommaire

---

<b>1.1</b>	<b>Introduction</b>	<b>19</b>
<b>1.2</b>	<b>Solution et choix technologiques</b>	<b>20</b>
<b>1.3</b>	<b>Etude de la balise</b>	<b>21</b>
1.3.1	Eléments constitutifs de la balise	21
1.3.2	Masquage de la balise	22
1.3.3	Codage de l'information	22
1.3.4	Localisation dans l'espace	23
1.3.5	Dimensions de la balise	25
<b>1.4</b>	<b>Principe général du système</b>	<b>26</b>
1.4.1	Correction des distorsions optiques	28
1.4.1.1	Formation de l'image.	28
1.4.1.2	Principe général de correction	30
1.4.1.3	Remarques sur l'implémentation de l'algorithme	32
1.4.1.4	Remarques sur la mire de calibrage	33
1.4.2	Binarisation et détection de contours	34
1.4.3	Etiquetage	34
1.4.4	Extraction des balises potentielles	35
1.4.5	Recherche des disques et localisation dans l'espace de la balise	36
1.4.6	Décodage de l'information portée par la balise	36

<b>1.5</b>	<b>Extension de la problématique . . . . .</b>	<b>37</b>
1.5.1	Extension du champ de vision . . . . .	37
1.5.2	Extension de la détection aux panneaux de signalisation	38
<b>1.6</b>	<b>Résultats expérimentaux . . . . .</b>	<b>39</b>
<b>1.7</b>	<b>Conclusion . . . . .</b>	<b>40</b>

---

## 1.1 Introduction

L'objectif est la réalisation d'un système de détection et de localisation d'obstacles pour l'aide à la conduite. Nous considérons pour commencer que les obstacles se situent à l'avant. Ils peuvent être fixes, comme la signalisation d'une zone en travaux, ou mobiles, comme les véhicules (figure 1.1). Il faut les rendre coopératifs à l'aide d'un marquage, pour faciliter leur détection.

Le système doit respecter un certain nombre de contraintes :

- ne pas interférer, notamment par le marquage ajouté sur les obstacles, avec les nombreuses informations déjà présentes pour le conducteur, en particulier la signalisation routière,
- être acceptable économiquement,
- être robuste aux pannes et peu sensible aux conditions de l'environnement (en particulier aux conditions climatiques).

Dans ce chapitre, nous présenterons les choix technologiques et le principe général de fonctionnement du système. Ensuite, nous étendrons le système à la détection de la signalisation routière, et à la détection d'obstacles tout autour du véhicule (pas seulement à l'avant). Enfin nous présenterons des résultats.



FIG. 1.1: Situation routière typique que le système doit analyser : le véhicule bleu doit détecter l'arrière du véhicule violet et le véhicule violet doit détecter la signalisation en travaux (l'image a été réalisée avec POV[oVR]<sup>1</sup>)

## 1.2 Solution et choix technologiques

Nous avons choisi de munir l'obstacle d'une balise dont la forme permet à la fois, sa détection et sa localisation dans l'espace. Pour être robuste aux pannes, cette balise est passive. En effet, la partie active est entièrement portée par notre véhicule pour qu'en cas de panne, il soit possible de prendre une décision en conséquence. Si la partie active se trouve sur l'obstacle et qu'elle tombe en panne, le système de détection échouera. Pour ne pas interférer avec les autres informations et ne pas gêner les utilisateurs, nous avons choisi un marquage dans le proche infrarouge (800-900nm) non visible pour l'oeil humain. L'utilisation du proche infrarouge a l'avantage de rendre le système théoriquement moins sensible aux conditions climatiques que ne l'est un système utilisant le visible (car moins gêné par temps de pluie ou de brouillard). Le système comprend donc deux parties : la balise, portée par l'obstacle, et le module de détection situé dans le véhicule.

La balise est constituée d'un matériau rétro-réfléchissant qui renvoie la lumière d'une manière structurée dans la direction de l'émetteur. Elle est masquée visuellement par un filtre qui laisse passer le proche infrarouge. Le réflecteur que nous avons utilisé pour nos tests est un ruban adhésif, composé de micro-billes dont le fond est réfléchissant. Pour la partie détection, nous avons pensé à un codage de type code barres. Il est facilement détectable malgré les déformations perspectives et permet en plus le codage d'informations, telles que la classe d'encombrement de l'obstacle. Pour la partie localisation, l'utilisation d'une caméra unique ne permet pas de retrouver d'informations tridimensionnelles de la scène. Les obstacles étant potentiellement en mouvement, il est exclu d'utiliser notre déplacement pour faire de la stéréovision. Il faudra donc utiliser la connaissance que l'on a, a priori, de la balise, pour localiser l'obstacle. Pour cela, nous nous sommes penchés sur le problème "*perspective-N-points*" qui permet de localiser, à partir d'une image unique, N points connus, par avance, dans un repère propre. La balise est donc composée, non seulement, de barres, mais elle dispose aussi d'un certain nombre de disques.

De son côté, le véhicule qui navigue est équipé d'une caméra orientée vers l'avant, d'une unité de calcul et d'une source lumineuse dans le proche infrarouge. La caméra est une caméra ccd standard. Nous utilisons pour nos tests la caméra sony

---

<sup>1</sup>Persistence of Vision Raytracer : [www.povray.org](http://www.povray.org)



XCD-X700. Le silicium utilisé dans les ccd permet naturellement de percevoir le proche infrarouge. Cela n'entraîne donc aucun coût supplémentaire. Les phares peuvent être une source lumineuse suffisante de nuit ; dans la journée, il peut être envisagé d'ajouter des diodes infrarouges.

Nous allons maintenant étudier en détail les deux parties du système.

## 1.3 Etude de la balise

### 1.3.1 Eléments constitutifs de la balise

La balise, comme nous l'avons dit, est composée d'un matériau rétro-réfléchissant formé de micro-billes dont le fond est réfléchissant (figure 1.2). Ce matériau, acheté en grande surface, a l'avantage d'être d'un coût extrêmement bas. Il n'est pas spécialement optimisé pour le proche infrarouge mais produit un résultat satisfaisant. En effet, la figure 1.3 nous montre que dans le proche infrarouge, le signal récupéré n'est que de 10% inférieur au visible.

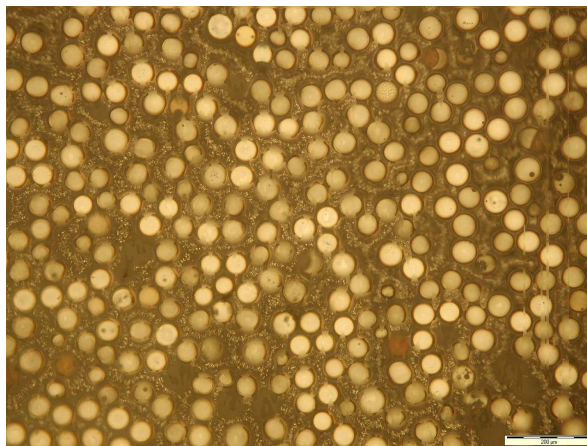


FIG. 1.2: Vue du réflecteur (Grossissement X100) : On discerne bien les multiples billes.

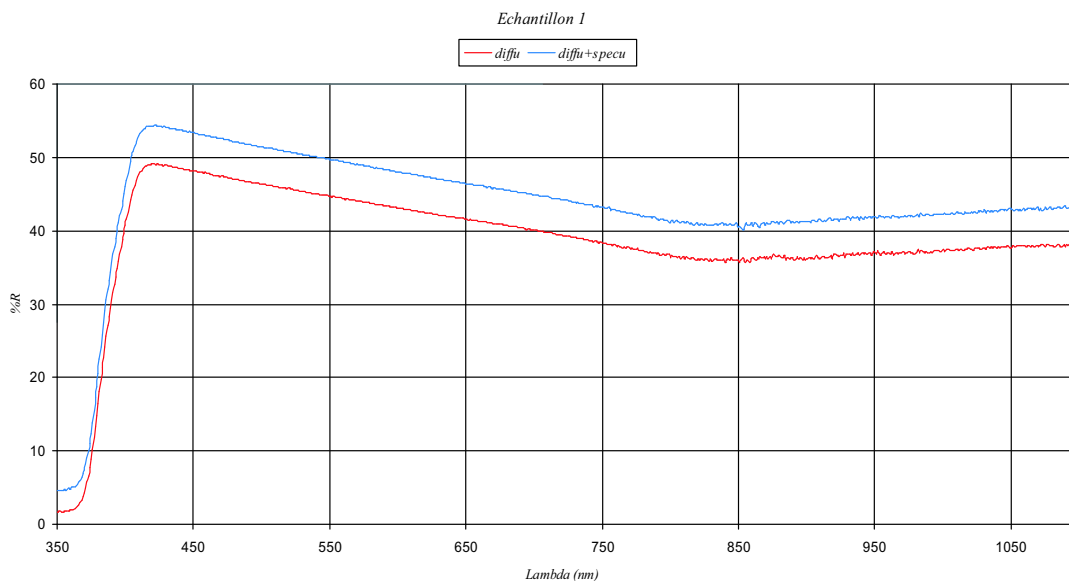


FIG. 1.3: Courbes de rétro-réflexion et de rétro-diffusion du matériau : Il y a un compromis à trouver entre la rétro-réflexion (directionnelle) et la rétro-diffusion (large champ).

### 1.3.2 Masquage de la balise

Le marquage de l'obstacle ne doit pas être visible, d'une part, pour ne pas interférer avec la signalisation routière déjà existante et gêner ainsi les utilisateurs et d'autre part pour des raisons d'esthétisme. Nous avons donc masqué la balise à l'aide de caches transparents dans l'infrarouge et non dans le visible (un simple transparent de rétro-projecteur recouvert d'encre noire a permis les essais de la figure 1.4). Cependant pour une application commerciale, l'utilisation d'un vernis coloré pourrait à la fois dissimuler et protéger la balise.

### 1.3.3 Codage de l'information

Nous nous sommes orientés vers un procédé de type code barres. Enormément de systèmes existent déjà pour extraire de l'information portée par une série de barres et ces systèmes fonctionnent malgré les transformations projectives. Nous

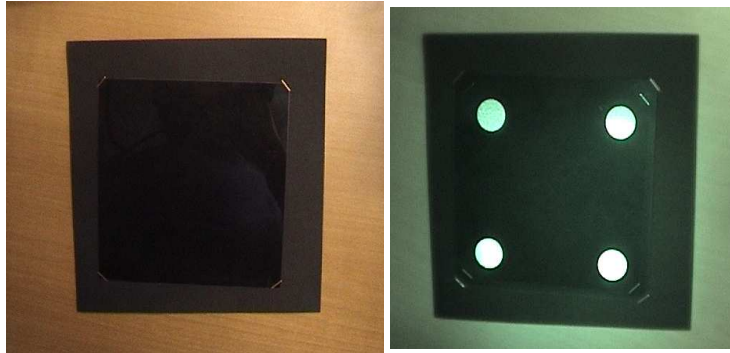


FIG. 1.4: Tentative de masquage de la balise : à gauche la balise est masquée dans le visible par un cache, à droite, les marques sont visibles dans le proche infrarouge.

avons envisagé plusieurs méthodes de codage/décodage. La première reposait sur l'utilisation du birapport. Mais les tests réalisés nous ont montré que le birapport était parfois instable, et surtout celui-ci ne fournissait pas assez de combinaisons différentes. La quantité d'informations portée par la balise était donc trop réduite. Nous avons donc utilisé un codage plus simple. Pour que la balise soit la plus compacte possible, nous utilisons les barres et les espaces inter-barres. Soit  $S$  une surface minimale de référence. Chaque barre (ou espace inter-barres) a une surface  $k S$  avec  $k \in \mathbb{N}^*$ . Pour le code, nous avons  $m$  barres et  $m$  espaces inter-barres. Ce code est encadré à chaque extrémité par une barre de calibrage de surface  $S$  (figure 1.5). La hauteur des barres étant fixe, cela revient à dire que l'on a une largeur minimale  $l$  et que chaque barre (ou espace inter-barre) a pour largeur  $k l$ . Le code étant de taille fixe, la somme de tout les coefficients  $k$  doit être égale à une constante  $L$ . Le code est porté par la succession de paramètres  $k$ . Nous avons choisi pour notre application de prendre  $L = 13$  et  $k \in \{1, 2, 3, 4\}$ , ce qui donne 728 combinaisons différentes possibles. En prenant des valeurs plus grandes, la balise devient trop large physiquement, en réduisant les valeurs, la combinatoire est trop réduite.

### 1.3.4 Localisation dans l'espace

Le second objectif de la balise est de permettre la localisation de l'obstacle dans l'espace. Pour cela nous avons dû résoudre le problème *Perspective-N-Points* qui,

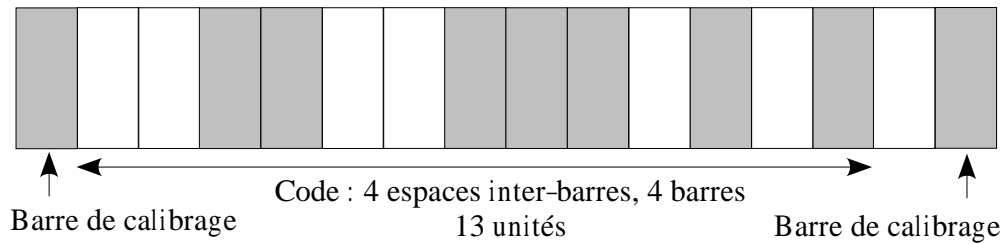


FIG. 1.5: Le codage porté par la balise : nous avons une barre de calibrage de calibrage en début et fin de code. L'information est portée par quatre espaces inter-barres et quatre barres, chacune d'une à quatre unités, pour un total de 13 unités.

à l'aide de  $N$  points de contrôle connus par avance et à l'aide d'une image unique de ces  $N$  points de contrôle, donne la position dans l'espace de ces points. Nous avons donc besoin de  $N$  points de contrôle sur notre balise pour pouvoir localiser notre obstacle. Nous avons donc muni notre balise de  $N$  disques. Le barycentre de chaque disque nous fourni un point de contrôle. La projection ne conservant pas les milieux, on pourrait se demander si l'utilisation du barycentre n'introduit pas une erreur dans nos calculs mais nous avons pu montrer que le déplacement du projeté du centre du disque était négligeable (voir plus loin les équations 1.24 et 1.25). Nous avons choisi de prendre  $N = 4$  car l'expression de la solution dans ce cas là est plus simple que dans le cas général. Nous verrons au chapitre 2 comment solutionner le problème *Perspective- $N$ -Points* dans le cas général et nous donnerons l'expression de la solution dans le cas qui nous intéresse ici.

Notre balise est donc composée de plusieurs barres verticales encadrées par 4 disques (figure 1.6).

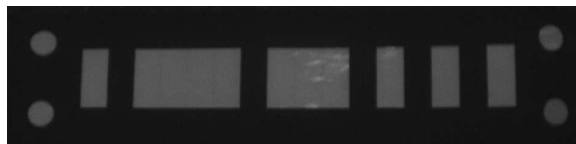


FIG. 1.6: La balise. Elle est composée par une série de barres encadrée par quatre disques.

### 1.3.5 Dimensions de la balise

Le choix des dimensions de la balise est problématique, en particulier la largeur (la hauteur ne pose pas de problèmes). Idéalement, la taille de la balise devrait être de l'ordre de grandeur de la taille d'une plaque d'immatriculation (d'après le journal officiel du 17/07/96, la zone utile d'une plaque arrière sur une seule ligne est de 510mmx100mm +/-2mm). Nous avons choisi à la section 1.3.3 un code sur 13 unités. A ces barres il faut ajouter les deux barres et l'espace qui encadre le code, ce qui ajoute 3 unités (sans compter les disques utilisés pour la localisation). Il faut donc choisir la largeur minimale d'une barre, pour que la balise rentre dans la zone utile de la plaque d'immatriculation. Si la taille minimale d'une barre fait 2 cm, la largeur totale du code mesurera 32 cm, si la taille minimale d'une barre fait 3 cm, la largeur totale du code mesurera 48 cm, enfin si la taille minimale d'une barre fait 4 cm, la largeur totale du code mesurera 64 cm. On voit qu'en prenant une largeur minimale de 3 cm, le code s'inscrit bien dans la partie utile de la plaque, laissant assez d'espace pour les disques.

En regardant le tableau 1.1 on peut voir que l'on peut lire la balise jusqu'à presque 30 m (en dessous de deux pixels, il est difficile d'analyser le motif), (ce qui équivaut environ à une distance parcourue en 2 secondes à 50 Km/h). Il faudrait donc augmenter la focale mais le tableau 1.2 nous indique qu'une focale de 10mm ne perçoit à 2 m que 1,3 m de large. C'est donc impossible. Il existe plusieurs solutions, on peut :

- augmenter la résolution de la caméra, mais cela entraîne une augmentation du coût,
- augmenter la focale, mais on l'a vu, le champ de vision est alors trop réduit,
- augmenter la taille de la balise, ce qui peut être envisagé si on arrive à positionner le marquage à l'arrière du véhicule (sur le pare-chocs par exemple),
- utiliser deux caméras, une pour voir de près et une pour voir de loin comme fait dans le projet "Promote Chauffeur", mais cela entraîne une augmentation du coût,
- utiliser une caméra munie d'un zoom, mais la complexité qu'introduit le zoom risque de diminuer la précision de localisation,
- choisir de se restreindre à la vision de près ou à la vision de loin.

focale (mm)	Taille du motif (cm)	2 m	10 m	30 m	40 m	60 m	80 m
4	2	6.4	1.3	<1	<1	<1	<1
4	3	9.6	1.9	<1	<1	<1	<1
4	4	12.8	2.6	<1	<1	<1	<1
6	2	9.6	1.9	<1	<1	<1	<1
6	3	14.4	2.9	<1	<1	<1	<1
6	4	19.2	3.8	1.3	<1	<1	<1
8	2	12.8	2.6	<1	<1	<1	<1
8	3	19.2	3.8	1.3	<1	<1	<1
8	4	25.6	5.1	1.7	1.3	<1	<1
10	2	16	3.2	1	<1	<1	<1
10	3	24	4.8	1.6	1.2	1	<1
10	4	32	6.4	2.2	1.6	1	<1

TAB. 1.1: Largeur minimale d'un motif en pixel dans l'image en fonction de la focale, de la taille du motif dans l'espace et de la distance à laquelle il est vu (taille pixelique  $6.25\mu\text{m}$ , résolution : 1024x768).

focale (mm)	demi angle (°)	2 m	10 m	30 m	40 m	60 m	80 m
4	39	3.2	16	48	64	96	128
6	28	2.1	10.7	32	42.7	64	85.4
8	22	1.6	8	24	32	48	64
10	18	1.3	6.4	19.2	25.6	38.4	51.2

TAB. 1.2: Ouverture du champ de la caméra en fonction de la focale et taille (en m) de la zone visible à 2m, 10m, 30m, 40m, 60m et 80m (taille pixelique  $6.25\mu\text{m}$ , résolution 1024x768).

Mais aucune n'est vraiment satisfaisante. Dans l'immédiat, nous faisons le choix de nous restreindre à la vision proche. Toutefois, à plus long terme, nous envisageons d'agrandir le marquage car c'est la solution la plus simple et la moins coûteuse.

## 1.4 Principe général du système

Le système de détection doit permettre en temps réel de localiser sur l'image la ou les balises, de décoder l'information portée par chaque balise puis de localiser précisément le ou les obstacle(s) dans l'espace. Les différentes étapes du processus sont visibles sur la figure 1.7. Par la suite nous avons légèrement modifié l'ordon-

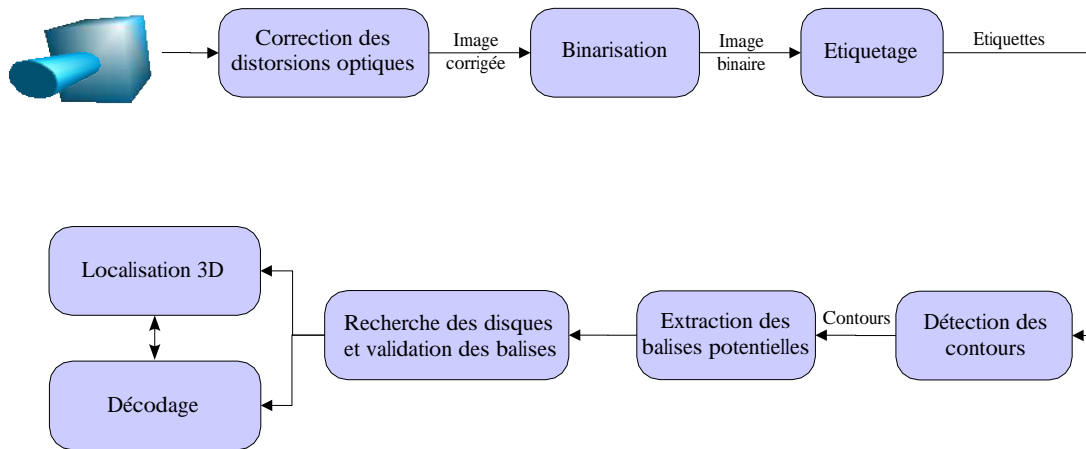


FIG. 1.7: Ordonnancement des différentes étapes du processus de détection et de localisation.

nancement des étapes pour pouvoir accélérer l'exécution de l'étape d'étiquetage. Nous avons interverti les étapes de détection de contours et d'étiquetage. Le nouvel ordonnancement est visible dans la figure 1.8. Voyons en détail chacune des étapes

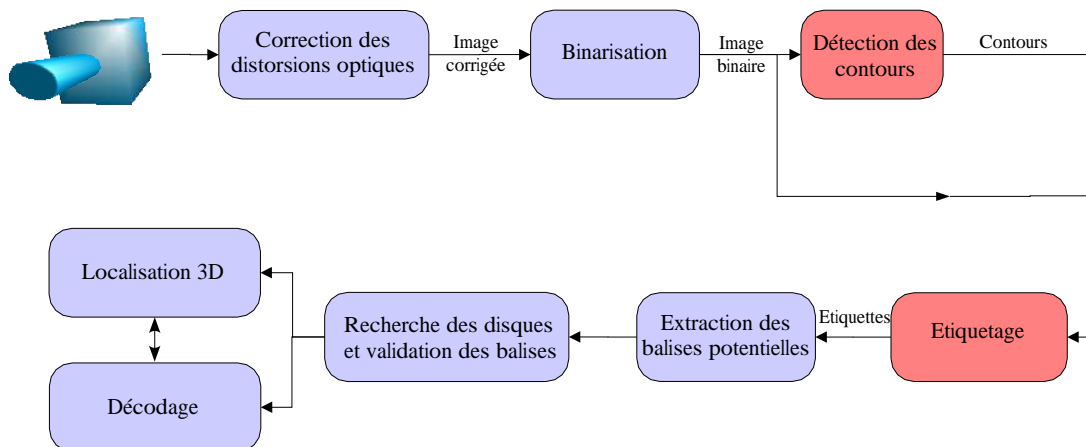


FIG. 1.8: Ordonnancement corrigé des étapes du processus de détection et de localisation : les étapes modifiées sont coloriées en rouge.

du processus.

### 1.4.1 Correction des distorsions optiques

Cette étape est incontournable. En effet, pour pouvoir localiser correctement les obstacles dans l'espace à partir de l'image issue de la caméra, il faut que cette image soit exempte de déformation. Une petite erreur dans la localisation d'un point de contrôle sur l'image entraîne une erreur énorme dans l'espace, même avec un algorithme de localisation très précis. L'algorithme utilisé est proche de celui mis au point par Zhang [Zha02]<sup>[1]</sup>.

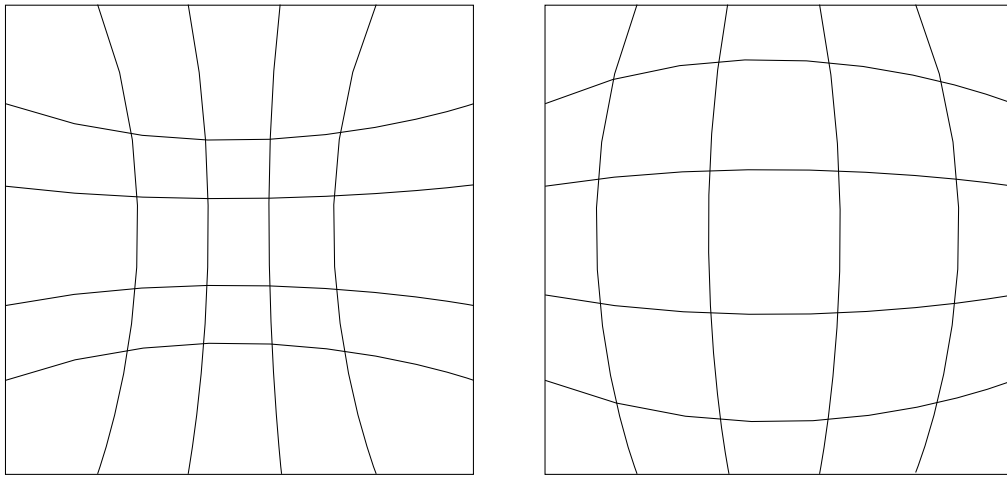


FIG. 1.9: Distorsions géométriques de l'image : à gauche la déformation en croissant et à droite la déformation en barillet.

#### 1.4.1.1 Formation de l'image.

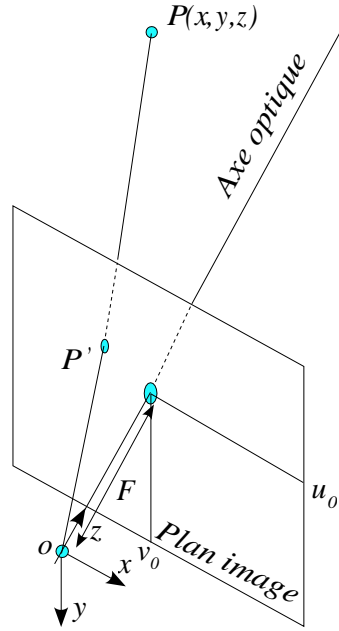
Notons  $P(x, y, z)$  un point de l'espace exprimé dans le repère caméra. Ce point se projette sur le plan image en  $P'(x', y', z')$  et il est vu en  $(u, v)$  sur l'image. L'axe optique coupe le plan image en  $(u_0, v_0)$ . Notons  $F$  la distance focale (figure 1.10). Nous avons :

$$x' = F \frac{x}{z} \quad (1.1)$$

---

[1] Zhengyou Zhang. A flexible new technique for camera calibration. Technical Report MSR-TR-98-71, Microsoft Research, August 2002.



FIG. 1.10: Le modèle *Pinhole*.

$$y' = F \frac{y}{z} \quad (1.2)$$

$$z' = F \quad (1.3)$$

et

$$u = \frac{x'}{d_x} + u_0 \quad (1.4)$$

$$v = \frac{y'}{d_y} + v_0 \quad (1.5)$$

avec  $d_x$  et  $d_y$  la taille d'un pixel.

Le calcul simplifié des rayons réfractés suppose le respect des conditions de Gauss ( $\sin(\theta) = \theta$ ) et néglige les termes en  $k_n \theta^{(2n+1)}$ . Les distorsions géométriques sont majoritairement la conséquence de cette simplification.

Dans le cas idéal,  $P'$  devrait se trouver en  $(x', y', z')$  mais il est en réalité en  $(x' + \Delta x', y' + \Delta y', z')$ . Notons  $\rho$ , la distance entre l'intersection de l'axe optique et le plan image  $(u_0, v_0)$  et  $P'(x', y')$ ,  $\Delta\rho$  est l'erreur radiale (l'erreur tangentielle

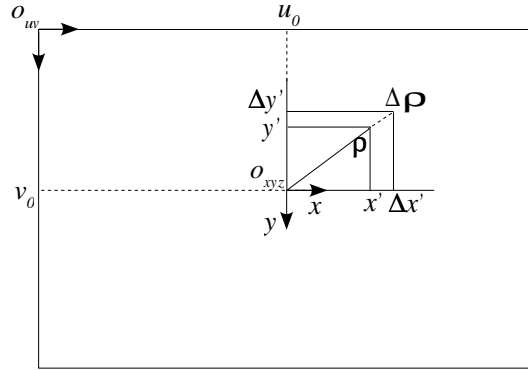


FIG. 1.11: Le plan image.

est négligée) (figure 1.11). Nous pouvons écrire  $\Delta\rho$  :

$$\Delta\rho = k_1\rho^3 + k_2\rho^5 + \dots \quad (1.6)$$

- Si  $k_1$  est négatif  $\Delta\rho$  est négatif et donc la déformation est en barillet.
- Si  $k_1$  est positif  $\Delta\rho$  est positif et donc la déformation est en croissant (figure 1.9).

#### 1.4.1.2 Principe général de correction

Exprimons les coordonnées  $(u, v)$  d'un point réel  $P'$  de l'image en fonction des coordonnées idéales  $(u_{id}, v_{id})$  de ce point.

Nous avons :

$$\frac{\Delta x'}{x'} = \frac{\Delta\rho}{\rho} \Rightarrow \Delta x' = x'\rho^2 k_1 \quad (1.7)$$

Par l'équation (1.4) nous pouvons déduire :

$$x' + \Delta x' = (u - u_0)d_x - e(x)d_x \quad (1.8)$$

avec  $e(x)$  l'erreur de discrétisation  $(-\frac{1}{2} < e(x) < \frac{1}{2})$ . ( $e(x)$  est négligé dans le reste du calcul). Nous pouvons écrire :

$$u = \frac{x'}{d_x} + \frac{\Delta x'}{d_x} + u_0 \quad (1.9)$$

A l'aide de l'équation (1.7), nous obtenons :

$$u = \frac{x'}{d_x} + \frac{x' \rho^2 k_1}{d_x} + u_0 \quad (1.10)$$

Remarquons que

$$\rho^2 = x'^2 + y'^2 \quad (1.11)$$

donc :

$$u = \frac{x'}{d_x} (1 + k_1 (x'^2 + y'^2)) + u_0 \quad (1.12)$$

A l'aide des équations (1.1-1.2)

$$u = \frac{F x}{z d_x} \left( 1 + k_1 \left( \left( \frac{F x}{z} \right)^2 + \left( \frac{F y}{z} \right)^2 \right) \right) + u_0 \quad (1.13)$$

$$\Rightarrow u = \frac{F x}{z d_x} + u_0 + k_1 \frac{F^3 x}{d_x z} \left( \left( \frac{x}{z} \right)^2 + \left( \frac{y}{z} \right)^2 \right) \quad (1.14)$$

notons que :

$$u_{id} = \frac{F x}{z d_x} + u_0 \text{ et } \frac{x}{z} = \frac{(u_{id} - u_0) d_x}{F} \quad (1.15)$$

(1.14) devient :

$$u = u_{id} + k_1 (u_{id} - u_0) (d_x^2 (u_{id} - u_0)^2 + d_y^2 (v_{id} - v_0)^2) \quad (1.16)$$

donc

$$u = P_u(u_{id}, v_{id}) \quad (1.17)$$

La démarche est strictement identique pour  $v$ , on obtient :

$$v = P_v(u_{id}, v_{id}) \quad (1.18)$$

avec  $P_u$  et  $P_v$  deux polynômes de degré 3. Nous avons une relation qui lie les positions idéales et les positions réelles des points de l'image. Nous pouvons donc corriger l'image : pour chaque point idéal, on peut chercher sa valeur dans l'image réelle.

### 1.4.1.3 Remarques sur l'implémentation de l'algorithme

L'implémentation de cet algorithme est en deux parties. La première consiste en la détermination des coefficients des polynômes. La seconde est la correction. Pour estimer les coefficients du polynôme, on calcule :

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} P_u \\ P_v \end{bmatrix} [UV_{id}] \quad (1.19)$$

avec les 2 x 10 paramètres du polynôme :

$$P_u = [p_{u9} \cdots p_{u0}] \quad (1.20)$$

$$P_v = [p_{v9} \cdots p_{v0}] \quad (1.21)$$

et

$$UV_{id} = [u_{id}^3 \quad v_{id}^3 \quad u_{id}^2 v_{id} \quad v_{id}^2 u_{id} \quad u_{id}^2 \quad v_{id}^2 \quad u_{id} v_{id} \quad u_{id} \quad v_{id} \quad 1]^t \quad (1.22)$$

Nous avons :

$$\begin{bmatrix} u \\ v \end{bmatrix} - \begin{bmatrix} P_u \\ P_v \end{bmatrix} [UV_{id}] = 0 \quad (1.23)$$

Comme nous avons 2 x 10 paramètres à estimer, il nous faut au moins 10 couples de points. En pratique, pour obtenir un résultat suffisamment précis, nous utilisons bien plus que dix points.

Pour obtenir une correction en temps réel, la correction à appliquer pour chaque pixel doit être pré-calculée. Pour chaque point idéal  $(u_{id}, v_{id})$  l'antécédent dans l'image réelle  $(u, v)$  doit être dans une LUT.

Avec un caméra couleur, pour améliorer encore la précision de la correction, il est possible de calculer un couple de polynômes pour chaque plan de couleur. Comme l'indice de réfraction varie en fonction de la longueur d'onde, la déformation peut varier en fonction de la couleur.

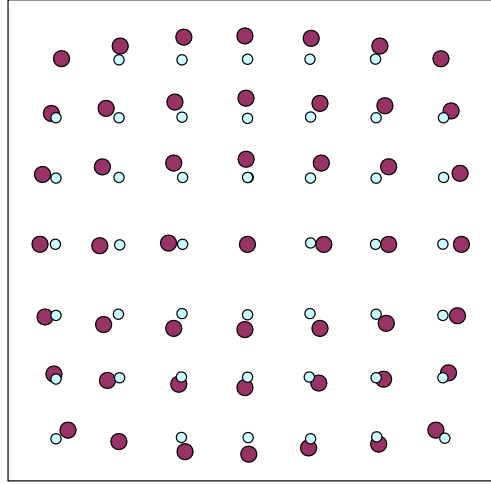


FIG. 1.12: La mire de calibrage : En clair la position idéale et en sombre la position réelle.

#### 1.4.1.4 Remarques sur la mire de calibrage

Pour calculer les coefficients du polynôme, nous utilisons une mire de calibrage formée par une matrice de disques. Nous extrayons le centre de gravité de chaque disque sur l'image et nous l'associons à la position idéale qu'il devrait occuper (La mire de calibrage doit couvrir l'intégralité de l'image et surtout la périphérie sans quoi une partie de l'image serait déformée).

Le choix du motif de calibrage est important. Il est impossible d'utiliser une mire formée par des lignes et d'extraire les points de contrôle par l'intersection de ces lignes car, du fait des distorsions, les lignes ne sont pas droites. Nous utilisons le centre de gravité des disques. La projection de disques donne des ovoïdes (sauf cas particuliers). Le centre de gravité du disque ne correspond donc pas tout à fait à la projection du centre du disque. Toutefois nous avons montré que l'erreur du placement du centre de gravité  $(\Delta G_x, \Delta G_y)$  est :

$$\Delta G_x \approx \frac{D^2 F(2t_x t_y^2 - t_y r_x r_y + t_x r_x^2)}{2t_z^3(2 - r_x^2 - r_y^2)} \quad (1.24)$$

$$\text{de même pour } \Delta G_y \quad (1.25)$$

avec  $D$  le diamètre du disque,  $(t_x t_y t_z)$  la translation,  $(r_x r_y r_z)$  développement limité au premier ordre (il s'agit de l'erreur de positionnement) et  $F$  la distance focale de la caméra. L'erreur est donc en  $\frac{1}{f^2}$ , ce qui est clairement négligeable.

La position idéale des points de contrôle est donnée relativement à un facteur d'échelle, l'algorithme corrigera les distorsions optiques mais aussi des petites erreurs sur les paramètres intrinsèques et extrinsèques (relativement à un facteur près).

Pour gagner du temps, il serait préférable de calculer la position des points idéaux en fonction des points réels. En effet nous n'aurions à corriger que les points de contrôle et non toute l'image. Mais pour l'instant nous ne sommes pas parvenus à inverser le modèle. Il existe toutefois un modèle de distorsions inversible ([MCM03]<sup>[1]</sup>) mais ce modèle de correction est moins performant que le modèle que nous utilisons. Il repose sur un polynôme de correction de degré deux.

## 1.4.2 Binarisation et détection de contours

Ces étapes ne présentent pas de difficultés particulières. Nous utilisons un seuil adaptatif pour la binarisation et les contours sont extraits par morphologie mathématique (érosion puis soustraction de l'original et de l'érodé). Nous profitons aussi de la binarisation pour supprimer les pixels "isolés".

## 1.4.3 Etiquetage

Initialement l'étiquetage était un étiquetage classique en L. Cet type d'étiquetage s'effectue en 2 passes, une première pour étiqueter les régions et une seconde pour fusionner les régions connexes n'ayant pas une étiquette commune. Pour améliorer la vitesse d'exécution, nous voulions remplacer cet algorithme par un étiquetage par segments. Mais cet algorithme, comme le précédent, nécessite deux passes, il n'optimise que la première passe. Il serait intéressant d'avoir un algorithme en une seule passe.

Cela est possible, si l'on regarde notre schéma général : on effectue une binari-

---

[1] Lili Ma, YangQuan Chen, and Kevin L. Moore. A new analytical radial distortion model for camera calibration. *arXiv:cs.CV/0307046v1*, July 2003.

sation, l'étiquetage puis une détection de contours sur les régions étiquetées. En inversant l'étiquetage et la recherche de contours on réussit à effectuer l'étiquetage en une passe.

Le principe est simple, on parcourt l'image des contours ligne à ligne. Chaque nouveau contour rencontré est étiqueté par un suiveur de contours puis, l'étiquette est propagée horizontalement sur l'image binarisée jusqu'à l'extrémité de la zone. De même, pour chaque contour déjà étiqueté rencontré, on propage horizontalement le numéro de son étiquette dans l'image binarisée jusqu'à la fin de la zone.

Nous pouvons, de manière simpliste, estimer le temps gagné par cette modification. Si l'on a une image de taille  $N \times M$ , on peut dire :

- en ce qui concerne l'étiquetage en L, pour chaque pixel nous accédons à trois pixels au premier passage puis à un pixel au second, il y a donc  $3(N * M) + N * M$  accès,
- en ce qui concerne l'étiquetage par segments, nous accédons à un pixel au premier passage puis un au second, il a donc  $N * M + N * M$  accès,
- en ce qui concerne notre étiquetage, nous parcourons une fois l'image et, durant ce parcours, nous suivons les contours de chaque région. Si on note  $P_i$  le périmètre de la région  $i$  nous avons donc  $\sum_i K_i P_i + N * M$ .

Comme nous pouvons facilement dire que  $\sum_i P_i \ll N * M$ , nous avons un gain certain de temps. De plus on s'affranchit du mécanisme de fusion des étiquettes qui est un peu lourd (durant l'étiquetage plusieurs calculs sont effectués sur chaque région : aire, centre de gravité... donc à chaque fusion d'étiquettes il faut aussi fusionner toutes ces informations). Le seul cas, où l'algorithme n'est pas plus rapide, est le cas où aucune région ne présente de concavités. Dans ce cas, l'étiquetage par segments est plus rapide. En effet, il n'y a pas lieu de faire de second passage.

Il est possible de se passer de l'image des contours en utilisant le suiveur de contours directement sur l'image binarisée (ce qui permet en plus d'en extraire les contours pendant la phase d'étiquetage). Mais, dans le cas où les régions comportent des "trous", l'algorithme est un peu plus compliqué à mettre en oeuvre.

#### 1.4.4 Extraction des balises potentielles

L'extraction des balises se fait par recherche des alignements des barres. On cherche dans l'image un alignement de centres de gravité de régions à l'aide de la

transformée de Hough. Cela va très vite car il y a, en général, peu de régions (déjà sélectionnées sur des critères de taille). Ensuite on vérifie un certain nombre de critères : les axes d'inertie des régions doivent être colinéaires ou orthogonaux avec le vecteur directeur de la droite qui porte les centres de gravité des barres, les tailles des régions ainsi que les distances entre chaque région doivent être cohérentes. Une fois l'ensemble des critères vérifié, on peut dire que l'on a une balise potentielle. On cherche le quadrilatère englobant pour bien localiser la zone utile.

#### 1.4.5 Recherche des disques et localisation dans l'espace de la balise

Une fois la balise potentielle localisée, il est aisé de localiser les disques car nous savons dans quelle direction chercher autour de la balise. Cela sert en même temps de vérification car si aucun disque n'est trouvé dans une zone où il devrait y en avoir un ou si une région est trouvée mais qu'elle ne répond pas aux critères de forme et de taille donnés, la balise potentielle est rejetée.

Une fois la balise détectée, les disques trouvés, nous nous servons, pour chaque disque, de son barycentre pour extraire un point de contrôle et, à l'aide de tout ces points de contrôle, on localise l'obstacle précisément dans l'espace (la relative insensibilité de la projection des barycentres des disques est encore exploitée). Nous donnerons le détail de ce calcul au chapitre 2.

#### 1.4.6 Décodage de l'information portée par la balise

Comme nous l'avons dit, la balise transmet une information. Dans le cas de l'obstacle, il s'agit d'une classe d'encombrement pour permettre d'évaluer le volume englobant de l'obstacle.

Pour pouvoir décoder le code défini à la section 1.3.3, il suffit de faire le rapport entre la surface image de la barre considérée et la surface image de l'une des aires de calibrage (ce qui donne  $k S/S=k$ ). Mais avant de pouvoir appliquer ce calcul, il faut corriger les aires des barres car elles ont été modifiées par la projection. Pour cela nous utilisons le rapport des surfaces images des deux barres de calibrage (aux extrémités) pour conclure sur une loi affine de l'évolution de l'aire lorsque l'on se



déplace le long de la balise. Cette loi nous donne pour chaque barre un coefficient à appliquer pour corriger son aire. A partir de cette nouvelle valeur, on peut décoder la balise. Cette loi ne modélise pas la réalité mais elle est suffisante dans notre cas. Nous envisageons toutefois de changer de méthode pour une méthode plus coûteuse en temps mais plus robuste : puisque nous connaissons la position de la balise dans l'espace, il nous est possible de corriger les déformations projectives de la balise. On peut donc générer une image de la balise dans une position simple et avec une dimension fixe. Il ne nous reste plus qu'à décoder l'information. Ce qui peut être fait facilement car la dimension et l'orientation de la balise sont maintenant connues. On peut décoder la balise en faisant par exemple un histogramme des pixels en fonction de leur abscisse.

## 1.5 Extension de la problématique

Il est possible de faire évoluer notre problématique pour rendre notre système bien plus pertinent.

### 1.5.1 Extension du champ de vision

Notre problématique initiale était la détection d'obstacles situés vers l'avant. Comme nous l'avons déjà précisé, il est possible d'étendre le problème à la détection d'obstacles tout autour du véhicule. Effectivement, quand on roule, le risque maximum de collision est situé à l'avant. Pourtant, pour développer une assistance à la conduite efficace et complète, il est indispensable de ne pas se restreindre à la vision avant. Il faut pouvoir éviter une collision latérale lors d'un changement de voie, d'un virage ou encore une collision arrière lors d'un freinage trop brutal. Il est pourtant impossible de multiplier le nombre de caméras : une à l'avant, une de chaque côté, une à l'arrière... Nous proposons à l'aide d'un unique capteur de surveiller l'intégralité de l'environnement autour du véhicule. Il s'agit de capteurs catadioptriques panoramiques. A l'aide d'un miroir de révolution placé face à une caméra, on perçoit l'environnement à 360 ° à la vitesse d'une caméra classique (mais au prix d'une diminution de la résolution). L'utilisation en robotique mobile de ce

type de capteur se répand de plus en plus.

Pour pouvoir utiliser ces capteurs, une étude préliminaire importante est nécessaire. Il est possible d'utiliser un capteur en renfort de notre caméra déjà en place. Ce qui réglerait le problème de la section 1.3.5, la caméra classique surveillant les obstacles loin devant et la caméra panoramique surveillant tout autour du véhicule. Nous proposons une étude sur ce type de capteurs pour les calibrer et localiser les obstacles.

Il est aussi possible de profiter d'une zone commune dans le champ de vision des deux caméras pour faire de la stéréovision et, ainsi, améliorer le système général. Pour cela, nous étudierons la géométrie épipolaire de couples caméra panoramique/caméra plan et aussi caméra panoramique/caméra panoramique.

### 1.5.2 Extension de la détection aux panneaux de signalisation

Le système tel que nous l'avons développé peut être transposé facilement pour résoudre un problème similaire : la détection de la signalisation routière.

Pour adapter notre système à ce problème, il suffit d'équiper chaque panneau d'un panneau muni d'une balise du même type que celui que l'on a présenté pour les obstacles. La différence sera que l'information portée sur le codage ne sera plus une information concernant l'obstacle mais, le panneau lui-même. Le codage que nous avons permis d'obtenir 728 combinaisons différentes or, si l'on dénombre les panneaux de signalisation (voir [Rou]<sup>[1]</sup>), on trouve 32 panneaux de dangers, 1 panneau d'obligation de céder le passage, 1 panneau de stop, 2 panneaux de route prioritaire, 18 panneaux d'obligation et 18 panneaux de fin d'interdiction ou fin d'obligation. Cela fait en tout un minimum de 72 panneaux importants. A cela on peut ajouter les panneaux provisoires (chantier). Notre codage est donc largement suffisant. Il reste toutefois les panneaux d'information (carrés) qui ne sont, la plupart du temps, pas nécessaires à la navigation.

Par ailleurs, les disques destinés à la localisation permettront de positionner le panneau dans l'espace afin d'en déduire la distance à notre véhicule et, pourquoi pas, en fonction de sa position, de conclure si l'information portée nous est destinée

---

[1] La Sécurité Routière. <http://www.securiteroutiere.equipement.gouv.fr>.

ou pas.

## 1.6 Résultats expérimentaux

Voyons sur des images le résultat de la localisation de véhicules. Nous pouvons constater sur les images de la figure 1.13 que la localisation est correcte. La distance renvoyée par le programme, quoi que difficile à confirmer semble correcte. Nous ne sommes toutefois pas en mesure de quantifier l'erreur de localisation.

Si l'on essaie d'estimer le volume englobant du véhicule, les résultats sont plus



FIG. 1.13: Résultats de la localisation sur des scènes de parking : Le véhicule est correctement localisé (la luminosité a été corrigée pour mettre en évidence l'environnement).

décevants (figure 1.14). C'est pourtant prévisible : si nous positionnons un des points de contrôle de la mire dans l'espace avec une erreur de 5 cm dans le plan horizontal, cela fait une erreur d'environ  $8^\circ$ . A l'avant de la voiture, longue de presque 3,5 m, cette erreur devient de l'ordre de 50 cm !

Le procédé de localisation, permet bien de repérer l'arrière du véhicule. Il fournis

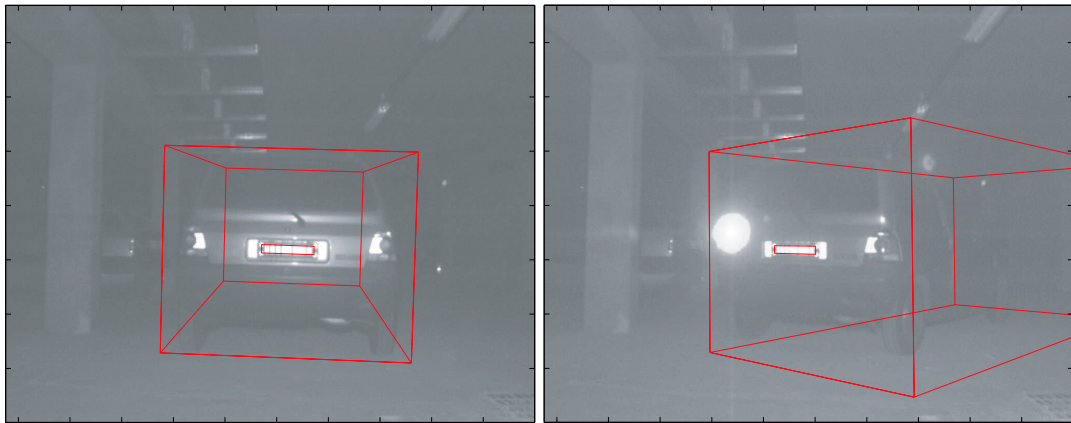


FIG. 1.14: Estimation des volumes englobants des véhicules : à gauche, le résultat est correct mais pas à droite. Dans certain cas, le résultat est décevant.

une bonne estimation de la distance. Toutefois, l'orientation n'étant pas parfaite, le volume englobant est peu précis à l'extrémité du véhicule.

## 1.7 Conclusion

Nous avons présenté le principe général de notre système et nous avons détaillé certaines étapes notamment la correction des distorsions optiques et l'étiquetage pour lesquelles nous proposons des méthodes nouvelles. Nous avons aussi étendu notre problème à la détection de la signalisation routière. De même, nous avons étendu la détection d'obstacles aux obstacles situés tout autour du véhicule et pas seulement à l'avant.

Les résultats sont encourageant même si le volume englobant retrouvé est peu précis à l'extrémité du véhicule détecté. A terme, la balise devrait être agrandie, ce qui réglerait plusieurs problèmes et améliorerait la précision de localisation.

Une autre amélioration sensible du système serait d'utiliser une méthode de suivi

(filtre de Kalman...). Elle permettrait d'atténuer l'imprécision du volume englobant. De même, elle pourrait permettre de gérer certains cas d'occultation de la balise.

Enfin nous avons conduit quelques tests, avec des diodes infrarouges, pour rendre la balise évolutive au cours du temps (afin de pouvoir transmettre des informations). Mais, il nous était impossible de distinguer les diodes effectivement allumées et celle qui étaient éteintes du fait du reflet de notre éclairage. Nous n'avons donc pas eu de résultats significatifs mais il serait intéressant de conduire plus de tests afin de permettre une communication entre l'obstacle et le véhicule.



# Chapitre 2

## Localisation Monoculaire : Le Problème Perspective-N-Points

### Sommaire

---

<b>2.1</b>	<b>Introduction et état de l'art</b> . . . . .	<b>44</b>
<b>2.2</b>	<b>Solution proposée</b> . . . . .	<b>51</b>
<b>2.3</b>	<b>Exemples</b> . . . . .	<b>53</b>
2.3.1	Problème P-3-P pour des points alignés . . . . .	53
2.3.2	Problème P-4-P pour des points coplanaires . . . . .	54
2.3.3	Problème P-5-P pour des points coplanaires . . . . .	56
2.3.4	Problème P-5-P pour des points non coplanaires . . . . .	58
<b>2.4</b>	<b>Cas considéré</b> . . . . .	<b>60</b>
2.4.1	Expression de la solution . . . . .	61
2.4.2	Influence de l'erreur de localisation des points sur l'image . . . . .	62
<b>2.5</b>	<b>Résultats expérimentaux</b> . . . . .	<b>65</b>
<b>2.6</b>	<b>Conclusion</b> . . . . .	<b>65</b>

---

## 2.1 Introduction et état de l’art

Le problème “Perspective-N-Points” (P-N-P) est un problème assez célèbre et relativement ancien. Il consiste, à partir de  $N$  points de contrôle connus dans un repère qui leur est propre (repère objet) et à partir d’une image unique de ces  $N$  points, à retrouver la distance entre le capteur et les points (La correspondance entre les points image et les points espace est supposée connue). On trouve de nombreuses contributions à ce problème et de nombreuses solutions existent déjà. Le problème peut être abordé de deux manières différentes : la première consiste à trouver la translation et la rotation qui permet de passer du repère caméra au repère objet et la seconde consiste à trouver la distance qui sépare le foyer de la caméra et chacun des points de contrôle. Si les deux formulations sont très similaires, elles ne sont pas équivalentes ([HW02]<sup>[1]</sup>).

Pour  $N = 1$ , le problème a une infinité de solutions, pour  $N = 2$  aussi. A partir de  $N = 3$ , le problème a un nombre fini de solutions (deux la plupart du temps mais il est possible d’en avoir jusqu’à quatre, Wolfe et al. [WMSM91]<sup>[2]</sup> montrent graphiquement les différents cas tandis que Gao et al. [GHTC03]<sup>[3]</sup> donnent en plus les critères analytiques qui déterminent le nombre de solutions). Pour  $N = 4$ , il peut y avoir jusqu’à cinq solutions [HW02]<sup>[1]</sup> et pour  $N = 5$  jusqu’à deux [FB81, HW02]<sup>[4,1]</sup>. L’unicité de la solution n’apparaît que pour  $N = 6$ . Toutefois il est possible en ajoutant des contraintes sur les points d’obtenir une solution unique avec moins de points. Par exemple pour  $N = 4$ , on peut, en contraignant le positionnement, n’avoir qu’une solution. C’est le cas si les points sont coplanaires.

Remarquons ici que nous parlons de solutions physiquement réalisables. En réalité,

- 
- [1] Z. Y. Hu and F. C. Wu. A note on the number of solutions of the noncoplanar p4p problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):550–555, 2002.
  - [2] William J. Wolfe, Donald Mathis, Cheryl Weber Sklair, and Michael Magee. The perspective view of three points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(1):66–73, January 1991.
  - [3] Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, and Hang-Fei Cheng. Complete solution classification for the perspective-three-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8):930–943, August 2003.
  - [4] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communication of the ACM*, 24(6):381–395, 1981.



il y a toujours le double de solutions, chaque solution ayant son symétrique derrière le centre de projection mais ces solutions négatives sont immédiatement exclues. Le cas  $N = 3$  a été très étudié [FB81, HLON91, WMSM91, DD92, Alt94, GHTC03]<sup>[4, 5, 2, 6, 7, 3]</sup> et le cas  $N = 4$  aussi [FB81, AC95, HW02]<sup>[4, 8, 1]</sup>.

Parmi les nombreuses solutions existantes, plusieurs partent de la formulation introduite par Grunert en 1841 [HLON91, FB81, GHTC03]<sup>[5, 4, 3]</sup>. Elle consiste à appliquer la formule du cosinus généralisé dans la pyramide perspective. Mais cette formulation entraîne un système complexe : si on note  $P_1$  à  $P_n$  les points de contrôle,  $P'_1$  à  $P'_n$  leurs projetés sur le plan image,  $F$  le foyer de la caméra et  $d_{PQ}$  la distance du point  $P$  au point  $Q$ . On peut écrire pour 3 points (figure 2.1) le système :

$$d_{P_1 P_2}^2 = d_{F P_1}^2 + d_{F P_2}^2 - 2d_{F P_1} d_{F P_2} \cos \widehat{P_1 F P_2} \quad (2.1)$$

$$d_{P_2 P_3}^2 = d_{F P_2}^2 + d_{F P_3}^2 - 2d_{F P_2} d_{F P_3} \cos \widehat{P_2 F P_3} \quad (2.2)$$

$$d_{P_1 P_3}^2 = d_{F P_1}^2 + d_{F P_3}^2 - 2d_{F P_1} d_{F P_3} \cos \widehat{P_1 F P_3} \quad (2.3)$$

Il existe plusieurs solutions analytiques à ce système : Haralick et al. dans [HLON91]<sup>[5]</sup> en résumant plusieurs et comparent leur stabilité (Grunert 1841, Finsterwalder 1903, Merritt 1949, Fischler et Bolles 1981, Linnainmaa et al. 1988 et Grafarend et al. 1989). Ces solutions sont toutes établies pour le cas  $N = 3$ . Nous ne verrons pas ici l'intégralité de ces méthodes. Nous rappellerons toutefois la méthode de Grunert, les autres étant généralement du même type.

Pour résoudre le système (2.1) à (2.3) Grunert propose les changements de va-

- 
- [5] Robert M Haralick, Chung Nan Lee, Kartsen Ottenberg, and Michael Nölle. Analysis and solutions of the three points perspective pose estimation problem. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 592–598, 1991.
  - [6] Daniel DeMenthon and Larry S. Davis. Exact and approximate solutions of the perspective-three-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(11):1100–1105, November 1992.
  - [7] T. D. Alter. 3-d pose from 3 points using weak-perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8):802–808, 1994.
  - [8] M. A. Abidi and T. Chandra. A new efficient and direct solution for pose estimation using quadrangular targets: Algorithm and evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):534–538, 1995.

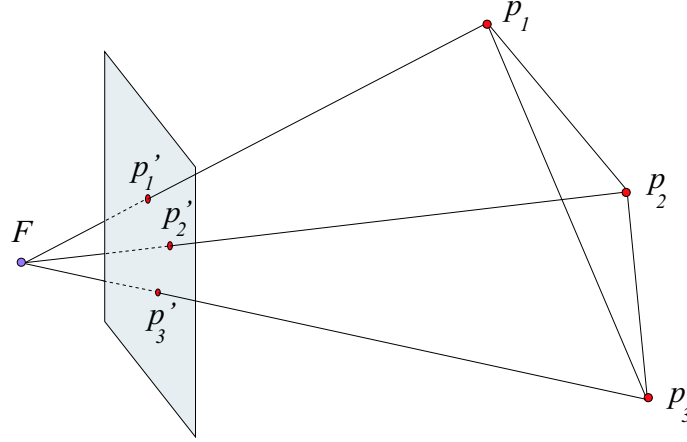


FIG. 2.1: Le problème P-3-P : 3 points de contrôle de l'espace se projettent sur l'image. Il faut retrouver la distance entre ces points de contrôle et le capteur d'image.

riables suivant :

$$d_{FP_2} = u d_{FP_1} \quad (2.4)$$

$$d_{FP_3} = v d_{FP_1} \quad (2.5)$$

ce qui permet d'avoir :

$$d_{FP_1}^2 = \frac{d_{P_2P_3}^2}{u^2 + v^2 - 2uv \cos \widehat{P_2FP_3}} = \frac{d_{P_1P_3}^2}{1 + v^2 - 2v \cos \widehat{P_1FP_3}} = \frac{d_{P_1P_2}^2}{1 + u^2 - 2u \cos \widehat{P_1FP_2}} \quad (2.6)$$

d'où

$$u^2 + \frac{d_{P_1P_3}^2 - d_{P_2P_3}^2}{d_{P_1P_3}^2} - 2uv \cos \widehat{P_2FP_3} + \frac{2d_{P_2P_3}^2}{d_{P_1P_3}^2} v \cos \widehat{P_1FP_3} - \frac{d_{P_2P_3}^2}{d_{P_1P_3}^2} = 0 \quad (2.7)$$

$$u^2 - \frac{d_{P_1P_2}^2}{d_{P_1P_3}^2} v^2 + 2v \frac{d_{P_1P_2}^2}{d_{P_1P_3}^2} \cos \widehat{P_1FP_3} - 2u \cos \widehat{P_1FP_2} - \frac{d_{P_1P_3}^2 - d_{P_1P_2}^2}{d_{P_1P_3}^2} = 0 \quad (2.8)$$

De (2.7) on peut tirer une expression de  $u^2$  et, en introduisant cette expression dans (2.8), on peut écrire :

$$u = \frac{\left(-1 + \frac{d_{P_2P_3}^2 - d_{P_1P_2}^2}{d_{P_1P_3}^2}\right) v^2 - 2 \left(\frac{d_{P_2P_3}^2 - d_{P_1P_2}^2}{d_{P_1P_3}^2}\right) \cos v + 1 + \frac{d_{P_2P_3}^2 - d_{P_1P_2}^2}{d_{P_1P_3}^2}}{2(\cos \widehat{P_1FP_2} - v \cos \widehat{P_2FP_3})} \quad (2.9)$$

Cette expression peut être réintroduite dans (2.7) pour obtenir le polynôme :

$$A_4v^4 + A_3v^3 + A_2v^2 + A_1v^1 + A_0 = 0 \quad (2.10)$$

La solution générale du système repose sur la résolution de ce polynôme de degré 4. Une fois la valeur de  $v$  extraite on peut, à l'aide de (2.9), tirer  $u$ . Avec  $u$  et  $v$  on peut évaluer  $d_{FP_1}$  (2.6). Enfin on peut obtenir  $d_{FP_2}$  et  $d_{FP_3}$  avec (2.4) et (2.5). Les autres solutions résumées dans l'article de Haralick et al. [HLON91]<sup>[1]</sup> sont des variantes non dépourvues d'intérêt.

Remarquons que Fischler et Bolles [FB81]<sup>[2]</sup> ne proposent pas uniquement une solution analytique pour le cas  $N = 3$  mais aussi pour le cas  $N = 4$  lorsque les points sont coplanaires.

Gao et al. [GHTC03]<sup>[3]</sup> donnent aussi une solution analytique du p-3-p ainsi que des critères pour prévoir le nombre de solutions. L'utilisation combinée de la solution analytique et des critères leur permet de définir un algorithme (CASSC) qui leur permet d'obtenir de manière robuste les solutions du p-3-p.

Toutes ces solutions analytiques reposent sur des polynômes de degré supérieur à 2 la plupart du temps. Ils rendent fréquemment les solutions assez instables numériquement.

D'autres approches permettent d'extraire les solutions du système en le linéarisant ([QL99, AQT02]<sup>[4,5]</sup>)

Certains ont changé la formulation initiale et ont exprimé les solutions en utilisant

- 
- [1] Robert M Haralick, Chung Nan Lee, Kartsen Ottenberg, and Michael Nölle. Analysis and solutions of the three points perspective pose estimation problem. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 592–598, 1991.
  - [2] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communication of the ACM*, 24(6):381–395, 1981.
  - [3] Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, and Hang-Fei Cheng. Complete solution classification for the perspective-three-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8):930–943, August 2003.
  - [4] Long Quan and Zhong-Dan Lan. Linear n-point camera pose determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):774–780, August 1999.
  - [5] M-A. Ameller, Long Quan, and B. Triggs. Camera pose revisited: New linear algorithms. In *14ème Congrès Francophone de Reconnaissance des Formes et Intelligence Artificielle*, 2002. Paper in French.

des modèles simplifiés de caméra (voir [HCD94]<sup>[1]</sup>). DeMenthon et Davis comparent dans [DD92]<sup>[2]</sup> les solutions en fonction du modèle de caméra pour le cas  $N = 3$ . Alter propose dans [Alt94]<sup>[3]</sup> de résoudre le problème pour  $N = 3$  en utilisant une perspective faible, le modèle *weak perspective* qui consiste à approximer la projection perspective par une projection orthogonale et une mise à l'échelle (figure 2.2). Ce modèle devient aberrant dans le cas où les points sont à des distances trop différentes de la caméra. Si l'on respecte les notations de l'article (rappelé à la

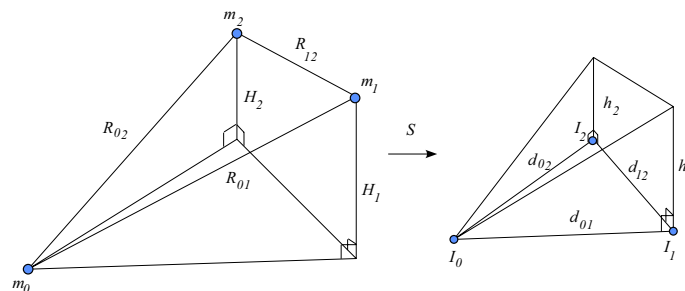


FIG. 2.2: Solution du problème P-3-P en utilisant le modèle *weak perspective* qui approxime la projection perspective par une projection orthogonale et une mise à l'échelle.

figure 2.2) on peut écrire, d'après le théorème de Pythagore :

$$\begin{aligned} h_1^2 + d_{01}^2 &= r_{01}^2 \\ h_2^2 + d_{02}^2 &= r_{02}^2 \\ (h_1 - h_2)^2 + d_{12}^2 &= r_{12}^2 \end{aligned} \tag{2.11}$$

ce qui donne :

$$h_1^2 + d_{01}^2 = sR_{01}^2$$

- 
- [1] Radu Horaud, Stéphane Christy, and Fadi Dornaika. Object pose: The link between weak perspective, para perspective, and full perspective. Technical Report 2356, Institut National de Recherche en Informatique et en Automatique, September 1994.
- [2] Daniel DeMenthon and Larry S. Davis. Exact and approximate solutions of the perspective-three-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(11):1100–1105, November 1992.
- [3] T. D. Alter. 3-d pose from 3 points using weak-perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8):802–808, 1994.

$$\begin{aligned} h_2^2 + d_{02}^2 &= sR_{02}^2 \\ (h_1 - h_2)^2 + d_{12}^2 &= sR_{12}^2 \end{aligned} \quad (2.12)$$

Du système 2.12 on peut tirer :

$$a s^4 - 2b s^2 + c = 0 \quad (2.13)$$

D'où les solutions :

$$s = \sqrt{\frac{b + \sqrt{b^2 - ac}}{a}} \quad (2.14)$$

$$(h_1, h_2) = +/- \left( \sqrt{(sR_{01})^2 - d_{01}^2}, \sigma \sqrt{(sR_{02})^2 - d_{02}^2} \right) \quad (2.15)$$

$$(H_1, H_2) = \frac{1}{s}(h_1, h_2) \quad (2.16)$$

avec  $\sigma = 1$  si  $d_{01}^2 + d_{02}^2 - d_{12}^2 \leq s^2(R_{01}^2 + R_{02}^2 - R_{12}^2)$  et  $-1$  sinon. De là il est possible de trouver :

$$\vec{m}_0 = \frac{1}{s}(x_0, y_0, w) \quad (2.17)$$

$$\vec{m}_1 = \frac{1}{s}(x_1, y_1, h_1 + w) \quad (2.18)$$

$$\vec{m}_2 = \frac{1}{s}(x_2, y_2, h_2 + w) \quad (2.19)$$

avec  $w$  la distance focale de la caméra.

Une solution complètement différente et très intéressante est celle proposée par Abidi et Chandra dans [AC95]<sup>[4]</sup> qui permet de solutionner le cas de 4 points coplanaires avec une caméra munie d'un zoom. Ils n'utilisent pas la formulation basée sur les expressions des cosinus mais partent de l'expression de l'aire de la base  $A = |\overrightarrow{FP_i}x\overrightarrow{FP_j}|/2$  et du volume  $V = |\overrightarrow{FP_k} \cdot (\overrightarrow{FP_i}x\overrightarrow{FP_j})|/6$  du tétraèdre. Or le volume peut être aussi exprimé par  $V = Ah/3$  avec  $h$  la hauteur du tétraèdre. Comme ils utilisent 4 points, ils peuvent exprimer l'aire et le volume par 4 équations différentes de ce type et tirer du système obtenu 4 expressions différentes pour  $h$

---

[4] M. A. Abidi and T. Chandra. A new efficient and direct solution for pose estimation using quadrangular targets: Algorithm and evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):534–538, 1995.

faisant intervenir les distances que l'on cherche et la focale. En combinant ces expressions, ils tirent trois des distances recherchées (par exemple  $[Fp_2]$ ,  $[Fp_3]$ , et  $[Fp_4]$ ) en fonction de la quatrième ( $[Fp_1]$ ) et de la distance focale ( $f$ ). Enfin  $[p_1p_2]$  peut être exprimé en fonction de  $[Fp_1]$ ,  $[Fp_2]$  et  $f$  donc seulement en fonction de  $[Fp_1]$  et  $f$ . De même  $[p_1p_3]$  peut être exprimé en fonction de  $[Fp_1]$ ,  $[Fp_3]$  et  $f$  donc seulement en fonction de  $[Fp_1]$  et  $f$ . En divisant la première distance par la seconde,  $[Fp_1]$  disparaît de l'expression et il ne reste plus qu'une inconnue  $f$  (Le rapport entre les deux distances est connu). Il est donc possible de tirer  $f$  puis, de là, les autres inconnues.

Plusieurs approches itératives sont aussi proposées. La plupart du temps elle consistent à estimer la position des points dans l'espace, générer leurs positions sur l'image à l'aide des positions estimées puis, à comparer les positions calculées des points sur l'image avec leurs positions réelles et estimer une erreur qui permet de corriger le positionnement [Low80, Low87, Low91]<sup>[1,2,3]</sup>.

Il est aussi possible d'inclure l'erreur de positionnement directement dans le modèle de projection comme le fait DeMenthon dans [DD95]<sup>[4]</sup>

Enfin il est possible de résoudre le problème de pose en changeant de primitive comme le font Horaud et al. dans [HCLL89]<sup>[5]</sup> qui donnent une solution analytique du problème P-4-P en remplaçant les quatre points par trois lignes ou Qiang et al. qui présentent dans [JCHS00]<sup>[6]</sup> une méthode pour retrouver la position d'un objet à la fois avec une mise en correspondance de points mais aussi de lignes et d'ellipses.

- 
- [1] D.G. Lowe. Solving for the parameters of object models from image descriptions. *ARPA Image Understanding Workshop*, pages 121–127, April 1980.
  - [2] David G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31(3):355–395, 1987.
  - [3] David G. Lowe. Fitting parameterized three-dimensional models to images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):441–450, 1991.
  - [4] Daniel F. DeMenthon and Larry S. Davis. Model-based object pose in 25 lines of code. *International Journal of Computer Vision*, 15(1-2):123–141, 1995.
  - [5] Radu Horaud, Bernard Conio, Olivier Le Boulleux, and Bernard Lacolle. An analytic solution for the perspective-4-point problem. *Computer Vision, Graphics and Image Processing*, 47:33–44, 1989.
  - [6] Qiang Ji, Mauro S. Costa, Robert M. Haralick, and Linda G. Shapiro. A robust linear least-squares estimation of camera exterior orientation using multiple geometric features. *ISPRS Journal Of Photogrammetry And Remote Sensing*, 55(2):75–93, June 2000.

Carceroni et Brown dans [CB97]<sup>[7]</sup> passent en revue la plupart des méthodes importantes existantes pour résoudre le problème P-N-P.

Comme nous venons de le voir, le P-N-P a été très largement traité dans la littérature et des solutions très variées existent. Pourtant nous n'avons pas trouvé de solution qui convienne à notre problème, particulièrement parce qu'on espère pouvoir résoudre ce problème avec des capteurs non linéaires. Essayons donc de trouver une nouvelle solution, plus simple, adaptable à des capteurs panoramiques (non linéaires).

## 2.2 Solution proposée

La plupart des méthodes présentées dans la littérature partent de l'expression du cosinus généralisé. Leurs solutions sont souvent complexes. Les solutions analytiques présentées, reposent sur la résolution de polynômes de degré supérieur à deux qui peuvent introduire une instabilité numérique dans la solution. Nous avons besoin d'une méthode extrêmement rapide et qui permette d'utiliser des capteurs ne disposant pas de foyer de projection unique. Toutes les méthodes déjà existantes que nous avons étudié nécessitent un centre de projection unique. Nous avons formulé le problème différemment, ce qui nous a conduit à trouver une démarche systématique pour obtenir une solution analytique simple. Étonnamment dans la littérature, malgré le nombre de contributions, nous n'avons trouvé aucune solution proche de ce que nous proposons. Notre approche permet de trouver une solution analytique dans le cas de quatre points coplanaires et plus et cinq points non coplanaires et plus. Pour trois points, nous ne sommes pas parvenus à trouver une solution à moins que les trois points ne soient alignés. Comme les solutions que nous obtenons ne font intervenir que des polynômes de degré deux nous espérons avoir une meilleure stabilité que les solutions analytiques déjà existantes.

Pour trouver ces solutions, Nous avons utilisé une mise en équation différentielle. Si l'on note  $E$  un ensemble de  $N$  points de contrôle  $P_i$  (avec  $i$  un entier inférieur à

---

[7] Rodrigo L. Carceroni and Christopher M. Brown. Numerical methods for model-based pose recovery. Technical Report TR659, University of Rochester - Computer Science Department, 1997.

$N$ ) dans l'espace. Ces points se projettent sur le plan image en  $P'_i$ . Soit  $F$  le point focal de la caméra et soit  $\vec{v}_i$  le vecteur directeur de la droite  $(F p_i)$ . On peut écrire :

$$p_i = F + \lambda_i \vec{v}_i \quad (2.20)$$

avec

$$\lambda_i \gg 0 \quad (2.21)$$

On suppose la caméra calibrée et les distorsions optiques corrigées, donc  $\vec{v}_i$  et  $F$  sont connus :

$$\vec{v}_i = \frac{p_i - F}{\|p_i - F\|} \quad (2.22)$$

Par ailleurs on peut simplifier cette expression en plaçant l'origine du repère en  $F$ . Écrit sous cette forme, localiser les points  $p_i$  revient à évaluer les  $\lambda_i$ . Pour y parvenir il nous faut trouver des expressions faisant intervenir ces inconnues.

Ecrivons :

$$\overrightarrow{p_i p_j} = p_j - p_i \quad (2.23)$$

$$= \lambda_j \vec{v}_j - \lambda_i \vec{v}_i \quad (2.24)$$

On peut, soit en écrivant des combinaisons linéaires qui font intervenir les vecteurs  $\overrightarrow{p_i p_j}$ , soit en utilisant des contraintes connues sur les points (principalement la distance entre deux points), obtenir un certain nombre d'expressions faisant intervenir les inconnues  $\lambda$ . La complexité de la solution dépendra des contraintes exprimées.

Pour une démarche plus systématique, il suffit de former une base du plan ou de l'espace (selon que les points sont coplanaires ou non) et exprimer les autres points dans cette base. Par exemple si les points sont coplanaires, on choisit l'un des points comme origine (par exemple  $p_1$ ) et on forme une base avec deux autres points  $p_2$  et  $p_3$ . On peut exprimer les points restants :

$$\overrightarrow{p_1 p_i} = K_{xi} \overrightarrow{p_1 p_2} + K_{yi} \overrightarrow{p_1 p_3} \quad (2.25)$$

Comme les points sont connus dans leur référentiel, les coefficients  $K_{xi}$  et  $K_{yi}$  sont connus par avance. De l'ensemble de ces expressions on peut tirer l'expression analytique des  $\lambda$  en fonction de l'un d'entre eux. Il ne reste que l'expression du  $\lambda$



manquant à trouver. On peut l'obtenir en utilisant l'une des distances entre deux points (de préférence, pour un maximum de précision, la plus grande).

$$\|\overrightarrow{p_i p_j}\| = L_{ij} \quad (2.26)$$

Il ne reste plus qu'à réinjecter l'expression trouvée dans les expressions des autres  $\lambda$  pour avoir l'intégralité des autres solutions.

## 2.3 Exemples

Nous allons appliquer, pas à pas, notre approche sur différents cas pour voir comment elle fonctionne précisément et montrer qu'elle n'est pas restreinte à un cas particulier. Nous aborderons les problèmes P-3-P, P-4-P et P-5-P.

### 2.3.1 Problème P-3-P pour des points alignés

Essayons de trouver une expression analytique du problème P-3-P lorsque les points sont alignés.

Prenons les trois points  $p_1$ ,  $p_2$  et  $p_3$ . Les points sont alignés, nous pouvons donc écrire la relation :

$$\overrightarrow{p_1 p_2} = K \overrightarrow{p_1 p_3} \quad (2.27)$$

ce qui donne :

$$\lambda_2 \vec{v}_2 - \lambda_1 \vec{v}_1 - K (\lambda_3 \vec{v}_3 - \lambda_1 \vec{v}_1) \quad (2.28)$$

on peut en tirer :

$$\lambda_2 = \frac{-v_{y1}v_{x3} - K v_{y1}v_{x3} + v_{y3}v_{x1} + K v_{y3}v_{x1}}{-v_{y2}v_{x3} + v_{x2}v_{y3}} \lambda_1 \quad (2.29)$$

$$\lambda_3 = \frac{-v_{x1}v_{y2} - K v_{y2}v_{x1} + v_{y1}v_{x2} + K v_{y1}v_{x2}}{(-v_{y2}v_{x3} + v_{x2}v_{y3}) K} \lambda_1 \quad (2.30)$$

avec

$$\vec{v}_i = [v_{xi} \ v_{yi} \ v_{zi}]^t \quad (2.31)$$

Il ne nous reste plus qu'à trouver une expression de  $\lambda_1$ . Pour cela nous pouvons écrire :

$$\|\overrightarrow{p_1 p_2}\| = L_{12} \quad (2.32)$$

ce qui donne :

$$(\lambda_2 v_{x2} - \lambda_1 v_{x1})^2 + (\lambda_2 v_{y2} - \lambda_1 v_{y1})^2 + (\lambda_2 v_{z2} - \lambda_1 v_{z1})^2 - L_{12}^2 = 0 \quad (2.33)$$

En combinant les équations (2.29) et (2.33) on obtient le polynôme :

$$((\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2) \lambda_1^2 - L_{12}^2 = 0 \quad (2.34)$$

avec

$$\zeta = \frac{-v_{y1}v_{x3} - K v_{y1}v_{x3} + v_{y3}v_{x1} + K v_{y3}v_{x1}}{-v_{y2}v_{x3} + v_{x2}v_{y3}} \quad (2.35)$$

d'où

$$\lambda_1 = \frac{L_{12}}{\sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.36)$$

en combinant avec les expressions (2.29) et (2.30)

$$\lambda_2 = \frac{(-v_{y1}v_{x3} - K v_{y1}v_{x3} + v_{y3}v_{x1} + K v_{y3}v_{x1})L_{12}}{(-v_{y2}v_{x3} + v_{x2}v_{y3})\sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.37)$$

$$\lambda_3 = \frac{(-v_{x1}v_{y2} - K v_{y2}v_{x1} + v_{y1}v_{x2} + K v_{y1}v_{x2})L_{12}}{(-v_{y2}v_{x3} + v_{x2}v_{y3})K\sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.38)$$

Nous avons donc une solution analytique au problème P-3-P lorsque les trois points sont alignés. Remarquons que pour avoir la solution numérique, il est nécessaire de calculer une racine carrée. Cette racine apporte deux solutions : une positive et une négative. Comme nous disions dans l'introduction, seule la solution positive est à retenir.

Voyons maintenant le cas du P-4-P.

### 2.3.2 Problème P-4-P pour des points coplanaires

En suivant la même démarche que pour 3 points, nous pouvons obtenir une solution analytique pour 4 points coplanaires. Soit nos quatre points  $p_1$ ,  $p_2$ ,  $p_3$  et

$p_4$ . Nous pouvons écrire :

$$K_1 \overrightarrow{p_1 p_2} + K_2 \overrightarrow{p_1 p_4} - \overrightarrow{p_1 p_3} = 0 \quad (2.39)$$

$$\begin{bmatrix} K_1(\lambda_2 v_{x2} - \lambda_1 v_{x1}) + K_2(\lambda_4 v_{x4} - \lambda_1 v_{x1}) - \lambda_3 v_{x3} + \lambda_1 v_{x1} \\ K_1(\lambda_2 v_{y2} - \lambda_1 v_{y1}) + K_2(\lambda_4 v_{y4} - \lambda_1 v_{y1}) - \lambda_3 v_{y3} + \lambda_1 v_{y1} \\ K_1(\lambda_2 v_{z2} - \lambda_1 v_{z1}) + K_2(\lambda_4 v_{z4} - \lambda_1 v_{z1}) - \lambda_3 v_{z3} + \lambda_1 v_{z1} \end{bmatrix} = 0 \quad (2.40)$$

ce qui donne :

$$\begin{bmatrix} K_1 v_{x2} & -v_{x3} & K_2 v_{x4} \\ K_1 v_{y2} & -v_{y3} & K_2 v_{y4} \\ K_1 v_{z2} & -v_{z3} & K_2 v_{z4} \end{bmatrix} \begin{bmatrix} \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \lambda_1 \begin{bmatrix} v_{x1}(K_1 + K_2 - 1) \\ v_{y1}(K_1 + K_2 - 1) \\ v_{z1}(K_1 + K_2 - 1) \end{bmatrix} \quad (2.41)$$

Notons

$$\Delta_{ijk} = \begin{vmatrix} v_{xi} & v_{xj} & v_{xk} \\ v_{yi} & v_{yj} & v_{yk} \\ v_{zi} & v_{zj} & v_{zk} \end{vmatrix} \quad (2.42)$$

d'où

$$\lambda_2 = \frac{-(K_1 + K_2 - 1)(K_2)\Delta_{134}}{-(K_1)(K_2)\Delta_{234}} \lambda_1 \quad (2.43)$$

$$\lambda_3 = \frac{(K_1)(K_1 + K_2 - 1)(K_2)\Delta_{214}}{-(K_1)(K_2)\Delta_{234}} \lambda_1 \quad (2.44)$$

$$\lambda_4 = \frac{-(K_1)(K_1 + K_2 - 1)\Delta_{231}}{-(K_1)(K_2)\Delta_{234}} \lambda_1 \quad (2.45)$$

Il ne reste plus qu'à trouver une expression de  $\lambda_1$ . Pour cela utilisons l'expression :

$$\|\overrightarrow{p_1 p_2}\| = L_{12} \quad (2.46)$$

ce qui donne :

$$(\lambda_2 v_{x2} - \lambda_1 v_{x1})^2 + (\lambda_2 v_{y2} - \lambda_1 v_{y1})^2 + (\lambda_2 v_{z2} - \lambda_1 v_{z1})^2 - L_{12}^2 = 0 \quad (2.47)$$

En substituant  $\lambda_2$  dans (2.47) par son expression (2.43) on obtient :

$$((\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2) \lambda_1^2 - L_{12}^2 = 0 \quad (2.48)$$

avec

$$\zeta = \frac{(K_1 + K_2 - 1)\Delta_{134}}{(K_1)\Delta_{234}} \quad (2.49)$$

Nous pouvons donc tirer  $\lambda_1$

$$\lambda_1 = \frac{L_{12}}{\sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.50)$$

Combiner avec les équations (2.43) à (2.101) on obtient :

$$\lambda_2 = \frac{(K_1 + K_2 - 1)\Delta_{134}L_{12}}{K_1\Delta_{234}\sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.51)$$

$$\lambda_3 = \frac{(K_1 + K_2 - 1)\Delta_{214}L_{12}}{-\Delta_{234}\sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.52)$$

$$\lambda_4 = \frac{(K_1 + K_2 - 1)\Delta_{231}L_{12}}{K_2\Delta_{234}\sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.53)$$

Nous avons donc obtenu une solution analytique du problème P-4-P dans le cas où les points sont coplanaires. Voyons maintenant le cas du problème P-5-P.

### 2.3.3 Problème P-5-P pour des points coplanaires

Essayons de trouver une solution analytique au problème P-5-P dans le cas où les points sont coplanaires.

Nous pouvons écrire les relations suivantes :

$$K_1 \overrightarrow{p_1 p_2} + K_2 \overrightarrow{p_1 p_3} - \overrightarrow{p_1 p_4} = 0 \quad (2.54)$$

$$K_3 \overrightarrow{p_1 p_2} + K_4 \overrightarrow{p_1 p_3} - \overrightarrow{p_1 p_5} = 0 \quad (2.55)$$

on peut en déduire le système suivant :

$$\begin{bmatrix} K_1 (\lambda_2 v_{x2} - \lambda_1 v_{x1}) + K_2 (\lambda_3 v_{x3} - \lambda_1 v_{x1}) - \lambda_4 v_{x4} + \lambda_1 v_{x1} \\ K_1 (\lambda_2 v_{y2} - \lambda_1 v_{y1}) + K_2 (\lambda_3 v_{y3} - \lambda_1 v_{y1}) - \lambda_4 v_{y4} + \lambda_1 v_{y1} \\ K_1 (\lambda_2 v_{z2} - \lambda_1 v_{z1}) + K_2 (\lambda_3 v_{z3} - \lambda_1 v_{z1}) - \lambda_4 v_{z4} + \lambda_1 v_{z1} \\ K_3 (\lambda_2 v_{z2} - \lambda_1 v_{z1}) + K_4 (\lambda_3 v_{z3} - \lambda_1 v_{z1}) - \lambda_5 v_{z5} + \lambda_1 v_{z1} \end{bmatrix} = 0 \quad (2.56)$$

ce qui donne :

$$\begin{bmatrix} K_1 v_{x2} & K_2 v_{x3} & -v_{x4} & 0 \\ K_1 v_{y2} & K_2 v_{y3} & -v_{y4} & 0 \\ K_1 v_{z2} & K_2 v_{z3} & -v_{z4} & 0 \\ K_3 v_{z2} & K_4 v_{z3} & 0 & -v_{z5} \end{bmatrix} \begin{bmatrix} \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 \end{bmatrix} = \lambda_1 \begin{bmatrix} v_{x1}(K_1 + K_2 - 1) \\ v_{y1}(K_1 + K_2 - 1) \\ v_{z1}(K_1 + K_2 - 1) \\ v_{z1}(K_3 + K_4 - 1) \end{bmatrix} \quad (2.57)$$

On peut en tirer les expressions :

$$\lambda_2 = \frac{(-1)(K_2)(K_1 + K_2 - 1)(-v_{z5}\Delta_{134})}{(-1)(K_1)(K_2)(-v_{z5}\Delta_{234})} \lambda_1 \quad (2.58)$$

$$\lambda_3 = \frac{(-1)(K_1 + K_2 - 1)(K_1)(-v_{z5}\Delta_{214})}{(-1)(K_1)(K_2)(-v_{z5}\Delta_{234})} \lambda_1 \quad (2.59)$$

$$\lambda_4 = \frac{(K_1)(K_2)(K_1 + K_2 - 1)(-v_{z5}\Delta_{231})}{(-1)(K_1)(K_2)(-v_{z5}\Delta_{234})} \lambda_1 \quad (2.60)$$

$$\lambda_5 = \frac{d}{(-1)(K_1)(K_2)(-v_{z5}\Delta_{234})} \lambda_1 \quad (2.61)$$

avec  $\Delta$  définie par l'équation 2.42 et

$$d = \begin{vmatrix} K_1 v_{x2} & K_2 v_{x3} & -v_{x4} & v_{x1}(K_1 + K_2 - 1) \\ K_1 v_{y2} & K_2 v_{y3} & -v_{y4} & v_{y1}(K_1 + K_2 - 1) \\ K_1 v_{z2} & K_2 v_{z3} & -v_{z4} & v_{z1}(K_1 + K_2 - 1) \\ K_3 v_{z2} & K_4 v_{z3} & 0 & v_{z1}(K_3 + K_4 - 1) \end{vmatrix} \quad (2.62)$$

Il ne reste qu'à trouver l'expression de  $\lambda_1$ . Pour cela on utilise l'expression de la longueur du segment  $[p_1 p_2]$  :

$$\|p_1 p_2\| - L_{12} = 0 \quad (2.63)$$

Ce qui donne :

$$((\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2) \lambda_1^2 - L_{12}^2 = 0 \quad (2.64)$$

avec

$$\zeta = \frac{(K_1 + K_2 - 1)(\Delta_{134})}{(K_1)(\Delta_{234})} \quad (2.65)$$

d'où

$$\lambda_1 = \frac{L_{12}}{\sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.66)$$

d'où les solutions :

$$\lambda_2 = \frac{(K_1 + K_2 - 1)\Delta_{134} L_{12}}{K_1 \Delta_{234} \sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.67)$$

$$\lambda_3 = \frac{(K_1 + K_2 - 1)\Delta_{214} L_{12}}{K_2 \Delta_{234} \sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.68)$$

$$\lambda_4 = \frac{(K_1 + K_2 - 1)\Delta_{231} L_{12}}{-\Delta_{234} \sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.69)$$

$$\lambda_5 = \frac{d L_{12}}{K_1 K_2 (v_{z5} \Delta_{234}) \sqrt{(\zeta v_{x2} - v_{x1})^2 + (\zeta v_{y2} - v_{y1})^2 + (\zeta v_{z2} - v_{z1})^2}} \quad (2.70)$$

Nous avons donc une solution analytique du problème perspective-5-points quand les points sont coplanaires. Essayons maintenant de résoudre le même problème lorsque les points ne sont pas coplanaires.

### 2.3.4 Problème P-5-P pour des points non coplanaires

Essayons de trouver une solution analytique au problème P-5-P lorsque les points ne sont pas coplanaires. Si l'on prend  $p_1$  comme centre d'un repère et  $(\overrightarrow{p_1 p_2}, \overrightarrow{p_1 p_3}, \overrightarrow{p_1 p_4})$  une base de l'espace. On peut exprimer  $\overrightarrow{p_1 p_5}$  dans ce repère :

$$K_1 \overrightarrow{p_1 p_2} + K_2 \overrightarrow{p_1 p_3} + K_3 \overrightarrow{p_1 p_4} - \overrightarrow{p_1 p_5} = 0 \quad (2.71)$$

on obtient le système :

$$\begin{bmatrix} K_2 v_{x3} & K_3 v_{x4} & -v_{x5} \\ K_2 v_{y3} & K_3 v_{y4} & -v_{y5} \\ K_2 v_{z3} & K_3 v_{z4} & -v_{z5} \end{bmatrix} \begin{bmatrix} \lambda_3 \\ \lambda_4 \\ \lambda_5 \end{bmatrix} = - \begin{bmatrix} (-K_1 - K_2 - K_3 + 1)v_{x1} & K_1 v_{x2} \\ (-K_1 - K_2 - K_3 + 1)v_{y1} & K_1 v_{y2} \\ (-K_1 - K_2 - K_3 + 1)v_{z1} & K_1 v_{z2} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} \quad (2.72)$$

Cette fois-ci, nous ne nous pourrions pas exprimer l'ensemble des inconnues en fonction de l'une d'entre elles mais de deux. Les expressions seront donc bien plus complexes.

Il est possible de tirer les solutions du système 2.72 :

$$\lambda_3 = \frac{-AK_3\Delta_{145}\lambda_1 - BK_3\Delta_{245}\lambda_2}{-K_2K_3\Delta_{345}} \quad (2.73)$$

$$\lambda_4 = \frac{-AK_2\Delta_{315}\lambda_1 - BK_2\Delta_{325}\lambda_2}{-K_2K_3\Delta_{345}} \quad (2.74)$$

$$\lambda_5 = \frac{AK_2K_3\Delta_{341}\lambda_1 + BK_2K_3\Delta_{342}\lambda_2}{-K_2K_3\Delta_{345}} \quad (2.75)$$

( $A = K_1 + K_2 + K_3 - 1$  et  $B = -K_1$ ). Puis en posant :

$$\lambda_2 = u \lambda_1 \quad (2.76)$$

on peut dire que  $\lambda_3$  est de la forme :

$$\lambda_3 = \frac{P + Qu}{R} \lambda_1 \quad (2.77)$$

or on peut écrire :

$$\|p_1 p_2\| - L_{12} = 0 \quad (2.78)$$

$$\Leftrightarrow \lambda_2^2 - 2\vec{v}_1\vec{v}_2\lambda_2\lambda_1 + \lambda_1^2 - L_{12}^2 = 0 \quad (2.79)$$

$$\Leftrightarrow u\lambda_1^2 - 2\vec{v}_1\vec{v}_2u\lambda_1^2 + \lambda_1^2 - L_{12}^2 = 0 \quad (2.80)$$

$$\Leftrightarrow \lambda_1^2 = \frac{L_{12}^2}{1 - 2\vec{v}_1\vec{v}_2u + u^2} \quad (2.81)$$

de même :

$$\|p_1 p_3\| - L_{13} = 0 \quad (2.82)$$

$$\Leftrightarrow \lambda_3^2 - 2\vec{v}_1\vec{v}_3\lambda_3\lambda_1 + \lambda_1^2 - L_{13}^2 = 0 \quad (2.83)$$

Combinée avec 2.77

$$\left(\frac{P + Qu}{R}\right)^2 \lambda_1^2 - 2\vec{v}_1\vec{v}_3\frac{P + Qu}{R}\lambda_1^2 + \lambda_1^2 - L_{13}^2 = 0 \quad (2.84)$$

puis avec 2.81

$$\frac{L_{12}^2}{1 - 2\vec{v}_1\vec{v}_2u + u^2} \left( \left(\frac{P + Qu}{R}\right)^2 - 2\vec{v}_1\vec{v}_3\frac{P + Qu}{R} + 1 \right) - L_{13}^2 = 0 \quad (2.85)$$

De cette équation il est possible de tirer deux expressions différentes de  $u$  avec comme contrainte  $1 - 2\vec{v}_1\vec{v}_2u + u^2 \neq 0$ . Ces expressions permettent d'obtenir les expressions de  $\lambda_1$  avec l'équation 2.81 puis  $\lambda_2$  avec 2.76 et donc enfin l'ensemble des  $\lambda$  avec 2.73 à 2.75.

Nous pouvons donc obtenir une solution analytique dans le cas de 5 points non coplanaires, néanmoins cette solution est un peu lourde à manipuler, de plus comme le font remarquer Fischler et Bolles dans [FB81]<sup>[1]</sup>, cette configuration comporte des cas ambigus qui donnent nécessairement plusieurs solutions. Il faut donc un traitement supplémentaire pour choisir la bonne solution parmi toutes les solutions. A partir de 6 points, il n'y a plus d'ambiguïté et notre méthode fournit une expression bien plus agréable. Dans le cas de 6 points et plus, le système linéaire de départ permet d'obtenir une expression pour chaque inconnue en fonction d'une seule d'entre elles et non de deux comme dans le cas de 5 points.

Maintenant que nous sommes convaincus que notre méthode permet de trouver la solution analytique au problème posé, adaptons cette solution dans le cas qui nous intéresse, c'est à dire le cas de 4 points qui forment un rectangle.

## 2.4 Cas considéré

Pour notre application nous devons utiliser ce calcul pour localiser un véhicule ou un panneau muni de notre balise. Pour cela, nous avons disposé 4 disques à chaque angle. Les quatre points forment donc un rectangle. Nous avons privilégié 4 points coplanaires car c'est la solution la plus simple n'ayant pas d'ambiguïté dans la solution. La disposition en rectangle permet en plus de simplifier la solution générale du problème P-4-P pour des points coplanaires. Exprimons la solution spécifique pour ce cas.

---

[1] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communication of the ACM*, 24(6):381–395, 1981.



### 2.4.1 Expression de la solution

Nous disposons d'un rectangle, nous pouvons donc écrire :

$$\overrightarrow{p_1 p_2} + \overrightarrow{p_3 p_4} = 0 \quad (2.86)$$

ce qui peut se ré-écrire :

$$\begin{bmatrix} \lambda_2 v_{x2} - \lambda_1 v_{x1} + \lambda_4 v_{x4} - \lambda_3 v_{x3} \\ \lambda_2 v_{y2} - \lambda_1 v_{y1} + \lambda_4 v_{y4} - \lambda_3 v_{y3} \\ \lambda_2 v_{z2} - \lambda_1 v_{z1} + \lambda_4 v_{z4} - \lambda_3 v_{z3} \end{bmatrix} = 0 \quad (2.87)$$

ce qui donne :

$$\begin{bmatrix} v_{x2} & -v_{x3} & v_{x4} \\ v_{y2} & -v_{y3} & v_{y4} \\ v_{z2} & -v_{z3} & v_{z4} \end{bmatrix} \begin{bmatrix} \lambda_2 \\ \lambda_3 \\ \lambda_4 \end{bmatrix} = \begin{bmatrix} \lambda_1 v_{x1} \\ \lambda_1 v_{y1} \\ \lambda_1 v_{z1} \end{bmatrix} \quad (2.88)$$

Nous pouvons exprimer chaque  $\lambda$  en fonction de  $\lambda_1$ .

$$\lambda_2 = \frac{\Delta_{134}}{\Delta_{234}} \lambda_1 \quad (2.89)$$

$$\lambda_3 = -\frac{\Delta_{214}}{\Delta_{234}} \lambda_1 = \frac{\Delta_{124}}{\Delta_{234}} \lambda_1 \quad (2.90)$$

$$\lambda_4 = \frac{\Delta_{231}}{\Delta_{234}} \lambda_1 \quad (2.91)$$

Il ne reste qu'à trouver une expression de  $\lambda_1$ . Pour cela utilisons l'expression :

$$\|\overrightarrow{p_1 p_2}\| - L_{12} = 0 \quad (2.92)$$

$$\Leftrightarrow (\lambda_2 \vec{v}_2 - \lambda_1 \vec{v}_1)^2 - L_{12}^2 = 0 \quad (2.93)$$

$$\Leftrightarrow \lambda_2^2 v_2^2 + \lambda_1^2 v_1^2 - 2\vec{v}_1 \vec{v}_2 \lambda_1 \lambda_2 - L_{12}^2 = 0 \quad (2.94)$$

$$\Leftrightarrow \lambda_2^2 + \lambda_1^2 - 2\vec{v}_1 \vec{v}_2 \lambda_1 \lambda_2 - L_{12}^2 = 0 \quad (2.95)$$

$$(\|\vec{v}_i\| = 1) \quad (2.96)$$

Combinée avec l'expression de  $\lambda_2$  (équation 2.89) :

$$\lambda_1^2 \left( \left( \frac{\Delta_{134}}{\Delta_{234}} \right)^2 - 2\vec{v}_1 \vec{v}_2 \frac{\Delta_{134}}{\Delta_{234}} + 1 \right) - L_{12}^2 = 0 \quad (2.97)$$

d'où

$$\lambda_1 = \frac{L_{12}}{\sqrt{\left(\frac{\Delta_{134}}{\Delta_{234}}\right)^2 - 2\vec{v}_1\vec{v}_2\frac{\Delta_{134}}{\Delta_{234}} + 1}} \quad (2.98)$$

En combinant avec les équations 2.89 à 2.91 on trouve :

$$\lambda_2 = \frac{\Delta_{134}}{\Delta_{234}} \frac{L_{12}}{\sqrt{\left(\frac{\Delta_{134}}{\Delta_{234}}\right)^2 - 2\vec{v}_1\vec{v}_2\frac{\Delta_{134}}{\Delta_{234}} + 1}} \quad (2.99)$$

$$\lambda_3 = -\frac{\Delta_{214}}{\Delta_{234}} \frac{L_{12}}{\sqrt{\left(\frac{\Delta_{134}}{\Delta_{234}}\right)^2 - 2\vec{v}_1\vec{v}_2\frac{\Delta_{134}}{\Delta_{234}} + 1}} \quad (2.100)$$

$$\lambda_4 = \frac{\Delta_{231}}{\Delta_{234}} \frac{L_{12}}{\sqrt{\left(\frac{\Delta_{134}}{\Delta_{234}}\right)^2 - 2\vec{v}_1\vec{v}_2\frac{\Delta_{134}}{\Delta_{234}} + 1}} \quad (2.101)$$

Nous avons donc une solution analytique simplifiée au problème P-4-P dans le cas où les points forment un parallélogramme (pour nous un rectangle). Nous pouvons donc, à l'aide de nos 4 disques inclus à notre balise, retrouver, à partir d'une image unique, la position de la balise dans l'espace.

### 2.4.2 Influence de l'erreur de localisation des points sur l'image

Les procédés de localisation peuvent vite être mis en défaut. En effet, il est difficile d'obtenir une bonne précision : une erreur minime de positionnement d'un point de contrôle peut entraîner une erreur très importante dans le positionnement de ce point (mais pas uniquement) dans l'espace. En admettant que la distance  $d$  entre le centre de notre capteur et le point de contrôle soit correcte mais qu'il y ait une erreur de positionnement de la projection du point de contrôle sur le capteur d'image qui entraîne un angle  $\theta$ , on localisera le point dans l'espace avec une erreur  $e = 2d \sin(\frac{\theta}{2})$ . Si le point est situé à 30 mètres du capteur, une erreur de 2 pixels entraîne une erreur approximative de 10 cm dans l'espace avec une focale de 4 mm et une taille de pixel de  $6.25 \mu\text{m}$ . Etudions donc l'impact des différents paramètres sur l'erreur de positionnement commise.

Idéalement, le point  $p_1$  devrait se trouver en  $\lambda_1\vec{v}_1$  mais, lors du calcul, nous com-

mettons une erreur de  $\lambda_e \vec{v}_e$ . Nous avons donc

$$\vec{E} = \lambda_1 \vec{v}_1 + \lambda_e \vec{v}_e \quad (2.102)$$

ce qui donne :

$$\vec{E} = \lambda_e (\vec{v}_1 + \vec{e}_1) - \lambda_1 \vec{v}_1 \quad (2.103)$$

$$\Leftrightarrow \vec{E} = (\lambda_e - \lambda_1) \vec{v}_1 + \lambda_1 \vec{e}_1 \quad (2.104)$$

$$\Leftrightarrow \vec{E} = \delta_{\lambda_1} \vec{v}_1 + \lambda_1 \vec{e}_1 \quad (2.105)$$

d'où

$$\|\vec{E}\|^2 = \delta_{\lambda_1}^2 + 2\lambda_1 \delta_{\lambda_1} \vec{v}_1 \vec{e}_1 + \lambda_1^2 \|\vec{e}_1\|^2 \quad (2.106)$$

Il est possible de trouver une expression de  $\delta_{\lambda_1}$ . Pour cela posons :

$$d_{ijk} = \Delta_{ijk} + e (|\vec{e}_i \vec{v}_j \vec{v}_k| + |\vec{v}_i \vec{e}_j \vec{v}_k| + |\vec{v}_i \vec{v}_j \vec{e}_k|) + O(e^2) \quad (2.107)$$

(avec  $e$  un coefficient proche de 0). Nous obtenons :

$$\delta_{\lambda_1} = \frac{L_{12}}{\sqrt{\left(\frac{d_{134}}{d_{234}}\right)^2 - 2\vec{v}_1 \vec{v}_2 \frac{d_{134}}{d_{234}} + 1}} - \frac{L_{12}}{\sqrt{\left(\frac{\Delta_{134}}{\Delta_{234}}\right)^2 - 2\vec{v}_1 \vec{v}_2 \frac{\Delta_{134}}{\Delta_{234}} + 1}} \quad (2.108)$$

$$\delta_{\lambda_1} = \frac{L_{14}(-\Delta_{134}^2 d_{234} + \Delta_{134} d_{134} \Delta_{234} - \vec{v}_1 \vec{v}_2 \Delta_{234}^2 d_{134} + \vec{v}_1 \vec{v}_2 \Delta_{234} \Delta_{134} d_{234})e}{\Delta_{234} \sqrt{\left(\frac{\Delta_{134}}{\Delta_{234}}\right)^2 - 2\vec{v}_1 \vec{v}_2 \frac{\Delta_{134}}{\Delta_{234}} + 1} (-\Delta_{134}^2 + 2\vec{v}_1 \vec{v}_2 \Delta_{134} \Delta_{234} - \Delta_{234}^2)} + O(e^2) \quad (2.109)$$

A l'aide de l'équation 2.97, on peut effectuer deux substitutions :

$$\delta_{\lambda_1} = \frac{\lambda_1^3}{L_{14}^2} \frac{e(\Delta_{134}^2 d_{234} - \Delta_{134} d_{134} \Delta_{234} + \vec{v}_1 \vec{v}_2 \Delta_{234}^2 d_{134} - \vec{v}_1 \vec{v}_2 \Delta_{234} \Delta_{134} d_{234})}{\Delta_{234}^3} \quad (2.110)$$

Pour alléger l'expression nous noterons :

$$\delta_{\lambda_1} = \frac{\lambda_1^3}{L_{14}^2} \frac{e(K_{err})}{\Delta_{234}^3} \quad (2.111)$$

Ré-introduit dans l'équation 2.106 :

$$\|\vec{E}\|^2 = \left( \frac{\lambda_1^3}{L_{14}^2} \frac{e(K_{err})}{\Delta_{234}^3} \right)^2 + 2\lambda_1 \left( \frac{\lambda_1^3}{L_{14}^2} \frac{e(K_{err})}{\Delta_{234}^3} \right) \vec{v}_1 \vec{e}_1 + \lambda_1^2 \|\vec{e}_1\|^2 \quad (2.112)$$

Cette expression met en évidence que, plus la distance entre le capteur et le point de contrôle est grande, plus l'erreur augmente, ce qui est prévisible. On n'évalue ici que l'erreur sur la distance mais, on a vu, plus haut, que l'erreur en angle augmente aussi considérablement.

Par ailleurs cette expression montre bien que, plus  $L_{14}$  est grand, plus l'erreur diminue. Nous avons donc intérêt à choisir la distance la plus grande parmi les segments formés par nos points dans l'espace. Cela peut aisément se comprendre : si une erreur de 1 pixel sur un segment de 10 cm donne une erreur relative  $e$  dans l'espace, cette même erreur de 1 pixel donnera une erreur relative de  $e/10$  pour un segment de 1 m.

Enfin cette expression met en évidence l'influence des déterminants : plus la valeur du déterminant est faible, plus l'erreur est importante (on a au dénominateur  $\Delta_{234}^3$ , équation 2.98). Même si nous n'avons pas de solutions dégénérées car ce déterminant ne sera jamais nul (les vecteurs directeurs ne seront jamais colinéaires), il est possible que la stabilité numérique de la solution soit dégradée si ce déterminant tend vers 0. Pour cette raison, il serait intéressant de trouver toutes les solutions analytiques du cas considéré, et de n'utiliser que celle qui a au dénominateur le déterminant le plus grand. Le second déterminant de l'expression de la solution (ici  $\Delta_{123}$ ) intervient aussi, dans une mesure moins importante. Là encore, plus le déterminant sera grand plus l'erreur sera petite.

Le même raisonnement peut être conduit pour les expressions des autres  $\lambda$  à ceci près qu'il dépendent tous de l'expression de  $\lambda_1$  donc, l'erreur de l'expression de  $\lambda_1$  se propage sur les expressions des autres  $\lambda$ .

Par ailleurs, nous avons pu mettre en évidence que les solutions sont de la forme :

$$\lambda_{estimé} = \lambda_{idéal} + e(erreur) \quad (2.113)$$

avec  $e$  un terme d'erreur proche de 0, il s'agit donc d'un terme d'erreur additif.

Nous avons donc pu mettre en évidence les termes ayant une influence sur l'erreur de localisation. Voyons maintenant, les résultats obtenus par cette méthode.

## 2.5 Résultats expérimentaux

Les premiers tests réalisés donnaient des résultats assez décevants. Après plusieurs tentatives, nous avons réussi à obtenir des résultats satisfaisants en corrigeant les distorsions optiques de l'image à l'aide de l'algorithme présenté au chapitre 1 avant de procéder à la localisation. Finalement nous parvenons à localiser précisément des obstacles marqués (figure 2.3). La distance retournée par le programme est correcte. Nous parvenons aussi à retrouver le vecteur normal au plan de chaque mur (en violet en haut à gauche de la figure 2.3).

## 2.6 Conclusion

Nous avons présenté une démarche simple et nouvelle pour obtenir une solution strictement analytique au problème P-N-P. Nous obtenons des solutions analytiques pour  $N > 3$  si les points sont coplanaires, et  $N > 4$  si les points sont non coplanaires. Pour le cas  $N = 3$  notre méthode permet de trouver une solution seulement si les trois points sont alignés. Les solutions obtenues sont bien adaptées à un traitement en temps réel car le temps de calcul pour évaluer numériquement les solutions est très faible et constant. De plus notre solution sera plus stable numériquement que les précédentes solutions analytiques proposées car elle repose uniquement sur un polynôme de degré 2 (sauf pour le cas de 5 points non coplanaires).

Nous avons illustré cette méthode sur plusieurs exemples. Ensuite, nous nous sommes penchés sur le cas de 4 points formant un parallélogramme : c'est le cas qui nous intéresse pour notre application. Cette configuration est la plus simple fournissant, sans ambiguïté, une solution unique. Sur cet exemple nous avons mis en évidence, analytiquement, l'influence de chaque paramètre. Enfin nous avons montré des résultats à la fois sur des images de synthèse et des images réelles.

Ce problème est posé depuis longtemps mais les approches proposées n'étaient pas pleinement satisfaisantes.

Les solutions obtenues par notre approche peuvent être adaptées à des capteurs non linéaires, nous verrons plus loin comment.

Si le nombre de points de contrôle est trop élevé, il est possible, plutôt que de cal-



FIG. 2.3: Résultat de la résolution du problème perspective-N-points sur des objets dont la taille est connue par avance : les objets sont localisés précisément dans l'espace.

culer une solution prenant en compte tous les points, de calculer plusieurs solutions avec plusieurs sous-ensembles de points et de fusionner l'ensemble des résultats. Le résultat numérique en sera d'autant plus précis.

Par ailleurs, ces solutions peuvent parfois être instables, et nous résolvons parfois des systèmes surdéterminés en supprimant des équations ce qui n'est pas toujours acceptable numériquement. Il est donc parfois préférable d'utiliser une méthode

itérative. Dans ce cas, la solution proposée ici servira à trouver les valeurs d'initialisation du processus. L'initialisation étant déjà très proche de la solution finale, le processus de minimisation sera plus rapide et aura plus de chance de converger vers la solution et pas vers un minimum local.





Deuxième partie

Etude des Capteurs  
Catadioptriques Panoramiques



# Chapitre 3

## Présentation et Calibrage des Capteurs Catadioptriques Panoramiques

### Sommaire

---

<b>3.1</b>	<b>Introduction</b>	<b>73</b>
<b>3.2</b>	<b>Présentation des capteurs catadioptriques panoramiques</b>	<b>73</b>
<b>3.3</b>	<b>Etat de l'art</b>	<b>75</b>
3.3.1	Méthodes proposées par Sing Bing Kang	76
3.3.2	Méthode de Mičušík et Pajdla	79
3.3.3	Méthodes proposées par Christopher Geyer et Kostas Daniilidis	79
3.3.4	Méthode de Xianghua Ying et Zhanyi Hu	82
3.3.5	Méthode de Cyril Cauchois	83
3.3.6	Méthode de l'équipe de Carnegie Mellon University	84
3.3.7	Bilan	85
<b>3.4</b>	<b>Notre Calibrage</b>	<b>86</b>
3.4.1	Rappel de la méthode des deux plans	87
3.4.2	Principe du calibrage	89
3.4.3	Récapitulatif	92
3.4.4	Extraction des contours du miroir	92

3.4.5	Validation et résultats . . . . .	94
3.4.5.1	Validation sur images de synthèse . . . . .	95
3.4.5.2	Validation sur images réelles . . . . .	96
3.4.6	Amélioration du calibrage par des miroirs adaptés . . .	98
<b>3.5</b>	<b>Conclusion . . . . .</b>	<b>104</b>

---

## 3.1 Introduction

Un capteur catadioptrique est l'association d'une caméra classique et d'un miroir. Selon le type de miroir utilisé, il est possible d'obtenir différentes propriétés. La vision panoramique est une de ces propriétés. Elle est possible grâce à l'utilisation d'un miroir de révolution. Cette configuration est particulièrement bien adaptée à la robotique mobile ou à la vidéo surveillance. Dans ces types d'applications, l'élargissement du champ de vision est un atout majeur. Il existe d'autres moyens pour obtenir une vision panoramique : l'utilisation d'une lentille *fish-eye* [ZSE86, HxTP<sup>+</sup>97]<sup>[1,2]</sup> ou d'une caméra tournante [Sma03]<sup>[3]</sup>. Contrairement aux dispositifs à base de caméra tournante, lors de l'utilisation d'un miroir de révolution, l'accroissement du champ de vision ne se fait pas au détriment de la vitesse d'acquisition. Cependant, le champ de vision ayant été élargi sans pour autant que la taille du capteur d'image ne soit particulièrement agrandie, la résolution de ce type de capteur est réduite.

Pour ce capteur, comme pour beaucoup d'autres, le calibrage est une étape essentielle. Nous verrons, dans ce chapitre, après un bref rappel de la géométrie de ce capteur, notre méthode de calibrage.

## 3.2 Présentation des capteurs catadioptriques panoramiques

Les capteurs panoramiques catadioptriques, sont composés d'un miroir de révolution placé face à une caméra (figure 3.1). Ils permettent l'obtention d'un champ de vision à  $360^\circ$ , à la cadence d'une caméra classique (figure 3.2). Ces capteurs peuvent être classés en deux groupes selon qu'ils respectent ou non la contrainte du foyer unique. Cette contrainte n'est respectée que par quelques profils de mi-

- 
- [1] Cao Z, Oh S, and Hall E. Dynamic omnidirectional vision for mobile robots. *Journal of Robotics Systems*, 3(1):5–17, 1986.
  - [2] J. Hong, x. Tan, B. Pinette, R. Weiss, and E. M. Wisserman. Image based homing. *Proc. of Conference on Robotics and Automation*, 1:181–187, 1997.
  - [3] Laurent Smadja. *Génération d'environnements 3D denses à partir d'images panoramiques cylindriques*. PhD thesis, 2003.

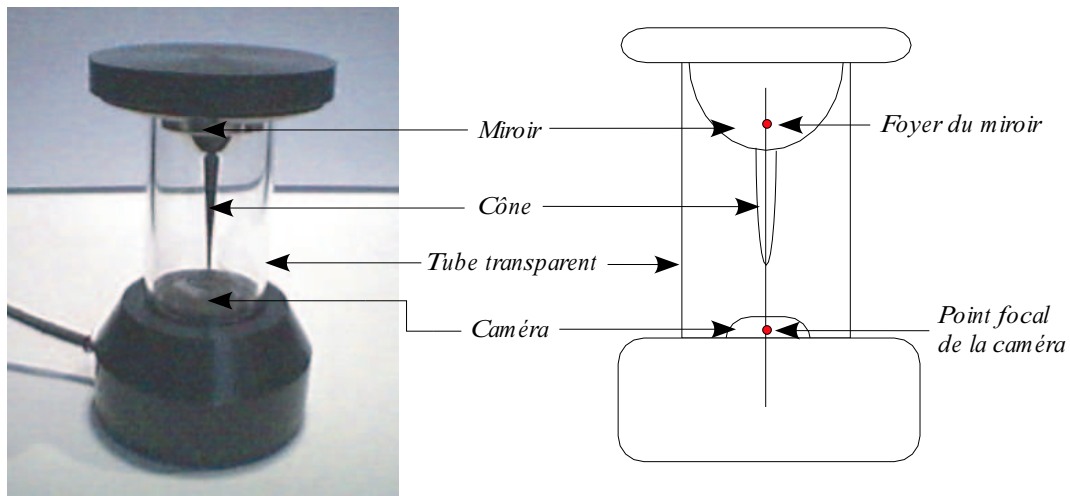


FIG. 3.1: Capteur catadioptrique panoramique : Un miroir de révolution est placé face à une caméra.

roirs [BN99]<sup>[1]</sup>. De plus elle impose un positionnement parfait du miroir face à la caméra. Elle assure au capteur d'avoir un foyer de projection et un seul, ce qui la rend particulièrement intéressante car, grâce à son foyer de projection unique, elle facilite, entre autres, l'écriture de la réflexion des rayons lumineux sur le miroir (les seuls rayons lumineux qui atteignent la caméra sont ceux qui vont en direction du foyer du miroir (figure 3.3)). Par ailleurs, il est difficile sans foyer de projection unique d'écrire la contrainte épipolaire d'une paire stéréo faisant intervenir un tel capteur.

Parmi les capteurs catadioptriques panoramiques, il existe une configuration particulière très répandue : les capteurs *paracatadioptriques*. Ils sont composés d'un miroir paraboloidique et d'une caméra munie d'une lentille télécentrique. Celle-ci permet une projection orthogonale (à la place d'une projection perspective) qui simplifie la plupart des calculs. Ces capteurs respectent la contrainte du foyer unique mais la lentille télécentrique en augmente le poids, la taille et, la complexité. Le capteur, utilisé pour réaliser nos tests et mettre en place nos algorithmes, nous a été donné par le Professeur Iroshi Ishiguro de l'université de Wakayama (Japon) (figure 3.1). Il possède un miroir à profil hyperbolique, ainsi qu'une petite aiguille au centre du miroir qui permet d'éviter certaines réflexions parasites dues au tube

[1] Simon Baker and Shree K. Nayar. A theory of single-viewpoint catadioptric image formation. *International Journal of Computer Vision*, 35(2):175 – 196, 1999.



FIG. 3.2: Image prise par un capteur catadioptrique panoramique d'une situation routière.

transparent.

Pour l'ensemble des méthodes présentées, nous définissons trois repères ortho-normés (figure 3.4) :

- un repère caméra, centré sur le foyer de la caméra et dont l'axe des  $Z$  est confondu avec l'axe optique,
- un repère miroir, centré sur le foyer du miroir et dont l'axe des  $z$  est orienté vers le haut,
- un repère monde quelconque.

### 3.3 Etat de l'art

Dans les systèmes de vision, le calibrage est une brique incontournable si l'on veut une correspondance espace-image. Les méthodes de calibrage pour les caméras classiques ne manquent pas. Il n'en est pas de même pour les capteurs catadioptriques panoramiques. Passons en revue les méthodes déjà existantes.

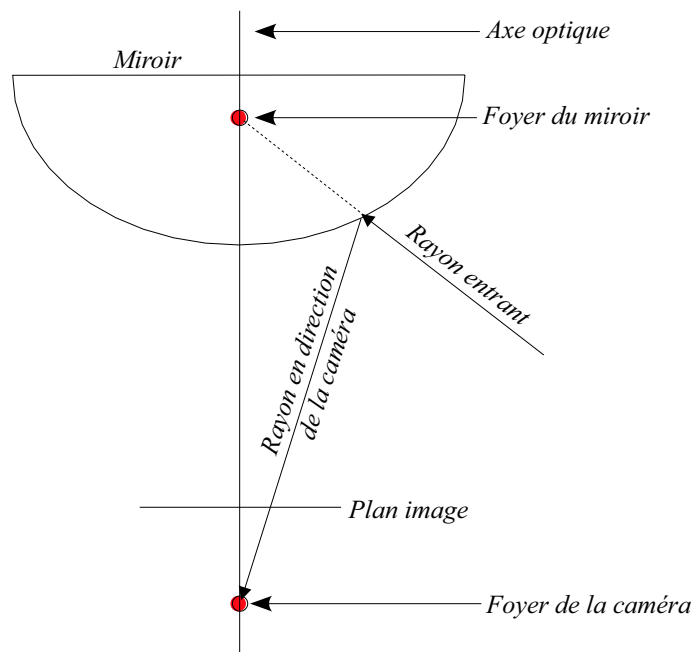


FIG. 3.3: Réflexion des rayons lumineux sur un capteur catadioptrique respectant la contrainte du foyer unique : seuls les rayons entrants dans le capteur en direction du foyer du miroir se réfléchissent en direction du foyer de la caméra.

### 3.3.1 Méthodes proposées par Sing Bing Kang

Kang propose deux méthodes pour calibrer les capteurs catadioptriques composés d'un miroir paraboloidique et d'une lentille télécentrique, l'une est appelée *direct circle-based self-calibration method* et l'autre *self-calibration method* [Kan00]<sup>[1]</sup>.

La première méthode permet de déterminer le centre de l'image et le paramètre de la parabole. Pour trouver le centre de l'image, il extrait le cercle qui représente la bordure du miroir et en déduit, de son centre, la position du centre de l'image. Pour déterminer le paramètre de la parabole, il part de l'équation du profil du

---

[1] Sing Bing Kang. Catadioptric self-calibration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 201–207, 2000.



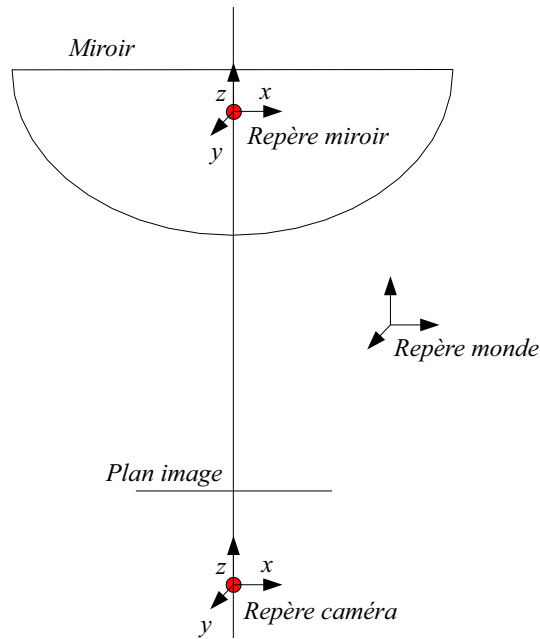


FIG. 3.4: Définitions des trois repères liés au capteur et à l'espace : le repère caméra, le repère miroir et le repère monde.

miroir [Nay97]<sup>[2]</sup> :

$$z = \frac{h^2 - r^2}{2h} \quad (3.1)$$

avec  $(r, z)$  les coordonnées d'un point du miroir dans le repère centré sur le foyer du miroir et  $h$  le paramètre de la parabole (figure 3.5). Comme on a

$$\tan \theta = \frac{r}{z} \quad (3.2)$$

on a

$$\tan \theta = \frac{2rh}{h^2 - r^2} \quad (3.3)$$

donc

$$h = r \frac{1 + \sqrt{1 + \tan^2 \theta}}{\tan \theta} = r \cot \frac{\theta}{2} \quad (3.4)$$

en prenant un couple  $(r, \theta)$  connu, par exemple  $(r_{max}, \theta_{max})$ , il peut en tirer le paramètre  $h$  de la parabole.

---

[2] S. K. Nayar. Catadioptric omnidirectional camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 1997.

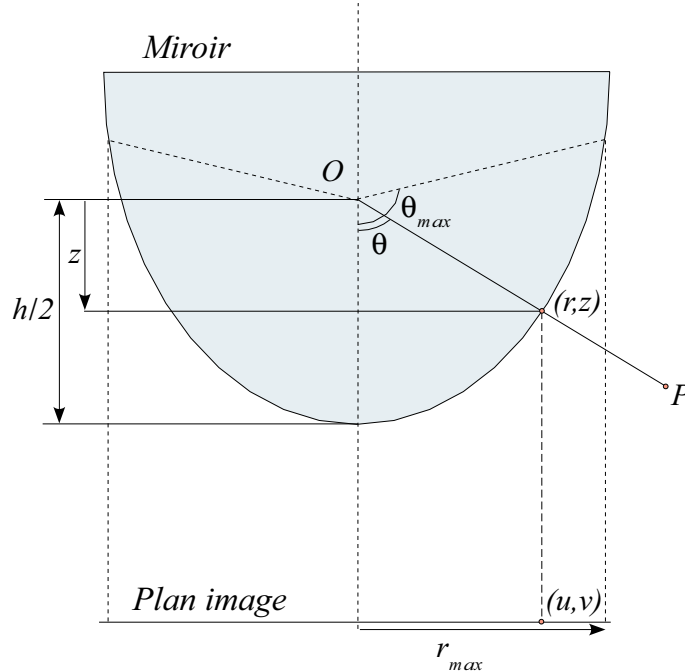


FIG. 3.5: Réflexion d'un point  $P$  de l'espace sur le plan image.

La seconde méthode permet, non seulement, d'estimer le centre de l'image et le paramètre  $h$  de la parabole, mais aussi le *skew*  $s$  (rapport de la largeur du pixel sur la hauteur du pixel) ainsi que l'*aspect ratio*  $\alpha$  (défaut d'orientation du capteur d'image par rapport à l'axe optique). Pour cela, il écrit la contrainte épipolaire  $q^t E q'^t$  sur des couples de points  $q$  et  $q'$  en fonction des paramètres du capteur. Ensuite, caméra en mouvement, il suit un certain nombre de points caractéristiques sur plusieurs images. Il peut donc obtenir plusieurs couples  $(q, q')$  sur lesquels il peut appliquer la contrainte épipolaire et minimiser les expressions afin d'estimer les paramètres du capteur. Pour effectuer la minimisation, il teste deux critères :  $(q^t E q')^2$  et  $d^2(q^t E q'') + d^2(q''^t E q')$  avec  $d(m, n)$  la distance image entre le point  $m$  et la courbe épipolaire déduite de  $n$ . (Le second critère étant plus robuste). Le processus de minimisation est initialisé avec les résultats de sa première méthode de calibrage et avec 0 et 1 pour  $\alpha$  et  $s$  respectivement. Il estime donc le paramètre de la parabole, le centre de l'image, l'*aspect ratio* et le *skew*.

### 3.3.2 Méthode de Mičušík et Pajdla

Ils proposent dans [MP04]<sup>[1]</sup> une amélioration de la méthode de Kang pour estimer le paramètre de la parabole. Deux différences sont à noter. La première est que les expressions de la géométrie épipolaire sont analytiques, ce qui évite le recours à des méthodes itératives. La seconde est l'utilisation du RANSAC, qui permet d'éliminer des points mal appariés.

Le centre de l'image est assimilé au centre du champ de vu. Le *skew* ainsi que l'*aspect ratio* sont retrouvés en estimant la bordure du miroir (ellipse) et en calculant la transformation qui modifie cette ellipse en un cercle.

Par ailleurs, ils proposent aussi d'étendre leur méthode aux miroirs hyperboloïdiques. Cette méthode, comme celle de Kang, nécessite un déplacement entre les deux prises de vue et impose que la scène ne soit pas modifiée entre les deux prises de vue.

### 3.3.3 Méthodes proposées par Christopher Geyer et Kostas Daniilidis

Ils proposent deux méthodes pour calibrer les capteurs catadioptriques composés d'un miroir paraboloidique et d'une lentille télécentrique [GD99, GD02]<sup>[2,3]</sup>. Ces méthodes reposent sur la propriété suivante : avec ce type capteur, une ligne de l'espace donne un cercle sur l'image.

Dans la première méthode, ils utilisent le fait qu'un ensemble de lignes parallèles dans l'espace donne un ensemble de cercles dans l'image. Ces cercles ont leurs centres alignés et se coupent en deux points  $\pi_0^i$  et  $\pi_1^i$ . Ces deux points définissent une droite. Si l'on prend plusieurs ensembles de droites parallèles, on

- 
- [1] Branislav Mičušík and Tomáš Pajdla. Para-catadioptric camera auto-calibration from epipolar geometry. *Proceedings of the Asian Conference on Computer Vision*, January 2004.
  - [2] Christopher Geyer and Kostas Daniilidis. Catadioptric camera calibration. In *Proceedings of the 7th International Conference on Computer Vision, Kerkyra*, pages 398–404, 1999.
  - [3] Christopher Geyer and Kostas Daniilidis. Paracatadioptric camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):687–695, 2002.

obtient plusieurs couples de points qui donnent un ensemble de droites. Ces droites sont concourantes et leur intersection donne le centre de l'image (figure 3.6). Par ailleurs, l'ensemble des points  $\pi_0^i$  est porté par un cercle. Si on note  $r$  le rayon de ce cercle et  $d$  la distance entre le centre de l'image et le centre du cercle on peut calculer la focale  $f$  de la parabole :

$$f = \frac{1}{2} \sqrt{(r+d)(r-d)} \quad (3.5)$$

Dans le cas où l'*aspect ratio* n'est pas égal à 1, une ligne de l'espace ne donne plus un cercle mais une ellipse dans l'image. Pour se ramener au cas précédent et pouvoir appliquer leur algorithme de calibrage, Geyer et Daniilidis estiment l'*aspect ratio* d'après le rapport des deux demi-axes de l'ellipse puis ils recalculent une image corrigée ayant un *aspect ratio* de 1 à partir de l'image originale. Ils affirment que cette méthode peut être facilement adaptée aux capteurs munis d'un miroir hyperboloïdique et d'une caméra perspective.

Ils retrouvent donc le centre de l'image, la focale de la parabole et, éventuellement, l'*aspect ratio* qu'ils ramènent à 1.

Dans la seconde méthode, à partir de 3 lignes quelconques de l'espace, dans une configuration non singulière (c-a-d non concourantes, non parallèles à l'axe optique), ils retrouvent, si l'image n'a pas de *skew* ( $\beta = 0$ ), le centre de l'image  $\xi$ , la focale du miroir  $f_0$  et l'*aspect ratio*  $a$ , et en donnent une expression analytique. Dans le cas contraire, seul le centre de l'image et la focale ont une expression. L'*aspect ratio*, et, au besoin, le *skew*, sont estimés en retrouvant la transformation affine qui transforme les cercles théoriques des lignes images en ellipses.

L'expression analytique provient du fait que chaque ligne de l'espace donne un arc de cercle dans l'image. Cet arc peut être extrapolé en une sphère et cette sphère passe par les points  $2f$  et  $-2f$ . Donc avec un minimum de 3 lignes espaces, on obtient 3 sphères. Ces sphères se coupent en  $-2f$  et  $2f$  on peut donc trouver le centre et la focale du miroir à partir de ces sphères (figure 3.7). L'intersection des sphères permet de tirer les expressions :

$$f_0^2 = \frac{1}{4n} \sum_{i=1}^n (r_i^2 - (\xi - c_i)^t (\xi - c_i)) \quad (3.6)$$

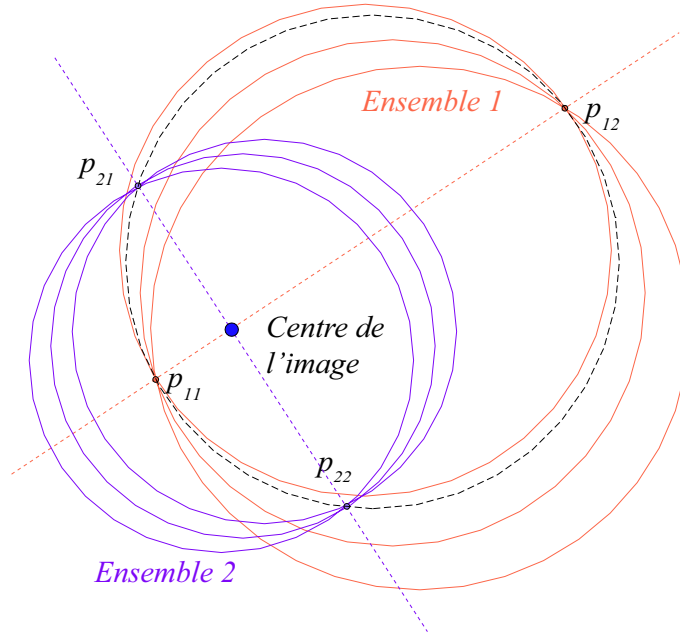


FIG. 3.6: Projection de deux ensembles de lignes parallèles de l'espace en deux ensembles de cercles sur l'image : chaque ensemble de cercles donne deux points et les droites formées par ces points convergent vers le centre de l'image.

$$\xi_0 = -\frac{1}{2}A^{-1}b \quad (3.7)$$

avec

$$A = \sum_{i,j,k}^n (c_k - c_i)^t (c_j - c_i) \quad (3.8)$$

et

$$b = \sum_{i,j,k}^n (c_i^t c_i - r_i^2 - c_j^t c_j + r_j^2) (c_k - c_i) \quad (3.9)$$

(avec  $c_i$  et  $r_i$  centre et rayon respectivement des cercles issus des lignes espaces). En résumé, si le *skew* est nul, ils retrouvent analytiquement le centre de l'image, la focale de la parabole ainsi que l'*aspect ratio*. Sinon seul le centre de l'image et la focale de la parabole ont une expression, l'*aspect ratio* et le *skew* sont estimés numériquement.

Si les algorithmes présentés par Geyer et Daniilidis sont élégants, ils reposent malheureusement sur une propriété non démontrée. En effet, ses auteurs démontrent

qu'une ligne de l'espace donne, dans le cas général, un cercle (ou une ellipse) mais ils utilisent la réciproque en extrayant des cercles de l'image et en supposant que ceux-ci proviennent de ligne de l'espace.

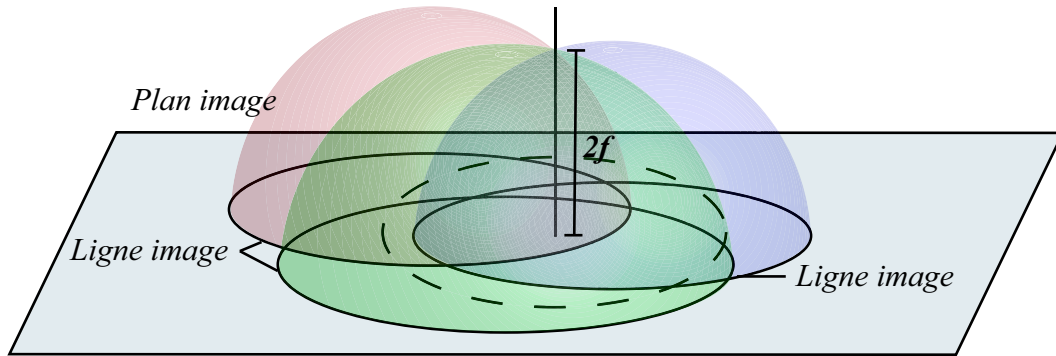


FIG. 3.7: Projection de 3 lignes espace en trois arcs de cercle puis extrapolation des arcs en sphère pour en tirer  $f$  et le centre.

### 3.3.4 Méthode de Xianghua Ying et Zhanyi Hu

Ils proposent dans [YH03]<sup>[1]</sup> une généralisation de la méthode de Geyer et Kotas, en utilisant soit un minimum de 2 lignes soit un minimum de 3 sphères pour calibrer n'importe quel type de capteur respectant la contrainte du foyer unique. Les auteurs privilégient le calibrage à base de sphère car il est bien plus précis que celui à base de lignes : les lignes donnent en général des portions de coniques alors que les sphères donnent des coniques complètes ce qui conduit à une extraction de coniques plus précise. Toutefois, il faut modérer cette approche car il est bien plus fréquent d'avoir des lignes dans des scènes issues de situations courantes plutôt que des sphères (à moins d'avoir la possibilité d'en ajouter dans l'environnement).

---

[1] Xianghua Ying and Zhanyi Hu. Catadioptric camera calibration using geometric invariants. In *Proceedings of the 9th International Conference on Computer Vision, Nice, 2003*.

### 3.3.5 Méthode de Cyril Cauchois

Il propose une méthode, qu'il modifiera avantageusement par la suite, pour calibrer un capteur muni d'un miroir conique (Il travaille sur le capteur SYCLOP : Système Conique pour la Localisation et la Perception)[Cau01]<sup>[2]</sup>.

La première approche se fait en deux étapes : la première consiste à ajouter une mire à 2 plans (figure 3.8) qui se fixe sur le miroir et à calibrer la caméra plan. Il estime les paramètres intrinsèques de la caméra comprenant un terme de distorsion ainsi que la transformation repère caméra/repère miroir. Dans un deuxième temps, l'utilisation d'une seconde mire, prise à plusieurs positions différentes autour du système permet, par minimisation, d'affiner la précision des paramètres estimés à l'étape précédente et d'évaluer la transformation repère miroir/repère monde. La mire extérieure n'étant pas adaptée au profil du miroir elle est difficilement utilisable, une grande partie des motifs sont inexploitable sur l'image ce qui a conduit à un changement de technique.

Dans la seconde méthode, la première étape est supprimée, il ne reste que la seconde étape mais avec une mire plus adaptée au profil du miroir. Celle-ci s'emboîte au-dessus du cône (figure 3.9). Elle englobe correctement toute la surface visible et comme les motifs de la mire sont plus adaptés, les points de contrôle sont extraits avec une bien meilleure précision. Il écrit la relation qui lie les points de l'espace et ceux de l'image en fonction des paramètres du capteur puis minimise cette expression connaissant des couples de points (portés par la mire). Il retrouve ainsi tout l'ensemble des paramètres du système (intrinsèques et extrinsèques).

---

[2] Cyril Cauchois. *Modélisation et Calibration du Capteur Omnidirectionnel SYCLOP : Application à la Localisation Absolue en Milieu Structuré*. PhD thesis, 2001.



FIG. 3.8: Mire de calibration à deux plans qui se fixe sur le capteur SYCLOP.

### 3.3.6 Méthode de l'équipe de Carnegie Mellon University

Ils proposent dans [SMKS01]<sup>[1]</sup> une méthode pour calibrer les capteurs catadioptriques. Cette méthode permet de retrouver la transformation qui permet de passer du repère miroir au repère caméra (et aussi du repère miroir à un repère monde). Mais la technique exposée ne permet pas d'estimer les paramètres intrinsèques de la caméra, par conséquent il faut préalablement démonter le dispositif pour calibrer la caméra indépendamment.

Pour estimer les changements de repère, ils minimisent l'erreur

$$\chi^2 = \sum_{i=1}^n \|x_i - \Pi(R_c p_i + t_c)\|^2 \quad (3.10)$$

entre la position observée  $x_i$  des points de calibration  $p_i$  et la position calculée de ces mêmes points à l'aide des paramètres.  $R_c$  et  $t_c$  sont la transformation qui fait passer du repère monde au repère capteur et  $\Pi$  est la projection des points

---

[1] D. Strelow, J. Mishler, D. Koes, and S. Singh. Precise omnidirectional camera calibration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 689–694, 2001.



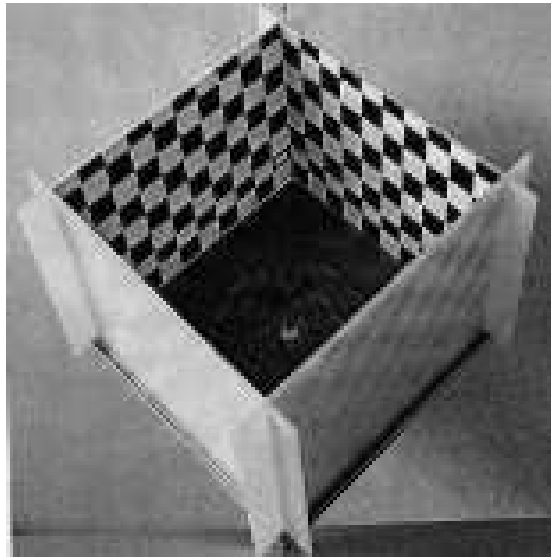


FIG. 3.9: Mire de calibration qui se fixe sur le cône du capteur SYCLOP.

espaces sur l'image en fonction des paramètres recherchés (Il est sûrement possible d'améliorer cette minimisation en utilisant un critère différent étant donné qu'un déplacement du miroir entraînera un déplacement non rigide de l'ensemble des points de calibration sur l'image).

### 3.3.7 Bilan

Peu de méthodes existent pour calibrer des capteurs catadioptriques et elles sont souvent restreintes à des configurations particulières : elles sont généralement spécifiques à un profil de miroir. Par ailleurs ces méthodes sont souvent trop restrictives, imposant soit des contraintes dans l'environnement (présence de lignes, sphères ou même mire de calibration) soit des contraintes d'utilisation (caméra en mouvement, obligation d'un montage/démontage d'une mire de calibration directement sur le capteur).

Les capteurs paracatadioptriques ont fait l'objet de travaux plus approfondis. Toutefois, peu de travaux prennent en compte les distorsions apportées par la

lentille télécentrique [GBL03, MP04]<sup>[1,2]</sup>. Par ailleurs, on peut, grâce à la lentille télécentrique, considérer qu'une translation du miroir selon l'axe des  $x$ , des  $y$  ou des  $z$  ne modifie en rien l'efficacité du capteur, mais que se passe-t-il si le miroir subit une petite rotation ? Comment réagissent les différents algorithmes de calibrage ?

Les autres types de capteurs n'ont pas bénéficié d'autant d'attention. Il est donc nécessaire de se pencher sur le problème du calibrage afin de mettre au point un algorithme qui ne souffre pas des restrictions que l'on vient de rencontrer. Il serait agréable d'avoir une méthode qui ne nécessite pas d'intervention humaine ni de suppositions sur l'environnement.

### 3.4 Notre Calibrage

Comme nous venons de le voir, il existe peu de méthodes pour calibrer un capteur omnidirectionnel et elles sont, soit trop spécifiques, soit trop contraignantes. Il nous faut mettre au point une méthode de calibrage non contraignante et adaptable sur tous types de capteurs.

Nous sommes partis de l'idée originale que le miroir peut servir de mire de calibrage. Il présente plusieurs avantages, il est toujours entièrement présent (pas d'occultation même partielle), il est parfaitement connu. De là, nous nous sommes inspirés de la méthode des deux plans pour mettre au point notre méthode de calibrage. Pour mieux comprendre comment adapter cette méthode à notre cas, revoyons les principales étapes de cette méthode comme elle est décrite dans [GTK88]<sup>[3]</sup>.

- 
- [1] Jose-Joel Gonzalez-Barbosa and Simon Lacroix. Un algorithme rapide de stéréovision panoramique dense. Technical Report 03561, LAAS-CNRS, 2003.
  - [2] Branislav Mičušík and Tomáš Pajdla. Para-catadioptric camera auto-calibration from epipolar geometry. *Proceedings of the Asian Conference on Computer Vision*, January 2004.
  - [3] K. Gremban, Charles Thorpe, and Takeo Kanade. Geometric camera calibration using systems of linear equations. In *Proceedings of the 1988 DARPA Image Understanding Workshop*, pages 820–825, April 1988.

### 3.4.1 Rappel de la méthode des deux plans

La caméra est placée face à une mire de calibrage parfaitement connue. Une homographie  $H_1$  est calculée entre le plan image  $P$  et le plan de la mire  $P_1$ . La mire est ensuite translatée (éloignée ou rapprochée) de la caméra d'une distance parfaitement connue  $t$ . Une seconde homographie  $H_2$  est calculée entre le plan image et le plan de la mire  $P_2$  (figure 3.10). A l'aide de ces homographies, un

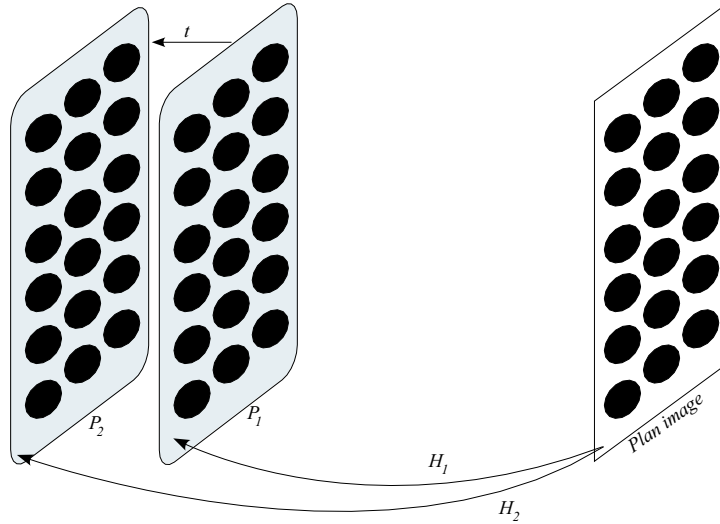


FIG. 3.10: Calcul des deux homographies  $H_1$  et  $H_2$  entre le plan image et les deux plans de calibrage  $P_1$  et  $P_2$ .

ensemble de points  $q_i$  de l'image est *projeté* sur chacun des plans de calibrage. Donc chaque point  $q_i$  donne deux points  $q'_i$  et  $q''_i$  sur  $P_1$  et  $P_2$  respectivement (figure 3.11). L'ensemble des couples  $(q'_i, q''_i)$  donne une droite dans l'espace et l'ensemble des droites convergent vers le point focal  $F$  de la caméra (figure 3.11). Il est ensuite possible de trouver les paramètres intrinsèques de la caméra. Si on note :

$$\vec{V}_i = \frac{q'_i - F}{\|q'_i - F\|} \quad (3.11)$$

$$\vec{q}_i = [\rho_i \ \gamma_i]^t \quad (3.12)$$

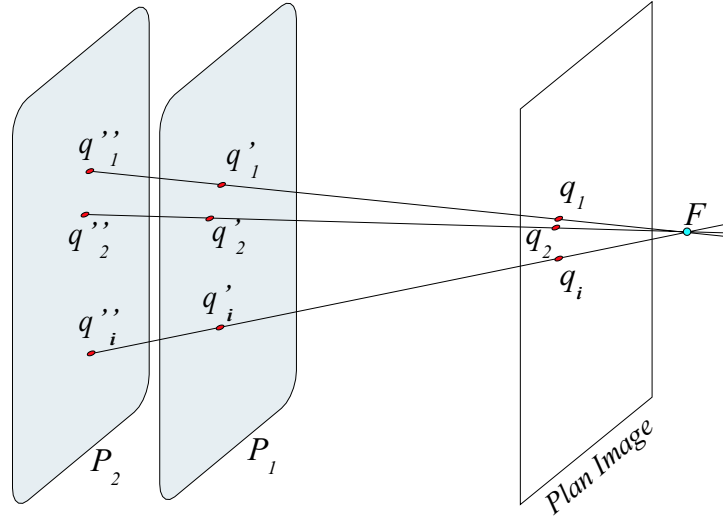


FIG. 3.11: Projection d'un ensemble de points de l'image sur les deux plans de calibrage. Chaque point donne un couple de points dans l'espace. Chaque couple de points donne une droite et l'ensemble des droites convergent vers le foyer de la caméra.

on peut écrire :

$$\rho_i = \vec{V}_i \vec{R} + \rho_p \quad (3.13)$$

$$\gamma_i = \vec{V}_i \vec{C} + \gamma_p \quad (3.14)$$

avec  $\vec{R}$  et  $\vec{C}$  les vecteurs de positionnement du capteur d'image.

$$\vec{R} = [R_x \ R_y \ R_z]^t \quad (3.15)$$

$$\vec{C} = [C_x \ C_y \ C_z]^t \quad (3.16)$$

et  $[\rho_p \ \gamma_p]$  le point d'intersection entre l'axe optique et le plan image.

On peut réécrire les équations 3.13 et 3.14 en une seule :

$$[\rho_i \ \gamma_i] = [V_{ix} \ V_{iy} \ V_{iz} \ 1.0] \begin{bmatrix} R_x & C_x \\ R_y & C_y \\ R_z & C_z \\ \rho_p & \gamma_p \end{bmatrix} \quad (3.17)$$

En utilisant plusieurs couples  $(q_i, \vec{V}_i)$  on peut calculer  $\vec{R}$ ,  $\vec{C}$  et  $[\rho_p \ \gamma_p]$ .

Voyons maintenant comment transposer cette méthode pour résoudre notre problème.

### 3.4.2 Principe du calibrage

Pour pouvoir adapter cet algorithme à notre cas, il nous faut trouver deux plans qui pourront servir de plans de calibrage. Si l'on revient à la géométrie de notre capteur, on peut sélectionner un plan  $P_1$ , en haut de notre miroir, plan contenant la frontière du miroir et du support et  $P_2$  en bas, plan contenant la jonction du miroir et de la pointe. Dans l'espace, chaque plan contient un cercle localisé précisément dans le repère miroir (figure 3.12). Ces deux cercles ( $C_1$  et  $C_2$ ) donnent sur l'image deux courbes ( $E_1$  et  $E_2$ ). Si le miroir est bien positionné et si l'axe optique de la caméra est confondu avec l'axe des Z du miroir, les cercles dans l'espace seront vus comme des cercles sur l'image (si l'on ne tient pas compte de l'*aspect ratio*). Si le miroir a subi une translation en X ou en Y (mais pas les deux) la projection des cercles donnera des ellipses. Enfin si le miroir a subi des translations suivant les deux axes, la projection des cercles donnera des ovoïdes. Dans l'ensemble, le décalage du miroir suivant sa position idéale (si décalage il y a) est supposé petit, donc nous pouvons, sans introduire une erreur significative, approximer ces ovoïdes par des ellipses. De même, dans le cas où le miroir aurait subi une petite rotation, la courbe obtenue sur l'image pourra être approximée par une ellipse. Nous avons

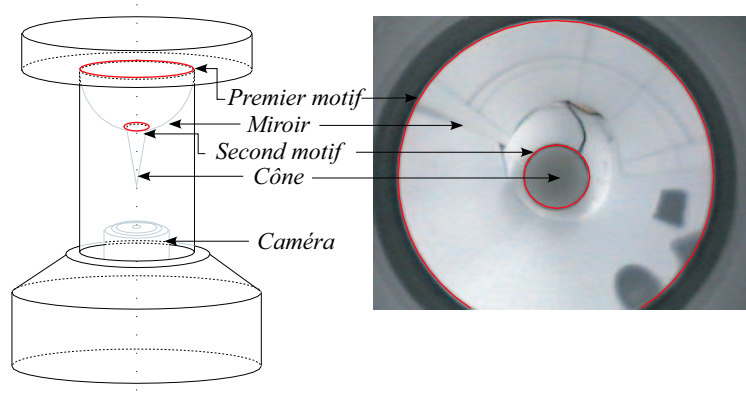


FIG. 3.12: Les deux plans de calibrage sur le miroir et les motifs de calibrage dans l'image.

donc un motif sur chacun des deux plans et sa correspondance sur l'image. Nous pouvons calculer l'homographie  $H_1$  qui fait passer du plan image au premier plan et, l'homographie  $H_2$  qui fait passer du plan image au second plan (figure 3.13).

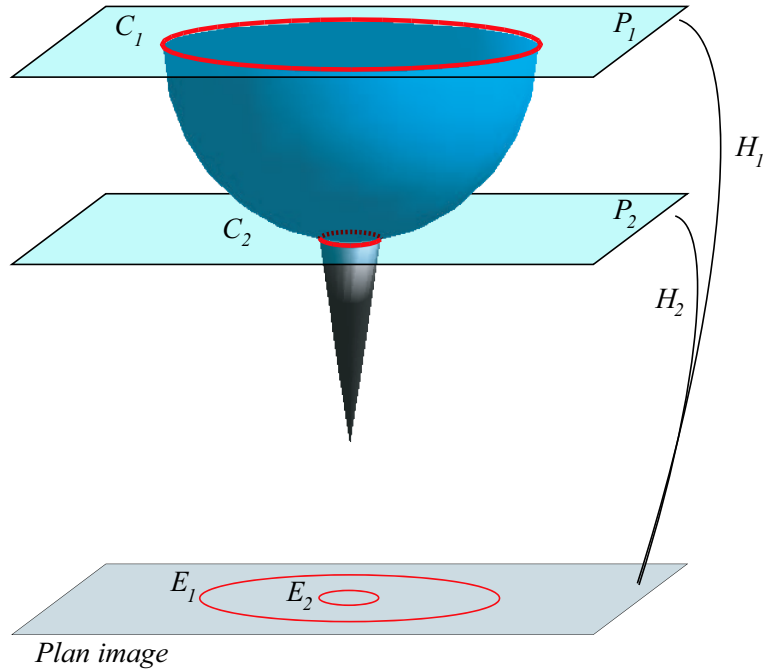


FIG. 3.13: Calcul des homographies entre le plan image et les plans de calibration.

L'étape suivante consiste à trouver la localisation du miroir relativement au repère caméra. Pour cela on prend un ensemble de points de l'image que l'on projette sur chacun des plans (pour simplifier et pour éviter l'ajout d'erreur on peut prendre le motif de calibration issu du plan  $P_1$  que l'on projette sur le plan  $P_2$ ). Nous obtenons deux nuages de points dans l'espace, exprimés dans le repère miroir, le premier sur le premier plan et le second sur le second plan (figure 3.14). Chaque point de l'image donne un couple de points dans l'espace (un point sur chaque plan) et chaque couple de points donne une droite de l'espace. L'ensemble des droites obtenues converge vers un point, le point focal de la caméra (figure 3.14). Remarquons que les droites sont exprimées dans le repère miroir, le foyer de la caméra est donc trouvé dans le repère miroir. Nous avons ainsi la distance foyer du miroir/foyer de la caméra, nous pouvons aussi vérifier leur alignement.

Continuons d'appliquer la méthode des deux plans afin de déterminer les vecteurs

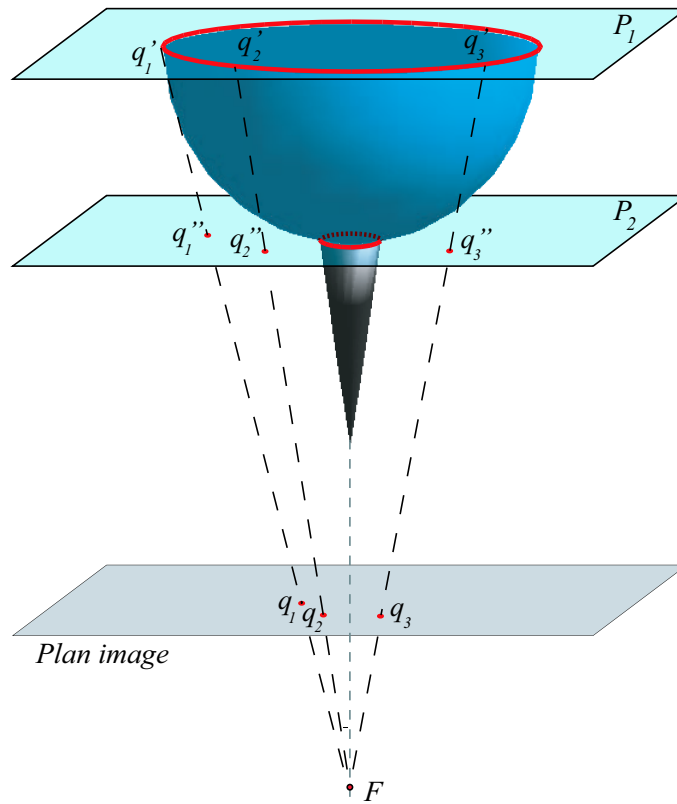


FIG. 3.14: Projection d'un ensemble de points de l'image sur les plans de calibrage. Chaque point de l'image donne un couple de points dans l'espace. Chaque couple de points donne une ligne et l'ensemble des lignes converge vers  $F$ .

de positionnement du capteur d'image ainsi que le centre de l'image. Pour cela nous pouvons résoudre l'équation 3.17 en utilisant la pseudo-inverse comme indiqué dans l'article [GTK88]<sup>[1]</sup>. Dans la pratique, les premiers tests ont donné des valeurs aberrantes pour  $\vec{R}$  et  $\vec{C}$ . En effet, physiquement ces deux vecteurs doivent être orthogonaux, ce qui n'était pas le cas lors de nos premiers essais. Nous avons dû forcer la contrainte d'orthogonalité :

$$\vec{R}^t \vec{C} = 0 \quad (3.18)$$

---

[1] K. Gremban, Charles Thorpe, and Takeo Kanade. Geometric camera calibration using systems of linear equations. In *Proceedings of the 1988 DARPA Image Understanding Workshop*, pages 820–825, April 1988.

en introduisant le multiplicateur de Lagrange  $\lambda \vec{R}^t \vec{C}$ .

Nous pouvons maintenant, à l'aide d'une méthode simple, calibrer les capteurs catadioptriques pour retrouver les paramètres intrinsèques de la caméra et la position du miroir relativement à la caméra.

### 3.4.3 Récapitulatif

Récapitulons les étapes du calibrage :

- **Etape 1** : Extraction des frontières du miroir.
- **Etape 2** : Calcul des homographies entre le plan image et les plans de calibrage.
- **Etape 3** : Projection d'un ensemble de points sur les deux plans de calibrage.
- **Etape 4** : Détermination du point de convergence de l'ensemble des lignes obtenues (localisation du foyer de la caméra dans le repère miroir).
- **Etape 5** : Détermination des vecteurs de positionnement du capteur d'image et du centre de l'image.

### 3.4.4 Extraction des contours du miroir

L'étape d'extraction des frontières du miroir n'est pas une étape facile et surtout, la précision du calibrage dépend de la précision de l'extraction. Il faut donc soigner cette étape. Plusieurs méthodes s'offrent à nous, notamment la transformée de Hough ou l'utilisation de "snakes" contraints.

Pour assurer une précision maximale lors du calibrage, nous délimitons les frontières du miroir à partir d'une séquence entière et non d'une seule image. Nous postulons que la zone image du miroir change fréquemment alors que le reste de l'image est fixe. Avec un nombre suffisant d'images, la zone du miroir est retrouvée en faisant la différence des images deux à deux et la moyenne des résultats obtenus (figure 3.15). Il reste à extraire les points de contour à la frontière du miroir et à approximer par une ellipse la courbe formée par ces points. Pour cela nous uti-



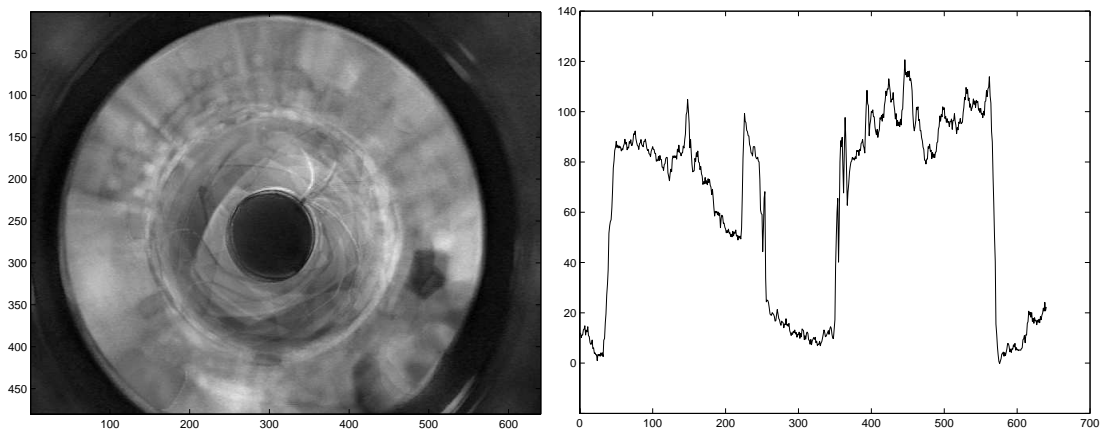


FIG. 3.15: A gauche la moyenne des résultats obtenus par soustractions successives : la frontière du miroir est alors facilement extraite par une binarisation et par morphologie mathématique. A droite une coupe du résultat.

lisons la technique décrite dans [PFF96]<sup>[1]</sup> (figure 3.16). Notre méthode restreint

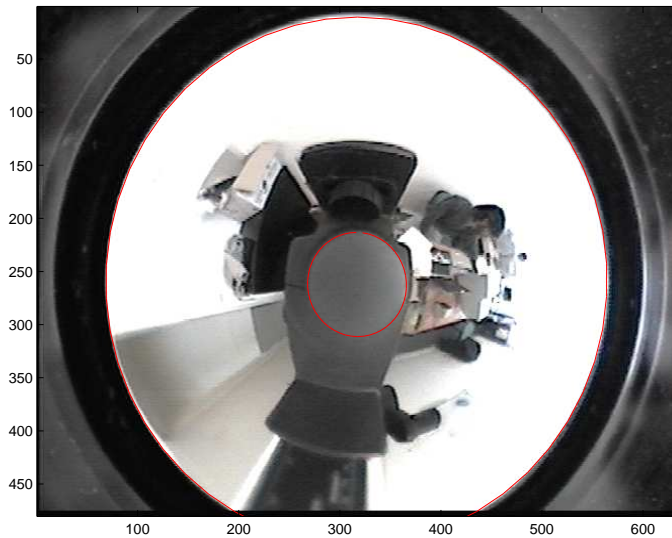


FIG. 3.16: Résultat de l'extraction des ellipses.

légèrement l'utilisation du calibrage car il ne peut être fait que lorsque la caméra est en mouvement et non pas à tout instant. Toutefois, elle permet une très bonne

---

[1] M. Pilu, A. Fitzgibbon, and R. Fisher. Ellipse-specific direct least-square fitting. In *IEEE International Conference on Image Processing*, September 1996.

précision d'extraction, ce qui est capital pour le reste du processus.

Les tests réalisés au laboratoire donnent de très bons résultats. En revanche, dans le cadre de la conduite assistée, les résultats sont beaucoup moins probants. L'ellipse extérieure est facilement extraite mais pas l'ellipse intérieure. Notre méthode étant basée sur les variations d'intensité lumineuse sur la zone image, nous ne parvenons pas à extraire aisément l'ellipse intérieure car la frontière du bas du miroir correspond au capot du véhicule. Cette partie de l'image est peu sujette aux variations lumineuses exceptées celles produites par les effets de lumières sur la carrosserie. Or le cône réfléchit beaucoup la lumière et les conditions d'éclairage ne suffisent pas. A l'avenir il faudrait appliquer une transformée de Hough pour

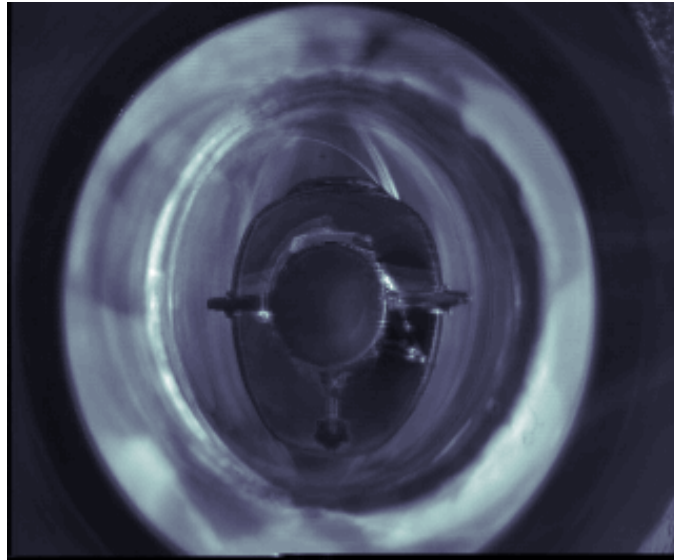


FIG. 3.17: Moyenne des résultats obtenus par soustractions successives d'images tirées d'une séquence routière. La frontière du haut du miroir apparaît clairement mais la frontière du bas du miroir est difficile à extraire.

extraire la frontière du bas, ce qui tirerait à notre avantage le fait que la zone image varie peu au cours du temps.

### 3.4.5 Validation et résultats

Nous allons vérifier la validité de notre approche et, surtout, nous convaincre de l'importance de l'utilisation du calibrage sur des images de synthèse. Par la

suite nous ferons des tests réels.

### 3.4.5.1 Validation sur images de synthèse

Pour effectuer nos tests sur images de synthèse, nous avons créé un capteur virtuel inspiré de notre capteur (capteur constitué d'un miroir à profil hyperbolique respectant la contrainte du foyer unique). Nous avons généré, à l'aide de ce capteur, une image virtuelle d'une mire placée sous le capteur en décalant de 5 mm suivant l'axe des Z, la position du foyer du miroir par rapport au foyer de la caméra (figure 3.18). Après avoir extrait les contours des motifs de la mire dans cette

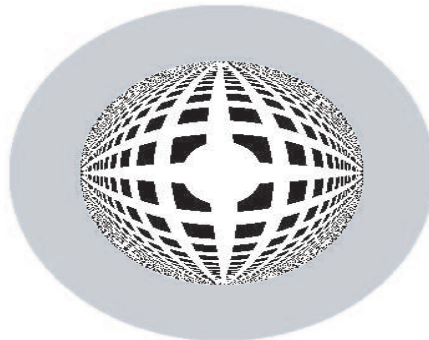


FIG. 3.18: Image de synthèse de la mire de calibration vue par notre capteur virtuel. Le foyer du miroir est déplacé de 5 mm selon l'axe des Z.

image, nous les avons reprojétés sur le plan de la mire en utilisant les paramètres idéaux du capteur et donc, en utilisant les simplifications apportées par le respect de la contrainte du foyer unique. Le résultat est visible sur la figure 3.19. On peut constater qu'il est loin d'être parfait. Nous avons une nouvelle fois reprojété les contours des motifs en utilisant les résultats donnés par le calibrage (les résultats du calibrage nous indiquent que la contrainte du foyer unique n'est plus respectée, nous utiliserons donc les lois classiques de la réflexion en utilisant les normales

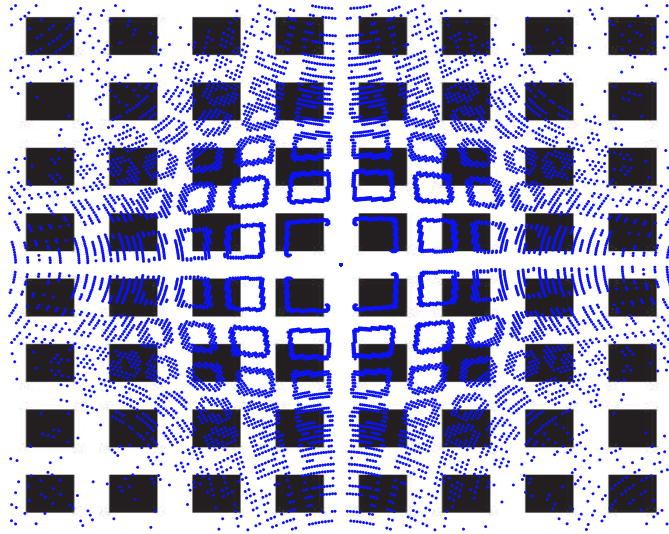


FIG. 3.19: Projection des contours des motifs avec les paramètres idéaux et en utilisant les simplifications apportées par le respect de la contrainte du foyer unique : la reprojexion ne correspond pas aux contours des motifs.

au miroir). Le résultat est visible sur la figure 3.20. Cette fois le résultat de la reprojexion colle parfaitement avec les motifs de la mire. Le calibrage nous a donc permis, d'une part, d'obtenir les valeurs exactes des paramètres du capteur et, d'autre part, d'éviter d'utiliser les simplifications apportées par la contrainte du foyer unique car l'alignement miroir/caméra n'était plus respecté.

Voyons maintenant le même test avec une translation de quatre millimètres du miroir dans le plan x-y. Le résultat est visible sur la figure 3.21. Si le résultat obtenu avec les paramètres estimés par notre calibrage n'est pas parfait, le résultat obtenu avec les paramètres idéaux du capteur est complètement inutilisable. Nous pouvons voir sur ces figures que l'étape du calibrage est indispensable.

Voyons maintenant le résultat du calibrage sur des images réelles.

### 3.4.5.2 Validation sur images réelles

Réalisons le montage que nous avons simulé précédemment. Posons notre capteur sur une mire et prenons une image (figure 3.22). Un certain nombre de points caractéristiques sont repérés sur cette image.

Le capteur est ensuite calibré. Enfin, les points de contrôle sont reprojetés sur le

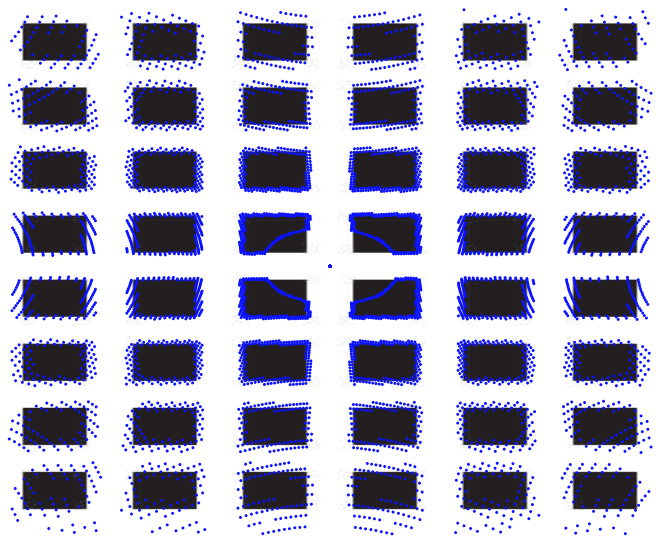


FIG. 3.20: Projection des contours des motifs avec les paramètres trouvés par le calibrage : les contours des motifs sont correctement reprojectés.

plan du sol, d'abord, en utilisant les paramètres constructeur du capteur et en utilisant les simplifications apportées par le respect de la contrainte du foyer unique (figure 3.23 à gauche), puis, en utilisant les paramètres fournis par le calibrage (figure 3.23 à droite). Dans les deux cas, les paramètres intrinsèques utilisés sont ceux fournis par le calibrage. La différence entre les deux cas portera donc sur la position relative du miroir par rapport à la caméra.

Nous pouvons voir que, si les résultats de la reprojection des points caractéristiques à l'aide des paramètres fournis par le calibrage (figure 3.23 à droite) ne sont pas parfaits, ils sont nettement moins distordus que les résultats reposant sur les paramètres idéaux du capteur. Particulièrement, les distances ainsi que les angles sont respectés, excepté l'angle en haut à gauche qui est légèrement faux. Sans le calibrage la reprojection est moins précise (figure 3.23 à gauche) notamment dans la partie inférieure où les distances ne sont plus respectées.

Dans les deux cas, on constate une légère rotation. Cette rotation est due au fait que la caméra panoramique est cylindrique et qu'il n'est pas facile de la positionner idéalement dans une direction particulière.

Nous avons donc pu voir à la fois sur des images de synthèse et sur des images réelles, l'apport incontestable du calibrage dans l'utilisation de tels capteurs. Pourtant notre méthode comporte une faiblesse : l'extraction des frontières du miroir. Il

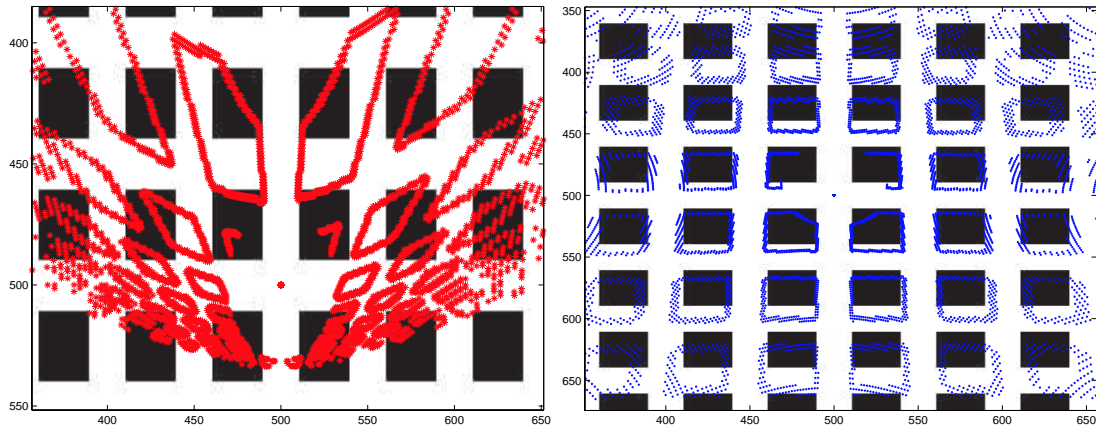


FIG. 3.21: Le miroir a subi une translation de 4 mm dans le plan x-y. Reprojection des motifs de calibrage, à gauche avec les paramètres idéaux du capteur et à droite avec les paramètres du calibrage.

est indispensable, pour accroître la robustesse du système, d'éliminer cette étape. Pour cela, nous proposons de fabriquer des miroirs facilitant l'application de notre méthode sur le capteur.

### 3.4.6 Amélioration du calibrage par des miroirs adaptés

Le miroir que nous avons était doté d'une aiguille qui a permis d'avoir le second plan de calibrage. Peu de miroirs sont dotés d'une telle aiguille, il est donc difficile de leur appliquer notre calibrage. De plus, il existe une infinité d'homographie entre deux coniques, il est donc difficile de calculer l'homographie entre le cercle de l'espace et l'ellipse extraite dans l'image. Pour contourner le problème nous avons échantillonné les courbes mais cet échantillonnage, un peu arbitraire, peut introduire une erreur dans le calcul des homographies. Il serait intéressant de créer les miroirs en vue de faciliter l'application de cette méthode aux capteurs et de rendre les capteurs auto-calibrables instantanément.

Plusieurs solutions sont envisageables : il est possible d'ajouter des marques en haut, autour du miroir et en bas, sur l'aiguille (figure 3.24) mais peu de miroirs sont équipés de cette aiguille, il faut donc une autre configuration. On peut ajouter un socle avec deux rebords (figure 3.25). L'avantage de cette configuration réside dans le fait que même des miroirs non prévus pour ce calibrage peuvent se voir

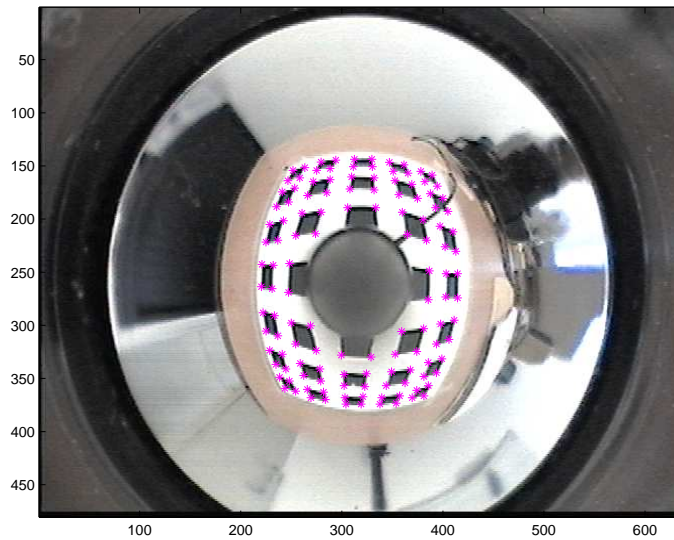


FIG. 3.22: Image d'une mire : le capteur est posé sur une mire parfaitement connue, une image est prise et un certain nombre de points caractéristiques sont repérés.

ajouter un socle muni de marques. Toutefois, la précision de la méthode est fortement dépendante de la distance qui sépare les deux plans de calibrage et cette configuration peut souffrir d'un manque de précision. En effet, si les deux plans sont trop rapprochés, une petite erreur de positionnement des points dans l'espace introduit une forte erreur dans l'angle des droites et donc la convergence vers le foyer de la caméra risque d'être fortement faussée. Une autre solution pour les miroirs ne disposant pas d'aiguille serait d'ajouter une série de marques en haut et aussi dans la zone centrale du miroir (figure 3.26). Cette zone est inutile car elle comporte le reflet de la caméra. Cette configuration est bien plus précise que la précédente car les plans de calibrage sont suffisamment éloignés.

L'utilisation d'un tel miroir permettrait, d'une part, de pouvoir appliquer précisément l'algorithme de calibrage tel qu'il est décrit dans ce chapitre et, d'autre part, il serait aussi possible d'accroître la précision du positionnement du miroir donné par le calibrage en utilisant la solution du problème perspective-N-points exposé au chapitre 2. Notons que sans ces marques, il est possible d'appliquer les solutions proposées au problème perspective-N-points : nous connaissons l'équation du motif (l'ellipse). Nous pouvons choisir deux points quelconques, distincts l'un de l'autre sur cette ellipse et trouver les points diamétralement opposés. Les antécédents de ces points dans l'espace forment un rectangle dont on ne connaît pas la longueur

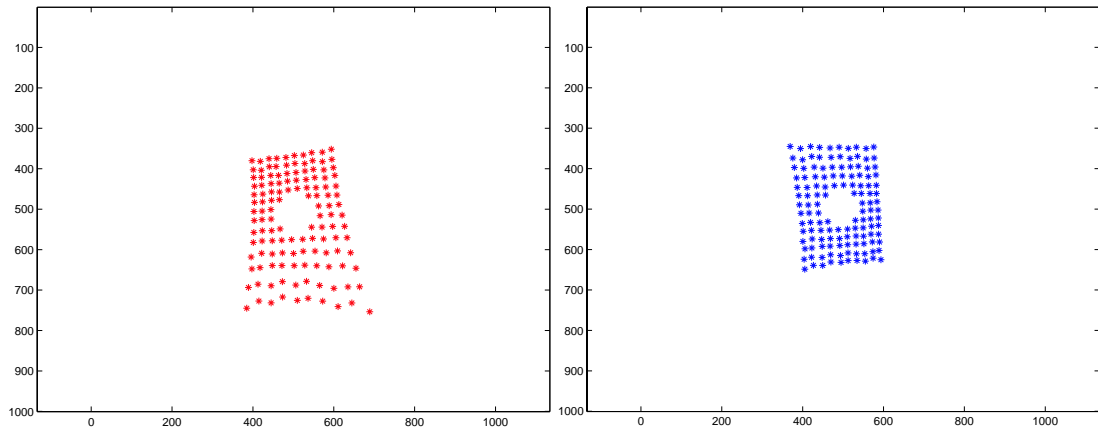


FIG. 3.23: Les points de contrôle sont reprojétés sur le sol, à gauche avec les paramètres idéaux du capteur et en utilisant les simplifications issues du respect de la contrainte du foyer unique et à droite en utilisant les paramètres donnés par le calibrage.

des cotés mais seulement la longueur des diagonales, ce qui est suffisant : on a un rectangle dont une distance est connue (voir chapitre 2).

Dans tous les cas, l'utilisation des calculs conduits au chapitre 2 peut accroître la précision du calibrage. Toutefois, il faut modérer cette approche car elle repose sur les valeurs des paramètres intrinsèques de la caméra. Ces valeurs sont estimées par notre méthode de calibrage.



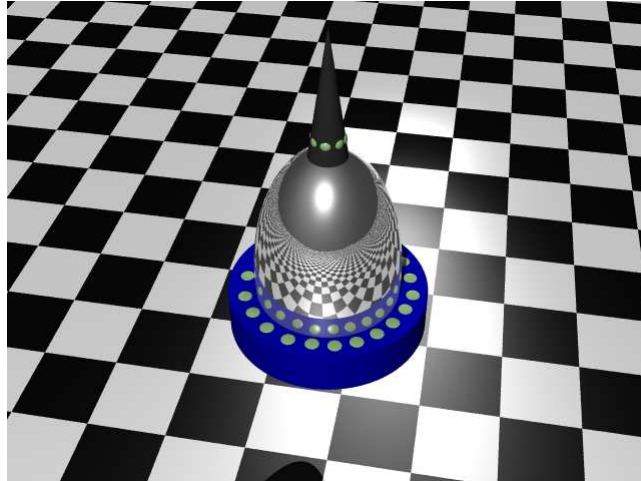


FIG. 3.24: Miroir auto-calibrant avec une aiguille.

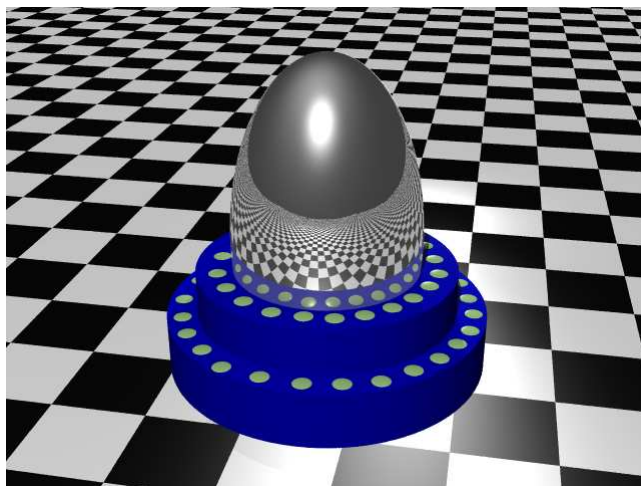


FIG. 3.25: Miroir auto-calibrant sans aiguille.

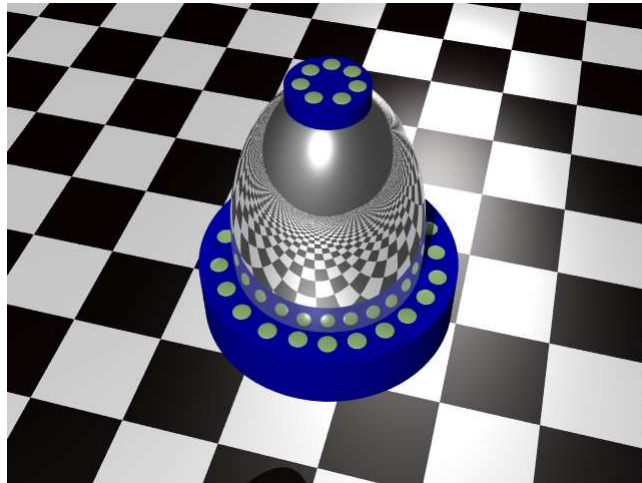


FIG. 3.26: Miroir auto-calibrant sans aiguille mais avec une bonne précision.

TAB. 3.1: Comparaison des différentes méthodes de calibrage

Méthode	Type de capteur	Paramètres estimés	Paramètres manquants	Contraintes
Direct circle-based self-calibration	seulement paracatadioptrique	le paramètre de la parabole, le centre de l'image	le <i>skew</i> et l' <i>aspect ratio</i>	aucune
Self-calibration method	seulement paracatadioptrique	le paramètre de la parabole, le centre de l'image, le <i>skew</i> et l' <i>aspect ratio</i>	aucun	tracking des points (caméra en mouvement et environnement statique)
Mičušík et Pajdla	seulement paracatadioptrique et hyperboloïdique	le paramètre de la parabole, le centre de l'image, le <i>skew</i> et l' <i>aspect ratio</i>	aucun	tracking des points (caméra en mouvement et environnement statique)
Geyer et Daniilidis	seulement paracatadioptrique	le paramètre de la parabole, le centre de l'image, le <i>skew</i> et l' <i>aspect ratio</i>	aucun	plusieurs ensembles de lignes parallèles dans l'espace
Geyer et Daniilidis	seulement paracatadioptrique	le paramètre de la parabole, le centre de l'image, le <i>skew</i> et l' <i>aspect ratio</i>	aucun	au moins 3 droites dans l'espace
Xianghua Ying et Zhanyi Hu	uniquement les capteurs respectant la contrainte du foyer unique	la distance focale, le centre de l'image, le <i>skew</i> , l' <i>aspect ratio</i> et la distance entre le point focal et le centre de l'image	aucun	au moins 2 droites ou 3 sphères
Cyril Cauchois	seulement muni d'un miroir conique	intrinsèques (avec distorsion) et extrinsèques	aucun	montage d'une mire sur le capteur
Carnegie Mellon University	tous types de miroir	position du miroir par rapport à la caméra	paramètres intrinsèques de la caméra	marquages précis de points de contrôle tout autour du capteur
Notre calibrage	tous types sauf ceux munis d'une lentille télécentrique	paramètres intrinsèques de la caméra et position relative miroir-caméra	aucun	aucune

## 3.5 Conclusion

Après avoir rappelé brièvement ce qu'était un capteur catadioptrique, nous avons présenté une nouvelle méthode pour calibrer ce type de capteurs en utilisant le miroir comme mire de calibrage. Nous avons testé cette méthode sur des images de synthèse et des images réelles. Cette méthode permet d'estimer quasi instantanément les paramètres intrinsèques de la caméra ainsi que la position relative du miroir par rapport à la caméra. De ce fait, elle peut être aussi utilisée pour évaluer si le capteur est bien positionné pour le respect de la contrainte du foyer unique. Comme le tableau de comparaison (3.1) le fait apparaître, notre méthode est la plus complète, la plus universelle et la moins contraignante des méthodes existantes. Elle possède plusieurs avantages non négligeables : tout d'abord elle ne dépend pas du profil du miroir (elle fonctionne quel que soit le profil du miroir pourvu que l'on n'utilise pas de lentille télécentrique), de plus elle ne nécessite aucun élément supplémentaire (mire de calibrage, connaissances ou suppositions a priori sur la scène) ni d'intervention humaine. Le calibrage peut donc être appliqué dans toutes circonstances : on peut calibrer le capteur à intervalles de temps réguliers ou après un évènement particulier. Dans notre cas, la navigation, on peut, par exemple, calibrer à chaque démarrage du véhicule, après un choc... L'étape d'extraction des frontières du miroir nécessite encore des améliorations. Pour cela, nous proposons de fabriquer des miroirs facilitant l'extraction des motifs de calibrage et donc améliorant le calibrage. Des miroirs de ce type permettraient en plus une précision de localisation accrue car ils permettraient aussi l'application du calcul présenté au chapitre 2.

# Chapitre 4

## Localisation Monoculaire : Le Problème Perspective-N-Points Adapté aux Capteurs Catadioptriques

### Sommaire

---

4.1	Introduction . . . . .	106
4.2	Solution du problème Perspective-N-Points . . . . .	106
4.3	Exemple . . . . .	108
4.3.1	Cas de 4 points coplanaires . . . . .	108
4.4	Cas particulier des capteurs respectant la contrainte du foyer unique . . . . .	110
4.5	Résultats expérimentaux . . . . .	111
4.5.1	Résultats sur images de synthèse . . . . .	111
4.5.2	Résultats sur images réelles . . . . .	112
4.6	Conclusion . . . . .	113

---

## 4.1 Introduction

Nous avons vu au chapitre 2 le problème perspective-N-points et nous en avons donné une solution analytique. Cette solution fonctionne avec une caméra classique. Notre désir maintenant est de changer de capteur afin d'utiliser des capteurs catadioptriques (particulièrement panoramiques) et profiter des nombreux avantages qu'ils apportent. Le problème perspective-N-points n'a jamais été abordé dans le cas panoramique. Nous proposons donc d'adapter notre démarche à ce type de capteurs. Nous verrons dans un premier temps comment adapter notre méthode pour trouver une solution analytique. Nous appliquerons cette méthode sur le cas qui nous intéresse (4 points coplanaires). Puis nous verrons les simplifications qu'apportent le respect de la contrainte du foyer unique. Enfin nous présenterons des résultats sur des images de synthèse et des images réelles.

## 4.2 Solution du problème Perspective-N-Points

Aucune solution présentée dans la littérature n'est facilement adaptable aux capteurs catadioptriques du fait de la nécessité du foyer unique de convergence des rayons lumineux. En ce qui concerne notre méthode, il est possible de l'adapter à condition de trouver une paramétrisation des rayons lumineux. Notre capteur est préalablement calibré (avec la méthode décrite au chapitre 3), la transformation entre le repère miroir et le repère caméra est donc connue. Il est donc suffisant de localiser les points de contrôle dans le repère miroir.

Si l'on regarde la figure 4.1, le point  $p_i$  de l'espace va se projeter en un point  $p'_i$  sur le miroir puis en un point  $p''_i$  sur le plan image. On voit qu'il est possible de trouver une paramétrisation pour localiser le point  $p_i$  :

$$p_i = p'_i + \lambda_i \vec{v}_i \quad (4.1)$$

Reste maintenant à trouver un moyen d'estimer le vecteur directeur  $\vec{v}_i$  : il est possible de trouver le vecteur directeur  $\vec{v}'_i$  de la droite  $(p''_i, p'_i)$ . D'après les lois de la réflexion il est donc possible de trouver le vecteur  $\vec{v}_i$  :

$$\vec{v}_i = \vec{v}'_i - 2\vec{N}_i(\vec{N}_i \vec{v}'_i) \quad (4.2)$$

avec  $\vec{N}_i$ , la normale au miroir au point  $p'_i$ .

Nous avons maintenant une expression des points de contrôle dans le repère miroir.

Il est donc possible d'exprimer les vecteurs  $\overrightarrow{p_i p_j}$  :

$$\overrightarrow{p_i p_j} = p'_j + \lambda_j \vec{v}_j - p'_i - \lambda_i \vec{v}_i \quad (4.3)$$

Bien que cette expression soit légèrement plus compliquée, nous nous sommes ramenés à la formulation précédente. On peut donc suivre la méthode et exprimer

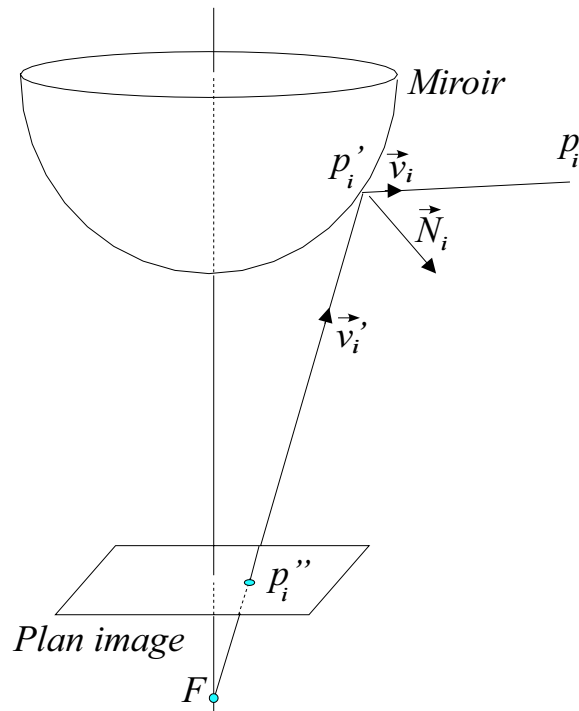


FIG. 4.1: Réflexion des rayons lumineux sur le miroir

des contraintes sur les vecteurs  $\overrightarrow{p_i p_j}$  afin d'obtenir un système d'équation. De ce système nous pourrions extraire les inconnues  $\lambda$  puis, localiser les points de contrôle dans le repère miroir.

### 4.3 Exemple

Appliquons notre méthode au cas qui nous intéresse pour essayer d'en tirer une solution analytique.

#### 4.3.1 Cas de 4 points coplanaires

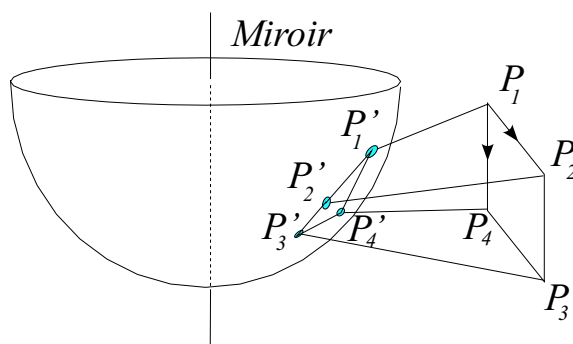


FIG. 4.2: Projection des 4 points de l'espace sur le miroir

Supposons que nous ayons 4 points dans l'espace  $p_1, p_2, p_3$  et  $p_4$ . Ces points forment un parallélogramme et se projettent sur le miroir en  $p'_1, p'_2, p'_3$  et  $p'_4$  respectivement. Enfin ces points se projettent sur le plan image en  $p''_1, p''_2, p''_3$  et  $p''_4$  respectivement (figure 4.2). A l'aide des points  $p''_1, p''_2, p''_3$  et  $p''_4$  ainsi que du foyer  $f$  de la caméra, nous pouvons calculer les vecteurs directeur  $\vec{v}_1, \vec{v}_2, \vec{v}_3$  et  $\vec{v}_4$  des droites  $(fp'_1), (fp'_2), (fp'_3)$  et  $(fp'_4)$ . De ces vecteurs nous pouvons calculer les vecteurs :  $\vec{v}_1, \vec{v}_2, \vec{v}_3$  et  $\vec{v}_4$  par la formule 4.2.

Comme les points forment un parallélogramme on peut écrire :

$$\overrightarrow{p_1 p_2} + \overrightarrow{p_3 p_4} = 0 \quad (4.4)$$

ce qui donne :

$$\begin{bmatrix} p'_{x2} + \lambda_2 v_{x2} - (p'_{x1} + \lambda_1 v_{x1}) + p'_{x4} + \lambda_4 v_{x4} - (p'_{x3} + \lambda_3 v_{x3}) \\ p'_{y2} + \lambda_2 v_{y2} - (p'_{y1} + \lambda_1 v_{y1}) + p'_{y4} + \lambda_4 v_{y4} - (p'_{y3} + \lambda_3 v_{y3}) \\ p'_{z2} + \lambda_2 v_{z2} - (p'_{z1} + \lambda_1 v_{z1}) + p'_{z4} + \lambda_4 v_{z4} - (p'_{z3} + \lambda_3 v_{z3}) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (4.5)$$



soit, écrit différemment :

$$\begin{bmatrix} \lambda_2 v_{x2} - \lambda_3 v_{x3} + \lambda_4 v_{x4} \\ \lambda_2 v_{y2} - \lambda_3 v_{y3} + \lambda_4 v_{y4} \\ \lambda_2 v_{z2} - \lambda_3 v_{z3} + \lambda_4 v_{z4} \end{bmatrix} = \begin{bmatrix} \lambda_1 v_{x1} \\ \lambda_1 v_{y1} \\ \lambda_1 v_{z1} \end{bmatrix} + \begin{bmatrix} v_{x0} \\ v_{y0} \\ v_{z0} \end{bmatrix} \quad (4.6)$$

avec

$$v_0 = p'_1 - p'_2 + p'_3 - p'_4 \quad (4.7)$$

Si on pose :

$$\Delta_{ijk} = \begin{vmatrix} v_{xi} & v_{xj} & v_{xk} \\ v_{yi} & v_{yj} & v_{yk} \\ v_{zi} & v_{zj} & v_{zk} \end{vmatrix} \quad (4.8)$$

La solution du système (4.6) est :

$$\lambda_2 = \frac{\Delta_{340} + \Delta_{134} \lambda_1}{\Delta_{234}} \quad (4.9)$$

$$\lambda_3 = \frac{\Delta_{240} + \Delta_{124} \lambda_1}{\Delta_{234}} \quad (4.10)$$

$$\lambda_4 = \frac{\Delta_{230} + \Delta_{123} \lambda_1}{\Delta_{234}} \quad (4.11)$$

Nous avons une expression de chaque  $\lambda_i$  relativement à l'un d'entre eux. Nous devons maintenant trouver l'expression du dernier  $\lambda$ . Si on pose  $L_{14}$  la longueur du segment  $[p_1 p_4]$ , on peut écrire :

$$\|\overrightarrow{p_1 p_4}\| = L_{14} \quad (4.12)$$

$$\Leftrightarrow \|p'_4 + \lambda_4 \vec{v}_4 - (p'_1 + \lambda_1 \vec{v}_1)\| = L_{14} \quad (4.13)$$

$$\Leftrightarrow \lambda_1^2 - 2\lambda_1 \lambda_4 \vec{v}_1 \vec{v}_4 + 2(p'_1 - p'_4)(\lambda_1 \vec{v}_1 - \lambda_4 \vec{v}_4) + \lambda_4^2 + p_1'^2 + p_4'^2 - 2p_1' p_4' = L_{14}^2 \quad (4.14)$$

$$(\text{avec } \|\vec{v}_i\| = 1 \text{ et } L_{14} > 0)$$

Combinée avec l'expression de  $\lambda_4$  (Eq. 4.11), on obtient :

$$\lambda_1^2 A + \lambda_1 B + C = 0 \text{ avec } \lambda_1 > 0 \quad (4.15)$$

et

$$A = 1 + \left( \frac{\Delta_{123}}{\Delta_{234}} \right)^2 - 2 \left( \frac{\Delta_{123}}{\Delta_{234}} \right) \vec{v}_1 \vec{v}_4 \quad (4.16)$$

$$B = 2 \frac{\Delta_{230} \Delta_{123}}{\Delta_{234}^2} + 2(p'_1 - p'_4) \left( \vec{v}_1 - \frac{\Delta_{123}}{\Delta_{234}} \vec{v}_4 \right) - 2 \frac{\Delta_{230}}{\Delta_{234}} \vec{v}_1 \vec{v}_4 \quad (4.17)$$

$$C = -2(p'_1 - p'_4) \frac{\Delta_{230}}{\Delta_{234}} \vec{v}_4 + (p'_1 - p'_4)^2 + \left( \frac{\Delta_{230}}{\Delta_{234}} \right)^2 - l^2 = 0 \quad (4.18)$$

Il est facile de trouver l'expression de la solution de ce polynôme (équation (4.15)) et donc d'avoir une expression de  $\lambda_1$ . Cette solution, combinée avec les équations (4.9) à (4.11), donne une expression de chaque  $\lambda_i$ . Comme nous l'avons expliqué au chapitre 2, l'équation (4.15) a deux solutions. La solution qui nous intéresse est la solution positive.

Nous avons expliqué comment obtenir une solution analytique au problème P-N-P puis, nous avons donné un exemple simple comportant 4 points coplanaires. Maintenant, voyons les simplifications qu'apportent le respect de la contrainte du foyer unique.

#### 4.4 Cas particulier des capteurs respectant la contrainte du foyer unique

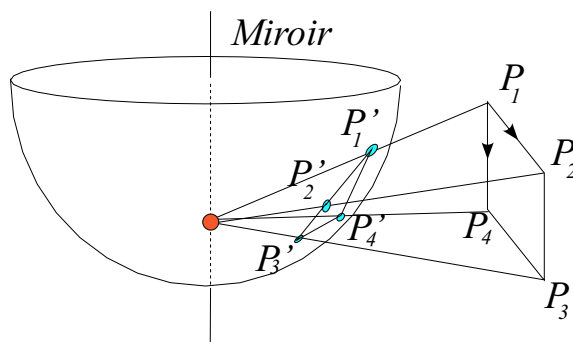


FIG. 4.3: Projection des 4 points de l'espace sur le miroir en utilisant la contrainte du foyer unique.

Certains capteurs catadioptriques disposent, nous l'avons vu, d'un foyer unique

de projection. Comme le montre la figure 4.3, le problème est simplifié et s'apparente au cas du P-N-P faisant intervenir une caméra plan. En effet, d'une part le calcul des vecteurs directeurs peut être fait à l'aide des projections des points sur le miroir et du foyer :

$$\vec{v}_i = \frac{p'_i - f}{\|p'_i - f\|} \quad (4.19)$$

et d'autre part les points de l'espace peuvent être exprimés comme précédemment :

$$p_i = f + \lambda_i \vec{v}_i \quad (4.20)$$

La solution sera donc strictement identique au cas plan.

Nous noterons que dans ce cas, la plupart des solutions proposées dans la littérature fonctionnent sans difficultés.

## 4.5 Résultats expérimentaux

Nous avons vérifié que l'expression de la solution était correcte. Pour cela nous avons effectué différents tests, d'abord avec des images de synthèse puis sur des images réelles.

### 4.5.1 Résultats sur images de synthèse

Pour vérifier l'expression de la solution trouvée, nous l'avons testée sur des images idéales. nous avons, à cette fin, développé une plateforme de tests. Elle permet de créer facilement une scène et d'en avoir, à la fois, un rendu classique (3D) et un rendu panoramique (par raytracing). Nous avons donc créé un véhicule virtuel et nous lui avons ajouté quatre marques (figure 4.4). Le simulateur analyse l'image panoramique, puis, à l'aide des marques, estime par notre méthode la distance et la compare avec la distance réelle. Quelque soit la position et l'éloignement des marques, la différence entre la distance estimée et la distance réelle est quasi nulle.

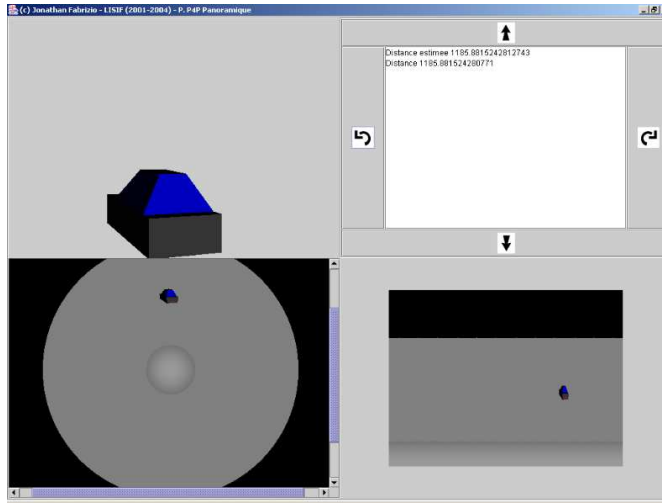


FIG. 4.4: La plateforme de simulation. Nous pouvons voir : en haut à gauche, la vue 3D de la scène, en bas à gauche, la vue panoramique de la scène, en haut à droite, les commandes qui permettent de déplacer le véhicule, en bas à droite la vue “aplatie” de la vue panoramique.

#### 4.5.2 Résultats sur images réelles

Nous avons testé notre solution sur des images réelles. Pour cela, nous avons pris des images d’une mire et nous avons pu localiser la mire correctement. Toutefois, il était difficile de vérifier la précision de la localisation. Nous avons donc réalisé des tests différents : nous avons pris, avec notre capteur, une image à une position donnée. Nous avons localisé le capteur relativement à quatre points de la scène puis, nous l’avons déplacé et nous avons pris une seconde image. Nous avons à nouveau localisé le capteur relativement aux mêmes points de contrôle. A l’aide de l’algorithme ICP[BM92]<sup>[1]</sup> nous avons recalé les points de la première localisation avec les points de la seconde localisation. Nous avons donc pu estimer une transformation entre les deux prises de vues. Pour vérifier que cette transformation était exacte nous avons sélectionné un certain nombre de points (coplanaires dans l’espace) sur la première image et nous les avons mis en correspondance avec leurs homologues de la seconde image (figure 4.5). Nous avons pu retrouver leurs positions dans l’espace (figure 4.7) en utilisant la transformation estimée. Les dis-

[1] Paul J. Besl and Neil McKay. A methode for registration of 3d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, February 1992.

tances et les angles sont bien conservés. Les points sont bien coplanaires. Pour nous convaincre que la précision obtenue est bien dépendante de la précision de l'estimation de la transformation qui lie les deux prises de vue, nous avons légèrement modifié la transformation estimée et nous avons pu constater que la reconstruction des points dans l'espace était très bruitée.

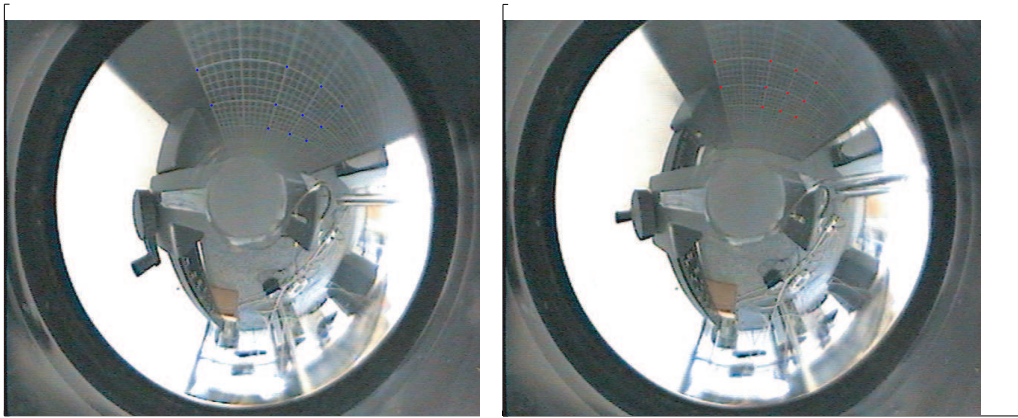


FIG. 4.5: Prises de vue différentes d'une même scène. Sur une image, des points de contrôle sont sélectionnés et sont mis en correspondance avec leurs homologues sur l'autre image.

## 4.6 Conclusion

Nous avons pu adapter notre méthode pour trouver une solution analytique au problème perspective-N-points avec des capteurs catadioptriques. Cette solution se simplifie dans le cas où le capteur respecte la contrainte du foyer unique. Dans ce cas, la solution obtenue est identique au cas plan. Nous avons appliqué notre méthode au cas simple qui nous intéresse à savoir le cas de quatre points coplanaires. Enfin nous avons vérifié sur des images de synthèse et réelles que cette solution fonctionne.

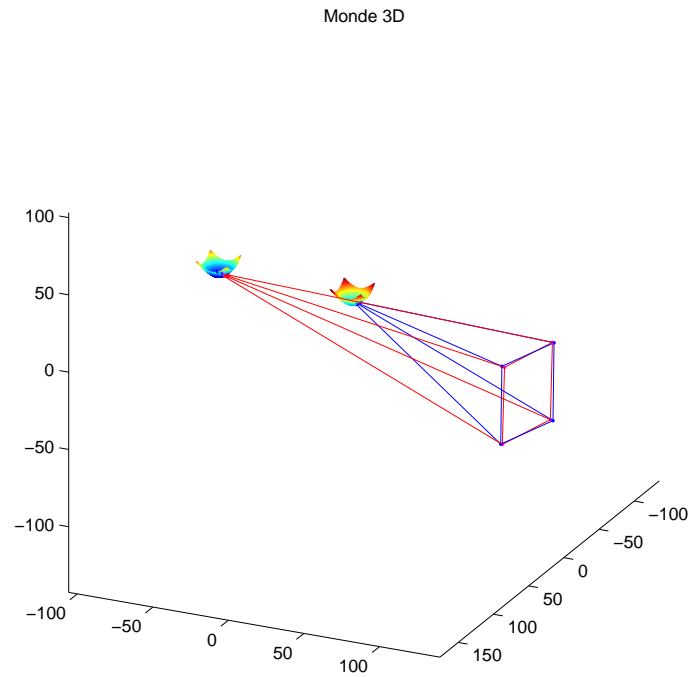


FIG. 4.6: La transformation qui lie les deux prises de vue est estimée.

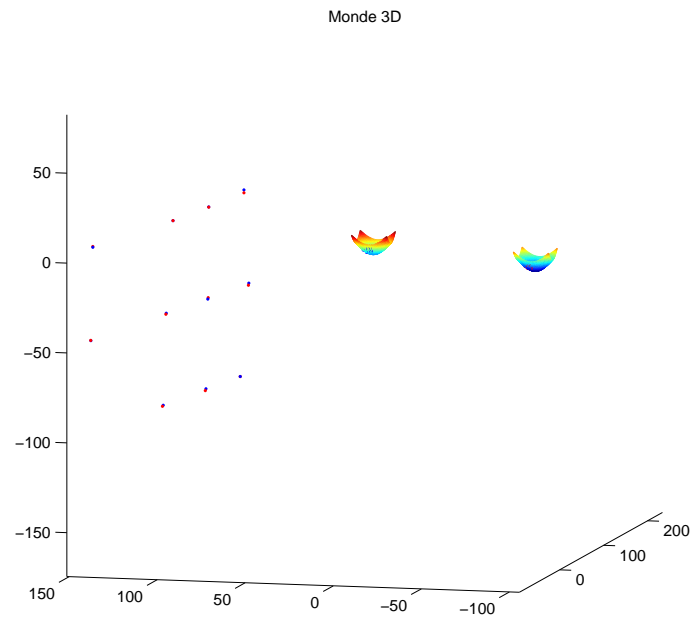


FIG. 4.7: L'ensemble des points sélectionnés est repositionné dans l'espace.

## Troisième partie

### Stéréovision Caméra Panoramique/Caméra plan





# Chapitre 5

## Stéréovision panoramique/plan

### Sommaire

---

<b>5.1</b>	<b>Introduction</b>	<b>118</b>
<b>5.2</b>	<b>Principe de la méthode</b>	<b>121</b>
<b>5.3</b>	<b>Etapes de l'algorithme</b>	<b>121</b>
5.3.1	Recherche de pixels candidats	122
5.3.2	Sélection des pixels de la courbe épipolaire	122
5.3.3	Appariement et localisation dans l'espace	123
<b>5.4</b>	<b>Efficacité de l'algorithme</b>	<b>124</b>
<b>5.5</b>	<b>Résultats expérimentaux</b>	<b>125</b>
5.5.1	Couple panoramique/plan	125
5.5.1.1	Résultats sur des images simulées	125
5.5.1.2	Résultat sur des images réelles	129
5.5.2	Couple panoramique/panoramique	131
<b>5.6</b>	<b>Conclusion</b>	<b>132</b>

---

## 5.1 Introduction

La stéréovision est maintenant un domaine maîtrisé du traitement de l'image. La possibilité d'utiliser des capteurs panoramiques, comme par exemple, les capteurs catadioptriques, élargit le champ de ce domaine. Néanmoins, l'usage des capteurs panoramiques rend les systèmes plus complexes que les caméra classiques. Plusieurs travaux examinent déjà cette possibilité (pas uniquement avec des capteurs catadioptriques).

Citons d'abord les travaux sur la stéréovision à base de caméras tournantes. Ishiguro et al. dans [IYT92]<sup>[1]</sup> utilisent deux colonnes de pixels éloignées de l'image, elle même issue d'une caméra en rotation, pour simuler un couple de capteurs panoramiques. Benosman dans [Ben97]<sup>[2]</sup> utilise une paire de barrettes CCD linéaires en rotation l'une au-dessus de l'autre. Dans ce cas, la géométrie épipolaire est extrêmement simple. Enfin, plus récemment, Smadja [Sma03]<sup>[3]</sup> étudie une paire stéréo, composée de vues de caméras CCD linéaires en rotation, dans une position quelconque.

Les capteurs catadioptriques ont aussi fait l'objet de plusieurs travaux de recherche. Une configuration simple à utiliser consiste en deux capteurs panoramiques catadioptriques placés l'un au dessus de l'autre comme le propose Gluckman et al. dans [GTN98]<sup>[4]</sup>. En effet, dans ce cas, les courbes épipolaires sont les lignes radiales de l'image. De même, Lin et Bajcsy proposent dans [LB03]<sup>[5]</sup> un capteur constitué de deux caméras co-axiales tournées vers un unique miroir. Ce dispositif permet, comme le dispositif constitué de deux capteurs l'un au-dessus de l'autre, d'avoir des courbes épipolaires simples à extraire. Une autre astuce pro-

---

[1] Hiroshi Ishiguro, Masashi Yamamoto, and Saburo Tsuji. Omni-directional stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):257–262, 1992.

[2] Ryad Benosman. *Reconstruction d'environnements percus par un capteur panoramique de stéréovision et détermination des modèles CAO*. PhD thesis, 1997.

[3] Laurent Smadja. *Génération d'environnements 3D denses à partir d'images panoramiques cylindriques*. PhD thesis, 2003.

[4] Joshua M. Gluckman, Keith Thorek, and Shree K. Nayar. Real time panoramic stereo. *Proc. of Image Understanding Workshop*, 1998.

[5] S. S. Lin and R. Bajcsy. High resolution catadioptric omni-directional stereo sensor for robot vision. *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1694–1699, September 2003.

posée dans [SBFR96, FB02]<sup>[6,7]</sup>, consiste à utiliser un miroir à double courbure. Là encore, les problèmes posés par l'établissement de la géométrie épipolaire sont contournés.

Plus généralement, Svoboda, dans [Svo99]<sup>[8]</sup>, donne les expressions de la géométrie épipolaire pour des capteurs catadioptriques munis de miroirs hyperboliques ainsi que pour des capteurs paracatadioptriques. Nene et Nayar, dans [NN98]<sup>[9]</sup>, donnent en plus les expressions des courbes épipolaires pour des capteurs catadioptriques munis de miroirs plans ou ellipsoïdaux. Enfin Sturm, dans [Stu02]<sup>[10]</sup>, envisage de coupler des capteurs classiques avec des capteurs paracatadioptriques.

Par ailleurs, en vue d'accélérer et de simplifier le processus d'appariement, des solutions sont proposées pour rectifier les courbes épipolaires [GBL03, GD03]<sup>[11,12]</sup>. Néanmoins, ces méthodes changent les voisinages de chaque pixel, on ne peut donc pas utiliser de méthode d'appariement basée sur les voisinages.

Dans tous les cas présentés ici, la géométrie épipolaire est proposée, soit pour des capteurs ayant un foyer de convergence, soit dans des configurations spatiales particulières, rendant triviale l'expression des courbes épipolaires. En effet, dans le cas de capteurs sans foyer de convergence dans une configuration quelconque, l'écriture de la contrainte épipolaire est presque impossible. Il faut prendre en considération les courbes caustiques [SGK01]<sup>[13]</sup>. L'appariement est aussi un

- 
- [6] D. Southwell, A. Basu, M. Fiala, and J. Reyda. Panoramic stereo. *International Conference on Pattern Recognition*, 1996.
  - [7] Mark Fiala and Anup Basu. Feature extraction and calibration for stereo reconstruction using non-svp optics in a panoramic stereo-vision sensor. *Proceedings of the 3rd IEEE Workshop on Omnidirectional Vision*, pages 79–86, June 2002.
  - [8] Tomáš Svoboda. *Central Panoramic Cameras Design, Geometry, Egomotion*. PhD thesis, 1999.
  - [9] Sameer A. Nene and Shree Nayar. Stereo with mirrors. *Proceedings of the 6th International Conference on Computer Vision, Bombay*, January 1998.
  - [10] Peter Sturm. Mixing catadioptric and perspective cameras. *Proceedings of the 3rd IEEE Workshop on Omnidirectional Vision*, pages 37–44, June 2002.
  - [11] Jose-Joel Gonzalez-Barbosa and Simon Lacroix. Un algorithme rapide de stéréovision panoramique dense. Technical Report 03561, LAAS-CNRS, 2003.
  - [12] Christopher Geyer and Kostas Daniilidis. Conformal rectification of omnidirectional stereo pairs. *Proceedings of the 4th Workshop on Omnidirectional Vision*, June 2003.
  - [13] R. Swaminathan, M. D. Grossberg, and Nayar S. K. Caustics of catadioptric cameras. In *IEEE International Conference on Computer Vision*, Vancouver, Canada, 2001.

problème du fait de la complexité des voisinages sur les images issues de capteurs catadioptriques [IBD03]<sup>[1]</sup>.

Dans le cadre de notre application, nous avons pensé au départ que nous pourrions utiliser la caméra classique pour voir loin devant car elle possède une bonne résolution et utiliser la caméra panoramique pour voir tout autour du véhicule mais à plus courte distance. Toutefois, les deux capteurs ont une zone commune dans leur champ de vision, la zone située devant le véhicule. Cette zone est la plus importante lorsque l'on navigue. Nous avons donc envisagé de faire de la stéréovision avec ce couple hétérogène de capteurs pour renforcer les décisions du système coopératif. Pour cela, il faut estimer la géométrie épipolaire de ce couple. Seulement, si la caméra classique dispose d'un foyer de convergence, la caméra panoramique n'en dispose pas toujours. Il est donc impossible de formaliser une telle géométrie. Nous aurions pu choisir de nous restreindre à un profil de miroir disposant d'un foyer unique. Dans ce cas, nous aurions pu estimer aisément la géométrie épipolaire entre les deux capteurs mais, après un travail pour définir un profil de miroir adéquat, nous avons abandonné cette idée. En effet, les profils de miroir pouvant respecter la contrainte du foyer unique n'ont qu'un seul degré de liberté. Nous n'étions donc même pas en mesure de contraindre un secteur angulaire du champ de vision de la caméra avec de tels profils (nous sommes partis sur la base d'un profil hyperbolique). En plus de cela, les résultats apportés par le calibrage nous ont aussi fortement découragés d'utiliser un capteur à foyer unique tant il est difficile d'assurer la contrainte du foyer unique. Pour ces raisons, nous avons essayé de mettre en place une méthode permettant d'estimer les courbes épipolaires pour n'importe quels types de capteurs. Ce qui nous permet d'utiliser un profil de miroir quelconque.

Nous allons donc voir une méthode systématique pour estimer les courbes épipolaires. Cette méthode fonctionne quelque soit le type de capteur utilisé. Puis nous verrons, sur des cas concrets, les résultats obtenus par la méthode. Nous la testerons premièrement, en simulant un cas idéal avec un couple plan/panoramique. Nous choisirons un capteur panoramique ayant un profil de miroir hyperbolique

---

[1] S. H. Ieng, R. Benosman, and J. Devars. An efficient dynamic multi-angular feature points matcher for catadioptric views. *Proceedings of the 4th Workshop on Omnidirectional Vision*, June 2003.

et respectant la contrainte du foyer unique (ce qui nous permettra d'introduire la géométrie épipolaire d'un tel couple et de vérifier par le calcul, les résultats obtenus par notre méthode). Puis, nous testerons notre méthode sur des images réelles issues d'un couple panoramique/plan. Enfin, nous testerons la méthode sur des images réelles d'un couple panoramique/panoramique et nous conclurons.

## 5.2 Principe de la méthode

N'ayant pas forcément de foyer, il est impossible malheureusement de trouver les équations des courbes épipolaires. Il va donc falloir estimer ces courbes numériquement. Pour cela, il suffit, pour chaque point d'une image d'un capteur, de définir quels points de l'image de l'autre capteur peuvent potentiellement faire partie de la courbe (ce qui nous impose de connaître la transformation entre les deux repères de chaque capteur). En d'autres termes, il faut, pour chaque pixel de l'image d'un capteur, déterminer la droite ( $D$ ) dans l'espace d'où il provient et trouver dans l'autre image, tous les pixels portés par des droites sécantes à ( $D$ ). La méthode est assez simple quoiqu'un peu lourde car la combinatoire est élevée mais elle a l'avantage de fonctionner avec n'importe quel type de capteur. Toutefois, dans la pratique, cet algorithme n'est pas évident à mettre en oeuvre. Voyons les détails de l'implémentation de cet algorithme.

## 5.3 Etapes de l'algorithme

Cet algorithme est composé de deux étapes. Durant la première étape, nous cherchons, pour un pixel de l'image du premier capteur, un certain nombre de pixels candidats pouvant potentiellement faire partie de la courbe épipolaire. Durant la seconde étape, nous sélectionnons parmi ces points candidats, les points faisant effectivement partie de la courbe. Puis, nous réitérons ces étapes pour chaque pixel utile de la première image.

### 5.3.1 Recherche de pixels candidats

Dans un premier temps, nous cherchons les pixels candidats, appartenant potentiellement à la courbe épipolaire. Pour cela, nous sélectionnons un point dans la première image, et passons en revue tous les pixels de la seconde image. Nous estimons la droite espace dont ils sont issus et nous ne gardons que ceux dont la droite est sécante avec la droite du pixel d'origine. Malheureusement, dans la pratique il est quasiment impossible de trouver deux droites sécantes provenant de chaque capteur (du fait de la discrétisation). Il faut donc chercher deux droites proches l'une de l'autre, dont la distance est inférieure à un seuil. Cela ne résout pas le problème. Nous ne pouvons pas fixer ce seuil une fois pour toute car il varie en fonction du capteur, de la distance entre les deux capteurs et même de la position du point considéré dans l'image... Ce seuil doit donc être adaptatif. Mais là encore, les critères à prendre en compte sont nombreux et varient trop d'un capteur à l'autre. Il paraît difficile de trouver une loi robuste pour fixer ce seuil. Nous nous contenterons de noter, pour chaque pixel, un score. Ce score dépend de la distance entre les droites. Lorsque la distance tend vers zéro, le score tend vers le score maximum. Lorsque la distance devient trop élevée ou lorsque l'intersection a lieu derrière l'un des capteurs, le score est nul. La distance entre les droites est calculée simplement : soient deux droites non coplanaires  $D(A, \vec{u})$  et  $D'(A', \vec{u}')$ . La distance  $d$  entre ces deux droites est donnée par l'équation :

$$d(D, D') = \frac{\overline{AA'} \cdot (\vec{u} \wedge \vec{u}')}{\|\vec{u} \wedge \vec{u}'\|} \quad (5.1)$$

Nous obtenons donc une carte de scores. Les pixels faisant potentiellement partie de la courbe épipolaire ont un score différent de zéro. Il ne reste plus qu'à trouver les pixels faisant effectivement partie de la courbe.

### 5.3.2 Sélection des pixels de la courbe épipolaire

Les points faisant partie de la courbe épipolaire sont les points de crête de la table des scores. Il ne reste donc plus qu'à extraire les crêtes. Ce qui peut se faire aisément avec deux masques (figure 5.1) auxquels on peut aussi ajouter deux autres masques (figure 5.2). Le résultat de cette recherche de crêtes nous fournira

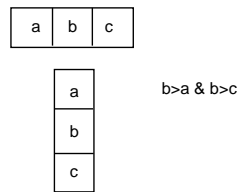


FIG. 5.1: Masques 3x1 et 1x3 pour la recherche de crêtes.

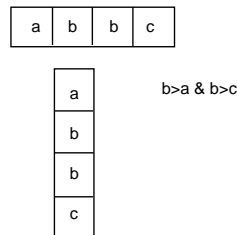


FIG. 5.2: Masques 4x1 et 1x4 pour la recherche de crêtes.

la courbe épipolaire. Toutefois, il peut apparaître parfois des artefacts. Pour les éviter, nous ajouterons quelques contraintes :

- l'orientation de la crête en un point doit être orthogonal, ou presque, à l'orientation du gradient,
- un point de crête ne doit jamais se trouver à côté d'un point dont le score est nul dans la table des scores,
- éventuellement, les points de crête doivent avoir un score supérieur à un seuil calculé par rapport à la moyenne des scores.

(Le dernier critère est facultatif car il pose à nouveau le problème du choix du seuil).

### 5.3.3 Appariement et localisation dans l'espace

Une fois la courbe estimée, nous pouvons envisager d'apparier les points des deux images. Pour l'instant nous n'avons pas de méthode robuste. Les algorithmes d'appariement robustes mettent en correspondance une courbe épipolaire avec l'autre courbe épipolaire. Dans notre cas, nous ne disposons pas de deux courbes. Nous devons, pour un pixel donné, rechercher son homologue sur la courbe de l'autre image. Le résultat est moins efficace. Pour rendre robuste cet appariement

il faut, d'une part, prendre en compte les voisinages des points et, d'autre part, confirmer l'appariement en partant du pixel sélectionné sur la seconde image et en cherchant son homologue sur le premier capteur. Pour l'instant, nous n'avons pas de résultats significatifs. Particulièrement, le choix du voisinage est un problème complexe.

Si l'on suppose l'appariement effectué, il est facile de retrouver la position du point dans l'espace : en notant  $I$  le point de  $(D)$  le plus proche de  $(D')$  et inversement  $I'$  le point de  $(D')$  le plus proche de  $(D)$  on peut dire que le point que l'on cherche est entre  $I$  et  $I'$  et que :

$$\overrightarrow{IA} = \alpha \vec{u} \quad (5.2)$$

$$\overrightarrow{A'I'} = \beta \vec{v} \quad (5.3)$$

avec

$$\alpha = \frac{\begin{vmatrix} -\vec{u} \overrightarrow{AA'} & \vec{u}\vec{v} \\ -\vec{v} \overrightarrow{AA'} & \vec{v}^2 \end{vmatrix}}{\begin{vmatrix} \vec{u}^2 & \vec{u}\vec{v} \\ \vec{u}\vec{v} & \vec{v}^2 \end{vmatrix}} \quad (5.4)$$

$$\beta = \frac{\begin{vmatrix} \vec{u}^2 & -\vec{u} \overrightarrow{AA'} \\ \vec{u}\vec{v} & -\vec{v} \overrightarrow{AA'} \end{vmatrix}}{\begin{vmatrix} \vec{u}^2 & \vec{u}\vec{v} \\ \vec{u}\vec{v} & \vec{v}^2 \end{vmatrix}} \quad (5.5)$$

Voyons maintenant comment améliorer l'algorithme.

## 5.4 Efficacité de l'algorithme

La combinatoire de cet algorithme est énorme, la vitesse d'exécution est donc trop grande pour une utilisation en temps réel. Il est toutefois possible d'envisager des astuces pour rendre cet algorithme plus rapide : il est possible de ne calculer le score que d'un pixel sur deux et ne calculer le score du pixel central que si au moins un des quatre pixels qui l'entourent a un score différent de zéro. Nous divisons ainsi



le temps d'exécution presque par deux. Mais cela n'est pas suffisant pour tenir le temps réel. Il faut donc précalculer les courbes épipolaires, tout en tenant compte de la phase d'appariement. Il faut donc, comme proposé dans [GBL03, GD03]<sup>[1,2]</sup>, ne plus avoir de courbe à parcourir mais des lignes. Pour cela, il faut redresser les courbes épipolaires. Mais, comme nous l'avons dit, cette opération change les voisinages des pixels. A la place, nous proposons donc de faire un adressage indexé des pixels. Les coordonnées de chaque pixel d'une même courbe sont rangées dans un tableau linéaire. Pour parcourir la courbe, il suffit de parcourir le tableau. Celui-ci va nous fournir, pour chaque pixel, sa position dans l'image, nous pourrions donc avoir aussi son voisinage.

## 5.5 Résultats expérimentaux

Nous allons montrer le fonctionnement de cette méthode, dans un premier temps, dans le cas plan/panoramique puis, dans le cas panoramique/panoramique.

### 5.5.1 Couple panoramique/plan

Nous allons vérifier notre méthode, d'abord, sur des images simulées et, ensuite, sur des images réelles.

#### 5.5.1.1 Résultats sur des images simulées

Vérifions dans un premier temps par le calcul notre algorithme. Pour cela, testons notre méthode dans un cas où les courbes épipolaires peuvent être estimées. Prenons le cas d'un capteur plan et d'une caméra panoramique muni d'un miroir à profil hyperbolique. Écrivons la contrainte épipolaire de ce couple.

- 
- [1] Jose-Joel Gonzalez-Barbosa and Simon Lacroix. Un algorithme rapide de stéréovision panoramique dense. Technical Report 03561, LAAS-CNRS, 2003.
- [2] Christopher Geyer and Kostas Daniilidis. Conformal rectification of omnidirectional stereo pairs. *Proceedings of the 4th Workshop on Omnidirectional Vision*, June 2003.

Le miroir  $M$  a pour équation (dans le repère miroir) :

$$\frac{(z + e)^2}{a^2} - \frac{x^2 + y^2}{b^2} = 1 \quad (5.6)$$

avec

$$e = \sqrt{a^2 + b^2} \quad (5.7)$$

ce qui donne :

$$(z + e)^2 b^2 - (x^2 + y^2) a^2 - a^2 b^2 = 0 \quad (5.8)$$

Soit  $F_1$  le foyer du miroir,  $F'_1$  le foyer de la caméra panoramique (exprimé dans le repère miroir) et  $F_2$  le foyer de la caméra plan. L'équation du plan ( $F_2, F_1, P_i$ ) (avec  $P_i$  un point image de la caméra plan) est :

$$\begin{vmatrix} x - F_{2x} & y - F_{2y} & z - F_{2z} \\ x - F_{1x} & y - F_{1y} & z - F_{1z} \\ x - P_{ix} & y - P_{iy} & z - P_{iz} \end{vmatrix} = 0 \quad (5.9)$$

ce qui donne :

$$(F_{2y}P_{iz} - F_{2z}P_{iy})x + (P_{ix}F_{2z} - F_{2x}P_{iz})y + (F_{2x}P_{iy} - P_{ix}F_{2y})z = 0 \quad (5.10)$$

Le plan épipolaire a donc pour équation :

$$px + qy + sz = 0 \quad (5.11)$$

ce qui donne :

$$z = -\frac{p}{s}x - \frac{q}{s}y \quad (5.12)$$

Il nous faut calculer l'intersection de ce plan avec le miroir, pour cela on injecte 5.12 dans 5.10 on trouve :

$$\left(-\frac{p}{s}x - \frac{q}{s}y + e\right)^2 b^2 - (x^2 + y^2)^2 a^2 - a^2 b^2 = 0 \quad (5.13)$$

Ce qui peut se ré-écrire

$$\begin{pmatrix} x & y & 1 \end{pmatrix} \begin{pmatrix} p^2b^2 - s^2a^2 & -pqb^2 & -sepb^2 \\ -pqb^2 & q^2b^2 - s^2a^2 & -seqb^2 \\ -sepb^2 & -seqb^2 & s^2e^2b^2 - s^2a^2b^2 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = 0 \quad (5.14)$$

On a donc l'expression de la solution de l'intersection du plan avec l'hyperboloïde. Cette solution est une conique que nous noterons  $A(P_i)$ . Elle est exprimée dans le plan épipolaire. Pour trouver l'équation de cette courbe dans l'espace, nous avons deux possibilités : soit en utilisant le fait que la courbe appartient au miroir, soit en utilisant le fait que la courbe appartient au plan. Nous utiliserons le fait que la courbe appartient au plan.

Soit  $m$  un point de la conique :

$$m = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (5.15)$$

Dans le repère miroir on a donc (d'après 5.12).

$$M = \begin{pmatrix} x \\ y \\ -\frac{px}{s} - \frac{qy}{s} \end{pmatrix} \quad (5.16)$$

ce qui donne :

$$M = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\frac{p}{s} & -\frac{q}{s} & 0 \end{pmatrix} m \quad (5.17)$$

Dans le repère caméra on obtient :

$$M - F'_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -\frac{p}{s} & -\frac{q}{s} & 2e \end{pmatrix} m \quad (5.18)$$

d'où

$$m = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{p}{2se} & \frac{q}{2se} & \frac{1}{2e} \end{pmatrix} (M - F'_1) \quad (5.19)$$

Posons :

$$N = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{p}{2se} & \frac{q}{2se} & \frac{1}{2e} \end{pmatrix} \quad (5.20)$$

donc l'équation 5.14 devient :

$$(M - F'_1)^t N^t A(P_i) N (M - F'_1) = 0 \quad (5.21)$$

Que l'on note :

$$(M - F'_1)^t B(P_i) (M - F'_1) = 0 \quad (5.22)$$

Il faut maintenant trouver l'équation de cette conique dans l'image de la caméra panoramique. Pour cela, on peut exprimer  $M$  de manière paramétrique :

$$M = F'_1 + \lambda \vec{w} \quad (5.23)$$

En calculant l'intersection avec le plan épipolaire on trouve :

$$\lambda = \mathcal{F}(\vec{w}) = \frac{2se}{pw_x + qw_y + sw_z} \quad (5.24)$$

donc

$$M - F'_1 = \mathcal{F}(\vec{w}) \vec{w} \quad (5.25)$$

donc l'équation 5.22 devient :

$$(\mathcal{F}(\vec{w}) \vec{w})^t B(P_i) (\mathcal{F}(\vec{w}) \vec{w}) = 0 \quad (5.26)$$

Ce qui peut se ré-écrire :

$$w^t C(P_i) w = 0 \quad (5.27)$$

avec

$$C(P_i) = K B(P_i) \quad (5.28)$$

$$K = 4s^2e^2 \quad (5.29)$$

Si on a, pour un pixel  $q$  de l'image panoramique :

$$w = Hq \quad (5.30)$$

En remplaçant  $w$  dans l'expression 5.27 on obtient :

$$q^t H^t C(P_i) H q = 0 \quad (5.31)$$

Ce qui donne :

$$q^t H^t K (N^t A(P_i) N) H q = 0 \quad (5.32)$$

On a donc une expression de la courbe épipolaire dans le plan image de la caméra panoramique. Du côté de l'image de la caméra classique, la solution est simple et connue, elle ne sera pas détaillée. Maintenant que nous avons les équations des courbes épipolaires, vérifions que notre algorithme produit bien le même résultat. Pour cela simulons des images des capteurs : prenons un ensemble de points de l'espace, projetons les, d'une part, sur la caméra classique puis sur la caméra panoramique. Choisissons l'un de ces points ( $p_1$ ) sur la caméra plan et estimons la courbe épipolaire sur l'image panoramique, à la fois par l'équation 5.32 et par notre méthode. Pour calculer l'équation de la courbe épipolaire, nous calculons l'intersection du plan épipolaire avec le miroir (figure 5.3). On obtient une ellipse que l'on exprime dans l'espace (figure 5.4). Il ne reste plus qu'à déterminer la projection de cette conique dans le plan image (figure 5.5). La courbe passe bien par la projection du point  $p_1$  (figure 5.6). Calculons maintenant la courbe épipolaire par notre méthode. Dans un premier temps, calculons la carte des scores (figure 5.7). Puis extrayons les crêtes (figure 5.8). Les deux courbes épipolaires obtenues par les deux méthodes sont identiques. Mieux, notre algorithme a éliminé une partie de la courbe car cette partie est masquée par le miroir.

### 5.5.1.2 Résultat sur des images réelles

Maintenant que nous avons l'assurance que notre méthode fonctionne correctement, voyons sur des images réelles les résultats obtenus. Dans un premier temps,

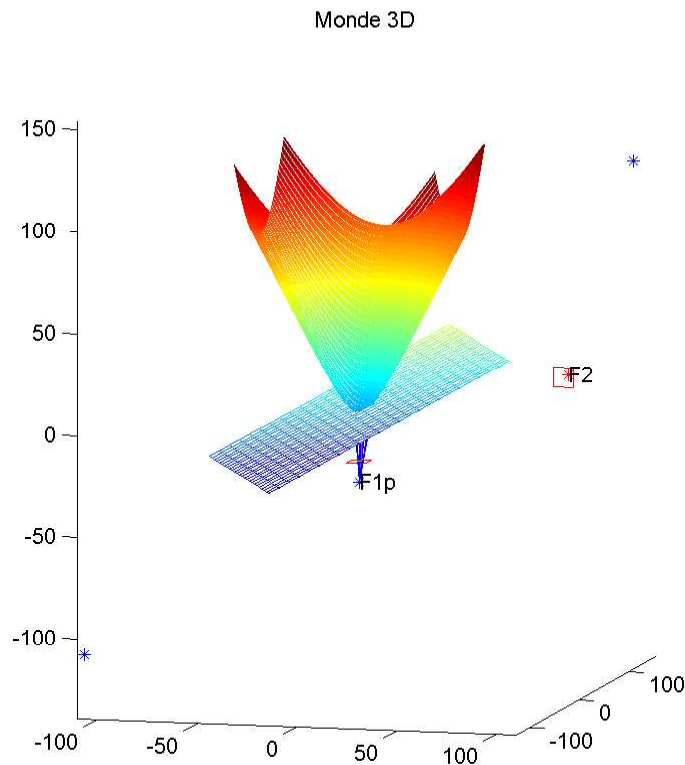


FIG. 5.3: Intersection du plan épipolaire avec le miroir.

calibrons nos deux caméras (le capteur panoramique est calibré avec notre méthode de calibrage cf chapitre 3). Prenons ensuite deux images d'une même scène, l'une avec une caméra classique, et l'autre avec un capteur panoramique et estimons la transformation entre chaque capteur et des points de contrôle de l'image. Pour cela nous utilisons les calculs des chapitres 2 et 4. A l'aide de l'algorithme ICP [BM92]<sup>[1]</sup>, nous pouvons recalibrer les points de contrôle de l'espace et ainsi, déduire la transformation entre les deux capteurs (figure 5.9). Nous pouvons donc enfin, choisir un point d'une des deux images (figure 5.10), calculer les scores (figure 5.11) et rechercher les crêtes dans la table des scores pour en déduire la courbe épipolaire (figure 5.12).

Il est possible de partir de l'autre image (figure 5.13), et d'appliquer le même processus sur l'autre capteur en calculant les scores (figure 5.14), et en recherchant les crêtes (figure 5.15). Nous avons donc pu estimer les courbes épipolaires en par-

[1] Paul J. Besl and Neil McKay. A method for registration of 3d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, February 1992.

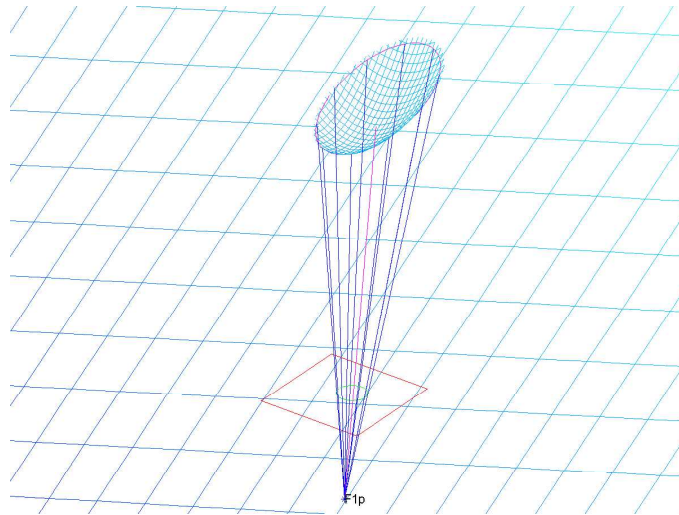


FIG. 5.4: Intersection du plan épipolaire avec le miroir. En violet, la conique à l'intersection du plan épipolaire et du miroir, en vert, la projection de cette conique sur le plan image.

tant soit de la caméra panoramique, soit en partant de la caméra classique. Voyons maintenant le cas de deux caméras panoramiques.

### 5.5.2 Couple panoramique/panoramique

Vérifions la validité de notre méthode avec un couple de capteurs panoramiques. Nous calibrons nos capteurs à l'aide de notre algorithme (chapitre 3), puis, nous estimons la transformation entre les deux capteurs comme nous l'avons fait dans le cas d'une caméra plan et d'une caméra panoramique (en utilisant les calculs du chapitre 4).

Si on sélectionne un point (figure 5.16), on peut estimer la courbe épipolaire sur l'autre image (figure 5.17). On peut voir la carte des scores (figure 5.18). On a donc pu vérifier sur un couple de capteurs panoramiques que notre méthode fonctionne correctement.

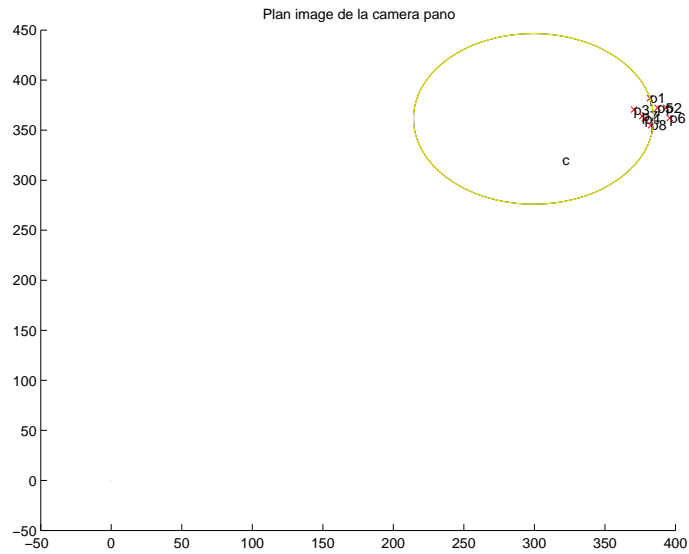


FIG. 5.5: Courbe épipolaire sur le plan image obtenue analytiquement.

## 5.6 Conclusion

Nous avons présenté une méthode d'estimation des courbes épipolaires fonctionnant quelque soit le type de capteur utilisé. Cette algorithmne réduit au mini-

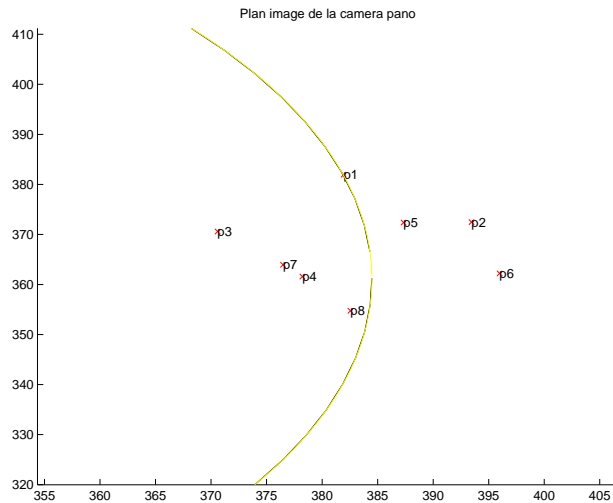


FIG. 5.6: Courbe épipolaire sur le plan image obtenue analytiquement. La courbe passe bien par le point  $p_1$ .



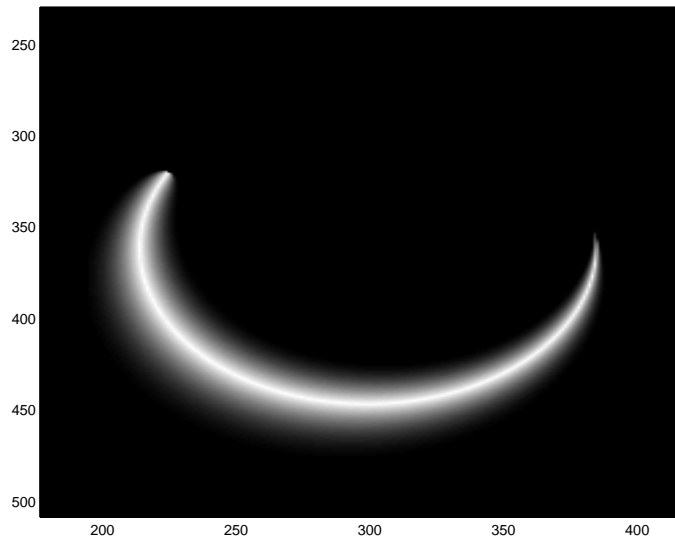


FIG. 5.7: Carte des scores

minimiser le nombre de pixels de la courbe épipolaire en éliminant ceux invisibles par l'autre capteur.

Ensuite, nous avons vérifié son fonctionnement sur des images de simulation, reposant sur un couple panoramique/plan. Nous avons choisi, pour la simulation

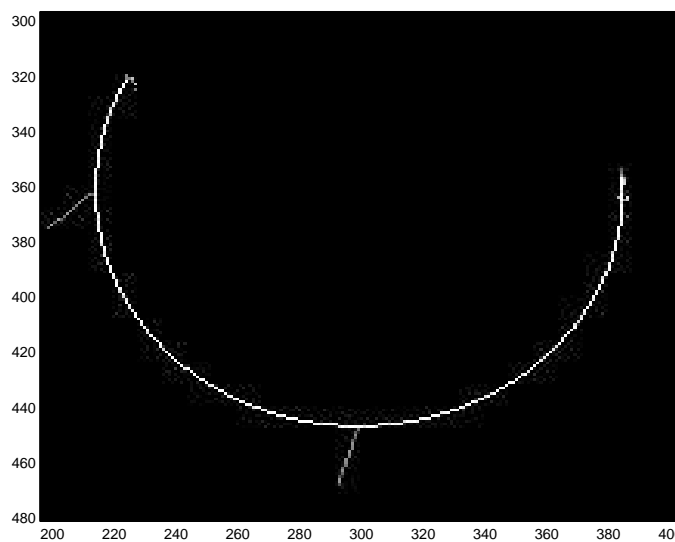


FIG. 5.8: Courbe épipolaire estimée par notre méthode (en blanc). Les points en gris sont des points de crêtes qui ne respectent pas les contraintes supplémentaires.

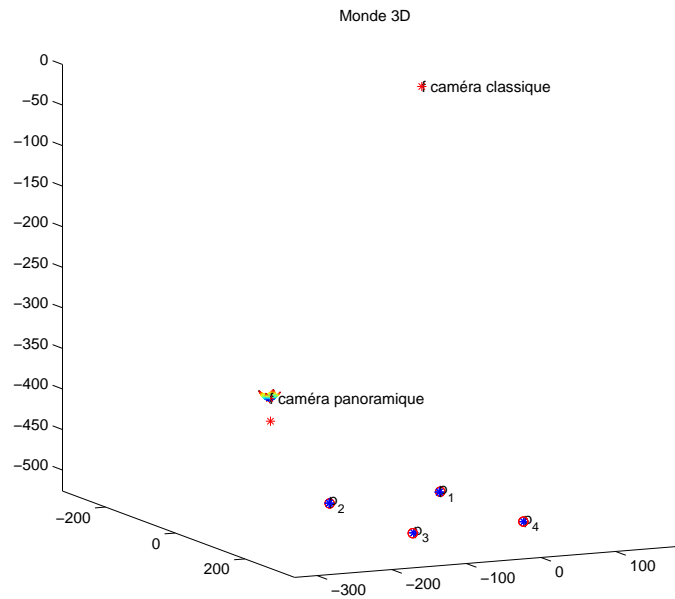


FIG. 5.9: La transformation entre les deux capteurs est estimée : des points de contrôle sont localisés dans le repère de chaque capteur puis recalés.

un capteur catadioptrique panoramique avec un miroir à profil hyperbolique respectant la contrainte du foyer unique. Nous avons exprimé analytiquement les courbes épipolaires pour ce couple, ce qui nous a permis de vérifier analytiquement le résultat fourni par notre algorithme.

Enfin, nous avons illustré notre méthode sur des images réelles issues d'un couple de capteur panoramique/plan ainsi que panoramique/panoramique.

Faute d'un algorithme robuste d'appariement, nous n'avons pas pu inclure de stéréovision dans notre système de localisation d'obstacles. La prochaine étape sera donc de mettre en place un appariement robuste, pour cela, une étude sur les voisinages des pixels doit être faite.

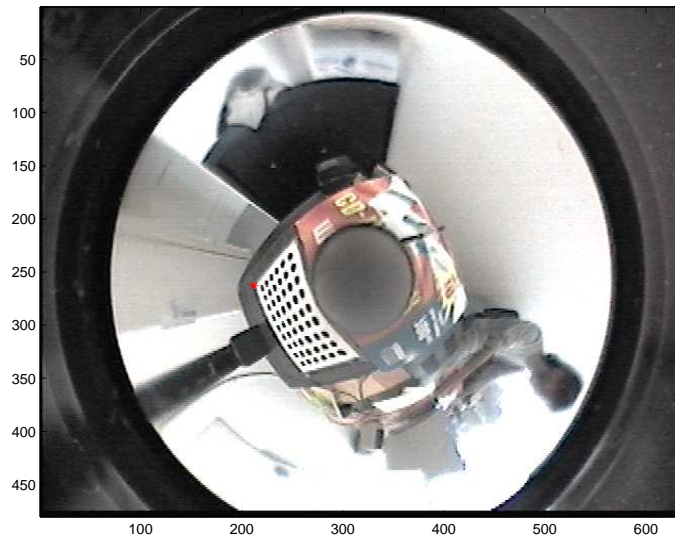


FIG. 5.10: Sélection d'un point dans l'image panoramique

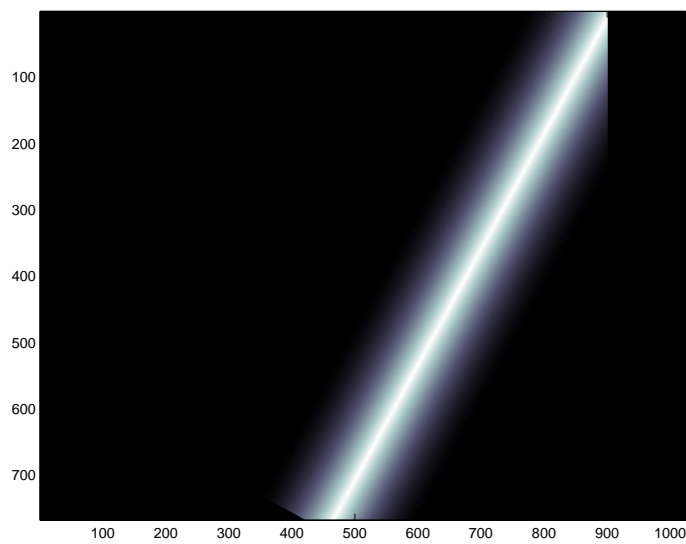


FIG. 5.11: Table des scores.

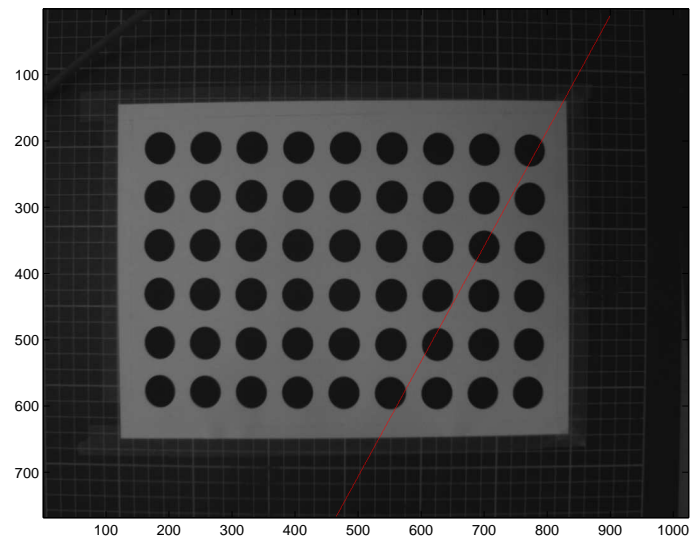


FIG. 5.12: Courbe épipolaire obtenue sur l'image de la caméra classique.

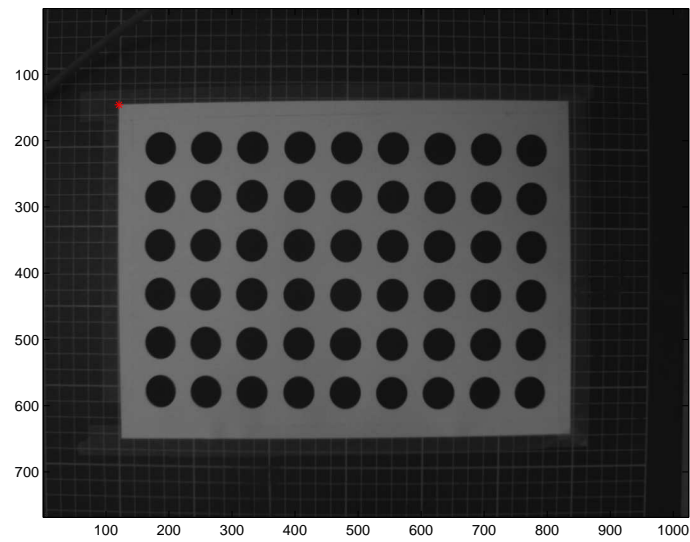


FIG. 5.13: Sélection d'un point sur l'image de la caméra classique.

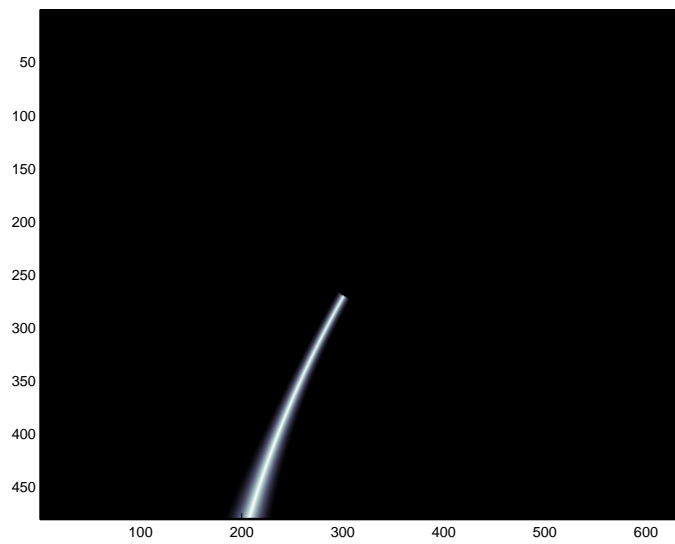


FIG. 5.14: Table des scores.

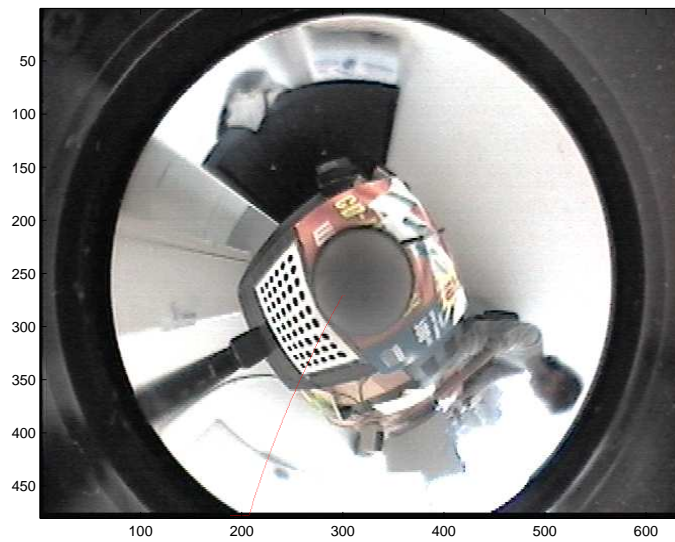


FIG. 5.15: Courbe épipolaire estimée sur l'image de la caméra panoramique.

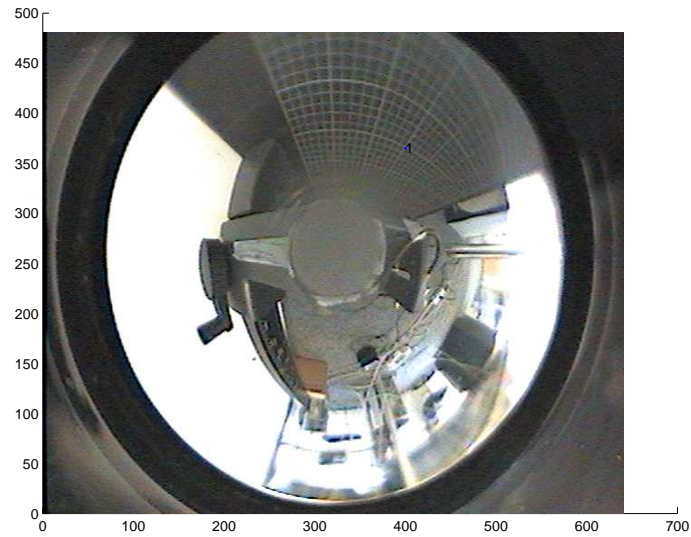


FIG. 5.16: Sélection d'un point dans l'image panoramique.

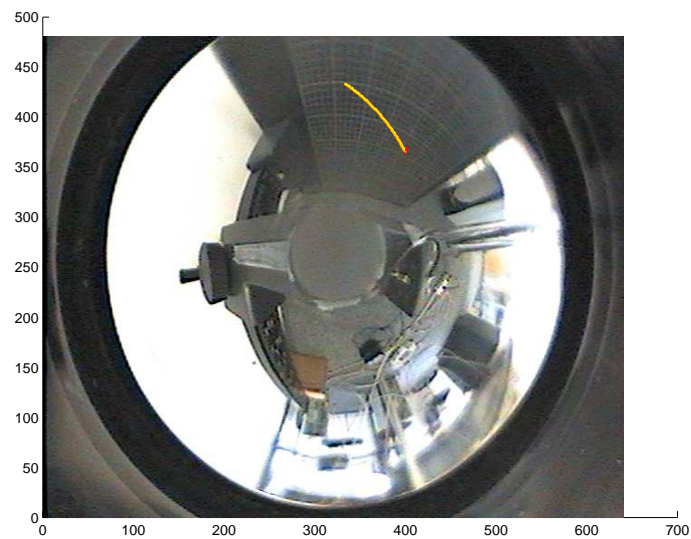


FIG. 5.17: Courbe épipolaire estimée dans l'image du second capteur : la courbe (en jaune) passe bien par le point sélectionné sur la première image. En rouge, les points de crête exclus de la courbe épipolaire.

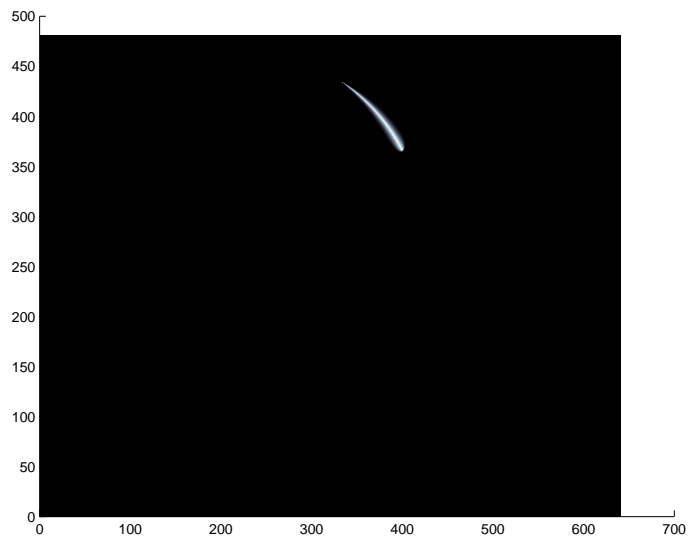


FIG. 5.18: Table des scores.





# Conclusion

Nous avons, dans nos travaux, posé les bases d'un système de détection d'obstacles coopératifs. Ce système repose sur l'utilisation de retro-rélecteurs, visibles, uniquement dans le proche infrarouge. Nous proposons un balisage coopératif, dont la forme permet de coder une information et de calculer la localisation de l'obstacle dans l'espace.

Nous avons, dans un premier temps, cherché à localiser ces obstacles avec une caméra classique. Pour cela, nous avons proposé une chaîne complète pour détecter et localiser la balise. Dans cette chaîne, nous présentons un algorithme rapide d'étiquetage des régions qui fonctionne en une passe unique même avec des objets non convexes. Il utilise les contours des régions. Pour la partie localisation, nous proposons de résoudre analytiquement et de manière originale le problème perspective- $N$ -points. Pour le cas particulier  $N = 4$ , nous étudions les différents paramètres qui influencent la stabilité de la solution. Pour accroître la précision de la localisation nous corrigeons les distorsions optiques de l'image. Ce système est étendu sans difficulté à la détection et à la lecture de la signalisation routière. Il peut aussi être utilisé pour faciliter la navigation de robots dans des environnements contraints (en usine ou en milieu hostile : centrales nucléaires...) où le marquage des obstacles est aisément envisageable.

Ensuite, nous avons étudié la possibilité d'utiliser des capteurs catadioptriques panoramiques dans notre système. Ces capteurs sont de plus en plus utilisés en robotique mobile. Nous avons dû analyser les particularités et les défauts de ce type de capteur. Après avoir passé en revue plusieurs méthodes existantes pour

calibrer ces capteurs, ces méthodes se sont avérées soit trop contraignantes, soit trop restrictives. Nous avons donc mis au point une méthode de calibrage simple, basée sur l'idée que le miroir pouvait servir de mire de calibrage. Notre méthode est peu contraignante : elle peut être complètement automatisée. Elle est plus universelle car elle fonctionne pour tous les types de capteurs sauf ceux munis d'une lentille télécentrique. Et elle est plus complète, elle permet, en effet, d'obtenir les paramètres intrinsèques et la position relative miroir/caméra. L'apport de cette méthode est démontré sur des images de synthèse et sur des images réelles.

Une fois le calibrage maîtrisé, nous avons adapté le système de localisation monoculaire aux capteurs catadioptriques panoramiques. Pour cela, nous proposons une solution analytique au problème "perspective-N-points" dans le cas de capteurs catadioptriques panoramiques.

Enfin, nous couplons des caméras classiques et des catadioptriques panoramiques pour faire de la stéréovision. Nous proposons une méthode générique pour estimer les courbes épipolaires quel que soit le type de capteur utilisé.

Dans l'ensemble des méthodes présentées applicables aux capteurs catadioptriques, un effort particulier est fait pour ne pas en restreindre l'usage à un capteur spécifique et surtout pas aux capteurs ne disposant que d'un foyer unique de projection. En effet, durant nos recherches, nous avons pris conscience que, malgré les nombreux avantages que le foyer unique apporte, il est trop restrictif. Par ailleurs nos travaux sur le calibrage nous ont montré que la contrainte d'alignement miroir/caméra imposée par ce type de capteurs est difficile à remplir. Un décalage même minime conduit à des erreurs énormes.

A ce système, il serait souhaitable d'ajouter un suivi des obstacles au cours du temps car, à l'heure actuelle, le système localise les obstacles, image par image, sans intégrer les informations des résultats précédents. Un tel ajout rendrait le système plus robuste et plus stable, surtout en cas d'occultation partielle ou temporaire de la balise.





# Bibliographie

- [AC95] M. A. Abidi and T. Chandra. A new efficient and direct solution for pose estimation using quadrangular targets : Algorithm and evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5) :534–538, 1995.
- [ADP96] S. Abdou, P. Daviet, and M. Parent. Platooning for vehicles and automatic parking by scheduling robotics actions. *International Symposium on Robotics and Manufacturing, WAC*, 1996.
- [Alt94] T. D. Alter. 3-d pose from 3 points using weak-perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8) :802–808, 1994.
- [AQT02] M-A. Ameller, Long Quan, and B. Triggs. Camera pose revisited : New linear algorithms. In *14ème Congrès Francophone de Reconnaissance des Formes et Intelligence Artificielle*, 2002. Paper in French.
- [Ben97] Ryad Benosman. *Reconstruction d’environnements percus par un capteur panoramique de stéréovision et détermination des modèles CAO*. PhD thesis, 1997.
- [BM92] Paul J. Besl and Neil McKay. A methode for registration of 3d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2) :239–256, February 1992.
- [BN99] Simon Baker and Shree K. Nayar. A theory of single-viewpoint catadioptric image formation. *International Journal of Computer Vision*, 35(2) :175 – 196, 1999.

- [Cau01] Cyril Cauchois. *Modélisation et Calibration du Capteur Omnidirectionnel SYCLOP : Application à la Localisation Absolue en Milieu Structuré*. PhD thesis, 2001.
- [CB97] Rodrigo L. Carceroni and Christopher M. Brown. Numerical methods for model-based pose recovery. Technical Report TR659, University of Rochester - Computer Science Department, 1997.
- [DD92] Daniel DeMenthon and Larry S. Davis. Exact and approximate solutions of the perspective-three-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(11) :1100–1105, November 1992.
- [DD95] Daniel F. DeMenthon and Larry S. Davis. Model-based object pose in 25 lines of code. *International Journal of Computer Vision*, 15(1-2) :123–141, 1995.
- [dISR] Centre National de la Sécurité Routière. <http://www.securiteroutiere.gouv.fr/cnsr>.
- [FB81] Martin A. Fischler and Robert C. Bolles. Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography. *Communication of the ACM*, 24(6) :381–395, 1981.
- [FB02] Mark Fiala and Anup Basu. Feature extraction and calibration for stereo reconstruction using non-svp optics in a panoramic stereo-vision sensor. *Proceedings of the 3rd IEEE Workshop on Omnidirectional Vision*, pages 79–86, June 2002.
- [FR02] Jay Farrell and UC Riverside. Developing magnetic and gps-aided inertial navigation for path's vehicle guidance system. *Intellimotion*, 9(4), 2002.
- [GBL03] Jose-Joel Gonzalez-Barbosa and Simon Lacroix. Un algorithme rapide de stéréovision panoramique dense. Technical Report 03561, LAAS-CNRS, 2003.
- [GD99] Christopher Geyer and Kostas Daniilidis. Catadioptric camera calibration. In *Proceedings of the 7th International Conference on Computer Vision, Kerkyra*, pages 398–404, 1999.

- [GD02] Christopher Geyer and Kostas Daniilidis. Paracatadioptric camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5) :687–695, 2002.
- [GD03] Christopher Geyer and Kostas Daniilidis. Conformal rectification of omnidirectional stereo pairs. *Proceedings of the 4th Workshop on Omnidirectional Vision*, June 2003.
- [GHTC03] Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, and Hang-Fei Cheng. Complete solution classification for the perspective-three-point problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8) :930–943, August 2003.
- [GLM<sup>+</sup>01] Paul Griffiths, Dirk Langer, James A. Misener, Mel Siegel, and Chuck Thorpe. Sensor-friendly vehicle and roadway systems. *IEEE Instrumentation and Measurement Technology Conference, Budapest, Hungary*, May 2001.
- [GPTZ96] Jürgen Guldner, Satyajit Patwardhan, Han-Shue Tan, and Wei-Bin Zhang. Magnetic markers on i-15 test track for nahsc demonstration. *Intellimotion*, 5(4) :02–05, 1996.
- [GTK88] K. Gremban, Charles Thorpe, and Takeo Kanade. Geometric camera calibration using systems of linear equations. In *Proceedings of the 1988 DARPA Image Understanding Workshop*, pages 820–825, April 1988.
- [GTN98] Joshua M. Gluckman, Keith Thorek, and Shree K. Nayar. Real time panoramic stereo. *Proc. of Image Understanding Workshop*, 1998.
- [HCD94] Radu Horaud, Stéphane Christy, and Fadi Dornaika. Object pose : The link between weak perspective, para perspective, and full perspective. Technical Report 2356, Institut National de Recherche en Informatique et en Automatique, September 1994.
- [HCLL89] Radu Horaud, Bernard Conio, Olivier Leboulleux, and Bernard Lacolle. An analytic solution for the perspective-4-point problem. *Computer Vision, Graphics and Image Processing*, 47 :33–44, 1989.
- [HLON91] Robert M Haralick, Chung Nan Lee, Kartsen Ottenberg, and Michael Nölle. Analysis and solutions of the three points perspective pose esti-

- mation problem. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 592–598, 1991.
- [HW02] Z. Y. Hu and F. C. Wu. A note on the number of solutions of the noncoplanar p4p problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4) :550–555, 2002.
- [HxTP<sup>+</sup>97] J. Hong, x. Tan, B. Pinette, R. Weiss, and E. M. Wisserman. Image based homing. *Proc. of Conference on Robotics and Automation*, 1 :181–187, 1997.
- [IBD03] S. H. Ieng, R. Benosman, and J. Devars. An efficient dynamic multi-angular feature points matcher for catadioptric views. *Proceedings of the 4th Workshop on Omnidirectional Vision*, June 2003.
- [Int98] *Intellimotion*, 7(4), 1998.
- [IYT92] Hiroshi Ishiguro, Masashi Yamamoto, and Saburo Tsuji. Omnidirectional stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2) :257–262, 1992.
- [JCHS00] Qiang Ji, Mauro S. Costa, Robert M. Haralick, and Linda G. Shapiro. A robust linear least-squares estimation of camera exterior orientation using multiple geometric features. *ISPRS Journal Of Photogrammetry And Remote Sensing*, 55(2) :75–93, June 2000.
- [Kan00] Sing Bing Kang. Catadioptric self-calibration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 201–207, 2000.
- [LB03] S. S. Lin and R. Bajcsy. High resolution catadioptric omni-directional stereo sensor for robot vision. *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1694–1699, September 2003.
- [Low80] D.G. Lowe. Solving for the parameters of object models from image descriptions. *ARPA Image Understanding Workshop*, pages 121–127, April 1980.
- [Low87] David G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31(3) :355–395, 1987.



- [Low91] David G. Lowe. Fitting parameterized three-dimensional models to images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5) :441–450, 1991.
- [LYR99] Ty A. Lasky, Kin S. Yen, and Bahram Ravani. The advanced snowplow driver assistance system. in *Proceedings of the Sixth World Congress on Intelligent Transportation System*, November 1999.
- [Mar00] François Marmoiton. *Détection et suivi par vision monoculaire d’obstacles mobiles coopératifs à partir d’un véhicule expérimental automobile*. PhD thesis, 2000.
- [MCM03] Lili Ma, YangQuan Chen, and Kevin L. Moore. A new analytical radial distortion model for camera calibration. *arXiv :cs.CV/0307046v1*, July 2003.
- [MP04] Branislav Mičušík and Tomáš Pajdla. Para-catadioptric camera auto-calibration from epipolar geometry. *Proceedings of the Asian Conference on Computer Vision*, January 2004.
- [MTH<sup>+</sup>99] James A. Misener, Chuck Thorpe, Ron Hearne, Lee Johnson, and Andrew c. Segal. Enhancing drivers-assist sensors : Background and concepts for sensors-friendly vehicles and roadways. in *The 6th World Congress on Intelligent Transportation Systems*, November 1999.
- [Nay97] S. K. Nayar. Catadioptric omnidirectional camera. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 1997.
- [NE02] Marthand Nookala and Brad Estochen. Minnesota, usa intelligent vehicle initiative. *Intelligent Vehicle*, June 2002.
- [NN98] Sameer A. Nene and Shree Nayar. Stereo with mirrors. *Proceedings of the 6th International Conference on Computer Vision, Bombay*, January 1998.
- [oVR] Persistence of Vision Raytracer. <http://www.povray.org>.
- [PFF96] M. Pilu, A. Fitzgibbon, and R. Fisher. Ellipse-specific direct least-square fitting. In *IEEE International Conference on Image Processing*, September 1996.

- [QL99] Long Quan and Zhong-Dan Lan. Linear n-point camera pose determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8) :774–780, August 1999.
- [Rou] La Sécurité Routière. <http://www.securiteroutiere.equipement.gouv.fr>.
- [SBFR96] D. Southwell, A. Basu, M. Fiala, and J. Reyda. Panoramic stereo. *International Conference on Pattern Recognition*, 1996.
- [SGK01] R. Swaminathan, M. D. Grossberg, and Nayar S. K. Caustics of catadioptric cameras. In *IEEE International Conference on Computer Vision*, Vancouver, Canada, 2001.
- [Sma03] Laurent Smadja. *Génération d'environnements 3D denses à partir d'images panoramiques cylindriques*. PhD thesis, 2003.
- [SMKS01] D. Strelow, J. Mishler, D. Koes, and S. Singh. Precise omnidirectional camera calibration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 689–694, 2001.
- [Stu02] Peter Sturm. Mixing catadioptric and perspective cameras. *Proceedings of the 3rd IEEE Workshop on Omnidirectional Vision*, pages 37–44, June 2002.
- [Svo99] Tomáš Svoboda. *Central Panoramic Cameras Design, Geometry, Egomotion*. PhD thesis, 1999.
- [TP96] Han-Shue Tan and Satyajit Patwardhan. Magnetic sensor for automatic steering control. *Intellimotion*, 5(3) :10–11, 1996.
- [WMSM91] William J. Wolfe, Donald Mathis, Cheryl Weber Sklair, and Michael Magee. The perspective view of three points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(1) :66–73, January 1991.
- [YH03] Xianghua Ying and Zhanyi Hu. Catadioptric camera calibration using geometric invariants. In *Proceedings of the 9th International Conference on Computer Vision, Nice*, 2003.
- [Zha02] Zhengyou Zhang. A flexible new technique for camera calibration. Technical Report MSR-TR-98-71, Microsoft Research, August 2002.
- [ZSE86] Cao Z, Oh S, and Hall E. Dynamic omnidirectional vision for mobile robots. *Journal of Robotics Systems*, 3(1) :5–17, 1986.

- [ZTS<sup>+</sup>99] Wei-Bin Zhang, Han-Shue Tan, Aaron Steinfeld, Benedicte Bougler, Dan Empey, Kun Zhou, and Masayoshi Tomizuka. Implementing advanced vehicle control and safety systems (avcss) for highway maintenance operation. *in Proceedings of the Sixth World Congress on Intelligent Transportation System*, November 1999.