



HAL
open science

Construction de modèles 3D à partir de données vidéo fisheye : application à la localisation en milieu urbain

Julien Moreau

► **To cite this version:**

Julien Moreau. Construction de modèles 3D à partir de données vidéo fisheye : application à la localisation en milieu urbain. Vision par ordinateur et reconnaissance de formes [cs.CV]. Université de Technologie de Belfort-Montbéliard, 2016. Français. NNT : 2016BELF0290 . tel-01417996

HAL Id: tel-01417996

<https://theses.hal.science/tel-01417996>

Submitted on 16 Dec 2016

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

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

SPIM

Thèse de Doctorat



école doctorale sciences pour l'ingénieur et microtechniques
UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

Construction de modèles 3D à partir de données vidéo fisheye

Application à la localisation en milieu urbain

■ JULIEN MOREAU



SPIM

Thèse de Doctorat



école doctorale sciences pour l'ingénieur et microtechniques
UNIVERSITÉ DE TECHNOLOGIE BELFORT-MONTBÉLIARD

N° 2 | 9 | 0

THÈSE présentée par

JULIEN MOREAU

pour obtenir le

Grade de Docteur de

l'Université de Technologie de Belfort-Montbéliard

Spécialité : **Informatique**

Construction de modèles 3D à partir de données vidéo fisheye Application à la localisation en milieu urbain

Unité de Recherche :

IRTES-SET et IFSTTAR-COSYS-LÉOST

Soutenue publiquement le 7 juin 2016 devant le Jury composé de :

PHILIPPE BONNIFAIT	Rapporteur	Professeur à l'UTC
PASCAL VASSEUR	Rapporteur	Professeur à l'Université de Rouen
JEAN-CHARLES NOYER	Examineur	Professeur à l'ULCO
MOUNIR ADJRAD	Examineur	Enseignant-chercheur à l'University College London
JULIETTE MARAIS	Invitée	Chargée de recherche à l'IFSTTAR
YASSINE RUICHEK	Directeur de thèse	Professeur à l'UTBM
SÉBASTIEN AMBELLOUIS	Co-directeur	Chargé de recherche à l'IFSTTAR

REMERCIEMENTS

Je remercie tous ceux sans qui la thèse n'aurait pas fonctionné comme cela a été le cas. Vous apportez chacun une histoire et intervenez de près ou de loin comme collègue, ami, famille ou rencontre fortuite à travers les nombreuses périodes et lieux de résidence occupés durant ces quelques années. Une excellente chose qu'offre la thèse c'est la possibilité de se rapprocher de personnes de milieux variés et multiculturels. La thèse est l'opportunité d'une ouverture globale avec au centre le plan humain, elle n'est pas que périple scientifique.

Pour rester dans le contexte professionnel ne sont citées que les personnes m'ayant aidé ou ayant échangé avec moi de loin ou de près pour le projet ou avec qui un travail commun a été réalisé. En premier, Sébastien et Yassine, qui ont concrétisé mon désir de me lancer dans la recherche et m'ont supporté par leur encadrement. De Belfort : Jean-Noël, Jean-Charles, Cyril, Cindy, Frédérick, Baudouin, Julian, Bofei, Étienne, Dhouha, Houssam, Damien, Ali. De Lille : Juliette, Amaury, Yann, Jean-Pierre. De Nantes : David. De Limoges : Philippe, Xavier. De Bordeaux : Yannick, Marc, Andrei.

Merci aussi aux auteurs des différents outils utilisés pour les travaux effectués et à Stéphane pour le gabarit L^AT_EX du manuscrit de thèse suivant la charte de l'école doctorale SPIM.

Vous autres proches et compagnons de galère, j'espère exprimer ma reconnaissance lors de nos rencontres simplement en passant de bons moments ensemble ☺. Merci pour votre soutien.

*Le monde entier est un cactus
Il est impossible de s'asseoir
Dans la vie, il y a qu'des cactus
Moi je me pique de le savoir
Aïe aïe aïe ! Ouille ! Aïe aïe aïe !*

*Dans leurs cœurs, il y a des cactus
Dans leurs porte-feuilles, il y a des cactus
Sous leurs pieds, il y a des cactus
Dans l'heure qu'il est, il y a des cactus
Aïe aïe aïe ! Ouille ouille ! Aïe !*

*Pour me défendre de leurs cactus
À mon tour j'ai pris des cactus
Dans mon lit, j'ai mis des cactus
Dans mon slip, j'ai mis des cactus
Aïe aïe aïe ! Ouille ! Aïe aïe aïe !*

*Dans leurs sourires, il y a des cactus
Dans leurs ventres, il y a des cactus
Dans leurs bonjours, il y a des cactus
Dans leurs cactus, il y a des cactus
Aïe aïe aïe ! Ouille ! Aïe !*

*Le monde entier est un cactus
Il est impossible de s'asseoir
Dans la vie, il y a qu'des cactus
Moi je me pique de le savoir
Aïe aïe aïe ! Ouille ! Aïe ! Aïe ! Aïe !*

Aïe... Ouille... Aïe... Ouille... Aïe... Ouille...

Les Cactus, écrit par Jacques Lanzmann et composé par Jacques Dutronc

TABLE DES MATIÈRES

1	Introduction	3
1.1	Contexte	3
1.1.1	Projet CAPLOC et collaborations	4
1.1.2	Estimer la structure 3D	4
1.1.2.1	Capteur passif par vision	6
1.2	Objectifs et contributions de la thèse	7
1.2.1	Contribution à la modélisation 3D multi-vues fisheye	8
1.2.2	Amélioration du positionnement GNSS- <i>ranging</i>	9
1.3	Solutions proposées	9
1.3.1	Difficultés liées à des capteurs fisheye orientés vers le ciel	9
1.3.2	Les configurations expérimentales utilisées	10
1.3.3	Repère image et repère 3D	12
1.3.4	Plan de la thèse	12
2	État de l'art : Stéréovision omnidirectionnelle	15
2.1	Introduction	15
2.2	Perception 3D omnidirectionnelle par vision	16
2.2.1	Rotation d'un capteur de stéréovision classique	17
2.2.2	Capteur catadioptrique	17
2.2.3	Caméra à objectif fisheye	18
2.2.3.1	Traitements via une rectification des distorsions fisheye	20
2.2.3.2	Traitements sans rectification des distorsions fisheye	20
2.2.4	Stéréoscopes hybrides	22
2.2.5	Bilan	23
2.3	Bibliographie sur le calibrage sphérique fisheye	23
2.3.1	Unification d'un modèle pour catadioptrique et fisheye	24
2.3.2	Géométrie épipolaire des caméras sphériques	24
2.3.3	Calibrage manuel	25
2.3.3.1	Calibrage fort	25
2.3.3.2	Calibrage faible	26
2.3.3.3	Stéréovision omnidirectionnelle avec court entraxe	26
2.3.3.4	Stéréovision omnidirectionnelle avec grand entraxe	27
2.3.4	Calibrage automatique	27
2.4	<i>Structure from Motion</i>	28

2.4.1	Méthodes d'estimation de la pose	29
2.4.1.1	Méthodes globales basées sur l'apparence	29
2.4.1.2	Méthodes par suivi de <i>features</i>	30
2.5	État de l'art de la mise en correspondance stéréo	31
2.5.1	Méthodes éparses et denses	32
2.5.1.1	Mise en correspondance éparses	32
2.5.1.2	Mise en correspondance dense	33
2.5.2	Méthodes locales et globales	33
2.5.2.1	Mise en correspondance locale	33
2.5.2.2	Mise en correspondance à optimisation globale	34
2.5.2.3	Méthodes globales basées sur les graphes	36
2.6	Conclusion	37
3	Calibrage fisheye automatique	39
3.1	Introduction	39
3.1.1	Généralités	41
3.1.2	Comparaison des profils des modèles de projection centrale	43
3.1.3	Estimation de la relation entre repère capteur et repère image	44
3.2	Calibrage d'un système multi-vues fisheye	46
3.2.1	Extraction automatique de paires de points correspondants	47
3.2.2	Projections étudiées formulées pour la méthode des 9 points	49
3.2.3	Algorithme des 9 points adapté à de nouveaux modèles fisheye	50
3.2.4	RANSAC et variantes	52
3.2.4.1	Paramétrage du RANSAC	54
3.2.4.2	LO-RANSAC robuste appliqué à l'algorithme des 9 points	56
3.2.5	Affinement de l'étalonnage par Levenberg-Marquardt	57
3.3	Extraction de la géométrie épipolaire	57
3.3.1	Calcul d'une courbe épipolaire	58
3.3.2	Calcul des épipôles et test de stabilité	58
3.3.3	Calcul des rotations internes pour un alignement épipolaire	59
3.4	Évaluation du calibrage automatique	62
3.4.1	Scènes évaluées	62
3.4.1.1	Scènes simulées	63
3.4.1.2	Scènes réelles	64
3.4.2	Critères évalués	65
3.4.3	Seuil d'erreur angulaire	66
3.4.4	Choix du bon modèle de projection fisheye	68
3.4.5	Efficacité et stabilité entre RANSAC et LO-RANSAC	70
3.5	Conclusion	71
4	Mise en correspondance multi-vues fisheye pour la 3D	73

4.1	Introduction	74
4.2	Mise en correspondance stéréo fisheye	74
4.2.1	Indexation des courbes épipolaires	75
4.2.2	Calcul des positions 3D	76
4.2.2.1	Projection des rayons issus des deux capteurs	76
4.2.2.2	Utilisation des distances à l'intérieur des plans épipolaires	76
4.2.3	L'algorithme de Fortsmann	77
4.2.3.1	Généralités	77
4.2.3.2	Évaluation des correspondances et choix des paramètres optimaux	78
4.2.4	De l'algorithme de Forstmann au graphe 3D	83
4.2.4.1	Généralités	83
4.2.4.2	Évaluation de la programmation dynamique avec graphe 3D	85
4.2.5	Évaluation sur des images réelles	87
4.2.6	Conclusion	90
4.3	Correspondance mono-caméra fisheye couleur	91
4.3.1	Adaptation de notre méthode graphe 3D	91
4.3.2	Évaluation de séquences d'images réelles	93
4.4	Analyse des nuages de points 3D	96
4.4.1	Post-traitement des nuages de points 3D	98
4.4.2	Fusion des nuages de points 3D	100
4.4.3	Cas mono-caméra	100
4.4.4	Cas binoculaire	102
4.4.5	Évaluation de l'odométrie visuelle	105
4.5	Conclusion	108
4.6	Annexe : Estimation de la largeur de la rue	109
5	Application : Amélioration de la localisation GNSS	111
5.1	Introduction	112
5.2	Informations et données satellitaires	113
5.2.1	Heures	113
5.2.2	Données élémentaires	114
5.2.3	<i>Dilution Of Precision</i>	115
5.3	Mise en application du modèle 3D estimé pour GNSS	116
5.3.1	Effet multi-trajet en milieu urbain	116
5.3.2	État de l'art sur l'étude des signaux NLOS en milieu urbain	117
5.3.2.1	Modèle de propagation des signaux GNSS en milieu urbain	118
5.3.2.2	Amélioration de la localisation par exclusion NLOS	118
5.3.2.3	Amélioration de la position sans corriger les pseudodistances	119
5.3.2.4	Révision des pseudodistances NLOS pour améliorer la position	120
5.3.3	Modèle d'erreur et correction des pseudodistances	122

5.3.3.1	Modèle de réflexion simple limité à une réflexion	122
5.3.3.2	Modèle de réflexion à une, deux ou trois réflexions	123
5.4	Extraction des plans	125
5.4.1	État de l'art des méthodes de <i>plane fitting</i>	125
5.4.2	<i>Plane fitting</i> en 3D basé sur le RANSAC	126
5.4.3	<i>Plane fitting</i> par transformée de Hough 2D sur cartographie	127
5.4.3.1	Estimation de l'empreinte au sol	127
5.4.3.2	Détection des murs par transformée de Hough	128
5.4.4	Évaluation des informations extraites sur scènes réelles	131
5.4.4.1	Critères d'évaluation	131
5.4.4.2	Résultat des évaluations	132
5.5	Représentation des satellites dans la partie vision	134
5.5.1	Projection des satellites et rayons GNSS sur la trace au sol	134
5.5.2	Projection des satellites dans les images fisheye	134
5.5.3	Création de masques de visibilité d'après les plans détectés	135
5.6	Évaluation en conditions réelles	135
5.6.1	Procédure d'évaluation	136
5.6.2	Scènes et positions évaluées	137
5.6.3	Position améliorée avec correction des pseudodistances	137
5.7	Conclusion	141
6	Conclusion générale et perspectives	145
6.1	Bilan	145
6.1.1	Résultats	146
6.1.2	Contributions	148
6.2	Perspectives	149

NOTATIONS

La convention définie ci-après est employée dans tout ce manuscrit :

$\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$	les ensembles usuels
$\llbracket a; b \rrbracket$	un intervalle d'entiers
$[a; b]$	un intervalle de réels
$\mathcal{A}, \mathcal{B}, \mathcal{C} \dots$	pour un ensemble d'éléments (images, acquisitions GPS, nuages 3D, instants, etc)
$a, b, c \dots A, B, C \dots$	pour un élément ou un scalaire
$\mathbf{a}, \mathbf{b}, \mathbf{c} \dots \mathbf{A}, \mathbf{B}, \mathbf{C} \dots$	pour un vecteur ou un axe
$A, B, C \dots$	pour une matrice
\dots^T	pour un vecteur ou une matrice transposée
\mathfrak{B}	pour une base
\mathfrak{R}	pour un repère
$\widetilde{\dots}$	pour une approximation
$\widehat{\dots}$	pour une moyenne
\dots^*	pour une valeur initiale ou inexacte
$\ \dots\ $	pour une norme
$\dots \wedge \dots$	pour un produit vectoriel

Cas particuliers en absence de définition dans le texte :

Id_n	la matrice identité carrée de dimensions (n, n)
$\mathbf{x}_{l \times c}$	un vecteur ou une matrice de dimensions (l, c) , dont tous les coefficients valent x
$\text{Diag}(a, b, c \dots)$	une matrice diagonale dont les éléments sont a, b, c , etc
R	une rotation sous forme matricielle
\mathbf{t}	une translation sous la forme du vecteur correspondant
E	une matrice essentielle
F	une matrice fondamentale

Note : un nuage de points sera noté avec la notation de l'ensemble d'éléments, on pose par exemple \mathcal{N} un nuage de points. Chaque élément du nuage est un point, on considère par exemple p un point de \mathcal{N} .

Par ailleurs, dans les formules, on privilégiera la notation du vecteur correspondant au point pour plus de lisibilité. On pourra aussi appeler ce point \mathbf{p} .

Sans précision complémentaire, les repères et les bases définis sont orthonormés et directs.

INTRODUCTION

La reconstruction 3D à partir de la vision est un domaine très actif. Ce manuscrit relate de la modélisation à grande échelle d'environnements urbains. Cette reconstruction est expérimentée en conditions réelles et difficilement maîtrisables pour une application visant à améliorer la précision de la localisation GNSS en ville.

1.1/ CONTEXTE

Cette thèse s'inscrit dans le cadre du projet CAPLOC¹ (Combinaison de l'Analyse d'images et de la connaissance de la Propagation des signaux pour la LOCALisation). En milieu urbain, les signaux GNSS sont fortement perturbés par les réflexions multiples sur les façades des bâtiments présents le long du trajet d'un véhicule. Ces perturbations modifient directement les valeurs de mesure de pseudodistance de satellite à véhicule et mènent à une estimation imprécise de la position GNSS, voire conduisent à l'indisponibilité du service de localisation quand les signaux sont bloqués. L'objet de ce projet est de montrer comment il est possible d'améliorer la localisation GNSS d'un véhicule en environnement urbain à partir de son modèle 3D estimé par vision; ce modèle permettant d'intégrer des connaissances sur les caractéristiques de propagation des signaux dans le processus d'estimation de la position du véhicule.

Cette méthode de correction a été protégée par un brevet déposé le 29/09/2006 : [Dispositif de localisation](#).

Numéro de publication : WO/2008/040896.

Date de publication internationale : 10 avril 2008.

Ce mémoire expose les travaux menés sur l'estimation du modèle 3D d'un environnement urbain par la mise en correspondance du contenu de plusieurs images provenant d'un système de perception à objectif fisheye. La méthode proposée est fondée sur la construction d'un graphe 3D et sur l'usage d'une optimisation par programmation dynamique. Nous l'avons appliquée avec succès à des images provenant d'un système stéréoscopique et d'un système mono-caméra en mouvement. Les modèles 3D estimés sont exploités dans le dernier chapitre pour montrer

1. <http://caploc.ifsttar.fr/>

que, sous certaines conditions, une correction géométrique des erreurs de pseudodistance mène à une position GNSS plus précise.

1.1.1/ PROJET CAPLOC ET COLLABORATIONS

CAPLOC est un projet français financé par le Ministère de l'Écologie, du Développement Durable et de l'Énergie (MEDDE). Il fait partie du Programme de REcherche et D'Innovation dans les Transports terrestres (PREDIT). Les laboratoires partenaires pour ce projet sont l'IFSTTAR-COSYS-LEOST, coordinateur du projet localisé à Villeneuve-d'Ascq (Nord) et l'IRTES-SET, localisé à Belfort (Territoire-de-Belfort).

Le programme de CAPLOC est divisé en trois tâches techniques :

1. Analyse d'images pour la détection et la caractérisation des obstacles perturbant la réception des signaux satellitaires ;
2. Construction d'un modèle 3D urbain par analyse d'une séquence d'images ;
3. Utilisation des informations de l'image pour une localisation GNSS sûre et précise.

Dans ce contexte, deux thèses applicatives ont été entreprises pour les deux premières tâches :

1. [Segmentation d'images par combinaison adaptative couleur/texture et classification de pixels : Application à la caractérisation de l'environnement de réception de signaux GNSS](#), soutenue le 3 octobre 2013 par Dhouha ATTIA [Attia, 2013].
2. [Construction de modèles 3D à partir de données vidéo fisheye—Application à la localisation en milieu urbain](#), qui est l'objet de ce manuscrit.

Les résultats des travaux de [Attia, 2013] permettent de déterminer l'état de réception des satellites GPS à partir de la segmentation des images en deux catégories : ciel et non ciel. La localisation GNSS est ainsi améliorée en tenant compte l'état de réception, bloqué ou réfléchi, des satellites reçus.

Le contenu de ce manuscrit fait partie des tâches 2 et 3 du projet CAPLOC : il consiste à générer un modèle 3D de l'environnement urbain autour du véhicule par la vision et à appliquer le principe de correction des informations GNSS à partir du modèle généré pour finalement aboutir à un meilleur positionnement.

1.1.2/ ESTIMER LA STRUCTURE 3D

De nombreuses techniques existent pour estimer la structure 3D d'une scène urbaine. L'estimation est assurée en analysant les nuages de points 3D générés par deux types de capteur :

- Les télémètres laser : des capteurs actifs émettant des rayons qu'ils reçoivent après réflexion sur les structures et à partir desquels ils estiment la distance des objets ;
- Les caméras vidéo.

Le lidar (mot-valise issu de l'anglais *light* et *radar*) est un capteur qui exploite un faisceau laser afin de déterminer la position 3D du point de la scène sur lequel le faisceau s'est réfléchi. Le



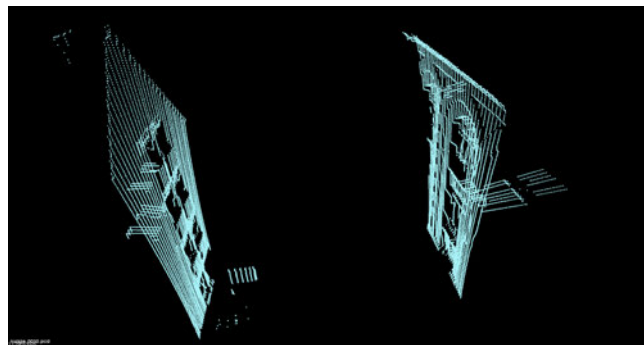
FIGURE 1.1 – Exemple de lidar 2D Sick.

lidar ne permet pas de capter les couleurs des surfaces sur lesquelles le laser se réfléchit, mais il a l'avantage de fonctionner de jour comme de nuit. Un lidar classique peut avoir une portée proche de 100 mètres avec une très bonne précision ($\pm 5\text{cm}$ à plusieurs dizaines de mètres de distance).

Lorsque ce faisceau est en rotation, il associe une distance à une position angulaire et fournit un nuage de points 3D référencé dans un repère centré sur la position du lidar. La figure 1.1 montre un lidar de la marque Sick. Ce lidar ne possède qu'un seul plan de mesure et le nuage de points produit correspond à l'intersection de ce plan avec les surfaces présentes dans la scène et directement visibles par le capteur.



(a) Velodyne HDL-32E.



(b) Tranche de rue pour une position fixe.

FIGURE 1.2 – Lidar rotatif Velodyne et exemple de nuage de points acquis à Belfort.

Le lidar Velodyne présenté sur la figure 1.2 est un exemple de lidar plus perfectionné, composé de 32 plans avec une précision absolue de 2cm et une portée comprise entre 80m et 100m. Il possède un champ de vision horizontal de 360° et un champ de vision vertical de 40° . Ce capteur fournit un nuage jusqu'à 700 000 points 3D par seconde. Ce lidar peut être associé à un récepteur GPS qui lui permet de synchroniser les acquisitions et de géo-référencer chaque nuages de points.

Les lidars sont souvent embarqués sur des véhicules pour des applications de détection d'obstacles, pour produire des modèles d'environnement à grande échelle [Zhang et al., 2015] et pour des applications de télémétrie et télédétection aéroportée tel que dans les travaux de [Tarsha Kurdi, 2008].

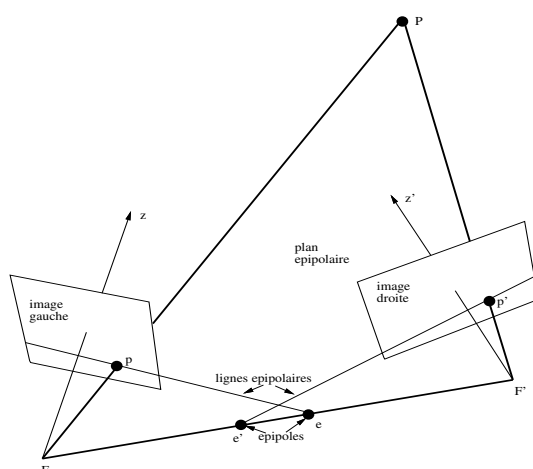


FIGURE 1.3 – Projection perspective d'un point dans deux caméras. L'estimation 3D se fait par l'opération inverse à partir des points des deux images. Issu de [Horaud et al., 1995].

1.1.2.1/ CAPTEUR PASSIF PAR VISION

Obtenir un modèle 3D par vision se fait à partir de configurations et d'analyses multi-vues. Lorsque le système est composé de deux caméras, nous parlons de stéréovision. La figure 1.3 illustre la modélisation du principe de stéréovision. Les informations des deux vues sont appariées pour finalement estimer les structures 3D de la scène captée. En effet, si le système est calibré, la mise en correspondance rend possible l'estimation de la contre-projection des points correspondants et celle de leur position 3D par triangulation. La précision est moins bonne qu'avec un lidar mais l'usage de caméras permet d'associer des informations colorimétriques et de texture au nuage de points 3D et donc de fournir un modèle plus riche. La précision dépend de l'entraxe entre les deux caméras et de l'éloignement des structures aux caméras. Dans [Zhang et al., 2011], l'auteur étudie le choix théorique de l'entraxe optimal pour réduire l'erreur d'estimation des points.

Lorsque le système ne compte qu'une seule caméra, le principe décrit précédemment reste applicable mais sur un couple d'images acquises au cours du déplacement de la caméra. Dans ce contexte, la structure de l'environnement est donc retrouvée grâce au mouvement de la caméra. On parle alors de *Structure from Motion*. Un nuage de points similaire au cas stéréo est produit mais à un coefficient d'échelle près. Pour retrouver ce coefficient, la caméra est généralement associée à un autre capteur proprioceptif tel qu'un odomètre ou une centrale inertielle.

Un système vidéo qui serait composé d'une ou deux caméras munies d'objectifs traditionnels (objectif dit *pinhole*) ne permet pas d'assurer la surveillance et la reconstruction à 360° autour du système. Pour cela, il est nécessaire

1. de multiplier le nombre de caméras afin de couvrir le champ de vision complet autour du véhicule,
2. d'associer l'objectif à des miroir et des systèmes optiques spécifiques (menant à des capteurs dits catadioptriques)

3. ou d'utiliser des objectifs à projection sphérique (objectif fisheye).

Beaucoup de travaux proposent de fusionner les informations télémétriques avec la vision. Ils utilisent généralement le lidar pour la structure 3D et les caméras pour les informations de texture et de couleur [Früh et al., 2004, De Luca, 2006, Fleck et al., 2009, Deschaud, 2010]. Toutefois, même dans ce cas, l'une des solutions présentées précédemment est nécessaire pour assurer un champ de vision le plus large entre les deux capteurs.

1.2/ OBJECTIFS ET CONTRIBUTIONS DE LA THÈSE

Ce manuscrit présente une recherche sur la reconstruction omnidirectionnelle de modèles 3D pour une application à la localisation en milieu urbain. L'objectif est d'améliorer la précision de la localisation GPS d'un véhicule en environnement urbain dans lequel les multitrajets sont très importants et altèrent fortement les signaux satellitaires. Avec des mesures dans ces conditions, la précision de la localisation au sol est de l'ordre de 20 mètres. L'intérêt de la démarche est que nous proposons de générer dynamiquement un modèle 3D de l'environnement à partir duquel il est possible d'extraire les caractéristiques de la propagation des signaux dans le canyon urbain pour finalement en tenir compte lors d'une étape de correction de la position du véhicule. Cet objectif se différencie d'autres travaux de la littérature qui utilisent des modèles 3D urbains pré-enregistrés pour accéder à ces propriétés et corriger la position GNSS [Bétaille et al., 2014, Adjrad et al., 2015].

Nous avons choisi d'exploiter un système vidéo composé d'une ou deux caméras munies chacune d'un objectif fisheye, orientées vers le ciel afin d'observer tous les bâtiments autour de leur position. Ce choix permet de réduire le nombre de caméras pour assurer la couverture à 180° et à 360° autour du véhicule. L'intégration de ce système est plus aisée et son coût est inférieur à celui d'un lidar. Les caméras utilisées sont des caméras haute résolution donc même si la résolution finale de notre système fisheye est inférieure à celle du système Stereopolis de l'IGN (décrit dans [Cannelle et al., 2012]), nous montrons qu'elle est suffisante pour améliorer la position GNSS.

Dans [Marais, 2002], l'auteur exploite un stéréoscope traditionnel orienté vers le ciel pour reconstruire la ligne d'horizon et spécifier l'état de chaque satellite (reçu directement ou occulté par un bâtiment). Son application est la localisation GNSS de trains. Ce que nous proposons est une suite logique de ce précédent travail.

Très peu d'auteurs travaillent en configuration de stéréovision fisheye orientée vers le ciel et aucun dans le domaine du positionnement GNSS urbain.

Dans [Zhao et al., 2000], les auteurs reconstruisent des scènes urbaines simples en images de synthèse avec une caméra placée sur le toit d'un véhicule et orientée en direction du ciel. Dans [Herrera et al., 2009, Herrera et al., 2011] les auteurs travaillent en environnement forestier et souhaitent identifier et compter les arbres autour de la position des prises de vue.

Il existe toutefois une application de localisation basée sur la vision fisheye orientée vers le

ciel dans [Ramalingam et al., 2010]. Les auteurs font de la localisation en environnement urbain dans un modèle 3D géoréférencé connu à partir de la détection et la reconnaissance des silhouettes des immeubles en vision omnidirectionnelle en direction du ciel, entre des projections fisheye simulées depuis le modèle et des acquisitions fisheye réelles.

1.2.1/ CONTRIBUTION À LA MODÉLISATION 3D MULTI-VUES FISHEYE

En vision omnidirectionnelle, des auteurs comme [Abraham et al., 2005], [Nishimoto et al., 2007], [Li, 2008], [Arıcan, 2010] ou [Lhuillier, 2011] appliquent les principes de la stéréovision pour modéliser des éléments en 3D avec un large champ de vision.

Retrouver la structure 3D d'un environnement nécessite de produire un nuage de points 3D de la scène. Un tel nuage est estimé en mettant en correspondance le contenu de plusieurs vues. Pour cela, nous proposons une adaptation de la méthode écrite par [Forstmann et al., 2004] au contexte des images fisheye : les appariements sont estimés grâce à l'optimisation d'un graphe 3D construit à partir des pixels de deux courbes épipolaires conjuguées.

Un objectif fisheye déforme fortement la géométrie de la scène lors du processus de projection. Il est important d'en tenir compte lors des tâches de mise en correspondance et de reconstruction précise du nuage de points 3D. Pour y parvenir, nous proposons une technique de calibrage automatique c'est-à-dire une technique qui assure l'estimation de la géométrie épipolaire à partir de deux vues fisheye d'une scène ; ces deux vues pouvant provenir d'un stéréoscope ou d'une unique caméra s'étant déplacée.

Ce calibrage fait l'hypothèse que la projection fisheye répond à un modèle sphérique. Nous l'avons décrit pour des modèles de projection équisolide, équidistant et stéréographique.

La géométrie épipolaire une fois estimée, nous définissons géométriquement tous les couples de courbes épipolaires conjuguées à partir desquelles le graphe 3D est construit et dans lequel nous cherchons le plus court chemin par programmation dynamique. La profondeur du graphe 3D est définie par le nombre de paires de courbes utilisées. Par ailleurs, ce graphe ne dépend que de deux paramètres pour lesquels nous définissons le sous-espace qui mène aux résultats optimaux.

Tous nos développements sont évalués objectivement sur une base d'images de synthèse. Par ailleurs, nous en montrons le bon fonctionnement en les appliquant sur une base d'images réelles acquises :

1. à Lille par un stéréoscope fisheye
2. et à Belfort par une seule caméra en mouvement sur un véhicule.

Sur le plan de l'application, nous proposons une chaîne complète de post-traitements des nuages de points 3D locaux obtenus. Cette chaîne assure le débruitage des nuages de points, la densification par fusion de nuages de points, la détection des plans des bâtiments dans la scène et leur exploitation pour l'amélioration du positionnement GNSS.

1.2.2/ AMÉLIORATION DU POSITIONNEMENT GNSS-*ranging*

À ce jour et à notre connaissance, aucune publication ne fait état de l'usage d'un système de vision 3D (tout particulièrement fisheye) pour estimer une structure urbaine exploitée ensuite pour augmenter la précision de la position GNSS en corrigeant géométriquement les signaux reçus par réflexion. La position GNSS-*ranging* est calculée à partir des pseudodistances des satellites reçus (distance estimée du satellite au récepteur). Or, cette information est calculée en fonction de l'heure d'émission par le satellite et l'heure de réception, considérant une visibilité directe. En milieu urbain de nombreux obstacles cachent les satellites pour lesquels l'information est souvent non plus reçue de manière directe mais de manière réfléchie, voire plusieurs fois en multitrajets. Dans ce manuscrit on s'intéresse au cas simplifié avec l'hypothèse d'un seul trajet par signal reçu, et particulièrement aux signaux NLOS (*Non Line Of Sight*, reçus de manière indirecte). Dans ce contexte, la distance parcourue par le rayon est plus grande et donc le satellite est considéré plus loin qu'il ne l'est en réalité. La localisation est estimée par trilatération à partir des pseudodistances et s'en retrouve alors faussée. Il s'agit par conséquent d'une part de détecter les signaux GNSS NLOS, et d'autre part de mesurer et corriger leur retard grâce au modèle généré. Il peut y avoir une ou plusieurs réflexions, en fonction de l'élévation du satellite, de la hauteur des obstacles et de leur distance au récepteur. Pour valider les résultats de localisation, on utilise un récepteur GPS bas-coût de type u-blox comme système à corriger et un récepteur GPS RTK pour la vérité terrain. Des expérimentations, validations, évaluations et mesures de précision sont proposées pour toutes les étapes, avec différentes méthodes et des scènes simulées et réelles.

1.3/ SOLUTIONS PROPOSÉES

Les solutions proposées sont basées sur des vues fisheye orientées vers le ciel avec un grand entraxe pour optimiser la précision des structures d'échelle urbaine.

1.3.1/ DIFFICULTÉS LIÉES À DES CAPTEURS FISHEYE ORIENTÉS VERS LE CIEL

Utiliser les caméras fisheye orientées vers le ciel avec un grand entraxe a différents impacts. Le centre des images fisheye présente moins de distorsions que les bords mais correspond au ciel et n'est donc pas exploité. Les structures sont visualisées sur les bords des images fortement distordus.

Pour le calibrage, les caméras doivent visualiser des éléments communs facilement reconnaissables et les chapitres 2 et 3 montrent qu'il n'est pas simple de couvrir le champ de vision de deux caméras fisheye avec un large entraxe en même temps. La procédure de calibrage est donc plus complexe avec les caméras posées sur le toit du véhicule et orientées en direction du ciel.

La mise en correspondance nécessite le calcul d'une mesure de corrélation entre les pixels des deux images. Cette mesure est bien entendu moins représentative lorsqu'elle est estimée entre

deux pixels des bords des images pour lesquels les distorsions sont extrêmes. Or nous devons atteindre une précision de reconstruction 3D métrique à partir des pixels des bords des images fisheye, issus des structures urbaines. La robustesse de la mise en correspondance est donc mise à rude épreuve. Estimer les paramètres les plus exacts du modèle de projection de la caméra fisheye utilisée est une étape difficile mais indispensable si nous voulons une précision suffisante.

1.3.2/ LES CONFIGURATIONS EXPÉRIMENTALES UTILISÉES

L'estimation de la structure 3D de l'environnement exploite un système de perception vidéo composé de caméras à objectif fisheye (voir figure 1.4). Les objectifs fisheye, associés à des caméras numériques dont la résolution est élevée, offrent un bon compromis entre résolution et précision de la reconstruction et le nombre de caméras nécessaires pour assurer la surveillance complète de l'environnement.

[Zhang et al., 2011] proposent une étude théorique pour trouver l'entraxe optimal en stéréovision qui minimise les erreurs de reconstruction 3D. Cette étude considère la précision de la reprojection 3D et analyse les résultats pour des angles d'observation variables. Ils proposent une expression de l'entraxe optimal proportionnelle à la profondeur du point 3D considéré et tenant compte d'erreurs d'échantillonnage dans les deux images. À travers un exemple, pour une focale de 17mm et une profondeur de 100mm, ils arrivent à un entraxe optimal de 200mm. L'étude du choix de l'entraxe optimal est trop complexe pour être détaillée et étendue ici au cas fisheye, il dépend de la position du pixel considéré, augmente avec une focale plus courte, etc. En fisheye la longueur focale est extrêmement courte et il faut tenir compte de la fonction de projection interne utilisée avant d'appliquer la formulation de [Zhang et al., 2011]. Dans tous les cas l'entraxe optimal est supérieur à la profondeur prise en compte pour le calcul. Dans l'application visée de reconstruction urbaine à partir de la vision fisheye, une précision optimale est cherchée pour les distances supérieures ou égales à 2m du système de vision. L'entraxe devrait donc être supérieur à 2 mètres. En pratique il est inconcevable d'équiper un véhicule d'un dispositif de stéréovision plus long. Pour ces raisons l'entraxe choisi pour le stéréoscope est fixé à 2 mètres, cet entraxe est conservé pour les acquisitions par système mono-caméra pour comparer les dispositifs dans des conditions les plus proches possibles.

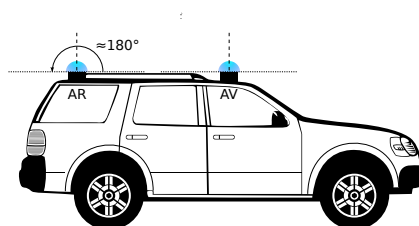


FIGURE 1.4 – Configuration du stéréoscope fisheye. Les deux caméras sont placées sur le toit du véhicule le long de l'axe longitudinal.

Les travaux effectués et décrits dans ce manuscrit ont donné lieu à plusieurs configurations expérimentales. Les principales comprennent des récepteurs GPS pour l'application CAPLOC.

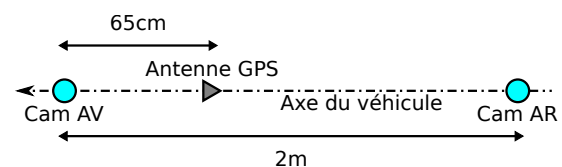
Au début de cette recherche, nous avons estimé qu'une synchronisation de l'ordre du centième de seconde entre les données 3D et GNSS est suffisante. L'idée est que, en mouvement, les décalages de synchronisation peuvent être pris en compte pour précisément localiser le récepteur GNSS dans le modèle 3D en fonction de la vitesse du véhicule et palier à cette difficulté. Les configurations étudiées sont :

(Cfg 1) Système stéréo binoculaire composé de deux caméras JAI délivrant des images en niveaux de gris (monture Nikon, résolution de 3296×2520 pixels, assemblage de deux capteurs CCD). Elles sont munies d'un objectif fisheye Sigma 4.5mm F2.8 EX HSM IF (180° et modèle de projection angle équisolide d'après les données constructeurs). Le disque image obtenu a un diamètre d'environ 2310 pixels. L'entraxe entre les deux caméras est de deux mètres. C'est le camion de l'IFSTTAR (Villeneuve-d'Ascq), voir figure 1.5. Les images sont enregistrées avec une fréquence de 17Hz et synchronisées avec une horloge GPS. Une position GPS étant calculée chaque seconde, ceci permet une extraction des images correspondantes avec un décalage de synchronisation maximal de 0,059s.

Partie localisation GPS : Le camion est équipé d'un récepteur GPS bas-coût u-blox qui stocke les données brutes (pseudodistances) pour pouvoir les traiter et améliorer l'estimation de la position.



(a) Camion de l'IFSTTAR.



(b) Disposition des capteurs utilisés.

FIGURE 1.5 – Camion en configuration binoculaire en niveaux de gris (Cfg 1). La caméra du milieu est une caméra couleur pour la de segmentation du ciel de [Attia, 2013].

(Cfg 2) Quelques expérimentations ont été faites avec un stéréoscope composé de deux caméras Pixellink en couleurs (monture C, résolution de 1280×1024 pixels) munies d'objectifs fisheye Fujinon FE185C057HA-1 (185° et projection équidistante d'après les données constructeurs). Le disque utile obtenu dans les images a un diamètre d'environ 920 pixels.

(Cfg 3) Système mono-caméra couleur composé d'un appareil photo reflex Fujifilm S5 Pro (monture Nikon, résolution de 4281×2871 pixels), avec objectif fisheye Sigma 4.5mm F2.8 EX HSM IF (180° et modèle de projection angle équisolide d'après les données constructeurs). Le disque image obtenu a un diamètre d'environ 2350 pixels.

(Cfg 4) Système mono-caméra couleur composé d'un appareil photo reflex Nikon D3100 (résolution de 4608×3072 pixels), avec objectif fisheye Sigma 4.5mm F2.8 EX HSM IF (180° et modèle de projection angle équisolide d'après les données constructeurs). Le disque image

obtenu a un diamètre d'environ 2540 pixels. C'est la configuration utilisée sur la SetCar de l'IRTES-SET (Belfort), voir figure 1.6. Les images sont acquises tous les deux mètres par un odomètre qui déclenche l'appareil photo. L'odomètre est construit sur la base d'un compteur incrémental sur roue avec une carte Arduino et un relais. La vitesse du véhicule est limitée à 15km/h pour ne pas saturer l'enregistrement en rafale des images par l'appareil photo.

Partie localisation GPS : La voiture est équipée d'un récepteur GPS bas-coût u-blox qui stocke les données brutes (pseudodistances) pour pouvoir les traiter et améliorer l'estimation de la position. Les positions vérités sont enregistrées grâce à un récepteur GPS RTK Magellan ProFlex 500 précis à quelques centimètres près.

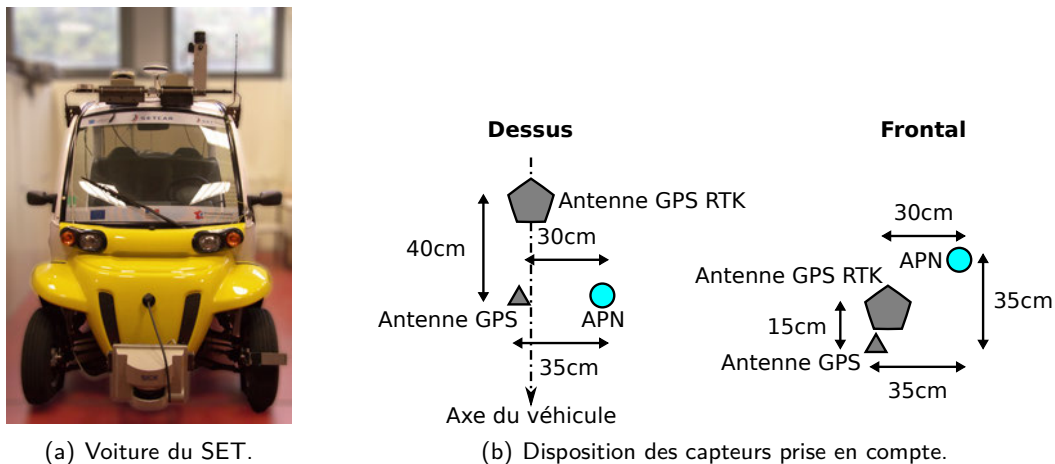


FIGURE 1.6 – SetCar en configuration monoculaire couleur avec un appareil photo (Cf g 4).

1.3.3/ REPÈRE IMAGE ET REPÈRE 3D

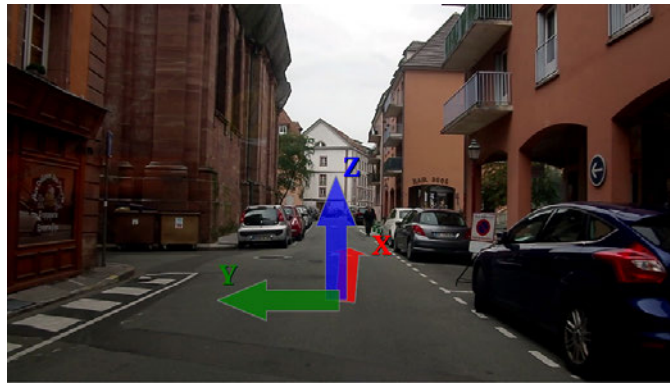
On fait en sorte d'avoir l'orientation de l'image telle que l'axe X soit dans la direction du véhicule et de la rue. Dans ces travaux, l'orientation des axes X et Y du repère caméra est posée identique à celle des axes des images codées en informatique, usuellement l'axe X vers la droite et Y vers le bas. Le repère image est illustré figure 1.7(c).

Dans le repère 3D, on travaille avec un repère direct. Les axes X et Y correspondent aux directions des mêmes axes vus de l'image comme on le voit en figure 1.7. L'axe X correspond à la direction du véhicule, et l'axe Y est dirigé du côté de la cathédrale dans l'exemple illustré (rue du Général Roussel à Belfort). Le repère 3D étant direct, l'axe Z est orienté en direction du ciel, ce qui a l'avantage d'être naturel dans le sens où les hauteurs des structures s'en trouvent positives. Le repère 3D dans la rue du Général Roussel est aussi illustré figure 1.7(a), avec X en face, Y vers la gauche (côté cathédrale), et Z vers la haut.

1.3.4/ PLAN DE LA THÈSE

La structure de ce manuscrit est la suivante :

— Le chapitre 2 donne un contexte et un état de l'art de la vision omnidirectionnelle, avec en



(a) Position et repère dans le monde réel.



(b) Carte et repère local d'une caméra.



(c) Image fisheye correspondante.

FIGURE 1.7 – Repères caméra à la position du véhicule dans la situation rue du Général Roussel. L'axe Z représente le haut et est en direction du ciel.

particulier les différentes fonctions de projection fisheye. Il est suivi d'une bibliographie sur la stéréovision, contenant d'une part le calibrage et d'autre part la mise en correspondance permettant d'estimer un nuage 3D et à la structure d'un environnement.

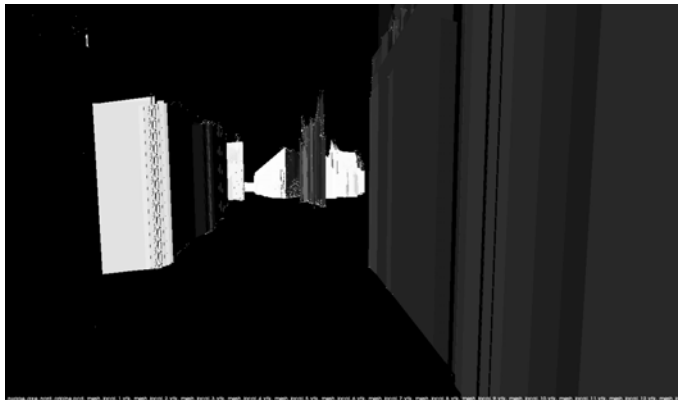
- Le chapitre 3 décrit la méthode de calibrage entièrement automatique déployée et expérimentée dans le cadre des travaux présentés. Des réflexions et évaluations sont présentées sur les différents algorithmes qui interviennent dans le processus de calibrage automatique des paramètres internes et externes des capteurs fisheye.
- Le chapitre 4 expose l'algorithme de mise en correspondance multi-vues utilisé pour mesurer les disparités angulaires sphériques et finalement la position 3D des points de la scène. Il décrit les adaptations à faire pour le cas fisheye et détermine le sous-espace optimal des paramètres grâce à une évaluation menée sur plusieurs séquences de synthèse. Nous étudions ensuite l'exploitabilité de cette méthode dans le cas d'un système mono-caméra couleur et les différents choix qui ont été faits pour filtrer les nuages de points locaux. Ce chapitre se termine par la description d'une technique d'assemblage de nuages de points successifs en vue d'obtenir un nuage de points 3D du trajet complet.
- Le chapitre 5 présente d'une part une analyse des informations contenues dans les nuages 3D dont l'objectif est d'extraire le plan de façades des bâtiments et leurs empreintes. D'autre

part, la fin de ce chapitre est consacré à l'utilisation de ces nuages de points 3D pour améliorer le positionnement GNSS.

- Enfin, la conclusion chapitre 6 résume les contributions et les extensions possibles à ces travaux et reporte les problèmes qui demeurent.



(a) Nuage de points.



(b) Plans issus du *plane fitting*.

FIGURE 1.8 – Résultat de la reconstruction 3D du point de vue de la situation rue du Général Roussel. La caméra d'acquisition fisheye est placée à environ 2m de hauteur.

Le but de cette thèse applicative est donc de modéliser l'environnement en 3D à partir de plusieurs vues fisheye. La figure 1.8 est un exemple de modèle obtenu sous forme de nuage de points et de plans extraits. Les méthodes sont testées et comparées en conditions réelles et extérieures. Les évaluations ne se cantonnent pas à des cas idéaux ou parfaitement modélisés sur images de synthèse. À ce stade de la recherche il ne s'agit pas de créer le modèle 3D en temps réel, même si rien n'empêche de réfléchir à des implémentations temps réel des algorithmes développés.

ÉTAT DE L'ART : STÉRÉOVISION OMNIDIRECTIONNELLE

Pour construire le modèle 3D à grande échelle le projet définit une configuration de vision omnidirectionnelle : la stéréovision fisheye. Les caméras utilisées sont orientées en direction du ciel pour voir les structures tout autour du point d'observation.

Sommaire

2.1	Introduction	15
2.2	Perception 3D omnidirectionnelle par vision	16
2.2.1	Rotation d'un capteur de stéréovision classique	17
2.2.2	Capteur catadioptrique	17
2.2.3	Caméra à objectif fisheye	18
2.2.4	Stéréoscopes hybrides	22
2.2.5	Bilan	23
2.3	Bibliographie sur le calibrage sphérique fisheye	23
2.3.1	Unification d'un modèle pour catadioptrique et fisheye	24
2.3.2	Géométrie épipolaire des caméras sphériques	24
2.3.3	Calibrage manuel	25
2.3.4	Calibrage automatique	27
2.4	Structure from Motion	28
2.4.1	Méthodes d'estimation de la pose	29
2.5	État de l'art de la mise en correspondance stéréo	31
2.5.1	Méthodes éparées et denses	32
2.5.2	Méthodes locales et globales	33
2.6	Conclusion	37

2.1/ INTRODUCTION

Le contexte présenté chapitre 1 a besoin d'un champ de vision dans toutes les directions pour modéliser la scène complète. L'environnement doit être représenté à l'échelle réelle avec une précision métrique, suffisante pour modéliser les structures. La solution retenue est basée sur la vision et le principe de stéréovision. La spécificité du projet CAPLOC est d'utili-

ser une perception visuelle omnidirectionnelle, où les caméras sont orientées en direction du ciel.

Nous distinguons trois étapes fondamentales pour retrouver le relief de la scène en stéréovision :

1. **Le calibrage**, consiste à retrouver les paramètres des capteurs :
 - (a) l'estimation des paramètres intrinsèques des caméras.
 - (b) l'estimation de la relation entre les points de vues :
 - cas stéréo ou multi-capteurs** : le calibrage extrinsèque pour estimer le passage entre les repères des caméras à l'échelle réelle
 - OU**
 - cas monoculaire** : l'estimation de la pose entre des vues décalées temporairement, cette estimation visuelle ne suffit pas à connaître l'échelle réelle.
2. **La mise en correspondance** (ou appariement) pour retrouver les points homologues entre les deux images (les projections des mêmes points de la scène).
3. **La reconstruction 3D**, on est capable d'estimer les structures 3D à partir des paramètres des capteurs par triangulation des points correspondants dans les images.

La section 2.2 énonce les travaux d'auteurs ayant cherché à modéliser l'environnement en 3D à partir de différents types de capteurs de vidéo omnidirectionnelle. La section 2.3 donne les principes du calibrage pour ces capteurs. La section 2.4 introduit l'extension et l'utilisation des estimateurs des paramètres de calibrage externes pour le *Structure from Motion*. Enfin, la section 2.5 présente de nombreuses méthodes de mise en correspondance de la littérature.

2.2/ PERCEPTION 3D OMNIDIRECTIONNELLE PAR VISION

Pour avoir une vision omnidirectionnelle il est possible d'utiliser un réseau ou une ceinture de caméras. [Deschaud, 2010] présente par exemple la plate-forme Stereopolis V2 de l'IGN qui comporte une ceinture de caméras (ainsi que des lidars).

Pour réduire le nombre de caméras, certains auteurs utilisent des systèmes omnidirectionnels dioptriques (à lentille) ou catadioptriques (à lentille et miroir) offrant un champ de vision pouvant atteindre $360^\circ \times 180^\circ$. Les capteurs omnidirectionnels ont l'avantage de voir davantage d'informations sur la scène que des capteurs classiques. [Mouaddib, 2005b, Mouaddib, 2005a, Gonzalez-Barbosa, 2004, Ragot, 2009] présentent différentes façons d'accéder à la vision omnidirectionnelle ou panoramique. [Ragot, 2009] présente en outre des modèles de capteurs centraux (à point de vue ou centre de projection unique) et non-centraux (avec une caustique comme zone centrale de projection).

Projection centrale et non-centrale

Pour un système au point de vue unique [Ramalingam, 2006], la géométrie des rayons de projection est donnée par le point de vue et leur direction. Dans un système non central, la caustique peut être utilisée pour représenter la géométrie des rayons de projection. Une

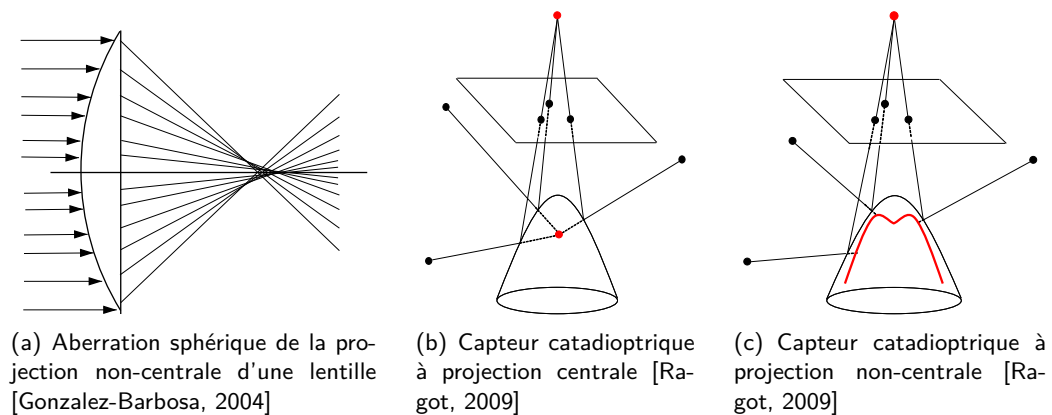


FIGURE 2.1 – Projection centrale et non-centrale cas dioptrique et catadioptrique.

caustique fait référence au lieu dans l'espace 3D où se situent l'ensemble des points de vues d'un système non central. On parle de diacaustique pour les systèmes dioptriques (équipés de simples objectifs), et de catacaustique pour les systèmes catadioptriques (basés sur l'utilisation d'un miroir). La figure 2.1 montre que les rayons conjugués qui traversent une lentille forment une caustique au lieu de passer en un point de projection central unique. De la même manière certains capteurs catadioptriques ne sont pas centraux et le rayon réfléchi sur le miroir est en fait réfléchi selon la tangente à la surface caustique.

Un objectif fisheye n'a pas de centre de projection unique mais une diacaustique. En vision par ordinateur, [Arican, 2010] dit qu'il est souvent admis d'approximer la zone de projection comme un point de vue unique au moins pour des objets lointains. Les configurations centrales sont préférées aux systèmes non centraux parce qu'elles permettent la génération d'images perspective géométriquement correctes à partir des images capturées. Un modèle à centre de projection unique est beaucoup plus simple qu'une caustique, et la plupart des théories et algorithmes développés pour les caméras à projection perspective conventionnelles restent valables pour les caméras omnidirectionnelles à point de vue unique.

2.2.1/ ROTATION D'UN CAPTEUR DE STÉRÉOVISION CLASSIQUE

Cette approche pose des problèmes de positionnement, de synchronisation des prises de vues et de temps de prise des images. Avec une caméra classique, il faut plusieurs acquisitions avec un dispositif tournant pour obtenir l'image omnidirectionnelle. Ce procédé n'est pas applicable pour des scènes dynamiques. [Gehrig, 2005] et [Lin et al., 2008] expérimentent une paire de caméras alignées en capteur de stéréovision, en rotation selon un axe centré.

2.2.2/ CAPTEUR CATADIOPTRIQUE

Les capteurs catadioptriques sont souvent utilisés en configuration monoculaire pour des systèmes de robotique mobile ou de SLAM. La figure 2.2 en montre un utilisé par [Lhuillier, 2011]. Ils ont une caractéristique spécifique, outre la vision omnidirectionnelle, qui est l'observation simultanée du sol. Cette vue peut être avantageuse car elle aide à des applications



(a) Capteur catadioptrique. (b) Vue de rue avec ce capteur.

FIGURE 2.2 – Capteur catadioptrique avec un miroir hyperbolique utilisé par [Lhuillier, 2011].

telles que le maintien de la gyroskopie comme le fait [Bazin et al., 2011], ou la détection des obstacles au sol pouvant aussi contribuer à l'odométrie visuelle. En revanche un capteur catadioptrique ne verra pas toujours entièrement le hauts des bâtiments.

Ce type de capteur ne convient pas à certaines applications car la partie centrale de l'image est inexploitable, on y voit le reflet de la caméra elle-même. Et, l'astigmatisme est fort. C'est une aberration optique due à une géométrie imparfaite du système optique (pour la symétrie radiale de l'ensemble objectif et miroir). Il en résulte du flou dans l'image obtenue. Les capteurs catadioptriques centraux sont difficiles à construire et à entretenir.

Les modèles catadioptriques sont listés entre autres par [Ramalingam, 2006]. De nombreux auteurs utilisent des capteurs catadioptriques pour la 3D. On peut citer [Gonzalez-Barbosa, 2004, Ragot, 2009] qui travaillent en configuration à deux caméras, et [Kawanishi et al., 2009, Arican, 2010, Lhuillier, 2011] qui travaillent en configuration mono-capteur avec déplacement pour générer des modèles 3D depuis plusieurs points de vue.

Il existe par ailleurs des capteurs catadioptriques plus particuliers. L'un développé dans [Weissig et al., 2012] qui est un mélange entre réseau de paires de caméras et miroirs plans.

D'autres proposés par [Mouaddib et al., 2005], qui imaginent des conceptions de capteur catadioptrique stéréo unique. Ils utilisent une seule caméra avec plusieurs miroirs, disposés de telle sorte à obtenir différents points de vue sur la seule image obtenue.

2.2.3/ CAMÉRA À OBJECTIF FISHEYE

Un objectif fisheye est un objectif ultra grand angle, de focale très courte, appelé aussi objectif hypergone. Les objectifs fisheye présentent un champ de vision proche de 180° , ce qui peut être comparé à une vision en demie-sphère. Ils introduisent de fortes distorsions dans l'image, en particulier sur la périphérie où la résolution est faible. L'objectif ne nécessite pas l'alignement d'élément supplémentaire. Un exemple est donné figure 2.3.

La vision fisheye orientée vers le ciel répond bien à l'application étudiée dans cette thèse car

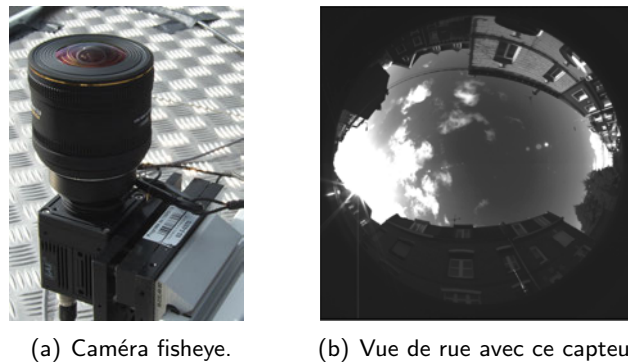


FIGURE 2.3 – Caméra munie d'un objectif fisheye, de la configuration (Cfg 1) avec l'objectif photo Sigma (section 1.3.2).

elle observe les structures dans toute leur hauteur. Par contre elle ne permet pas de voir le sol.

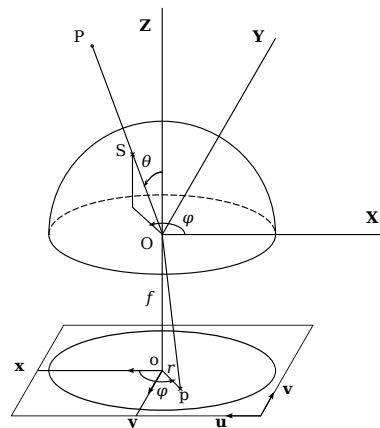


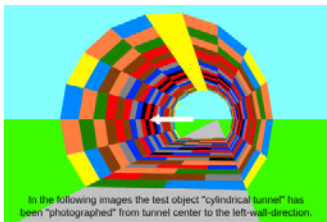
FIGURE 2.4 – Modèle de projection centrale décrivant la relation entre l'objectif fisheye et le plan image. Le rayon r sur le plan image est fonction de l'angle d'incidence θ du rayon perçu.

Des auteurs tels que [Li, 2006a], [Ramalingam, 2006] et [Hughes et al., 2010] présentent les principaux modèles fisheye. Dans ce manuscrit on admet l'hypothèse de la projection centrale. La projection fisheye peut-être représentée par un modèle sphérique, les points de l'espace se projettent sur une sphère unitaire avant d'atteindre le capteur, comme sur la figure 2.4. Un objectif fisheye présente une symétrie radiale autour de son axe optique. Il est intéressant de noter que la fonction de projection fisheye s'apparente aux fonctions de modélisation des distorsions radiales des caméras perspectives avec le modèle sténopé classique. Les modélisations des distorsions radiales expriment un décalage de la position du point considéré en fonction de sa distance au centre optique. Il est tout à fait possible de formaliser les fonctions fisheye de la même manière. On retrouve ainsi le modèle polynomial, le modèle champ de vision et le modèle division [Claus et al., 2005]. [Hughes et al., 2010] exprime d'ailleurs les fonctions fisheye avec la forme des fonctions de distorsions.

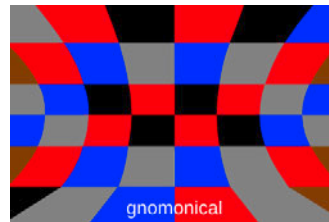
Le tableau 2.1 référence les modèles de projection fisheye. Chaque modèle donne soit une fonction exprimant la relation entre l'angle incident et le rayon du point projeté sur le plan

image selon l'interprétation sphérique, soit une fonction de distorsion passant du rayon du point projeté en perspective sans distorsions au rayon avec prise en compte des distorsions ou inversement. Pour l'expression suivant le modèle sphérique, θ est l'angle de colatitude, r est le rayon dans le plan image et f est la longueur focale de l'objectif (voir figure 2.4). Pour l'expression comme une fonction de modélisation de distorsions radiales, r_i est le rayon initial et r_f le rayon final. Certaines fonctions sont facilement inversibles. C'est une propriété intéressante pour faire des projections et contre-projection sans générer de lourds calculs. Les modèles plus précis et plus complexes sont généralement plus difficiles à inverser.

Les illustrations des déformations des modèles fisheye sont données pour la scène de tunnel montrée figure 2.5(a), la caméra est placée à l'origine de la flèche et regarde dans sa direction. Pour comparer, une caméra perspective classique (fonction $r = f \tan(\theta)$) donne l'image 2.5(b).



(a) Scène exemple.



(b) Vue perspective à la position de la caméra dans la scène.

FIGURE 2.5 – Les illustrations des projections fisheye sont issues du site www.fisheye1ens.de.

2.2.3.1/ TRAITEMENTS VIA UNE RECTIFICATION DES DISTORSIONS FISHEYE

Certains auteurs comme [Shah et al., 1997, Zhao et al., 2000, Nishimoto et al., 2007, Yamaguchi, 2011] choisissent de d'abord rectifier les images fisheye pour se placer dans un cas basé sur la projection perspective classique et le modèle sténopé. L'avantage est qu'il est ensuite possible d'appliquer sans modification tout algorithme créé pour les caméras classiques.

Notons que [Yamaguchi, 2011] donne des illustrations de la vue fisheye selon différentes orientations du capteur pouvant être intéressantes suivant les applications : vue frontale, en contre-plongée en direction du ciel et en plongée en direction du sol.

[Gehrig et al., 2008] n'utilisent qu'une bande horizontale au centre des images fisheye. Ils appliquent une rectification cylindrique pour obtenir des images à projection classique. D'après eux la rectification cylindrique altère moins l'image qu'une rectification planaire. Cette méthode est incapable de calculer les correspondances des objets proches du stéréoscope, car le même point de la scène est déformé de manière différente dans les deux images.

2.2.3.2/ TRAITEMENTS SANS RECTIFICATION DES DISTORSIONS FISHEYE

D'autres auteurs préfèrent traiter les données fisheye sans appliquer de rectification. L'idée est de limiter les interpolations et d'éviter de déformer les images d'origine ce qui dégrade

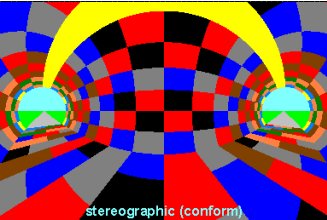
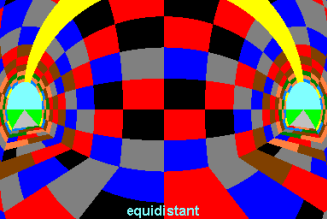
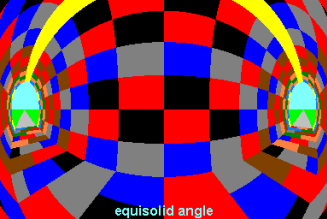
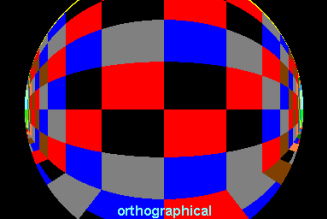
Modèle		Fonction de projection $r = \text{proj}(\theta)$ et illustration ou $r_f = f(r_i)$ ou $r_i = f(r_f)$ pour les distorsions radiales	
Fisheye (considéré comme ayant un point de vue unique)	projection stéréographique	$r = 2f \tan(\frac{\theta}{2})$ (facilement inversible)	
	projection équidistante (ou équiangulaire)	$r = f\theta$ (facilement inversible)	
	projection à angle équisolide	$r = 2f \sin(\frac{\theta}{2})$ (facilement inversible)	
	projection orthographique	$r = f \sin(\theta)$ (facilement inversible)	
	modèle combiné stéréographique et équisolide	$r = a \tan(\frac{\theta}{b}) + b \sin(\frac{\theta}{d})$ (facilement inversible)	
	modèle de distorsion polynomial	$r_f = k_0 + k_1 r_i + k_2 r_i^2 + k_3 r_i^3 + \dots$ (difficilement inversible) ou $r_i = r_f(k_1 + k_2 r_f^2 + k_4 r_f^4 + \dots)$ (difficilement inversible)	
	modèle de distorsion champ de vision	$r_f = \frac{1}{\omega} \arctan(2r_i \tan(\frac{\omega}{2}))$ (facilement inversible)	
	modèle de distorsion division	$r_i = \frac{r_f}{1 + k_1 r_f^2 + k_2 r_f^4 + \dots}$ (difficilement inversible)	

TABLEAU 2.1 – Tableau récapitulatif des différents modèles de projection de capteurs fisheye.

fortement les bords. Les algorithmes doivent être adaptés à la projection sphérique. Dans ce cas le calibrage est généralement fait selon une fonction de projection spécifique à la projection fisheye au lieu de suivre le modèle sténopé.

Modélisation 3D éparsé de points d'intérêt

C'est ce que font [Mičušík et al., 2004, Terabayashi et al., 2009]. Des points d'intérêt sont

appariés entre les images et projetés en 3D par triangulation suivant la projection fisheye. Les auteurs peuvent appliquer ensuite un maillage du modèle éparsé par triangulation de Delaunay.

Transformation épipolaire tenant compte de la projection fisheye

C'est l'étude d'une transformation épipolaire adaptée à la projection fisheye. Pour la géométrie épipolaire, se référer à la section 2.3.2. L'idée est de traiter les courbes épipolaires plus efficacement comme des lignes épipolaires pour l'étape de mise en correspondance stéréo. Des auteurs tels que [Abraham et al., 2005] et [Li, 2006b, Li, 2008] ont proposé leurs méthodes de transformation épipolaire des images fisheye.

Utilisation d'une portion du champ de vision pour s'approcher du modèle sténopé

[Hrabar et al., 2005] utilisent une partie rectangulaire des images fisheye autour du centre. Ils appliquent une méthode de stéréovision classique car la portion d'image choisie de cette manière présente peu de distorsions. Ils utilisent d'autres parties des images fisheye pour calculer le flot optique de leurs drones et pouvoir le cumuler à la stéréovision.

[Kita, 2011] propose une méthode pour mesurer les reliefs du sol pour un robot humanoïde avec de la stéréovision fisheye. La partie basse des images sont aplanies de sorte à ce que uniquement le sol soit projeté sur un plan incluant les pieds du robot à leur position actuelle.

Caractérisation des éléments sans modification des images fisheye

Dans [Herrera et al., 2009, Herrera et al., 2011], les auteurs travaillent en environnement forestier avec des caméras fisheye orientées vers le ciel et proposent deux méthodes d'appariement basées sur une segmentation ou sur les caractéristiques des troncs des arbres.

2.2.4/ STÉRÉOSCOPES HYBRIDES

[Smadja et al., 2005] associent un capteur cylindrique omnidirectionnel par rotation d'une caméra linéaire calibré à une caméra sténopé non calibrée.

[Eynard, 2011] présente un capteur composé d'une caméra omnidirectionnelle fisheye et d'une caméra perspective standard. La caméra omnidirectionnelle permet de connaître l'orientation du drone et d'avoir une observation globale de l'environnement. La caméra conventionnelle permet d'estimer le mouvement et d'observer une zone avec précision.

[Caron et al., 2011] proposent une méthode de calibrage des capteurs pour toute combinaison stéréo hybride composée de capteurs catadioptriques, fisheye ou sténopés classiques. Ils combinent différents modèles de projection. Leur travail a mené à la réalisation d'un logiciel

d'étalonnage stéréoscopique HySCaS¹ (pour *Hybrid Stereo Calibration Software*).

2.2.5/ BILAN

[Gehrig, 2005] essaye différents dispositifs permettant de faire de la stéréo en couvrant un très grand angle de champ : deux caméras conventionnelles sur un système PTU (*Pan Tilt Unit*), deux capteurs catadioptriques hyperboliques et deux caméras munies d'objectifs fisheye. Il en conclut que c'est le système PTU qui donne la plus grande précision avec l'inconvénient de ne pas couvrir un grand champ à un instant donné. Les dispositifs omnidirectionnels permettent d'avoir en permanence ce grand angle de champ mais avec une moindre résolution. Pour la même résolution d'image, la configuration stéréo fisheye optimise mieux l'utilisation du plan image, maintient une meilleure résolution et bénéficie d'une meilleure stabilité du calibrage.

2.3/ BIBLIOGRAPHIE SUR LE CALIBRAGE SPHÉRIQUE FISHEYE

La méthode de modélisation 3D par vision n'est pas immédiate, elle demande en premier lieu une attention particulière au calibrage. Il faut faire face aux distorsions géométriques, aux autres aberrations optiques et à l'échantillonnage du capteur.

De la même manière qu'avec les caméras classiques, on distingue plusieurs manières de faire le calibrage d'une caméra omnidirectionnelle. On distingue trois types de méthodes de calibrage (ou étalonnage), comme le montrent [Ramalingam, 2006, Ragot, 2009] :

— **le calibrage manuel fort,**

utilisation d'une mire constituée de points 3D de positions connues avec précision. On estime les paramètres d'un capteur, c'est-à-dire les paramètres de la fonction de projection pour faire correspondre les projections estimées aux projections réelles. Voir section 2.3.3.1.

— **le calibrage faible,**

estimation de la structure d'un système stéréoscopique, la géométrie épipolaire décrite par les paramètres extrinsèques. Lorsque les paramètres intrinsèques des capteurs sont connus la géométrie épipolaire est décrite par la matrice essentielle, autrement l'estimation des paramètres internes peut être portée par la matrice fondamentale. Voir section 2.3.3.2.

— **l'auto-calibrage,**

estimation des paramètres du ou des capteurs en utilisant les informations contenues dans les images (des points d'intérêts doivent être détectés avec certitude). Les paramètres externes ne peuvent être déterminés qu'à un facteur d'échelle près. Généralement un calibrage faible pour deux images avec estimation directe de la matrice fondamentale. Voir section 2.3.4.

On trouve deux approches distinctes pour transposer la projection omnidirectionnelle :

— **le calibrage paramétrique,**

les paramètres d'un modèle mathématique traduisant le phénomène de projection sont es-

1. www.hyscas.com

timés. Le problème se résume à la minimisation d'une fonction de coût. Elle correspond à une erreur de reprojection tenant compte des paramètres extrinsèques et intrinsèques.

— **le calibrage non-paramétrique,**

aucun modèle n'est utilisé, chaque pixel capture l'intensité lumineuse provenant d'une unique ligne vue dans l'espace 3D. Il s'agit d'un maillage des projections et contre-projections des points à différentes positions sur l'image obtenus avec une mire. Les projections des autres points sont déterminées par interpolation. Ce type de calibrage est rare et fastidieux à utiliser.

2.3.1/ UNIFICATION D'UN MODÈLE POUR CATADIOPTRIQUE ET FISHEYE

La vision fisheye et catadioptrique ont des caractéristiques proches. Il est possible d'utiliser la même modélisation sphérique pour les deux types de capteur. [Geyer et al., 2000] sont les premiers à proposer une théorie commune. Plus tard, [Ying et al., 2004] continue ces investigations. Le modèle unifié est une extension du modèle des capteurs catadioptriques centraux. [Barreto, 2006] étudie la représentation unifiée pour la vision fisheye, catadioptrique, mais aussi perspective, avec des modèles sphérique et parabolique.

[Courbon et al., 2012] proposent une évaluation du modèle unifié avec plusieurs objectifs fisheye. Ils disent qu'ils peuvent avoir moins d'erreur avec ce modèle qu'avec certains modèles spécifiques au fisheye comportant parfois plus de paramètres à estimer.

2.3.2/ GÉOMÉTRIE ÉPIPOLAIRE DES CAMÉRAS SPHÉRIQUES

Des configurations stéréo fisheye de la littérature ont été introduites dans la section 2.2.3.

La géométrie épipolaire est le nom donné aux règles géométriques qui régissent les relations entre deux points de vue. On peut citer [Torii et al., 2008] qui traitent la géométrie sphérique et sténopé pour de la vision à plusieurs points de vue.

Il y a aussi [Arıcan et al., 2010] qui présentent l'estimation de grands cercles épipolaires dans les images sphériques. Dans la figure 2.6, un point X de la scène se projette respectivement en x_1 et x_2 sur les sphères unitaires représentant les caméras 1 et 2. X , C_1 et C_2 forment le plan épipolaire Π qui coupe les sphères unitaires en formant une paire de courbes épipolaires conjuguées. Le passage du repère de la caméra 1 à la caméra 2 est constitué d'une rotation R et d'une translation T . L'entraxe est selon l'axe joignant les centres optiques C_1 et C_2 . Il coupe les sphères de la projection sphérique en formant les épipôles. Il y a deux épipôles par caméra sphérique et ils sont situés à l'opposé l'un de l'autre. Pour toute position sur une caméra sphérique il existe un plan épipolaire, et donc un couple de cercles épipolaires conjugués. Tout cercle épipolaire sur une caméra passe par ses deux épipôles.

Dans le cas d'images omnidirectionnelles, les lignes épipolaires deviennent des courbes épipolaires. Une autre différence entre les géométries épipolaires perspective et omnidirectionnelle est l'existence d'un deuxième épipôle dans une image omnidirectionnelle.

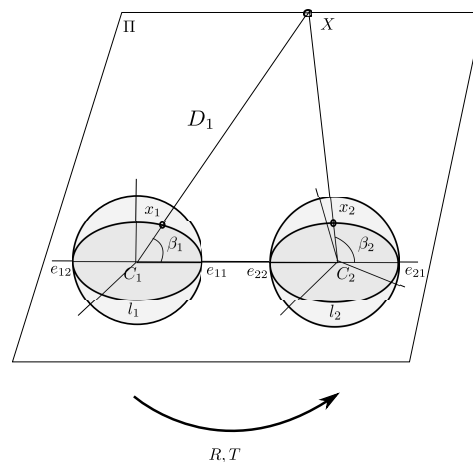


FIGURE 2.6 – Géométrie épipolaire pour des images sphériques [Arıcan et al., 2010].

2.3.3/ CALIBRAGE MANUEL

Il existe plusieurs méthodes de calibrage stéréo des caméras omnidirectionnelles avec l'hypothèse du modèle central. Les plus efficaces utilisent le modèle sphérique, il décrit bien la symétrie des distorsions géométriques des caméras omnidirectionnelles. Le calibrage manuel nécessite l'intervention d'un opérateur pour positionner mire et caméras. Dans ce cas le calibrage fort intervient avant le calibrage faible.

2.3.3.1/ CALIBRAGE FORT

Le calibrage fort est effectué à partir de données de référence dans le monde réel. C'est un calibrage individuel des caméras pour déterminer les paramètres intrinsèques. En calibrage manuel les données de référence sont décrites par une mire de calibrage dont les dimensions sont connues. On peut noter certains travaux visant à développer des mires spécifiques comme le fait [Ragot, 2009] pour un capteur catadioptrique. Il propose une mire à placer autour de la caméra, en forme de cylindre parsemé de leds. Dans le cas du calibrage manuel les paramètres internes sont déterminés avec la meilleure précision possible, ainsi que les paramètres extrinsèques par rapport au repère de la mire dans le monde réel.

[Shah et al., 1996] présentent une procédure de calibrage des paramètres internes pour les caméras fisheye avec un rayon laser et une mire plane.

Plus tard [Scaramuzza et al., 2006, Scaramuzza, 2008], [Mei et al., 2007] et [Bastanlar et al., 2008, Puig et al., 2011] ont développé des méthodes et outils de calibrage des capteurs omnidirectionnels. Ils utilisent une mire plane, ou 3D pour [Bastanlar et al., 2008, Puig et al., 2011]. Plusieurs photos sont nécessaires, une procédure efficace est de prendre une série de clichés de la mire déplacée en tournant autour du capteur par rapport à l'axe optique. Ils s'intéressent surtout aux capteurs catadioptriques mais proposent des modèles qui fonctionnent aussi avec les caméras fisheye. [Mei et al., 2007] et [Bastanlar et al., 2008, Puig

et al., 2011] utilisent un modèle sphérique unifié comme vu en section 2.3.1.

Autrement, [Ho et al., 2005] proposent de calibrer les caméras fisheye avec une mire plane et une plate-forme rotative sur laquelle ils font tourner la caméra.

[Srestasathien et al., 2014] se basent sur la détection de lignes droites et non pas de coniques pour améliorer la stabilité.

[Tardif et al., 2005, Tardif et al., 2009] calibrent les caméras fisheye et catadioptrique avec des mires projetées en lumière structurée sur des plans. Leur méthode permet de travailler en modes de projection centrale et non-centrale.

2.3.3.2/ CALIBRAGE FAIBLE

Le calibrage faible consiste à déterminer la géométrie épipolaire entre deux caméras (stéréovision). Si manuel, le calibrage stéréo peut se calculer à partir des paramètres internes et externes, à condition d'observer la mire avec les deux caméras sans les déplacer l'une relativement à l'autre. C'est l'estimation de la transformation d'un repère caméra à l'autre via le repère de la mire (paramètres extrinsèques).

Pour les modèles sténopé et sphérique, l'information de calibrage stéréo est contenue dans la matrice essentielle ou fondamentale. La matrice essentielle exprime le passage d'un repère caméra à l'autre. La matrice fondamentale le passage d'un repère image à l'autre, elle inclut les transformations des projections internes.

En vision omnidirectionnelle l'utilisation d'une mire est difficile compte tenu de l'extrême champ de vision des caméras. Les applications de robotique présentent souvent une configuration avec un court entraxe, permettant l'utilisation d'une mire de calibrage. Dans les cas où il est nécessaire de travailler en stéréo avec un grand entraxe comme par exemple pour les applications transport en extérieur, à de nombreuses positions, une mire ne peut pas être vue convenablement par les deux caméras.

2.3.3.3/ STÉRÉOVISION OMNIDIRECTIONNELLE AVEC COURT ENTRAXE

L'utilisation de mires pour le calibrage est possible. Le principe consiste à acquérir des images pour différentes positions de la mire (ou du stéréoscope), et à calculer les paramètres en minimisant une fonction d'erreur. L'échelle est connue grâce aux dimensions fixées de la mire.

[Abraham et al., 2005] et [Li, 2008] calibrent un capteur de stéréovision fisheye avec une mire de calibrage en forme de demi-boîte.

[Gehrig et al., 2008] calibrent un capteur de stéréovision fisheye avec une mire plane et la méthode pour des caméras sténopé avec distorsions extrêmes de [Krüger et al., 2004].

2.3.3.4/ STÉRÉOVISION OMNIDIRECTIONNELLE AVEC GRAND ENTRAXE

Dans le cas de grands entraxes, les mires peuvent être remplacées par des marques sur les bâtiments. Ceci revient à une mire géante. Il est difficile de positionner les marques avec une grande précision. C'est ce que font [Cannelle et al., 2012], qui placent des marques sur des structures connues. Ils exploitent le système de cartographie mobile de l'IGN : Stereopolis V2.

Dans le cadre du projet BOSS [Bos, 2009], plusieurs caméras fisheye sont calibrées individuellement et entre elles avec un grand entraxe (environ 1 mètre) à l'aide d'un objet de calibrage. C'est un cube d'arêtes de 1 mètre de long, dont chaque face est un damier. La figure 2.7 illustre le type d'image obtenu avec l'objet de calibrage.



FIGURE 2.7 – Cube utilisé pour le calibrage des caméras dans le projet BOSS [Bos, 2009].

2.3.4/ CALIBRAGE AUTOMATIQUE

Le calibrage automatique ne nécessite pas l'intervention d'un opérateur. Le calibrage automatique peut estimer directement et simultanément les paramètres intrinsèques des caméras et extrinsèques du stéréoscope à l'exception de l'échelle du monde réel. Le plus souvent il est utilisé avec des caméras déjà calibrées individuellement pour ne faire automatiquement que le calibrage faible. Dans ce cas les paramètres internes sont connus et on parle d'estimation de pose de la caméra entre les vues. Parfois il permet de déterminer les paramètres internes comme le font [Devernay et al., 2001, Mičušik, 2004].

Lorsqu'il est effectué en l'absence de mire, à partir de deux images de la même scène, des primitives sont détectées et appariées entre les images. La précision est souvent inférieure au calibrage manuel car les éléments détectés et utilisés sont connus avec une précision moindre que des projections de points de mires. L'autre différence avec la mire c'est que l'on ne connaît pas les dimensions ni les positions relatives et absolues des points. Pour cette raison le calibrage est assuré à une transformation projective près c'est-à-dire à l'échelle près.

Les méthodes de calibrage automatiques sont souvent basées sur l'algorithme RANSAC pour ignorer les paires d'images aberrantes (*outliers*) et ne considérer que les couples appariés avec certitude (*inliers*) pour l'estimation finale des paramètres. À partir de quelques correspondances peut être estimée la matrice fondamentale, qui représente la géométrie épipolaire. On trouve en calibrage faible automatique les algorithmes à 7 et 8 points [Hartley et al., 2004]. Tels [Mičušik et al., 2004] ou [Lhuillier et al., 2004] (en catadioptrique), les auteurs qui font

de la stéréovision avec des images non-calibrées les calibrent automatiquement au moins pour estimer des relations géométriques entre elles. Ces relations sont nécessaires en stéréovision.

Avantages :

- Pas besoin de mire.
- Entièrement automatique et sans opérateur.
- Nécessaire pour une configuration stéréo monoculaire quand les images proviennent d'un unique capteur en mouvement (*Structure from Motion*).

Inconvénients :

- L'échelle réelle demande une étape supplémentaire pour être connue.
- Précision plus faible due à la détection et l'appariement automatique des amers visuels sur du contenu pas toujours très contrasté.

Quelques auteurs s'intéressent au calibrage automatique.

En vision sténopé, [Lemondé, 2005] étudie le calibrage automatique pour un capteur de stéréovision embarqué sur un véhicule. Il dit que le calibrage doit être automatique pour recalibrer régulièrement les capteurs. Les dispositifs embarqués subissent chocs et vibrations qui les déplacent légèrement et continuellement. Il calibre une première fois son dispositif manuellement comme initialisation. Puis, un recalibrage automatique à partir des paramètres précédents est exécuté régulièrement pour travailler toujours avec des données optimales.

En vision omnidirectionnelle, on peut citer [Mičušik, 2004] et [Geyer et al., 2007]. Ils proposent chacun leurs algorithmes à partir de 9 paires de points correspondants entre les images pour estimer les paramètres internes et la matrice fondamentale. [Mičušik, 2004] propose aussi un algorithme des 15 points pour une fonction de projection plus précise à 2 paramètres qu'il a développée pour le capteur fisheye qu'il utilise.

L'algorithme des 9 points de [Geyer et al., 2007] est appliqué à un capteur catadioptrique central. Ils optimisent en plus l'estimation du centre de projection de l'image. Ils proposent aussi un algorithme des 15 points qui gère des paramètres internes différents aux deux capteurs.

2.4/ *Structure from Motion*

Le *Structure from Motion* (SfM) est un procédé dont le principe consiste à estimer en 3D les structures visualisées à partir du déplacement d'une caméra. Le SfM est basé sur le calibrage et la stéréovision. Une seule caméra peut suffire étant donné que le traitement est effectué pour différents points de vue avec déplacement du capteur. On peut utiliser le SfM avec une paire d'images, et aussi l'étendre à une succession de paires d'images pour former un grand modèle qui est l'assemblage de plusieurs modèles locaux.

Pour fonctionner, le SfM a besoin d'une connaissance des paramètres de calibrage des capteurs

utilisés. Ce qui varie entre chaque point de vue sont des paramètres externes : la pose relative entre la nouvelle photo et la précédente.

Le SfM repose sur une estimation de la pose (*ego-motion estimation*) à chaque nouvelle acquisition. Il s'agit d'une forme de calibrage stéréoscopique automatique pour les paramètres extrinsèques. La pose représente le changement de repère entre les vues. Cette transformation rigide contient une rotation et une translation. Par la translation, elle inclut une mesure de la distance parcourue, l'odométrie visuelle. Cette odométrie n'est pas forcément à l'échelle réelle mais il est possible de mesurer le facteur d'échelle d'un modèle local à un autre. Un des problèmes dans de nombreuses applications est de trouver la longueur réelle d'une translation pour travailler à échelle réelle. On distingue deux cas :

1. En configuration binoculaire ce problème n'existe pas car un calibrage fort du stéréoscope ou la connaissance de l'entraxe permettent de projeter les points 3D à l'échelle.
2. En configuration monoculaire le calibrage ne permet d'estimer que les paramètres internes. Dans ce cas, en plus d'estimer la pose, il faut trouver l'échelle pour au moins un déplacement. Ceci peut être fait par l'adjonction d'un odomètre externe.

2.4.1/ MÉTHODES D'ESTIMATION DE LA POSE

On peut distinguer deux types d'approches : les études globales basées sur l'apparence de toute l'image, et les méthodes basées sur des amers visuels (*features*) extraits et suivis dans les deux images successives.

L'estimation de pose peut être suivie d'une étape d'affinement. L'affinement se base sur les éléments utilisés pour l'estimation de pose ou en exploite plus au fur et à mesure que le modèle se densifie, comme en applications SLAM (*Simultaneous Localisation And Mapping*, localisation et cartographie simultanées). On distingue deux types de procédure d'affinement :

- la détection et fermeture de boucles, indispensable lorsque l'on veut faire de la cartographie pour du déplacement ;
- la consolidation des informations 3D (*registration*, ou *bundle adjustment*) pour réduire les écarts d'alignement entre les données 3D.

2.4.1.1/ MÉTHODES GLOBALES BASÉES SUR L'APPARENCE

Il existe peu de méthodes globales car elles sont souvent moins précises et plus coûteuses que les méthodes utilisant des éléments spécifiques.

On peut citer [Caron et al., 2012] qui proposent une estimation de pose basée modèle pour des caméras sténopés. Ils travaillent à partir d'un modèle 3D connu qui sert à générer des cartes des profondeurs comparées de manière globale à la carte obtenue avec les images réelles jusqu'à trouver la bonne pose.

2.4.1.2/ MÉTHODES PAR SUIVI DE *features*

[Scaramuzza et al., 2011] résumant les différentes approches de la bibliographie et les classifient en fonction des bases dans lesquelles elles travaillent :

- **2D-2D**, besoin au moins de 5 paires de points correspondants entre les deux images. Les propriétés de la géométrie épipolaire sont utilisées et la matrice essentielle est estimée. Elle peut être décomposée pour extraire la rotation et la translation (la décomposition est décrite entre autres dans [Baselgia, 2010, Scaramuzza et al., 2011]). Ceci aboutit aux algorithmes des 5 points [Nistér, 2004], des 7 points [Hartley et al., 2004] et des 8 points [Hartley, 1997]. Moins on a besoin de points mieux c'est pour une méthode encadrée par un RANSAC (*RANdom SAmple Consensus*), stratégie quasi-obligatoire par tirages aléatoires. Pour améliorer la probabilité de ne tirer que de bons éléments on peut réduire la quantité de points nécessaires en allégeant le nombre de degrés de liberté du mouvement. Ceci donne une estimation approchée de la pose à affiner ensuite. C'est ce que font [Ortín et al., 2001, Scaramuzza, 2011, Troiani et al., 2014].
- **3D-3D**, besoin au moins de 3 paires de points correspondants non-colinéaires dans le monde 3D, ceci est possible en conditions de reconstruction d'un modèle 3D de l'environnement à la même échelle. Il n'est pas possible d'utiliser la méthode 3D-3D en configuration monoculaire car l'échelle entre modèles 3D est toujours différente.
- **3D-2D**, pour estimer le déplacement depuis au moins 3 points correspondants entre un modèle 3D et leurs projections 2D dans une image. Ce problème est connu sous le nom de PnP (*Perspective from n Points* ou estimation de la Pose à partir de n Points).

Vision perspective non omnidirectionnelle

[Milella et al., 2006] et [Howard, 2008] travaillent en configuration binoculaire. [Milella et al., 2006] effectuent une minimisation 3D-3D par moindres carrés sur les projections de points d'intérêts. Ils appliquent ensuite une consolidation 3D avec l'algorithme ICP (*Iterative Closest Point*). [Howard, 2008] fait une minimisation d'erreur de projection 3D-2D pour des coins détectés dans les images.

[Pradeep et al., 2013] et [Bourmaud, 2015] utilisent l'algorithme des 5 points en configuration mono-caméra. [Pradeep et al., 2013] fusionnent plusieurs méthodes, après cette estimation 2D-2D ils appliquent une estimation 3D-2D à partir de 3 points, et finalement ils effectuent une évaluation globale. [Bourmaud, 2015] applique l'estimation de la pose dans l'algèbre de Lie. Il utilise conjointement un EKF (*Extended Kalman Filter*) et effectue une détection des boucles pour cartographier des environnements intérieurs.

[Eudes, 2011] étudie l'estimation de pose par ajustement de faisceaux sur un modèle 3D global ou un ensemble réduit de modèles 3D locaux. Il travaille en configuration monoculaire et réduit l'erreur en 2D-2D ou 2D-3D grâce à la minimisation de Levenberg-Marquardt.

[Zhu et al., 2012] proposent de l'odométrie visuelle en configuration monoculaire. Ils appliquent l'algorithme à 2 points avec l'hypothèse que les mouvements sont restreints dans le plan. Pour affiner la pose avec tous les degrés de liberté possibles en 3D, ils se basent ensuite sur un algorithme à 3 paires de points en 2D-3D.

Vision omnidirectionnelle fisheye ou catadioptrique

La vision omnidirectionnelle permet d'avoir une estimation plus robuste de la pose. Son champ de vision extra-large lui permet de détecter des éléments mieux répartis autour de la caméra.

[Bunschoten et al., 2003] et [Terabayashi et al., 2009] utilisent l'algorithme des 8 points, en configuration monoculaire, respectivement catadioptrique et fisheye. [Bunschoten et al., 2003] font du suivi de points d'intérêts avec la méthode de Kanade-Lucas-Tomasi (KLT). Ils utilisent aussi la distance du capteur au sol pour obtenir le facteur d'échelle et les dimensions réelles.

[Lhuillier, 2005, Lhuillier, 2011] présente une méthode de *Structure from Motion* avec une caméra catadioptrique. Il initialise l'estimation en considérant un modèle central, puis effectue un *bundle adjustment* basé sur les erreurs de reprojection des points correspondants et tenant compte de la forme non-centrale du miroir.

[Caruso et al., 2015] proposent une méthode de SLAM temps-réel pour la vision omnidirectionnelle fisheye. Ils se basent sur le modèle central unifié de [Geyer et al., 2000] et [Ying et al., 2004] (voir section 2.3.1). Ils détectent les arrêtes des structures et mettent à jour des cartes des distances probabilistes semi-denses à partir des estimations précédentes pour augmenter la précision. La pose est estimée par application de l'algèbre de Lie. Une détection des boucles est également implémentée pour les fermer et affiner le modèle.

2.5/ ÉTAT DE L'ART DE LA MISE EN CORRESPONDANCE STÉRÉO

Après le calibrage, l'autre grande étape pour arriver au modèle 3D par la stéréovision est la mise en correspondance stéréoscopique. Les méthodes de mise en correspondance se distinguent principalement par deux grandes familles : les méthodes locales et les méthodes globales. La plupart des auteurs ([Scharstein et al., 2002, Chambon, 2005, Fakhfakh, 2011]) décomposent les procédures d'appariement selon ces composantes :

1. Éventuel pré-traitement des images, pour améliorer le contraste, réduire le bruit, détecter et décrire des éléments d'intérêt, exploiter le gradient, segmenter les images, etc.
2. Choix de la mesure du coût de correspondance.
3. Pour les méthodes locales, agrégation du coût dans une région (ou fenêtre).

4. Calcul des disparités et optimisation, selon le type de méthode :
 - Pour les méthodes locales, association de chacun des pixels avec celui ayant le coût optimal.
 - Pour les méthodes globales, trouver pour chaque pixel une disparité optimisant une énergie globale à définir. De nombreux algorithmes peuvent être utilisés pour en trouver un extremum.
5. Éventuel post-traitement d'affinement des disparités, appariement au sous-pixel, interpolation des disparités pour éliminer des trous, lissage des surfaces homogènes par estimation de l'orientation et de l'inclinaison des plans ou des segments de disparités, gestion et élimination des erreurs d'appariement, etc.

Différentes contraintes permettent d'améliorer la robustesse ou de réduire le temps de calcul de l'appariement. Les plus communes sont :

- la contrainte épipolaire : l'idée est de limiter la recherche dans l'image (2D) à une recherche dans les lignes épipolaires conjuguées (1D) pour gagner en temps de calcul et en précision. C'est une application des propriétés de la géométrie épipolaire, elle nécessite un calibrage stéréo.
- la contrainte d'unicité : les appariements multiples ne sont pas autorisés.
- la contrainte d'ordre : l'ordre des points appariés dans l'image de recherche doit être le même que dans la référence.
- la contrainte de symétrie, appelée aussi contrainte de consistance forte (ou faible si une tolérance est accordée) ou encore vérification bidirectionnelle ou LRC *Left-Right Consistency* : pour être validés les appariements cherchés de l'image de référence à l'image de recherche doivent correspondre aux appariements cherchés de l'image de recherche vers l'image de référence (inversion des rôles).

Pour choisir une méthode de mise en correspondance, il faut définir si elle doit être faite pour tous les pixels de l'image : on parle de correspondance dense, ou pour une partie des pixels : on parle alors de mise en correspondance partielle, non dense ou éparse.

2.5.1/ MÉTHODES ÉPARSES ET DENSES

Les méthodes de mise en correspondance peuvent être définies pour traiter tous ou une partie des pixels des images.

2.5.1.1/ MISE EN CORRESPONDANCE ÉPARSE

La mise en correspondance éparse prend compte de points particuliers de la scène, ou plus généralement de primitives. Elle est robuste mais ne permet pas une reconstruction 3D complète

de la scène. Dans certains cas ces primitives ont un sens sémantique pouvant être exploité pour la reconstruction de la scène. Différents types de primitives peuvent être étudiées [Chambon, 2005, Fakhfakh, 2011] :

- Les coins ([Lhuillier et al., 2004, Megyesi et al., 2004]).
- Les contours ([Pajares et al., 2003, Torr et al., 2004, Nieradka et al., 2009]).
- Les courbes ([Fabbri et al., 2010]),
efficace avec un calibrage grossier mais nécessite de nombreuses vues pour fonctionner.

Les attributs utilisés sont souvent les niveaux d'intensité du pixel (niveaux de gris ou couleurs). [Chambon, 2005] a aussi répertorié des méthodes d'agrégation prenant en compte de grandes déformations entre les images :

- Histogramme local ([Carneiro et al., 2004]),
à chaque point est associé un vecteur d'attributs qui dépend du pixel et de ses voisins.
- Caractéristiques des droites qui forment un coin ([Stock et al., 2003]),
sensible à des rotations d'image mais robuste à des variations d'intensité lumineuse.
- Caractérisation de chaque point par des invariants locaux ([Schmid et al., 1995]),
le calcul de mesures différentielles invariantes de la luminosité permet d'effectuer une mise en correspondance dans le cas d'une transformation importante entre deux images.

2.5.1.2/ MISE EN CORRESPONDANCE DENSE

La mise en correspondance dense applique un algorithme d'appariement à l'ensemble des pixels formant l'image. Il en découle plus de détails, ce qui permet plus de précision dans la reconstruction. Calculer une carte de disparité complète nécessite un temps plus long.

Certaines publications font une mise en correspondance quasi-dense, que l'on classera dans cette catégorie de par leur nature dense. On peut citer [Lefebvre, 2008], qui attribue une mesure de confiance par logique floue à chaque appariement et supprime les moins sûrs. Une autre stratégie est de partir d'une mise en correspondance éparse sûre et de propager les estimations par croissance de germes ou propagation de croyance dans l'image. C'est ce que font [Lhuillier et al., 2004, Wu et al., 2010, Caruso et al., 2015]. Notons que [Caruso et al., 2015] propage les estimations temporellement en suivant les éléments en vidéo.

2.5.2/ MÉTHODES LOCALES ET GLOBALES

Deux catégories principales de mise en correspondance stéréoscopique des pixels sont recensées, classées selon le type de méthode d'optimisation utilisée.

2.5.2.1/ MISE EN CORRESPONDANCE LOCALE

On cherche à établir le degré de corrélation entre le pixel de référence à appairier et un ensemble de candidats de l'image de recherche dans une zone de recherche autour de la

position de la référence. Pour chacun des pixels à tester, on prend en considération une zone d'agrégation, que l'on compare via une mesure de corrélation à la zone d'agrégation de référence. Le pixel dont la zone d'agrégation sera mesurée la plus semblable à celle associée au pixel de référence étudié sera considéré comme son homologue.

Il convient de choisir la forme de la fenêtre d'agrégation en fonction des besoins. Pour une petite fenêtre, les erreurs d'appariement sont dues à l'ambiguïté et au bruit, alors que pour une grande fenêtre, les disparités des pixels dans la fenêtre ont plus de chances d'être différentes à cause des occultations et des discontinuités de disparité. [Fakhfakh, 2011] classifie différentes formes de fenêtre d'agrégation :

- Fenêtre fixe ([Okutomi et al., 1993]).
- Fenêtre de taille adaptative ([Belli et al., 2000, Hosni et al., 2010, Gupta et al., 2010]).
- Fenêtres multiples ou déplaçables ([Mordohai et al., 2006, Anil Adhyapak et al., 2007]), ensemble de fenêtres de tailles différentes. Celle avec laquelle la fonction de vraisemblance donne le coût optimal est choisie pour résoudre des problèmes d'occultations géométriques.
- Fenêtre pondérée ([Yoon et al., 2006, Ding et al., 2011]), pondérer les pixels dans une fenêtre. Méthode précise mais longue à exécuter.

On trouve les méthodes d'appariement locales suivantes [Chambon, 2005] :

- Recherche exhaustive appelée aussi WTA (*Winner Take All*) ([Okutomi et al., 1993]), comparaison naïve de chaque autre possibilité avec la meilleure de la comparaison précédente.
- Croissance de germes ([Lhuillier et al., 2004, Wei et al., 2004, Megyesi et al., 2004]), détecter des points d'intérêt ou segmenter l'image, mettre en correspondance des pixels avec une grande certitude, puis augmenter progressivement le nombre de pixels mis en correspondance à partir des précédents.
- Méthode coopérative locale ([Zitnick et al., 1999, Mayer, 2003]), mises à jour itératives pour affiner les correspondances tenant compte localement de la contrainte de continuité (support local) et de la contrainte d'unicité (zone d'inhibition). Voir la figure 2.8.

2.5.2.2/ MISE EN CORRESPONDANCE À OPTIMISATION GLOBALE

On parle de méthode globale lorsque l'on cherche une minimisation globale du coût d'une fonction objective. Toutes ces méthodes ou presque font appel à la même modélisation de départ : les champs de Markov aléatoires. Contrairement aux méthodes locales, les méthodes globales peuvent lever des ambiguïtés d'appariement des pixels appartenant à des régions de couleur homogène ou de texture répétitive.

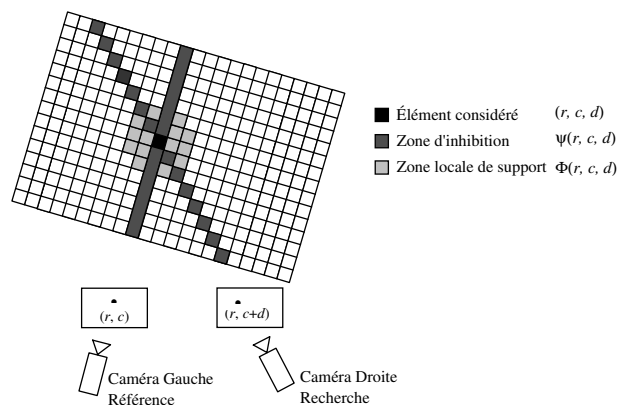


FIGURE 2.8 – Méthode coopérative locale. Illustration pour une tranche 2D de l'espace 3D des disparités [Zitnick et al., 1999].

Dans une démarche de recherche globale, on cherche à effectuer l'optimisation d'énergie d'une certaine fonction objective ou *fitness*. Pour cela une technique d'optimisation est appliquée à la fonction d'énergie correspondante. Cette fonction objective est construite à partir des différentes contraintes que l'on choisit de respecter (contrainte épipolaire, d'unicité, d'ordre, etc). On la définit comme un cumul de termes représentant chaque contrainte, de telle manière à ce que son extremum corresponde à la meilleure des solutions pour les appariements.

Il existe de nombreuses méthodes d'optimisations globales, entre autres [Chambon, 2005, Fakhfakh, 2011] :

- La régularisation ([Aydin et al., 2010, Pock et al., 2010]),
lissage itératif des disparités jusqu'à ce que ne ressortent plus que les discontinuités correspondant aux différents plans de la scène.
- La relaxation ([Brockers et al., 2004]),
à chaque appariement est attribué une mesure de confiance. Ils évoluent de manière itérative, en ne mémorisant que la correspondance ayant la meilleure probabilité.
- Le recuit simulé ([Barnard, 1986, Lee et al., 2002]),
méthode d'optimisation stochastique (basée sur des phénomènes aléatoires). En partant d'une température très élevée pour une initialisation des disparités, on cherche à optimiser une fonction d'énergie mesurée après chaque changement d'état et que l'on ne garde que s'il diminue la température globale.
- *segment-based method* ([Gutiérrez et al., 2004, Bleyer et al., 2005]),
tient compte des régions segmentées pour compléter une méthode d'optimisation globale. Elle sert à mieux estimer les corrélations dans les zones homogènes ou peu texturées.
- Le *Tensor voting* ([Mordohai et al., 2006, Wu et al., 2010]),
permet d'estimer la structure d'une scène à partir de données manquantes ou bruitées avec plusieurs possibilités d'appariements pour chaque pixel et un processus de vote.
- Le *Stereo matting* ([Taguchi et al., 2008]),

l'intérêt de cette méthode est la gestion de la transparence.

2.5.2.3/ MÉTHODES GLOBALES BASÉES SUR LES GRAPHS

On peut classer à part les méthodes basées sur les graphes, généralement très efficaces :

- La programmation dynamique ([Ohta et al., 1985, Torr et al., 2004, Salmen et al., 2009]), en vision par ordinateur, c'est l'algorithme de Viterbi qui est exploité. Le principe consiste à découper un problème global en une succession de petits problèmes plus faciles à résoudre. C'est l'algorithme qui est développé et étudié dans le chapitre 4.
- La coupure de graphes ([Ishikawa et al., 1998, Deng et al., 2005]), une coupure du graphe représente une mise en correspondance globale possible. La figure 2.9 représente un graphe construit pour une paire de lignes épipolaires conjuguées. Un arc coupé entre deux nœuds induit la vérification de son scénario (appariement, occultation, contrainte, etc). Le coût total de la coupe est égal au cumul des capacités des arcs coupés. Le but est de trouver la coupe minimale en résolvant le problème du flux maximal.

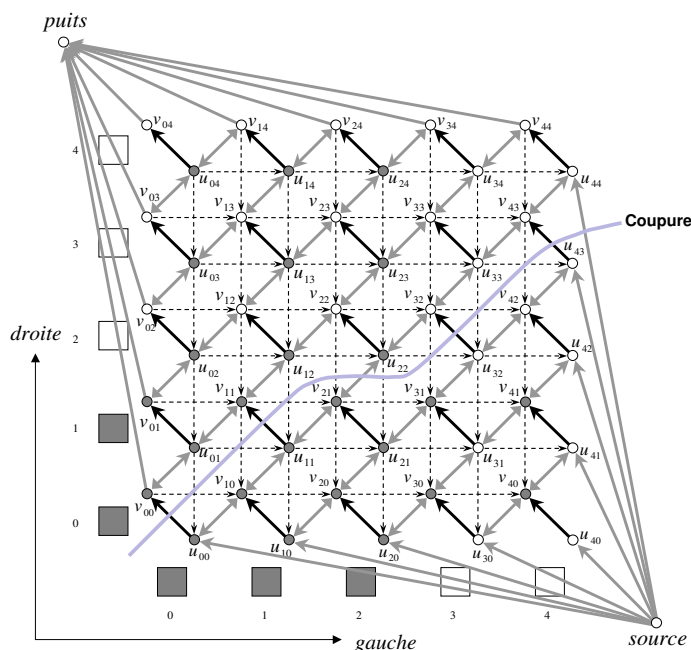


FIGURE 2.9 – La coupure de graphe consiste à trouver la coupure de capacité minimale [Ishikawa et al., 1998].

- Les réseaux de neurones de Hopfield ([Ruichek, 2005]), la mise en correspondance est effectuée sur des points d'intérêt. À partir d'un état initial, le réseau évolue jusqu'à un état stable correspondant à un minimum de la fonction d'énergie. Un réseau de neurones représente tous les couples possibles entre les points des lignes épipolaires gauche et droite, voir la figure 2.10. L'état binaire d'un neurone tend à évoluer vers 1 si l'appariement est correct, vers 0 sinon.
- La propagation de croyance ([Yang et al., 2010], [Cuscó Mach, 2009]), fonctionne par transmission de messages dans un graphe non dirigé proche d'un réseau

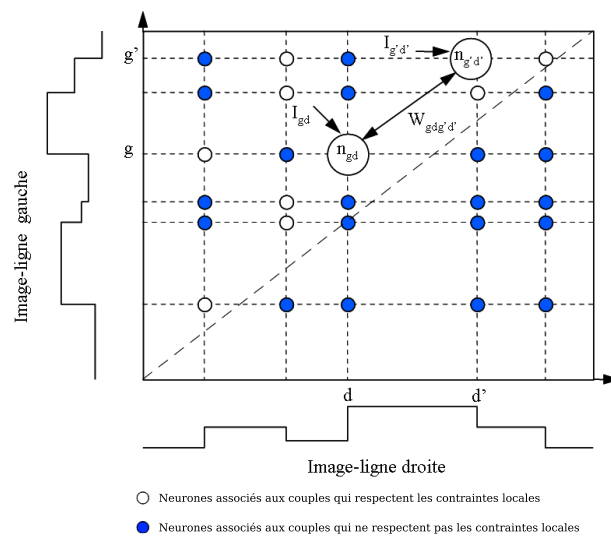


FIGURE 2.10 – Les réseaux de neurones, ce graphe représente les neurones pour l'appariement de primitives correspondant à des points de contours [Ruichek, 2005].

de neurones. À chaque itération, l'énergie est mise à jour en prenant en compte les nouvelles confiances affectées à chaque neurone.

- La propagation de croyance sélective ([Fakhfakh, 2011]), propagation de croyance à l'intérieur de régions segmentées pour pouvoir résoudre le problème des disparités dans les zones occultées et homogènes.
- Les algorithmes génétiques ([Ruichek, 2005, Paulinas et al., 2007]), consistent à faire évoluer une population d'individus avec des opérateurs aléatoires qui sont la sélection, le croisement (ou reproduction) et la mutation. Pour chaque paire de lignes épipolaires conjuguées, une matrice est construite dont les nœuds correspondent à une mise en correspondance possible représentée par un état binaire. Ils sont appelés les gènes. L'ensemble des gènes d'une matrice mis bout-à-bout constitue un chromosome. Un algorithme génétique utilise une fonction *fitness* pour évaluer chaque chromosome. Lorsque l'on a fait subir aux chromosomes un certain nombre de fois la boucle [Sélection](#), [Reproduction](#), [Mutation](#), [Évaluation](#) et [Construction de la nouvelle population](#), on arrête.

2.6/ CONCLUSION

Ce chapitre montre différents dispositifs de vision omnidirectionnelle utilisés en stéréovision et leur calibrage. Le projet CAPLOC exploite des caméras fisheye orientées vers le ciel car elles sont peu coûteuses, simples à intégrer sur un véhicule et offrent une haute résolution comprise entre 1000 et 5000 pixels de côté. Elles permettent de voir les bâtiments des alentours sur toute leur hauteur. Différentes méthodes de calibrage existent. En vision omnidirectionnelle c'est le modèle de projection sphérique qui permet de modéliser le plus efficacement la caméra et ses distorsions. La deuxième étape pour atteindre la modélisation 3D à l'échelle réelle est la mise en correspondance stéréo des points des images. De nombreuses méthodes existent

avec un coût et une robustesse variables. Il faut correctement traiter les images issues de la projection sphérique et de l'échantillonnage des capteurs. Compte-tenu des spécificités des caméras fisheye et du projet CAPLOC, les solutions suivantes sont retenues :

1. Un calibrage automatique avec recalibrage régulier, pour compenser les décalibrages connus par le système embarqué sur véhicule ([Lemondé, 2005]).
2. Une mise en correspondance stéréo par la méthode d'optimisation par programmation dynamique basée sur les graphes avec proposition d'un graphe 3D pour gagner en robustesse aux problèmes d'échantillonnage (plus forts pour des courbes épipolaires sphériques que pour des lignes épipolaires classiques).

La configuration expérimentale utilisée pour les travaux présentés est typiquement un système à grand entraxe (voir section 1.3.2). Le calibrage manuel est rédhibitoire pour une configuration omnidirectionnelle à grand entraxe à cause des contraintes qu'une mire adaptée impose (voir section 2.3.3.4). Un calibrage automatique utilisé comme recalibrage régulier est recommandable dans le cas d'une configuration amenée à varier. Il est toujours plus précis qu'un calibrage manuel décalibré après un laps de temps de fonctionnement. C'est le cas des systèmes embarqués dans les véhicules et qui subissent des vibrations dérégulant la configuration [Lemondé, 2005]. Le système étudié dans ce manuscrit présente ces deux caractéristiques critiques. Dans ce contexte, le calibrage doit être automatique.

Pour cela on se base sur les travaux de [Mičušík, 2004], et on choisit d'utiliser un modèle central et à un seul paramètre pour ne pas complexifier inutilement l'estimation automatique. [Hughes et al., 2010] montre que les modèles de projection linéaires à un paramètre sont assez précis, et même meilleurs que des modèles à plus de deux paramètres avec l'ajout d'un seul paramètre supplémentaire. [Mičušík, 2004] n'étudie que la fonction de projection équidistante. Dans le chapitre 3, on propose d'étendre cette méthode aux modèles de projection à angle équisolide et à projection stéréographique. L'algorithme de calibrage choisi permet en outre d'effectuer du *Structure from Motion* par odométrie visuelle 2D-2D.

Pour conserver l'intégrité des données images issues de la projection fisheye, on choisit de ne pas appliquer de rectification des distorsions générant interpolations et pertes d'informations. On propose une procédure de mise en correspondance sur les points des courbes épipolaires alignés. Dans le chapitre 4, la mise en correspondance stéréo est adaptée à ce choix et aux problèmes d'échantillonnage. Une méthode de correspondance dense par programmation dynamique est retenue et développée pour sa relative simplicité d'implémentation, sa capacité à gérer certaines zones uniformes et sa bonne robustesse par rapport au coût calculatoire. Des modèles 3D locaux sont ainsi calculés. De plus, en configuration monoculaire, avec l'ajout d'informations odométriques, du SfM est effectué pour modéliser des scènes de rues complètes grâce à l'estimation visuelle du déplacement permise par l'algorithme de calibrage utilisé.

CALIBRAGE FISHEYE AUTOMATIQUE

Différents procédés permettent de modéliser l'environnement en 3D. Après la bibliographie vient le moment de décrire les choix et les travaux effectués. Dans ce projet, on utilise un système de stéréovision fisheye. Ce type de vision nécessite des études spécifiques. La connaissance de l'entraxe est nécessaire pour estimer les dimensions réelles.

Sommaire

3.1	Introduction	39
3.1.1	Généralités	41
3.1.2	Comparaison des profils des modèles de projection centrale	43
3.1.3	Estimation de la relation entre repère capteur et repère image	44
3.2	Calibrage d'un système multi-vues fisheye	46
3.2.1	Extraction automatique de paires de points correspondants	47
3.2.2	Projections étudiées formulées pour la méthode des 9 points	49
3.2.3	Algorithme des 9 points adapté à de nouveaux modèles fisheye	50
3.2.4	RANSAC et variantes	52
3.2.5	Affinement de l'étalonnage par Levenberg-Marquardt	57
3.3	Extraction de la géométrie épipolaire	57
3.3.1	Calcul d'une courbe épipolaire	58
3.3.2	Calcul des épipôles et test de stabilité	58
3.3.3	Calcul des rotations internes pour un alignement épipolaire	59
3.4	Évaluation du calibrage automatique	62
3.4.1	Scènes évaluées	62
3.4.2	Critères évalués	65
3.4.3	Seuil d'erreur angulaire	66
3.4.4	Choix du bon modèle de projection fisheye	68
3.4.5	Efficacité et stabilité entre RANSAC et LO-RANSAC	70
3.5	Conclusion	71

3.1/ INTRODUCTION

Nous souhaitons développer un système multi-vues fisheye capable de produire un modèle 3D de l'environnement. Que ces vues soient acquises par un stéréoscope ou par une seule caméra au cours du déplacement du véhicule, une étape de calibrage est requise. Elle permet de déterminer

les paramètres du modèle de projection auquel répond les caméras et les paramètres de la transformation reliant le contenu des vues fournies par le système. La figure 2.4 représente le mécanisme de projection sphérique utilisé pour modéliser la projection de tout point d'une scène sur le capteur d'une caméra munie d'un objectif fisheye.

La figure 3.1 illustre la géométrie épipolaire liée à un stéréoscope composé de deux caméras fisheye. Cette géométrie est la même lorsque les images proviennent d'une seule caméra fisheye en mouvement. Nous pouvons remarquer avec ce système que les caméras comportent deux épipôles e_1 et e_2 , au lieu d'un unique épipôle qui se projette théoriquement à l'infini dans le cas d'une caméra sténopé.

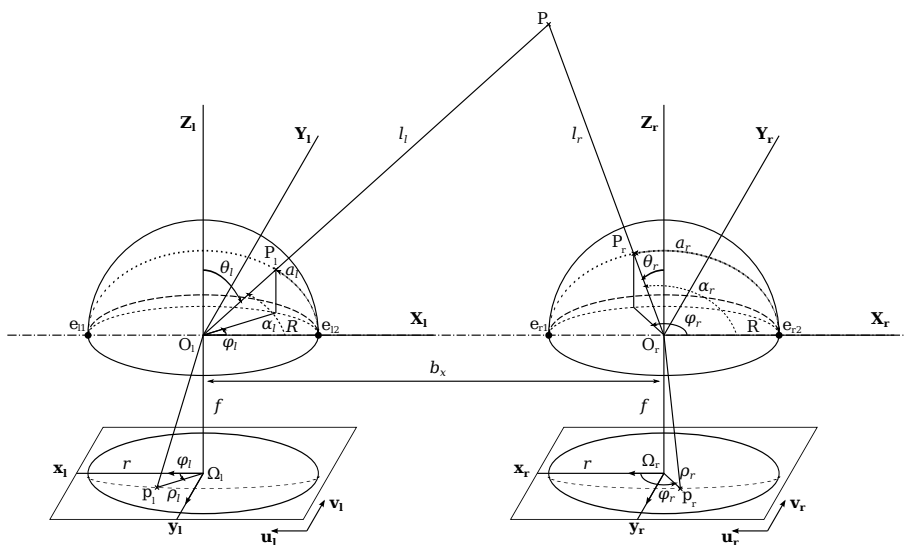


FIGURE 3.1 – Capteur stéréoscopique fisheye calibré. Le point P de la scène est projeté sur les sphères gauche et droite respectivement en P_l et P_r , et est vu comme p_l et p_r dans les images captées.

Dans la configuration décrite sur la figure 3.1, les points e_{l1} , O_l , e_{l2} , e_{r1} , O_r et e_{r2} sont alignés et forment l'axe épipolaire. Les courbes épipolaires sont définies comme les intersections du plan (O_l, P, O_r) avec les sphères représentant les objectifs fisheye. Les deux courbes épipolaires sont projetées sur le capteur des caméras suivant l'un des modèles de projection présentés dans le tableau 2.1. Comme pour le modèle sténopé, une étape de calibrage permet :

- d'estimer les paramètres du modèle reliant tout point P de la scène à sa projection p dans l'image ;
- de déterminer les paramètres requis pour le calcul des courbes épipolaires conjuguées le long desquelles l'étape de mise en correspondance est assurée.

Dans le cas d'un stéréoscope dont l'entraxe est faible, la plupart des techniques de calibrage utilisent une mire 2D aux dimensions et à la géométrie connues. Dans [Li, 2008], l'auteur utilise une mire 3D en demi-boîte.

Lorsque la distance entre les deux centres optiques est élevée, comme c'est le cas de nos configurations (voir sections 1.3.2 et 2.3.3.4), utiliser une mire ne convient plus car ses dimensions

doivent être très grandes. Une solution est de placer des marqueurs sur des bâtiments ou des structures bien connus comme le fait [Cannelle et al., 2012]. Toutefois, cette solution nécessite de placer le système installé sur le véhicule dans une configuration de calibrage donnée avant même de réaliser la mission d'acquisition et de reconstruction 3D. Par ailleurs, pour des capteurs embarqués sur un véhicule, [Lemondé, 2005] montre que les capteurs subissent les vibrations du véhicule en mouvement et que l'étalonnage n'est pas fiable dans le temps. Ainsi un recalibrage peut s'avérer nécessaire.

En première partie de ce chapitre nous proposons une méthode de calibrage automatique basée sur les travaux de [Mičušik et al., 2003, Mičušik, 2004]. Cette méthode repose sur la détection et l'appariement entièrement automatique d'un ensemble de points extraits des deux images. Mičušik propose un algorithme des 9 points qui est une adaptation au modèle de projection sphérique des algorithmes des 7 et 8 points présentés entre autres par [Hartley et al., 2004] pour le cas d'une paire d'images sténopé. Il utilise l'algorithme d'optimisation RANSAC afin de minimiser une erreur d'estimation de paramètres à partir de points extraits dans l'une et l'autre des images. Dans ce chapitre, nous décrivons l'algorithme des 9 points de Mičušik en l'étendant aux modèles de projection à angle équisolide et stéréographique en section 3.2.3, Mičušik l'ayant uniquement décrit avec la projection équidistante.

Les paramètres ainsi estimés permettent de déterminer la géométrie épipolaire reliant deux vues d'une même scène. Grâce à eux, il est possible de définir les courbes épipolaires conjuguées le long desquelles l'étape de mise en correspondance est réalisée.

Dans la deuxième partie de ce chapitre, nous décrivons une méthode géométrique capable de produire l'ensemble des courbes épipolaires conjuguées d'une paire d'images fisheye (voir section 3.3).

Enfin, la dernière partie est consacrée à l'évaluation de l'efficacité de la méthode sur données simulées et données réelles(voir section 3.4).

3.1.1/ GÉNÉRALITÉS

La projection fisheye est décrite de manière efficace au travers du modèle sphérique normalisé comme présenté à la figure 3.2. Ce modèle exploite une fonction de projection spécifique qui décrit le lien entre l'angle d'incidence et le rayon du point sur le capteur. La géométrie fisheye peut s'interpréter selon une coupe de profil avec symétrie radiale autour de l'axe optique. Les équations qui suivent établissent la relation entre le repère caméra 3D centré sur le centre optique de la caméra, et le repère 2D du plan capteur d'origine la projection du centre optique. Dans la figure 3.2 le profil est représenté par la fonction $g(r, \theta)$. C'est ce profil de courbe qui répartit la projection des points sur le capteur Π en décrivant le lien entre l'angle d'incidence θ et la distance r du point projeté au centre optique. Avec le modèle normé, Π est tangent à la sphère unitaire selon l'axe optique. r et θ sont reliés par la fonction de projection $r = \text{proj}(\theta)$ comme dans le tableau 2.1 de la section 2.2.3. Dans le cas d'une fonction inversible, le passage

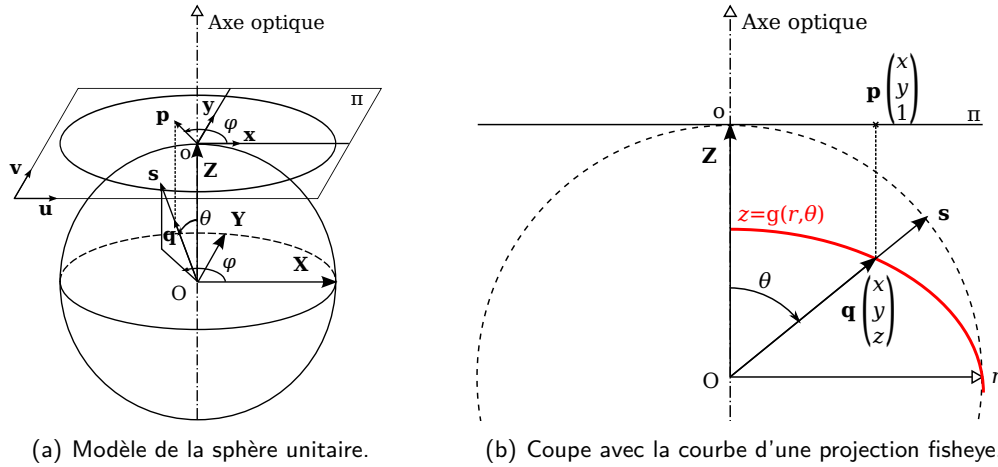


FIGURE 3.2 – Modélisation de la projection omnidirectionnelle fisheye avec centre de projection unique et symétrie autour de l'axe optique.

de r à θ s'exprime par $\theta = \text{proj}^{-1}(r)$ et

$$z = g(r, \theta) = \frac{r}{\tan(\theta)} \quad (3.1)$$

Dans ce manuscrit on n'emploie que des fonctions fisheye inversibles. z s'exprime de deux manières :

$$z = g(\theta) = \frac{\text{proj}(\theta)}{\tan(\theta)} \quad (3.2)$$

$$z = g(r) = \frac{r}{\tan(\text{proj}^{-1}(r))} \quad (3.3)$$

À partir de ces formulations on exprime la projection et la contre-projection fisheye de la manière suivante.

Projection de l'espace au plan capteur

Soit P un point de l'espace observé par une caméra fisheye et \mathbf{P} son vecteur associé dans le repère centré sur le centre optique de la caméra. Le vecteur \mathbf{P} est obtenu par la succession des étapes suivantes :

$$\mathbf{P} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \xrightarrow{\div \|\mathbf{P}\|} \mathbf{s} \begin{pmatrix} x_s \\ y_s \\ z_s = \cos(\theta) \end{pmatrix} \xrightarrow{\arccos \theta} \xrightarrow{g(\theta)} z \rightarrow \mathbf{q} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \mathbf{s} \cdot \frac{z}{z_q} \rightarrow \mathbf{p} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

\mathbf{s} est la projection de \mathbf{P} sur la sphère unitaire, d'angle d'incidence θ selon l'axe optique. \mathbf{q} est colinéaire à \mathbf{s} et va dans la même direction. Ils sont donc identiques à un facteur près. Connaissant \mathbf{s} on a $\theta = \arccos(z_s)$ et par conséquent z par application de l'équation (3.2). La relation de \mathbf{s} à \mathbf{q} est donnée par $\mathbf{q} = \mathbf{s} \frac{z}{z_q}$. \mathbf{p} est la projection orthogonale de \mathbf{q} sur Π selon

l'axe optique colinéaire à \mathbf{Z} .

Contre-projection depuis le plan capteur

La contre-projection d'un point \mathbf{p} du plan capteur Π aboutit à la demi-droite d'origine le centre optique O et de vecteur directeur unitaire \mathbf{s} . Elle représente toutes les positions de l'espace qui se projettent en \mathbf{p} .

$$\mathbf{p} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \rightarrow r = \sqrt{x^2 + y^2} \xrightarrow{g(r)} \mathbf{q} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \xrightarrow{\div \|\mathbf{q}\|} \mathbf{s} \begin{pmatrix} x_s \\ y_s \\ z_s \end{pmatrix}$$

Tout point \mathbf{p} du capteur Π se projette sur la rétine de la sphère unitaire par le rayon \mathbf{s} . o est la projection du centre optique O sur Π . Le rayon de \mathbf{p} au centre o est r . Connaissant r alors on a z d'après la formule (3.3). On dispose ainsi des 3 composantes de \mathbf{q} , qu'il suffit de normer pour aboutir à \mathbf{s} .

Selon le modèle de projection utilisé les paramètres de la fonction $r = \text{proj}(\theta)$ sont différents. Des modèles plus complexes et plus précis ont plus de paramètres. Dans la suite de ce manuscrit nous formulons le calibrage pour trois modèles centraux simples, de fonctions monotones et inversibles. Ils n'ont qu'un paramètre et correspondent aux trois types de projection fisheye suivants : la projection équidistante, la projection à angle équisolide et la projection stéréographique.

3.1.2/ COMPARAISON DES PROFILS DES MODÈLES DE PROJECTION CENTRALE

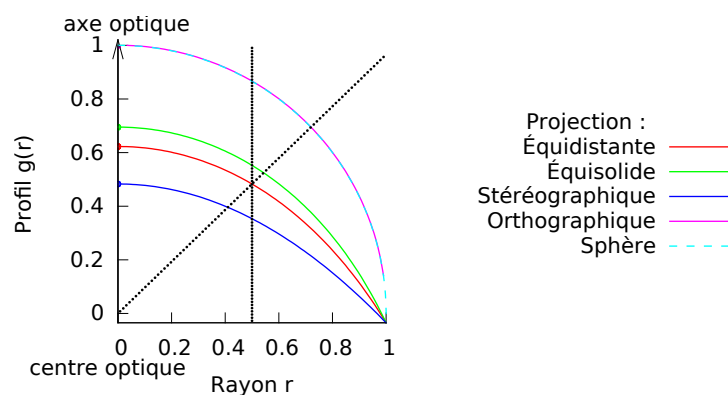


FIGURE 3.3 – Profils de modélisation des projections centrales présentées pour un champ de vision de 185°. Chaque rayon incident subit une projection orthogonale sur le capteur à sa rencontre avec le profil de modèle fisheye considéré.

La figure 3.3 montre les profils des courbes $g(r)$ que l'on obtient avec les modèles de projection centrale à 1 paramètre. La sphère est représentée pour le modèle sphérique et on remarque qu'elle est presque confondue avec le modèle de projection orthographique.

Prenons un point donné vu par l'objectif. Son rayon incident avec le centre optique rencontre

dans cet ordre le profil d'une sphère et le profil orthographique, le profil équisolide, le profil équidistant puis le stéréographique. La projection sur le capteur suivant chacun de ces modèles est à un rayon r de plus en plus petit selon le même ordre.

Les capteurs des caméras actuelles présentent une matrice de photosites régulière qui forme un échantillonnage fixe. Il est intéressant d'analyser la répartition de la scène sur les pixels de l'image en fonction du modèle de projection. En projection stéréographique plus l'angle d'incidence est fort, plus le nombre de pixels sur lesquels se projette la partie correspondante de la scène est élevé. Nous observons la tendance inverse pour la projection orthographique et la projections équisolide. Avec la projection équidistante, la répartition est divisée en moitiés mais pas de manière linéaire. D'après ces profils, les images obtenues en vision fisheye présentent moins de détails sur les éléments du bord du champ de vision que ceux du centre. Seule la projection stéréographique présente l'effet inverse. On peut voir ces effets sur les illustrations du tableau 2.1.

Cette dernière remarque est particulièrement intéressante dans le cadre de l'application CAPLOC puisque nous utilisons une caméra fisheye orientée vers le ciel pour laquelle les structures urbaines sont localisées à des angles d'incidence élevés. La projection stéréographique présente donc les caractéristiques optimales pour avoir la meilleure résolution sur les zones représentant les bâtiments. Dans la suite nous présentons un calibrage pour cette projection. Toutefois, il existe assez peu d'objectifs fisheye à projection stéréographique sur le marché. Nous avons acheté un tel objectif, un Samyang fisheye 8mm F3.5 CS, qui s'en approche et est disponible pour de nombreuses montures photo [Charles, 2009]. Toutefois, aucune image n'a encore été acquise avec cet objectif au moment où ce mémoire est rédigé.

3.1.3/ ESTIMATION DE LA RELATION ENTRE REPÈRE CAPTEUR ET REPÈRE IMAGE

Le passage d'un point du repère capteur (coordonnées en (x, y)) au repère image (coordonnées pixels en (u, v)) fait intervenir une transformation affine 2D A et une translation $\mathbf{t} \begin{pmatrix} u_0 \\ v_0 \end{pmatrix}$. Appelons $o_{px} \begin{pmatrix} u_0 \\ v_0 \end{pmatrix}$ la position de la projection du centre optique dans l'image. Les axes de ces repères sont dessinés dans la figure 3.2(a). On écrit :

$$\mathbf{p}_{px} = \begin{pmatrix} c & d \\ e & 1 \end{pmatrix} \mathbf{p}_{2D} + \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} = A \cdot \mathbf{p}_{2D} + \mathbf{t} \quad (3.4)$$

avec $\mathbf{p}_{2D} = \begin{pmatrix} x \\ y \end{pmatrix}$ et $\mathbf{p}_{px} = \begin{pmatrix} u \\ v \end{pmatrix}$. On a ainsi également :

$$\mathbf{p}_{2D} = A^{-1} (\mathbf{p}_{px} - \mathbf{t}) \quad (3.5)$$

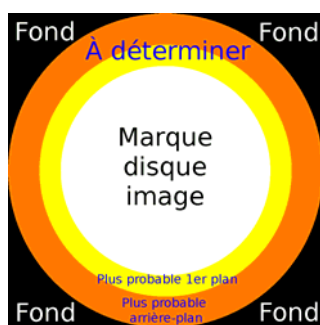
Les images projetées sur les capteurs forment des ellipses très proches d'un cercle et non pas un cercle parfait. Ceci est dû au fait que les photosites du capteur n'ont pas toujours des proportions exactement carrées et que les lignes et les colonnes de la matrice de photosites ne

sont pas parfaitement alignées, on appelle ce défaut d'alignement le cisaillement ou la non-orthogonalité (en anglais *skew*). Les valeurs c , d et e contenues dans A sont les paramètres décrivant la transformation de l'image en forme d'ellipse en un cercle parfait, en plus du changement d'échelle et de la rotation.

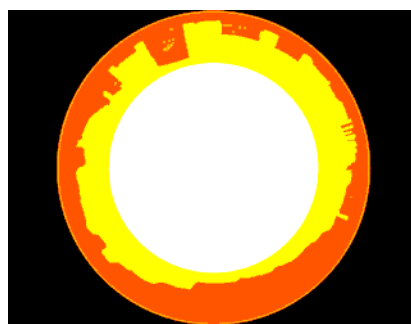
Dans ce manuscrit on propose une manière totalement automatique d'estimer A et t , en admettant une simplification supplémentaire : on néglige la différence d'agrandissement entre les deux axes et la non-orthogonalité pour considérer la projection d'un cercle parfait sur le capteur. Ainsi on pose $A = \text{Id}_2 \cdot r_{max}$.

En effet pour le modèle normé toutes les coordonnées des points dans le repère capteur par rapport au centre du disque image ont un rayon maximal égal à 1. r_{max} représente le rayon maximal en pixels dans les coordonnées pixelliques de l'image.

Pour estimer r_{max} et déterminer le centre du disque image $o_{px} \begin{pmatrix} u_0 \\ v_0 \end{pmatrix}$, on propose une étape automatique basée sur la détection du contour du disque image. C'est un avantage de la vision omnidirectionnelle, en sténopé on ne peut pas tirer parti du cercle de projection de l'optique.



(a) Initialisation.



(b) Résultat pour l'image avant de la paire Cap2/3039.

FIGURE 3.4 – Segmentation du disque image par *GrabCut*.

La détection du cercle est assurée par une étape de segmentation. Une segmentation simple par seuillage de l'intensité ne donne pas de bons résultats. Même si les points hors du disque image sont noirs ou presque, l'image elle-même peut contenir des éléments noirs, et inversement on peut voir apparaître des réflexions optiques en dehors du disque image qui causent la détection d'un cercle bien trop grand. Appliquer des lissages ne résout pas bien le problème et aboutit à un cercle détecté très imprécis. Une autre difficulté est que les bords du contour sont très flous, la frontière n'est pas nette.

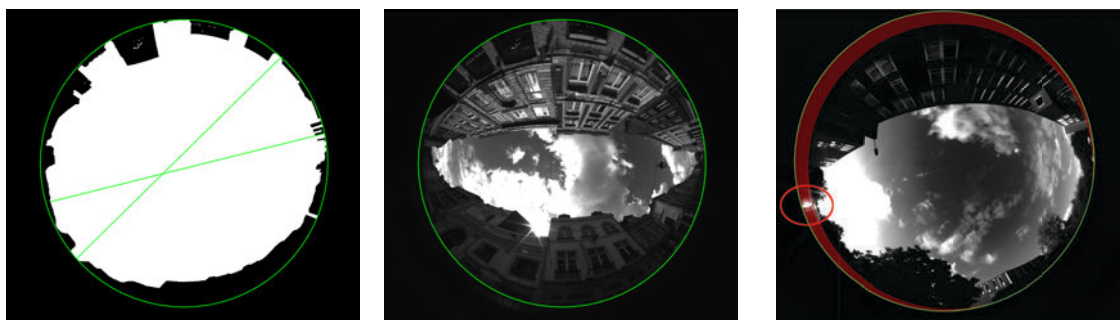
On propose donc d'utiliser une segmentation par *GrabCut*, connue pour sa très bonne robustesse. L'algorithme utilisé est l'implémentation *OpenCV* inspirée des travaux de [Rother et al., 2004]. Cette méthode nécessite une initialisation c'est-à-dire une pré-segmentation approximative de l'image. Le *GrabCut* cherche ensuite l'état des pixels inconnus.

Sur les images fisheye, à priori, les pixels centraux appartiennent au disque utile (le premier plan) et les pixels des quatre coins appartiennent au fond. On propose les marques initiales de la figure 3.4(a). Le disque est initialisé centré avec un diamètre égal à la dimension la plus faible

de l'image, et les deux niveaux d'incertitude forment les deux anneaux externes. On utilise le *GrabCut* avec une seule itération et les informations couleur.

La figure 3.4(b) montre le résultat de la segmentation et la figure 3.5 le résultat de la détection, pour une image de l'acquisition Cap2/3039 (utilisée dans les évaluations, voir section 3.4.1.2). L'équation du cercle est calculée avec trois points éloignés du contour de la zone segmentée, connu grâce à un détecteur de contours de type gradient appliqué au masque binaire extrait. Les deux premiers points du contour utilisés sont les plus éloignés entre eux. Pour une meilleure stabilité on propose de trouver la deuxième paire de points les plus éloignés entre eux, avec un écart d'au moins 30° entre la droite les joignant et la droite passant par les deux autres points. Le troisième point retenu pour calculer le cercle est celui qui est le moins proche de l'un des deux points de la première paire.

La détection du cercle donne de bons résultats mais les performances peuvent varier en fonction de certains phénomènes présents dans les images. Le cercle est bien détecté dans l'image de la paire Cap2/3039 qui est pourtant très sombre. Elle peut poser problème en présence de reflets dans les lentilles de l'objectif répercutant de la lumière sur le capteur à l'extérieur du disque de projection de l'image comme dans l'exemple 3.5(c).



(a) Segmentation et cercle détecté (b) Superposition du cercle sur l'image. (c) Détection faussée à cause d'un reflet projeté hors de la zone du disque image (entouré).

FIGURE 3.5 – Résultat de la détection du disque de l'image. (a) et (b) : pour l'image avant de l'acquisition Cap2/3039. (c) : exemple de cas où elle échoue.

La position du centre donne u_0 et v_0 , et le rayon donne la valeur de r_{max} . En configuration à deux caméras on devrait trouver des cercles de même rayon ou presque. Pour simplifier on considère qu'ils sont de même rayon, fixé à la valeur moyenne des deux rayons estimés.

3.2/ CALIBRAGE D'UN SYSTÈME MULTI-VUES FISHEYE

L'estimation des paramètres du modèle nécessite 9 paires de points appariés entre les deux vues. Même si 9 paires de points suffisent pour donner une estimation, la précision de l'estimation peut varier d'un ensemble de 9 paires de points à un autre. Il est donc classique d'utiliser un RANSAC (*R*andom *S*ample *C*onsensus) pour améliorer la robustesse de l'estimation afin d'identifier et rejeter les *outliers*, ces points qui ne font pas consensus. Cette étape de rejet est

assurée en testant l'ensemble des autres paires de points sur les paramètres calculés grâce aux 9 paires de points choisies aléatoirement au préalable. Le processus itératif est mené jusqu'à ce que la probabilité d'avoir convergé vers une bonne estimation du calibrage est élevée. 9 paires de points permettent de calibrer des systèmes fisheye répondant à des fonctions de projection centrale à un paramètre qui sont décrites dans le tableau 2.1 section 2.2.3.

Avec la méthode automatique proposée, pour simplifier le problème, on considère une projection parfaitement centrale en section 3.1.1, et on admet certaines approximations pour l'estimation d'un ensemble de paramètres internes en section 3.1.3. Le dernier paramètre interne à estimer est le paramètre a influençant la fonction de projection. La section 3.2.2 montre que a inclut l'information de focale f et diffère en fonction du modèle de projection employé. Les paramètres externes sont décrits par une matrice essentielle E ou fondamentale F . E contient la rotation et la translation qui expriment le passage d'une caméra à l'autre dans le monde réel et peut être décomposée. F contient les informations de E et de projection interne des caméras, elle représente en fait le changement de repère d'une image à l'autre (à distinguer des repères caméra).

Cette section explique comment étalonner simultanément les derniers paramètres nécessaires pour la reconstruction 3D : a et E . La méthode RANSAC exploite un ensemble de paires de points donné et est paramétrée en section 3.2.4, ainsi qu'une amélioration LO-RANSAC plus robuste avec boucles internes ([Chum, 2005]).

3.2.1/ EXTRACTION AUTOMATIQUE DE PAIRES DE POINTS CORRESPONDANTS

Une partie du travail a été de définir quels points sont à extraire des images et quels descripteurs nous devrions utiliser pour produire l'ensemble des paires de points appariés. Ce choix n'est pas anodin car nous devons détecter un nombre élevé de points et être capable de les appairer avec précision. Il existe de nombreux algorithmes de détection de points ou d'éléments d'intérêt dans les images. Selon leur nature et l'objectif qui a guidé leur développement, ils sont plus ou moins efficaces et robustes à certaines conditions. [Mikolajczyk et al., 2005] indiquent les différents types de robustesses attendus :

- la robustesse au flou (ou plus exactement au manque de netteté qui peut avoir plusieurs causes comme par exemple le *flare*, une optique de mauvaise qualité, etc),
- la robustesse aux changements de point de vue,
- la robustesse aux changements d'échelle (ou zoom dans le cas de changements de focale),
- la robustesse aux rotations (orientations),
- la robustesse aux changements de luminosité (clair et sombre, variations de la couleur de l'éclairage environnant, ombres, etc),
- la robustesse à la compression des enregistrements numériques des images avec pertes, et en particulier le format JPEG, ces méthodes d'enregistrement pouvant créer des artefacts.

Pour l'application de stéréovision omnidirectionnelle, les algorithmes employés doivent impérativement être robustes aux :

- changements de rotations et orientations, car les caméras embarquées suivent le mouvement du véhicule et qu'il rencontre forcément des tournants au cours de son parcours ;
- aux changements de point de vue, car le véhicule se déplaçant, la scène sera observée de différents points de vues ;
- et aux changements d'échelle, car un élément n'aura pas la même taille dans l'image en fonction du point de vue, mais aussi car les déformations optiques des objectifs fisheye font que plus les éléments sont situés en périphérie, plus ils sont déformés dans l'image.

Le tableau 3.1 est une liste d'algorithmes de détection de points d'intérêt proposés dans la littérature, auxquels sont associées les propriétés de robustesse nécessaires aux applications de vision omnidirectionnelle. Le tableau 3.2 est une liste équivalente pour les descripteurs.

TABLEAU 3.1 – Liste des détecteurs étudiés. Les détecteurs marqués en gras sont ceux qui sont donnés assez robustes aux changements d'échelle et d'orientation, et qui ont donc été retenus pour les évaluations sur les images fisheye.

Algorithme	Éléments	Robuste échelles	Robuste orientations
Harris [Harris et al., 1988]	coins	non	non
Shi-Tomasi [Shi et al., 1994]	coins	non	non
Shi-Tomasi sur Harris	coins	non	non
SIFT [Lowe, 2004]	<i>blobs</i>	oui	oui
FAST [Rosten et al., 2006]	coins	non	non
SURF [Bay et al., 2006]	<i>blobs</i>	oui	moyen
CenSurE [Agrawal et al., 2008]	<i>blobs</i> ⁽¹⁾	oui	moyen+
MSER [Nistér et al., 2008]	régions	faible	faible
ORB [Rublee et al., 2011]	coins	oui	oui
BRISK [Leutenegger et al., 2011]	coins	oui	oui

⁽¹⁾ : Le CenSurE détecte des *blobs*, mais dont le but est d'être plus précis que des coins.

Nous avons réalisé des évaluations avec les algorithmes de détection SIFT, CenSurE, ORB et BRISK que nous avons croisés avec les descripteurs SIFT, ORB, BRISK et FREAK. Nous avons utilisé les métriques suivantes pour comparer leur performances dans le contexte d'une projection sphérique :

- la répétabilité du détecteur qui mesure la capacité du détecteur (seul) à détecter les mêmes points dans deux images différentes ;
- le taux de bons appariements qui mesure la capacité de la combinaison détecteur et descripteur à atteindre un appariement correct.

L'évaluation a été menée sur plusieurs séquences d'images de synthèse d'une caméra fisheye en mouvement, chacune d'entre elles présentant des difficultés différentes. Ces séquences sont présentées en section 3.4.1.1 et exploitées également pour évaluer le calibrage global. Ayant connaissance des caractéristiques géométriques et dimensionnelles des scènes, les métriques ont été exploitées en définissant l'identité de deux points (ou leur appariement) dans l'espace 3D. L'appariement des points a également été évalué, en comparant les résultats pour trois algorithmes de mises en correspondance.

TABLEAU 3.2 – Liste des descripteurs étudiés. Les descripteurs marqués en gras sont ceux qui sont donnés assez robustes aux changements d'échelle et d'orientation, et qui ont donc été retenus pour les évaluations sur les images fisheye.

Algorithme	Robuste échelles	Robuste orientations
SIFT [Lowe, 2004]	oui	oui
SURF [Bay et al., 2006]	oui	moyen
BRIEF [Calonder et al., 2010]	non	non
ORB [Rublee et al., 2011]	oui	oui
BRISK [Leutenegger et al., 2011]	oui	oui
FREAK [Alahi et al., 2012]	oui	oui

Nous ne présentons pas tous les résultats de cette étude dans ce mémoire pour ne pas le surcharger. Toutefois, nous partageons ses conclusions qui indiquent le détecteur SIFT associé au descripteur SIFT et un appariement par recherche exhaustive comme le choix optimal dans le contexte fisheye de cette thèse. Compte-tenu des résultats obtenus, pour une scène en conditions moyennes (principalement flou de mouvement par translation), on peut estimer que l'on obtient avec ce choix un taux de bons appariements proche de 35% pour une tolérance de 2cm. L'étude indique également que le détecteur CenSurE associé au descripteur ORB, plus rapides, peuvent être un choix pertinent si le temps de calcul est une contrainte forte de l'application.

3.2.2/ PROJECTIONS ÉTUDIÉES FORMULÉES POUR LA MÉTHODE DES 9 POINTS

Comme le décrivent [Hughes et al., 2010], en réalité, les objectifs ne suivent pas exactement les modèles centraux théoriques bien qu'ils soient étudiés pour les approcher. Ils montrent que les modèles centraux à un paramètre ne sont pas les plus précis, mais qu'ils peuvent être suffisamment bons lorsqu'ils sont utilisés avec les objectifs les approchant. Ils montrent aussi que ces modèles sont meilleurs que les modèles centraux plus complexes lorsqu'on les accompagne d'un paramètre de compensation des distorsions supplémentaire. Ce sont donc les modèles de base à privilégier.

Les paramètres sont différents selon le modèle de projection utilisé. Dans nos expérimentations les modèles centraux simples présentent un seul paramètre a . Ils ont pour autre avantage d'être inversibles, c'est-à-dire que la projection et la contre-projection peuvent être faites toutes les deux avec l'application directe d'une fonction. Ceci permet de réduire les coûts en calculs et en mémoire.

Les fonctions des modèles étudiés sont formulées dans le tableau 3.3. La valeur de a dépend de r_{max} et θ_{max} . Afin de déterminer a , il est nécessaire de définir une valeur initiale a^* de a à partir de laquelle le RANSAC est lancé. En posant $r_{max} = 1$ pour correspondre au modèle de la sphère unitaire (section 3.1.3) et en fixant une valeur approchée $\tilde{\theta}_{max}$ de θ_{max} grâce aux données du fabriquant de l'objectif, a^* peut se calculer en utilisant $r_{max} = 1$ et $\tilde{\theta}_{max}$.

$r = \text{proj}(\theta)$	$\theta = \text{proj}^{-1}(r, a)$	$g(r, a)$	Calcul de a
Projection équidistante (Cfg 2)			
$r = f.\theta$	$\theta = a.r$	$g(r, a) = \frac{r}{\tan(a.r)}$	$a = \frac{1}{f} = \frac{\theta_{max}}{r_{max}}$
Projection à angle équisolide (Cfg 1), (Cfg 3) et (Cfg 4)			
$r = 2f \sin(\frac{\theta}{2})$	$\theta = 2 \arcsin(a.r)$	$g(r, a) = \frac{r}{\tan(2 \arcsin(a.r))}$	$a = \frac{1}{2f} = \frac{\sin(\frac{\theta_{max}}{2})}{r_{max}}$
Projection stéréographique			
$r = 2f \tan(\frac{\theta}{2})$	$\theta = 2 \arctan(a.r)$	$g(r, a) = \frac{r}{\tan(2 \arctan(a.r))}$	$a = \frac{1}{2f} = \frac{\tan(\frac{\theta_{max}}{2})}{r_{max}}$

TABLEAU 3.3 – Fonctions de modélisation des projections fisheye centrales à un paramètre étudiées et définition du paramètre a correspondant.

3.2.3/ ALGORITHME DES 9 POINTS ADAPTÉ À DE NOUVEAUX MODÈLES FISHEYE

Cette estimation s'appuie sur la résolution d'un PEP (*Polynomial Eigenvalue Problem*, problème aux valeurs propres), problème non-linéaire basé sur l'équation de la géométrie épipolaire pour un nombre minimal de paires de point correspondants. Elle est formulée entre autres par [Baselgia, 2010] (dans son équation 3.4), avec les notations de ce manuscrit :

$$\mathbf{q}_2^T \mathbf{E} \mathbf{q}_1 = 0 \quad (3.6)$$

L'algorithme des 9 points se distingue des algorithmes des 7 et 8 points en cela qu'il estime non seulement E mais aussi le paramètre de la fonction modélisant la projection fisheye dans un paramètre a séparé. Dans ce cas particulier on arrive à la résolution d'un QEP (*Quadratic Eigenvalue Problem*, problème aux valeurs propres non-linéaires d'ordre 2). Pour le résoudre, il faut construire des matrices spécifiques D_i adaptées au modèle de projection. L'algorithme utilise 9 paires de points correspondants dans les images gauche et droite, obtenues préalablement par une méthode automatique (voir section 3.2.1). Les matrices D_i sont construites à partir d'approximations des projections des points en rayons 3D, basées sur une valeur initiale a^* donnée proche de la valeur optimale du paramètre a à déterminer ; la valeur de a n'étant pas encore connue.

Le choix d'une bonne estimation de a^* est important pour obtenir de bons résultats de calibrage. L'expression de a^* dépend de la fonction de projection utilisée, il faut estimer a^* avec une estimation de θ_{max} connue grâce aux données constructeur (section 3.2.2).

La méthode pour obtenir les matrices D_i et résoudre le QEP est inspirée de ce qui est décrit dans [Mičušik et al., 2003, Mičušik, 2004]. Pour utiliser l'un ou l'autre des modèles présentés dans le manuscrit, il suffit de choisir l'équation de $g(r, a)$ dans le tableau 3.3 selon le modèle de projection fisheye adopté, et de l'appliquer suivant la méthode décrite ci-après.

La série de Taylor de degré 1 $\tilde{g}(r, a)$ de $g(r, a)$, avec le paramètre a proche de la valeur a^* , est

donnée par :

$$\tilde{g}(r, a) = g(r, a^*) + \frac{\partial g(r, a^*)}{\partial a} (a - a^*) \quad (3.7)$$

[Mičušík, 2004] développe l'algorithme des 9 points avec la projection fisheye équidistante. Il arrive ainsi à une expression de $\tilde{g}(r, a)$ définie avec $g(r, a^*)$ pour le modèle équidistant (voir le tableau 3.3) et en conséquence avec :

$$\frac{\partial g(r, a^*)}{\partial a} = -(1 + \tan^2(a^*r)) \left(\frac{r}{\tan(a^*r)} \right)^2$$

Dans les travaux de thèse présentés dans ce manuscrit et la section 3.2.2, nous montrons qu'il est plus intéressant d'utiliser des objectifs fisheye avec des caractéristiques mieux adaptées aux besoins applicatifs. Nous étudions donc en plus deux autres fonctions de projection, pour lesquelles nous obtenons les équations suivantes (voir section 3.2.2) :

— **Projection à angle équisolide,**

$$\frac{\partial g(r, a^*)}{\partial a} = -[1 + \tan^2(2 \arcsin(a^*r))] \frac{2}{\sqrt{1 - a^{*2}r^2}} \left(\frac{r}{\tan(2 \arcsin(a^*r))} \right)^2$$

— **Projection stéréographique,**

$$\frac{\partial g(r, a^*)}{\partial a} = -[1 + \tan^2(2 \arctan(a^*r))] \frac{2}{1 + a^{*2}r^2} \left(\frac{r}{\tan(2 \arctan(a^*r))} \right)^2$$

Le vecteur $\tilde{\mathbf{q}}$ s'exprime par :

$$\tilde{\mathbf{q}} = \begin{pmatrix} \mathbf{p}_{2D} \\ \tilde{g}(r, a) \end{pmatrix} = \begin{pmatrix} \mathbf{p}_{2D} \\ g(r, a^*) - a^* \frac{\partial g(r, a^*)}{\partial a} + a \frac{\partial g(r, a^*)}{\partial a} \end{pmatrix} = \begin{pmatrix} \mathbf{p}_{2D} \\ \tilde{z} \end{pmatrix} + a \begin{pmatrix} \mathbf{0}_{2 \times 1} \\ s \end{pmatrix} \quad (3.8)$$

Donc on a :

$$\tilde{z} = g(r, a^*) - a^* \frac{\partial g(r, a^*)}{\partial a} \text{ et } s = \frac{\partial g(r, a^*)}{\partial a}$$

De la même manière pour tous les modèles à un paramètre, on utilise l'équation ^(3.6) pour estimer E à partir d'approximations des autres valeurs :

$$\tilde{\mathbf{q}}_2^T \mathbf{E} \tilde{\mathbf{q}}_1 = 0 \quad (3.9)$$

que l'on développe pour la rendre applicable au QEP (voir [Mičušík, 2004]) :

$$(\mathbf{D}_1 + a\mathbf{D}_2 + a^2\mathbf{D}_3)\mathbf{f} = 0 \quad (3.10)$$

où les matrices $\mathbf{D}_i \in \mathbb{R}^{9 \times 9}$ sont construites ligne par ligne à partir des 9 paires de points

correspondants comme suit :

$$\begin{aligned} \text{ligne}_{D_1} &= [x_2x_1 \ x_2y_1 \ x_2\tilde{z}_1 \ y_2x_1 \ y_2y_1 \ y_2\tilde{z}_1 \ \tilde{z}_2x_1 \ \tilde{z}_2y_1 \ \tilde{z}_2\tilde{z}_1], \\ \text{ligne}_{D_2} &= [0 \ 0 \ x_2s_1 \ 0 \ 0 \ y_2s_1 \ s_2x_1 \ s_2y_1 \ \tilde{z}_2s_1 + s_2\tilde{z}_1], \\ \text{ligne}_{D_3} &= [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ s_2s_1]. \end{aligned}$$

avec x_j , y_j , \tilde{z}_j et s_j issus des expressions des $\tilde{\mathbf{q}}_j$ correspondant à un couple de points dans les deux images identifiées par $j = 1$ et 2 .

Avec la procédure telle qu'elle est décrite on estime la matrice essentielle E , et non pas la matrice fondamentale F . F décrit le passage d'un point de l'image 1 \mathbf{p}_1 à un point de l'image 2 \mathbf{p}_2 . Elle tient compte de la projection interne des caméras qui dépend du capteur et de l'objectif. Là on utilise l'équation à partir des projections fisheye \mathbf{q}_i des points \mathbf{p}_i , donc après sortie du repère image au repère caméra. En effet c'est le passage à \mathbf{q}_i qui tient compte des transformations internes. Les paramètres internes sont estimés à part, ils ne sont pas intégrés à la matrice essentielle. La matrice essentielle décrit le passage d'un point de la fonction fisheye de la caméra 1 \mathbf{q}_1 à un point de la projection fisheye de la caméra 2 \mathbf{q}_2 .

À la résolution du QEP à partir des matrices D_i , on obtient 9 solutions potentielles. Seul un couple (\mathbf{f}_i, a_i) est correct, où i est la i^{e} solution potentielle.

Tout d'abord il ne faut conserver que les valeurs a_i cohérentes pour le calibrage, c'est-à-dire les valeurs qui sont des réels purs (non complexes), positifs et non infinis. En principe, de une à quatre solutions sont conservées, et elles doivent être évaluées pour déterminer la meilleure d'entre elles. Pour chaque valeur a_i possible il y a le vecteur colonne \mathbf{f}_i correspondant, dont les éléments permettent la construction d'une matrice essentielle E possible. La solution donnant l'erreur globale la plus faible est considérée comme la meilleure.

Avec l'algorithme des 9 points, on détermine simultanément E et le paramètre a de la fonction de projection pour les caméras. Ce paramètre est identique aux deux caméras, la précision n'est pas forcément aussi bonne que si on estimait le paramètre de chaque caméras individuellement. L'intérêt de la matrice essentielle, c'est qu'elle est directement décomposable pour extraire la rotation de la caméra 1 à la caméra 2, ainsi que le vecteur de translation à un facteur d'échelle près. [Baselgia, 2010] explique la décomposition de E à cet effet dans son manuscrit présentant les projections sténopé et fisheye pour la stéréovision.

3.2.4/ RANSAC ET VARIANTES

C'est la méthode RANSAC qui régit le calibrage automatique. L'idée est d'arriver à estimer correctement l'étalonnage, sachant que l'ensemble des données utilisées (les paires de points entre les deux vues) contient des éléments corrects (*inliers*) et incorrects (*outliers*). La difficulté c'est qu'on ne sait pas lesquels sont *inliers*, c'est le RANSAC qui permet de les retenir. Pour cela il se base sur des tirages aléatoires d'un sous-ensemble minimal pour estimer

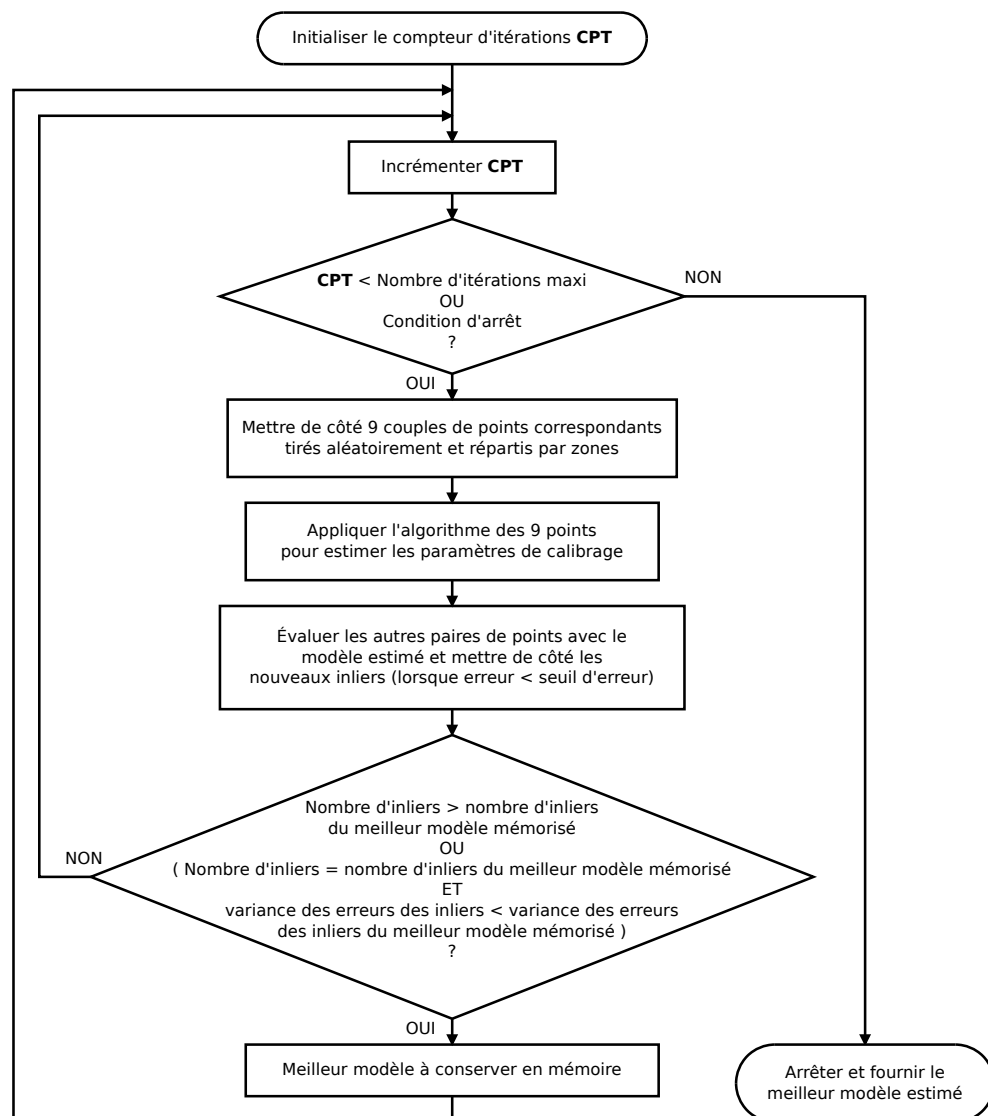


FIGURE 3.6 – Pseudo-algorithme du RANSAC pour l'application de l'algorithme des 9 points.

le calibrage. C'est là qu'intervient l'algorithme des 9 points. Le résultat est ensuite évalué pour le sous-ensemble minimal, puis s'il est assez bon, il est évalué pour l'ensemble complet des points. L'évaluation est une mesure d'erreur détaillée en section 3.2.4.1. Le statut *inlier* ou *outlier* de chaque paire de points est donné selon un seuil sur cette évaluation. À la fin des itérations RANSAC, le modèle retenu est celui ayant aboutit au plus grand ensemble d'*inliers*. Les étapes de l'algorithme de calibrage sont présentées dans le pseudo-algorithme figure 3.6.

Les solutions possibles pour le paramètre a et la matrice essentielle E sont estimées à partir des points choisis et d'une valeur initiale a^* correspondant à une valeur approchée connue de a .

Les estimations calculées sont évaluées en fonction du nombre d'*inliers* qu'elles impliquent. Un meilleur modèle est un modèle avec plus d'*inliers*. En cas d'égalité, nous choisissons de retenir

comme meilleure estimation celle où la variance de l'erreur des *inliers* est la plus faible.

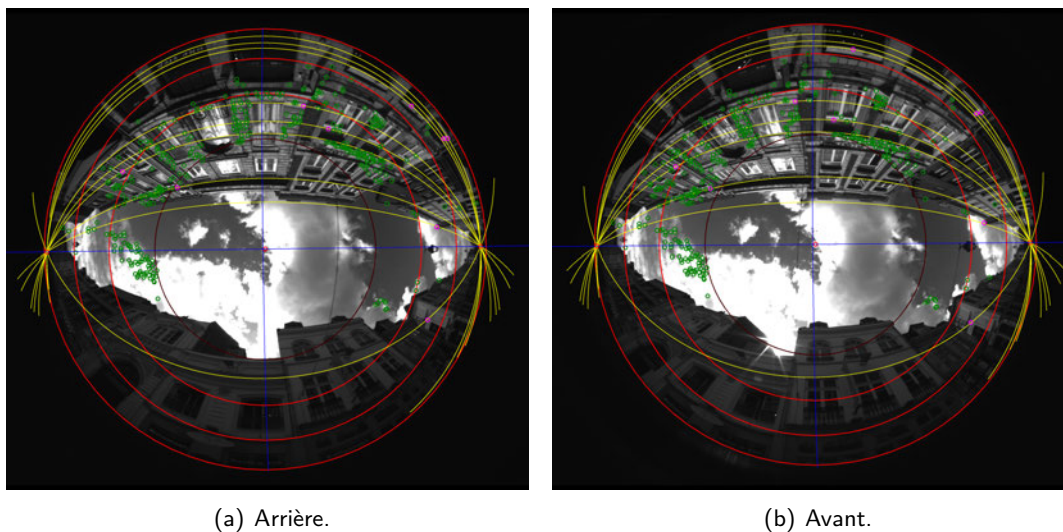


FIGURE 3.7 – Illustration du calibrage avec la paire Cap2/3039 des acquisitions de Lille.

[Mičušík, 2004] montre qu'utiliser une méthode de répartition par zones pour les points choisis aboutit à une meilleure stabilité dans l'estimation du modèle. L'objectif est de sélectionner le même nombre de points par zones pour l'algorithme des 9 points, en fonction de leur distance au centre. Cela consiste à diviser le disque image en quatre zones concentriques de même aire : le disque central et trois anneaux.

En premier lieu, il recommande de ne pas utiliser les points de la zone centrale car ils ne contiennent pas assez d'informations permettant de bien définir les paramètres de l'objectif (les distorsions sont trop faibles). Les trois anneaux fournissent des points avec des taux de distorsions de paliers plus forts en fonction de l'éloignement au centre. D'après lui, une bonne distribution des points est choisie en prenant autant points dans chacun des trois anneaux. De cette manière, on augmente les chances d'obtenir une bonne estimation.

On voit le tracé de ces anneaux en rouge dans la figure 3.7 qui montre le résultat du calibrage pour la paire Cap2/3039 des acquisitions de Lille.

3.2.4.1/ PARAMÉTRAGE DU RANSAC

Condition d'arrêt du RANSAC

En RANSAC un meilleur modèle est un modèle validé par un plus grand nombre d'éléments de l'ensemble traité. [Chum, 2005] indique que la condition d'arrêt du RANSAC se base sur la probabilité de ne pas avoir trouvé un meilleur modèle que celui trouvé après les itérations passées. Aux premières itérations, cette probabilité vaut toujours 1 et diminue petit à petit que l'on trouve des modèles avec plus d'*inliers* et que l'on fait plus d'itérations. Un seuil classique à comparer avec cette probabilité pour terminer le RANSAC est $s_0 = 95\%$. On pourrait baisser ce seuil mais le temps de calcul serait décuplé pour terminer avec un modèle rarement plus précis.

En pratique on ne trouve normalement pas le tout meilleur modèle, mais l'un des meilleurs. En fonction des différentes conditions à respecter, le nombre d'itérations RANSAC qu'il faudrait pour trouver une bonne estimation est ([Chum, 2005]) :

$$\frac{\ln(1 - s_0)}{\ln(1 - P(I))} \quad \text{avec} \quad P(I) = \underbrace{\prod_{j=0}^{m-1} \frac{I - j}{N - j}}_{\text{D'autant plus proches que } N \text{ est grand}} \leq \left(\frac{I}{N}\right)^m$$

m est le nombre de paires de points tirées aléatoirement dans le RANSAC, m vaut 9. I est le nombre de bons appariements, N est le nombre total de paires de points. Compte-tenu des taux de bons appariements auxquels nous avons abouti lors de l'évaluation des détecteur et des descripteurs nous nous plaçons un cas moyen où le taux de bons appariements $\frac{I}{N} = 0,35$ (voir section 3.2.1). On peut approximer le calcul par :

$$\frac{\ln(1 - 0,95)}{\ln(1 - 0,35^9)} = 38008 \text{ arrondi au supérieur}$$

On devrait arriver à une bonne estimation à environ 38008 itérations. Avec cette information, on a choisi de limiter l'algorithme RANSAC à 50000 exécutions s'il ne converge pas correctement avec la condition d'arrêt. Ce cas ne l'empêche pas pour autant de trouver une estimation au modèle étudié. Il est courant de voir des auteurs calculer le nombre d'itérations théorique nécessaire de la même manière, puis effectuer simplement un RANSAC à ce nombre donné d'itérations. C'est plus simple mais on ne peut pas rendre compte si on est en conditions difficiles ou non car le nombre d'itérations variables en est un bon indicateur.

Mesure d'erreur pour une paire de points correspondants

Pour évaluer la qualité des estimations obtenues sur chaque paire de points il faut se baser sur une mesure d'erreur. Les rayons 3D estimés avec les paramètres de projection et la matrice essentielle E à évaluer sont calculés pour chaque couple de points correspondants entre les deux images. Puis, l'erreur angulaire directionnelle de chaque paire de vecteurs est mesurée selon la formulation donnée par [Oliensis, 2002]. Il propose cette approximation d'une erreur angulaire et démontre sa meilleure robustesse avec des caméras à grand angle de vision face à la mesure d'erreur géométrique de Sampson réputée très bonne ([Oliensis, 2002], [Hartley et al., 2004]). L'erreur angulaire a été reprise par [Mičušik, 2004] pour la vision omnidirectionnelle. On exprime cette erreur ϵ à partir des rayons normés par :

$$\epsilon(\mathbf{s}_1, \mathbf{s}_2, E) = \frac{\mathbf{A}}{2} - \sqrt{\frac{\mathbf{A}^2}{4} - \mathbf{B}}, \quad (3.11)$$

avec

$$\mathbf{A} = \mathbf{s}_1^T E^T E \mathbf{s}_1 + \mathbf{s}_2^T E E^T \mathbf{s}_2 \text{ et } \mathbf{B} = (\mathbf{s}_2^T E \mathbf{s}_1)^2.$$

L'avantage d'avoir une telle expression explicite d'une mesure d'erreur est que nous n'avons

plus à nous soucier d'autres critères à minimiser que celui-ci. Cette erreur est un angle, et on peut l'exploiter aussi bien avec la projection sphérique qu'avec la projection sténopé.

On a vu en section 3.2.4 que l'on considère *inlier* un couple de points dont l'erreur directionnelle angulaire est inférieure à un seuil donné.

Il est bon de définir ce seuil en fonction des caractéristiques du capteur et des images extraites pour que l'algorithme se comporte toujours de la même manière indépendamment de la résolution des images. Pour cela on a besoin du champ de vision de l'objectif (on dispose d'une approximation d'après les données constructeur égale à $2\tilde{\theta}_{max}$) et du rayon r_{max} du disque image projeté en pixels. Un décalage de 1 pixel selon le cercle équateur aux épipôles donne un nouveau plan épipolaire. On a donc un décalage angulaire moyen de $\frac{\theta_{max}}{r_{max}}$ entre chaque plan épipolaire (ce décalage est vrai en tout point pour une projection sphérique parfaite, pas pour une fonction de projection fisheye quelconque, mais la moyenne est la même). On propose d'exprimer le seuil en proportion p de cette grandeur :

$$\epsilon_{seuil} = p \cdot \frac{\tilde{\theta}_{max}}{r_{max}}$$

Différentes valeurs de $p \in \{0, 01; 0, 001; 0, 0001; 0, 00001\}$ sont testées en section 3.4.3.

3.2.4.2/ LO-RANSAC ROBUSTE APPLIQUÉ À L'ALGORITHME DES 9 POINTS

Pour le cas sténopé, [Chum, 2005] montre que l'algorithme des 7 points est largement employé car il est plus précis que l'algorithme des 8 points et nécessite une paire de correspondants de moins. Il est donc plus intéressant dans le cadre du RANSAC. Toutefois, la précision de l'algorithme des 7 points n'est meilleure que celle de l'algorithme (dit) des 8 points que lorsque ce dernier est utilisé avec 8 points. L'algorithme des 8 points peut s'utiliser avec plus de 8 paires de points correspondants. L'avantage d'utiliser plus de points dans l'optimisation linéaire est d'être moins sensible au bruit et d'arriver à une estimation bien plus précise (voir [Hartley, 1997]).

L'algorithme des 9 points peut aussi s'utiliser avec plus de 9 paires de points correspondants. Rappelons que l'on cherche à résoudre l'équation 3.10. Si on utilise plus de 9 points les matrices D_i ne sont pas carrées. Pour pouvoir résoudre le QEP ces matrices doivent être carrées. [Mičušik, 2004] indique que l'on peut aboutir à la même solution en modifiant l'équation 3.10 en :

$$\left(\underbrace{D_1^T D_1}_{\text{matrice carrée}} + a \underbrace{D_1^T D_2}_{\text{matrice carrée}} + a^2 \underbrace{D_1^T D_3}_{\text{matrice carrée}} \right) \mathbf{f} = 0 \quad (3.12)$$

[Chum, 2005] montre que l'on peut faire des estimations à partir de plus de points dans des boucles internes au RANSAC, sur la population d'*inliers* trouvée dans le RANSAC père à chaque fois que l'on trouve une population plus grande. Il appelle cette évolution au RANSAC LO-RANSAC, comme (*Local Optimized RANSAC*), elle permet d'accélérer la convergence globale de l'estimation. Il compare LO-RANSAC à d'autres variantes du RANSAC et montre de

meilleurs résultats.

Il distingue plusieurs variantes de LO-RANSAC, et d'après ses expérimentations avec l'algorithme des 8 points, la meilleure est celle qui utilise un RANSAC interne avec itérations. En conséquence, nous proposons de reprendre le LO-RANSAC pour l'algorithme des 9 points avec la variante utilisant un RANSAC interne avec itérations pour la meilleure efficacité possible. Les RANSAC internes sont répétés 10 fois, et le processus itératif part d'un multiplicateur de 4 réduit de un en un jusqu'à 1.

3.2.5/ AFFINEMENT DE L'ÉTALONNAGE PAR LEVENBERG-MARQUARDT

La précision des estimations peut être améliorée en tenant compte de tous les *inliers* au lieu d'utiliser une estimation basée sur un ensemble réduit de points. Dans [Chum, 2005], Chum ne parle pas d'étape d'affinement, il se limite à l'amélioration de la précision par l'utilisation d'un plus grand sous-ensemble d'éléments dans le LO-RANSAC.

Dans les travaux présentés dans ce manuscrit, une étape d'affinement basée sur le Levenberg-Marquardt est ajoutée, comme le font [Hartley et al., 2004] et [Geyer et al., 2007]. Le Levenberg-Marquardt est un algorithme d'optimisation non linéaire par minimisation des moindres carrés. Il est plus stable que le Gauss-Newton. Les valeurs initiales pour l'algorithme de Levenberg-Marquardt sont issues de la meilleure estimation donnée par le RANSAC. Le Levenberg-Marquardt est appliqué en tenant compte de l'ensemble complet des *inliers*.

On exprime l'erreur $\epsilon(\mathbf{s}_1, \mathbf{s}_2, E)$ par l'équation (3.11).

Nous avons besoin des jacobiniennes de ϵ (donc de A et B), dont les variables sont les paramètres à déterminer, c'est-à-dire les valeurs des éléments de E. Ainsi on exprime :

$$\epsilon'(\mathbf{s}_1, \mathbf{s}_2, E) = \frac{A'}{2} - \frac{\left(\frac{A^2}{4} - B\right)'}{2\sqrt{\frac{A^2}{4} - B}} = \frac{A'}{2} - \frac{A'A - 2B'}{4\sqrt{\frac{A^2}{4} - B}} \quad (3.13)$$

Pour affiner les paramètres de la fonction de calibrage, il faut rajouter la prise en compte de dérivées selon eux. Elles sont toutes égales à 0.

3.3/ EXTRACTION DE LA GÉOMÉTRIE ÉPIPOLAIRE

Comme on le voit dans la figure 3.1, il est possible de détecter les courbes épipolaires conjuguées et de les utiliser pour mettre en correspondance chacun des points leur appartenant. Nous proposons une technique de recherche des courbes épipolaires conjuguées qui consiste à déterminer les rotations des caméras pour se placer dans le cas où l'entraxe est aligné avec l'axe \mathbf{X} . Nous appelons ces rotations les rotations épipolaires, pour passer des repères caméras aux repères épipolaires de base commune. Ce cas parfaitement aligné permet de travailler plus facilement avec les courbes épipolaires conjuguées.

3.3.1/ CALCUL D'UNE COURBE ÉPIPOLAIRE

Différents auteurs, comme [Li et al., 2005] et [Terabayashi et al., 2009], ont montré que les propriétés de la géométrie épipolaire pour des caméras sténopées s'appliquent aussi aux caméras sphériques. Ainsi, la propriété $l' = Es$ peut être utilisée, où E est la matrice essentielle, s un rayon normé de la caméra 1 et l' une droite épipolaire de la caméra 2.

Une droite épipolaire l' dans un repère caméra se calcule à partir d'un point s de l'autre caméra par la relation $l' = Es$. L'équation de la droite épipolaire est calculée dans un espace restreint au plan Π (voir figure 3.2). La courbe épipolaire sphérique s'obtient en projetant l' sur la sphère unitaire en direction de son centre optique. Cette projection forme un cercle illustré figure 3.8.

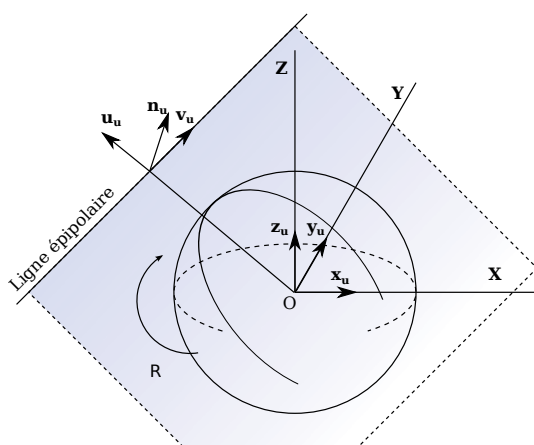


FIGURE 3.8 – Un cercle épipolaire est la projection d'une ligne épipolaire sur la sphère unitaire. Il appartient au plan défini par la ligne épipolaire et le centre de la sphère.

Les courbes épipolaires des 9 points ayant donné la meilleure estimation dans le RANSAC après un calibrage de la scène Cap2/3039 sont dessinées en jaune dans la figure 3.7. Elles passent bien par les points correspondants, ces 9 paires de points sont dessinées en mauve. Les points verts sont les autres paires de points *inliers* trouvées avec le RANSAC.

3.3.2/ CALCUL DES ÉPIPÔLES ET TEST DE STABILITÉ

Calculer la position des épipôles de manière analytique est possible avec la matrice essentielle grâce aux propriétés de la géométrie épipolaire. L'une de ces propriétés est $Ee = 0$ et $E^T e' = 0$, où E est la matrice essentielle, e un épipôle de la caméra 1 et e' un épipôle de la caméra 2.

En écrivant $E \begin{pmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{pmatrix}$ et $e \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix}$, il est possible de calculer e_1 et e_2 :

$$e_2 = -e_3 \cdot K \text{ et } e_1 = -\frac{e_2 f_{12} + e_3 f_{13}}{f_{11}} \text{ avec } K = \frac{f_{11} f_{23} - f_{13} f_{21}}{f_{11} f_{22} - f_{12} f_{21}} \quad (3.14)$$

Pour que l'épipôle appartienne à la sphère unitaire, on a $e_1^2 + e_2^2 + e_3^2 = 1$ et donc e_3 :

$$e_3 = \pm \frac{1}{\sqrt{1 + K^2 + \frac{(f_{13} - f_{12} \cdot K)^2}{f_{11}^2}}} \quad (3.15)$$

On peut calculer de la même manière les épipôles de la deuxième caméra avec E^T et e' .

La solution obtenue donne un épipôle, le deuxième étant son opposé. Avec la connaissance des positions 3D des épipôles sur la sphère unitaire, leurs positions 2D dans l'image se calculent aisément en appliquant la contre-projection du modèle fisheye (voir section 3.1.1).

Il est intéressant de remarquer que les résultats de la méthode par calcul analytique sont instables. Par principe, toutes les courbes épipolaires convergent en deux points qui sont les épipôles. Dans les résultats il y a toujours une marge d'erreur, et les courbes se croisent en différents points dans les deux zones autour des épipôles. On exploite la propriété $Ee = 0$ pour estimer l'erreur de distance des croisements des courbes avec les véritables épipôles. Il s'agit de retenir la meilleure intersection, c'est-à-dire pour le plus faible Ee , en tant qu'épipôle parmi toutes les possibilités à partir des courbes obtenues avec l'ensemble des *inliers* du calibrage.

La figure 3.9 montre la distribution de l'estimation d'un épipôle par le calcul analytique et par l'analyse géométrique. Il s'agit du calibrage évalué dans la section 3.4.5 pour la paire simulée 40 en projection équisolide, avec l'utilisation du LO-RANSAC. La scène en question et les informations vérifiées sont exprimées dans la section 3.4.1.1. Le calibrage a été exécuté 1000 fois. À chaque itération les deux procédures de calcul des épipôles ont été exécutées.

On voit nettement que la méthode analytique est très instable. Notre méthode géométrique donne toujours un résultat assez proche de la vérité. L'imprécision des calculs analytiques peut être liée au codage informatique des nombres à virgule flottante ou à la géométrie projective dans un plan capteur Π mal adapté à des points issus d'une sphère, en particulier lorsqu'ils sont alignés le long d'un axe quasiment parallèle à Π comme c'est le cas de l'axe épipolaire avec les configurations étudiées.

Les épipôles sont dessinés en rouge dans la figure 3.7, calibrage de la scène Cap2/3039. Ce sont les points où se croisent les courbes épipolaires.

3.3.3/ CALCUL DES ROTATIONS INTERNES POUR UN ALIGNEMENT ÉPIPOLAIRE

Estimer les courbes épipolaires est nécessaire pour les algorithmes de mise en correspondance stéréo basés sur la contrainte épipolaire. En fisheye, une courbe épipolaire est un grand cercle de la sphère unitaire, passant par les épipôles et par un point de référence. Pour les déterminer, nous avons donc besoin des deux épipôles de chaque image (voir section 3.3.2).

Pour obtenir les courbes épipolaires conjuguées, il faut considérer les plans épipolaires comme

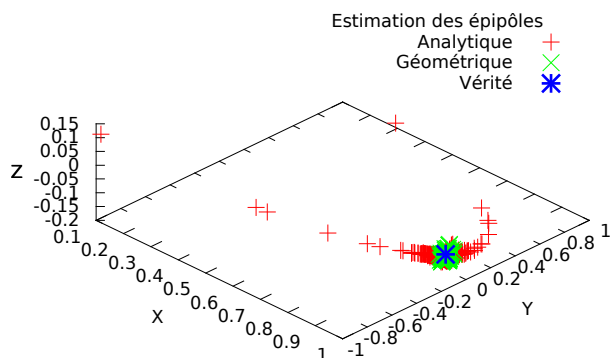


FIGURE 3.9 – Distribution des estimations de l'épipôle caméra arrière ayant $X > 0$ pour la paire simulée 40 en projection équivalente avec utilisation du LO-RANSAC.

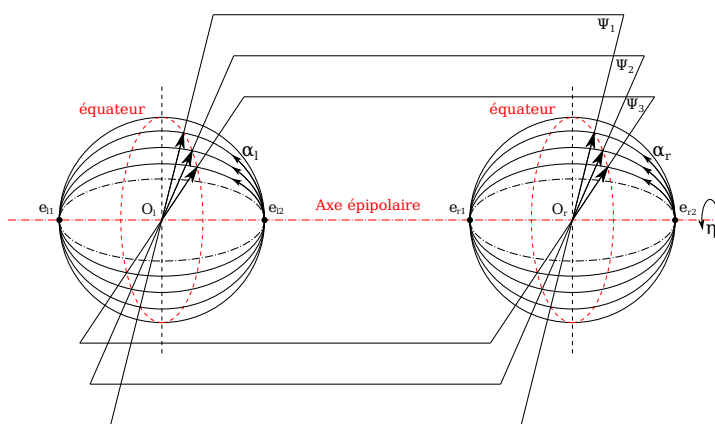


FIGURE 3.10 – Les courbes épipolaires à l'intérieur du même plan épipolaire (plans Ψ_i) peuvent être décrites par le même rayon ou le même angle η le long de l'arc de scan, défini comme l'équateur entre les deux épipôles.

décrit dans la figure 3.10. Un plan épipolaire passe par l'axe épipolaire et croise les sphères unitaires en formant de grands cercles : les cercles épipolaires. Un moyen immédiat pour détecter les cercles épipolaires conjugués est de choisir des rayons correspondants issus des deux sphères, à l'intérieur du même plan épipolaire. Ces rayons peuvent être obtenus en parcourant avec le même angle η les deux cercles de scan que l'on peut définir comme l'équateur entre les deux épipôles. L'arc de scan, portion utile du cercle de scan, est projeté en une courbe sur le plan capteur et dans l'image.

La méthode permettant d'obtenir le cercle de scan nécessite de travailler dans les repères épipolaires, tels que décrits dans les figures 3.11 et 3.12. L'important est de bien distinguer le repère épipolaire du repère caméra. Le repère caméra est le repère 3D obtenu à partir de la projection fisheye de l'image enregistrée et il est lié à l'orientation du capteur (caméra + objectif). La modélisation de la lentille et le repère caméra sont représentés en rouge. Le repère épipolaire est le repère adapté à la géométrie épipolaire, c'est-à-dire que les repères

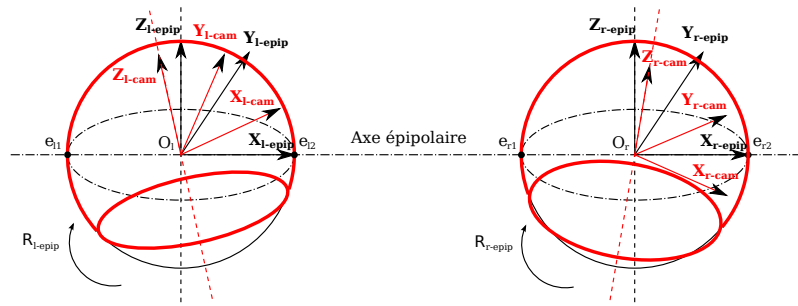


FIGURE 3.11 – Distinction entre les repères caméras et épipolaires. Les repères épipolaires sont obtenus par des rotations des repères caméras, qui sont définies par les axes des repères épipolaires.

épipolaires des deux caméras doivent avoir les mêmes bases avec une translation les séparant le long de l'axe \mathbf{X}_{epip} . La sphère du modèle sphérique et le repère épipolaire sont tracés en noir.

Comme on le voit dans la figure 3.12, déterminer ce repère nécessite la connaissance de l'axe \mathbf{X}_{epip} , donné par l'axe épipolaire (grâce à la position d'un épipôle sur la sphère), et par le calcul de l'axe \mathbf{Z}_{epip} ou de l'axe \mathbf{Y}_{epip} dans le repère caméra (la connaissance de deux axes étant suffisante pour calculer le troisième dans le repère orthonormal). Nous proposons la méthode suivante pour obtenir des repères épipolaires alignés à partir des informations contenues dans les images et de la matrice essentielle E :

1. Considérer le centre de la sphère gauche O_l et un point de l'arc de scan gauche (orthogonal à l'axe $\mathbf{X}_{l\text{-epip}}$) pour définir l'axe $\mathbf{Z}_{l\text{-epip}}$. Dans les configurations orientées en direction du ciel on choisit le point de l'équateur qui se projette dans l'image au plus près de son centre pour obtenir des axes \mathbf{Z}_{epip} proches du zénith.
2. Calculer la projection \mathbf{z} de $\mathbf{Z}_{l\text{-epip}}$ dans le plan capteur Π_l , et l'exploiter avec la matrice essentielle E via la relation $\mathbf{l}' = E\mathbf{z}$ pour estimer la droite épipolaire associée \mathbf{l}' dans Π_r .
3. Calculer l'arc projeté de \mathbf{l}' sur la sphère unitaire droite.
4. Calculer l'arc de scan droite sur la sphère unitaire droite.
5. L'intersection de ces arcs est le point de la sphère qui appartient à l'arc de scan et au même plan épipolaire que le point aboutissant à $\mathbf{Z}_{l\text{-epip}}$ pour le capteur gauche. Cela signifie que le vecteur défini par O_r et ce point d'intersection est $\mathbf{Z}_{r\text{-epip}}$ et qu'il est parallèle à $\mathbf{Z}_{l\text{-epip}}$.
6. $\mathbf{X}_{l\text{-epip}}$ et $\mathbf{X}_{r\text{-epip}}$ étant identiques et connus grâce aux épipôles, les définitions de $\mathbf{Z}_{l\text{-epip}}$ et $\mathbf{Z}_{r\text{-epip}}$ permettent de calculer les deux axes \mathbf{Y}_{epip} par $\mathbf{Y}_{\text{epip}} = \mathbf{Z}_{\text{epip}} \wedge \mathbf{X}_{\text{epip}}$.

Les rotations $R_{\text{epip}} = (\mathbf{X}_{\text{epip}} \mathbf{Y}_{\text{epip}} \mathbf{Z}_{\text{epip}})^T$ définies par les axes des repères épipolaires et visibles en figure 3.11 permettent pour chaque capteur de passer du repère caméra au repère épipolaire.

Le cercle de scan est le grand cercle sur la sphère défini dans le plan $(O, \mathbf{Y}_{\text{epip}}, \mathbf{Z}_{\text{epip}})$ et comme l'équateur entre les épipôles. Le calcul des courbes épipolaires est effectué en utilisant

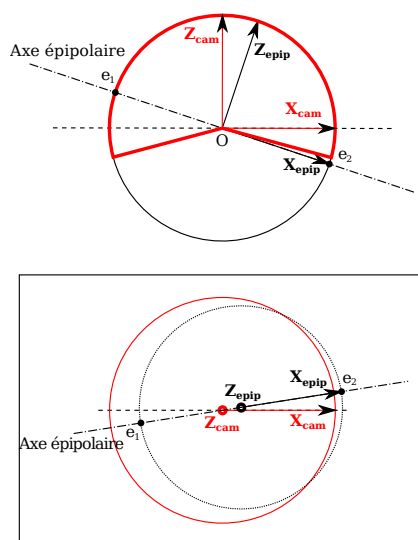


FIGURE 3.12 – Rotation du repère caméra vers le repère épipolaire. En haut : vue de coupe de la sphère le long de l'axe épipolaire. En bas : vue du plan image. \mathbf{X}_{epip} représente l'axe épipolaire. Dans le cas d'un objectif dont l'angle de vision est supérieur à 180° , le diamètre du disque image est plus grand que la distance entre les deux épipôles. L'axe épipolaire passe toujours par le centre de la sphère unitaire.

le rayon donnant le point de référence à l'intérieur du cercle de scan et les positions des deux épipôles. Le point de référence détermine la rotation d'un grand arc allant d'un épipôle à l'autre. Cet arc est un arc épipolaire, et se projette dans l'image en une courbe épipolaire. En utilisant le même point de référence (c'est-à-dire avec le même angle de rotation) dans les deux images selon les repères épipolaires, on obtient les courbes épipolaires conjuguées, appartenant au même plan épipolaire (voir figure 3.10).

Les équations des courbes épipolaires conjuguées peuvent donc être déterminées dans les repères épipolaires à partir du même angle de référence η dans l'arc de scan. η définit la rotation à appliquer au cercle unitaire centré en O du plan $(\mathbf{X}_{\text{epip}} \mathbf{Y}_{\text{epip}})$ pour obtenir leur trajet.

3.4/ ÉVALUATION DU CALIBRAGE AUTOMATIQUE

Des expérimentations pour des images simulées et des images réelles sont présentées. Comme le fait [Mičušík, 2004] on mesure la stabilité du calibrage aussi bien avec les images réelles qu'avec les images simulées. Pour cela le calibrage est effectué 1000 fois pour chaque couple d'images présenté. Les images simulées permettent en plus de mesurer la précision du calibrage. L'évaluation des images réelles montre la pertinence de la méthode dans des cas pratiques.

3.4.1/ SCÈNES ÉVALUÉES

On se place en configuration binoculaire, la translation et la rotation entre caméras sont donc fixes. Les séquences d'une même série d'acquisitions présentent théoriquement les mêmes

paramètres. Nous proposons une évaluation sur des images réelles et de synthèse.

3.4.1.1/ SCÈNES SIMULÉES

Il s'agit d'images de synthèses générées avec le logiciel de modélisation 3D blender. Les caméras fisheye ont un champ de vision de $181,8^\circ$. On considère une valeur initiale de 180° pour l'algorithme des 9 points. Elles sont orientées en direction du zénith et alignées selon l'axe \mathbf{X} . Une forte rotation supplémentaire appliquée à la caméra arrière d'angles $R_{\mathbf{X},\text{vérité}} = -8^\circ$, $R_{\mathbf{Y},\text{vérité}} = 6^\circ$ et $R_{\mathbf{Z},\text{vérité}} = -7^\circ$. L'entraxe utilisé pour les expérimentations est de 2 mètres. Pour approcher au mieux la réalité, des distorsions ou caractéristiques rencontrées dans la réalité sont reproduites dans les images de synthèse photoréalistes : textures des bâtiments issues de photos de murs réels, source lumineuse imitant le soleil, ciel réaliste, flou de mouvement en fonction du déplacement de la caméra, franges violettes aux séparations entre bâtiments et ciel, atténuation et contour bleu au pourtour du cercle de l'image, flare et réflexions internes du soleil dans les lentilles de l'objectif, léger flou de dispersion optique.

On propose deux variantes de séries simulées pour la même scène. La première respecte la fonction de projection à angle équisolide, les disques images ont un diamètre de 1124 pixels. La seconde suit la projection équidistante, les disques images ont un diamètre de 1200 pixels.

Les images simulées sont associées à une vérité terrain connue et précieuse pour évaluer la précision du calibrage. On est capable d'évaluer l'angle de vision de l'objectif et l'estimation des rotations. Deux paires d'images sont présentées dans la figure 3.13 :

- Paire 40 sans difficulté particulière, présente un mouvement de translation simple.
- Paire 24 qui présente une translation plus une rotation avec un flou de mouvement très fort et peu de bâtiments car positionnée dans un carrefour.

Compte-tenu des informations énoncées ci-dessus, on peut déterminer les paramètres du calibrage vérité. E_{simu} décrit la transformation épipolaire de gauche (ou arrière) à droite (ou avant) dans les repères caméras. On a donné les caractéristiques internes et de pose des caméras simulées. Les rotations sont celles de la caméra arrière (la caméra avant est alignée avec le repère de coordonnées de la scène et l'axe épipolaire), il faut les inverser pour la construction de E_{simu} . On connaît la translation de la caméra arrière à la caméra avant, qui n'est définie que selon l'axe \mathbf{X} . On aboutit à une matrice E_{simu} suivante :

$$E_{\text{simu}} = \begin{pmatrix} \tilde{0} & \tilde{0} & \tilde{0} \\ -0,10452846 & -0,13841070 & -0,98484327 \\ 0,12120173 & 0,98111386 & -0,15075057 \end{pmatrix} \quad (3.16)$$

Hormis le cas particulier de comparaison des modèles de projection on travaille avec les images simulées en projection équisolide et on connaît le champ de vision. Par conséquent, d'après

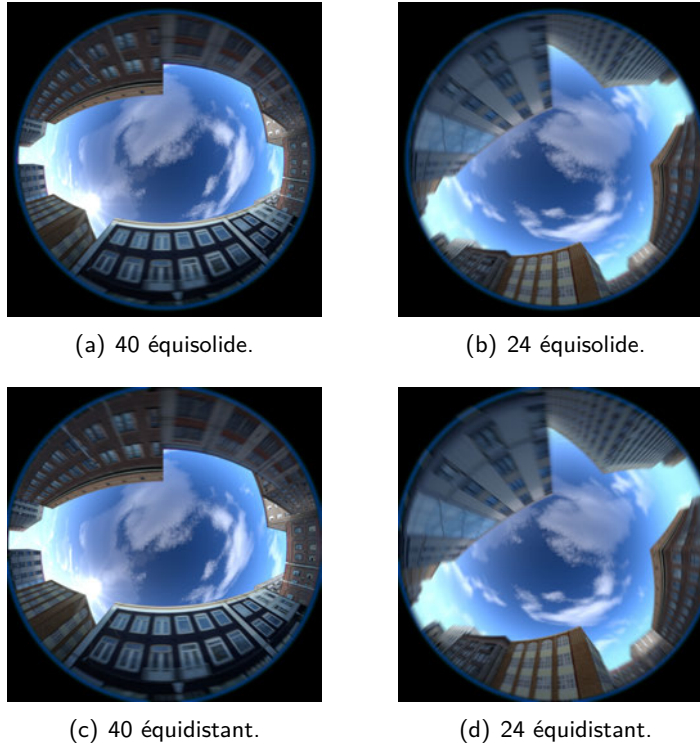


FIGURE 3.13 – Images simulées présentées.

l'équation correspondante dans le tableau 3.3, le paramètre de la fonction de projection vaut :

$$a_{\text{simu},\text{équisolide}} = \sin\left(\frac{181,8^\circ}{4}\right) = 0,71263852 \quad (3.17)$$

Avec les simulations on a $A_{\text{simu}} = \text{Id}_2$ et $t_{\text{simu}} = \begin{pmatrix} -\frac{1}{2}\text{largeur}_{px} \\ -\frac{1}{2}\text{hauteur}_{px} \end{pmatrix} = \begin{pmatrix} -600 \\ -600 \end{pmatrix}$ pour les deux caméras. Le diamètre du disque image vérité avec la projection à angle équisolide est connu et est de 1124 pixels. Dans le repère caméra gauche, les épipôles de l'image arrière sont donnés par $e_{\text{simu},\text{AR}}$; dans le repère caméra droite, les épipôles de l'image avant sont donnés par $e_{\text{simu},\text{AV}}$:

$$e_{\text{simu},\text{AR}} = \pm \begin{pmatrix} -0,98710888 \\ 0,13512243 \\ 0,08577871 \end{pmatrix}, e_{\text{simu},\text{AV}} = \pm \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad (3.18)$$

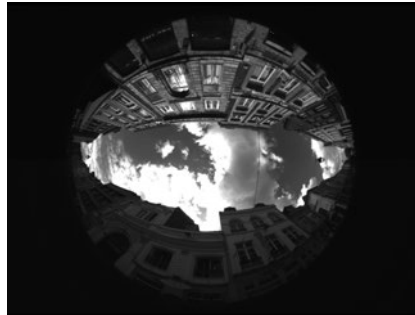
3.4.1.2/ SCÈNES RÉELLES

Pour valider les travaux présentés on propose d'évaluer le calibrage d'images réelles acquises sur un véhicule en mouvement. Ces scènes réelles ont été acquises dans la campagne d'acquisitions de Lille le 19 août 2014 en configuration stéréovision binoculaire. Les caméras sont disposées de manière identique pour les deux paires d'images présentées. Elles ont été enregistrées avec le système d'acquisition de la configuration (Cf g 1).

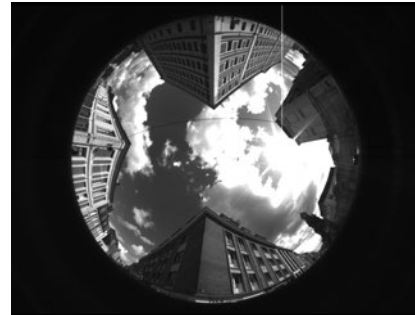
Deux couples d'images avec un contenu différent équivalent aux caractéristiques des images

simulées sont choisis et montrés dans la figure 3.14 :

- Paire Cap2/3039 (conditions standard), mouvement en ligne droite rue de la Monnaie.
- Paire Cap3/2865 (conditions difficiles), translation et rotation dans le virage de la rue du Magasin à la rue Royale, et présence du soleil qui provoque une traînée verticale saturée (*vertical smear*, défaut possible avec les capteurs CCD en présence d'une forte source lumineuse).



(a) Cap2/3039



(b) Cap3/2865

FIGURE 3.14 – Images réelles présentées.

Pour l'évaluation de l'utilisation de différents modèles de projection, en section 3.4.4, on propose en outre une même scène capturée avec deux systèmes différents : l'un avec un objectif en projection à angle équisolide, et l'autre à projection équidistante. Elles ont été acquises au Techn'hom de Belfort le 6 mars 2013. Les deux systèmes correspondent aux configurations (Cfg 2) et (Cfg 3). Les images sont montrées dans la figure 3.15.

3.4.2/ CRITÈRES ÉVALUÉS

Pour les scènes simulées, nous montrons des mesures d'erreur par rapport à la vérité. Le RANSAC est un algorithme probabiliste, il ne donne pas le même résultat d'une exécution à une autre. Dans tous les cas (simulés et réels), on présente aussi des mesures de stabilité. On recherche la stabilité du RANSAC pour avoir une estimation prévisible, c'est-à-dire dans un intervalle de valeurs aussi restreint que possible. La qualité du calibrage ne dépend pas tout à fait de la stabilité mais une bonne stabilité assure un calibrage de qualité constante. Et de manière générale, on considère l'intervalle de valeurs du calibrage autour de la vérité.

On peut espérer qu'un intervalle plus fin implique une estimation plus proche de la vérité. C'est en tout cas ce que l'on observe avec les images simulées disposant des vérités. En conséquence, pour chaque paire d'images, l'algorithme est exécuté 1000 fois pour montrer des mesures statistiques.

Les mesures suivantes sont retranscrites dans les tableaux récapitulatifs :

Critères : (l'axe **W** peut être **X**, **Y** ou **Z**)

FoV : Champ de vision estimé (à partir de la relation entre θ et a , tableau 3.3)

|Err FoV| : Erreur absolue du champ de vision estimé dans le cas simulé



(a) Image issue d'une caméra Pixelink avec objectif Fujinon.



(b) Image issue de l'appareil photo reflex Fujifilm avec l'objectif Sigma.

FIGURE 3.15 – Images réelles de la même scène avec deux configurations à projection diérente.

- $R_{\mathbf{W},av \rightarrow ar}$: Rotation estimée du repère caméra avant au repère caméra arrière selon \mathbf{W}
- $|\text{Err } R_{\mathbf{W},av \rightarrow ar}|$: Erreur absolue de l'angle de rotation estimé selon \mathbf{W} dans le cas simulé
- Nb inliers : Nombre d'*inliers* obtenus
- % inliers : Pourcentage d'*inliers* obtenus
- Nb itérations : Nombre d'itérations RANSAC

Données pour chaque critère :

Moy : Moyenne du critère

É-type : Écart-type du critère

3.4.3/ SEUIL D'ERREUR ANGULAIRE

Comme dit section 3.2.4.1, il faut bien choisir le seuil d'erreur $\epsilon_{seuil} = p \cdot \frac{\tilde{\theta}_{max}}{r_{max}}$ de décision si un couple de points est *inlier* ou *outlier*. Cette partie présente les résultats pour $p \in \{10^{-2}; 10^{-3}; 10^{-4}; 10^{-5}\}$, afin de trouver un réglage convenable en toutes circonstances.

Pour cette évaluation on garde la méthode générique sans modification. Le RANSAC classique est utilisé et le réglage de la fonction de projection est choisie pour coïncider avec la projection spécifiée par le constructeur.

Les essais faits pour un seuil de $10^{-5} \frac{\tilde{\theta}_{max}}{r_{max}}$ arrivent presque toujours au bout des 50000 itérations maximales fixées pour toutes les séquences étudiées. Il a été montré en section 3.2.4.1 que le RANSAC devrait converger autour de 38008 itérations. Ce seuil est trop faible. À la vue des résultats présentés dans les tableaux 3.4, 3.5, 3.6, 3.7, 3.8 et 3.9, on arrive aux conclusions suivantes. Plus le seuil d'erreur est petit plus on arrive à une estimation précise et stable. Cependant dans un cas les résultats avec seuil faible sont beaucoup moins stables : la paire 24 des images simulées en projection équisolide. Les diagrammes de distribution des erreurs 3.16 semblent indiquer que ce sont les rotations, en particulier selon \mathbf{Y} et \mathbf{Z} , qui sont moins bonnes et qui dégénèrent lorsque le seuil est trop faible. La paire 24 est un cas extrême de flou de l'image par la rotation, et elle est beaucoup plus floue que les images

Seuil d'erreur (°)	$0,01 \times \frac{\theta_{max}}{r_{max}}$		$10^{-3} \times \frac{\theta_{max}}{r_{max}}$		$10^{-4} \times \frac{\theta_{max}}{r_{max}}$	
	Moy	E-type	Moy	E-type	Moy	E-type
FoV (°)	181,68	8,34	181,92	4,31	182,79	2,05
Err FoV (°)	6,63	5,06	3,45	2,58	1,83	1,37
$R_{X,av \rightarrow ar}$ (°)	-7,88	1,37	-8,03	0,51	-8,05	0,17
Err $R_{X,av \rightarrow ar}$ (°)	0,97	0,98	0,35	0,37	0,14	0,12
$R_{Y,av \rightarrow ar}$ (°)	4,92	0,74	4,90	0,33	4,97	0,14
Err $R_{Y,av \rightarrow ar}$ (°)	1,14	0,65	1,10	0,33	1,03	0,14
$R_{Z,av \rightarrow ar}$ (°)	-7,69	0,34	-7,69	0,11	-7,71	0,04
Err $R_{Z,av \rightarrow ar}$ (°)	0,70	0,32	0,69	0,11	0,71	0,04
Nb inliers	176,11	19,50	141,35	14,11	93,96	5,95
% inliers	56,63	6,27	45,45	4,54	30,21	1,91
Nb itérations	27,33	18,31	167,98	110,40	4698,03	2213,80

TABLEAU 3.4 – Résultat des tests du seuil d'erreur angulaire paire simulée 40 en projection équisolide.

Seuil d'erreur (°)	$0,01 \times \frac{\theta_{max}}{r_{max}}$		$10^{-3} \times \frac{\theta_{max}}{r_{max}}$		$10^{-4} \times \frac{\theta_{max}}{r_{max}}$	
	Moy	E-type	Moy	E-type	Moy	E-type
FoV (°)	183,06	8,78	183,12	4,01	182,60	1,94
Err FoV (°)	6,67	5,85	3,39	2,51	1,70	1,23
$R_{X,av \rightarrow ar}$ (°)	-8,01	1,46	-8,17	0,53	-8,19	0,20
Err $R_{X,av \rightarrow ar}$ (°)	1,05	1,02	0,43	0,35	0,24	0,15
$R_{Y,av \rightarrow ar}$ (°)	5,15	0,65	5,14	0,25	5,11	0,13
Err $R_{Y,av \rightarrow ar}$ (°)	0,88	0,61	0,86	0,25	0,89	0,13
$R_{Z,av \rightarrow ar}$ (°)	-7,59	0,26	-7,60	0,11	-7,61	0,05
Err $R_{Z,av \rightarrow ar}$ (°)	0,60	0,24	0,60	0,11	0,61	0,05
Nb inliers	203,50	22,78	158,63	15,54	109,35	6,90
% inliers	56,84	6,36	44,31	4,34	30,54	1,93
Nb itérations	29,26	20,26	197,33	117,72	4052,43	1835,18

TABLEAU 3.5 – Résultat des tests du seuil d'erreur angulaire paire simulée 40 en projection équidistante.

Seuil d'erreur (°)	$0,01 \times \frac{\theta_{max}}{r_{max}}$		$10^{-3} \times \frac{\theta_{max}}{r_{max}}$		$10^{-4} \times \frac{\theta_{max}}{r_{max}}$	
	Moy	E-type	Moy	E-type	Moy	E-type
FoV (°)	173,03	22,10	173,76	12,02	178,45	6,38
Err FoV (°)	17,12	16,48	10,20	10,25	5,53	4,62
$R_{X,av \rightarrow ar}$ (°)	-7,63	1,50	-7,56	0,85	-7,28	5,11
Err $R_{X,av \rightarrow ar}$ (°)	1,20	0,97	0,70	0,65	1,01	5,06
$R_{Y,av \rightarrow ar}$ (°)	4,13	1,94	4,20	1,15	5,16	3,54
Err $R_{Y,av \rightarrow ar}$ (°)	2,07	1,72	1,82	1,12	1,79	3,16
$R_{Z,av \rightarrow ar}$ (°)	-7,31	1,48	-7,44	0,63	-7,13	3,77
Err $R_{Z,av \rightarrow ar}$ (°)	1,05	1,10	0,62	0,45	1,45	3,49
Nb inliers	49,91	4,42	35,94	2,41	23,90	1,79
% inliers	46,21	4,10	33,27	2,23	22,11	1,66
Nb itérations	173,32	102,60	3340,3	1576,14	48822,49	4345,04

TABLEAU 3.6 – Résultat des tests du seuil d'erreur angulaire paire simulée 24 en projection équisolide.

réelles acquises. Il est très difficile d'y détecter des points correspondants de manière fiable. Le nombre d'itérations pour convergence du RANSAC est toujours plus important avec la scène 24, à tel point que l'on est souvent limité par les 50000 itérations maximum. Le RANSAC n'arrive pas à converger avec cette acquisition pour un seuil strict. Pour finir, on peut noter

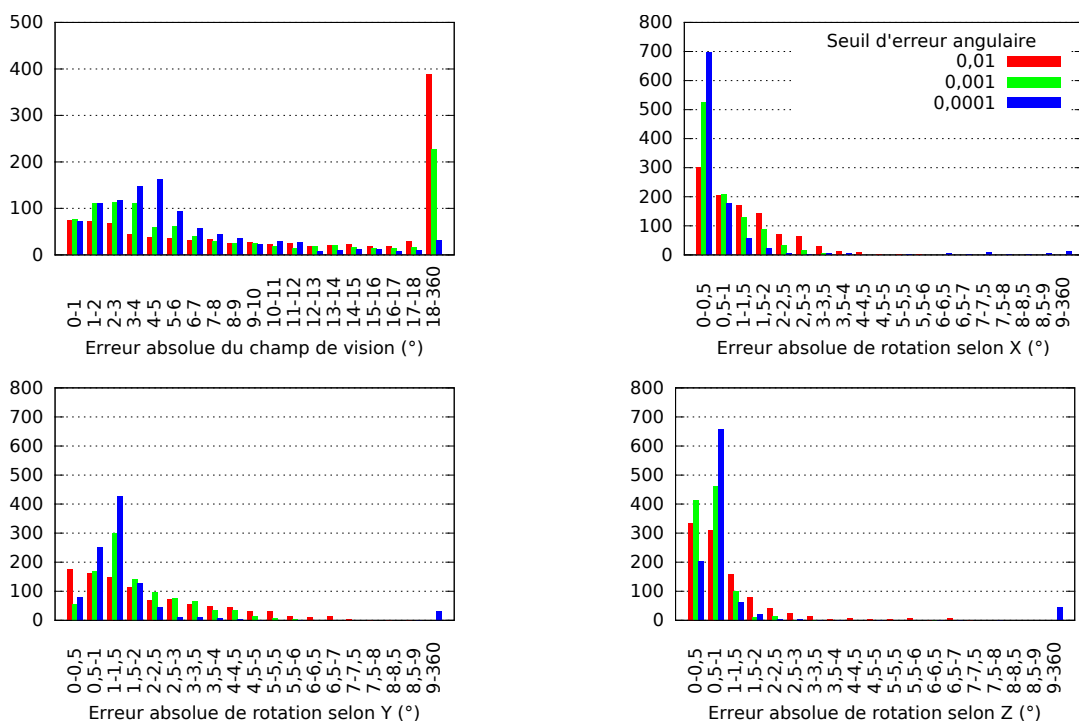


FIGURE 3.16 – Diagrammes de distribution des erreurs d'estimation du champ de vision et de rotation paire simulée 24 en projection équisolide.

Seuil d'erreur (°)	$0,01 \times \frac{\theta_{max}}{r_{max}}$		$10^{-3} \times \frac{\theta_{max}}{r_{max}}$		$10^{-4} \times \frac{\theta_{max}}{r_{max}}$	
	Moy	E-type	Moy	E-type	Moy	E-type
FoV (°)	178,11	16,38	173,83	13,10	175,34	7,89
Err FoV (°)	12,30	11,43	11,84	9,73	8,06	6,24
$R_{X,av \rightarrow ar}$ (°)	-8,14	1,87	-7,69	1,26	-7,51	0,81
Err $R_{X,av \rightarrow ar}$ (°)	1,44	1,19	1,01	0,80	0,76	0,56
$R_{Y,av \rightarrow ar}$ (°)	5,02	1,56	4,44	1,29	4,53	0,98
Err $R_{Y,av \rightarrow ar}$ (°)	1,36	1,25	1,67	1,13	1,58	0,81
$R_{Z,av \rightarrow ar}$ (°)	-7,90	1,53	-7,68	0,89	-7,73	0,74
Err $R_{Z,av \rightarrow ar}$ (°)	1,22	1,29	0,76	0,82	0,78	0,69
Nb inliers	56,86	4,74	42,04	3,01	28,18	1,94
% inliers	44,43	3,71	32,85	2,35	22,02	1,52
Nb itérations	197,57	105,81	3398,16	1648,18	48908,58	4536,27

TABLEAU 3.7 – Résultat des tests du seuil d'erreur angulaire paire simulée 24 en projection équidistante.

via les diagrammes 3.17 que la paire 24 en projection équidistante se comporte mieux et donne tout de même les meilleurs résultats avec le seuil le plus faible.

On conserve pour la suite de ces études le réglage du seuil d'erreur à $10^{-4} \frac{\theta_{max}}{r_{max}}$.

3.4.4/ CHOIX DU BON MODÈLE DE PROJECTION FISHEYE

Le RANSAC classique est utilisé, le seuil d'erreur est déterminé à l'issue de la section 3.4.3.

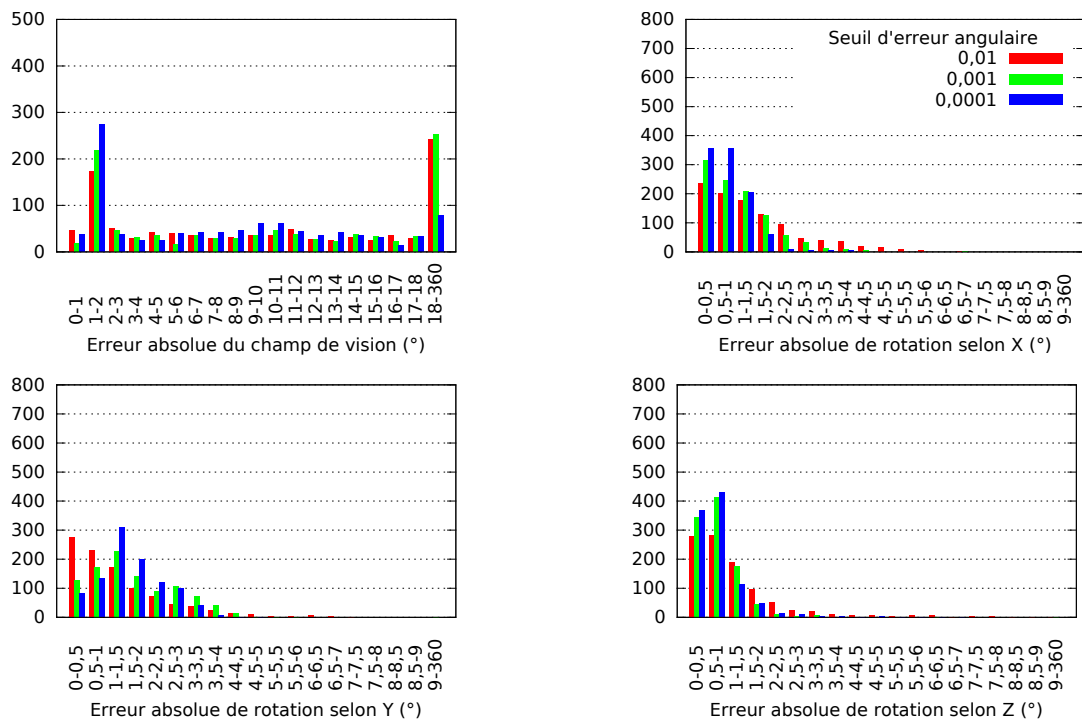


FIGURE 3.17 – Diagrammes de distribution des erreurs d’estimation du champ de vision et de rotation paire simulée 24 en projection équidistante.

Seuil d’erreur (°)	$0,01 \times \frac{\theta_{max}}{r_{max}}$		$10^{-3} \times \frac{\theta_{max}}{r_{max}}$		$10^{-4} \times \frac{\theta_{max}}{r_{max}}$	
	Moy	E-type	Moy	E-type	Moy	E-type
FoV (°)	184,69	4,67	184,59	2,29	184,45	1,2
$R_{X,av \rightarrow ar}$ (°)	1,26	2,36	1,17	0,23	1,24	0,13
$R_{Y,av \rightarrow ar}$ (°)	0,17	1,18	0,02	0,29	-0,03	0,17
$R_{Z,av \rightarrow ar}$ (°)	0,06	1,15	-0,23	0,22	-0,31	0,11
Nb inliers	794,39	76,38	638,94	58,89	423,95	24,83
% inliers	57,61	5,54	46,33	4,27	30,74	1,80
Nb itérations	19,18	12,49	109,48	61,93	2993,56	1256,86

TABLEAU 3.8 – Résultat des tests du seuil d’erreur angulaire paire réelle Cap2/3039 (projection équisolide).

Seuil d’erreur (°)	$0,01 \times \frac{\theta_{max}}{r_{max}}$		$10^{-3} \times \frac{\theta_{max}}{r_{max}}$		$10^{-4} \times \frac{\theta_{max}}{r_{max}}$	
	Moy	E-type	Moy	E-type	Moy	E-type
FoV (°)	182,43	10,91	183,25	5,49	184,92	3,18
$R_{X,av \rightarrow ar}$ (°)	1,10	3,47	0,93	0,29	0,95	0,19
$R_{Y,av \rightarrow ar}$ (°)	0,01	0,63	-0,02	0,29	-0,03	0,14
$R_{Z,av \rightarrow ar}$ (°)	-1,09	0,78	-1,05	0,26	-1,09	0,12
Nb inliers	1036,81	123,37	782,49	62,38	480,47	21,43
% inliers	52,93	6,30	39,94	3,18	24,53	1,09
Nb itérations	50,80	34,45	362,48	200,33	20249,09	6771,98

TABLEAU 3.9 – Résultat des tests du seuil d’erreur angulaire paire réelle Cap3/2865 (projection équisolide).

Pour valider le choix du modèle de projection il est inutile de tester les scènes simulées car elles correspondent exactement aux équations des modèles décrits dans la section 3.2.2 et

il est évident que le choix adapté fonctionnera mieux. Par contre il peut être intéressant de vérifier s'il y a un gain significatif ou non en utilisant correctement les fonctions de projection pour des objectifs réels.

Objectif utilisé Projection appliquée	Fujinon (Proj. équidistante) $\widetilde{\text{FoV}} = 185^\circ$ (données constructeur)				Sigma (Proj. équisolide) $\widetilde{\text{FoV}} = 184,6^\circ$ (calibrage fort)			
	Équidistante		Équisolide		Équidistante		Équisolide	
	Moy	É-type	Moy	É-type	Moy	É-type	Moy	É-type
FoV ($^\circ$)	183,84	11,15	179,98	10,31	182,57	2,11	185,72	2,21
$R_{X,av \rightarrow ar}$ ($^\circ$)	-0,11	0,39	-0,16	0,40	0,10	0,24	0,08	0,19
$R_{Y,av \rightarrow ar}$ ($^\circ$)	0,22	0,25	0,39	0,20	0,27	0,05	0,31	0,04
$R_{Z,av \rightarrow ar}$ ($^\circ$)	-0,20	0,13	-0,23	0,12	-2,98	0,04	-2,99	0,04
Nb inliers	103,11	4,52	99,98	4,78	572,14	50,77	579,11	55,01
% inliers	33,37	1,46	32,36	1,55	48,28	4,28	48,87	4,64
Nb itérations	1521,38	525,28	2115,88	749,80	73,88	43,41	73,24	43,09

TABLEAU 3.10 – Résultat des tests du choix de la fonction de projection en fonction de l'objectif utilisé pour des paires réelles acquises au même endroit (Techn'hom de Belfort).

Dans le tableau 3.10 les résultats en stabilité sont proches pour les deux modèles de projections comparés. Néanmoins, le nombre d'itérations est plus faible dans le cas du choix du modèle correspondant à l'objectif. Et parallèlement, le nombre d'*inliers* est plus grand avec le bon modèle de projection. Ceci signifie qu'il est plus difficile de converger lorsque le modèle choisi n'est pas le bon, la recherche d'*inliers* est moins efficace car le modèle est moins approprié. On voit très bien l'intérêt de la bonne fonction de projection dans le cas de l'objectif Fujinon. De plus, il est donné pour couvrir un champ de vision de 185° , et l'estimation moyenne avec le bon modèle de projection est effectivement bien plus proche de cette valeur.

En revanche les résultats obtenus avec l'objectif Sigma ne semblent pas confirmer cette thèse. Ils sont quasiment identiques avec les deux modèles de projection mis en concurrence. La seule différence notable est la valeur moyenne estimée du champ de vision, qui se trouve être plus proche des données constructeur (180°) dans le cas de la fonction de projection non adaptée ! Avec un tel résultat il est permis de se demander si les spécifications du constructeur concernant le champ de vision ou le modèle de projection sont correctes. Devant l'impossibilité d'obtenir des données précises de la part du fabricant, nous avons calibré l'objectif Sigma avec l'outil de calibrage de [Scaramuzza, 2008]. Compte tenu d'une mesure manuelle du rayon de l'image on calcule un champ de vision de $184,6^\circ$ au lieu de 180° . Finalement le choix du modèle de projection nominal est bien le plus pertinent.

Se tromper de modèle de projection peut fonctionner mais est moins précis et moins efficace.

3.4.5/ EFFICACITÉ ET STABILITÉ ENTRE RANSAC ET LO-RANSAC

Les méthodes RANSAC et sa variante LO-RANSAC sont comparés, avec le seuil d'erreur déterminé à l'issue de la section 3.4.3. L'intérêt est de valider et d'étendre les résultats que [Chum, 2005] obtient pour l'algorithme des 8 points avec l'algorithme des 9 points dans le

contexte fisheye. Le réglage de la fonction de projection est choisie pour coïncider avec la projection spécifiée par le constructeur ou la simulation (travail en projection équisolide).

Extrait Variante RANSAC	Paire 40				Paire 24			
	RANSAC		LO-RANSAC		RANSAC		LO-RANSAC	
	Moy	É-type	Moy	É-type	Moy	É-type	Moy	É-type
FoV (°)	182,79	2,05	182,75	1,48	178,45	6,38	177,82	5,98
Err FoV (°)	1,83	1,37	1,46	0,98	5,53	4,62	5,55	4,55
$R_{X,av \rightarrow ar}$ (°)	-8,05	0,17	-8,05	0,10	-7,28	5,11	-6,77	7,49
Err $R_{X,av \rightarrow ar}$ (°)	0,14	0,12	0,08	0,08	1,01	5,06	1,46	7,45
$R_{Y,av \rightarrow ar}$ (°)	4,97	0,14	4,96	0,09	5,16	3,54	5,11	4,24
Err $R_{Y,av \rightarrow ar}$ (°)	1,03	0,14	1,04	0,09	1,79	3,16	2,11	3,78
$R_{Z,av \rightarrow ar}$ (°)	-7,71	0,04	-7,70	0,03	-7,13	3,77	-6,90	4,19
Err $R_{Z,av \rightarrow ar}$ (°)	0,71	0,04	0,70	0,03	1,45	3,49	1,61	3,87
Nb inliers	93,96	5,95	110,29	6,45	23,90	1,79	25,93	2,60
% inliers	30,21	1,91	35,46	2,07	22,11	1,66	24,01	2,41
Nb itérations	4698,03	2213,80	1226,68	888,47	48822,49	4345,04	42808,84	11471,17

TABLEAU 3.11 – Résultat des tests de variantes RANSAC pour les paires simulées.

Extrait Variante RANSAC	Paire Cap2/3039				Paire Cap3/2865			
	RANSAC		LO-RANSAC		RANSAC		LO-RANSAC	
	Moy	É-type	Moy	É-type	Moy	É-type	Moy	É-type
FoV (°)	184,45	1,25	184,75	0,58	184,92	3,18	185,52	2,51
$R_{X,av \rightarrow ar}$ (°)	1,24	0,13	1,23	0,07	0,95	0,19	0,87	0,51
$R_{Y,av \rightarrow ar}$ (°)	-0,03	0,17	0,03	0,09	-0,03	0,14	0,03	0,12
$R_{Z,av \rightarrow ar}$ (°)	-0,31	0,11	-0,29	0,06	-1,09	0,12	-1,08	0,12
Nb inliers	423,95	24,83	502,09	21,75	480,47	21,43	513,87	21,83
% inliers	30,74	1,80	36,41	1,58	24,53	1,09	26,23	1,11
Nb itérations	2993,56	1256,86	642,10	375,30	20249,09	6771,98	11461,60	5177,62

TABLEAU 3.12 – Résultat des tests de variantes RANSAC pour les paires réelles.

Les tableaux 3.11 et 3.12 montrent que le LO-RANSAC a l'avantage sur le RANSAC. Il présente plus d'*inliers*, converge en beaucoup moins d'itérations, la stabilité est meilleure et l'erreur plus faible pour tous les paramètres. La seule séquence où le gain n'est pas si net est la scène simulée 24, scène très difficile car extrêmement floue. Là les erreurs de rotation sont un peu plus grandes et moins stables. Ce cas étant extrême et plus difficile que les cas réels rencontrés, la dégradation du résultat observée n'est pas assez forte pour renoncer au LO-RANSAC, qui est sinon meilleur de manière significative en tous points.

3.5/ CONCLUSION

Ce chapitre est consacré au calibrage d'un système multi-vues utilisant une ou deux caméras à objectif fisheye. Les objectifs fisheye sont traditionnellement représentés par un modèle de projection sphérique associé à une fonction stéréographique, équidistante ou équisolide reliant l'angle d'incidence d'un point de la scène au rayon de sa projection sur le capteur de la caméra. Ici nous nous sommes limités à des fonctions à un seul paramètre.

Nous proposons une méthode de calibrage automatique fondée sur 9 paires de points extraits

de deux images fisheye pour déterminer les paramètres du modèle de projection et les paramètres qui définissent la géométrie épipolaire relative à ces deux vues. Nous avons adapté les travaux de [Mičušík, 2004] aux projections à angle équisolide et stéréographique. Nous avons choisi d'utiliser un LO-RANSAC et une étape d'affinement en appliquant l'algorithme Levenberg-Marquardt pour améliorer la précision et la rapidité de l'estimation. Cette méthode est totalement automatique ; elle ne nécessite aucune intervention humaine et permet de déterminer toutes les paires de courbes épipolaires conjuguées nécessaires lors de l'étape de mise en correspondance. Elle s'applique pour deux images provenant aussi bien d'un stéréoscope que d'une unique caméra en mouvement.

Le calibrage donne des résultats précis comme le montre l'évaluation réalisée sur des images de synthèse. Sur des images réelles, la méthode converge vers une estimation de manière stable. Le modèle pour lequel nous calibrons le système doit correspondre exactement à celui du matériel utilisé. Employer un modèle approché convergera mais plus lentement et la précision sera plus faible. Dans l'avenir nous envisageons donc les améliorations suivantes :

- Détecter une ellipse au lieu d'un cercle pour aboutir à une matrice A et estimer plus précisément le centre optique ;
- Étendre le calibrage à une matrice fondamentale 4×4 afin de tenir compte plus précisément des distorsions [Barreto et al., 2005] ;
- Augmenter le nombre de paramètres afin de prendre en compte les distorsions radiales et améliorer la généralité du modèle de projection [Hughes et al., 2010]. Cette amélioration nécessite plus de points appariés pour converger vers la solution. Utiliser les mêmes équations que l'algorithme présenté pour n paramètres nécessite $9 + 5(n - 1) + \frac{n(n-1)}{2}$ couples de points : il en faut 15 pour 2 paramètres, 22 pour 3 paramètres etc. La convergence itérative n'est donc plus garantie. Une solution que nous imaginons pour résoudre ce problème de convergence est d'utiliser le modèle à plusieurs paramètres dans une boucle interne du LO-RANSAC, avec un RANSAC père basé sur une estimation avec un modèle à paramètre unique.

MISE EN CORRESPONDANCE MULTI-VUES FISHEYE POUR ESTIMER UN NUAGE DE POINTS 3D

Le calibrage effectué, il est plus aisé d'apparier le contenu de deux vues d'une même scène en exploitant la contrainte épipolaire. Dans ce chapitre, nous proposons une méthode de mise en correspondance fondée sur l'usage d'un graphe 3D construit à partir de plusieurs couples de courbes épipolaires conjuguées. Nous en définissons le sous-espace optimal des paramètres dont nous évaluons quelques jeux sur des séquences d'images de synthèse et sur des séquences d'images réelles. Grâce à la mise en place d'un algorithme d'odométrie visuelle nous montrons qu'il est alors possible de parvenir à une fusion assez précise de ces nuages de points 3D locaux.

Sommaire

4.1	Introduction	74
4.2	Mise en correspondance stéréo fisheye	74
4.2.1	Indexation des courbes épipolaires	75
4.2.2	Calcul des positions 3D	76
4.2.3	L'algorithme de Fortsmann	77
4.2.4	De l'algorithme de Forstmann au graphe 3D	83
4.2.5	Évaluation sur des images réelles	87
4.2.6	Conclusion	90
4.3	Correspondance mono-caméra fisheye couleur	91
4.3.1	Adaptation de notre méthode graphe 3D	91
4.3.2	Évaluation de séquences d'images réelles	93
4.4	Analyse des nuages de points 3D	96
4.4.1	Post-traitement des nuages de points 3D	98
4.4.2	Fusion des nuages de points 3D	100
4.4.3	Cas mono-caméra	100
4.4.4	Cas binoculaire	102
4.4.5	Évaluation de l'odométrie visuelle	105
4.5	Conclusion	108
4.6	Annexe : Estimation de la largeur de la rue	109

4.1/ INTRODUCTION

Le calibrage permet de trouver les relations géométriques entre les caméras. Ces relations sont particulièrement utiles pour la mise en correspondance stéréo des points. La mise en correspondance est l'étape de comparaison des points d'une image à l'autre pour trouver les paires de pixels homologues. Une paire de pixels homologues entre les images permet de calculer la position 3D du point correspondant dans l'espace.

Ce chapitre détaille une analyse des images fisheye suivant les courbes épipolaires pour respecter la contrainte épipolaire. Une méthode de mise en correspondance par programmation dynamique avec graphe 3D est proposée et évaluée. Une étude du système est proposée pour des configurations binoculaires ou monoculaires, en niveaux de gris ou en couleurs.

4.2/ MISE EN CORRESPONDANCE STÉRÉO FISHEYE

Dans le chapitre précédent nous avons exposé comment estimer la géométrie épipolaire liée à deux vues d'une même scène, ces deux vues pouvant être acquises au même instant par un capteur stéréoscopique ou acquises à deux instants et positions différents par une seule caméra en mouvement. La connaissance de la géométrie épipolaire permet de définir les courbes épipolaires conjuguées dans les deux vues le long desquelles est appliquée la mise en correspondance. Dans le cas fisheye, on a montré dans la section 3.3 que les courbes épipolaires peuvent être représentées par des arcs de cercles sur la sphère unitaire, centrés sur le centre optique et passant par les épipôles. La plupart des travaux existants ([Shah et al., 1997, Li, 2008]) transforment les images sphériques en images rectifiées de telle sorte à obtenir des lignes épipolaires parfaitement horizontales (figure 4.1). Le but est d'appliquer un algorithme de mise en correspondance de la même manière que dans le cas sténopé. Toutefois, ces travaux imposent une étape d'interpolation qui vient modifier l'information contenue dans les images.

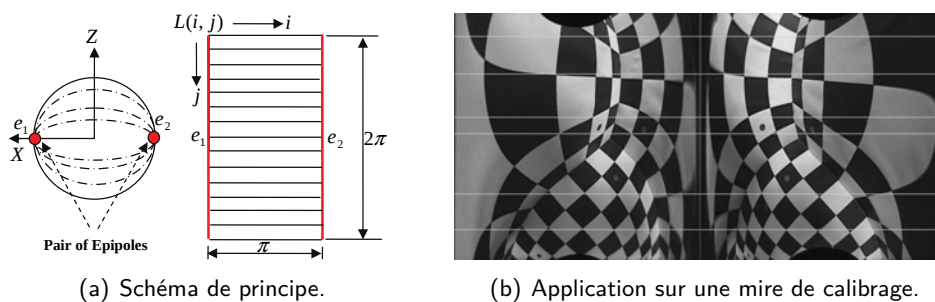


FIGURE 4.1 – Transformation épipolaire « latitude-longitude » proposée par [Li, 2006b, Li, 2008], dans le but de transformer les arcs épipolaires en lignes épipolaires horizontales.

Dans ce manuscrit, nous proposons une indexation à angle régulier proche de celle proposée par [Li, 2008] mais n'imposant aucune interpolation pour ne pas adoucir davantage l'information texture déjà bien altérée par la projection sphérique (section 4.2.1).

Parmi toutes les techniques de mise en correspondance évaluées par la méthode « Middlebury¹ » nous avons retenu celle proposée par [Forstmann et al., 2004]. Cette technique exploite un processus d'optimisation globale par programmation dynamique et estime une carte de disparité dense en propageant une contrainte de continuité entre les pixels voisins. Cet algorithme offre une bonne qualité de carte au regard de la complexité algorithmique et est portable sur une carte GPU [Kalarot et al., 2010].

La section 4.2.3 montre les adaptations que nous avons faites de l'algorithme de [Forstmann et al., 2004] afin de l'appliquer aux images fisheye. Nous présentons ensuite son extension à un graphe 3D (section 4.2.4) qui tient compte des courbes épipolaires voisines afin notamment de palier aux imprécisions d'estimation et d'échantillonnage des courbes épipolaires.

Des évaluations sont effectuées tout au long de ce chapitre pour montrer les apports des différentes contributions. Elles sont faites sur scènes simulées photoréalistes et sur scènes réelles.

4.2.1/ INDEXATION DES COURBES ÉPIPOLAIRES

La solution d'indexation que nous proposons ici est inspirée de [Li, 2008]. Elle consiste à indexer en ligne les valeurs des pixels par lesquels passent les courbes épipolaires. L'estimation des courbes épipolaires conjuguées est réalisée à partir de l'estimation des rotations épipolaires internes exposée en section 3.3.3.

L'objectif est de définir un ensemble de courbes épipolaires passant par tous les pixels des images. Nous proposons de les calculer à partir d'un arc de scan que nous définissons sur la sphère unitaire comme décrit sur la figure 3.10. Nous définissons ainsi $n_{courbes}$ courbes épipolaires. Le nombre de pixels par lesquels passe chacune d'entre elles varie entre $2r_{max}$ et $2\theta_{max}r_{max}$, où r_{max} est le rayon du disque image en pixels, θ_{max} est la moitié du champ de vision de l'objectif. Ces deux paramètres sont estimés lors de l'étape de calibrage. Dans la suite de ce mémoire, nous choisissons $n_{courbes} = \frac{2}{\sqrt{2}}\theta_{max}r_{max}$ afin que tous les pixels soient indexés au moins une fois. Chaque courbe épipolaire est décrite par $l_{courbes} = 2\theta_{max}r_{max}$ pixels.

Avec ce processus, un même pixel peut être indexé plus d'une fois sur une courbe épipolaire et plusieurs fois sur des courbes épipolaires successives. L'étape de mise en correspondance peut donc appairer un même pixel à différents autres pixels dans l'autre image. Nous proposons d'appliquer une stratégie de vote majoritaire pour ne garder, au final, qu'un seul pixel homologue pour chaque pixel de l'image de référence. Le pixel homologue retenu est celui qui a été apparié le plus de fois avec le pixel de référence sur une même courbe épipolaire ou sur des courbes épipolaires successives.

Dans le cas du modèle sténopé, la disparité est définie comme l'écart (en pixels) entre deux points appariés entre les images et permet de calculer les coordonnées 3D du point P correspondant dans la scène. Dans le cas fisheye c'est un écart angulaire, appelé la disparité sphérique et défini le long des courbes épipolaires par :

$$\Delta(\alpha) = \alpha_r - \alpha_l \quad (4.1)$$

1. <http://vision.middlebury.edu/stereo>

où α_r et α_l sont les positions angulaires d'un couple de pixels appariés, voir figure 3.1. Les paires de pixels ainsi définies, il est alors possible de calculer les coordonnées 3D des points correspondants dans la scène. Ce calcul est détaillé dans la section suivante.

4.2.2/ CALCUL DES POSITIONS 3D

Cette partie décrit deux manières analytiques de calculer les coordonnées 3D des points à partir des correspondances réalisées à l'intérieur des courbes épipolaires conjuguées.

4.2.2.1/ PROJECTION DES RAYONS ISSUS DES DEUX CAPTEURS

Dans la figure 3.1, les droites issues de points correspondants projetés sur les sphères se croisent à la position du point de la scène observé dans l'espace. Les coordonnées de P , dans le repère cartésien de la caméra gauche $(O_l, \mathbf{X}_l, \mathbf{Y}_l, \mathbf{Z}_l)$, peuvent être exprimées en fonction des coordonnées sphériques de ses projections $P_l(R, \theta_l, \varphi_l)$ et $P_r(R, \theta_r, \varphi_r)$. θ est l'angle de colatitude, φ l'angle de longitude.

Dans le repère caméra gauche les coordonnées du centre optique sont $O_l(0\ 0\ 0)^T$ et $O_r(b_x\ 0\ 0)^T$ avec b_x l'entraxe entre les deux vues. Les droites O_lP et O_rP sont données par :

$$O_lP \begin{pmatrix} t \sin(\theta_l) \cos(\varphi_l) \\ t \sin(\theta_l) \sin(\varphi_l) \\ t \cos(\theta_l) \end{pmatrix}_{\text{repère gauche}} \quad \text{avec } t \in [0; +\infty[\quad (4.2)$$

$$O_rP \begin{pmatrix} b_x + t' \sin(\theta_r) \cos(\varphi_r) \\ t' \sin(\theta_r) \sin(\varphi_r) \\ t' \cos(\theta_r) \end{pmatrix}_{\text{repère gauche}} \quad \text{avec } t' \in [0; +\infty[\quad (4.3)$$

P est à l'intersection de O_lP et O_rP , ses coordonnées s'obtiennent en remplaçant t ou t' par :

$$t = \frac{b_x}{\sin(\theta_l) \cos(\varphi_l) - \cos(\theta_l) \tan(\theta_r) \cos(\varphi_r)} \quad (4.4)$$

$$t' = \frac{b_x}{\cos(\theta_r) \tan(\theta_l) \cos(\varphi_l) - \sin(\theta_r) \cos(\varphi_r)} \quad (4.5)$$

Dans la réalité, des erreurs liées à l'échantillonnage font que les rayons projetés ne se croisent pas vraiment. Il faut choisir le point au barycentre du lieu où les droites sont les plus proches.

4.2.2.2/ UTILISATION DES DISTANCES À L'INTÉRIEUR DES PLANS ÉPIPOLAIRES

La disparité sphérique est définie en section 4.2.1. Comme décrit dans la figure 3.1, considérons la distance entre le point P et le centre des deux sphères. Soit l_l et l_r ces distances, respectivement à partir des centres des objectifs gauche et droit. D'après la loi des sinus :

$$l_l = \frac{b_x \sin(\alpha_r)}{\sin(\alpha_r - \alpha_l)} \quad \text{et} \quad l_r = \frac{b_x \sin(\alpha_l)}{\sin(\alpha_r - \alpha_l)} \quad (4.6)$$

Ces résultats permettent de dessiner les cartes des distances et de calculer les coordonnées des points de la scène. Les valeurs α_i peuvent être obtenues de deux manières :

- par la relation $\alpha_i = \arccos(\sin(\theta_i) \cos(\varphi_i))$,
- ou en reprenant l'angle de parcours des courbes épipolaires conjuguées au moment de l'étape de mise en correspondance, qui équivaut à l'angle α_i du point traité.

Les coordonnées de P dans le repère de référence gauche sont définies par :

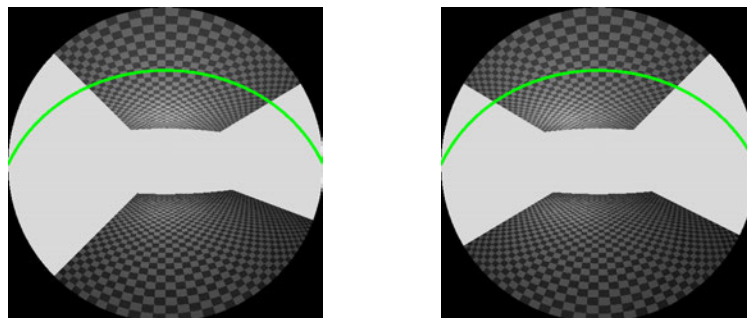
$$P \begin{pmatrix} l_l \cos(\alpha_l) \\ l_l \sin(\alpha_l) \cos(\eta) \\ l_l \sin(\alpha_l) \sin(\eta) \end{pmatrix}_{\text{repère gauche}} \quad (4.7)$$

η est l'angle de rotation du plan épipolaire par rapport à $(O, \mathbf{X}, \mathbf{Y})$, il correspond à la position dans l'arc de scan qui détermine les courbes épipolaires conjuguées (figure 3.10 section 3.3.3). Les angles de longitude φ et de colatitude θ correspondants aux contre-projections de chaque pixels des courbes épipolaires sur la sphère unitaire sont sauvegardés afin de calculer les coordonnées 3D du point P de la scène.

4.2.3/ L'ALGORITHME DE FORTSMANN

La méthode d'appariement que nous avons choisie est une adaptation de l'algorithme de programmation dynamique proposé par [Forstmann et al., 2004]. Il n'est pas question ici de décrire l'intégralité de l'algorithme que le lecteur pourra retrouver dans la publication citée. L'objectif est de définir les parties de l'algorithme que nous avons optimisées pour répondre efficacement à des couples d'images fisheye.

4.2.3.1/ GÉNÉRALITÉS



(a) Courbe épipolaire de l'image 1.

(b) Courbe épipolaire de l'image 2.

FIGURE 4.2 – Paire d'images fisheye générées avec POV-Ray sur lesquelles sont affichées deux courbes épipolaires conjuguées.

La méthode de Forstmann estime la carte des disparités en construisant un graphe pour chaque paire de courbes épipolaires indexées et en recherchant le plus court chemin dans ce graphe. Chaque arc du graphe est défini pour toutes les paires de pixels possibles à partir de la fonction

de distance SAD (*Sum of Absolute Difference*). Si cette technique permet d'assurer la continuité de la disparité le long des structures, elle souffre malheureusement du déséquilibre entre textures mises en correspondance.

Comme nous l'avons présenté dans l'introduction de ce mémoire, nous avons fait le choix de positionner la caméra fisheye verticalement en direction du ciel. Dans cette configuration, la texture des zones de l'image proches des épipôles est fortement atténuée et les discontinuités de la carte des disparités sont perdues. Ce problème est illustré sur la figure 4.3. Sur cette figure, le graphe est calculé à partir des courbes épipolaires conjuguées (en vert) du couple d'images fisheye de la figure 4.2. L'algorithme classique de [Forstmann et al., 2004] donne la priorité à la diagonale lors de la recherche du chemin minimum pour une meilleure continuité des plans. Avec cette stratégie, nous constatons que la discontinuité gauche du bâtiment est effacée. Nous avons donc modifié la recherche du chemin minimum de l'algorithme en privilégiant le parcours vertical du graphe en cas d'égalité. Grâce à cette modification, la discontinuité est préservée et le résultat est meilleur.

Il est important de noter qu'un tel choix favorise les mises en correspondance multiples. La figure 4.4 illustre le phénomène des correspondances multiples sur le nuage de points 3D obtenu. En limitant le nombre de correspondances multiples, nous réduisons la densité de la carte des disparités mais nous réduisons également les erreurs liées à la propagation de la disparité des zones texturées (les bâtiments) dans les zones plus homogènes (le ciel). Le nuage de points 3D est amélioré en conséquence.

Dans un premier temps, nous étudions les performances de l'algorithme de [Forstmann et al., 2004] dans le cas fisheye au travers des deux paramètres suivants :

— **L'offset de corrélation** : *offset*

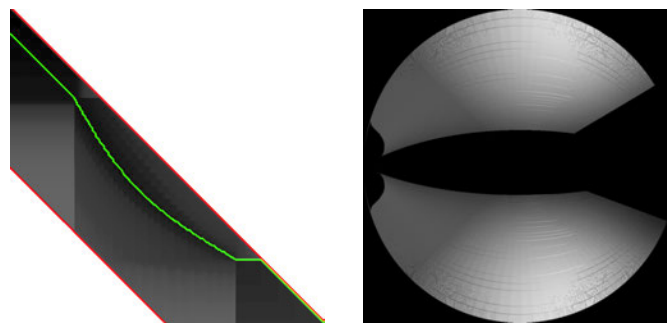
Il s'agit d'ajouter une valeur constante à la mesure de distance afin de réduire l'impact de la mesure précédente ajoutée. L'offset favorise la recherche en diagonale du chemin le plus court, c'est-à-dire qu'il assure la continuité des disparités le long de la courbe épipolaire. Plus la valeur de l'offset est élevée et plus la contrainte de continuité est forte.

— **Le facteur de continuité le long des graphes adjacents** : *const*

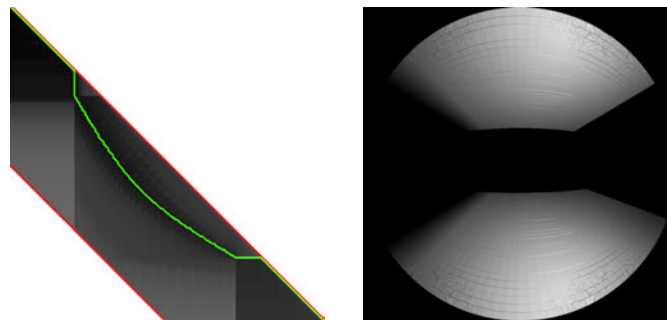
Le but est de tenir compte des chemins les moins coûteux estimés pour les graphes relatifs aux courbes épipolaires conjuguées précédentes. Ce facteur permet d'éviter de grandes différences de disparités entre les courbes épipolaires adjacentes. Ce facteur ne doit pas être trop élevé afin d'éviter de propager trop brutalement les disparités d'une paire de courbes épipolaires à une autre.

4.2.3.2/ ÉVALUATION DES CORRESPONDANCES ET CHOIX DES PARAMÈTRES OPTIMAUX

L'évaluation de la mise en correspondance a été réalisée en utilisant des séquences d'images de synthèse pour lesquelles nous possédons les distances vérité. Ces séquences sont en partie identiques à celles utilisées pour évaluer la méthode de calibrage dans le chapitre précédent. Nous avons ajouté la paire numéro 3 (figure 4.5). La rue de la scène simulée est d'une largeur d'environ 16 mètres, correspondant à des rues assez larges dans la réalité, avec des bâtiments



(a) Priorité diagonale à l'étape de lecture.



(b) Priorité verticale à l'étape de lecture.

FIGURE 4.3 – Calcul du plus court chemin sur un graphe de mise en correspondance, avec étape de lecture selon le sens normal (indice min à max) et en priorité selon (a) la diagonale ou (b) la verticale. À gauche : graphe (limites en rouge) et chemin minimum obtenu (en vert). À droite : carte des disparités sphériques obtenues.

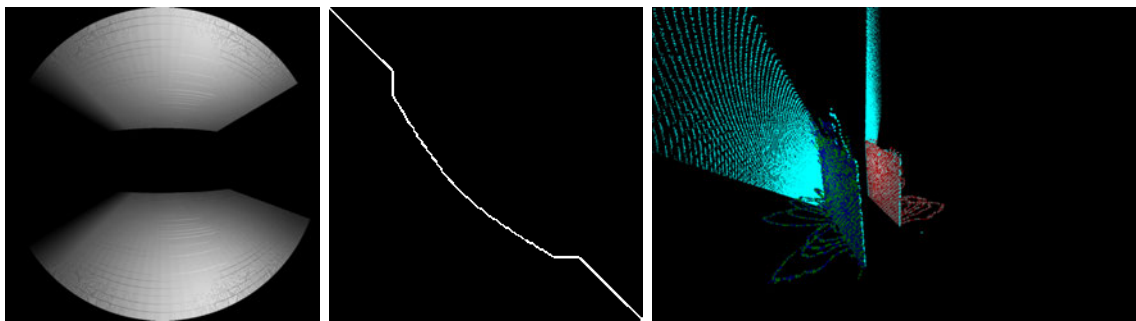
pour la plupart d'une hauteur d'environ 15 mètres. Elles sont listées et décrites dans le tableau 4.1. Nous classons les pixels des images en deux groupes \mathcal{U}_{TI} et \mathcal{I}_{NF} qui correspondent respectivement aux éléments urbains (distance inférieure à 50m) et au ciel (distance supérieure ou égale à 50m).

La mise en correspondance a été réalisée sur les trois séquences en utilisant d'une part les paramètres de calibrage exacts fixés lors de la production des images de synthèse et d'autre part les paramètres de calibrage estimés avec la méthode décrite au chapitre précédent. Le calibrage estimé présente les erreurs suivantes :

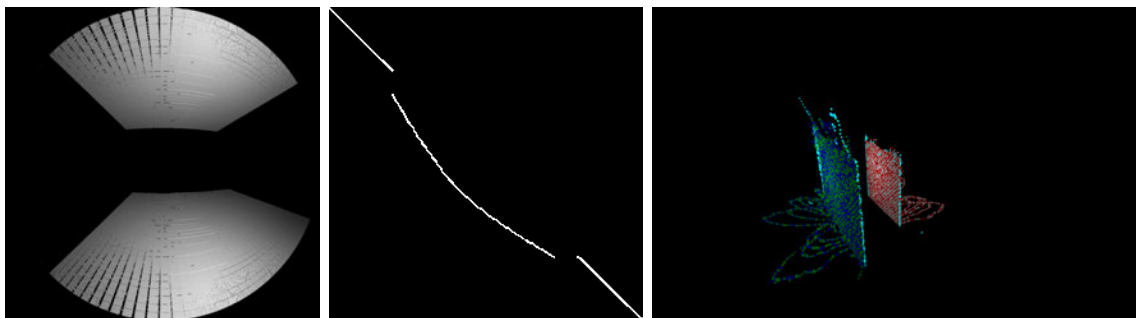
- $|\text{Err FoV}| = 3,89^\circ$,
- $|\text{Err } R_{X,av \rightarrow ar}| = 0,37^\circ$,
- $|\text{Err } R_{Y,av \rightarrow ar}| = 1,36^\circ$,
- $|\text{Err } R_{Z,av \rightarrow ar}| = 0,80^\circ$.

Contrairement aux méthodologies d'évaluation proposées dans la littérature [Scharstein et al., 2002, Chambon, 2005], nous utilisons des critères métriques et non pixelliques. Les métriques suivantes sont utilisées :

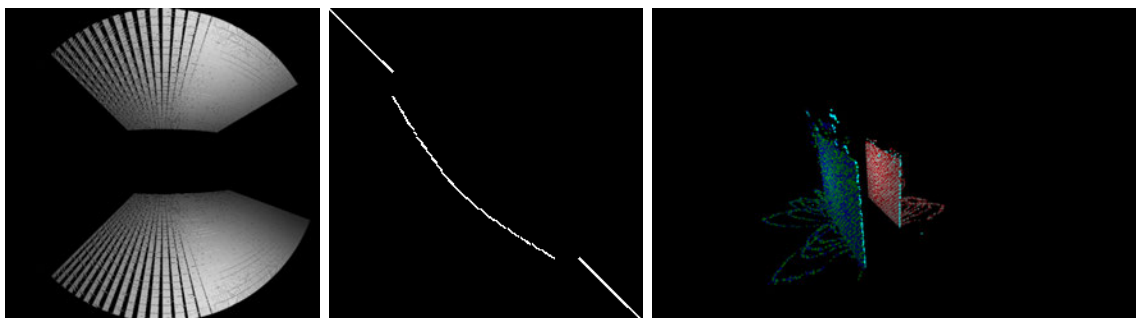
- Le pourcentage de pixels corrects et incorrects : un pixel de la position est correct lorsqu'il se situe à moins de 50cm de la vérité terrain ;



(a) Pas de limite du nombre de correspondants par point.



(b) Limite fixée à 10 correspondants par point.



(c) Interdiction de correspondants multiples, limite fixée à 1 correspondant par point.

FIGURE 4.4 – Conséquences de la limitation du nombre de correspondants par point. De gauche à droite : les cartes des disparités sphériques, l'effet sur un plus court chemin et le nuage de points 3D correspondant.

- Le pourcentage de pixels pour lesquels aucun correspondant n'est estimé ;
- L'erreur absolue moyenne sur tout ou une partie des pixels des images.

Dans la suite, les abréviations utilisées sont définies de la manière suivante :

Mesures sur l'ensemble des points :

COR : Pourcentage de valeurs correctes dans l'ensemble des pixels

OCC : Pourcentage de pixels non-appariés

FXØ : Pourcentage de valeurs incorrectes dans les pixels appariés

$\widehat{Er}|\emptyset$: Erreur absolue moyenne en mètres dans les pixels appariés

Mesures sur les points des structures ($\in U_{TI}$) :

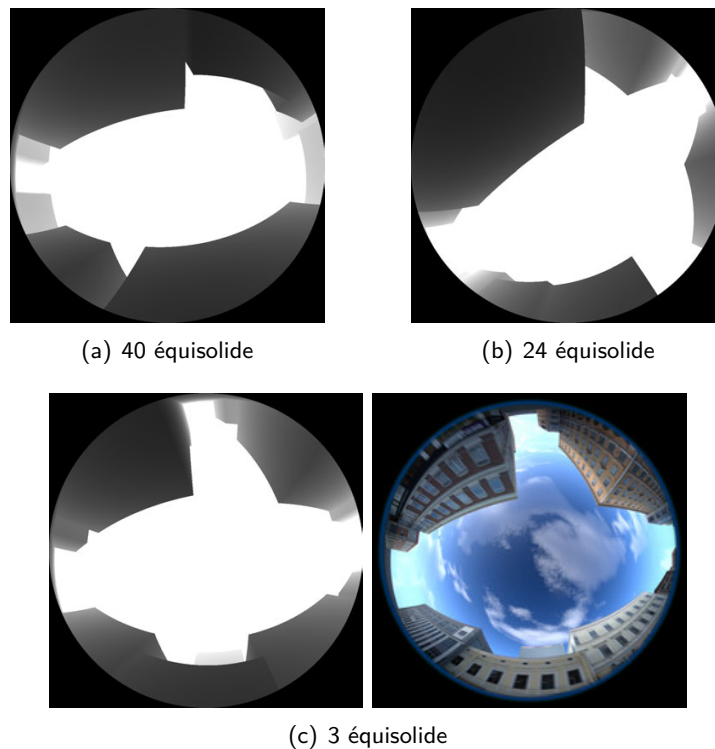


FIGURE 4.5 – Distances vérités des images synthétisées présentées, et en (c) l'image évaluée supplémentaire, représentant les conditions d'un croisement en T.

COU : Pourcentage de valeurs correctes dans les pixels utiles

OCU : Pourcentage de pixels utiles non-appariés

FxU : Pourcentage de valeurs incorrectes dans les pixels utiles appariés

$\widehat{|\text{Er}|U}$: Erreur absolue moyenne en mètres dans les pixels utiles appariés

L'évaluation de la mise en correspondance est menée en faisant varier la valeur d'*offset* sur l'ensemble $\{0, 5, 10, 15, 20, 30, 40, 50\}$ et le paramètre *const* sur l'ensemble $\{0, 1, 2, 5, 10, 15, 20, 25, 30\}$. Toutes les mesures sont calculées sur les images de chaque séquence et séparément sur les deux sous-ensembles de pixels \mathcal{U}_{TI} et \mathcal{I}_{NF} . Afin de définir l'espace des paramètres optimaux au regard de certaines contraintes, nous étudions de plus la moyenne des critères suivants sur toutes les séquences :

— la maximisation du rapport $ratio = \left(\frac{COR}{F_{X\emptyset}} + \frac{COU}{F_{XU}} \right) / 2$

— la minimisation de $\widehat{|\text{Er}|U}$

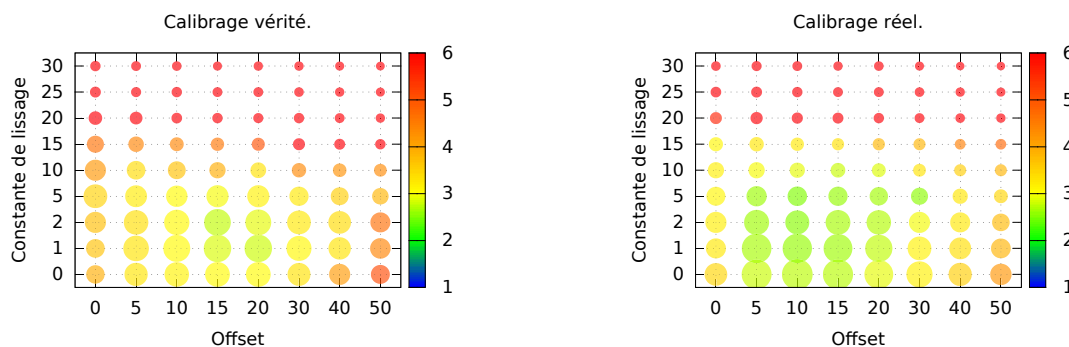
$\widehat{|\text{Er}|U}$ est calculée sur les structures urbaines situées à moins de 20m de la caméra. Ces hypothèses sont faites dans le cadre de l'application CAPLOC qui vise l'amélioration du positionnement GPS. Elle signifie que les signaux GPS sont perturbés majoritairement par les réflexions sur les bâtiments les plus proches de l'antenne.

N° de paire et description		% \mathcal{U}_{TI}	% \mathcal{I}_{NF}
40	mouvement rapide en translation, soleil et réflexions optiques	58,06	41,94
24	mouvement rapide et en virage, flou extrême	53,39	46,61
3	mouvement modéré en translation avec croisement de rue	48,62	51,38
Moyenne		53,36	46,64

TABLEAU 4.1 – Pourcentages de pixels utiles et à l’infini dans les images simulées étudiées.

La figure 4.6 illustre respectivement les valeurs du couple $(ratio, \overline{|Er|U})$ sous la forme d'un diagramme à bulles. Le diamètre des bulles évolue dans le même sens que $ratio$. La couleur des bulles, dont l'échelle est représentée dans les graphes, correspond à $\overline{|Er|U}$ en mètres. Une représentation met en évidence une zone commune de l'espace des paramètres pour laquelle les contraintes sont respectées au mieux. Ce sous-espace est défini par $\{(offset, const) \in \mathbb{N}^+ \times \mathbb{N}^+, offset \leq 20 \text{ et } const \leq 5\}$.

Nous fixons trois jeux de paramètres $(offset, const)$, notés (a), (b) et (c) (voir tableau 4.2) et pour lesquels nous présentons des résultats dans la figure 4.7 et les tableaux 4.3 et 4.4. La figure 4.7 indique également les zones qualifiées de digitales dans les images. Elle permet de mesurer l'offset et de l'offset sur la carte des distances tout particulièrement dans le ciel : un offset non nul permet de réduire le bruit dans le ciel. En étudiant l'image globale, le tableau 4.3 ne permet pas de conclure précisément sur le jeu de paramètres à retenir. Toutefois, pour le sous-groupe \mathcal{U}_{TI} , il montre clairement que le jeu (c) offre l'erreur moyenne la plus faible et le rapport $\frac{COR}{FXO}$ le plus élevé pour les trois séquences.

FIGURE 4.6 – Diagramme à bulles d'évaluation des paramètres de l'algorithme de programmation dynamique utilisé en niveaux de gris. Le diamètre évolue dans le même sens que le critère $ratio$, et la couleur représente $\overline{|Er|U}$ (mètres).

Le tableau 4.4 regroupe les résultats obtenus sur les trois séquences d'images mais avec des paramètres de calibrage estimés. Ce tableau montre que les erreurs moyennes sur l'image globale sont plus faibles que précédemment pour les jeux de paramètres (a) et (c). (c) propose une valeur du rapport $\frac{COR}{FXO}$ plus élevée que dans la première évaluation et la plus grande pour cette seconde évaluation. Toutefois, comme nous l'imaginions, l'erreur moyenne est plus grande sur les structures urbaines quelque soit le jeu de paramètres.

Nom du jeu de paramètres	(a)	(b)	(c)
Valeurs associées	$offset = 10$ $const = 5$	$offset = 0$	$offset = 15$ $const = 2$

TABLEAU 4.2 – Les trois jeux de paramètres ($offset, const$) retenus pour la présentation des résultats sur les scènes simulées 40, 24 et 3.

Calibrage vérité		% pour l'image globale				% pour les structures ($\widehat{U_{TI}}$)				Tps (s)
Params	Paire	COR	OCC	FxØ	$ \widehat{Er} Ø$ (m)	CoU	OcU	FxU	$ \widehat{Er} U$ (m)	
(a)	40	61,31	13,18	25,52	1,83	38,21	20,09	41,71	2,75	19,90
	24	57,34	12,55	30,11	1,93	30,58	19,30	50,12	2,20	18,78
	3	60,37	13,38	26,25	2,72	28,36	21,32	50,32	5,35	20,53
(b)	40	49,42	31,69	18,89	2,30	27,44	42,51	30,06	3,59	20,62
	24	47,38	30,92	21,70	2,04	23,80	40,06	36,13	2,64	20,10
	3	48,63	31,70	19,68	2,84	19,17	42,83	38,00	6,02	20,75
(c)	40	64,90	11,62	23,48	1,82	44,06	18,00	37,93	2,65	19,04
	24	62,98	10,57	26,45	1,93	40,40	16,60	43,00	2,14	18,98
	3	60,98	11,40	27,61	2,90	31,28	18,57	50,15	4,75	20,71

TABLEAU 4.3 – Évaluation pour les jeux de paramètres retenus avec les paires d'images simulées 40, 24 et 3 et les paramètres de calibrage exacts.

Calibrage réel		% pour l'image globale				% pour les structures ($\widehat{U_{TI}}$)				Tps (s)
Params	Paire	COR	OCC	FxØ	$ \widehat{Er} Ø$ (m)	CoU	OcU	FxU	$ \widehat{Er} U$ (m)	
(a)	40	61,38	10,93	27,70	1,80	36,68	17,09	46,23	2,86	17,78
	24	57,80	10,98	31,22	1,46	26,84	17,25	55,91	2,12	17,55
	3	62,71	10,47	26,82	1,77	28,05	17,99	53,96	3,68	17,65
(b)	40	41,88	37,68	20,44	2,45	28,52	37,81	33,68	3,61	17,35
	24	42,81	35,02	22,17	1,67	23,44	36,76	39,80	2,51	17,31
	3	44,98	33,67	21,35	2,22	20,74	36,30	42,95	4,46	17,65
(c)	40	66,07	9,56	24,37	1,73	44,73	14,95	40,32	2,68	16,73
	24	63,31	9,30	27,39	1,67	37,13	14,74	48,13	2,28	16,46
	3	62,77	9,23	27,99	2,03	28,95	15,94	55,10	4,17	16,67

TABLEAU 4.4 – Évaluation pour les jeux de paramètres retenus avec les paires d'images simulées 40 et 24 et les paramètres de calibrage estimés.

4.2.4/ DE L'ALGORITHME DE FORSTMANN AU GRAPHE 3D

Dans un deuxième temps, nous proposons une extension 3D de la méthode de [Forstmann et al., 2004] afin d'améliorer la robustesse à un mauvais calibrage et aux effets de l'échantillonnage.

4.2.4.1/ GÉNÉRALITÉS

Pour cela, pour une courbe épipolaire de l'image de référence, nous construisons un graphe en faisant l'hypothèse que plusieurs courbes épipolaires de l'autre image peuvent, totalement ou partiellement, lui correspondre. Chaque couche de ce graphe correspond donc à un graphe 2D reliant la courbe épipolaire de référence et une courbe épipolaire de l'autre image. Les couches se suivent dans le même ordre que les courbes épipolaires indexées de la seconde image. Ainsi, comme le montre l'illustration 4.8, la valeur de chaque arc du graphe 3D est définie à partir de la valeur des arcs voisins situés sur chaque couche. Dans la suite de ce mémoire, un graphe 3D

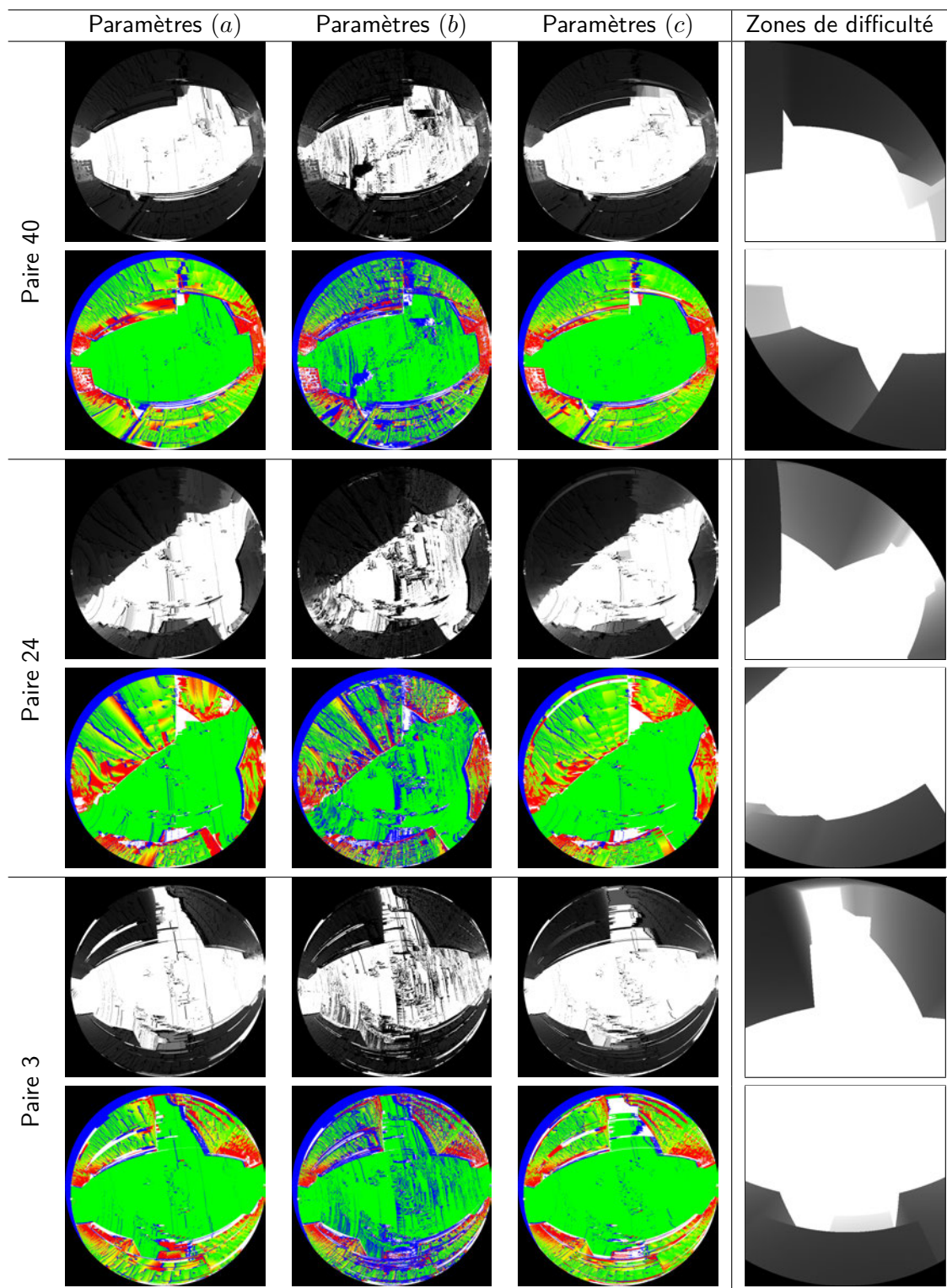


FIGURE 4.7 – Cartes des distances obtenues lors de la mise en correspondance pour les jeux de paramètres retenus avec les paires d'images simulées 40, 24 et 3. Chaque carte des distances est associée à une carte des erreurs graduée sur l'intervalle $[0;2m]$ du vert au rouge. La couleur jaune est utilisée pour une erreur de 1m. Lorsque l'erreur est supérieur à 10m les pixels sont colorés en blanc. Les pixels non-appariés sont représentés en bleu.

construit à partir de n couples de courbes épipolaires conjuguées sera appelé un **graphe 3D n lignes**. La recherche du plus court chemin dans ce nouveau graphe est une adaptation de celle proposée par Forstmann.

La section suivante offre une évaluation de cette nouvelle technique.

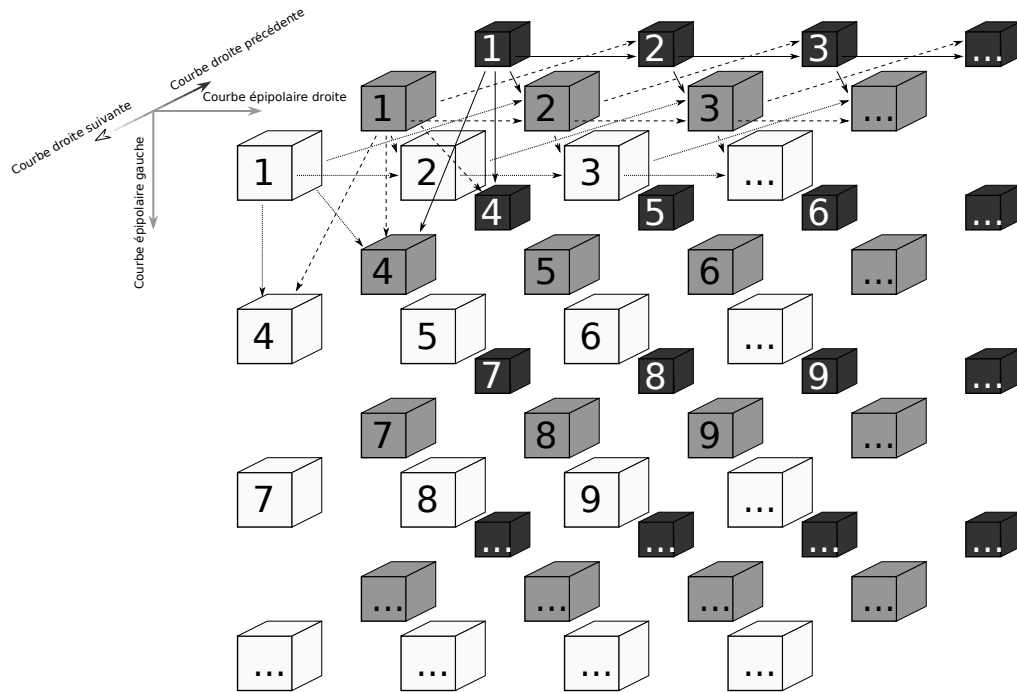


FIGURE 4.8 – Graphe 3D de programmation dynamique pour prendre en compte plusieurs lignes épipolaires voisines.

4.2.4.2/ ÉVALUATION DE LA PROGRAMMATION DYNAMIQUE AVEC GRAPHE 3D

Cette section présente les résultats obtenus avec le graphe 3D que nous avons décrit dans la section précédente. Cette généralisation de l'algorithme de [Forstmann et al., 2004] est caractérisé par les 2 paramètres *offset* et *const* tout comme la version originale. Nous avons mené l'évaluation des mêmes critères que dans la section précédente en faisant varier ces deux paramètres respectivement sur l'ensemble $\{0, 5, 10, 15, 20, 30, 40, 50\}$ et sur l'ensemble $\{0, 1, 2, 5, 10, 15, 20, 25, 30\}$. Nous rappelons que les critères mesurés sont calculés sur les trois séquences de synthèse et sont notés ainsi :

$$- \text{ratio} = \left(\frac{\text{COR}}{\text{Fx}\emptyset} + \frac{\text{CoU}}{\text{FxU}} \right) / 2$$

$$- \overline{|\text{Er}|U}$$

Nous avons testé deux graphes 3D, construits à partir de 3 et 5 paires de courbes épipolaires conjuguées. L'évolution des critères est représentée sur la figure 4.9 sous la forme de diagrammes à bulles pour les paramètres de calibrage exacts, les paramètres de calibrage estimés et pour un graphe 3D construit à partir de 3 et 5 couples de courbes épipolaires conjuguées. Nous observons que dans les deux cas de calibrage le sous-espace des paramètres optimaux est

défini par $\{(offset, const) \in \mathbb{N}^+ \times \mathbb{N}^+, offset \leq 20 \text{ et } const \leq 5\}$.

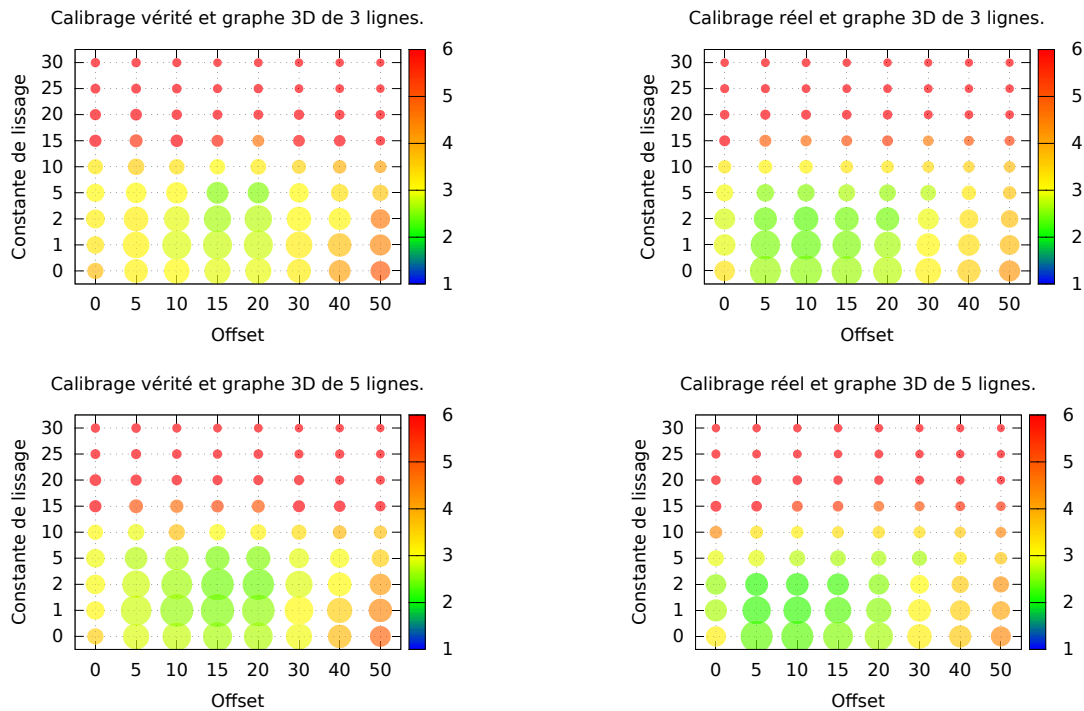


FIGURE 4.9 – Diagramme à bulles d'évaluation des paramètres de l'algorithme de programmation dynamique utilisé en niveaux de gris avec l'utilisation du graphe 3D. Le diamètre évolue dans le même sens que le critère $ratio$, et la couleur représente $\widehat{|Er|U}$ (mètres).

Le tableau 4.5 regroupe les résultats obtenus pour les valeurs $offset = 15$ et $const = 2$ et pour les deux graphes 3D considérés. Nous y rappelons les résultats obtenus pour le meilleur jeu de paramètre (c). Il apparaît clairement que les cartes des distances obtenues avec le graphe 3D construit sur 3 ou 5 lignes sont plus précises et plus denses que celle obtenue avec le jeu de paramètre (c). De plus, le graphe sur 5 lignes apporte de meilleurs résultats que le graphe sur 3 lignes.

Pour illustrer la qualité des résultats, la figure 4.10 montre les cartes de distances et les cartes d'erreurs obtenues pour chaque séquence avec le graphe 3D. Elles sont à comparer avec les cartes de la figure 4.7 obtenues avec le graphe classique. Ces nouvelles cartes de distances semblent plus précises.

Les résultats obtenus avec le calibrage estimé sont résumés dans le tableau 4.6. Les dernières lignes du tableau regroupent les performances obtenues avec le jeu de paramètres (c) dans les mêmes conditions de calibrage. Les résultats obtenus avec les deux graphes 3D sont plus précis que ceux obtenus avec (c). Toutefois, la valeur du $ratio$ est sensiblement la même d'une méthode à l'autre. Il faut donc s'attendre, dans les deux cas, à des cartes de distances et à des nuages de points 3D fortement bruités si le calibrage n'est pas assez précis.

Nous terminerons cette analyse en précisant que le temps de calcul du graphe 3D est supérieur à celui de la méthode classique : il est multiplié par environ 4,5 pour le graphe de 3 lignes

et il est multiplié par 7,5 pour le graphe 3D de 5 lignes. Un travail d'implantation sur une architecture adaptée est donc requise pour atteindre une cadence de traitement raisonnable.

Calibrage vérité		% pour l'image globale				% pour les structures (U_{TI})				Tps (s)
Params	Paire	COR	OCC	FxØ	$ \widehat{Er} Ø$ (m)	CoU	OcU	FxU	$ \widehat{Er} U$ (m)	
3D 3 lignes	40	66,54	9,30	24,16	1,81	46,28	14,56	39,16	2,60	86,64
	24	65,20	8,39	26,42	1,74	41,99	13,48	44,52	2,05	88,57
	3	62,50	9,26	28,24	2,82	33,67	15,47	50,85	4,57	85,31
3D 5 lignes	40	68,89	8,70	22,41	1,68	49,92	13,77	36,31	2,40	149,60
	24	66,51	7,96	25,53	1,69	43,84	13,17	42,99	2,05	141,88
	3	64,58	8,59	26,83	2,51	37,11	14,51	48,38	4,03	142,47
(c)	40	64,90	11,62	23,48	1,82	44,06	18,00	37,93	2,65	19,04
	24	62,98	10,57	26,45	1,93	40,40	16,60	43,00	2,14	18,98
	3	60,98	11,40	27,61	2,90	31,28	18,57	50,15	4,75	20,71

TABLEAU 4.5 – Évaluation de la mise en correspondance avec graphe 3D (3 et 5 lignes) sur les paires d'images simulées 40, 24 et 3 et les paramètres exacts de calibrage.

Calibrage réel		% pour l'image globale				% pour les structures (U_{TI})				Tps (s)
Params	Paire	COR	OCC	FxØ	$ \widehat{Er} Ø$ (m)	CoU	OcU	FxU	$ \widehat{Er} U$ (m)	
3D 3 lignes	40	67,02	7,25	25,73	1,70	45,86	11,41	42,73	2,62	78,12
	24	63,32	7,32	29,36	1,63	36,51	11,60	51,89	2,26	76,63
	3	64,89	7,32	27,80	1,77	32,47	12,94	54,59	3,51	75,23
3D 5 lignes	40	66,29	6,69	27,02	1,72	44,36	10,63	45,01	2,66	123,05
	24	62,09	6,90	31,01	1,62	33,98	11,04	54,98	2,24	127,15
	3	64,64	6,84	28,52	1,57	31,69	12,26	56,05	3,07	122,55
(c)	40	66,07	9,56	24,37	1,73	44,73	14,95	40,32	2,68	16,73
	24	63,31	9,30	27,39	1,67	37,13	14,74	48,13	2,28	16,46
	3	62,77	9,23	27,99	2,03	28,95	15,94	55,10	4,17	16,67

TABLEAU 4.6 – Évaluation de la mise en correspondance avec graphe 3D (3 et 5 lignes) sur les paires d'images simulées 40 et 24 avec les paramètres calibrage estimés.

4.2.5/ ÉVALUATION SUR DES IMAGES RÉELLES

Dans cette section sont présentées des cartes de distances obtenues sur des séquences d'images réelles. La base d'enregistrements stéréoscopiques fisheye utilisée a été acquise sur la métropole lilloise. Elle comporte des centaines d'images enregistrées le long de plusieurs trajets qui traversent des quartiers dont la géométrie est très variable et dans diverses conditions de luminosité. La figure 4.11 illustre quelques-unes de ces propriétés que nous décrivons ainsi :

- (a) Cap2/2012, espace large, les bâtiments occupent peu l'image, place Louise de Bettignies ;
- (b) Cap2/1402, de nombreux arbres d'un côté de la rue, avenue du Peuple Belge ;
- (c) Cap3/1852, reflets du ciel dans les vitres, rue Saint-André ;
- (d) Cap2/0702, sortie de passage sous un pont et soleil causant des reflets internes à l'optique, avenue du Peuple Belge ;
- (e) Cap3/2149, soleil et reflets dans l'optique, effet doublé par sa réflexion dans une fenêtre, rue Saint-André ;

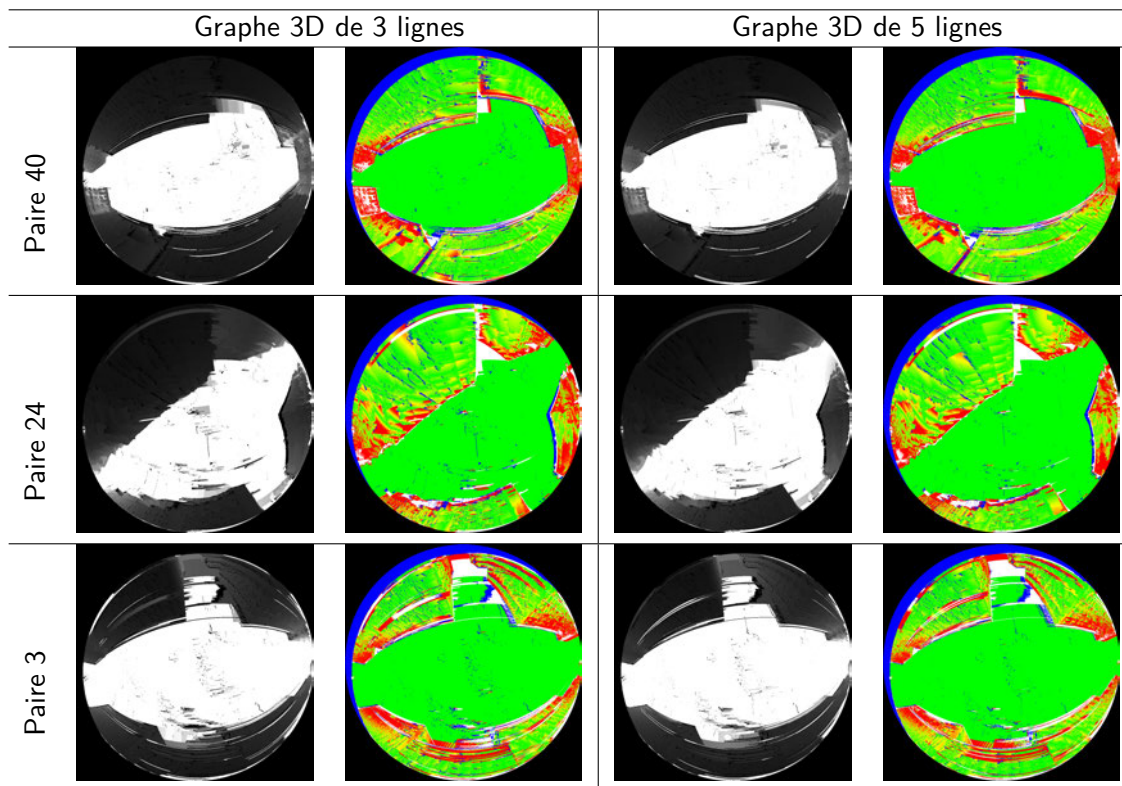


FIGURE 4.10 – Résultat de mise en correspondance avec graphe 3D (3 et 5 lignes) sur les paires d'images simulées 40, 24 et 3 et les paramètres exacts de calibrage.

- (f) Cap2/3094, soleil et reflets dans l'optique, avec réflexion dans une fenêtre et un croisement de rue, rue de la Monnaie ;
- (g) Cap3/2050, soleil et reflet dans l'optique, au milieu d'un carrefour, rue Saint-André.

Suite aux expérimentations menées dans les deux précédentes sections, nous présentons des cartes de distances obtenues en appliquant notre adaptation 3D de la méthode de [Forstmann et al., 2004] dont le graphe a été construit à partir de 3 paires de courbes épipolaires conjuguées. Ce choix a été fait car, dans le cas d'un calibrage estimé, les performances affichées étaient équivalentes à celles du graphe 3D construit à partir de 5 paires de courbes épipolaires et les temps de calcul sont bien moins élevés.

Pour les images réelles nous ne disposons pas de vérité terrain donc nous proposons d'évaluer visuellement la qualité de ces cartes. Par ailleurs nous fixons les valeurs $offset = 15$ et $const = 2$ correspondants au jeu de paramètres (c) dans tous nos tests, ce jeu de paramètres ayant donné les meilleurs résultats lors des évaluations.

Dans les différentes expérimentations, nous avons constaté que les zones situées autour des épipôles présentent des correspondances imprécises. Afin de réduire l'erreur de mise en correspondance portée par l'extrémité du graphe, nous modifions notre méthode en rejetant les pixels situés dans une zone située autour des deux pôles, comme le proposent [Fleck et al., 2009].

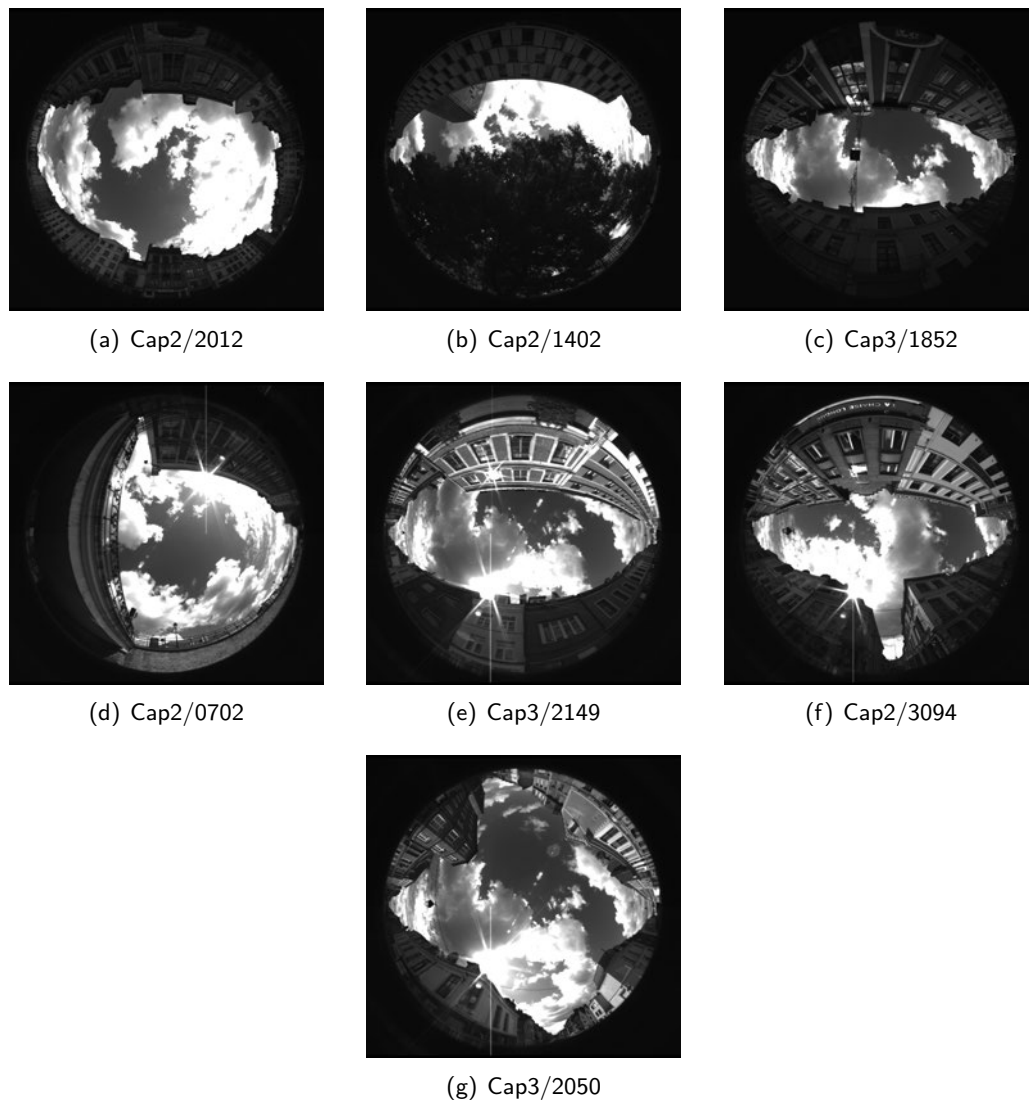


FIGURE 4.11 – Images de conditions réelles rencontrées à Lille.

Cette zone est définie en proportion de la longueur des courbes épipolaires échantillonnées. Nous appliquons cette modification pour une proportion de 25% et 30% de la longueur des courbes épipolaires. Les performances obtenues pour la proportion de 25% se sont avérées supérieures à celles obtenues avec la proportion de 30%. Appelons ($c+$) l'ensemble complet des paramètres utilisés, avec la proportion de rejet égale à 25%. En comparant le tableau 4.7 avec le tableau 4.5 (calibrage exact), nous constatons que le rejet offre des erreurs moyennes plus grandes mais augmente les rapports entre le nombre d'estimations précises et imprécises. Par contre, dans le cas du calibrage estimé, rejeter les régions épipolaires offre des cartes des distances plus précises et des rapports entre nombre d'estimations précises et nombre d'estimations imprécises plus élevés. Ce choix conduit également à la réduction du temps de calcul, est divisé par 1,6.

La figure 4.12 offre quelques cartes de distances calculées sur les images réelles.

Calibrage vérité		% pour l'image globale				% pour les structures (\mathcal{U}_{TI})				
Params	Paire	COR	OCC	FxØ	$ \widehat{Er} Ø$ (m)	CoU	OcU	FxU	$ \widehat{Er} U$ (m)	Tps (s)
(c+)	40	64,96	15,61	19,43	1,66	45,47	23,08	31,46	2,43	59,97
	24	63,04	15,15	21,81	2,13	38,48	25,18	36,34	2,96	58,75
	3	58,31	15,62	26,07	3,08	32,92	20,42	46,66	5,04	59,02

TABLEAU 4.7 – Évaluation de la programmation dynamique en ignorant les zones proches des épipôles pour les meilleures combinaisons de paramètres avec les paires d'images simulées 40, 24 et 3.

Calibrage réel		% pour l'image globale				% pour les structures (\mathcal{U}_{TI})				
Params	Paire	COR	OCC	FxØ	$ \widehat{Er} Ø$ (m)	CoU	OcU	FxU	$ \widehat{Er} U$ (m)	Tps (s)
(c+)	40	65,36	14,12	20,52	1,43	45,03	20,72	34,25	2,29	57,20
	24	61,29	14,44	24,28	1,52	33,59	23,95	42,46	2,10	57,19
	3	60,22	14,12	25,66	1,87	31,18	18,49	50,33	3,70	57,17

TABLEAU 4.8 – Évaluation de la programmation dynamique en ignorant les zones proches des épipôles pour les meilleures combinaisons de paramètres avec les paires d'images simulées 40, 24 et 3, en employant le calibrage réel.

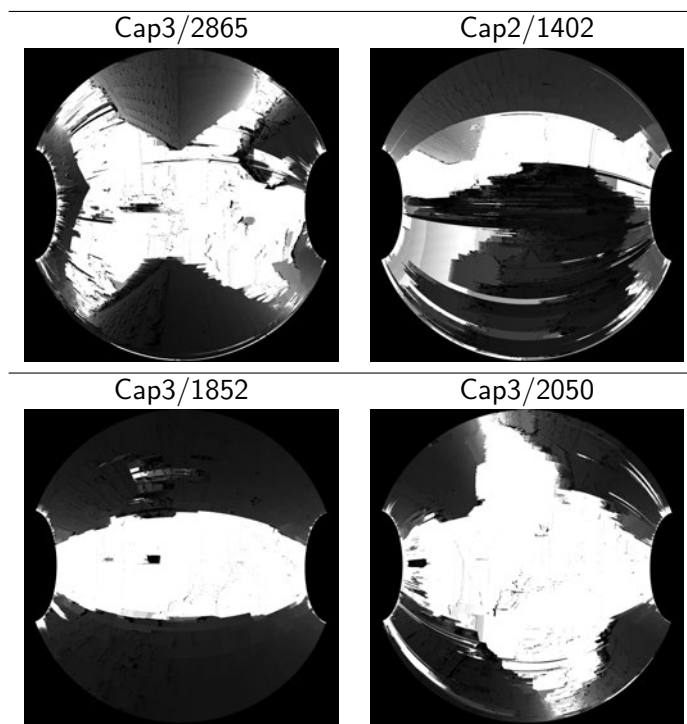


FIGURE 4.12 – Cartes des distances estimées sur des images réelles de Lille par graphe 3D 3 lignes avec $offset = 15$, $const = 2$ et rejet des zones épipolaires 25% (paramètres (c+)).

4.2.6/ CONCLUSION

Cette section a été consacrée à la présentation de la mise en correspondance d'un couple stéréoscopique d'images fisheye pour la reconstruction d'un nuage de points 3D. En premier lieu, nous avons proposé l'évaluation de la méthode proposée par [Forstmann et al., 2004] dans le contexte de ces images sphériques. Dans un second temps, nous avons proposé une adaptation

3D de cette technique associée à plusieurs évolutions qui ont, chacune, amené à l'amélioration de la précision et du rapport entre bonnes et mauvaises estimation. Les meilleurs résultats ont été obtenus lors des tests réalisés avec des paramètres de calibrage estimés correspondant à ce que nous rencontrons dans le contexte des images réelles.

Dans la suite de ce chapitre, nous proposons d'appliquer notre adaptation 3D de la méthode de Forstmann à un couple d'images provenant d'une seule caméra fisheye couleur en mouvement.

4.3/ CORRESPONDANCE MONO-CAMÉRA FISHEYE COULEUR

Dans cette section nous proposons de faire évoluer notre système d'estimation d'un nuage de points 3D vers un système monoculaire fisheye. L'objectif est d'utiliser d'une part l'outil de calibrage décrit dans le chapitre 3 pour retrouver les matrices de transformation entre deux vues acquises au cours du déplacement de la caméra, et d'autre part notre méthode de graphe 3D pour la mise en correspondance de leur contenu.

Un système stéréoscopique calibré permet de retrouver l'échelle du nuage de points 3D construit. Dans le cas mono caméra, le paramètre d'échelle est impossible à retrouver sans à-priori métrique de certains éléments dans la scène. Pour y remédier, nous exploitons un odomètre qui mesure l'écart de position à l'échelle métrique entre deux prises de vues.

Au début de ce chapitre, nous avons travaillé à partir d'un couple stéréoscopique d'images en niveaux de gris. Le système monoculaire que nous exploitons ici fournit des images en couleur que nous exploiterons pour améliorer la précision des cartes de distances estimées.

4.3.1/ ADAPTATION DE NOTRE MÉTHODE GRAPHE 3D

Dans le contexte stéréoscopique (système à deux caméras), puisque le système est rigidement fixé sur le véhicule, la procédure de calibrage n'est réalisée qu'au début de chaque série d'acquisition sous l'hypothèse que, le temps d'un enregistrement, un éventuel dérèglement du système n'a qu'un effet négligeable sur la qualité de la mise en correspondance. Ainsi, les paramètres de calibrage restent identiques pour tout les couples d'images traités.

Dans le cas mono-caméra, cette procédure de calibrage est réalisée à chaque nouvelle acquisition afin de déterminer les rotations et la translation qui permettent d'estimer la pose de la caméra à partir de celle de l'instant précédent. Les paramètres de calibrage ainsi estimés définissent les propriétés épipolaires reliant ces deux vues successives, tout comme dans le cas stéréoscopique pour deux vues acquises simultanément. Il est alors possible pour chaque paire d'images successives d'appliquer notre algorithme de mise en correspondance par graphe 3D proposé dans la section précédente.

Nous rappelons qu'un graphe 3D n lignes est construit à partir de n couples de courbes épipolaires conjuguées et que le plus court chemin dans ce graphe définit les disparités sphériques de chaque pixel de la courbe épipolaire de référence. Le contenu du graphe dépend du nombre n de couples de courbes épipolaires et des paramètres *offset* et *const*.

Le système mono-caméra étant composé d'une caméra couleur, nous avons modifié la fonction

de distance utilisée lors de la construction du graphe. [Chambon et al., 2004] ont étudié en détail comment tirer avantage des informations couleurs dans les mesures de corrélation pour la stéréovision. Les auteurs présentent différentes manières de combiner les trois canaux couleurs :

- Par fusion des scores de corrélation : les corrélations sont calculées à partir de chaque canal puis fusionnées, par exemple en prenant la moyenne ou un barycentre pondéré.
- Par fusion des disparités : les disparités sont calculées séparément avec chaque canal puis une étape de fusion est appliquée, elle consiste par exemple à garder la disparité commune au plus grand nombre de canaux ou celle obtenue avec la meilleure corrélation en cas d'égalité d'occurrences.
- Par analyse en composantes principales (ACP) : l'idée est de réduire le nombre de dimensions pour passer de 3 dimensions à une seule en combinant linéairement les canaux.
- Par une méthode dite directe : la fonction de corrélation est définie directement à partir des composantes couleur.

D'après leurs évaluations, [Chambon et al., 2004] concluent que les méthodes les plus efficaces sont la méthode par fusion de score et la méthode directe. Il faut également indiquer que le choix de l'espace colorimétrique influe sur les résultats. L'espace de couleurs par primaires donne de bons résultats et tout particulièrement l'espace $H_1H_2H_3$ qui est un espace de représentation de la couleur à axes indépendants pour lesquels les composantes sont décorréliées. Ce n'est pas le cas de l'espace RVB dans lequel la luminosité est présente sur chaque composante.

Nous proposons de calculer une mesure de corrélation directement à partir des composantes couleur. Nous choisissons le mode de représentation des couleurs $H_1H_2H_3$ car sa conversion depuis RVB est très simple et peu coûteuse. Il a été proposé par [Braquelair et al., 1997] et ses composantes sont décrites par :

$$\begin{cases} H_1 = R + V, & H_1 \in \llbracket 0; 510 \rrbracket \\ H_2 = R - V, & H_2 \in \llbracket -255; 255 \rrbracket \\ H_3 = B - \frac{R+V}{2}, & H_3 \in \llbracket -255; 255 \rrbracket \end{cases} \quad (4.8)$$

Dans cet espace couleur, nous définissons la corrélation entre deux pixels à l'aide de la distance euclidienne entre leur deux vecteurs couleur. Nous souhaitons que le paramètre *offset* ait le même impact qu'il s'agisse d'images en couleur et en niveaux de gris. Par conséquent, nous appliquons un facteur d'échelle aux mesures de distances couleur afin de les placer sur la même échelle que les distances calculées à partir d'images en niveaux de gris. Ainsi, on propose la mesure de distance couleur en échelle équivalente de la manière suivante :

$$\text{SAD}_{\text{couleur} \leftrightarrow \text{gris}} = \frac{\text{bsup}(\text{SAD})}{\text{bsup}(\text{SAD}_{\text{couleur}})} \text{SAD}_{\text{couleur}} \quad (4.9)$$

où $\text{bsup}(\text{SAD})$ et $\text{bsup}(\text{SAD}_{\text{couleur}})$ sont respectivement les bornes supérieures des intervalles de SAD et de $\text{SAD}_{\text{couleur}}$.

Afin de vérifier cette propriété, nous avons mené l'évaluation précédente sur des images en couleur. Nous avons donc fait varier *offset* et *const* respectivement sur les ensembles $\{0, 5, 10, 15, 20, 30, 40, 50\}$ et $\{0, 1, 2, 5, 10, 15, 20, 25, 30\}$ et avons analysé la qualité des résultats au travers des mêmes critères :

$$- \text{ratio} = \left(\frac{\text{COR}}{\text{FXO}} + \frac{\text{COU}}{\text{FXU}} \right) / 2$$

$$- \overline{|\text{Er}|U}$$

Les résultats sont présentés sur les diagrammes à bulles de la figure 4.13 pour le graphe 3D à 3 et 5 lignes. En comparant ces diagrammes avec ceux de la figure 4.9, nous constatons que le sous-espace optimal des paramètres *offset* et *const* est identique à celui identifié dans le cas des images en niveaux de gris. Dans la suite, nous utilisons le graphe 3D à 3 lignes avec les paramètres $\text{offset} = 10$, $\text{const} = 2$ et le rejet des zones épipolaires à 25% de la longueur des courbes épipolaires indexées, dont nous évaluons les performances sur des séquences d'images réelles acquises à Belfort. Cette évaluation est l'occasion de présenter les modifications proposées pour améliorer les cartes de distances et les nuages de points 3D.

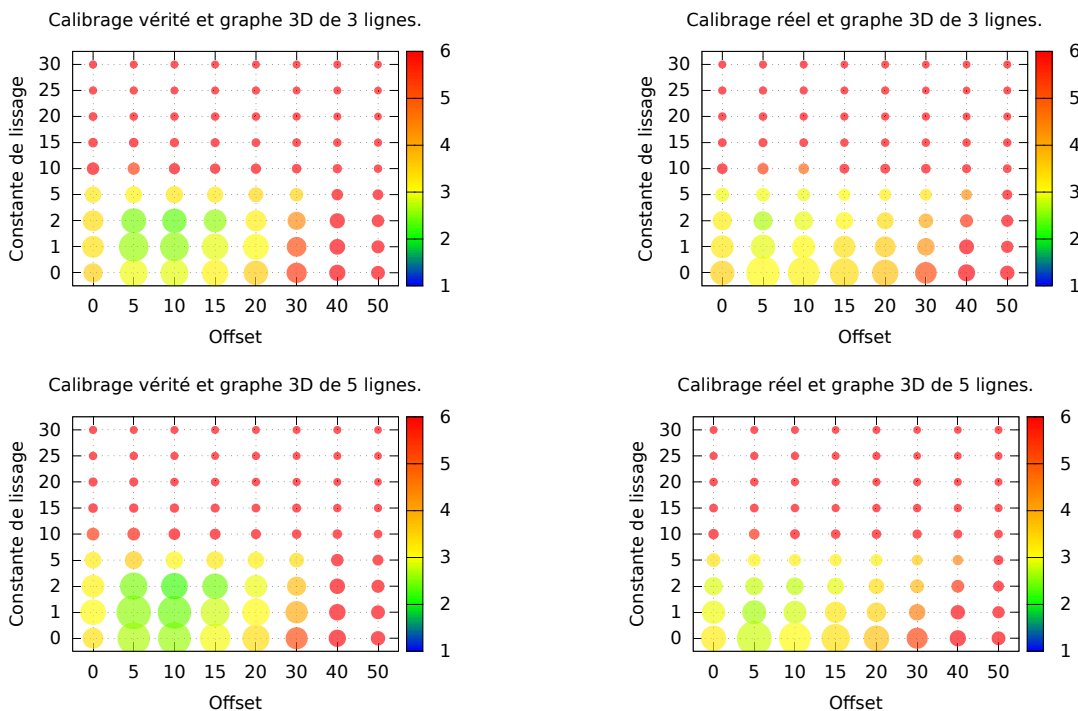


FIGURE 4.13 – Diagramme à bulles d'évaluation des paramètres de l'algorithme de programmation dynamique utilisé avec corrélation couleur et l'utilisation du graphe 3D. Le diamètre évolue dans le même sens que le critère *ratio*, et la couleur représente $\overline{|\text{Er}|U}$ (mètres).

4.3.2/ ÉVALUATION DE SÉQUENCES D'IMAGES RÉELLES

Les images acquises dans la ville de Belfort sont illustrées pour certaines d'entre elles sur la figure 4.14. Elles ont les particularités suivantes :

- (a) 3734, ligne droite, murs aux textures relativement uniformes, rue du Général Roussel ;
- (b) 3785, ligne droite dont l'un des côté est bordé de grands arbres, rues des Bons Enfants ;
- (c) 3694, rue étroite en ligne droite, Grand rue ;
- (d) 3908, rue étroite avec zones sombres et reflets de vitres, rue du Général Roussel ;
- (e) 3940, rue extrêmement étroite, zones sombres et reflets de vitres, rue du Canon d'Or ;
- (f) 4161, rue extrêmement étroite, soleil et rayons pour l'une des deux images, rue du Rosemont ;
- (g) 3705, rue avec bâtiments en retrait formant un « U » et rayons du soleil, Grand rue avec croisement rue de la Botte ;
- (h) 4192, entrée d'un carrefour, croisement entre Grand rue et rue de la Grande Fontaine ;
- (i) 3684, virage au milieu d'un carrefour, forte rotation, virage de rue du Quai à Grand rue ;
- (j) 4339, large place, bâtiments éloignés, place d'Armes ;
- (k) 4096, espace très large, les bâtiments couvrent très peu le champ de vision, rue du Rosemont.

Nous avons appliqué l'algorithme de mise en correspondance sur l'ensemble de ces séquences. Nous présentons certains résultats sur la figure 4.15. Ces cartes de distances sont estimées grâce aux équations de la section 4.2.2 et à partir des correspondances obtenues par le graphe 3D. La distance entre deux acquisitions est mesurée par l'odomètre installé sur le véhicule. Cette dernière mesure doit être précise. Les informations GPS RTK enregistrées simultanément ont permis de valider cette précision. Pour les acquisitions du 22 août 2014, sur 1255 images acquises, on mesure une distance parcourue de 2487 mètres avec le récepteur GPS RTK soit une moyenne de 1,98 mètres entre chaque image. Sachant que nous avons programmé l'acquisition d'une image tous les deux mètres, l'écart moyen obtenu est égal à 0,9%. Ces mesures sont donc parfaitement exploitables pour estimer la cartes des distances.

Sur certaines images de la figure 4.15, nous observons des erreurs dans le ciel. Pour corriger ces erreurs de propagation dues à des différences de texture entre les bâtiments et le ciel, nous proposons de combiner la carte des distances avec l'image de référence sur laquelle nous appliquons une opération de segmentation ciel et bâtiment. La segmentation est assurée en appliquant l'opérateur *MeanShift* [Comaniciu et al., 2002] suivi par une étape de fusion de régions. Nous avons limité le nombre de régions à segmenter afin de réduire le temps de calcul. Cette segmentation fournit donc un masque ciel et bâtiments que nous appliquons aux cartes de distances afin d'imposer une distance infinie aux pixels de ciel.

La figure 4.16 présente les cartes de distances obtenues pour les paires d'images réelles des acquisitions de Belfort après avoir appliqué le masque ciel et bâtiments. On voit une amélioration des cartes en particulier aux carrefours où le ciel présente de nombreuses fausses détections. Bien entendu, la précision des estimations n'est pas augmentée sur les pixels bâtiments. Toutefois, le nuage de points 3D est finalement bien plus exploitable comme nous le montrons dans le chapitre suivant.

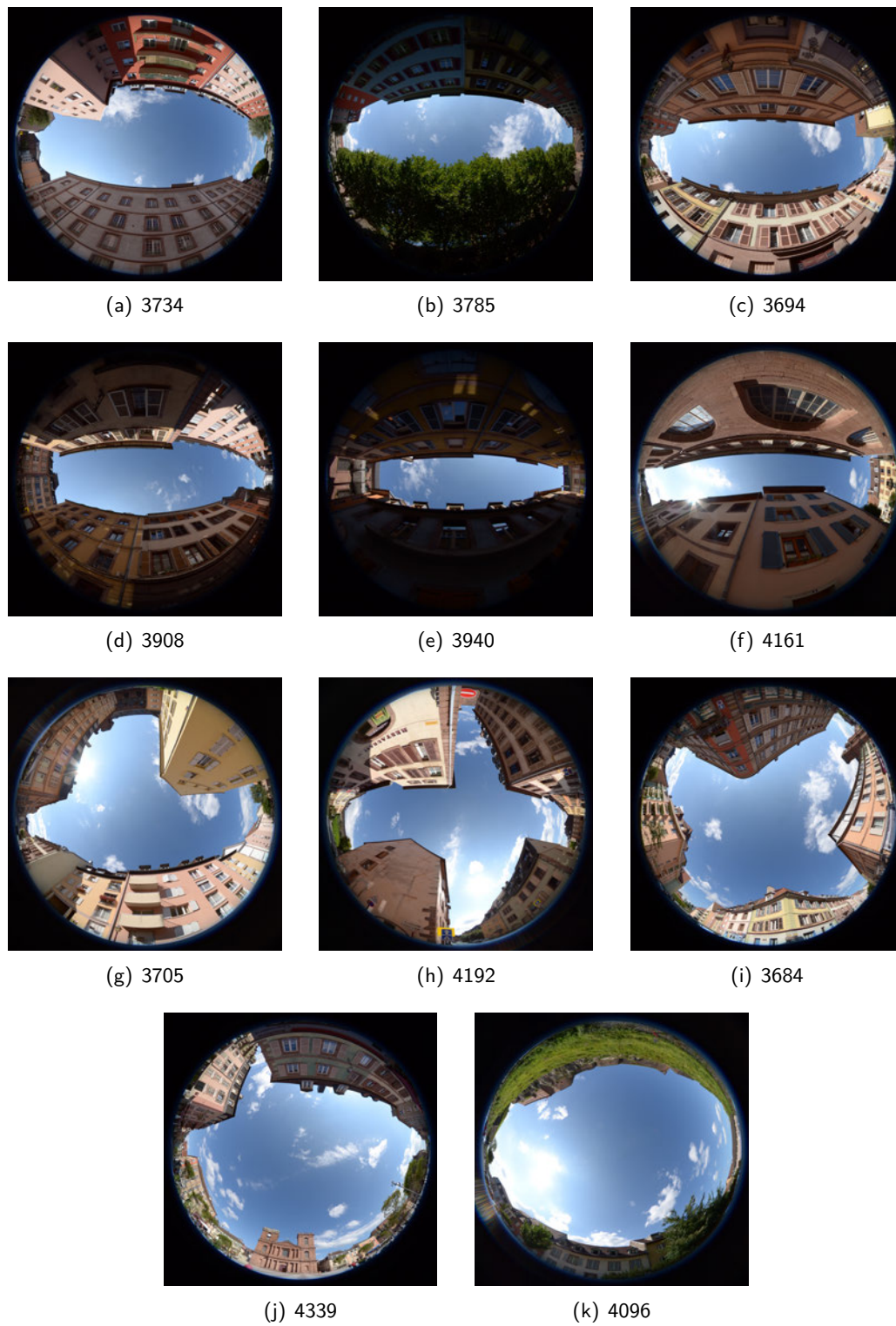


FIGURE 4.14 – Images réelles issues de la séquence de Belfort du 22 août 2014.

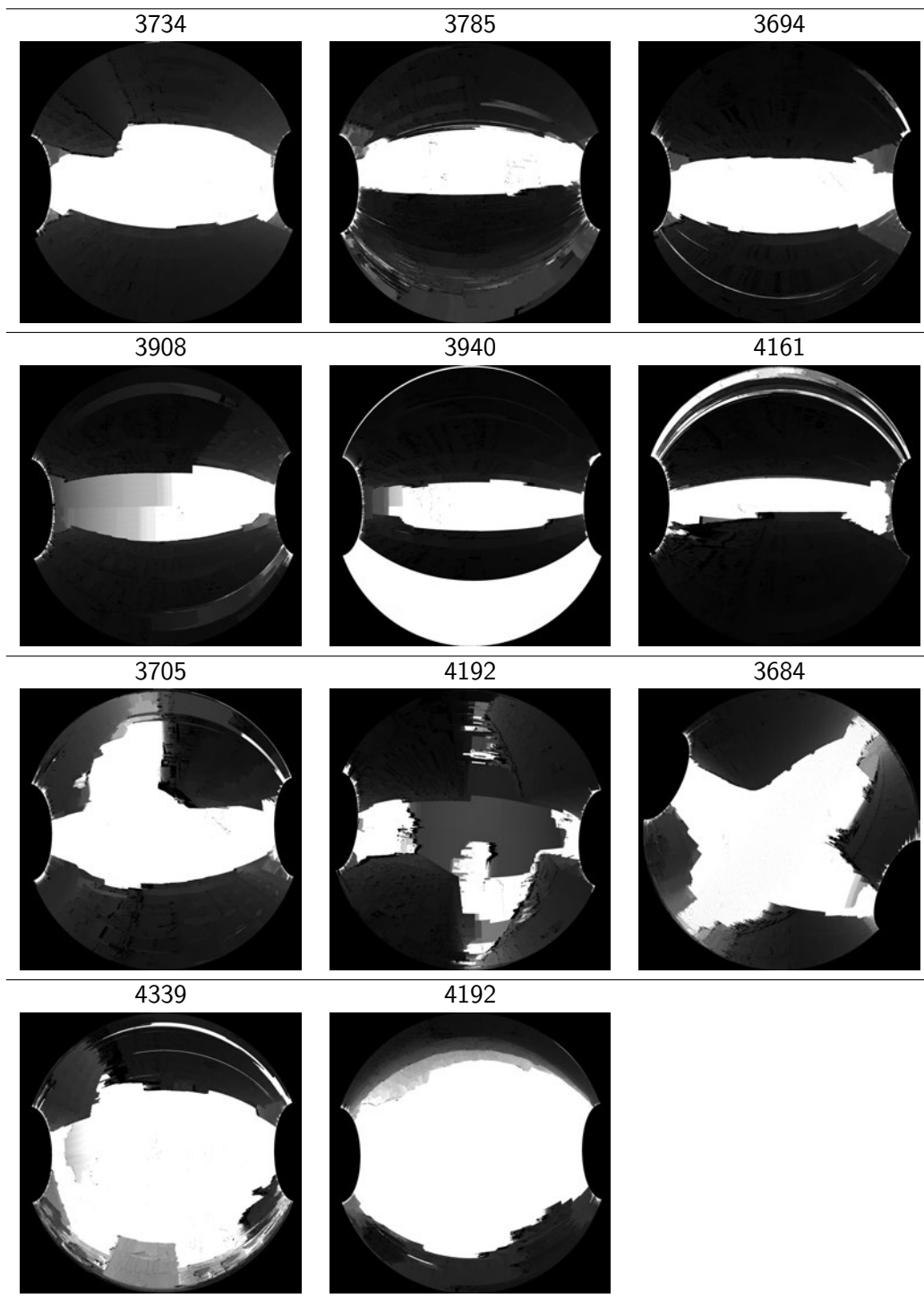


FIGURE 4.15 – Carte des distances estimées par la méthode du graphe 3D sur les paires d’images réelles de Belfort.

4.4/ ANALYSE DES NUAGES DE POINTS 3D

Chaque carte des distances fournit un nuage de points 3D local c’est-à-dire un nuage de points 3D centré sur la position de la caméra choisie comme référence.

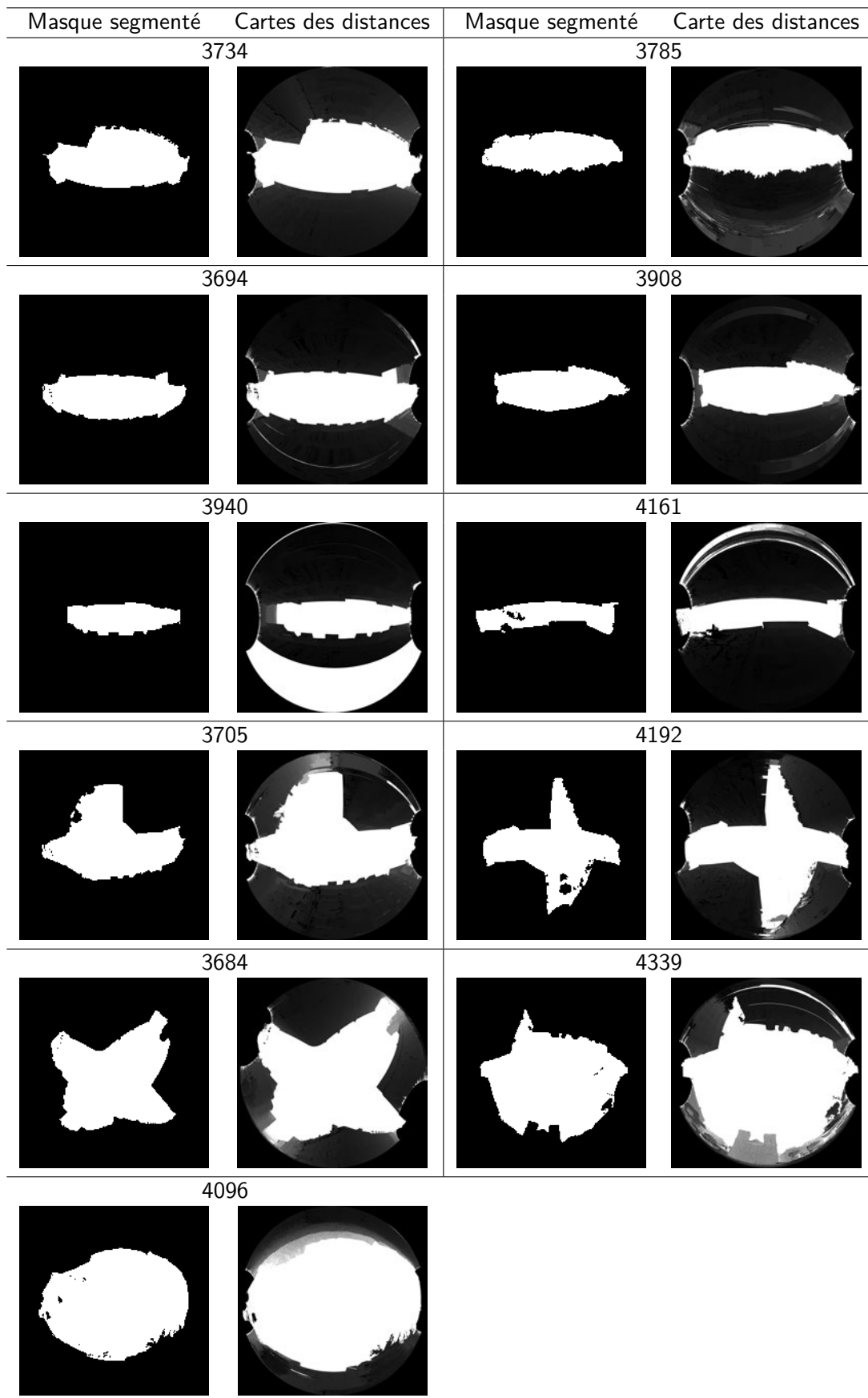


FIGURE 4.16 – Carte des distances estimées par la méthode du graphe 3D sur les paires d'images réelles de Belfort avec utilisation du masque du ciel.

4.4.1/ POST-TRAITEMENT DES NUAGES DE POINTS 3D

Le nuage de points obtenu a un nombre très élevé de points dont certains sont bruités. Dans un premier temps, nous ne conservons que l'ensemble des $\mathbf{P}(X, Y, Z)$ tel que :

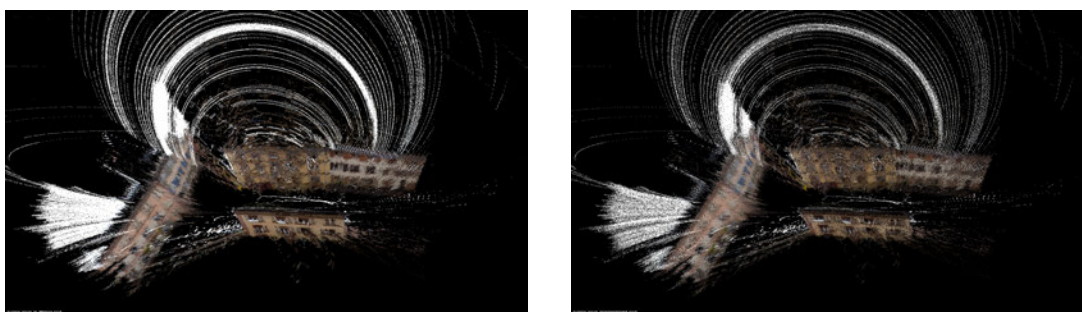
- $Z > 0\text{m}$
- $\|\mathbf{P}\| \leq 50\text{m}$
- $|X| \leq 20\text{m} + \text{entraxe}$ (dans le repère local l'entraxe est dans la même direction que \mathbf{X})
- $|Y| \leq 20\text{m}$

Dans un second temps, nous proposons de leur appliquer les deux traitements suivants :

- Sous-échantillonnage par grille de voxels
- Filtrage local de réduction du bruit

Sous-échantillonnage par grille de voxels

Pour réduire le nombre de points, on applique un filtre à peigne régulier ou plus précisément à grille 3D régulière. ce sous-échantillonnage est réalisé par un filtre appelé *VoxelGrid filter* [Rusu et al., 2011]. L'espace 3D est divisé en voxels de dimension fixe. Pour chaque voxel, s'il y a au moins un point présent, l'ensemble des points contenus sera échantillonné et approximé par un point dont la position est le barycentre de cet ensemble de points. La dimension des voxels utilisée dans les expérimentations est de 10cm. Elle est inférieure au mètre car on cherche une précision de reconstruction métrique. Ce sous-échantillonnage permet de conserver avec précision la forme des surfaces (meilleur que de considérer le centre du voxel par exemple), comme on peut le voir en figure 4.17. Le filtrage par grille de voxel permet par ailleurs de donner une densité égale à des nuages ayant un nombre de points différents. Il est important que la résolution de chaque nuage de points soit supérieure à la résolution de la grille de voxels employée.



(a) Nuage de points avant tout filtrage : 3295950 points.

(b) Nuage de points après sous-échantillonnage : 438442 points (13,3% de points conservés).

FIGURE 4.17 – Sous-échantillonnage d'un nuage de point obtenu lors de nos expérimentations (rue du Général Roussel, Belfort, France).

Filtrage local de réduction du bruit

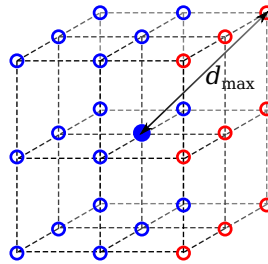


FIGURE 4.18 – Point 3D et ses voisins régulièrement placés dans une grille de voxels, avec d_{max} la distance maximale des points voisins dans le cube englobant.

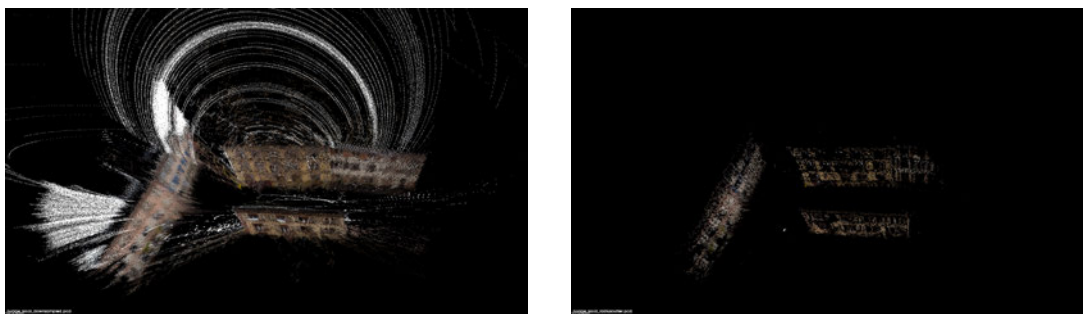
Il utilise une méthode appelée *RadiusOutlier removal* [Rusu et al., 2011] basée sur l'analyse de la densité locale de points en chaque position. En chaque point de l'ensemble, on compte le nombre de points dans un voisinage sphérique autour de lui. Si le nombre de points est inférieur à un seuil demandé alors on le retire du nuage. Dans nos expérimentations, on choisit un rayon d_{max} égal à $1,74 \approx \sqrt{3}$ fois la dimension des voxels utilisée pour le sous-échantillonnage (section 4.4.1), soit un rayon de 17,4cm. Compte-tenu du sous-échantillonnage en grille régulière, avec l'hypothèse de points alignés de manière régulière dans la grille comme le cas de la figure 4.18, on peut estimer qu'un point aura au maximum 26 voisins dans le rayon spécifié. Le seuil est fixé à 15 (26-9) et revient globalement à garder les points qui au pire n'ont aucun voisin d'un seul côté par rapport aux faces du cube de voxels qui l'englobe.

Après ces deux traitements, le nombre de points composant le nuage est réduit de manière drastique et une grande partie des points incohérents est éliminée. On peut noter que lorsque le filtrage est très fort quelques trous apparaissent dans les structures (voir la figure 4.19). Le choix d'un filtrage fort est justifié par la nécessité de réduire au maximum le bruit, les trous pouvant éventuellement être comblés grâce à l'assemblage des nuages de points locaux générés successivement au cours du déplacement de la ou des caméras.

Le tableau 4.9 présente les nuages de points obtenus à partir de certaines images acquises à Belfort. Nous pouvons constater que pour certains d'entre eux, le post-traitement appliqué aux nuages de points réduit considérablement leur densité. Cet effet est particulièrement fort pour l'image 3940 et peut s'avérer préjudiciable dans le cadre de notre application. Sur le tableau 4.9 sont reportées une estimation de la largeur de la rue (méthode décrite en annexe 4.6) et la largeur vérité, que nous avons extraite du plan cadastral et des données cartographiques disponibles sur la plateforme Géoportail de l'IGN². Il apparaît clairement que dans la plupart des cas, l'erreur d'estimation est inférieure à 4% mais elle se dégrade rapidement lorsque la densité de points est insuffisante.

Afin d'y remédier, dans la dernière partie de ce chapitre, nous proposons une étape de consolidation des nuages de points consistant à fusionner deux nuages locaux consécutifs.

2. www.geoportail.gouv.fr



(a) Nuage de points après sous-échantillonnage : 438442 points (13,3% de points conservés). (b) Nuage de points après réduction du bruit : 31938 points (0,99% des points du nuage original).

FIGURE 4.19 – Réduction du bruit par recherche des voisins dans un rayon donné pour un nuage de point obtenu lors de nos expérimentations (rue du Général Roussel, Belfort, France).

4.4.2/ FUSION DES NUAGES DE POINTS 3D

Que ce soit dans le cas stéréoscopique ou mono-caméra, la mise en correspondance par graphe 3D présentée précédemment permet d'estimer un nuage local de points 3D pour toute paire d'images fisheye. Il s'agit maintenant d'assembler ces nuages locaux successivement pour construire un modèle global au fur et à mesure du déplacement du véhicule. La contribution apportée ici est une procédure géométrique pour assembler les nuages de points locaux qui exploite la géométrie épipolaire et les repères épipolaires alignés de deux vues auxquelles la procédure de calibrage, décrite au chapitre 3, a été appliquée.

La fusion des nuages de point exploite les matrices de rotation et la translation estimées lors du calibrage. Dans le cas mono-caméra, ce dernier paramètre est estimé grâce aux informations fournies par l'odomètre. En revanche, dans le cas stéréoscopique, le paramètre d'échelle nécessite une étape supplémentaire d'odométrie visuelle. Une fois la géométrie épipolaire retrouvée, si O_{ref} est le centre optique de la vue de référence à l'instant considéré, alors un nuage de points 3D est défini dans le repère orthonormal épipolaire $\mathfrak{R}_{epip} = (O_{ref}, \mathbf{X}_{epip}, \mathbf{Y}_{epip}, \mathbf{Z}_{epip})$. L'axe \mathbf{X} est défini dans la direction de la translation et relie le centre optique des deux vues et l'axe \mathbf{Z}_{epip} est supposé confondu avec l'axe vertical, en direction du ciel.

4.4.3/ CAS MONO-CAMÉRA

Un nuage de points 3D est généré à chaque nouvelle acquisition : à l'instant t un nuage de points est estimé à partir des images acquises aux instants $t-1$ et t . Appelons \mathcal{N}_t le nuage local de l'instant t . \mathcal{N}_t est référencé dans le repère $\mathfrak{R}_{epip,t} = \mathfrak{R}_{epip,t,O(t-1)} = (O_{t-1}, \mathfrak{B}_{epip,t})$ centré en O_{t-1} le centre optique de la caméra à l'instant $t-1$ avec $\mathfrak{B}_{epip,t} = (\mathbf{X}_{epip,t}, \mathbf{Y}_{epip,t}, \mathbf{Z}_{epip,t})$. De la même manière, le nuage \mathcal{N}_{t-1} est décrit dans le repère $\mathfrak{R}_{epip,t-1} = \mathfrak{R}_{epip,t-1,O(t-2)}$. L'image à l'instant $t-1$ est commune à la création des nuages \mathcal{N}_{t-1} et \mathcal{N}_t . On cherche à projeter \mathcal{N}_t dans le repère $\mathfrak{R}_{epip,t-1}$ afin de fusionner les deux nuage de points. La figure 4.20 illustre les différents repères. La transformation de passage du repère local de \mathcal{N}_t au repère local de \mathcal{N}_{t-1} en coordonnées homogènes s'exprime par :

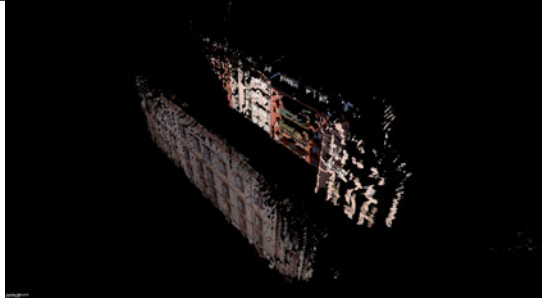
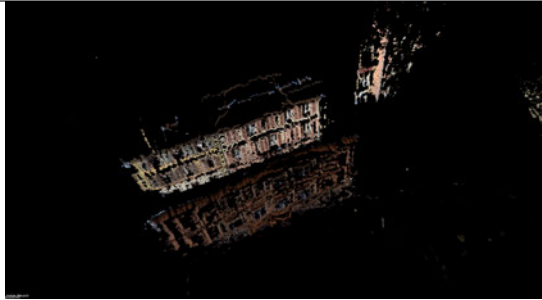
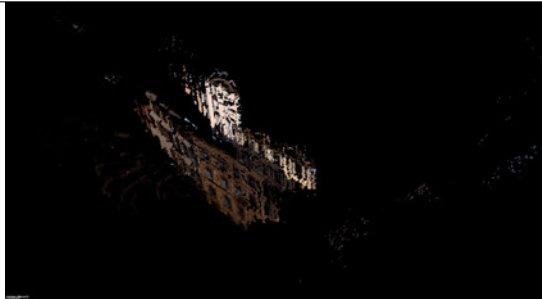
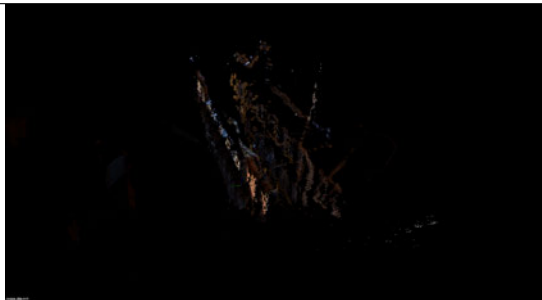
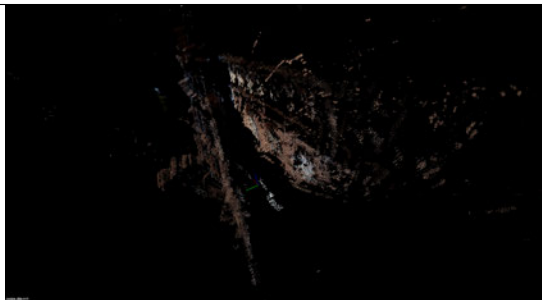
	Nuage de points	Largeur vérité	Mesures dans le nuage Bande 1m ($\delta = 0.5$)
3734		$\approx 16\text{m}$	15,6m (15,6m sans filtrage)
3694		$\approx 8\text{m}$	8,2m (8,3m sans filtrage)
3908		$\approx 8\text{m}$	8,2m (8,3m sans filtrage)
3940		$\approx 5\text{m}$	12,9m (5,0m sans filtrage)
4161		$\approx 5\text{m}$	5,2m (5,3m sans filtrage)

TABLEAU 4.9 – Évaluation de la largeur des rues pour les scènes réelles mono de Belfort en ligne droite avec mise en correspondance couleur, masque du ciel et filtrage des données 3D.

$$H(\mathfrak{R}_{epip,t} \rightarrow \mathfrak{R}_{epip,t-1}) = \begin{pmatrix} R(\mathfrak{R}_{epip,t} \rightarrow \mathfrak{R}_{epip,t-1,O(t-1)}) & \mathbf{t}(O_{t-2} \rightarrow O_{t-1}) \\ \mathbf{0}_{1 \times 3} & 1 \end{pmatrix} \quad (4.10)$$

où $\mathfrak{R}_{epip,t-1,O(t-1)} = \mathfrak{R}_{epip,t-1,O(t-2)} + (\text{entaxe}, 0, 0)^T$.

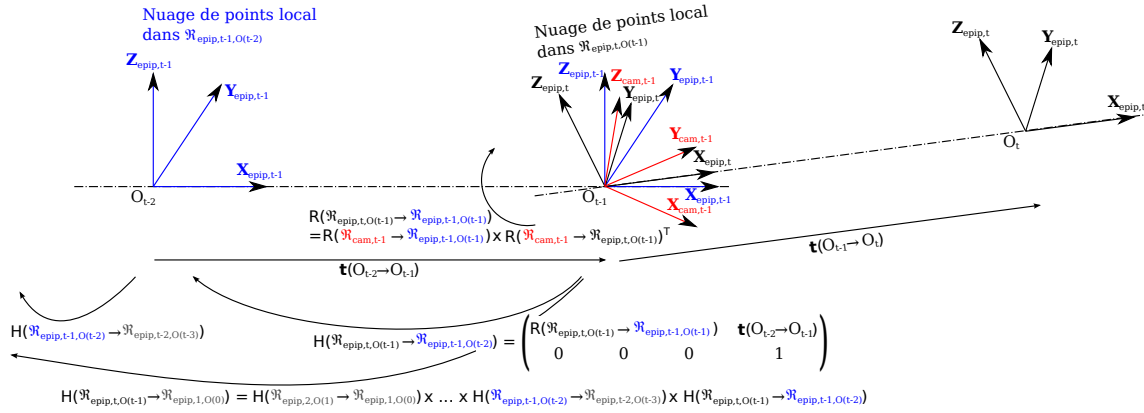


FIGURE 4.20 – Transformations pour changer de repère et pouvoir concaténer les nuages de points. $\mathfrak{R}_{cam,t} = (O_t, \mathbf{X}_{cam,t}, \mathbf{Y}_{cam,t}, \mathbf{Z}_{cam,t})$ est le repère caméra de la vue de l'instant t . $\mathfrak{R}_{epip,t,O(t')} = (O_{t'}, \mathbf{X}_{epip,t}, \mathbf{Y}_{epip,t}, \mathbf{Z}_{epip,t})$ est le repère épipolaire entre les vues des instants $t - 1$ et t , centré sur l'origine de l'instant t' .

La figure 4.21 montre l'assemblage de 10 et 20 enregistrements successifs en début de la rue du Général Roussel à Belfort. La figure 4.22 montre le modèle complet de cette même rue sur lequel sont représentées la position des caméras.



FIGURE 4.21 – Nuage de points après assemblage de 10 et 20 nuages de points 3D successifs (rue du Général Roussel, Belfort, France).

4.4.4/ CAS BINOCULAIRE

Dans le cas binoculaire, un nuage de points 3D est estimé à chaque instant pour un couple d'images stéréoscopiques. Une des difficultés pour l'assemblage des modèles locaux est que l'on ne dispose pas directement de l'information de rotation et de translation entre deux couples



FIGURE 4.22 – Reconstruction de la rue du Général Roussel avec affichage des caméras aux positions des acquisitions en jaune.

d'images de deux instants successifs. Dans cette configuration et en l'absence d'autres capteurs, nous proposons d'appliquer notre méthode de calibrage décrite dans le chapitre 3 au travers des étapes suivantes :

1. Estimer les matrices de passage entre la pose des deux caméras de référence de deux couples d'images stéréoscopiques (paramètres extrinsèques dans le chapitre 3) ;
2. Estimer l'échelle grâce aux reprojections 3D des paires de points *inliers* après le RANSAC de l'étape précédente ;
3. Reprojecter et assembler les nuages locaux dans le repère de référence.

Considérons des acquisitions stéréo successives aux instants $t - 1$ et t . À la différence du cas monoculaire (voir section 4.4.3), un nuage local est estimé à chacun de ces deux instants. Dans le cas monoculaire, l'origine du nuage de points est placée à la position de la prise de vue de l'instant précédent. Dans le cas binoculaire, son origine est placée à la position de la caméra dite arrière. Si O_{ar} est le centre optique de la caméra arrière, alors le repère épipolaire est défini par $\mathfrak{R}_{epip} = (O_{ar}, \mathbf{X}_{epip}, \mathbf{Y}_{epip}, \mathbf{Z}_{epip})$.

Soit \mathcal{N}_{t-1} et \mathcal{N}_t deux nuages de points que nous cherchons à assembler : \mathcal{N}_{t-1} a pour repère local $\mathfrak{R}_{epip,t-1}$ et \mathcal{N}_t a pour repère local $\mathfrak{R}_{epip,t}$. On appelle $R_{epip,t-1,ar}$ et $R_{epip,t,ar}$ les rotations épipolaires des caméras arrières aux instants $t - 1$ et t . On peut écrire la transformation du repère local de \mathcal{N}_t au repère local de \mathcal{N}_{t-1} en coordonnées homogènes par :

$$H(\mathfrak{R}_{epip,t} \rightarrow \mathfrak{R}_{epip,t-1}) = \begin{pmatrix} R_{t \rightarrow t-1} & \mathbf{t}_{t \rightarrow t-1} \\ \mathbf{0}_{1 \times 3} & 1 \end{pmatrix} \quad (4.11)$$

Le passage de $\mathfrak{R}_{epip,t}$ à $\mathfrak{R}_{epip,t-1}$ est composé d'une rotation $R_{t \rightarrow t-1}$ et d'une translation $\mathbf{t}_{t \rightarrow t-1}$. Appliquer un calibrage entre les deux vues de la caméra arrière à $t - 1$ et t permet de définir

$\mathfrak{R}'_{epip,(t-1,t)}$, le repère épipolaire pour cette paire d'images arrières, et d'estimer $R'_{epip,t-1,ar}$ et $R'_{epip,t,ar}$, leurs rotations épipolaires respectives. On peut alors exprimer $R_{t \rightarrow t-1}$ comme suit :

$$R_{t \rightarrow t-1} = R_{epip,t-1,ar} \times \underbrace{R'^T_{epip,t-1,ar} \times R'_{epip,t,ar}}_{\text{passage entre repères caméra}} \times R^T_{epip,t,ar} \quad (4.12)$$

La translation peut être exprimée simplement dans le repère épipolaire du calibrage des caméras arrières aux instants $t - 1$ et t par :

$$\mathbf{t}(O_t \rightarrow O_{t-1}) = \begin{pmatrix} \text{entraxe} \\ 0 \\ 0 \end{pmatrix} \quad (4.13)$$

C'est-à-dire que si on veut se placer dans le passage du repère $\mathfrak{R}_{epip,t}$ au repère $\mathfrak{R}_{epip,t-1}$, il faut écrire :

$$\mathbf{t}_{t \rightarrow t-1} = R_{epip,t-1,ar} \times R'^T_{epip,t-1,ar} \times \begin{pmatrix} \text{entraxe} \\ 0 \\ 0 \end{pmatrix}^T \times R'_{epip,t,ar} \times R^T_{epip,t,ar} \quad (4.14)$$

Nous proposons d'estimer la translation $\mathbf{t}_{t \rightarrow t-1}$ de la transformation en mettant en correspondance la projection 3D des *inliers* retenus lors du calibrage stéréoscopique à l'instant $t - 1$ et les *inliers* retenus lors du calibrage des deux images arrières aux instants $t - 1$ et t . L'image commune étant l'image de la caméra arrière acquise à l'instant $t - 1$, on ne garde que les *inliers* qui possèdent à la fois un correspondant dans l'image avant de l'instant $t - 1$ et un correspondant dans l'image arrière de l'instant t . On obtient ainsi des triplets de points correspondants dans les trois images. La projection 3D des *inliers* est assurée grâce aux relations de la section 4.2.2 : la projection dans le repère $\mathfrak{R}_{epip,t-1}$ s'effectue à partir des vues arrière et avant aux instant $t - 1$ et la projection dans le repère $\mathfrak{R}'_{epip,(t-1,t)}$ est réalisée à partir des vues arrières aux instants $t - 1$ et t .

À l'inverse de la projection 3D dans le repère $\mathfrak{R}_{epip,t-1}$ pour laquelle l'entraxe entre la caméra avant et arrière est connue, la projection dans le repère $\mathfrak{R}'_{epip,(t-1,t)}$ nécessite de fixer un entraxe arbitraire Δ . L'idée est alors de retrouver l'échelle en calculant les rapports r_k entre les distances séparant chaque paire de points 3D successifs décrits dans le repère $\mathfrak{R}_{epip,t-1}$ et celles de leurs homologues dans le repère $\mathfrak{R}'_{epip,(t-1,t)}$.

Ainsi si P_k est le k^e point de l'ensemble des *inliers* projetés en 3D dans $\mathfrak{R}_{epip,t-1}$ et P'_k son homologue dans $\mathfrak{R}'_{epip,(t-1,t)}$, on définit r_k de la manière suivante :

$$r_k = \frac{\|P_k - P_{k-1}\|}{\|P'_k - P'_{k-1}\|} \quad (4.15)$$

avec $k \in \llbracket 2; N \rrbracket$ et N le nombre d'*inliers*.

Nous définissons alors le facteur d'échelle comme la moyenne des r_k :

$$\hat{r} = \frac{1}{N-1} \sum_{k=2}^N r_k \quad (4.16)$$

et l'entraxe par la relations suivante :

$$entraxe = \hat{r} \cdot \Delta \quad (4.17)$$

Le calibrage stéréoscopique peut ne pas être réalisé tout au long du trajet. Mais cette méthode peut être appliquée simplement en remplaçant l'ensemble des *inliers* manquant par la projection 3D d'un ensemble de points d'intérêt 2D appariés extraits du couple d'images stéréoscopique à l'instant $t - 1$. Un RANSAC peut être utilisé pour éliminer les *outliers* et estimer en même temps le facteur d'échelle. Ce RANSAC se base sur une seule paire de points 3D à chaque tirage, ce qui permet d'arriver à une bonne stabilité.

4.4.5/ ÉVALUATION DE L'ODOMÉTRIE VISUELLE

Nous terminons cette section par une évaluation de la procédure d'odométrie visuelle que nous avons exploitée dans les sous-sections précédentes pour assembler les nuages locaux de points 3D. Pour chaque base de données vidéo acquise, nous avons enregistré l'ensemble des positions géographiques du véhicule au cours de son trajet grâce à un récepteur GPS RTK. Nous proposons donc de comparer les données d'odométrie estimées avec notre méthode aux données d'odométrie estimées à partir des informations RTK. Nous assurons cette évaluation en comparant les vecteurs de déplacement (RTK et odométrie visuelle) entre deux instants $t - 1$ et t .

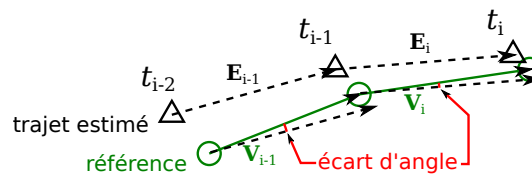


FIGURE 4.23 – Mesure de l'écart de module et d'argument entre les vecteurs liant les points successifs des trajectoires estimée et vérité.

Soit \mathbf{E}_t le vecteur entre les positions estimées des instants $t - 1$ et t et \mathbf{V}_t le vecteur entre les positions RTK en ces mêmes instants. La figure 4.23 montre le principe de cette évaluation. On propose des métriques ne dépendant pas de l'orientation initiale des trajets :

- L'erreur de longueur absolue : l'écart absolu de longueur (module) entre les vecteurs d'odométrie visuelle et RTK (en mètres)

$$|\text{Err}_{\text{longueur}}| = ||\mathbf{V}_i|| - ||\mathbf{E}_i|| \quad (4.18)$$

- L'erreur angulaire absolue : l'angle absolu entre les vecteurs d'odométrie visuelle et RTK

(en degrés),

$$|\text{Err}_{\text{angulaire}}| = |\arg(\mathbf{E}_i, \mathbf{V}_i)| \quad (4.19)$$

Nous proposons également de renseigner la moyenne et la valeur maximale de ces deux mesures pour chaque séquence évaluée ainsi que les mesures statistiques de médiane et d'écart-type inspirées des évaluations de [Eudes, 2011] (travail en caméra sténopé). L'erreur de position finale n'est donnée qu'à titre indicatif car elle dépend de l'erreur propagée ayant tendance à augmenter plus la séquence s'allonge.

Comme nous l'indiquons dans la section précédente, l'odométrie exploite pleinement la méthode de calibrage proposée. Nous comparons les estimations obtenues avec celles obtenues par la décomposition classique de la matrice essentielle décrite entre autres dans [Baselgia, 2010]. Nous ne montrons de résultats que pour la séquence d'images acquise à Belfort pour laquelle le récepteur GPS RTK n'est pas passé en mode dégradé DGPS garantissant ainsi une précision optimale et centimétrique des positions vérité. Deux extraits sont ainsi présentés :

- I : trajet en ligne, rue Roussel 3^e passage, 88 nuages locaux
- O : boucle, grand rue à rue du Quai, 122 nuages locaux

Les figures 4.24 et 4.26 illustrent les modèles 3D globaux obtenus par cumul des nuages de points locaux. Les figures 4.25 et 4.27 présentent les mesures d'erreur angulaire et de longueur. Pour les deux séquences l'erreur de longueur est en moyenne d'environ 40cm. Cette erreur correspond au système d'odométrie sur roue mis en place. On peut observer des pics d'erreur de plus de 1m, surtout dans les positions des virages. Cette erreur est assez grande compte-tenu du fait que le système d'odométrie sur roue utilisé à Belfort n'est pas très précis, en particulier dans les virages.

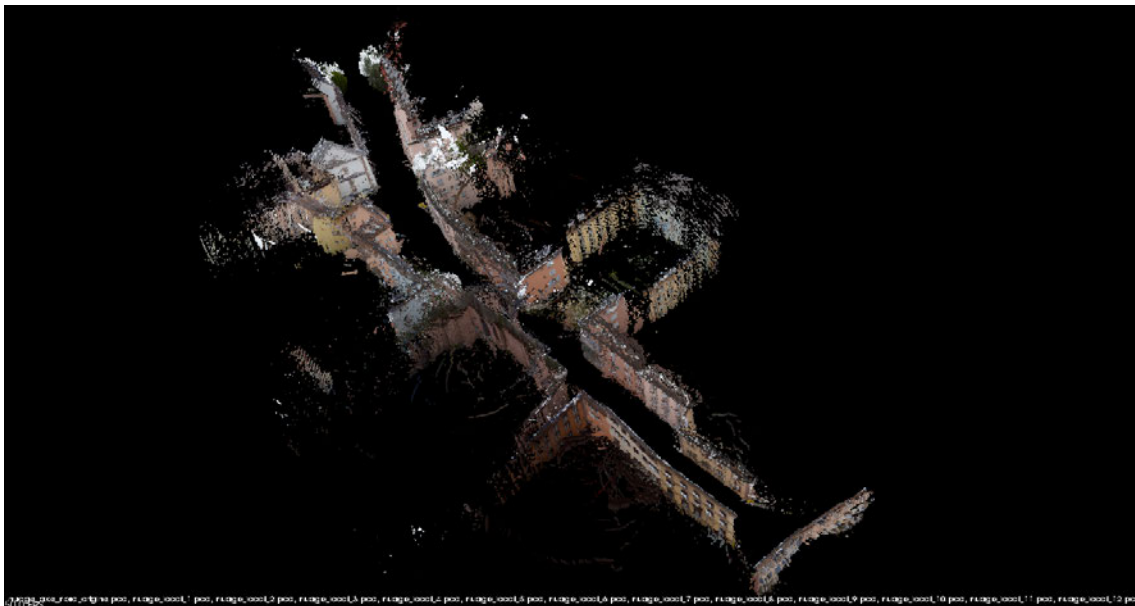
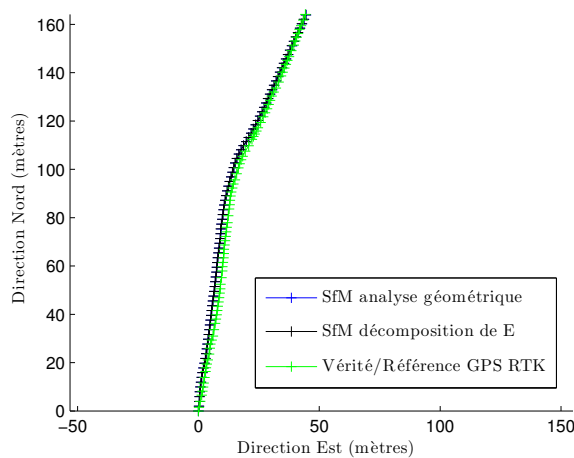


FIGURE 4.24 – Modèle 3D global pour la scène I de Belfort.

Les résultats obtenus par la décomposition de la matrice essentielle ou par notre méthode

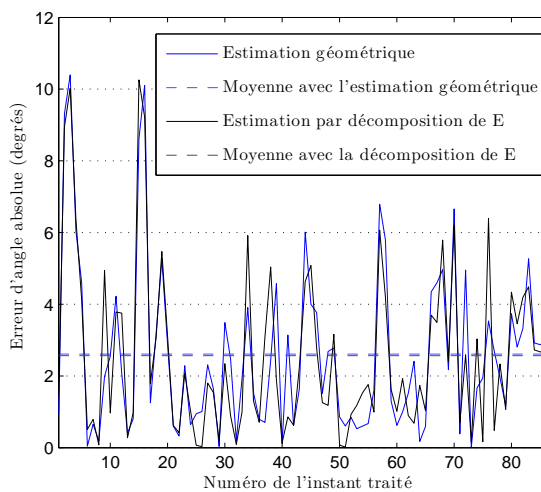


(a) Trajectoires vérité et estimées.

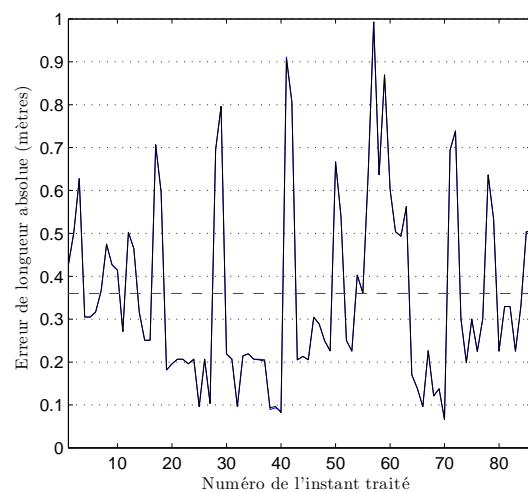
Estim. pose	Argument (degrés)			
	Moy	Méd	É-type	Max
Décomp. E	2,56	1,80	2,36	10,26
Géométrique	2,61	2,03	2,32	10,40

Estim. pose	Module (mètres)			
	Moy	Méd	É-type	Max
Décomp. E	0,36	0,30	0,22	0,99
Géométrique				

(b) Mesures d'erreur des vecteurs.



(c) Erreur angulaire absolue entre chaque position successive.



(d) Erreur de longueur absolue entre chaque position successive.

FIGURE 4.25 – Évaluation de l'odométrie visuelle sur la scène I de Belfort par rapport aux données vérité RTK.

géométrique sont très proches. Pour la séquence I, l'allure du trajet estimé correspond bien à la vérité, la ligne étant presque droite elle ne présente pas de difficulté particulière. La longueur totale du trajet est bien respectée, les points finaux sont proches de la vérité à 23cm pour la méthode d'estimation de la pose par décomposition de E et 36cm pour notre méthode géométrique.

L'erreur d'angle peut atteindre 10° avec une moyenne de $2,56^\circ$ pour la méthode par décomposition de la matrice essentielle et de $2,61^\circ$ avec la méthode géométrique proposée.

La forme de la boucle de la séquence O est respectée avec les estimations d'odométrie visuelle mais une légère déviation apparaît à la fin : nous relevons un décalage de 5,53m avec la méthode géométrique proposée. L'erreur moyenne des rotations sur la séquence O est deux fois plus grande que pour la séquence I : elle est de $4,39^\circ$ avec l'estimation par décomposition de E et de $4,83^\circ$ avec l'estimation géométrique proposée. Cette moyenne plus élevée est surtout due



FIGURE 4.26 – Modèle 3D global pour la situation O de Belfort.

à des pics d'erreur supérieurs allant parfois jusqu'à 20° , des pics généralement présents dans les virages, ainsi que la dernière portion du parcours où le décalage avec la vérité commence à être important.

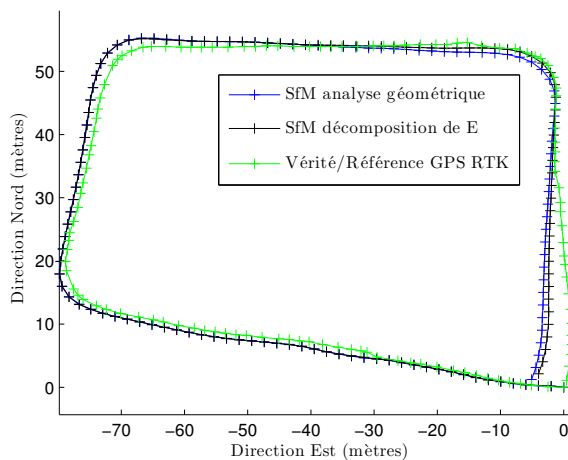
La méthode d'odométrie visuelle présentée donne globalement une précision inférieure à 3° pour les rotations, mais dévie un peu dans les virages ou quand le véhicule se décale.

4.5/ CONCLUSION

Dans cette section, nous avons proposé une méthode de mise en correspondance par graphe 3D du contenu de deux vues d'une même scène pour l'estimation d'un nuage de points 3D. Nous l'avons appliquée dans le contexte d'un système stéréoscopique fisheye à caméras en niveaux de gris et dans le contexte d'un système mono-caméra couleur en mouvement. La méthode est inspirée du travail de [Forstmann et al., 2004] et a été évaluée sur des séquences d'images de synthèse pour définir le sous-espace des paramètres sur lequel elle montre ses meilleurs résultats.

Après avoir défini un graphe 3D 3 lignes sur ce sous-espace des paramètres, nous l'avons appliqué à des séquences vidéo acquises en conditions réelle dans les villes de Belfort et Lille. Nous avons montré comment peut varier la densité des nuages de points en fonction de certaines conditions de l'acquisition et comment cette densité dégradée pouvait modifier, par exemple, l'estimation de la topologie de la rue telle que sa largeur.

Pour terminer, nous proposons une méthode de fusion de nuages de points locaux fondée sur une odométrie visuelle exploitant notre méthode de calibrage géométrique. Nous avons validé cette fusion sur les séquences d'images de Belfort et avons évalué la précision de l'algorithme d'odométrie visuelle mis en place en comparant les positions estimées à une vérité terrain

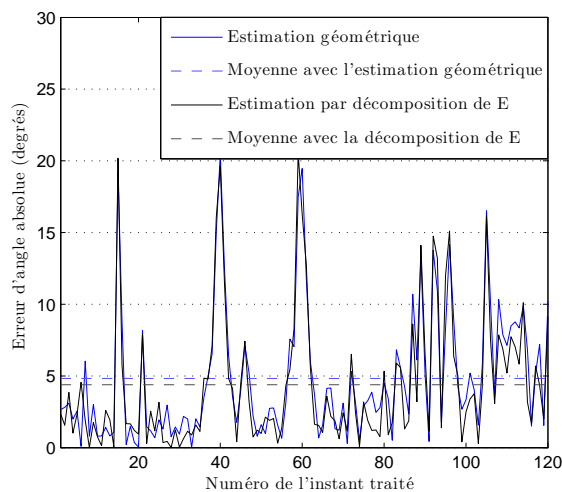


(a) Trajectoires vérité et estimées.

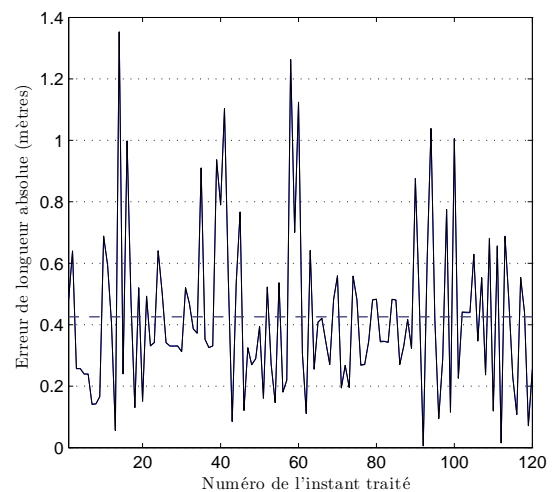
Estim. pose	Argument (degrés)			
	Moy	Méd	É-type	Max
Décomp. E	4,39	2,48	4,66	20,53
Géométrique	4,83	3,22	4,54	20,50

Estim. pose	Module (mètres)			
	Moy	Méd	É-type	Max
Décomp. E	0,43	0,35	0,26	1,35
Géométrique				

(b) Mesures d'erreur des vecteurs.



(c) Erreur angulaire absolue entre chaque position successive.



(d) Erreur de longueur absolue entre chaque position successive.

FIGURE 4.27 – Évaluation de l'odométrie visuelle sur la scène 0 de Belfort par rapport au données vérité GPS RTK.

obtenue par un récepteur GPS RTK Magellan ProFlex 500.

Dans la suite et pour terminer ce mémoire de thèse, le dernier chapitre est consacré à l'usage des nuages de points 3D fusionnés pour des applications de localisation GNSS.

4.6/ ANNEXE : ESTIMATION DE LA LARGEUR DE LA RUE

En l'absence de vérité pour les scènes réelles il est impossible d'évaluer précisément les résultats. Il est toutefois envisageable de mesurer des distances entre points spécifiques des scènes modélisées. Des grandeurs représentatives peuvent être la hauteur des structures ou la largeur des rues. Nous proposons d'évaluer l'estimation de la largeur de la rue à partir du nuage de points en la comparant avec la largeur vérité que nous extrayons manuellement des données cartographiques et cadastrales disponibles par exemple sur la plate-forme Géoportail de l'IGN.

La précision de cette vérité est légèrement inférieure au mètre.

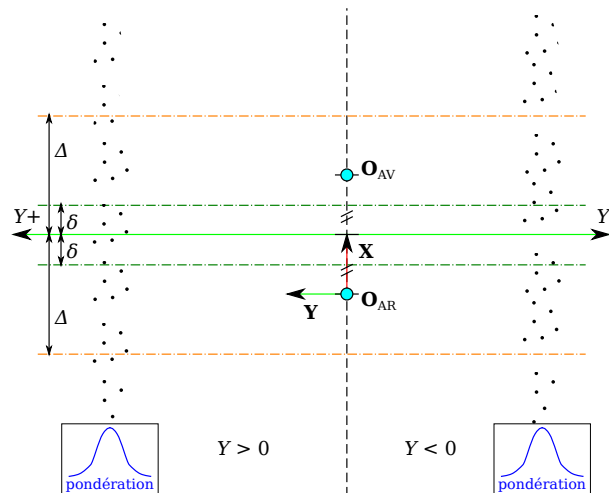


FIGURE 4.28 – Mesure de la largeur de la rue dans un nuage de points local.

La méthode d'estimation que nous proposons ne s'applique qu'à des segments de rue en ligne droite. La figure 4.28 montre le principe proposé. Les grandeurs sont exprimées en mètres (le modèle est construit à l'échelle). Le repère du modèle est centré sur la caméra de référence (caméra arrière dans le cas stéréoscopique) avec l'axe \mathbf{X} en direction du mouvement (ou de la caméra avant dans le cas stéréoscopique). \mathbf{X} est dans la direction du déplacement et par conséquent dans la direction de la rue. L'axe \mathbf{Y} est orthogonal à \mathbf{X} et orienté dans le sens indirect. L'axe \mathbf{Z} est orienté selon la verticale et représente la hauteur depuis le repère de construction (l'origine n'est pas au niveau du sol).

Le nuage de points 3D utilisé pour mesurer la largeur de la rue contient des données bruitées. Mesurer la largeur de la rue se base donc sur une analyse statistique des positions des points dans une bande centrée entre les caméras :

1. Ne garder que les points vérifiant $0 < Z \leq 10$ pour limiter le risque de considérer des points aberrants (les évaluations de cartes des distances des simulations montrent que les points bas, hauts et propagés dans le ciel sont les moins précis).
2. Considérer les points à l'intérieur d'une bande selon \mathbf{X} autour de l'axe parallèle à \mathbf{Y} et situé au milieu des deux caméras.
3. Séparer l'ensemble des points à $Y < 0$ (\mathcal{Y}_-) de l'ensemble des points à $Y > 0$ (\mathcal{Y}_+).
4. Calculer \widehat{Y}_- et \widehat{Y}_+ les moyennes pondérées des Y des points respectivement du côté négatif ($\in \mathcal{Y}_-$) et positif ($\in \mathcal{Y}_+$). Les pondérations sont faites selon des gaussiennes centrées sur les intervalles des distributions triées de \mathcal{Y}_- et \mathcal{Y}_+ , et dont les écarts-types sont $\sigma_- = \frac{\text{Card}(\mathcal{Y}_-)}{6}$ et $\sigma_+ = \frac{\text{Card}(\mathcal{Y}_+)}{6}$. Ces réglages correspondent à étaler 99% de l'énergie des gaussiennes sur les ensembles. Le but de ces pondérations est d'arriver à une mesure robuste au bruit.
5. Mesurer la largeur entre les bâtiments séparés par la rue $l = \widehat{Y}_+ - \widehat{Y}_-$.

Dans les évaluations présentées, on fixe $\delta = 0,5\text{m}$.

APPLICATION : AMÉLIORATION DE LA LOCALISATION GNSS

Les nuages de points obtenus à partir des travaux de la thèse peuvent être utilisés à diverses applications telles que la modélisation et la cartographie de nuages globaux. Ce chapitre montre l'utilisation des modèles 3D pour l'application de localisation CAPLOC. Il faut dans un premier temps identifier les plans des scènes, puis les exploiter pour estimer les réflexions des signaux GNSS.

Sommaire

5.1	Introduction	112
5.2	Informations et données satellitaires	113
5.2.1	Heures	113
5.2.2	Données élémentaires	114
5.2.3	<i>Dilution Of Precision</i>	115
5.3	Mise en application du modèle 3D estimé pour GNSS	116
5.3.1	Effet multi-trajet en milieu urbain	116
5.3.2	État de l'art sur l'étude des signaux NLOS en milieu urbain	117
5.3.3	Modèle d'erreur et correction des pseudodistances	122
5.4	Extraction des plans	125
5.4.1	État de l'art des méthodes de <i>plane fitting</i>	125
5.4.2	<i>Plane fitting</i> en 3D basé sur le RANSAC	126
5.4.3	<i>Plane fitting</i> par transformée de Hough 2D sur cartographie	127
5.4.4	Évaluation des informations extraites sur scènes réelles	131
5.5	Représentation des satellites dans la partie vision	134
5.5.1	Projection des satellites et rayons GNSS sur la trace au sol	134
5.5.2	Projection des satellites dans les images fisheye	134
5.5.3	Création de masques de visibilité d'après les plans détectés	135
5.6	Évaluation en conditions réelles	135
5.6.1	Procédure d'évaluation	136
5.6.2	Scènes et positions évaluées	137
5.6.3	Position améliorée avec correction des pseudodistances	137
5.7	Conclusion	141

5.1/ INTRODUCTION

Les travaux présentés dans ce chapitre concernent la partie localisation du projet CAPLOC. Nous allons décrire comment il est possible d'utiliser le nuage de points 3D généré grâce à la méthode décrite dans le chapitre précédent afin d'améliorer le positionnement GNSS (*Global Navigation Satellite System*) d'un véhicule qui se déplace en milieu urbain.

Plusieurs systèmes GNSS coexistent dans le monde : le système américain GPS (*Global Positioning System*), le système russe GLONASS, Beidou déployé par la Chine ... et le futur système européen Galileo.

Pour chacun de ces systèmes, des satellites sont positionnés en orbite autour du globe terrestre, organisés dans des plans orbitaux définis par rapport à l'équateur, et leurs positions sont connues à chaque instant. De faibles décalages d'orbite ou de vitesse de rotation peuvent apparaître, des mises à jour des éphémérides doivent donc être effectuées dans les almanachs. La figure 5.1 montre la constellation des satellites du système GPS. La localisation du récepteur se fait par trilatération à partir des distances estimées entre le récepteur et chaque satellite reçu. La position est donnée sous la forme de coordonnées géographiques c'est-à-dire par la latitude et la longitude pour la position en deux dimensions à la surface de la Terre, accompagnées de l'altitude par rapport au niveau de la mer pour la position 3D. Les satellites émettent leurs signaux en permanence, ces derniers ne sont pas synchrones car ils atteignent l'antenne du récepteur après des parcours de longueurs différentes. Par conséquent, 4 inconnues sont à estimer : 3 inconnues de position et une inconnue temporelle permettant la synchronisation. Un minimum de 3 satellites doivent donc être reçus pour estimer une position 2D et de 4 satellites pour accéder à une position 3D. Les distances estimées entre le satellite et le récepteur GNSS sont appelées les pseudodistances.

La section 5.2 liste les données principales utilisées par le récepteur GNSS.

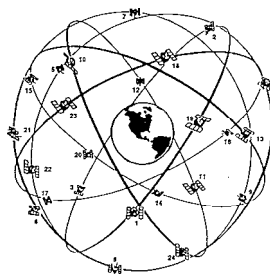


FIGURE 5.1 – Constellation des satellites de localisation GPS avec les plans orbitaux. Source : [Nav, 1996].

Dans le cadre des recherches pour le projet CAPLOC nous avons travaillé uniquement avec la constellation GPS. Mais le principe mis en place est applicable à tous les systèmes GNSS. Les satellites sont à une distance d'environ 20000 km de la surface de la Terre. Dans le contexte de la localisation GNSS en milieu urbain le positionnement est peu précis à cause du masquage des satellites par les structures urbaines comme l'explique la section 5.3.1. Le but est d'estimer

la correction à appliquer pour augmenter la précision du service sans perdre en disponibilité. Le problème du masquage des rayons GNSS en milieu urbain étant identifié, des recherches ont été effectuées pour corriger les mesures. La section 5.3 présente certains de ces modèles de correction. Pour estimer le retard des rayons GNSS réfléchis il faut se baser sur la structure 3D de l'environnement. Le modèle obtenu avec les méthodes décrites dans ce manuscrit est un nuage de points. Il faut en extraire les formes des structures et des informations sémantiques. Un maillage triangulaire de Delaunay pourrait être généré pour obtenir un modèle de surfaces. Le problème des nuages de points estimés est qu'ils sont beaucoup trop denses pour correctement estimer les surfaces et obtenir un modèle assez lisse. Par exemple, un mur est représenté par un ensemble de points mal alignés, des points disposés en parallèle les uns des autres selon la direction du mur ne permettent pas de trouver un bon maillage. La solution qui apparaît la plus intéressante est d'estimer les plans correspondants aux ensembles de points des murs. Cette étape de *plane fitting* est détaillée en section 5.4. À partir des plans il est possible de mesurer les réflexions des rayons GNSS en positionnant les satellites dans le modèle. La section 5.5 de ce chapitre explique cette étape. Des évaluations sont montrées en section 5.6.

5.2/ INFORMATIONS ET DONNÉES SATELLITAIRES

On distingue 3 types de données :

- L'heure nécessaire à la synchronisation.
- Les données propres à chaque satellite reçues par le récepteur.
- Les mesure globales par rapport à l'ensemble des données satellite reçues.

Les deux premiers éléments de la liste sont utilisés pour l'interprétation du signal envoyé par chaque satellite. Le dernier élément regroupe les données calculées par le récepteur pour estimer la position du véhicule.

5.2.1/ HEURES

L'heure GPS est composée du numéro de semaine GPS et du temps ITOW (*Interval Time Of Week*, Intervalle Temps de la Semaine) en secondes et millisecondes. D'après [Nav, 1996] (parties 3.4 et 11.4), la précision absolue de l'heure GPS indiquée par le récepteur est environ de 200ns, l'écart augmente d'environ 20ns en situation dynamique, et il est possible d'arriver à des précisions relatives entre récepteurs 10 fois meilleures.

Puisque nous proposons d'exploiter les images de l'environnement pour corriger son effet sur l'estimation de position, il est important de parvenir à synchroniser les données GNSS reçues et les images acquises. Pour cela, nous choisissons de travailler en ITOW qui présente les avantages suivants :

- L'ITOW s'exprime en secondes avec des décimales (pas de distinction heures, minutes...), plus facile pour comparer les temps car il s'agit d'une évaluation directe.

- La conversion de la date et heure UTC $H_{UTC}:M_{UTC}:S_{UTC}$ (*Coordinated Universal Time*, Temps Universel Coordonné) en temps ITOW T_{ITOW} est plus facile que l'inverse :

$$T_{ITOW} = S_{UTC} + 60(M_{UTC} + 60(H_{UTC} + 24.J)) + S_{leap}$$

où J est le numéro du jour dans la semaine (de 0 à 6 pour représenter les jours dans l'ordre en commençant au dimanche à 0), et S_{leap} sont les *leap seconds*¹.

L'heure GPS est basée sur une horloge atomique (c'est ce qui se fait de plus précis à l'heure actuelle). Elle dérive peu avec le temps, contrairement à l'heure UTC calée sur la vitesse de rotation de la Terre et qui a tendance à ralentir, d'où la nécessité de secondes intercalaires (*leap seconds*) pour permettre une synchronisation à la seconde entre les temps UTC et ITOW.

5.2.2/ DONNÉES ÉLÉMENTAIRES

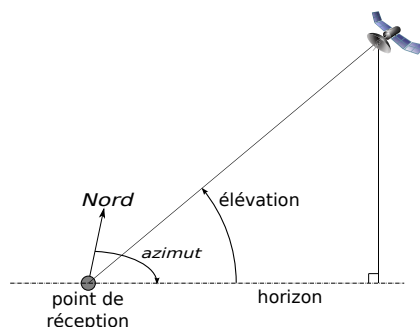


FIGURE 5.2 – Mesure de l'azimut et élévation d'un satellite à un point de réception, en fonction de l'horizon au point de réception.

L'azimut (*azimuth* en anglais) est l'angle du satellite depuis le Nord géographique dans le plan horizontal, par rapport au récepteur. Il augmente selon le sens de rotation horaire. Il s'exprime en degrés dans l'intervalle $[0; 360[$.

L'élévation d'un satellite est son angle dans le plan vertical depuis l'horizon à la hauteur du récepteur. Elle est exprimée en degrés et appartient à l'intervalle $[0; 90]$. L'azimut et l'élévation d'un satellite sont illustrés dans la figure 5.2.

Le C/N_0 , C/N_0 , CN_0 ou CN_0 (*Carrier-to-receiver Noise density*) est le rapport entre la puissance de la porteuse reçue et la densité de puissance du bruit du récepteur. Il s'exprime en dBHz. En environnement ouvert, un CN_0 élevé est signe que le rayon est très bien reçu et que le signal est de bonne qualité. Un CN_0 faible indique que le signal reçu a subi des interférences et a potentiellement rencontré des obstacles amenant à des atténuations, di ractions ou réflexions. Le lecteur trouvera dans [Joseph, 2010] de plus amples informations sur le SNR (*Signal-to-Noise Ratio*, rapport signal à bruit) et le CN_0 .

1. www.usno.navy.mil/USNO/time/master-clock/leap-seconds

La PR (*PseudoRange*, pseudodistance) est l'estimation de la distance entre le satellite et le récepteur utilisée pour estimer la position GNSS. Elle est exprimée en mètres. Elle est calculée par le récepteur à partir du temps mis par le signal pour arriver du satellite au récepteur et de la vitesse de la lumière. C'est en fait une distance géométrique apparente. Elle inclut les retards accumulés par des décalages d'horloges (les rayons des différents satellites n'arrivent pas en même temps) et en traversant l'atmosphère (la ionosphère de plusieurs centaines de kilomètres d'épaisseur qui contient de nombreuses particules, et la troposphère où sont les nuages avec environ 50km d'épaisseur). Les PR des satellites reçus servent à l'estimation de la position par trilatération. Le problème de la mesure des PR avec le décalage des horloges est abordé dans [Rao et al., 2012].

Le numéro PRN (*PseudoRandom Noise number*) est un numéro d'identification du satellite. Chaque satellite d'une constellation a un PRN différent, les PRN peuvent être modifiés notamment lors du remplacement de certains satellites.

5.2.3/ Dilution Of Precision

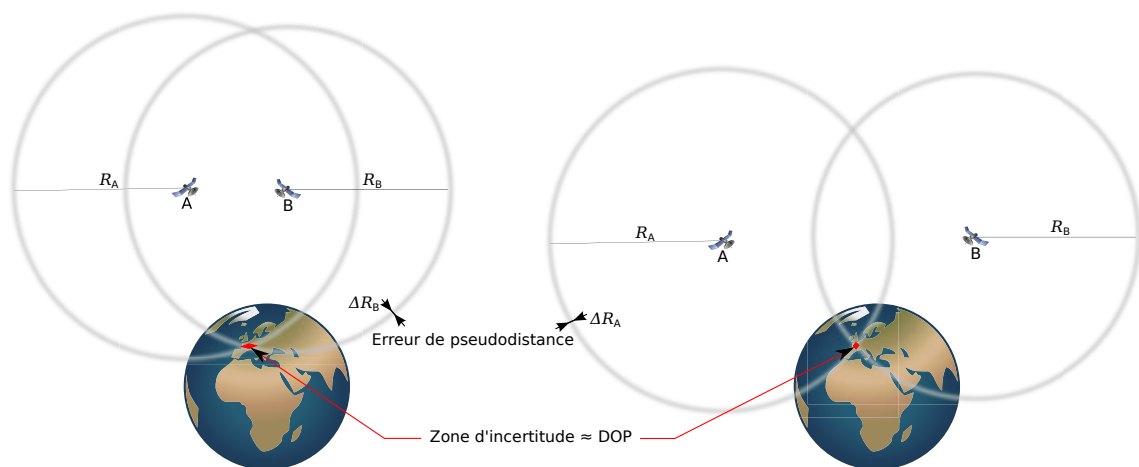


FIGURE 5.3 – Dilution Of Precision de la localisation, illustrée pour deux satellites.

Le GDOP *Geometric Dilution Of Precision*, est une mesure de la marge d'erreur de la localisation due à la répartition et au positionnement relatif géométrique des satellites perçus par le récepteur. Elle est sans unité. En environnement ouvert, un GDOP faible implique une bonne répartition des satellites observés et une grande précision de la position et de l'heure estimée, tandis qu'un GDOP élevé signifie le contraire. Dans la figure 5.3, la configuration de gauche a un grand GDOP et la configuration de droite a un petit GDOP. On voit à gauche que la position estimée peut prendre plus de valeurs possibles, elle est donc moins certaine. Le GDOP varie dans le temps avec les mouvements des satellites et dépend des satellites reçus. On peut distinguer plusieurs mesures DOP selon la dimension de l'espace considérée. Chaque DOP est le rapport de variance de l'estimation de la composante considérée par rapport à la variance de l'erreur de mesure des pseudodistances utilisées :

- HDOP *Horizontal Dilution Of Precision*, selon la variance de l'estimation de la position 2D par rapport à l'horizontale (qui dépend du nord et de l'est),
- VDOP *Vertical Dilution Of Precision*, par rapport à la variance de l'estimation de la position par rapport à la verticale (altitude),
- PDOP *Position Dilution Of Precision*, par rapport à la variance de l'estimation de la position 3D, $PDOP = \sqrt{HDOP^2 + VDOP^2}$,
- TDOP *Time Dilution Of Precision*, par rapport à la variance de l'estimation de l'heure,
- GDOP *Geometric Dilution Of Precision*, par rapport aux critères complets de position et du temps, $GDOP = \sqrt{PDOP^2 + TDOP^2}$.

Généralement seule l'information PDOP ou GDOP est utilisée. Le PDOP sert pour renseigner sur la précision à laquelle s'attendre pour le calcul de la position 3D (pour une position 2D horizontale c'est le HDOP qui est plus représentatif). Le GDOP est global pour la position et l'heure. Pour des détails sur le calcul des DOP, le lecteur peut se référer à [Langley, 1999].

5.3/ MISE EN APPLICATION DU MODÈLE 3D ESTIMÉ POUR GNSS

Cette section montre comment utiliser des informations sur l'environnement local pour améliorer la localisation GNSS, et positionne la solution proposée basée sur le modèle 3D généré de manière dynamique par la vision fisheye.

5.3.1/ EFFET MULTI-TRAJET EN MILIEU URBAIN

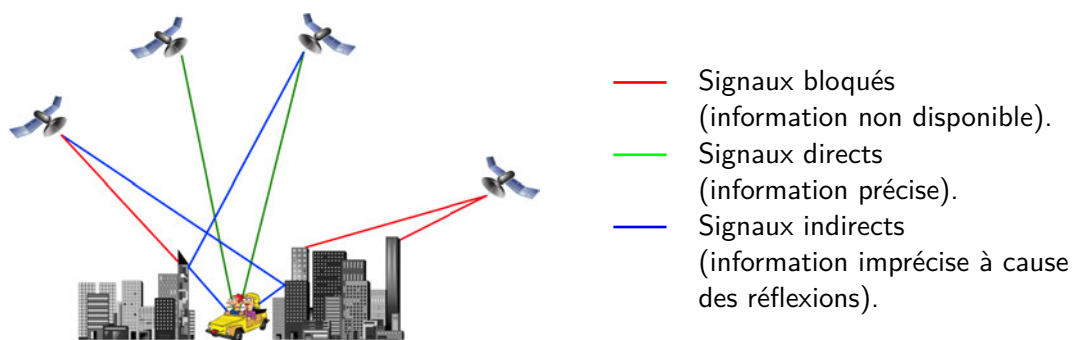


FIGURE 5.4 – Effet multi-trajets suite aux réflexions des rayons sur les structures urbaines.

En milieu urbain les signaux satellitaires reçus par un récepteur GNSS subissent des réflexions sur les structures urbaines (les bâtiments) conduisant à une information de position imprécise. La figure 5.4 illustre un canyon urbain et différents cas d'observation des signaux GNSS. Les satellites ou signaux vus de manière directe sont dits LOS (*Line Of Sight*, en ligne de mire), les satellites ou signaux perçus de manière indirecte ou réfléchi sont dits NLOS (*Non Line Of Sight*, pas en ligne de mire). Et ceux qui ne sont pas reçus sont dits bloqués. Comme le montre la figure 5.4, le même signal peut se propager par plusieurs chemins et finalement être reçu

plus d'une fois avec des déphasages. C'est l'effet multi-trajets, et il peut causer différents types d'interférences. Le multi-trajets est difficile à modéliser pour correspondre convenablement à la réalité. Ainsi, dans les travaux présentés, nous nous intéressons aux signaux indirects sans tenir compte de l'effet multi-trajets.

Un signal indirect est retardé car il parcourt un chemin plus long que la distance réelle entre la source et le récepteur. Ce retard de pseudodistance conduit à une imprécision en étant répercuté dans le calcul de la position. Localement, ce sont les réflexions qui causent le plus de retards de pseudodistances.

La source du signal retardé est donc un satellite NLOS, et le signal est reçu avec une mauvaise qualité. Cela se répercute généralement par un CN0 faible (en tout cas en champ ouvert, voir section 5.2.2). Des auteurs comme [Marais, 2002] ou [Viandier, 2011] tentent d'estimer une distinction entre les satellites reçus LOS et NLOS par application d'un seuil de CN0. D'après les résultats expérimentaux de [Marais, 2002] on peut utiliser un seuil de 40 dBHz, [Viandier, 2011] utilise un seuil de 42 dBHz. Ils considèrent qu'un CN0 inférieur au seuil a plus de chances de correspondre à un signal NLOS, et si le CN0 est supérieur ou égal au seuil alors le signal est plus probablement LOS. Notons que le choix du seuil optimal peut légèrement différer en fonction des conditions environnementales ainsi que de l'antenne et du récepteur utilisés ; ces auteurs ont utilisé un récepteur Septentrio PolaRx. Ce seuil n'est pas un critère très fin, même s'il est choisi d'après des expérimentations spécifiques aux configurations étudiées.

Un signal qui passe de l'état LOS à NLOS voit son CN0 diminuer, et plus il subit de réflexions plus il est atténué. De plus, un signal issu d'un satellite de faible élévation a plus de chances d'être perturbé par les structures urbaines. Ces observations peuvent aider à détecter les changements d'état de masquage des satellites dans le temps.

5.3.2/ ÉTAT DE L'ART SUR L'ÉTUDE DES SIGNAUX NLOS EN MILIEU URBAIN

Dans l'objectif d'améliorer la précision de la localisation GNSS du véhicule, une solution peut être d'ajouter des capteurs tels qu'un odomètre ou une centrale inertielle. Ce n'est pas l'objet de l'étude, l'idée est de se passer de tels capteurs parfois coûteux et nécessitant un calibrage, donc difficile à déployer à grande échelle. On peut distinguer les contributions de l'état de l'art selon plusieurs critères :

- Les études théoriques ou appliquées à des cas réels.
- Les auteurs qui procèdent à la détection des signaux NLOS sans corriger les retards de pseudodistance :
 - Le plus souvent ils se contentent d'exclure les signaux NLOS du calcul de position.
 - D'autres exploitent l'information avec des procédures différentes.
- Les auteurs qui cherchent à corriger le retard des pseudodistances.
- Une autre distinction est l'utilisation d'informations 3D sur les structures situées autour pour tester plusieurs positions.

Les auteurs utilisent souvent des modèles 3D ou des données similaires pré-enregistrés. Ce qui est proposé dans les travaux présentés du projet CAPLOC est de générer les données 3D sur le terrain. Les intérêts de l'approche proposée sont l'indépendance à des enregistrements coûteux et difficiles à actualiser, et surtout l'absence du besoin de recalage de la position dans le modèle 3D car elle est connue à la génération de ce dernier.

5.3.2.1/ MODÈLE DE PROPAGATION DES SIGNAUX GNSS EN MILIEU URBAIN

[Ercek et al., 2005] proposent une étude théorique sur les réflexions et la diffraction des signaux GNSS contre les structures urbaines. En réalité les signaux émis se propagent selon un cône de plus en plus large avec un maximum de puissance au centre, et les surfaces de réflexion ne réfléchissent ni ne dispersent avec les mêmes paramètres. Les auteurs traitent des données simulées dans le cas d'un canyon urbain en ligne droite, avec des bâtiments de différentes hauteurs. Un signal de la même source peut être reçu plusieurs fois au même moment avec des puissances différentes. Généralement on tient compte de la réception à la plus forte puissance que l'on considère comme le rayon ponctuel géométrique pour simplifier le problème. En tenant compte de tout cela, [Ercek et al., 2005] formulent des mesures de probabilité que le signal soit NLOS et qu'il présente au moins une réflexion ou un point de diffraction, en fonction des conditions géométriques de l'environnement et de la puissance des signaux reçus.

5.3.2.2/ AMÉLIORATION DE LA LOCALISATION PAR EXCLUSION NLOS

Certains auteurs montrent que si on ignore les satellites NLOS on peut gagner en précision du positionnement. Pourtant, ignorer des satellites nuit au GDOP et à la qualité de l'estimation. Le gain de précision par élimination des signaux NLOS peut compenser la perte de certitude géométrique si assez de satellites bien répartis sont LOS. Par ailleurs le fait de perdre des satellites peut réduire la disponibilité du service. S'il subsiste moins de 4 satellites après élimination des NLOS il n'est plus possible d'estimer la position 3D. Cette procédure peut être efficace pour l'utilisation simultanée de plusieurs constellations GNSS (par exemple GPS + GLONASS + Galileo), ce qui augmente le nombre total de satellites observés et réduit donc les risques de fortement dégrader le GDOP et de rendre le service indisponible.

[Peyret et al., 2011, Peyret et al., 2014] présentent des travaux effectués dans le cadre du projet CityVIP². Leur objectif est de détecter les satellites NLOS. Ils utilisent un modèle 3D pré-enregistré pour générer une vue de caméra virtuelle à la position du véhicule, avec la projection de la position des satellites reçus. La vue virtuelle sert à identifier les satellites masqués par les structures affichées. Cette méthode implique de positionner précisément le récepteur dans le modèle 3D, pour cela elle utilise l'odométrie du compteur de la voiture et un gyroscope.

Dans [Marais et al., 2012, Marais et al., 2014] les auteurs font une exclusion des signaux NLOS et montrent des résultats pour des acquisitions dans le cadre du projet CAPLOC.

2. www.projet_cityvip.byethost33.com

Les satellites NLOS sont identifiés avec la méthode de [Attia, 2013] à partir d'un masquage par segmentation des structures dans des vues fisheye (caméras fisheye orientées vers le ciel). Les auteurs ont choisi d'intégrer un filtre de Kalman dans les évaluations, avec et sans exclusion NLOS. Dans [Marais et al., 2014] les auteurs comparent la localisation avec cette méthode d'identification des satellites NLOS et une méthode basée sur un seuillage CN0, et constatent de meilleurs résultats avec cette dernière. L'inconvénient du seuillage CN0 est que le seuil est déterminé de manière empirique. Avec un seuil CN0 bien choisi ils arrivent à réduire l'imprécision moyenne de localisation pour une séquence de 13m à 4,6m. Un seuil CN0 mal choisi peut en revanche réduire la précision et diminuer la disponibilité du service. La solution basée sur le masquage n'améliore la précision que de 1m.

[Obst et al., 2012] proposent plusieurs manières d'identifier les satellites NLOS pour les exclure du calcul de position. Leurs solutions sont basées sur un modèle 3D pré-enregistré à partir de cartographies et de cartes d'élévation. Ils se basent sur un lancer de rayons pour modéliser les signaux GNSS comme des sources ponctuelles directes et tester s'ils sont bloqués. Pour une séquence, ils arrivent d'une précision initiale moyenne de 14m à une précision de 5,2m sans filtre supplémentaire.

[Kumar et al., 2014] n'utilisent pas de modèle 3D pré-enregistré pour la détection des signaux NLOS mais font appel à un télémètre laser pour le construire. Pour cela la méthode proposée consiste à projeter les satellites dans une image aplanie des structures autour de sa position. Pour des positions fixes, les auteurs passent d'une imprécision comprise entre 1 et 6m à une imprécision inférieure au mètre.

5.3.2.3/ AMÉLIORATION DE LA POSITION SANS CORRIGER LES PSEUDODISTANCES

Rejet remplacé par l'utilisation de l'information NLOS dans une étape de lissage

[Bourdeau et al., 2012] utilisent un modèle 3D pré-enregistré pour déterminer les satellites NLOS. Les auteurs implémentent un filtre de Kalman sur les pseudodistances amélioré par la prise en compte de cette information pour lisser la position estimée. La méthode proposée améliore la position et est validée sur des simulations.

[Sahmoudi et al., 2014] étendent le filtre de Kalman au VDLL (*Vector tracking Delay-Locked Loops*) avec des mesures statistiques. L'idée est de suivre la phase des signaux et de pouvoir maintenir le suivi des signaux faibles et estimer les variations des retards d'horloge qui augmentent pour les signaux indirects. Ils améliorent l'erreur de position d'une centaine de mètres à une dizaine de mètres pour une séquence au centre-ville de Toulouse.

Pondération au lieu d'un rejet pur et dur des signaux NLOS

[Tay et al., 2013] améliorent la localisation en pondérant la contribution des satellites NLOS.

Les auteurs exploitent des images fisheye acquises en mobilité dans le cadre du projet CAPLOC. Les satellites sont projetés sur les images fisheye acquises par des caméras fisheye orientées en direction du ciel. Les auteurs utilisent l'élévation du satellite pour voir s'il est masqué (masque développé par [Attia, 2013]) et son CN0 pour déterminer la pondération. Cette procédure semble arriver à de meilleurs résultats qu'une exclusion des signaux NLOS, sans perdre en disponibilité du service. Pour la séquence traitée, l'erreur de position moyenne est de 15m sans corrections et passe à un peu plus de 4m avec la méthode proposée.

Technique de localisation par *shadow matching*

Cette technique est décrite dans [Wang et al., 2013, Groves et al., 2015]. Elle opère en identifiant des zones de réception directe ou indirecte du signal d'un satellite en une position donnée. Les auteurs se basent sur un modèle 3D pré-enregistré. Le principe de la méthode proposée consiste à interpoler la position estimée en fonction des masques d'occultation des satellites dans le lieu pour qu'elle corresponde bien à l'état de réception (LOS, NLOS ou bloqué) et au CN0 réel de chaque satellite. Ils considèrent une grille de positions possibles autour de la position GNSS initiale. Le score d'une position se mesure par la comparaison des visibilitées observées des satellites (LOS, LOS avec faible CN0, ou NLOS) et des visibilitées prédites (selon un modèle géométrique de diffraction et de réflexion dans la modélisation 3D du lieu). Ils montrent une amélioration du positionnement GNSS avec cette méthode sur des données réelles. [Wang et al., 2013] passent d'une imprécision de la position dans les rues comprise entre 2 et 3m à une imprécision inférieure à 2m. Une bonne précision permet de détecter de manière assez fiable sur quelle voie de la rue le récepteur est placé.

5.3.2.4/ RÉVISION DES PSEUDODISTANCES NLOS POUR AMÉLIORER LA POSITION

[Marais et al., 2005] proposent un modèle simple de calcul du retard de pseudodistance d'un signal à une réflexion. Cette méthode géométrique simple basée sur le lancer de rayon peut être efficace même si les signaux réels ne sont pas ponctuels comme un rayon au sens propre. Le modèle proposé est limité à la prise en compte de l'élévation. Leur modèle est aussi relaté dans [Ercek et al., 2005] et [Viandier, 2011], mais avec la prise en compte de l'élévation et de l'azimut. Des estimations sont calculées à partir de simulations.

[Viandier, 2011] propose une modélisation du bruit source d'erreurs de pseudodistances par mélange de gaussiennes et processus de Dirichlet pour les corriger. Les paramètres des modèles de bruit et de position sont estimés par filtrage particulière. L'auteur traite les dégradations suivantes :

- le multi-trajet, pour un signal perçu plusieurs fois simultanément avec différents parcours,
- la réception alternée, c'est-à-dire avec une réflexion ; le signal est affaibli,
- la diffraction et diffusion, qui survient lorsque le signal se propage dans une végétation dense ou lorsqu'il rencontre le coin d'une structure ; la perte de puissance est importante.

L'auteur montre que sa méthode aboutit à une nette amélioration de la précision de la localisation pour des mesures réelles comparées avec des positions vérités issues d'un récepteur GPS RTK. Il montre des résultats où il diminue de moitié l'imprécision de la localisation brute (entre 14 et 21m avant corrections à entre 7 et 11m après application de son filtre).

Dans [Bétaille et al., 2013, Bétaille et al., 2014], les auteurs vérifient la visibilité des satellites et formalisent la mesure du retard de pseudodistance pour une, deux ou trois réflexions. Ils considèrent la rue avec les deux côtés parfaitement parallèles à l'axe du véhicule et de hauteurs constantes. Le modèle appelé « tranchée urbaine » dispose de quatre paramètres : la largeur de la tranchée, la hauteur à gauche et à droite, et la position transversale du véhicule par rapport à la largeur. Ce modèle simplifie le problème 3D, et les paramètres sont pré-enregistrés et embarqués par le système. Les auteurs génèrent un masque de visibilité selon l'axe du véhicule, avec la distinction possible des zones à 1, 2 ou 3 réflexions et plus. Avec la correction des pseudodistances, ils montrent une amélioration de la précision de la localisation de l'ordre de 30 à 70% sur données réelles.

Certains auteurs ont des approches de correction complétées par le *shadow matching* (voir [Wang et al., 2013]). [Miura et al., 2015] cherchent à mesurer le retard de pseudodistance des rayons réfléchis par lancer de rayon dans un modèle 3D enregistré. La difficulté est qu'on ne connaît pas précisément la localisation du récepteur dans le modèle 3D. La solution proposée consiste donc à considérer un maillage de positions possibles autour de la position GPS initiale, de 50×50 mètres avec un pas de 1 mètre. Les auteurs éliminent les propositions impossibles : position sur l'emplacement d'un bâtiment, incohérente avec la position précédente, corrélation avec l'état des satellites LOS ou NLOS respectivement considérés en fonction de leur CN0 (élevé ou faible).

[Hsu et al., 2015] présentent une vérification d'intégrité entre les mesures et la projection théorique des rayons dans un modèle 3D. Ils cumulent l'idée du *shadow matching* avec le RAIM FDE (*Receiver Autonomous Integrity Monitoring, Fault Detection and Exclusion*) visant à exclure les signaux qui présentent un écart de pseudodistance trop élevé par rapport aux prévisions théoriques. Les auteurs exploitent un modèle 3D pré-enregistré. De même, ils ne connaissent pas précisément la localisation du récepteur dans le modèle 3D. Par contre au lieu de faire un maillage régulier, ils prennent des positions candidates aléatoires.

Ces auteurs calculent ensuite une vraisemblance sur les états et les mesures de retard des signaux GNSS, utilisée pour pondérer la correction et l'estimation de la position GPS finale. Leurs méthodes sont validées et donnent de bons résultats en conditions réelles pour un récepteur GPS mobile tenu par un piéton. D'après les mesures de [Miura et al., 2015], ils passent d'une erreur de localisation moyenne de plus de 34m à une erreur moyenne proche de 20m sans lissages supplémentaires, avec une stabilité plus de deux fois supérieure. [Hsu et al., 2015] mesurent une erreur de positionnement latérale par rapport aux axes des routes. Ils montrent une réduction de l'erreur latérale moyenne de plus de 20m à une erreur comprise entre 3 et 4m, cette erreur finale étant plus importante pour un récepteur entouré de bâtiments plus hauts.

5.3.3/ MODÈLE D'ERREUR ET CORRECTION DES PSEUDODISTANCES

Nous proposons de corriger les erreurs de pseudodistance des trajets indirects liées aux réflexions, principale source des retards de pseudodistances en canyon urbain. Les réflexions ont besoin de modèles plus complexes pour être prises en compte.

On appelle PR la pseudodistance. Pour un signal indirect, on écrit :

$$PR_{\text{vraie}} = PR_{\text{mesure}} - r \quad (5.1)$$

où r est le retard de pseudodistance. Ces grandeurs s'expriment en mètres.

Pour simplifier, les modèles proposés supposent des murs opposés parallèles.

5.3.3.1/ MODÈLE DE RÉFLEXION SIMPLE LIMITÉ À UNE RÉFLEXION

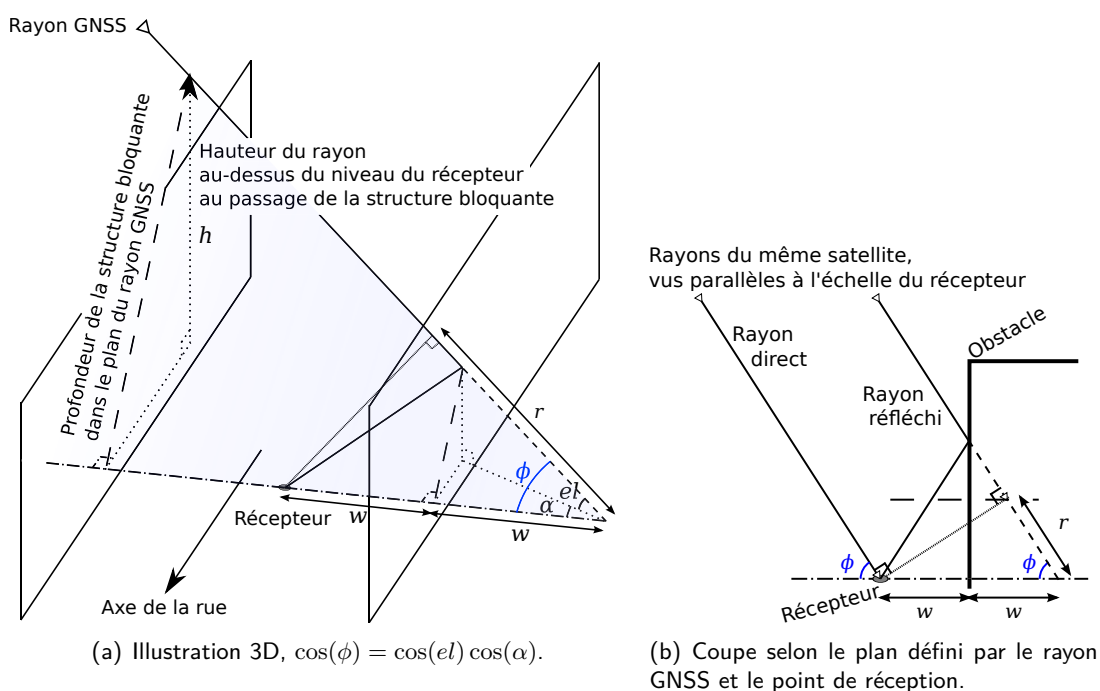


FIGURE 5.5 – Trajet du rayon GNSS avec une réflexion avant d'atteindre le récepteur.

La figure 5.5 montre le cas d'une réflexion. Le rayon réfléchi rencontre le mur opposé au mur bloquant. w est la distance de ce mur au récepteur selon sa normale. Le retard est donné par :

$$\begin{aligned} r &= 2w \cdot \cos(\phi) \\ r &= 2w \cdot \cos(el) \cos(\alpha) \end{aligned} \quad (5.2)$$

α est l'angle entre le rayon et la normale à la surface de réflexion. Pour simplifier on suppose que cette surface est plane. el est l'élévation du satellite, α dépend de son azimut (qui est référencé par rapport au Nord géographique) et de l'orientation de la normale au plan de

réflexion avec le Nord.

5.3.3.2/ MODÈLE DE RÉFLEXION À UNE, DEUX OU TROIS RÉFLEXIONS

Nous nous inspirons des travaux de [Bétaille et al., 2013] pour calculer les réflexions multiples. Au-delà de trois réflexions un signal est tellement faible qu'il n'est que rarement perçu par le récepteur. On considère donc une, deux ou trois réflexions. Si jamais le signal réel fait effectivement plus de trois réflexions alors il sera considéré comme ayant subi trois réflexions.

L'idée de [Bétaille et al., 2013, Bétaille et al., 2014] est de considérer un masque des satellites à plusieurs niveaux en fonction du nombre de réflexions que le rayon est censé subir. Pour pouvoir faire cela les auteurs simplifient le problème avec le modèle de « tranchée urbaine », voir la section 5.3.2.4. Les satellites avec une élévation plus faible rencontrent plus de réflexions.

Dans les travaux proposés nous détectons les structures urbaines avec une précision métrique. Notre modèle de représentation de la rue est plus précis que la « tranchée urbaine » pour distinguer les satellites LOS et NLOS, par exemple dans le cas où les bâtiments voisins ne font pas la même hauteur. Dans ce cas, générer un masque précis à différents niveaux en fonction du nombre de réflexion devient complexe, la méthode optimale serait d'estimer les trajets avec un lancer de rayon depuis le récepteur. Cependant, dans certains cas où par exemple le rayon bloqué a un azimut proche de l'axe de la rue, le modèle 3D local ne s'étend pas assez loin pour pouvoir calculer les lancers de rayon avec toutes les réflexions. Utiliser un modèle 3D global tenant compte des acquisitions précédentes assemblées au fur et à mesure comme présenté au chapitre 4 serait plus pertinent. Cependant, le côté de la direction dans laquelle avance le véhicule reste une zone d'ombre où les informations des structures ne sont pas encore connues. Nous proposons donc une solution intermédiaire : se baser sur le masque de visibilité pour identifier les satellites NLOS et sur le modèle de la « tranchée urbaine » pour mesurer le retard de pseudodistance. Le calcul du retard dépend des angles d'azimut et d'élévation des satellites NLOS, ainsi que des informations 3D : la distance des murs à l'axe de la rue passant par le récepteur et la hauteur de la structure bloquante. Cette solution n'est pas parfaite car elle n'est proche de la réalité que dans le cas des lignes droites. L'objectif est de montrer l'efficacité d'une rectification des pseudodistances à partir du modèle généré avec la stéréovision fisheye. La procédure proposée est la suivante :

1. Extraire les plans du modèle 3D et générer un masque de visibilité des satellites à la position du récepteur.
2. Identifier les satellites LOS et NLOS avec le masque.
3. Mesurer la distance des structures des deux côtés de la rue selon le modèle de la « tranchée urbaine ».
4. Pour les satellites NLOS, mesurer la hauteur de la structure qui le bloque et estimer le retard r pour calculer PR_{vraie} .

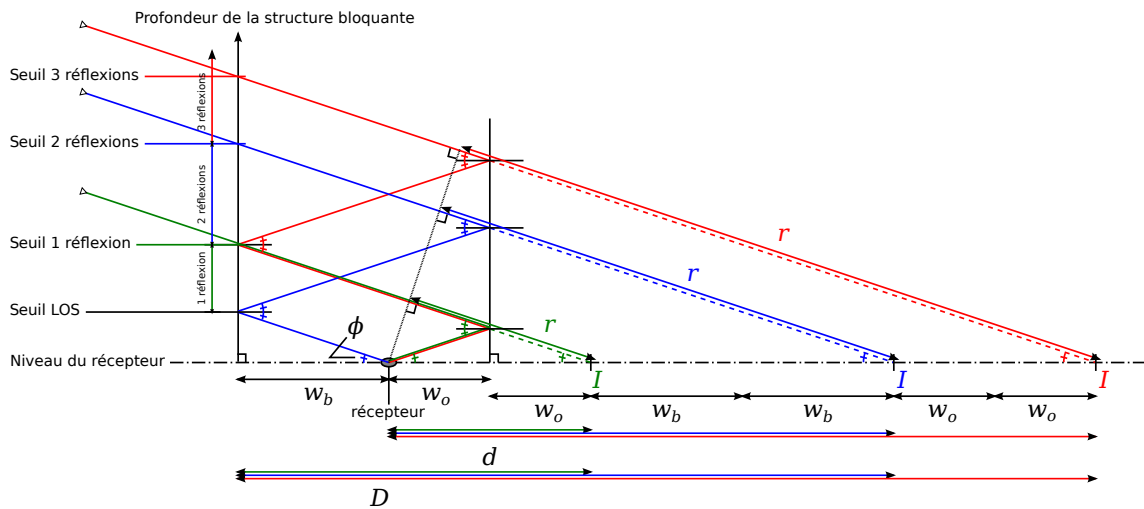


FIGURE 5.6 – Trajet des rayons réfléchis de 1 à 3 réflexions, selon la coupe dans le plan formé par le rayon GNSS et le point de réception.

On considère donc un rayon NLOS dans le modèle de la « tranchée urbaine ». Son élévation est notée el et son azimut az . L'angle α est l'angle entre l'azimut az du rayon et l'orientation de la normale au plan de réflexion par rapport au Nord, $\alpha \in [0; \frac{\pi}{2}]$. Le rayon avec l'hypothèse des réflexions multiples est représenté en figure 5.6. Soit w_b la distance entre le mur bloquant le rayon NLOS et l'axe de la rue passant par le récepteur, selon la normale au plan du mur. Soit w_o la distance entre le mur opposé et l'axe de la rue passant par le récepteur, selon la normale au mur. On exprime les seuils d'élévation qui déterminent le nombre de réflexions du rayon et le retard correspondant en fonction de la hauteur limite h du mur bloquant le signal (figure 5.5(a)). On appelle d la distance entre le récepteur et le point imaginaire I du rayon GNSS direct à la hauteur du récepteur. Soit D la distance entre I et le mur bloquant le rayon GNSS théorique. d et D dépendent de w_b et w_o , et $D = d + w_b$. On peut définir la hauteur à la position du mur bloquant pour un rayon d'une élévation donnée (et réciproquement) par (figure 5.5(a)) :

$$h = D \frac{\tan(el)}{\cos(\alpha)} \quad \text{et} \quad el = \arctan\left(\frac{h}{D} \cos(\alpha)\right) \quad (5.3)$$

Le retard lié aux réflexions est donné par $r = d \cdot \cos(\phi)$. On a par ailleurs $\cos(\phi) = \cos(el) \cos(\alpha)$. Les formules détaillées à appliquer sont inscrites dans le tableau 5.1.

$el \leq$	Réflexions	Retard $r = d \cdot \cos(el) \cos(\alpha)$
$\arctan\left(\frac{h}{2w_o + w_b} \cos(\alpha)\right)$	1	$2w_o \cdot \cos(el) \cos(\alpha)$
$\arctan\left(\frac{h}{2w_o + 3w_b} \cos(\alpha)\right)$	2	$2(w_o + w_b) \cdot \cos(el) \cos(\alpha)$
∞	3 (ou plus)	$2(2w_o + w_b) \cdot \cos(el) \cos(\alpha)$

TABLEAU 5.1 – Tableau récapitulatif des mesures du retard de pseudodistance d'un rayon NLOS dû aux réflexions.

L'extraction des plans est détaillée et évaluée en section 5.4. La méthode de mesure des distances utilisée est décrite et évaluée en section 5.4.4.2. On peut comparer plusieurs manières

de détecter l'état LOS ou NLOS d'un satellite. Avec la rectification des erreurs de PR nous allons montrer dans la section des évaluations 5.6 que l'on obtient une position plus précise.

5.4/ EXTRACTION DES PLANS

Il existe plusieurs procédures d'extraction des plans dans la littérature. Les plus rapides et efficaces exploitent des connaissances sur l'environnement étudié. Dans cette thèse, une méthode d'extraction des plans est proposée par traitement d'une trace au sol de l'environnement modélisé. On suppose qu'en environnement urbain les structures seront composées de plans verticaux. On peut donc générer une cartographie locale pour détecter les traces des murs. C'est une solution qui semble plus appropriée que de chercher à segmenter le nuage de points 3D sans utiliser les connaissances sur l'environnement. Une méthode classique basée sur l'algorithme RANSAC est comparée à la méthode proposée pour le contexte présenté. Nous exploitons les nuages de points filtrés comme décrit en section 4.4.1.

5.4.1/ ÉTAT DE L'ART DES MÉTHODES DE *plane fitting*

Les méthodes présentées sont généralement utilisées comme des méthodes de segmentation. Le *plane fitting* est à la 3D ce que la détection de lignes est à la 2D. Le *plane fitting* est utilisé pour déterminer les paramètres des plans en fonction des points leur appartenant. Les deux algorithmes dominants sont basés sur le RANSAC ou la croissance de région. Le RANSAC est efficace pour détecter de grands plans, mais très lent pour détecter de petits plans. La croissance de région est souvent basée sur la création d'un *Kd-tree* pour connaître les plus proches voisins de chaque point. Elle présente l'avantage d'être rapide dans le cas où l'on cherche à détecter de nombreux plans. Par contre, l'extraction des plans n'est pas très précise.

[Deschaud et al., 2010, Deschaud, 2010] proposent un algorithme basé sur le filtrage des normales et la croissance de voxel. Le but est d'avoir un algorithme rapide et précis.

[Kotthäuser et al., 2013] proposent une méthode pour reconstruire des pièces en modèles polygonaux à partir de nuages de points 3D. L'idée principale consiste à retrouver une configuration de pièce grâce à un dictionnaire pour la forme des pièces. Le but est d'être capable d'une part de modéliser la structure de la pièce, et d'autre part de séparer facilement les éléments de la pièce. Les auteurs exploitent la haute planéité des surfaces d'éléments importants tels que les murs pour être robuste aux occultations. Pour avoir une estimation initiale approximative de la structure de la pièce, les auteurs déterminent une trace au sol et au plafond. Elles forment un contour fermé. Ensuite ils segmentent les ensembles de points en plans et cherchent à déterminer leurs contours (avec une méthode de transformée de Hough 3D).

[Feng et al., 2014] proposent une méthode temps réel pour extraire les plans dans des

applications de robotique. Les données traitées sont des nuages de points organisés issus de capteurs Kinect. Les auteurs divisent d'abord le nuage de points en échantillonnant selon la profondeur. À partir de cela ils traitent un graphe où chaque ensemble ou échantillon de points constitue un nœud, ces nœuds étant liés aux ensembles voisins dans l'image. Les plans extraits sont affinés en utilisant la croissance de région. Cet algorithme pourrait être étendu à l'extraction d'autres primitives telles que des sphères et des cylindres.

[Hulik et al., 2014] proposent une méthode basée sur une transformée de Hough en 3D pour l'extraction de plans avec affinement dans le temps. Les auteurs travaillent aussi à partir d'images de profondeur type Kinect pour des applications de robotique. Ils comparent leur méthode avec une méthode classique de détecteur de plans basée sur le RANSAC. Leur méthode est plus stable pour la détection des plans et des faux positifs.

Dans [Holz et al., 2014], les auteurs décrivent une méthode de segmentation d'images de profondeur et de nuages de points organisés. Il s'agit d'une approche rapide de reconstruction des surfaces en moyennant un maillage polygonal. L'idée est de reconstruire approximativement les surfaces, et de segmenter l'image des profondeurs par une croissance des régions en utilisant les voisins locaux du maillage. Une segmentation des primitives (plans) peut être faite ensuite.

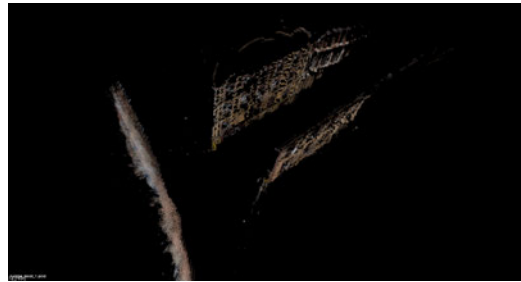
5.4.2/ *Plane fitting* EN 3D BASÉ SUR LE RANSAC

Le RANSAC est souvent expliqué avec la détection d'une droite dans un ensemble de points 2D. Pour le *plane fitting*, il s'agit d'une extension de cet exemple à la 3D, on cherche à détecter un plan dans un ensemble de points 3D. Il est utilisé pour détecter un plan et un seul, celui qui valide le plus de points.

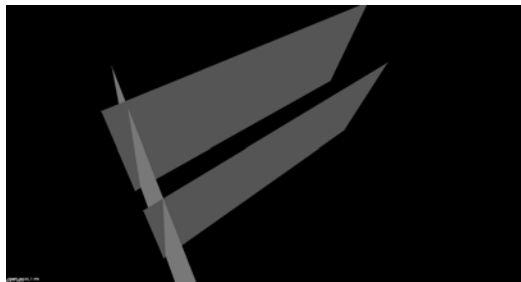
Le principe pour détecter plus d'un plan consiste à ne plus prendre en compte les points déjà assignés au premier plan détecté. Il s'agit de continuer à chercher tant qu'il reste une proportion donnée de points non assignés à un plan dans l'ensemble d'origine.

La segmentation par *plane fitting* RANSAC testée dans cette thèse est issue de l'implémentation de la bibliothèque PCL Point Cloud Library [Rusu et al., 2011]. L'inconvénient majeur de cette méthode est qu'elle détecte des plans infinis et ne considère pas des plans localement délimités. C'est une méthode globale sur l'ensemble des points. Souvent, un grand nombre de points dans une région décrit un plan localement. Avec cette méthode, le plan est détecté, mais il inclut des points d'autres régions locales de l'ensemble les empêchant de contribuer à d'autres plans potentiels. Il faut alors modifier l'algorithme de base pour qu'il se limite à détecter des polygones en se basant sur les équations des plans détectés. On pourrait choisir de traiter l'ensemble des points localement en le divisant en régions. Différentes stratégies d'échantillonnage avec ou sans chevauchement pour gagner en robustesse peuvent être envisagées.

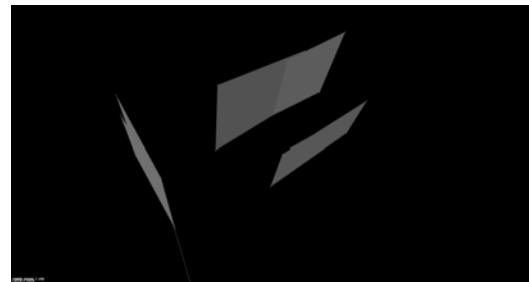
En l'état, la méthode développée se base sur les plans infinis. À l'extraction d'un plan elle considère l'ensemble des points *inliers* ayant contribué pour délimiter un polygone. Les limites du polygone sont définies par les points situés aux extrémités de l'ensemble des *inliers*. On rencontre des problèmes dans le cas de croisements de rues. Les plans traversent les espaces libres, la détection est donc incorrecte. La figure 5.7 illustre ce problème en 5.7(b) et montre la comparaison du résultat avec la méthode finalement proposée en 5.7(c), décrite à la section suivante 5.4.3.



(a) Nuage de points.



(b) Plans détectés avec la méthode RANSAC.



(c) Plans détectés avec la méthode Hough 2D.

FIGURE 5.7 – Détection des plans au croisement de la rue du Général Roussel, Belfort, avec (b) la méthode basée sur RANSAC et (c) la méthode de la transformée de Hough 2D.

5.4.3/ *Plane fitting* PAR TRANSFORMÉE DE HOUGH 2D SUR CARTOGRAPHIE

La méthode par RANSAC n'est pas utilisable telle qu'elle. Une méthode différente est proposée, basée sur le fait (connaissance à priori) que les murs des bâtiments forment des plans verticaux, c'est-à-dire que leur normale est orthogonale à l'axe vertical Z . Cela permet de passer à une analyse 2D de la trace au sol de l'environnement traité. Le problème est alors simplifié et la détection peut être plus efficace.

5.4.3.1/ ESTIMATION DE L'EMPREINTE AU SOL

À partir du nuage 3D, il est possible de créer une cartographie 2D des lieux reconstitués. Pour cela, on considère des zones autour d'une position choisie correspondant à une caméra pour la reprojection dans les images ou au récepteur pour l'application GNSS. Autour de cette position, on crée un histogramme en Z des points dans le plan XY . Pour cela, il faut échantillonner l'espace suivant une grille régulière de cases carrées dans XY . Il suffit

de compter le nombre de points 3D par région de cette matrice. Plus une structure aura localement des points suivant sa hauteur (Z), plus sa représentation dans l'histogramme 2D aura une valeur élevée. Plus le nombre de points est dense, plus on est sûr de l'existence de la structure à l'endroit considéré. Des points 3D bruités non filtrés apparaissent mais avec une faible représentation. Cet histogramme s'apparente à une projection orthographique selon Z dans XY , on peut traiter cela comme une cartographie 2D (ou trace au sol).

Une carte des hauteurs est aussi proposée. Là, c'est la hauteur (Z) maximale des points dans l'espace d'échantillonnage qui est comptabilisée. La carte des hauteurs est filtrée pour lisser les mesures et réduire le bruit. En effet elle peut ne pas être très régulière initialement. Le lissage appliqué est un filtre médian de taille 3×3 qui ignore les points de valeur négative ou nulle.

L'échantillonnage choisi est lié au filtrage de *downsampling* appliqué au nuage en section 4.4.1. Pour rappel, la dimension de voxel choisie est de 10cm. Ainsi, on échantillonne les projections au sol à la même dimension, soit par un pas de 10cm pour 1 pixel. La figure 5.8 montre un histogramme (a) et la carte des hauteurs correspondante (b). L'histogramme est seuillé pour saturer à la valeur d'intensité maximale de l'image (255) à partir d'un décompte équivalent à une hauteur d'au moins 2m au-dessus de la hauteur du récepteur GNSS. La trace au sol issue de l'histogramme est ainsi bien marquée et visible.

Les cartes sont calculées pour des bornes de 20m selon X et Y autour de la position de référence. Elles doivent être assez grandes pour permettre de détecter un ensemble assez complet d'information ne se limitant pas qu'à la proximité immédiate. Les traces obtenues sont épaisses et nécessitent des pré-traitements à leur exploitation. Dans les traces au sol, l'axe X est orienté vers la droite et l'axe Y vers la gauche. L'origine est en haut à gauche de l'image.

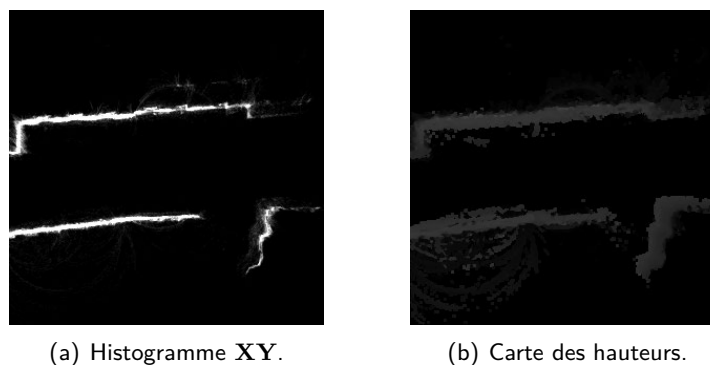


FIGURE 5.8 – Trace au sol à la position juste avant la cathédrale pour la situation rue du Général Roussel, Belfort. L'axe Y du monde est orienté vers le bas et l'axe X vers la droite.

5.4.3.2/ DÉTECTION DES MURS PAR TRANSFORMÉE DE HOUGH

Nous exploitons et fusionnons les informations des histogrammes en XY et des cartes des hauteurs présentées dans la section 5.4.3.1. La projection des points dans XY selon Z fournit

une trace au sol des structures. Pour avoir une trace au sol plus proche de la réalité, il faut filtrer les zones en fonction de leur intensité. En effet, le nuage de points présente du bruit et des problèmes de propagation de structures liés à l'appariement stéréo (voir le chapitre 4). Dans un tel cas, le nuage de points montre des débordements de points à des endroits où il n'y a pas de structure (au-dessus de la rue par exemple). En général, ces propagations de points n'apparaissent qu'au niveau de la hauteur maximale des structures reconstruites, et ne vont pas depuis le sol jusqu'à cette hauteur. C'est-à-dire qu'il y aura peu de points comptabilisés dans l'histogramme comparativement à une position sur un mur, qui lui va depuis le sol jusqu'à sa hauteur maximale. Un seuil basé sur l'intensité de l'histogramme permet donc d'éliminer de fausses détections.

À partir des traces au sol, nous proposons d'exploiter la transformée de Hough pour la détection des segments. Les segments servent de base à l'estimation des plans du modèle 3D. La transformée de Hough est expliquée dans [Illingworth et al., 1988] et [Fisher et al., 2004]. L'avantage de la transformée de Hough est qu'elle tolère la présence de trous dans les descriptions de bords à détecter et qu'elle est relativement robuste au bruit. Elle est utilisée pour détecter des droites ou des cercles dans une image noir et blanc. Il existe la transformée de Hough probabiliste [Matas et al., 2000], qui permet la détection de segments au lieu de droites. C'est cette variante qui est intéressante et utilisée pour l'application présentée.

La carte des traces au sol doit subir quelques traitements avant de détecter les segments : il faut des traces fines, peu bruitées, et en binaire. Pour cela, nous appliquons en pré-traitement une série d'opérateurs de morphologie mathématique et de filtres sur l'histogramme dans \mathbf{XY} :

1. Dilatation sur 1 pixel suivie d'une érosion sur 1 pixel (fermeture), le but est de « boucher » quelques trous présents dans la trace.
2. Seuillage binaire, un seuil relativement bas a été choisi, au quart de l'amplitude maximale soit 63 pour un codage de 0 à 255 (un quart de l'amplitude maximale).
3. Amincissement de Zhang et Suen [Zhang et al., 1984] (en anglais *thinning* ou *skeletonization*), nécessaire pour éviter des détections de segments qui se chevauchent.

À partir de la trace au sol mince, nous pouvons détecter les segments avec la transformée de Hough probabiliste. Les paramètres utilisés dans l'espace paramétrique polaire de Hough sont :

- Résolution pour r de 1 pixel (équivalent à 10cm du monde réel, voir section 4.4.1).
- Résolution angulaire fine avec θ de $0,5^\circ$.
- Seuil d'accumulation pour garder un segment fixé à 30, il est faible pour maximiser le nombre de segments, car peu de chevauchements sont possibles grâce à l'amincissement.
- Longueur minimale pour un segment fixée à 1m du monde réel pour tenter de s'approcher au mieux des variations possibles des murs du monde réel.
- Longueur maximale admissible pour un trou dans un segment fixée à 2m du monde réel, car certaines traces peuvent avoir été trop faibles pour afficher une ligne bien continue.

La figure 5.9 illustre les différentes étapes pour la détection des segments à partir de l'histogramme en \mathbf{XY} .

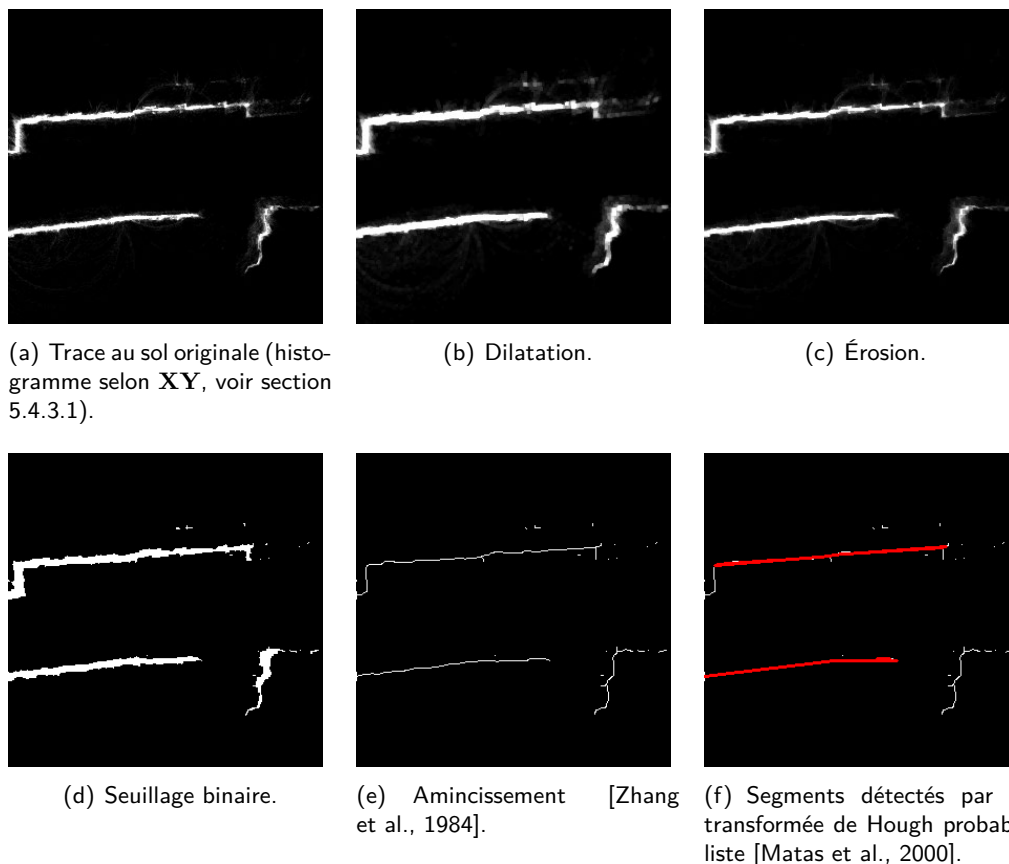


FIGURE 5.9 – Pré-traitements appliqués sur la trace au sol générée à partir du modèle 3D reconstruit et détection de segments par transformée de Hough probabiliste, pour la situation rue du Général Roussel, Belfort.

Après application de la détection des segments, nous disposons d'un ensemble de segments correspondants aux structures du nuage 3D. Il est possible de déterminer des polygones à partir des segments et de la carte des hauteurs comme suit :

- On considère que toutes les structures partent du niveau de la caméra d'origine ($\mathbf{Z} = 0$). On connaît donc les deux points du bas, à partir des positions des extrémités dans \mathbf{XY} , et en posant $\mathbf{Z} = 0$.
- Les deux points du haut sont définis avec les positions dans \mathbf{XY} de chaque extrémité, et le \mathbf{Z} correspondant donné par la carte des hauteurs.

Nous obtenons ainsi un maillage de quadrilatères. Une manière efficace de trianguler ce maillage consiste à diviser ces quadrilatères en deux triangles. Une méthode de séparation classique est la triangulation adaptative (voir [Holz et al., 2014]). Dans cette approche, on coupe le quadrilatère en deux triangles selon la diagonale la plus courte pour diviser les angles les plus grands. Il est dit que cela permet une meilleure fermeture des maillages triangulaires.

5.4.4/ ÉVALUATION DES INFORMATIONS EXTRAITES SUR SCÈNES RÉELLES

L'évaluation des plans extraits n'est pas évidente à réaliser. Se pose le problème de l'acquisition d'une vérité terrain. Sans modèle 3D complet ou MNE (Modèle Numérique d'Élévation) des environnements reconstruits, nous ne disposons d'aucun moyen de mesure des données chiffrées précises. Nous proposons des critères sans vérité qui semblent être adaptés aux données traitées et pour l'application présentée : une évaluation visuelle par reprojektion des éléments construits dans les images de départ, pour voir si les hauteurs sont bien estimées. Sinon, une mesure possible est la largeur de la rue à partir de données cadastrales.

5.4.4.1/ CRITÈRES D'ÉVALUATION

On ne peut pas évaluer directement les structures 3D extraites, en particulier la hauteur qui est un élément important pour l'application présentée. Il est possible toutefois de mesurer la largeur de la rue comme on l'a fait précédemment dans le chapitre 4. Il convient tout de même d'accompagner ces mesures d'une évaluation des hauteurs de visu, en projetant les points et plans estimés du monde 3D dans les images d'origine ayant servi à construire le modèle. Les critères peuvent être listés selon deux catégories, les critères qualitatifs :

- Traces au sol et segments détectés pour les modèles locaux du chapitre 4 ;
- Projection des points hauts dans les images d'après la carte des hauteurs ;
- Projection des plans détectés dans les images.

le critère quantitatif :

- Largeur entre les plans des structures qui s'opposent de part et d'autre de la rue.

Pour valider le modèle 3D, on peut en reprojeter les points dans les images acquises. On propose de reprojeter les points du haut des bâtiments au lieu de tous les points auquel cas les erreurs seraient plus difficiles à détecter. Pour cela, on utilise la carte des hauteurs conjointement à l'histogramme dans XY . Pour tous les points de la trace au sol générée autour de la position étudiée, on vérifie si l'intensité de l'histogramme est assez forte, ce qui permet de s'assurer de l'existence d'une structure grande et non pas d'un simple élément comme par exemple un balcon ou un store. On n'attache de l'importance qu'aux structures importantes comme les murs des bâtiments. On fixe un seuil d'acceptation au quart de l'amplitude maximale possible de l'histogramme. Et donc, pour ces positions où la structure est intéressante à considérer, on récupère l'information de hauteur la plus grande correspondante grâce à la carte des hauteurs. On peut donc positionner un point 3D au centre de l'échantillon en XY , dont la hauteur Z est lue dans la carte des hauteurs. Grâce aux paramètres de calibrage de la caméra fisheye, on peut contre-projeter tous ces points dans l'image d'origine, et voir si effectivement les points obtenus correspondent aux limites hautes des bâtiments dans la photo.

Selon le même principe, il est possible d'afficher la projection des plans estimés dans les images. Cela donne une information sur la cohérence des estimations obtenues.

5.4.4.2/ RÉSULTAT DES ÉVALUATIONS

Les plans opposés détectés ne sont pas parfaitement parallèles entre eux. Et, on peut trouver parfois des plans qui se chevauchent. Pour mesurer la largeur de la rue en exploitant les informations extraites, on la considère comme une tranchée de largeur constante dans le modèle local. La distance de chaque segment est mesurée selon leur normale par rapport à l'axe central de la rue. Pour chacun des deux côtés la mesure médiane est conservée comme distance de ce côté par rapport au centre de la tranchée. Ces deux distances sont cumulées pour former l'estimation de la largeur de la rue. Les tableaux 5.2 et 5.3 montrent les résultats des mesures de largeur de rue ainsi que les cartes des segments détectés, et la projection des points hauts et des plans dans les images fisheye.

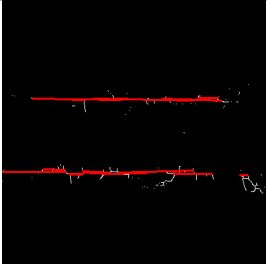

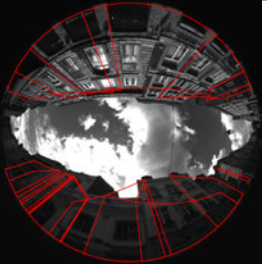
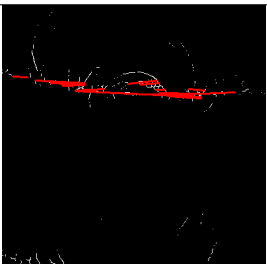
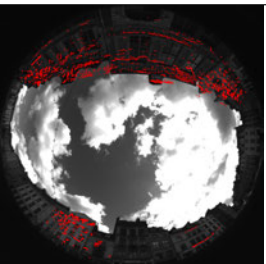
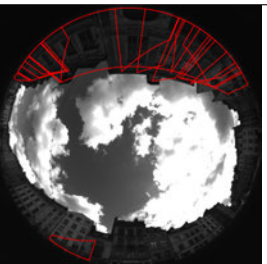
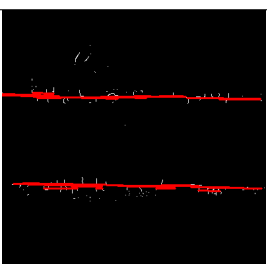
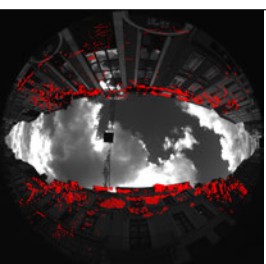
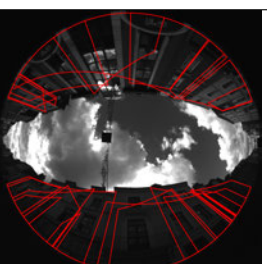
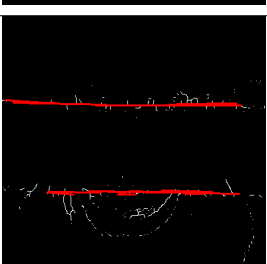

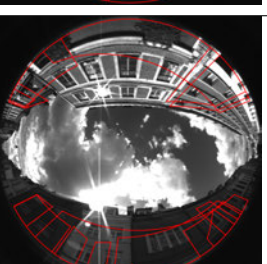
	Segments 2D	Cartes et projections Points hauts		Largeur vérité	Mesure tranchée
			Plans		
Cap2/3039				$\approx 10\text{m}$	11,2m
Cap2/2012				$\approx 25\text{m}$	17,2m
Cap3/1852				$\approx 12\text{m}$	13,6m
Cap3/2149				$\approx 12\text{m}$	13,5m

TABLEAU 5.2 – Évaluation de la largeur des rues pour les scènes réelles stéréo de Lille en ligne droite à partir des plans détectés.

Dans la scène Cap2/2012 (tableau 5.2), la mesure ne fonctionne pas correctement car les limites de détection des plans sont plus courtes que l'éloignement des murs d'un côté.

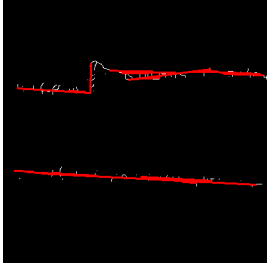
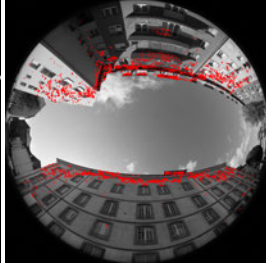
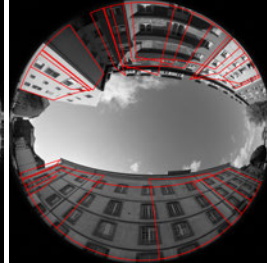
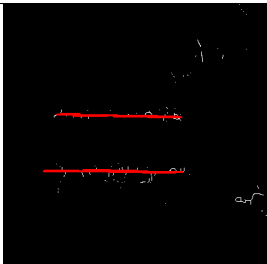
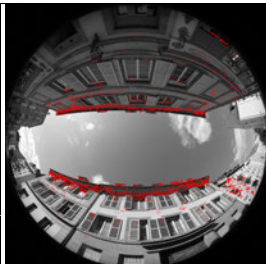
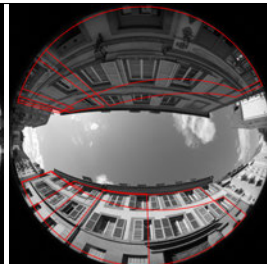
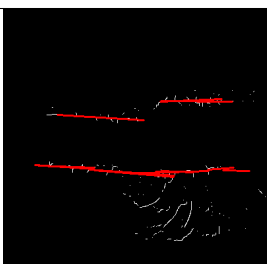
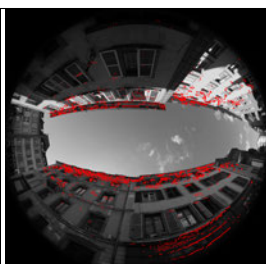



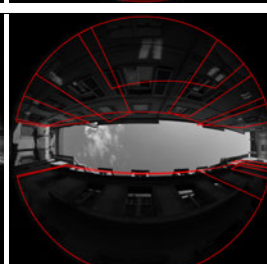
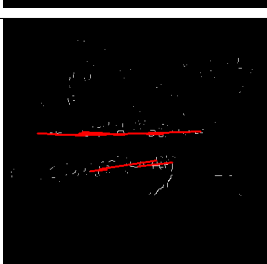


	Segments 2D	Cartes et projections Points hauts	Plans	Largeur vérité	Mesure tranchée
3734				≈ 16m	16,1m
3694				≈ 8m	8,4m
3908				≈ 8m	10,8m
3940				≈ 5m	5m
4161				≈ 5m	5,1m

TABLEAU 5.3 – Évaluation de la largeur des rues pour les scènes réelles mono de Belfort en ligne droite avec mise en correspondance en couleurs et masque du ciel à partir des plans détectés.

La précision des mesures de largeur est comparable aux mesures obtenues à partir des nuages

de points. Notons que le résultat est très précis dans le cas de la scène 3940 de Belfort (tableau 5.3), là où les mesures étaient aberrantes à partir du nuage de points. Mesurer les structures à partir des plans est plus robuste que de prendre les données des points de base.

Les points projetés dans les images sont pour la plupart bien positionnés en haut des structures. Les plans sont souvent de hauteur légèrement inférieure à la hauteur réelle des bâtiments. Ce sont pour les scènes de Belfort qu'ils sont le mieux estimés, la résolution des images utilisées étant plus importante.

5.5/ REPRÉSENTATION DES SATELLITES DANS LA PARTIE VISION

Les satellites peuvent être représentés grâce aux informations d'azimut et d'élévation accessibles dans les données fournies par le récepteur. Il faut de plus utiliser l'orientation du véhicule par rapport au Nord géographique, faite manuellement pour les résultats présentés.

5.5.1/ PROJECTION DES SATELLITES ET RAYONS GNSS SUR LA TRACE AU SOL

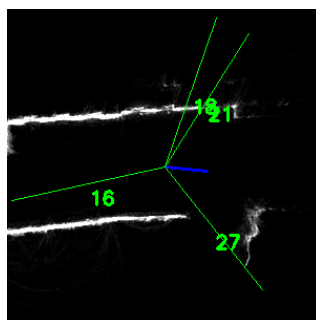


FIGURE 5.10 – Projection des satellites dans la trace au sol autour du récepteur GNSS. Le Nord géographique est indiqué en bleu. Situation rue du Général Roussel, Belfort.

Avec la connaissance de la position 3D des satellites par rapport au repère centré sur le récepteur, on peut facilement tracer la projection des signaux GNSS sur la trace au sol générée depuis le modèle 3D. La figure 5.10 illustre cette représentation, incluant le Nord. Cette représentation est centrée sur le récepteur, c'est-à-dire que sa position dans l'image correspond à son centre.

5.5.2/ PROJECTION DES SATELLITES DANS LES IMAGES FISHEYE

Pour reprojeter les positions des satellites dans les acquisitions images, il est nécessaire de les considérer au plus près de leur véritable position en 3D. Pour cette raison, on les considère en 3D à une distance donnée du récepteur, qui n'est autre que la pseudodistance. C'est ce point qui est ensuite reprojété dans l'image choisie en tenant compte du décalage entre l'antenne du récepteur GPS et la caméra, du modèle de projection et des paramètres décrivant l'objectif et la caméra obtenus au calibrage (voir le chapitre 3).



(a) Projection des satellites reçus dans l'image. L'axe Y de l'image est inversé, ce qui fait que l'Est et l'Ouest sont inversés. Le Nord est indiqué en bleu.

(b) Satellites théoriquement visibles d'après l'almanach YUMA. Les gros points représentent les satellites effectivement reçus par le récepteur.

FIGURE 5.11 – Comparaison entre la projection dans l'image des satellites reçus par le récepteur Safedrive et de la constellation de satellites disponibles au même moment au même endroit (le 25 août 2014, temps ITOW : 138970s, rue du Général Roussel à Belfort).

La figure 5.11 montre la correspondance entre les projections des satellites dans l'image fisheye et le diagramme *skyplot* obtenu à partir de l'almanach YUMA et dessinant tous les satellites potentiellement visibles (tracé avec le module `GPSofT`³ pour Matlab).

Il faut se placer dans le repère optique de la caméra avant d'appliquer la projection inverse tenant compte du modèle de projection fisheye.

5.5.3/ CRÉATION DE MASQUES DE VISIBILITÉ D'APRÈS LES PLANS DÉTECTÉS

La section 5.4 montre comment extraire les plans des modèles 3D estimés. Pour créer les masques de visibilité des satellites, il faut se baser sur les structures détectées.

Le masque est créé dans un *skyplot* à la position de l'antenne du récepteur GPS. Cela nécessite de connaître sa position dans le repère 3D local. C'est une information connue, car on a mesuré la position relative des capteurs entre eux, et la référence pour le repère 3D est la caméra arrière. Les mesures pour les configurations présentées sont données dans la section 1.3.2. Projeter les plans dans le *skyplot* revient à les projeter sur une sphère, puis à ignorer la hauteur pour se placer dans le « plan » du diagramme.

Des masques de visibilité dans le *skyplot* sont montrés dans les résultats de la section 5.6.3.

5.6/ ÉVALUATION EN CONDITIONS RÉELLES

Nous nous sommes rendus compte à travers nos expérimentations, et malgré nos efforts pour approcher une bonne synchronisation, que l'on ne peut pas évaluer de manière fiable l'amélioration de la localisation pour des trajets en dynamique. Les évaluations proposées concernent donc des positions fixes où l'on est sûr de trouver les informations GPS correspondant bien à la position donnée. Elles permettent de démontrer le bon fonctionnement de la méthode. Des commentaires sur les résultats obtenus en conditions dynamiques sont donnés à la fin de la

3. <http://gpsoftnav.com/>

section.

Les positions GPS sont généralement données en latitude, longitude, altitude dans le système de coordonnées géodésiques WGS 84. Pour les évaluations, les positions estimées sont converties au repère local cartésien ENU (*East North Up*, c'est-à-dire en mètres selon les axes correspondant à l'Est, au Nord et à la Hauteur), centré sur la position initiale de la séquence traitée. Cela permet de calculer directement l'écart en mètres entre les positions estimées et les positions vérités. En pratique nous ne disposons pas toujours des trois composantes des positions vérités, la hauteur faisant défaut lorsque nous avons reporté les positions manuellement (voir section 5.6.2). Pour cette raison, les mesures présentées concernent les positions estimées dans le plan horizontal 2D.

5.6.1/ PROCÉDURE D'ÉVALUATION

Pour évaluer la pertinence de la rectification GNSS proposée, nous utilisons un récepteur GPS Safedrive proposé par la société M3 Systems⁴. Il est équipé d'une puce bas-coût u-blox, utilise une antenne patch et stocke toutes les données brutes. L'intérêt est de traiter les données comme si elles étaient fournies par un récepteur grand-public, mais sans lissage ou amélioration artificielle des positions estimées. Le Safedrive est parfait pour cela : il est configuré pour exploiter les services EGNOS (*European Geostationary Navigation Overlay Service*) permettant les corrections classiques d'horloge, d'éphémérides, ionosphériques et troposphériques, pour contrôler l'intégrité des mesures PVT (Position, Vitesse, Temps), et pour ne pas appliquer de lissage des pseudodistances ou des positions. Ce dernier point est crucial pour effectuer un calcul brut des positions. Les positions sont calculées par les moindres carrés.

L'objectif de cette section est de montrer que le modèle 3D généré dans les travaux présentés peut-être utilisé pour améliorer la localisation GNSS. On utilise les nuages de points locaux non-assemblés. L'étape d'assemblage des nuages globaux peut compléter et améliorer le modèle local ou inversement être une source d'erreurs supplémentaires dans le cas où l'estimation de pose est très imprécise (chapitre 4).

Nous proposons d'évaluer la position 2D obtenue avec les 4 méthodes suivantes :

- Position 2D tenant compte des satellites LOS et NLOS sans modifications.
- Exclusion des NLOS, génère une perte en disponibilité du service et dégrade les DOP.
- Rectification des NLOS avec le modèle à une réflexion (section 5.3.3.1).
- Rectification des NLOS avec le modèle de une à trois réflexions (section 5.3.3.2).

De plus, ces méthodes sont évaluées pour deux procédures de distinction des NLOS :

4. <http://m3systems.net/>

- Seuil CN0, est considéré NLOS tout signal dont $CN0 < 40\text{dBHz}$ (section 5.3.1, reprise du seuil utilisé par [Marais, 2002]).
- Masque d'occultation à la position du récepteur décrit en section 5.5.3.

Pour appliquer le modèle de la « tranchée urbaine », les distances employées sont mesurées selon la méthode décrite en section 5.4.4.2.

5.6.2/ SCÈNES ET POSITIONS ÉVALUÉES

Pour ces évaluations, seule la constellation des satellites GPS est utilisée. Des positions ont été retenues pour les acquisitions de Lille du 19 août 2014. C'est la configuration (Cfg 1) en stéréovision binoculaire. Les données GPS Safedrive à traiter disposent de l'heure ITOW, et les images acquises sont datées à l'heure UTC (attention il a été constaté que l'heure UTC fournie ne tient pas compte des *leap seconds*). Le système est synchronisé avec l'heure GPS au début des acquisitions. Comme ces acquisitions ne disposent pas de vérités terrain enregistrées en même temps, des positions fixes 2D ont été repérées et géo-référencées manuellement à partir d'un logiciel de cartographie (Géoportail IGN). Ce système enregistre les vidéos en continu, il est pratique pour extraire les positions fixes. Les positions étudiées sont les suivantes et nommées avec le numéro d'acquisition utilisé pour construire le modèle 3D :

- Cap3/0900, temps ITOW $\in [218789; 218817]$, Nord géographique = 318° , avant le croisement entre la rue Saint André et la rue de la Halle
- Cap4/1560, temps ITOW $\in [219055; 219086]$, Nord géographique = 115° , avant le croisement entre la rue Esquermoise et la rue de Thiers

Une position issue des acquisitions de Belfort (25 août 2014) est également présentée. C'est la configuration (Cfg 4) en estimation des structures par vision monoculaire avec le déplacement du capteur. La position de référence est donnée par un récepteur GPS RTK Magellan ProFlex 500 avec une station de base auxiliaire. Pour harmoniser cette évaluation avec celles des positions de Lille, seule la position 2D est prise en considération. La position fixe permet de faire la synchronisation des informations manuellement entre les images dont l'heure est imprécise et les données GPS et GPS RTK pour lesquelles l'heure est précise :

- 6506, temps ITOW $\in [138824; 138826]$, Nord géographique = 24° , avant le croisement entre la rue François Noblat et la rue des Quatre Vents

5.6.3/ POSITION AMÉLIORÉE AVEC CORRECTION DES PSEUDODISTANCES

Les tableaux 5.4, 5.5 et 5.6 présentent les résultats pour les acquisitions étudiées. L'exclusion des signaux NLOS est assez imprévisible. Dans les cas montrés, elle augmente énormément l'erreur de position. L'autre inconvénient est la diminution de la disponibilité du service. Pour l'acquisition Cap4/1560 (tableau 5.5), on n'a jamais assez de satellites LOS pour obtenir une position. L'exclusion des satellites NLOS n'est pas une solution viable. Il faut absolument aller plus loin et utiliser l'information LOS ou NLOS dans une procédure spécifique d'amélioration

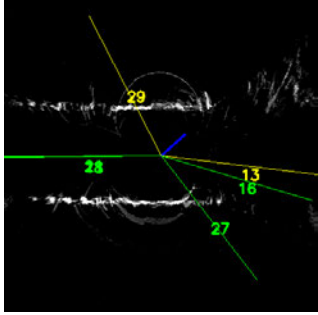
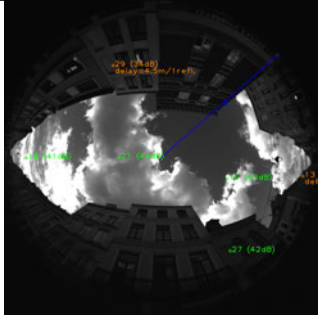
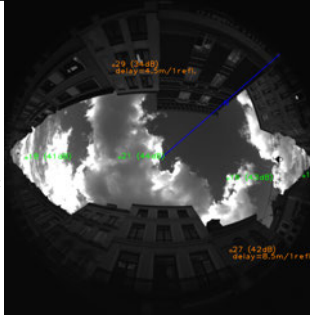
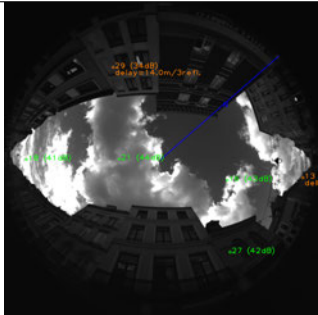
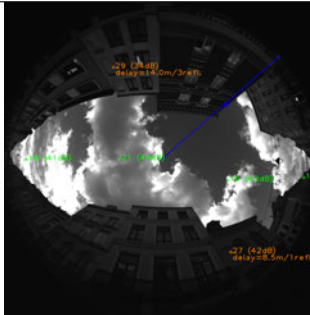
Correction	Méthode de distinction LOS et NLOS	
	Seuil CN0	Masque de visibilité <i>skyplot</i>
Sans	Erreur de position 2D = 22,73m ; Écart-type = 7,31m	
Exclusion NLOS	Erreur = 61,77m Écart-type = 3,21m Disponibilité = 85,71%	 Erreur = 98,73m Écart-type = 17,78m Disponibilité = 100%
1 réflexion	 Erreur = 21,34m Écart-type = 9,52m	 Erreur = 12,64m Écart-type = 1,80m
1 à 3 réflexions	 Erreur = 22,69m Écart-type = 11,59m	 Erreur = 11,50m Écart-type = 3,13m

TABLEAU 5.4 – Évaluation de la localisation 2D avec correction du retard des pseudodistances des satellites NLOS pour l'acquisition Cap3/0900 de Lille en configuration binoculaire (Cfg 1).

de la localisation.

Pour l'acquisition Cap3/0900 (tableau 5.4), corriger les NLOS déterminés selon le CN0 n'apporte aucune amélioration. L'écart-type est plus grand, ce qui signifie que la position corrigée est moins stable qu'à l'origine.

Sinon, dans tous les cas, la précision est meilleure et l'écart-type est plus faible pour les rectifications sur les NLOS détectés à partir du masque de visibilité dans le *skyplot* depuis les plans 3D à la position du récepteur qu'à partir du seuil sur le CN0. La distinction basée sur la 3D semble donc être plus fiable qu'avec le seuil CN0, contrairement aux résultats obtenus dans [Marais et al., 2014].

Avec l'acquisition 6560 (tableau 5.6), les résultats de détection des signaux NLOS sont

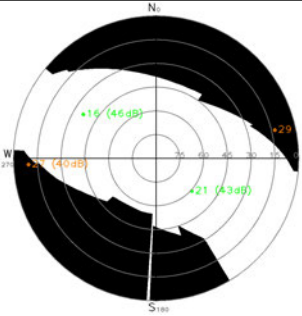
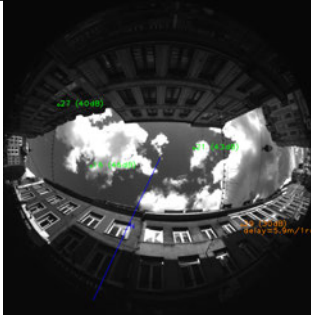
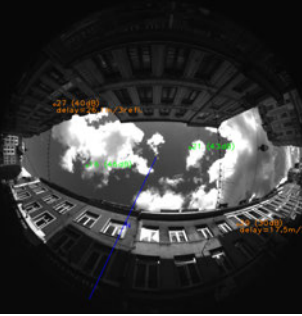
Correction	Méthode de distinction LOS et NLOS	
	Seuil CN0	Masque de visibilité <i>skyplot</i>
Sans	Erreur de position 2D = 16,70m ; Écart-type = 3,63m	
Exclusion NLOS	Disponibilité = 0%	 Disponibilité = 0%
1 réflexion	Erreur = 12,57m Écart-type = 4,74m	 Erreur = 12,16m Écart-type = 3,57m
1 à 3 réflexions	Erreur = 7,61m Écart-type = 6,81m	 Erreur = 6,47m Écart-type = 2,11m

TABLEAU 5.5 – Évaluation de la localisation 2D avec correction du retard des pseudodistances des satellites NLOS pour l'acquisition Cap4/1560 de Lille en configuration binoculaire (Cfg 1).

les mêmes avec les deux méthodes. Les corrections appliquées arrivent donc aux mêmes estimations.

Les résultats montrent que la correction avec réflexions multiples aboutit à une position finale plus précise que le modèle simple à une seule réflexion. Les résultats obtenus sont très encourageants, l'imprécision de position est réduite d'au moins 50% environ, et l'écart-type d'au moins 30% environ.

Effet de la méthode en positions dynamiques

Nous avons essayé d'appliquer la méthode de correction en conditions dynamiques. Dans ces conditions la position du véhicule n'est pas maintenue comme précédemment. Par

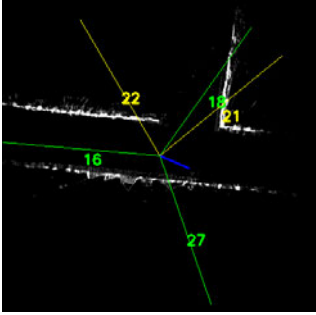
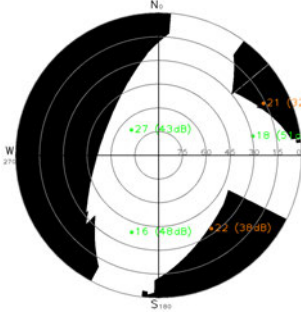
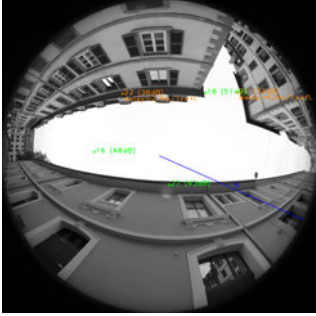
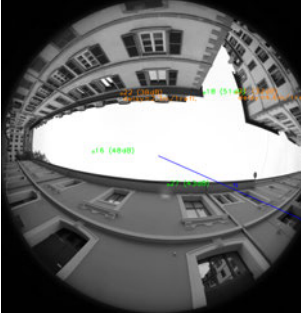
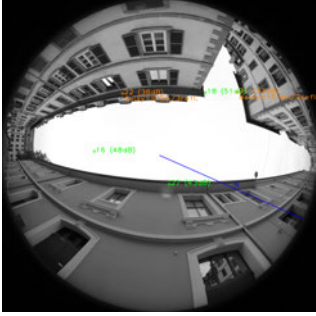
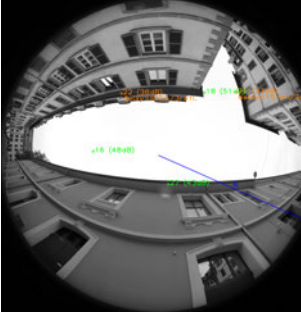
Correction	Méthode de distinction LOS et NLOS	
	Seuil CN0	Masque de visibilité <i>skyplot</i>
Sans	Erreur de position 2D = 12,60m ; Écart-type = 9,32m	
Exclusion NLOS	Disponibilité = 0% 	Disponibilité = 0% 
1 réflexion	Erreur = 9,80m Écart-type = 5,31m 	Erreur = 9,80m Écart-type = 5,31m 
1 à 3 réflexions	Erreur = 6,60m Écart-type = 0,28m 	Erreur = 6,60m Écart-type = 0,28m 

TABLEAU 5.6 – Évaluation de la localisation 2D avec correction du retard des pseudodistances des satellites NLOS pour l'acquisition 6506 de Belfort en configuration monoculaire (Cfgr 4).

conséquent, la correction est appliquée à chaque instant où les signaux sont reçus et que les pseudodistances sont estimées. La figure 5.12 montre les résultats obtenus pour une séquence avec déplacement acquise à Belfort avec le système en configuration (Cfgr 4). Les temps de référence pour la synchronisation sont les temps des données du récepteur GPS Safedrive. Les positions vérités sont connues avec le récepteur GPS RTK Magellan ProFlex 500, dont les données subissent une interpolation cubique pour être datées précisément aux temps de référence et être les plus proches possibles de la vérité. Avec cette configuration et pour les acquisitions de Belfort du 25 août 2014, on estime atteindre un écart-type de synchronisation entre les données images et les temps de référence d'environ 0,194s. Pour les vitesses du véhicule comprises entre 15km/h (vitesse maximale) et 7,2km/h, la précision de la synchronisation est variable et diminue à mesure que le véhicule est plus lent. Par ailleurs pour les données dynamiques, la position du récepteur GNSS dans le modèle 3D généré tient

compte de la position relative entre l'antenne et la caméra de référence, ainsi que de la vitesse du véhicule à l'instant considéré (issue des données GPS).

La figure 5.12(a) est une représentation des positions du véhicule pour plusieurs méthodes en coordonnées ENU centrées sur la position initiale, dans le plan horizontal. En vert sont représentées les positions vérités acquises avec le récepteur GPS RTK. Les trois autres courbes représentent les positions estimées avec les moindres carrés sans correction (rouge), avec exclusion des satellite NLOS (cyan avec les positions disponibles entourées par un losange) et avec correction géométrique (bleu). Nous observons que lorsque le véhicule se déplace, les moindres carrés sans correction et la méthode par exclusion sont plus proches de la vérité terrain que la méthode proposée. Comme nous le constatons sur la figure 5.12(b), l'erreur moyenne de la localisation 2D après correction est proche de 50m. À certains moments l'erreur est très élevée. Le positionnement est amélioré à de très rares instants. On remarque que l'exclusion des signaux NLOS réduit la disponibilité et n'améliore pas la localisation, toutefois elle n'engendre pas les pics d'erreur de notre méthode.

Nous tentons une première explication. Comme nous le présentions au début de ce chapitre, le temps et la synchronisation entre les données GPS (c'est-à-dire les signaux reçus) et les images sont deux données très sensibles. Sachant que la correction apportée aux pseudodistances dépend directement de la structure des bâtiments autour du véhicule à l'instant du positionnement GPS considéré, ces deux informations doivent être précisément synchronisées pour être certain que les caractéristiques de propagation correspondent exactement à la structure estimée à partir du système mono-caméra ou du stéréoscope.

Une seconde explication pourrait être que la structure même du récepteur et son algorithme ne sont pas compatibles avec une correction géométrique lorsque le véhicule est en mouvement : dans ce cas il faut tenir compte du filtrage appliqué aux pseudodistances estimées à partir des codes pseudoaléatoires de chaque satellite.

Une dernière explication pourrait être que, sur ce trajet, l'erreur sans correction est suffisamment faible y compris lorsqu'un satellite est réfléchi. Nous observons que l'erreur moyenne sans correction est de 10m. Dans ce dernier cas, il est facile d'imaginer qu'appliquer une correction sur un signal réfléchi puisse ne pas améliorer la précision de l'estimation et qu'une correction imprécise pourrait augmenter l'erreur initiale.

5.7/ CONCLUSION

Ce chapitre est consacré à l'utilisation d'un modèle 3D de l'environnement pour augmenter la précision de la localisation GNSS d'un véhicule. Nous décrivons une méthode qui extrait les façades des bâtiments et en estime la position à partir d'un nuage de points 3D obtenu par la méthode de calibrage et de mise en correspondance décrite aux chapitres précédents. Grâce à cela nous parvenons à créer le masque de visibilité (*skyplot*) autour du véhicule à partir duquel nous estimons l'état des satellites (LOS ou NLOS) et les erreurs des pseudodistances en présence de réflexions des signaux satellitaires.

Nous parvenons à améliorer la position GPS du véhicule en corrigeant les pseudodistances qui ont été rallongées après au moins une réflexion. Cette correction donne de bons résultats lorsque le véhicule est statique. Nous en montrons des résultats pour la ville de Lille et la ville de Belfort. Le modèle à une, deux ou trois réflexions est performant. Une extension serait de le compléter par un modèle de diffraction tel que celui présenté dans [Ercek et al., 2005] pour corriger les satellites dont le masquage est ambigu aux zones frontalières entre les parties visibles et masquées.

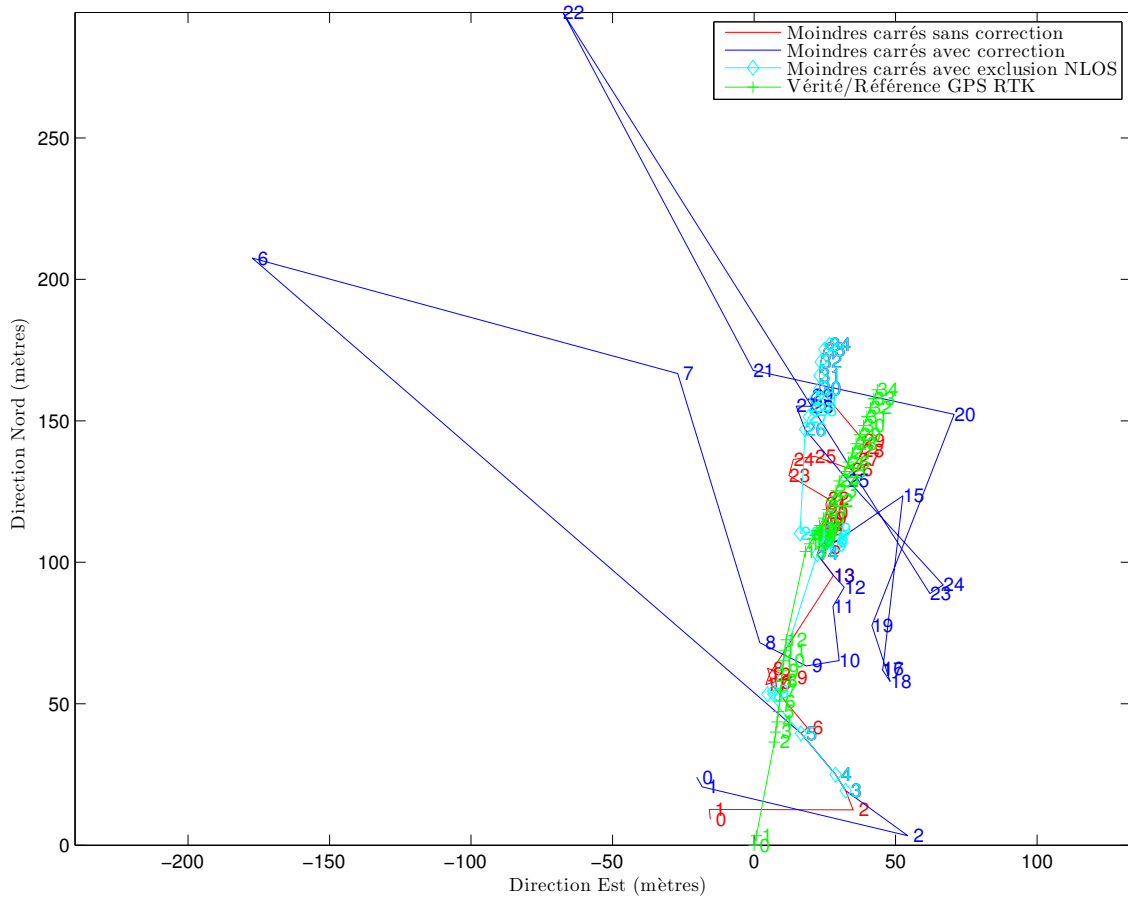
Une nouvelle méthode de masquage à la position du récepteur est présentée pour déterminer quels sont les satellites NLOS. Le masque est généré d'après le modèle 3D construit de manière dynamique. Un tel masquage est plus précis qu'un masquage dans une vue caméra car il tient compte du décalage de position entre le récepteur GNSS et la caméra de référence. Ce décalage peut être suffisant pour ne pas distinguer correctement un satellite ambigu si on se place à la position de la caméra. Le masquage construit à partir des plans extraits est donc plus performant qu'un masquage basé sur segmentation comme dans [Marais et al., 2014]. Dans ce même article, Marais note une mauvaise estimation des états LOS ou NLOS des satellites à partir des masques dans les images.

Des résultats en conditions dynamiques n'ont pas été montrés à cause du problème de synchronisation qui se pose entre les données images et GPS. La précision de synchronisation atteinte au cours des acquisitions présentées est légèrement inférieure à 2 dixièmes de seconde. Sans synchronisation beaucoup plus précise (de l'ordre du centième voire peut-être du millième de seconde), le problème est trop complexe et un modèle de réflexion classique ne peut pas être exploité.

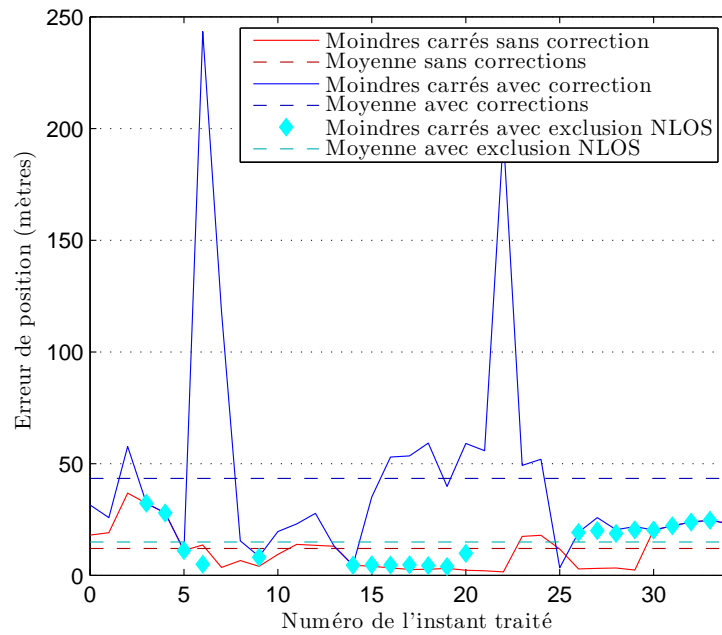
En plus de l'amélioration de la localisation obtenue, les avantages de la solution proposée sont :

- Une correction au lieu du rejet des NLOS, et donc pas de perte de disponibilité du service.
- Aucune donnée sur l'environnement à stocker.
- Pas de problème de mises à jour de données embarquées, le modèle est construit sur le terrain.
- Pas de problème de positionnement dans un modèle 3D, la position de référence est connue car le modèle 3D est construit autour d'elle.

Nous avons envisagé l'utilisation d'une méthode par *shadow matching* avec le modèle 3D que nous estimons dynamiquement. Comme nous l'indiquons dans le chapitre 4, les nuages locaux sont fusionnés au fur et à mesure du déplacement du véhicule. Toutefois, le modèle de l'environnement obtenu est précis et dense pour la partie de la rue déjà parcourue par le véhicule. La partie de la rue non encore traversée n'est que partiellement modélisée. En conséquence à cette contrainte, le *shadow matching* pourrait présenter des performances insuffisantes quand le modèle 3D de la zone urbaine d'intérêt est incomplet.



(a) Trajets vérité et estimés dans le plan horizontal Est-Nord.



(b) Erreur de position par rapport à la vérité RTK sans et avec correction.

FIGURE 5.12 – Estimation de la position pour une séquence dynamique, rue Roussel à Belfort, 25 août 2014.

CONCLUSION GÉNÉRALE ET PERSPECTIVES

À travers ce manuscrit a été présentée une méthode d'amélioration de la précision GNSS grâce à la vision 3D. C'est l'objectif du projet CAPLOC. De nombreuses étapes sont nécessaires pour proposer une solution autonome aux retards des pseudodistances. Les traitements impliqués concernent les domaines du calibrage, la détection de points d'intérêt, la segmentation d'images, la stéréovision, la 3D et la localisation GNSS.

6.1/ BILAN

En environnement urbain la localisation GNSS est très imprécise en raison de la structure des bâtiments et de leurs effets sur les signaux satellitaires. Outre l'atténuation induite par une diffraction ou une réflexion du signal, les signaux ayant subi une ou plusieurs réflexions allongent leur trajet à l'antenne et entraînent une sur-estimation de la pseudodistance correspondante. Or, ce sont les pseudodistances qui sont utilisées pour positionner le récepteur par trilatération. Les réflexions sont l'une des principales sources d'imprécision de la localisation GNSS en ville.

L'objectif du projet CAPLOC est d'améliorer la localisation en milieu urbain en modélisant les structures urbaines de manière dynamique au cours du trajet du véhicule puis en estimant et corrigeant les retards des rayons réfléchis lors du calcul de la position.

Nous proposons de retrouver la structure urbaine à partir de plusieurs vues de la scène acquises pendant le déplacement du véhicule. Ces vues sont fournies indistinctement par un stéréoscope fisheye, donnant deux vues simultanées, ou un système mono-caméra fisheye, donnant deux vues à deux instants différents (*Structure from Motion*).

Après une introduction, présentant le contexte et les outils à notre disposition et un état de l'art des différents domaines connexes à ce travail, nous présentons les généralités sur les caméras à objectif fisheye et notamment le modèle de projection sphérique. L'estimation de la structure 3D des éléments urbains est réalisée grâce à la mise en correspondance du

contenu des deux vues fournies par le système. Un pré-requis à cette étape d'appariement est l'estimation des paramètres de la projection mise en jeu par la caméra et les transformations permettant de relier le contenu des deux vues. Pour cela, nous avons proposé et évalué une méthode de calibrage automatique qui s'applique indifféremment dans le cas stéréo et dans le cas mono-caméra. Cette méthode permet de retrouver conjointement les paramètres liés au système objectif et caméra, et ceux décrivant la géométrie épipolaire de deux vues. Ce calibrage ne nécessite aucune intervention humaine et peut assurer un recalibrage régulier du système afin, par exemple, de compenser les vibrations subies par le système embarqué ou encore l'usage d'un objectif à focale variable.

Avec la connaissance du calibrage, une méthode de mise en correspondance tenant compte de la géométrie fisheye est étudiée. Elle est une adaptation d'une méthode basée sur la recherche du plus court chemin dans un graphe 2D construit à partir d'un couple de courbes épipolaires conjuguées. L'adaptation consiste à construire un graphe 3D à partir de plusieurs couples de courbes épipolaires conjuguées. Notre méthode comporte deux paramètres pour lesquelles nous avons défini le sous-espace des valeurs amenant aux résultats optimaux. Nous calculons une carte locale des disparités pour chaque couple d'images. Après une étape de réduction de bruit, les cartes locales sont fusionnées afin de produire un nuage de points 3D semi-dense de l'environnement.

Nous avons évalué la qualité des nuages de points ainsi produits sur des images de synthèse et des images réelles. Dans le dernier chapitre nous montrons que la qualité de la reconstruction est suffisante pour repositionner le véhicule dans son environnement et analyser les caractéristiques du canal de propagation entre les satellites et l'antenne GPS du véhicule. En effet, grâce à une détection des plans des façades des bâtiments, il nous est possible de définir si les satellites sont bloqués ou s'ils sont reçus par réflexion. Dans ce dernier cas, nous calculons le retard associé à ces signaux reçus et nous le corrigeons lors du calcul de la position. Même si nous considérons les cas jusqu'à 3 réflexions, les expérimentations ont montré que de telles corrections géométriques en dynamique ne permettent pas d'améliorer la précision de la localisation. Toutefois, nous avons montré une nette amélioration de la précision lorsque le véhicule est statique.

6.1.1/ RÉSULTATS

Un calibrage entièrement automatique utilisant un modèle de projection fisheye linéaire à un paramètre est proposé pour les projections à angle équisolide et stéréographique. Il converge généralement bien et est relativement précis. En revanche, il est moins robuste dans certaines conditions telles que :

- manque de textures dans les images ;
- présence de fenêtres produisant un effet miroir avec des reflets détaillées mais très différents entre les vues ;
- hauteur insuffisante des bâtiments qui engendre une scène projetée sur les bords du cercle

fish-eye ;

- présence de pluie sur l'objectif fish-eye ;
- sur-exposition des images, apparition de halos lumineux et, pour les capteurs CCD (exemple de la configuration (Cfg 1)), phénomène de *vertical smear* sur l'image.

À ces mauvaises conditions s'ajoutent celles liées à la vitesse du véhicule lorsque :

- le temps d'exposition de la caméra est trop long par rapport à la vitesse de déplacement du véhicule, ce qui génère un flou de bougé dans l'image ;
- le flou se ressent encore plus dans le cas d'une rotation car les traînées ne sont pas linéaires localement.

La mise en correspondance stéréo présentée permet de construire des modèles à l'échelle réelle avec une précision entre 1 et 2 mètres. Les meilleurs résultats sont obtenus lorsque les images sont en couleur et en utilisant une segmentation du ciel comme masque pour éliminer les propagations des structures estimées dans le ciel. Plus de 60% du nuage de points 3D estimé par cette méthode est évalué à moins de 50 centimètres de la vérité pour les scènes simulées présentées. Cette proportion est assez importante compte tenu des déformations infligées par le modèle de projection sphérique de nos objectifs et qu'une analyse théorique montre que l'on peut atteindre au mieux une précision d'environ 20 centimètres pour une correspondance parfaite. Il faut noter que nous parvenons à estimer la largeur des rues avec une très bonne précision, ce qui est primordial dans le cadre de notre application. En outre, ces mesures numériques tiennent compte des paramètres des scènes simulées qui sont construites pour des rues plus larges et des bâtiments plus hauts que celles et ceux rencontrés dans la plupart des centres-villes européens.

Nous avons proposé une méthode d'assemblage des nuages de points locaux basée sur l'estimation de la géométrie épipolaire du chapitre calibrage. Notre méthode est un peu moins précise que celle qui opère par décomposition de la matrice essentielle. Toutefois, elle montre une estimation plus stable et plus précise des épipôles. Nous parvenons à assembler les modèles locaux pour fournir un modèle complet du trajet avec une dérive modérée et cela sans étape d'affinement supplémentaire. Les résultats montrent un décalage final de position de 5 mètres environ pour une boucle de 250 mètres de longueur.

Enfin, en ce qui concerne l'application CAPLOC et le positionnement GNSS, nous avons montré la faisabilité de l'exploitation du modèle 3D pour l'estimation du retard des pseudodistances. Avec la méthode complète proposée l'imprécision du positionnement est diminuée de moitié, sans requérir ni le stockage de données supplémentaires déjà existantes telles que des modèles 3D des villes, ni le recalage du récepteur dans le modèle 3D car sa position est connue dans le repère local. Une évaluation des résultats avec correction des pseudodistances montre qu'un modèle de réflexion un peu plus sophistiqué améliore nettement les estimations du retard de pseudodistance. Le modèle allant jusqu'à 3 réflexions est plus efficace qu'un modèle plus simple à une réflexion fonctionnant sur le même principe. Le masque de visibilité issu des informations

3D extraites par la vision permet de plus de remplacer efficacement le seuil empirique sur le CN0 utilisé pour déterminer les signaux NLOS, d'autant plus que ce seuil est difficile à bien choisir.

6.1.2/ CONTRIBUTIONS

Les travaux effectués au cours de cette thèse touchent de nombreux domaines de la vision par ordinateur ainsi que la localisation GNSS en milieu urbain. Les contributions peuvent être classées en fonction de la compétence concernée :

- La simulation de vision fisheye :
 - Scènes fisheye simulées photoréalistes (avec simulation des défauts optiques) pour les modèles de projection équidistante et à angle équisolide utilisées pour les évaluations.
- Le calibrage fisheye entièrement automatique et l'estimation de la pose :
 - Détection automatique du cercle du contour de l'image fisheye par segmentation GrabCut pour éviter toute intervention de l'utilisateur.
 - Adaptation de l'algorithme des 9 points de [Mičušík, 2004] pour les modèles linéaires de projection à angle équisolide et de projection stéréographique.
 - Application du LO-RANSAC au calibrage automatique fisheye.
 - Recherche et preuve à travers des évaluations détaillées de l'algorithme de points d'intérêt classique optimal pour le fisheye : le SIFT.
 - Affinement du calibrage fisheye avec l'algorithme de Levenberg-Marquardt pour améliorer la précision en tenant compte de tous les *inliers* et pas seulement des quelques points tirés aléatoirement pour l'estimation initiale.
 - Estimation géométrique des épipôles et des rotations entre les caméras avec le modèle sphérique.
- La mise en correspondance de stéréovision fisheye :
 - Indexation des courbes épipolaires du modèle sphérique autour du cercle équateur entre les épipôles.
 - Étude approfondie des adaptations à apporter à la correspondance par programmation dynamique pour l'appliquer correctement à la vision omnidirectionnelle.
 - Amélioration de la correspondance par programmation dynamique avec la définition d'un graphe 3D tenant compte des courbes épipolaires voisines.
 - Estimation de la précision théorique atteignable de l'estimation des distances dans un plan épipolaire pour une distance de référence donnée et en fonction de l'entraxe entre les caméras et de leur résolution.
 - Procédure d'évaluation sur les cartes des distances fisheye basées sur les scènes simulées.

- Évaluation de l'échelle du modèle 3D reconstruit par mesure de la largeur de la rue à partir de la répartition globale des données du nuage de points selon une loi normale.
- Application d'un masque du ciel en post-traitement à la correspondance pour réduire les erreurs de propagation des appariements dans le ciel. Les masques présentés sont détectés par segmentation MeanShift sur des images couleur uniquement.
- L'extraction des plans :
 - Extraction des plans par détection des segments par transformée de Hough probabiliste dans les empreintes au sol des bâtiments et en utilisant les cartes des hauteurs.
- L'amélioration de la position GNSS :
 - Amélioration de la localisation sans besoin de données cartographiques embarquées ni d'autres connaissances à priori.
 - Création de masques de visibilité des satellites dans le diagramme *skypplot* à la position du récepteur, à partir des plans détectés dans le modèle 3D généré par stéréovision fisheye.

6.2/ PERSPECTIVES

La procédure complète que nous avons développé fait appel à de nombreuses méthodes et algorithmes qui touchent à des domaines très variés tels que la projection fisheye, la 3D, la fusion de capteurs, la robotique et la localisation. Chaque algorithme peut être poussé ou modifié pour apporter des résultats optimaux pour l'application visée et pour d'autres applications. Cela implique donc de nouvelles procédures d'évaluation plus complètes et plus précises et des améliorations directes des algorithmes choisis.

Le point particulier de l'évaluation s'adresse aux modèles 3D que nous avons obtenus sur les scènes réelles. Nous avons mené une évaluation du paramètre largeur de la rue. Une solution complète serait de construire une vérité terrain à l'aide d'un lidar permettant de travailler à l'échelle réelle. Dans le cadre des travaux de cette thèse des acquisitions au Velodyne HDL-32E ont été faites (voir la figure 1.2 dans l'introduction du manuscrit) mais l'assemblage des données demande un travail supplémentaire important qui n'a pas pu être finalisé. Le Velodyne était placé de sorte à avoir la rotation selon l'axe horizontal dans la direction de la rue pour être sûr de capturer les bâtiments tranche par tranche sur toute leur hauteur. Il faut par ailleurs calibrer le Velodyne avec les caméras pour estimer leurs poses et orientations relatives, et être capable d'assembler correctement les tranches de points 3D obtenues avec l'avancement variable du véhicule. Un travail supplémentaire pourrait être mené dans ce sens. En outre, l'utilisation de modèles très précis produits par l'IGN via une acquisition sol serait un atout majeur pour mener cette évaluation.

Les travaux présentés peuvent être étendus et optimisés pour chacune des étapes de la procédure du système. Ce qui les rend particulièrement intéressants pour la recherche c'est qu'ils permettent d'innombrables pistes d'évolutions et d'améliorations telles que :

- Utiliser ou développer des algorithmes de détection et description des points d'intérêts spécialisés pour les images fisheye, et donc tenant compte de la géométrie particulière des modèles de projection fisheye.
- Proposer un calibrage avec un modèle linéaire à un paramètre accompagné d'un terme supplémentaire également à un paramètre pour une précision optimale et un nombre minimal de points nécessaires au RANSAC. Cette solution semble à privilégier par rapport à l'utilisation d'un modèle polynomial nécessitant plus de deux paramètres pour être efficace.
- Estimer plus précisément ou affiner l'estimation du centre optique des images fisheye au calibrage.
- Gérer aussi le cisaillement (*skew*) des pixels du capteur pour un calibrage optimisé au maximum. Peut se faire par une détection d'ellipse au lieu de cercle du contour de l'image.
- Développer des algorithmes de mise en correspondance plus performants dans le cas fisheye. La coupure de graphe semble être un choix pertinent de correspondance dense.
- Développer des algorithmes de mise en correspondance éparses doit être aussi une bonne alternative. La fiabilité des points mis en correspondance avec une méthode éparses est plus grande et son temps d'exécution beaucoup plus rapide. La méthode éparses peut aussi devenir semi-dense en appliquant une stratégie de propagation de croyance.
- Ajouter des contraintes supplémentaires afin de tenir compte de la différence des textures locales dans les images : éviter les propagations des structures estimées dans le ciel.
- Appliquer des pré-traitement des images sources pour réduire le bruit ou faire ressortir les détails, ainsi que des post-traitements de lissage des cartes des disparités ou des distances pour améliorer les correspondances et aboutir à des estimations des distances moins bruitées.
- Trouver une méthode de détection de l'axe vertical par vision pour une meilleure fiabilité que de prendre l'axe Z de la première caméra, ceci si on cherche à éviter l'ajout d'un gyroscope.
- Appliquer une approche globale avec suivi temporel des points et des paramètres pour le calibrage, l'odométrie visuelle, la mise en correspondance, etc.
- Lisser les points 3D estimés avec le traitement des nouvelles vues en *Structure from Motion*.
- Mettre en œuvre des méthodes de consolidation des nuages de points pour un assemblage et un modèle global optimal, avec affinement des points obtenus.
- Appliquer une extraction des images-clef à utiliser dans le cadre d'applications *Structure from Motion* avec la configuration vidéo binoculaire (Cfg 1).

- Exploiter la méthode de construction 3D proposée pour cumuler la localisation dans le modèle à la localisation GNSS, en plus de se limiter à améliorer cete dernière.
- De même, appliquer une procédure de détection des boucles pour recalibrer la cartographie et en augmenter la précision, voire ouvrir des portes à de nouvelles applications de recherche de trajet et de SLAM (*Simultaneous Localisation And Mapping*).
- Détecter plus précisément les plans, en particulier leur hauteur qui est pour le moment déterminée de manière très simple.
- Détecter le tracé de l'horizon dans les images fisheye pour obtenir plus précisément le haut des bâtiments, voire évoluer vers une méthode différente d'estimation de plans verticaux en 3D, par correspondance des plans entre deux points de vue.
- Exploiter ou estimer automatiquement l'orientation du Nord géographique à partir des données du récepteur GNSS pour le positionnement des satellites dans le masque d'occlusion et le *skyplot*.
- En plus de l'évaluation des positions 2D dans le plan horizontal extraites à partir des positions 3D complètes, évaluer la méthode pour la localisation 3D pour tenir compte de l'altitude, ainsi que la localisation calculée directement en 2D, ce qui permet de n'avoir besoin que de 3 signaux satellitaires au moins et d'augmenter la disponibilité du service. Le positionnement 2D suffit pour de nombreuses applications de localisation du véhicule en environnement urbain.
- Utiliser des modèles d'estimation des retards de pseudodistance plus précis tenant compte des réflexions et des diffractions.
- Étudier quelle serait l'erreur de synchronisation minimale à avoir entre les données vision et GNSS traitées et chercher à l'atteindre.
- Adapter ou enrichir la méthode de correction des retards afin d'améliorer la précision de la position GNSS lorsque le véhicule est en mouvement. Pour cela, il est indispensable de bien comprendre l'effet des retards au sein même du récepteur.
- Étudier la réflectivité des surfaces des obstacles avec une étude des textures dans les images pour des modèles plus précis d'étude des réflexions.
- Faire une implémentation de la procédure complète pour une exécution plus rapide voire temps réel. Cela peut être réalisé en faisant appel à des architectures multi-processeurs CPU, GPU ou FPGA. Une telle implémentation demanderait l'adéquation entre les algorithmes et l'architecture et peut-être le remplacement de certains algorithmes par d'autres plus rapides et parfois moins efficaces.

BIBLIOGRAPHIE

- [Nav, 1996] (1996). **Navstar GPS User Equipment Introduction Public Release Version**.
- [Bos, 2009] (2009). **Boss project - on board wireless secured video surveillance**.
<http://www.multitel.be/BOSS/>.
- [Abraham et al., 2005] Abraham, S., et Forstner, W. (2005). **Fish-eye-stereo calibration and epipolar rectification**. *ISPRS Journal of Photogrammetry & Remote Sensing*, 59 :278–288.
- [Adjrad et al., 2015] Adjrad, M., et Groves, P. D. (2015). **Enhancing conventional gnss positioning with 3d mapping without accurate prior knowledge**. *Proceedings of the 28th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2015)*, pages 2397–2409. Tampa, Florida.
- [Agrawal et al., 2008] Agrawal, M., Konolige, K., et Blas, M. (2008). **Censure : Center surround extremas for realtime feature detection and matching**. Dans Forsyth, D., Torr, P., et Zisserman, A., éditeurs, *Computer Vision – ECCV 2008*, volume 5305 de *Lecture Notes in Computer Science*, pages 102–115. Springer Berlin Heidelberg.
- [Alahi et al., 2012] Alahi, A., Ortiz, R., et Vandergheynst, P. (2012). **Freak : Fast retina keypoint**. Dans *2012 IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR)*, IEEE Conference on Computer Vision and Pattern Recognition, pages 510–517, 345 E 47TH ST, NEW YORK, NY 10017 USA. IEEE, IEEE. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, JUN 16-21, 2012.
- [Anil Adhyapak et al., 2007] Anil Adhyapak, S., Kehtarnavaz, N., et Nadin, M. (2007). **Stereo matching via selective multiple windows**. *Journal of Electronic Imaging*, 16(1) :1–14.
- [Arican, 2010] Arican, Z. (2010). **Analysis of Multiview Omnidirectional Images in a Spherical Framework**. PhD thesis, École Polytechnique Fédérale de Lausanne.
- [Arican et al., 2010] Arican, Z., et Frossard, P. (2010). **Dense depth estimation from omnidirectional images**. *Journal of Computer Vision and Image Understanding*.
- [Attia, 2013] Attia, D. (2013). **Segmentation d’images par combinaison adaptative couleur-texture et classification de pixels : Applications à la caractérisation de l’environnement de réception de signaux GNSS**. PhD thesis. Thèse de doctorat dirigée par Meurie, CyrilRuichek, Yassine et Marais, Juliette Informatique Belfort-Montbéliard 2013.
- [Aydin et al., 2010] Aydin, T., et Akgul, Y. S. (2010). **Stereo depth estimation using synchronous optimization with segment based regularization**. *Pattern Recognition Letters*, 31 :2389–2396.

- [Barnard, 1986] Barnard, S. T. a. (1986). **A stochastic approach to stereo vision**. *AAAI Proceedings*, pages 676–680.
- [Barreto, 2006] Barreto, J. P. (2006). **A unifying geometric representation for central projection systems**. *Computer Vision and Image Understanding*, 103(3) :208 – 217. Special issue on Omnidirectional Vision and Camera Networks.
- [Barreto et al., 2005] Barreto, J. P., et Daniilidis, K. (2005). **Fundamental matrix for cameras with radial distortion**. Dans *Proc. ICCV*, pages 625–632.
- [Baselgia, 2010] Baselgia, C. (2010). **Fast camera motion estimation for visual navigation**. Master's thesis, Swiss Federal Institute of Technology Zurich.
- [Bastanlar et al., 2008] Bastanlar, Y., Puig, L., Sturm, P., Guerrero, J., et Barreto, J. P. (2008). **Dlt-like calibration of central catadioptric cameras**. Dans *OMNIVIS 2008 - 8th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras*, Marseille, France. Rahul Swaminathan and Vincenzo Caglioti and Antonis Argyros.
- [Bay et al., 2006] Bay, H., Tuytelaars, T., et Van Gool, L. (2006). **Surf : Speeded up robust features**. Dans *Proceedings of the 9th European Conference on Computer Vision - Volume Part I, ECCV'06*, pages 404–417, Berlin, Heidelberg. Springer-Verlag.
- [Bazin et al., 2011] Bazin, J.-C., et Kweon, I. S. (2011). **Catadioptric Vision for Robotic Applications : From Low-Level Feature Extraction to Visual Odometry**. Lambert Academic Publishing.
- [Belli et al., 2000] Belli, T., Cord, M., et Philipp-Foliguet, S. (2000). **Colour contribution for stereo image matching**. Dans *proceedings of International Conference on Color in Graphics and Image Processing, CGIP*, pages 317–322, Saint-Étienne, France.
- [Bleyer et al., 2005] Bleyer, M., et Gelautz, M. (2005). **A layered stereo matching algorithm using image segmentation and global visibility constraints**. *{ISPRS} Journal of Photogrammetry and Remote Sensing*, 59(3) :128 – 150.
- [Bourdeau et al., 2012] Bourdeau, A., Sahmoudi, M., et Tourneret, J.-Y. (2012). **Constructive use of gnss nlos-multipath : Augmenting the navigation kalman filter with a 3d model of the environment**. Dans *Information Fusion (FUSION), 2012 15th International Conference on*, pages 2271–2276.
- [Bourmaud, 2015] Bourmaud, G. (2015). **Estimation de paramètre évoluant sur des groupes de Lie : Application à la cartographie et à la localisation d'une caméra monoculaire**. Theses, Université de Bordeaux.
- [Braquelaire et al., 1997] Braquelaire, J.-P., et Brun, L. (1997). **Comparison and optimization of methods of color image quantization**. *Image Processing, IEEE Transactions on*, 6(7) :1048–1052.
- [Brockers et al., 2004] Brockers, R., Hund, M., et Mertsching, B. (2004). **Fast stereo vision for mobile robots by global minima of cost functions**. Rapport technique, GET Lab, University of Paderborn, Pohlweg 47-49, 33098 Paderborn, Germany.

- [Bunschoten et al., 2003] Bunschoten, R., et Kröse, B. (2003). **Visual odometry from an omnidirectional vision system**. Dans *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, volume 1, pages 577–583 vol.1.
- [Bétaille et al., 2014] Bétaille, D., Peyret, F., et Ortiz, M. (2014). **How to enhance accuracy and integrity of satellite positioning for mobility pricing in cities : the urban trench method**. Dans *Transport Research Arena 2014*, Paris.
- [Bétaille et al., 2013] Bétaille, D., Peyret, F., Ortiz, M., Miquel, S., et Fontenay, L. (2013). **A new modeling based on urban trenches to improve gnss positioning quality of service in cities**. *Intelligent Transportation Systems Magazine, IEEE*, 5(3) :59–70.
- [Calonder et al., 2010] Calonder, M., Lepetit, V., Strecha, C., et Fua, P. (2010). **Brief : Binary robust independent elementary features**. Dans *Proceedings of the 11th European Conference on Computer Vision : Part IV, ECCV'10*, pages 778–792, Berlin, Heidelberg. Springer-Verlag.
- [Cannelle et al., 2012] Cannelle, B., Paparoditis, N., Pierrot-Deseilligny, M., et Papelard, J.-P. (2012). **Off-line vs. on-line calibration of a panoramic-based mobile mapping system**. Dans *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, volume I-3, Melbourne, Australia.
- [Carneiro et al., 2004] Carneiro, G., et Jepson, A. D. (2004). **Pruning local feature correspondences using shape context**. Rapport technique, Department of Computer Science, University of Toronto, Toronto, ON, Canada.
- [Caron et al., 2012] Caron, G., Dame, A., et Marchand, E. (2012). **L'information mutuelle pour l'estimation visuelle directe de pose**. Dans *Actes de la conférence RFIA 2012*, pages 978–2–9539515–2–3, Lyon, France. Session "Articles".
- [Caron et al., 2011] Caron, G., et Eynard, D. (2011). **Étalonnage simultané de systèmes stéréoscopiques hybrides**. Rapport technique, Université de Picardie Jules Verne, laboratoire MIS, 33 rue Saint-Leu 80039 Amiens.
- [Caruso et al., 2015] Caruso, D., Engel, J., et Cremers, D. (2015). **Large-scale direct slam for omnidirectional cameras**. Dans *International Conference on Intelligent Robots and Systems (IROS)*.
- [Chambon, 2005] Chambon, S. (2005). **Mise en correspondance stéréoscopique d'images couleur en présence d'occultations**. PhD thesis, École Doctorale Informatique et Télécommunications, Université Paul Sabatier – Toulouse III, Institut de Recherche en Informatique de Toulouse — UMR 5505.
- [Chambon et al., 2004] Chambon, S., et Crouzil, A. (2004). **Mesures de corrélation pour des images couleur**. *Traitement du signal, numéro spécial L'image numérique couleur*, pages 635–659.
- [Charles, 2009] Charles, J. R. (2009). **Review of samyang 8 mm f/3.5 proportional projection ultra-wide angle lens**. Rapport technique.

- [Chum, 2005] Chum, O. (2005). **Two-View Geometry Estimation by Random Sample and Consensus**. PhD thesis, Faculty of the Electrical Engineering of the Czech Technical University in Prague.
- [Claus et al., 2005] Claus, D., et Fitzgibbon, A. W. (2005). **A rational function lens distortion model for general cameras**. Dans *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 213–219 vol. 1.
- [Comaniciu et al., 2002] Comaniciu, D., et Meer, P. (2002). **Mean shift : a robust approach toward feature space analysis**. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5) :603–619.
- [Courbon et al., 2012] Courbon, J., Mezouar, Y., et Martinet, P. (2012). **Evaluation of the unified model of the sphere for fisheye cameras in robotic applications**. *Advanced Robotics*, 26(8-9) :947–967.
- [Cuscó Mach, 2009] Cuscó Mach, J. (2009). **Dense disparity estimation for spherical images based on belief propagation**. Master's thesis, École Polytechnique Fédérale de Lausanne.
- [De Luca, 2006] De Luca, L. (2006). **Relevé et multi-représentations du patrimoine architectural Définition d'une approche hybride de reconstruction 3D d'édifices**. PhD thesis, École Nationale Supérieure d'Arts et Métiers, Centre d'Aix-en-provence.
- [Deng et al., 2005] Deng, Y., Yang, Q., Lin, X., et Tang, X. (2005). **A symmetric patch-based correspondence model for occlusion handling**. Dans *ICCV*, page 1316–1322. Association for Computing Machinery, Inc.
- [Deschaud, 2010] Deschaud, J.-E. (2010). **Traitements de nuages de points denses et modélisation 3D d'environnements par système mobile LiDAR/Caméra**. PhD thesis, École nationale supérieure des mines de Paris, MINES ParisTech, Centre de Robotique - Département Mathématiques et Systèmes.
- [Deschaud et al., 2010] Deschaud, J.-E., et Goulette, F. (2010). **A fast and accurate plane detection algorithm for large noisy point clouds using filtered normals and voxel growing**. Dans *3DPVT*, Paris, France.
- [Devernay et al., 2001] Devernay, F., et Faugeras, O. (2001). **Straight lines have to be straight : automatic calibration and removal of distortion from scenes of structured environments**. *Machine Vision and Applications*, 13(1) :14–24. Erratum available at <http://devernay.free.fr/publis/distcalib-erratum.html>.
- [Ding et al., 2011] Ding, J., Liu, J., Zhou, W., Yu, H., Wang, Y., et Gong, X. (2011). **Real-time stereo vision system using adaptive weight cost aggregation approach**. *EURASIP Journal on Image and Video Processing*, 20 :1–19.
- [Ercek et al., 2005] Ercek, R., De Doncker, P., et Grenez, F. (2005). **Study of pseudo-range error due to non-line-of-sight-multipath in urban canyons**. Dans *Proceedings of the 18th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS '05)*, Long Beach, California, USA.

- [Eudes, 2011] Eudes, A. (2011). **Localisation et cartographie simultanées par ajustement de faisceaux local : propagation d'erreurs et réduction de la dérive à l'aide d'un odomètre**. PhD thesis, Université Blaise Pascal - Clermont-Ferrand II.
- [Eynard, 2011] Eynard, D. (2011). **Capteur de stéréovision hybride pour la navigation des drones**. PhD thesis, École Doctorale Sciences et Santé, Université de Picardie Jules Verne, Laboratoire Modélisation, Information et Systèmes.
- [Fabbri et al., 2010] Fabbri, R., et Kimia, B. (2010). **3d curve sketch : Flexible curve-based stereo reconstruction and calibration**. Dans *CVPR*.
- [Fakhfakh, 2011] Fakhfakh, N. (2011). **Détection et localisation tridimensionnelle par stéréovision d'objets en mouvement dans des environnements complexes Application aux passages à niveau**. PhD thesis, École Centrale de Lille, Laboratoires ESTAS et LEOST de l'Institut Français des Sciences et Technologies des Transports, de l'Aménagement et des Réseaux (IFSTTAR), Laboratoire LPM de l'Ecole Polytechnique Fédérale de Lausanne (EPFL), Laboratoire LAGIS de l'Ecole Centrale de Lille.
- [Feng et al., 2014] Feng, C., Taguchi, Y., et Kamat, V. R. (2014). **Fast plane extraction in organized point clouds using agglomerative hierarchical clustering**. Dans *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 6218–6225.
- [Fisher et al., 2004] Fisher, R., Perkins, S., Walker, A., et Wolfart, E. (2004). **Hypermedia image processing reference 2 (hipr2)**. Website : http://homepages.inf.ed.ac.uk/rbf/HIPR2/hipr_top.htm.
- [Fleck et al., 2009] Fleck, S., Busch, F., Biber, P., et Straßer, W. (2009). **Graph cut based panoramic 3d modeling and ground truth comparison with a mobile platform – the wägele**. *Image and Vision Computing*, 27 :141–152.
- [Forstmann et al., 2004] Forstmann, S., Kanou, Y., Ohya, J., Thuring, S., et Schmitt, A. (2004). **Real-time stereo by using dynamic programming**. Dans *Computer Vision and Pattern Recognition Workshop*.
- [Früh et al., 2004] Früh, C., et Zakhor, A. (2004). **An automated method for large-scale, ground-based city model acquisition**. *International Journal of Computer Vision*, 60(1) :5–24.
- [Gehrig, 2005] Gehrig, S. K. (2005). **Large-field-of-view stereo for automotive applications**. Rapport technique, DaimlerChrysler.
- [Gehrig et al., 2008] Gehrig, S. K., Rabe, C., et Krüger, L. (2008). **6d vision goes fisheye for intersection assistance**. Dans *Canadian Conference on Computer and Robot Vision*, pages 34–41.
- [Geyer et al., 2000] Geyer, C., et Daniilidis, K. (2000). **A unifying theory for central panoramic systems and practical implications**. Dans *ECCV*, pages 445–461.
- [Geyer et al., 2007] Geyer, C., et Stewenius, H. (2007). **A nine-point algorithm for estimating para-catadioptric fundamental matrices**. Dans *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8.

- [Gonzalez-Barbosa, 2004] Gonzalez-Barbosa, J.-J. (2004). **Vision panoramique pour la robotique mobile : stéréovision et localisation par indexation d'images**. PhD thesis, École doctorale en informatique et télécommunications, Laboratoire d'Analyse et d'Architecture des Systèmes, Université Toulouse III.
- [Groves et al., 2015] Groves, P. D., Wang, L., Adjrad, M., et Ellul, C. (2015). **Gnss shadow matching : The challenges ahead**. *Proceedings of the 28th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2015)*, pages 2421–2443. Tampa, Florida.
- [Gupta et al., 2010] Gupta, R. K., et Cho, S.-Y. (2010). **Real-time stereo matching using adaptive binary window**. Rapport technique, Forensics and Security Laboratory, School of Computer Engineering, Nanyang Technological University - Singapore.
- [Gutiérrez et al., 2004] Gutiérrez, S., et Marroquín, J. L. (2004). **Robust approach for disparity estimation in stereo vision**. *Image and Vision Computing*, 22(3) :183 – 195.
- [Harris et al., 1988] Harris, C., et Stephens, M. (1988). **A combined corner and edge detector**. Dans *In Proc. of Fourth Alvey Vision Conference*, pages 147–151.
- [Hartley, 1997] Hartley, R. (1997). **In defense of the eight-point algorithm**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6) :580 –593.
- [Hartley et al., 2004] Hartley, R., et Zisserman, A. (2004). **Multiple View Geometry in Computer Vision**. Cambridge University Press, second édition.
- [Herrera et al., 2011] Herrera, P. J., Pajares, G., Guijarro, M., Ruz, J. J., et De La Cruz, J. M. (2011). **A stereovision matching strategy for images captured with fish-eye lenses in forest environments**. *Sensors*, 11 :1756–1783.
- [Herrera et al., 2009] Herrera, P. J., Pajares, G., Guijarro, M., Ruz, J. J., De La Cruz, J. M., et Montes, F. (2009). **A featured-based strategy for stereovision matching in sensors with fish-eye lenses for forest environments**. *Sensors*, 9 :9468–9492.
- [Ho et al., 2005] Ho, T.-H., Davis, C. C., et Milner, S. D. (2005). **Using geometric constraints for fisheye camera calibration**. Dans *Proc. IEEE OMNIVIS Workshop*.
- [Holz et al., 2014] Holz, D., et Behnke, S. (2014). **Approximate triangulation and region growing for efficient segmentation and smoothing of range images**. *Robot. Auton. Syst.*, 62(9) :1282–1293.
- [Horaud et al., 1995] Horaud, R., et Monga, O. (1995). **Vision par ordinateur : outils fondamentaux**. Éditions Hermès, second édition.
- [Hosni et al., 2010] Hosni, A., Bleyer, M., et Gelautz, M. (2010). **Near real-time stereo with adaptive support weight approaches**. Rapport technique, Institute for Software Technology and Interactive Systems, Vienna University of Technology.
- [Howard, 2008] Howard, A. (2008). **Real-time stereo visual odometry for autonomous ground vehicles**. Dans *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 3946–3952.

- [Hrabar et al., 2005] Hrabar, S., Sukhatme, G. S., Corke, P., Usher, K., et Roberts, J. (2005). **Combined optic-flow and stereo-based navigation of urban canyons for a uav**. Dans *IEEE International Conference on Intelligent Robots and Systems*.
- [Hsu et al., 2015] Hsu, L.-T., Gu, Y., et Kamijo, S. (2015). **Nlos correction/exclusion for gnss measurement using rain and city building models**. *Sensors*, 15(7) :17329.
- [Hughes et al., 2010] Hughes, C., Denny, P., Jones, E., et Glavin, M. (2010). **Accuracy of fish-eye lens models**. *Applied optics*, 49 :3338–3347.
- [Hulik et al., 2014] Hulik, R., Spanel, M., Smrz, P., et Materna, Z. (2014). **Continuous plane detection in point-cloud data based on 3d hough transform**. *Journal of Visual Communication and Image Representation*, 25(1) :86 – 97. Visual Understanding and Applications with RGB-D Cameras.
- [Illingworth et al., 1988] Illingworth, J., et Kittler, J. (1988). **A survey of the hough transform**. *Computer Vision, Graphics, and Image Processing*, 44(1) :87 – 116.
- [Ishikawa et al., 1998] Ishikawa, H., et Geiger, D. (1998). **Occlusions, discontinuities, and epipolar lines in stereo**. Dans *European Conference on Computer Vision*, pages 232–248.
- [Joseph, 2010] Joseph, A. (2010). **What is the difference between snr and C/N_0 ?** *Inside GNSS*, pages 20–25.
- [Kalarot et al., 2010] Kalarot, R., et Morris, J. (2010). **Implementation of symmetric dynamic programming stereo matching algorithm using cuda**. Dans *Proc. 16th Korea-Japan Joint Workshop on Frontiers of Computer Vision*. FCV.
- [Kawanishi et al., 2009] Kawanishi, R., Yamashita, A., et Kaneko, T. (2009). **Three-dimensional environment model construction from an omnidirectional image sequence**. Rapport technique, Department of Mechanical Engineering, Shizuoka University, Shizuoka, Japan.
- [Kita, 2011] Kita, N. (2011). **Direct floor height measurement for biped walking robot by fisheye stereo**. Dans *IEEE-RAS International Conference on Humanoid Robots*, pages 187–192.
- [Kotthäuser et al., 2013] Kotthäuser, T., Divband Soorati, M., et Mertsching, B. (2013). **Automatic reconstruction of polygonal room models from 3d point clouds**. Dans *Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on*, pages 661–667.
- [Krüger et al., 2004] Krüger, L., Wöhler, C., Würz-wessel, E., et Stein, F. (2004). **Infactory calibration of multiocular camera systems**. Dans *SPIE Photonics Europe (Optical Metrology in Production Engineering)*, pages 126–137.
- [Kumar et al., 2014] Kumar, A., Sato, Y., Oishi, T., et Ikeuchi, K. (2014). **Identifying reflected gps signals and improving position estimation using 3d map simultaneously built with laser range scanner**. Rapport technique, Computer Vision Laboratory, Institute of Industrial Science, The University of Tokyo.
- [Langley, 1999] Langley, R. B. (1999). **Dilution of precision**. *GPS World*, pages 52–59.

- [Lee et al., 2002] Lee, S. H., Kanatsugu, Y., et Park, J.-I. (2002). **Map-based stochastic diffusion for stereo matching and line fields estimation**. *International Journal of Computer Vision*, 47 :195–218.
- [Lefebvre, 2008] Lefebvre, S. (2008). **Approche monodimensionnelle de la mise en correspondance stéréoscopique par corrélation Application à la détection d'obstacles routiers**. PhD thesis, Université des Sciences et Technologies de Lille U.F.R. d'Informatique, Electronique, Electrotechnique et Automatique Ecole Doctorale des Sciences Pour l'Ingénieur, Laboratoire d'Electronique, Ondes et Signaux pour les Transports de l'INRETS.
- [Lemonde, 2005] Lemonde, V. (2005). **Stéréovision embarquée sur véhicule : de l'auto-calibrage à la détection d'obstacles**. Theses, INSA de Toulouse.
- [Leutenegger et al., 2011] Leutenegger, S., Chli, M., et Siegwart, R. Y. (2011). **Brisk : Binary robust invariant scalable keypoints**. Dans *Proceedings of the 2011 International Conference on Computer Vision, ICCV '11*, pages 2548–2555, Washington, DC, USA. IEEE Computer Society.
- [Lhuillier, 2005] Lhuillier, M. (2005). **Automatic structure and motion using a catadioptric camera**. Dans *Workshop on Omnidirectional Vision, Camera networks and Non-classical cameras OMNIVIS'05*.
- [Lhuillier, 2011] Lhuillier, M. (2011). **A generic error model and its application to automatic 3d modeling of scenes using a catadioptric camera**. *International Journal of Computer Vision*, 91(2) :175–199.
- [Lhuillier et al., 2004] Lhuillier, M., et Quan, L. (2004). **Reconstruction quasi-dense et modèles 3d à partir d'une séquence d'images**. *actes du Congrès AFRIF-AFIA Reconnaissance des Formes et Intelligence Artificielle, RFIA*, 2 :895–904.
- [Li, 2006a] Li, S. (2006a). **Monitoring around a vehicle by a spherical image sensor**. *IEEE Transactions on Intelligent Transportation Systems*, 7(4) :541–550.
- [Li, 2006b] Li, S. (2006b). **Real-time spherical stereo**. Rapport technique, Faculty of Engineering, Iwate University, Japan.
- [Li, 2008] Li, S. (2008). **Binocular spherical stereo**. *IEEE Transactions On Intelligent Transportation Systems*, 9 :589–600.
- [Li et al., 2005] Li, S., et Fukumori, K. (2005). **Spherical stereo for the construction of immersive vr environment**. Dans *Proceedings of the IEEE Virtual Reality*, pages 217–222.
- [Lin et al., 2008] Lin, T.-T., Hsiung, Y.-K., Hong, G.-L., Chang, H.-K., et Lu, F.-M. (2008). **Development of a virtual reality gis using stereo vision**. *Computers and Electronics in Agriculture*, 63 :38–48.
- [Lowe, 2004] Lowe, D. G. (2004). **Distinctive image features from scale-invariant keypoints**. *International Journal of Computer Vision*, 60(2) :91–110.
- [Marais, 2002] Marais, J. (2002). **Localisation de mobiles terrestres par satellites : mise en oeuvre d'outils permettant l'analyse de l'influence des conditions de propagation**

- et des effets de masques sur la disponibilité du service offert. PhD thesis, Université Lille1 - Sciences et Technologies.
- [Marais et al., 2012] Marais, J., Ambellouis, S., Flancquart, A., Lefebvre, S., Meurie, C., et Ruichek, Y. (2012). **Accurate localisation based on gnss and propagation knowledge for safe applications in guided transport**. *Procedia - Social and Behavioral Sciences*, 48(0) :796 – 805. Transport Research Arena 2012.
- [Marais et al., 2005] Marais, J., Berbineau, M., et Nahimana, D. F. (2005). **Effect of alternate path on satellite positioning accuracy in an urban environment**. Dans *International Communications Satellite Systems Conference (ICSSC) 2005, Rome, Italy*, page 12p.
- [Marais et al., 2014] Marais, J., Meurie, C., Attia, D., Ruichek, Y., et Flancquart, A. (2014). **Toward accurate localization in guided transport : Combining gnss data and imaging information**. *Transportation Research Part C : Emerging Technologies*, 43, Part 2(0) :188 – 197. Special Issue with Selected Papers from Transport Research Arena.
- [Matas et al., 2000] Matas, J., Galambos, C., et Kittler, J. (2000). **Robust detection of lines using the progressive probabilistic hough transform**. *Computer Vision and Image Understanding*, 78(1) :119 – 137.
- [Mayer, 2003] Mayer, H. (2003). **Analysis of means to improve cooperative disparity estimation**. Dans *proceedings of ISPRS Conference on Photogrammetric Image Analysis, PIA*, page 25–31.
- [Megyesi et al., 2004] Megyesi, Z., et Chetverikov, D. (2004). **Affine propagation for surface reconstruction in wide baseline stereo**. Dans *IEEE Conference Proceedings of International Conference on Pattern Recognition, ICPR*, volume 4, pages 76–79.
- [Mei et al., 2007] Mei, C., et Rives, P. (2007). **Single view point omnidirectional camera calibration from planar grids**. Dans *IEEE International Conference on Robotics and Automation*.
- [Mikolajczyk et al., 2005] Mikolajczyk, K., et Schmid, C. (2005). **A performance evaluation of local descriptors**. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(10) :1615–1630.
- [Milella et al., 2006] Milella, A., et Siegwart, R. (2006). **Stereo-based ego-motion estimation using pixel tracking and iterative closest point**. Dans *Computer Vision Systems, 2006 ICVS '06. IEEE International Conference on*, pages 21–21.
- [Miura et al., 2015] Miura, S., Hsu, L.-T., Chen, F., et Kamijo, S. (2015). **Gps error correction with pseudorange evaluation using three-dimensional maps**. *IEEE Transactions on Intelligent Transportation Systems*, 16(6) :3104–3115.
- [Mičušík, 2004] Mičušík, B. (2004). **Two-View Geometry of Omnidirectional Cameras**. PhD thesis, Czech Technical University in Prague.
- [Mičušík et al., 2004] Mičušík, B., Martinec, D., et Pajdla, T. (2004). **3d metric reconstruction from uncalibrated omnidirectional images**. Dans *Asian Conference on Computer*

Vision, Jeju Island, Korea. Center for Machine Perception, Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University.

- [Mičušík et al., 2003] Mičušík, B., et Pajdla, T. (2003). **Estimation of omnidirectional camera model from epipolar geometry**. Dans *Conference on Computer Vision and Pattern Recognition*.
- [Mordohai et al., 2006] Mordohai, P., et Medioni, G. (2006). **Stereo using monocular cues within the tensor voting framework**. *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 28(6) :968–982.
- [Mouaddib, 2005a] Mouaddib, E. M. (2005a). **Introduction à la vision panoramique catadioptrique**. *Traitement du Signal*, 22 :409–417.
- [Mouaddib, 2005b] Mouaddib, E. M. (2005b). **La vision omnidirectionnelle**. Dans Guidel, éditeur, *Journées nationales de la recherche en robotique*.
- [Mouaddib et al., 2005] Mouaddib, E. M., Sagawa, R., Echigo, T., et Yagi, Y. (2005). **Stereo vision with a single camera and multiple mirrors**. Dans *International Conference on Robotics and Automation*, pages 800–805.
- [Nieradka et al., 2009] Nieradka, G., et Butkiewicz, B. (2009). **Features stereo matching based on fuzzy logic**. Dans *International Fuzzy Systems Association - European Society for Fuzzy Logic and Technology (IFSA-EUSFLAT)*.
- [Nishimoto et al., 2007] Nishimoto, T., et Yamaguchi, J. (2007). **Three dimensional measurement using fisheye stereo vision**. Dans *SICE Annual Conference*, pages 2008–2012.
- [Nistér et al., 2008] Nistér, D., et Stewénius, H. (2008). **Linear time maximally stable extremal regions**. Dans *Proceedings of the 10th European Conference on Computer Vision : Part II, ECCV '08*, pages 183–196, Berlin, Heidelberg. Springer-Verlag.
- [Nistér, 2004] Nistér, D. (2004). **An efficient solution to the five-point relative pose problem**.
- [Obst et al., 2012] Obst, M., Bauer, S., et Wanielik, G. (2012). **Urban multipath detection and mitigation with dynamic 3d maps for reliable land vehicle localization**. Dans *Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION*, pages 685–691.
- [Ohta et al., 1985] Ohta, Y., et Kanade, T. (1985). **Stereo by intra- and inter-scanline search using dynamic programming**. Dans *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 7, pages 139–154.
- [Okutomi et al., 1993] Okutomi, M., et Kanade, T. (1993). **A multiple baseline stereo**. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 15(4) :353–363.
- [Oliensis, 2002] Oliensis, J. (2002). **Exact two-image structure from motion**. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24 :2002.
- [Ortín et al., 2001] Ortín, D., et Montiel, J. M. M. (2001). **Indoor robot motion based on monocular images**. *Robotica*, 19 :331–342.

- [Pajares et al., 2003] Pajares, G., et De La Cruz, J. M. (2003). **Stereovision matching through support vector machines**. *Pattern Recognition Letters*, 24(15) :2575–2583.
- [Paulinas et al., 2007] Paulinas, M., et Ušinskas, A. (2007). **A survey of genetic algorithms applications for image enhancement and segmentation**. *Information Technology and Control*, 36 :278–284.
- [Peyret et al., 2014] Peyret, F., Bétaille, D., Carolina, P., Toledo-Moreo, R., Gomez-Skarmeta, A., et Ortiz, M. (2014). **Gnss autonomous localization : Nlos satellite detection based on 3-d maps**. *Robotics Automation Magazine, IEEE*, 21(1) :57–63.
- [Peyret et al., 2011] Peyret, F., Bétaille, D., et Mougel, F. (2011). **Non-line-of-sight gnss signal detection using an on-board 3d model of buildings**. Dans *ITS Telecommunications (ITST), 2011 11th International Conference on*, pages 280–286.
- [Pock et al., 2010] Pock, T., Cremers, D., Bischof, H., et Chambolle, A. (2010). **Global solutions of variational models with convex regularization**. *SIM Journal on Imaging Sciences*, 3 :1122–1145.
- [Pradeep et al., 2013] Pradeep, V., Rhemann, C., Izadi, S., Zach, C., Bleyer, M., et Bathiche, S. (2013). **Monofusion : Real-time 3d reconstruction of small scenes with a single web camera**. Dans *ISMAR*.
- [Puig et al., 2011] Puig, L., Bastanlar, Y., Sturm, P., Guerrero, J., et Barreto, J. P. (2011). **Calibration of central catadioptric cameras using a dlt-like approach**. *International Journal of Computer Vision*, 93(1) :101–114.
- [Ragot, 2009] Ragot, N. (2009). **Conception d'un capteur de stéréovision omnidirectionnelle : architecture, étalonnage et applications à la reconstruction de scènes 3D**. PhD thesis, École Doctorale Sciences Physiques, Mathématiques et de l'Information pour l'Ingénieur, Université de Rouen, Institut de Recherche en Systèmes Electroniques EMbarqués.
- [Ramalingam, 2006] Ramalingam, S. (2006). **Generic Imaging Models : Calibration and 3D Reconstruction Algorithms**. PhD thesis, Institut National Polytechnique de Grenoble, Laboratoire GRAVIR-IMAG, Department of Computer Science, University of California, Santa Cruz.
- [Ramalingam et al., 2010] Ramalingam, S., Bouaziz, S., Sturm, P., et Brand, M. E. (2010). **Skyline2gps : Localization in urban canyons using omni-skylines**. Dans *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3816–3823.
- [Rao et al., 2012] Rao, M., et Falco, G. (2012). **How can pseudorange measurements be generated from code tracking ?** *Inside GNSS*, 7(1) :26–33.
- [Rosten et al., 2006] Rosten, E., et Drummond, T. (2006). **Machine learning for high-speed corner detection**. Dans *Proceedings of the 9th European Conference on Computer Vision - Volume Part I, ECCV'06*, pages 430–443, Berlin, Heidelberg. Springer-Verlag.

- [Rother et al., 2004] Rother, C., Kolmogorov, V., et Blake, A. (2004). **"grabcut" : Interactive foreground extraction using iterated graph cuts**. Dans *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, pages 309–314, New York, NY, USA. ACM.
- [Ruble et al., 2011] Rublee, E., Rabaud, V., Konolige, K., et Bradski, G. (2011). **Orb : An efficient alternative to sift or surf**. Dans *Proceedings of the 2011 International Conference on Computer Vision, ICCV '11*, pages 2564–2571, Washington, DC, USA. IEEE Computer Society.
- [Ruichek, 2005] Ruichek, Y. (2005). **Perception de l'environnement par stéréovision, application à la sécurité dans les systèmes de transports terrestres**. Habilitation à Diriger des Recherches, Université des Sciences et Technologies de Lille.
- [Rusu et al., 2011] Rusu, R. B., et Cousins, S. (2011). **3d is here : Point cloud library (pcl)**. Dans *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China.
- [Sahmoudi et al., 2014] Sahmoudi, M., Bourdeau, A., et Tourneret, J.-Y. (2014). **Deep fusion of vector tracking gnss receivers and a 3d city model for robust positioning in urban canyons with nlos signals**. Dans *Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing (NAVITEC), 2014 7th ESA Workshop on*, pages 1–7.
- [Salmen et al., 2009] Salmen, J., Schlipfing, M., Edelbrunner, J., Hegemann, S., et Lüke, S. (2009). **Real-time stereo vision : Making more out of dynamic programming**. *Computer Analysis of Images and Patterns*, 5702 :1096–1103.
- [Scaramuzza, 2008] Scaramuzza, D. (2008). **Omnidirectional Vision : from Calibration to Robot Motion Estimation**. PhD thesis, ETH Zurich.
- [Scaramuzza, 2011] Scaramuzza, D. (2011). **1-point-ransac structure from motion for vehicle-mounted cameras by exploiting non-holonomic constraints**. *International Journal of Computer Vision*, 95(1) :74–85.
- [Scaramuzza et al., 2011] Scaramuzza, D., et Fraundorfer, F. (2011). **Visual odometry [tutorial]**. *Robotics Automation Magazine, IEEE*, 18(4) :80–92.
- [Scaramuzza et al., 2006] Scaramuzza, D., Martinelli, A., et Siegwart, R. (2006). **A toolbox for easily calibrating omnidirectional cameras**. Dans *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 5695–5701.
- [Scharstein et al., 2002] Scharstein, D., et Szeliski, R. (2002). **A taxonomy and evaluation of dense two-frame stereo correspondence algorithms**. *International Journal of Computer Vision*, 47 :7–42.
- [Schmid et al., 1995] Schmid, C., et Mohr, R. (1995). **Matching by local invariants**. Rapport technique, Institut National de Recherche en Informatique et en Automatique.
- [Shah et al., 1996] Shah, S., et Aggarwal, J. K. (1996). **Intrinsic parameter calibration procedure for (high-distortion) fish-eye lens camera with distortion model and accuracy estimation**. *Pattern Recognition*, 29 :1775–1788.

- [Shah et al., 1997] Shah, S., et Aggarwal, J. K. (1997). **Mobile robot navigation and scene modeling using stereo fish-eye lens system**. *Machine Vision and Applications*, 10 :159–173.
- [Shi et al., 1994] Shi, J., et Tomasi, C. (1994). **Good features to track**. Dans *Computer Vision and Pattern Recognition CVPR '94*.
- [Smadja et al., 2005] Smadja, L., Benosman, R., et Devars, J. (2005). **Matrice semi fondamentale pour système de stéréoscopie hybride**. *Traitement du Signal*, 22 :513–526.
- [Srestasathiern et al., 2014] Srestasathiern, P., et Soontranon, N. (2014). **A novel camera calibration method for fish-eye lenses using line features**. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, pages 327–332.
- [Stock et al., 2003] Stock, C., et Pinz, A. (2003). **Similarity measure for corner re-detection**. Dans *proceedings of Scandinavian Conference on Image Analysis, SCIA*, pages 133–139.
- [Taguchi et al., 2008] Taguchi, Y., Wilburn, B., et Zitnick, C. L. (2008). **Stereo reconstruction with mixed pixels using adaptive over-segmentation**. Dans *CVPR*, pages 1–8.
- [Tardif et al., 2005] Tardif, J.-P., et Sturm, P. (2005). **Calibration of cameras with radially symmetric distortion**. Dans *6th Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras*, pages 44–51, Beijing, China.
- [Tardif et al., 2009] Tardif, J.-P., Sturm, P., Trudeau, M., et Roy, S. (2009). **Calibration of cameras with radially symmetric distortion**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(9) :1552–1566.
- [Tarsha Kurdi, 2008] Tarsha Kurdi, F. (2008). **Extraction et reconstruction de bâtiments en 3D à partir de relevés lidar aéroportés**. PhD thesis, Université Louis Pasteur Strasbourg I. Thèse de doctorat dirigée par Grussenmeyer, Pierre Sciences pour l'Ingénieur. Topographie. Géomatique Strasbourg 1 2008.
- [Tay et al., 2013] Tay, S., et Marais, J. (2013). **Weighting models for gps pseudorange observations for land transportation in urban canyons**. Dans *6th European Workshop on GNSS Signals and Signal Processing*, page 4p, Germany.
- [Terabayashi et al., 2009] Terabayashi, K., Mitsumoto, H., Morita, T., Aragaki, Y., Shimomura, N., et Umeda, K. (2009). **Measurement of three-dimensional environment with a fish-eye camera based on structure from motion - error analysis**. *Journal of Robotics and Mechatronics*, 21 :680–688.
- [Torii et al., 2008] Torii, A., Imiya, A., et Ohnishi, N. (2008). **Two- and three- view geometry for spherical cameras**. Rapport technique, School of Science and Technology, Institute of Media and Information Technology, Chiba University, Japan.
- [Torr et al., 2004] Torr, P. H., et Criminisi, A. (2004). **Dense stereo using pivoted dynamic programming**. *Image and Vision Computing*, 22 :795–806.

- [Troiani et al., 2014] Troiani, C., Martinelli, A., Laugier, C., et Scaramuzza, D. (2014). **2-point-based outlier rejection for camera-imu systems with applications to micro aerial vehicles**. Dans *2014 IEEE International Conference on Robotics and Automation, ICRA 2014, Hong Kong, China, May 31 - June 7, 2014*, pages 5530–5536.
- [Viandier, 2011] Viandier, N. (2011). **Modélisation et utilisation des erreurs de pseudo-distances GNSS en environnement transport pour l'amélioration des performances de localisation**. These, Ecole Centrale de Lille.
- [Wang et al., 2013] Wang, L., Groves, P. D., et Ziebart, M. K. (2013). **Gnss shadow matching : Improving urban positioning accuracy using a 3d city model with optimized visibility scoring scheme**. *Navigation*, 60(3) :195–207.
- [Wei et al., 2004] Wei, Y., et Quan, L. (2004). **Region-based progressive stereo matching**. Dans *Conference on Computer Vision and Pattern Recognition*.
- [Weissig et al., 2012] Weissig, C., Schreer, O., Eisert, P., et Kauff, P. (2012). **The ultimate immersive experience : Panoramic 3d video acquisition**. *Lecture Notes in Computer Science*, 7131 :671–681.
- [Wu et al., 2010] Wu, T.-P., Yeung, S.-K., Jia, J., et Tang, C.-K. (2010). **Quasi-dense 3d reconstruction using tensor-based multiview stereo**. Dans *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [Yamaguchi, 2011] Yamaguchi, J. (2011). **Advances in Theory and Applications of Stereo Vision**, chapitre 8 - Three Dimensional Measurement Using Fisheye Stereo Vision, pages 151–164. InTech.
- [Yang et al., 2010] Yang, Q., Wang, L., et Ahuja, N. (2010). **A constant-space belief propagation algorithm for stereo matching**. Dans *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [Ying et al., 2004] Ying, X., et Hu, Z. (2004). **Can we consider central catadioptric cameras and fisheye cameras within a unified imaging model**. *Lecture Notes Computer Science*, 3021 :442–455.
- [Yoon et al., 2006] Yoon, K.-J., et Kweon, I. S. (2006). **Adaptive support-weight approach for correspondence search**. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28 :650–656.
- [Zhang et al., 2011] Zhang, T., et Boulton, T. (2011). **Realistic stereo error models and finite optimal stereo baselines**. Dans *Applications of Computer Vision (WACV), 2011 IEEE Workshop on*, pages 426–433.
- [Zhang et al., 2015] Zhang, T., et Nakamura, Y. (2015). **Surface reconstruction with sparse point clouds of velodyne® sensor**. Dans *The 14th IFToMM World Congress*.
- [Zhang et al., 1984] Zhang, T. Y., et Suen, C. Y. (1984). **A fast parallel algorithm for thinning digital patterns**. *Commun. ACM*, 27(3) :236–239.

- [Zhao et al., 2000] Zhao, H., et Aggarwal, J. K. (2000). **3d reconstruction of an urban scene from synthetic fish-eye images**. Dans *Proceedings of the 4th IEEE Southwest Symposium on Image Analysis and Interpretation, SSIAI '00*, pages 219–, Washington, DC, USA. IEEE Computer Society.
- [Zhu et al., 2012] Zhu, M., Ramalingam, S., Taguchi, Y., et Garaas, T. (2012). **Monocular visual odometry and dense 3d reconstruction for on-road vehicles**. Dans Fusiello, A., Murino, V., et Cucchiara, R., éditeurs, *Computer Vision – ECCV 2012. Workshops and Demonstrations*, volume 7584 de *Lecture Notes in Computer Science*, pages 596–606. Springer Berlin Heidelberg.
- [Zitnick et al., 1999] Zitnick, C. L., et Kanade, T. (1999). **A cooperative algorithm for stereo matching and occlusion detection**. Rapport technique, The Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.

LISTE DES FIGURES

1.1	Exemple de lidar 2D Sick.	5
1.2	Lidar rotatif Velodyne et exemple de nuage de points acquis à Belfort.	5
1.3	Projection perspective d'un point dans deux caméras. L'estimation 3D se fait par l'opération inverse à partir des points des deux images. Issu de [Horaud et al., 1995].	6
1.4	Configuration du stéréoscope fisheye. Les deux caméras sont placées sur le toit du véhicule le long de l'axe longitudinal.	10
1.5	Camion en configuration binoculaire en niveaux de gris (Cfg 1). La caméra du milieu est une caméra couleur pour la de segmentation du ciel de [Attia, 2013].	11
1.6	SetCar en configuration monoculaire couleur avec un appareil photo (Cfg 4). .	12
1.7	Repères caméra à la position du véhicule dans la situation rue du Général Roussel. L'axe Z représente le haut et est en direction du ciel.	13
1.8	Résultat de la reconstruction 3D du point de vue de la situation rue du Général Roussel. La caméra d'acquisition fisheye est placée à environ 2m de hauteur. .	14
2.1	Projection centrale et non-centrale cas dioptrique et catadioptrique.	17
2.2	Capteur catadioptrique avec un miroir hyperbolique utilisé par [Lhuillier, 2011].	18
2.3	Caméra munie d'un objectif fisheye, de la configuration (Cfg 1) avec l'objectif photo Sigma (section 1.3.2).	19
2.4	Modèle de projection centrale décrivant la relation entre l'objectif fisheye et le plan image. Le rayon r sur le plan image est fonction de l'angle d'incidence θ du rayon perçu.	19
2.5	Les illustrations des projections fisheye sont issues du site www.fisheylens.de	20
2.6	Géométrie épipolaire pour des images sphériques [Arican et al., 2010].	25
2.7	Cube utilisé pour le calibrage des caméras dans le projet BOSS [Bos, 2009]. .	27
2.8	Méthode coopérative locale. Illustration pour une tranche 2D de l'espace 3D des disparités [Zitnick et al., 1999].	35
2.9	La coupure de graphe consiste à trouver la coupure de capacité minimale [Ishikawa et al., 1998].	36
2.10	Les réseaux de neurones, ce graphe représente les neurones pour l'appariement de primitives correspondant à des points de contours [Ruichek, 2005].	37

3.1	Capteur stéréoscopique fisheye calibré. Le point P de la scène est projeté sur les sphères gauche et droite respectivement en P_l et P_r , et est vu comme p_l et p_r dans les images captées.	40
3.2	Modélisation de la projection omnidirectionnelle fisheye avec centre de projection unique et symétrie autour de l'axe optique.	42
3.3	Profils de modélisation des projections centrales présentées pour un champ de vision de 185° . Chaque rayon incident subit une projection orthogonale sur le capteur à sa rencontre avec le profil de modèle fisheye considéré.	43
3.4	Segmentation du disque image par <i>GrabCut</i>	45
3.5	Résultat de la détection du disque de l'image. (a) et (b) : pour l'image avant de l'acquisition Cap2/3039. (c) : exemple de cas où elle échoue.	46
3.6	Pseudo-algorithme du RANSAC pour l'application de l'algorithme des 9 points.	53
3.7	Illustration du calibrage avec la paire Cap2/3039 des acquisitions de Lille.	54
3.8	Un cercle épipolaire est la projection d'une ligne épipolaire sur la sphère unitaire. Il appartient au plan défini par la ligne épipolaire et le centre de la sphère.	58
3.9	Distribution des estimations de l'épipôle caméra arrière ayant $X > 0$ pour la paire simulée 40 en projection équisolide avec utilisation du LO-RANSAC.	60
3.10	Les courbes épipolaires à l'intérieur du même plan épipolaire (plans Ψ_i) peuvent être décrites par le même rayon ou le même angle η le long de l'arc de scan, défini comme l'équateur entre les deux épipôles.	60
3.11	Distinction entre les repères caméras et épipolaires. Les repères épipolaires sont obtenus par des rotations des repères caméras, qui sont définies par les axes des repères épipolaires.	61
3.12	Rotation du repère caméra vers le repère épipolaire. En haut : vue de coupe de la sphère le long de l'axe épipolaire. En bas : vue du plan image. \mathbf{X}_{epip} représente l'axe épipolaire. Dans le cas d'un objectif dont l'angle de vision est supérieur à 180° , le diamètre du disque image est plus grand que la distance entre les deux épipôles. L'axe épipolaire passe toujours par le centre de la sphère unitaire.	62
3.13	Images simulées présentées.	64
3.14	Images réelles présentées.	65
3.15	Images réelles de la même scène avec deux configurations à projection différente.	66
3.16	Diagrammes de distribution des erreurs d'estimation du champ de vision et de rotation paire simulée 24 en projection équisolide.	68
3.17	Diagrammes de distribution des erreurs d'estimation du champ de vision et de rotation paire simulée 24 en projection équidistante.	69
4.1	Transformation épipolaire « latitude-longitude » proposée par [Li, 2006b, Li, 2008], dans le but de transformer les arcs épipolaires en lignes épipolaires horizontales.	74

4.2	Paire d'images fisheye générées avec POV-Ray sur lesquelles sont affichées deux courbes épipolaires conjuguées.	77
4.3	Calcul du plus court chemin sur un graphe de mise en correspondance, avec étape de lecture selon le sens normal (indice min à max) et en priorité selon (a) la diagonale ou (b) la verticale. À gauche : graphe (limites en rouge) et chemin minimum obtenu (en vert). À droite : carte des disparités sphériques obtenues.	79
4.4	Conséquences de la limitation du nombre de correspondants par point. De gauche à droite : les cartes des disparités sphériques, l'effet sur un plus court chemin et le nuage de points 3D correspondant.	80
4.5	Distances vérifiées des images synthétisées présentées, et en (c) l'image évaluée additionnelle, représentant les conditions d'un croisement en T.	81
4.6	Diagramme à bulles d'évaluation des paramètres de l'algorithme de programmation dynamique utilisé en niveaux de gris. Le diamètre évolue dans le même sens que le critère <i>ratio</i> , et la couleur représente $\overline{ Er }U$ (mètres).	82
4.7	Cartes des distances obtenues lors de la mise en correspondance pour les jeux de paramètres retenus avec les paires d'images simulées 40, 24 et 3. Chaque carte des distances est associée à une carte des erreurs graduée sur l'intervalle $[0;2m]$ du vert au rouge. La couleur jaune est utilisée pour une erreur de 1m. Lorsque l'erreur est supérieur à 10m les pixels sont colorés en blanc. Les pixels non-appariés sont représentés en bleu.	84
4.8	Graphe 3D de programmation dynamique pour prendre en compte plusieurs lignes épipolaires voisines.	85
4.9	Diagramme à bulles d'évaluation des paramètres de l'algorithme de programmation dynamique utilisé en niveaux de gris avec l'utilisation du graphe 3D. Le diamètre évolue dans le même sens que le critère <i>ratio</i> , et la couleur représente $\overline{ Er }U$ (mètres).	86
4.10	Résultat de mise en correspondance avec graphe 3D (3 et 5 lignes) sur les paires d'images simulées 40, 24 et 3 et les paramètres exacts de calibrage.	88
4.11	Images de conditions réelles rencontrées à Lille.	89
4.12	Cartes des distances estimées sur des images réelles de Lille par graphe 3D 3 lignes avec <i>offset</i> = 15, <i>const</i> = 2 et rejet des zones épipolaires 25% (paramètres (c+)).	90
4.13	Diagramme à bulles d'évaluation des paramètres de l'algorithme de programmation dynamique utilisé avec corrélation couleur et l'utilisation du graphe 3D. Le diamètre évolue dans le même sens que le critère <i>ratio</i> , et la couleur représente $\overline{ Er }U$ (mètres).	93
4.14	Images réelles issues de la séquence de Belfort du 22 août 2014.	95
4.15	Carte des distances estimées par la méthode du graphe 3D sur les paires d'images réelles de Belfort.	96
4.16	Carte des distances estimées par la méthode du graphe 3D sur les paires d'images réelles de Belfort avec utilisation du masque du ciel.	97

4.17	Sous-échantillonnage d'un nuage de point obtenu lors de nos expérimentations (rue du Général Roussel, Belfort, France).	98
4.18	Point 3D et ses voisins régulièrement placés dans une grille de voxels, avec d_{\max} la distance maximale des points voisins dans le cube englobant.	99
4.19	Réduction du bruit par recherche des voisins dans un rayon donné pour un nuage de point obtenu lors de nos expérimentations (rue du Général Roussel, Belfort, France).	100
4.20	Transformations pour changer de repère et pouvoir concaténer les nuages de points. $\mathfrak{R}_{cam,t} = (O_t, \mathbf{X}_{cam,t}, \mathbf{Y}_{cam,t}, \mathbf{Z}_{cam,t})$ est le repère caméra de la vue de l'instant t . $\mathfrak{R}_{epip,t,O(t')} = (O_{t'}, \mathbf{X}_{epip,t}, \mathbf{Y}_{epip,t}, \mathbf{Z}_{epip,t})$ est le repère épipolaire entre les vues des instants $t - 1$ et t , centré sur l'origine de l'instant t'	102
4.21	Nuage de points après assemblage de 10 et 20 nuages de points 3D successifs (rue du Général Roussel, Belfort, France).	102
4.22	Reconstruction de la rue du Général Roussel avec affichage des caméras aux positions des acquisitions en jaune.	103
4.23	Mesure de l'écart de module et d'argument entre les vecteurs liant les points successifs des trajectoires estimée et vérité.	105
4.24	Modèle 3D global pour la scène I de Belfort.	106
4.25	Évaluation de l'odométrie visuelle sur la scène I de Belfort par rapport aux données vérité RTK.	107
4.26	Modèle 3D global pour la situation O de Belfort.	108
4.27	Évaluation de l'odométrie visuelle sur la scène O de Belfort par rapport aux données vérité GPS RTK.	109
4.28	Mesure de la largeur de la rue dans un nuage de points local.	110
5.1	Constellation des satellites de localisation GPS avec les plans orbitaux. Source : [Nav, 1996].	112
5.2	Mesure de l'azimut et élévation d'un satellite à un point de réception, en fonction de l'horizon au point de réception.	114
5.3	<i>Dilution Of Precision</i> de la localisation, illustrée pour deux satellites.	115
5.4	Effet multi-trajets suite aux réflexions des rayons sur les structures urbaines.	116
5.5	Trajet du rayon GNSS avec une réflexion avant d'atteindre le récepteur.	122
5.6	Trajet des rayons réfléchis de 1 à 3 réflexions, selon la coupe dans le plan formé par le rayon GNSS et le point de réception.	124
5.7	Détection des plans au croisement de la rue du Général Roussel, Belfort, avec (b) la méthode basée sur RANSAC et (c) la méthode de la transformée de Hough 2D.	127
5.8	Trace au sol à la position juste avant la cathédrale pour la situation rue du Général Roussel, Belfort. L'axe \mathbf{Y} du monde est orienté vers le bas et l'axe \mathbf{X} vers la droite.	128

5.9 Pré-traitements appliqués sur la trace au sol générée à partir du modèle 3D reconstruit et détection de segments par transformée de Hough probabiliste, pour la situation rue du Général Roussel, Belfort. 130

5.10 Projection des satellites dans la trace au sol autour du récepteur GNSS. Le Nord géographique est indiqué en bleu. Situation rue du Général Roussel, Belfort. 134

5.11 Comparaison entre la projection dans l'image des satellites reçus par le récepteur Safedrive et de la constellation de satellites disponibles au même moment au même endroit (le 25 août 2014, temps ITOW : 138970s, rue du Général Roussel à Belfort). 135

5.12 Estimation de la position pour une séquence dynamique, rue Roussel à Belfort, 25 août 2014. 143

LISTE DES TABLEAUX

2.1	Tableau récapitulatif des différents modèles de projection de capteurs fisheye.	21
3.1	Liste des détecteurs étudiés. Les détecteurs marqués en gras sont ceux qui sont donnés assez robustes aux changements d'échelle et d'orientation, et qui ont donc été retenus pour les évaluations sur les images fisheye.	48
3.2	Liste des descripteurs étudiés. Les descripteurs marqués en gras sont ceux qui sont donnés assez robustes aux changements d'échelle et d'orientation, et qui ont donc été retenus pour les évaluations sur les images fisheye.	49
3.3	Fonctions de modélisation des projections fisheye centrales à un paramètre étudiées et définition du paramètre a correspondant.	50
3.4	Résultat des tests du seuil d'erreur angulaire paire simulée 40 en projection équisolide.	67
3.5	Résultat des tests du seuil d'erreur angulaire paire simulée 40 en projection équidistante.	67
3.6	Résultat des tests du seuil d'erreur angulaire paire simulée 24 en projection équisolide.	67
3.7	Résultat des tests du seuil d'erreur angulaire paire simulée 24 en projection équidistante.	68
3.8	Résultat des tests du seuil d'erreur angulaire paire réelle Cap2/3039 (projection équisolide).	69
3.9	Résultat des tests du seuil d'erreur angulaire paire réelle Cap3/2865 (projection équisolide).	69
3.10	Résultat des tests du choix de la fonction de projection en fonction de l'objectif utilisé pour des paires réelles acquises au même endroit (Techn'hom de Belfort).	70
3.11	Résultat des tests de variantes RANSAC pour les paires simulées.	71
3.12	Résultat des tests de variantes RANSAC pour les paires réelles.	71
4.1	Pourcentages de pixels utiles et à l'infini dans les images simulées étudiées.	82
4.2	Les trois jeux de paramètres (<i>offset, const</i>) retenus pour la présentation des résultats sur les scènes simulées 40, 24 et 3.	83
4.3	Évaluation pour les jeux de paramètres retenus avec les paires d'images simulées 40, 24 et 3 et les paramètres de calibrage exacts.	83
4.4	Évaluation pour les jeux de paramètres retenus avec les paires d'images simulées 40 et 24 et les paramètres de calibrage estimés.	83

4.5	Évaluation de la mise en correspondance avec graphe 3D (3 et 5 lignes) sur les paires d'images simulées 40, 24 et 3 et les paramètres exacts de calibrage. . . .	87
4.6	Évaluation de la mise en correspondance avec graphe 3D (3 et 5 lignes) sur les paires d'images simulées 40 et 24 avec les paramètres calibrage estimés.	87
4.7	Évaluation de la programmation dynamique en ignorant les zones proches des épipôles pour les meilleures combinaisons de paramètres avec les paires d'images simulées 40, 24 et 3.	90
4.8	Évaluation de la programmation dynamique en ignorant les zones proches des épipôles pour les meilleures combinaisons de paramètres avec les paires d'images simulées 40, 24 et 3, en employant le calibrage réel.	90
4.9	Évaluation de la largeur des rues pour les scènes réelles mono de Belfort en ligne droite avec mise en correspondance couleur, masque du ciel et filtrage des données 3D.	101
5.1	Tableau récapitulatif des mesures du retard de pseudodistance d'un rayon NLOS dû aux réflexions.	124
5.2	Évaluation de la largeur des rues pour les scènes réelles stéréo de Lille en ligne droite à partir des plans détectés.	132
5.3	Évaluation de la largeur des rues pour les scènes réelles mono de Belfort en ligne droite avec mise en correspondance en couleurs et masque du ciel à partir des plans détectés.	133
5.4	Évaluation de la localisation 2D avec correction du retard des pseudodistances des satellites NLOS pour l'acquisition Cap3/0900 de Lille en configuration binoculaire (Cfg 1).	138
5.5	Évaluation de la localisation 2D avec correction du retard des pseudodistances des satellites NLOS pour l'acquisition Cap4/1560 de Lille en configuration binoculaire (Cfg 1).	139
5.6	Évaluation de la localisation 2D avec correction du retard des pseudodistances des satellites NLOS pour l'acquisition 6506 de Belfort en configuration monoculaire (Cfg 4).	140

><(((⁰> See you <^o)))><

Construction de modèles 3D à partir de données vidéo fisheye — Application à la localisation en milieu urbain

Résumé :

Cette recherche vise à la modélisation 3D depuis un système de vision fisheye embarqué, utilisée pour une application GNSS dans le cadre du projet Predit CAPLOC. La propagation des signaux satellitaires en milieu urbain est soumise à des réflexions sur les structures, altérant la précision et la disponibilité de la localisation. L'ambition du projet est (1) de définir un système de vision omnidirectionnelle capable de fournir des informations sur la structure 3D urbaine et (2) de montrer qu'elles permettent d'améliorer la localisation.

Le mémoire expose les choix en (1) calibrage automatique, (2) mise en correspondance entre images, (3) reconstruction 3D ; chaque algorithme est évalué sur images de synthèse et réelles. De plus, il décrit une manière de corriger les réflexions des signaux GNSS depuis un nuage de points 3D pour améliorer le positionnement. En adaptant le meilleur de l'état de l'art du domaine, deux systèmes sont proposés et expérimentés. Le premier est un système stéréoscopique à deux caméras fisheye orientées vers le ciel. Le second en est l'adaptation à une unique caméra.

Le calibrage est assuré à travers deux étapes : l'algorithme des 9 points adapté au modèle « équisolide » couplé à un RANSAC, suivi d'un affinement par optimisation Levenberg-Marquardt. L'effort a été porté sur la manière d'appliquer la méthode pour des performances optimales et reproductibles. C'est un point crucial pour un système à une seule caméra car la pose doit être estimée à chaque nouvelle image.

Les correspondances stéréo sont obtenues pour tout pixel par programmation dynamique utilisant un graphe 3D. Elles sont assurées le long des courbes épipolaires conjuguées projetées de manière adaptée sur chaque image. Une particularité est que les distorsions ne sont pas rectifiées afin de ne pas altérer le contenu visuel ni diminuer la précision. Dans le cas binoculaire il est possible d'estimer les coordonnées à l'échelle. En monoculaire, l'ajout d'un odomètre permet d'y arriver. Les nuages successifs peuvent être calés pour former un nuage global en SfM.

L'application finale consiste dans l'utilisation du nuage 3D pour améliorer la localisation GNSS. Il est possible d'estimer l'erreur de pseudodistance d'un signal après multiples réflexions et d'en tenir compte pour une position plus précise. Les surfaces réfléchissantes sont modélisées grâce à une extraction de plans et de l'empreinte des bâtiments. La méthode est évaluée sur des paires d'images fixes géoréférencées par un récepteur bas-coût et un récepteur GPS RTK (vérité terrain). Les résultats montrent une amélioration de la localisation en milieu urbain.

Mots-clés : Stéréovision fisheye, Calibrage automatique, Nuage de points 3D, SfM, Extraction de plans, Localisation GNSS

Construction of 3D models from fisheye video data—Application to the localisation in urban area

Abstract:

This research deals with 3D modelling from an embedded fisheye vision system, used for a GNSS application as part of CAPLOC project. Satellite signal propagation in urban area implies reflections on structures, impairing localisation's accuracy and availability. The project purpose is (1) to define an omnidirectional vision system able to provide information on urban 3D structure and (2) to demonstrate that it allows to improve localisation.

This thesis addresses problems of (1) self-calibration, (2) matching between images, (3) 3D reconstruction; each algorithm is assessed on computer-generated and real images. Moreover, it describes a way to correct GNSS signals reflections from a 3D point cloud to improve positioning. We propose and evaluate two systems based on state-of-the-art methods. First one is a stereoscopic system made of two sky facing fisheye cameras. Second one is the adaptation of the former to a single camera. Calibration is handled by a two-steps process: the 9-point algorithm fitted to "equisolid" model coupled with a RANSAC, followed by a Levenberg-Marquardt optimisation refinement. We focused on the way to apply the method for optimal and repeatable performances. It is a crucial point for a system composed of only one camera because the pose must be estimated for every new image.

Stereo matches are obtained for every pixel by dynamic programming using a 3D graph. Matching is done along conjugated epipolar curves projected in a suitable manner on each image. A distinctive feature is that distortions are not rectified in order to neither degrade visual content nor to decrease accuracy. In the binocular case it is possible to estimate full-scale coordinates. In the monocular case, we do it by adding odometer information. Local clouds can be wedged in SfM to form a global cloud. The end application is the usage of the 3D cloud to improve GNSS localisation. It is possible to estimate and consider a signal pseudodistance error after multiple reflections in order to increase positioning accuracy. Reflecting surfaces are modelled thanks to plane and buildings trace fitting. The method is evaluated on fixed image pairs, georeferenced by a low-cost receiver and a GPS RTK receiver (ground truth). Study results show the localisation improvement ability in urban environment.

Keywords: Fisheye stereovision, Self-calibration, 3D point cloud, Structure from Motion, Plane fitting, GNSS localisation

The logo for SPIM (École doctorale SPIM) features the letters 'S', 'P', 'I', and 'M' in a stylized, white, sans-serif font. The 'S' is the largest and most prominent, with the other letters stacked to its right.

■ École doctorale SPIM - Université de Technologie Belfort-Montbéliard

F - 90010 Belfort Cedex ■ tél. +33 [0]3 84 58 31 39

■ ed-spim@univ-fcomte.fr ■ www.ed-spim.univ-fcomte.fr

