Capteur de stéréovision hybride pour la navigation des drones

Eynard Damien

▶ **To cite this version:**

Eynard Damien. Capteur de stéréovision hybride pour la navigation des drones. Robotique [cs.RO]. Université de Picardie Jules Verne, 2011. Français. NNT : . tel-00652615

HAL Id: tel-00652615
https://theses.hal.science/tel-00652615

Submitted on 15 Dec 2011

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

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

UNIVERSITÉ DE PICARDIE JULES VERNE

ÉCOLE DOCTORALE
SCIENCES ET SANTÉ

T H È S E

pour obtenir le titre de

Docteur en Sciences

de l'Université de Picardie Jules Verne
Mention : AUTOMATIQUE

Présentée et soutenue par Damien EYNARD

Capteur de stéréovision hybride pour la navigation des drones

Thèse dirigée par Pascal VASSEUR, Cédric DEMONCEAUX, Vincent FRÉMONT

préparée à l'UPJV, au sein du laboratoire MIS

Jury proposé :

Rapporteurs :	Simon LACROIX	-	LAAS - CNRS
	Tarek HAMEL	-	i3S - CNRS
Directeurs :	Pascal VASSEUR	-	Litis - Université de Rouen
	Cédric DEMONCEAUX	-	Le2i - CNRS
	Vincent FRÉMONT	-	Heudiasyc - CNRS
Examinateurs :	Claude PEGARD	-	MIS - Université UPJV
	Patrick FABIANI	-	ONERA

"L’expérience, c’est le nom que chacun donne à ses erreurs."
Oscar Wilde

Résumé : La connaissance de l'attitude, de l'altitude, de la segmentation du sol et du mouvement est essentielle pour la navigation d'un drone, en particulier durant les phases critiques de décollage et d'atterrissage. Dans ce travail de thèse, nous présentons un système stéréoscopique hybride composé d'une caméra fisheye et d'une caméra perspective pour estimer les paramètres de navigation d'un drone. À partir de ce capteur, une approche systémique est proposée. Contrairement aux méthodes classiques de stéréovision basées sur l'appariement de primitives, nous proposons des méthodes qui évitent toute mise en correspondance entre les vues hybrides. Une technique de *plane-sweeping* est suggérée pour déterminer l'altitude et détecter le plan du sol. La rotation et la translation du mouvement sont ensuite découplés : la vue fisheye contribue à évaluer l'attitude et l'orientation tandis que la vue perspective contribue à apporter l'échelle métrique de la translation. Le mouvement peut ainsi être estimé de façon robuste et à l'échelle métrique grâce à la connaissance de l'altitude. Cette méthode repose sur l'algorithme des 2-points complété par un filtre de Kalman.

Nous proposons des approches robustes, temps réel et précises, exclusivement basées sur la vision avec une implémentation C++. Bien que cette approche évite l'utilisation de capteurs autres que les caméras, ce système peut également être appuyé par une centrale inertielle.

Mots clés : Drone, stéréovision, hybride, navigation, altitude, mouvement, attitude, temps réel, étalonnage.

Hybrid Stereovision Sensor for UAV navigation.

Abstract : Knowledge of attitude, altitude, ground plane segmentation and motion is essential for an Unmanned Aerial Vehicle during critical maneuvers such as landing and take-off. In this thesis we present a hybrid stereoscopic rig composed of a fisheye and a perspective camera for vision-based navigation. This sensor is then exploited by systemic methods. In contrast to classical stereoscopic systems based on feature matching, we propose methods which avoid matching between hybrid views. A plane-sweeping approach is proposed for estimating altitude and detecting the ground plane. Rotation and translation are then estimated by decoupling : the fisheye camera contributes to evaluating attitude, while the perspective camera contributes to estimating the scale of the translation. The motion can be estimated robustly at the meter scale, thanks to the knowledge of the altitude. Our method uses a 2-point algorithm complemented by a Kalman filter.

We propose a robust, real-time, accurate, exclusively vision-based approach with a C++ implementation. Although this approach removes the need for any non-visual sensors, it can also be coupled with an Inertial Measurement Unit.

Keywords : UAV, hybrid, stereovision, motion, attitude, altitude, real time, calibration.

Remerciements

Je tiens à remercier l'ensemble des personnes qui m'ont accompagné, soutenu, celles avec qui j'ai travaillé aux laboratoires MIS et Heudiasyc pendant mon doctorat mais également durant les années précédent le doctorat.

Tout d'abord, la rencontre avec la plupart des mes collègues et encadrants a débuté à l'IUT en 2001. Ils m'ont transmis leur passion de la vision, du traitement d'images et de l'enseignement. Au fil des années, j'ai pu enrichir cette passion et les connaissances associées. C'est ce besoin de savoir et d'aller plus loin qui m'a donné l'envie de réaliser un doctorat. Durant cette thèse, ce fut un plaisir de travailler, de collaborer et d'échanger avec chacun d'entre eux. C'est pourquoi je tiens à remercier l'ensemble de mes encadrants Pascal Vasseur, professeur au Litis, Cédric Demonceaux, maître de conférences au Le2i et Vincent Frémont, maître de conférences à l'Heudiasyc, qui ont su me guider et me montrer le chemin de la recherche. Leur encadrement a été exemplaire mais aussi une excellente leçon d'apprentissage pour la suite de ma carrière. Ce suivi s'est réalisé de façon progressive, guidé au début, puis l'autonomie et la confiance se sont installés et ont permis de prendre goût au métier de chercheur.

Je remercie Claude Pégard, professeur et responsable d'équipe PR (Perception en Robotique) au laboratoire MIS, de m'avoir fait l'honneur d'accepter la présidence de ce jury. Claude Pegard m'a également accompagné tout au long de mon cursus, du DUT au doctorat et a toujours été disponible pour ses étudiants. Je tiens à le remercier de m'avoir accordé beaucoup de temps, du soutien et de la confiance pour réaliser l'ensemble des expériences avec les drones qui m'ont été indispensables à l'accomplissement de ma thèse et à la validation des résultats.

Je remercie également le directeur du MIS Gilles Kassel et le directeur adjoint El Mustapha Mouaddib pour nous avoir donné leur soutien, les moyens de réussir et en particulier de réaliser nos expériences aériennes malgré les contraintes environnementales.

J'adresse mes vifs remerciements à Simon Lacroix Directeur de recherches au LAAS et Tarek Hamel Professeur à l'i3S d'avoir accepté d'être rapporteurs de mon travail. Leurs remarques et discussions m'ont été précieuses pour l'amélioration de mes travaux de thèse.

Mes remerciements vont aussi vers Patrick Fabiani, directeur du département commande des systèmes et dynamique de vol à l'ONERA, d'accepter d'être membre du jury, d'avoir partagé son expérience et apporté des remarques constructives.

Je souhaite également remercier Pierre Detaille pour son aide précieuse à l'intégration, l'adaptation des composants pour l'embarqué et d'avoir toujours pu proposer très rapidement une solution aux problématiques qui lui étaient posées.

L'ensemble des travaux de recherche présentés dans cette thèse ont été réalisés

coinjointement au monitorat. Cette expérience d'enseignement était enrichissante et passionnante. Je remercie Alex Potelle d'avoir toujours été présent et d'excellent conseil dans son rôle de responsable pédagogique qu'il a parfaitement accompli.

Je souhaite remercier Valérie Facqui, Marie-lise Saveuse, Annick Crepy pour leur dynamisme, leur soutien et efficacité au sein du laboratoire.

Je tiens à saluer mes collègues doctorants. Guillaume Caron avec qui la collaboration a permis au logiciel Hyscas de voir le jour mais également pour les discussions scientifiques, sa bonne humeur, sa diplomatie et les extra non professionnels. Sang Ly avec qui j'ai commencé la thèse et découvert la Corée du Sud, Pauline Orzekowska (Merveilleux en début de thèse) pour ses échanges scientifiques, Romain Marie l'un de mes premiers étudiant à devenir collègue, Jean-Charles Bazin, Amina Radgui, Ali Ghorayeb, Ibrahim Abdi, Fatima Zahra Benamar, Jean-François Layerle et Ashutosh Natraj.

Je remercie Michel Pichon, de m'avoir donné l'opportunité de réaliser deux années d'alternance dans le domaine de la vision industrielle, de m'avoir accompagné et fait confiance. Michel Pichon m'a apporté à la fois les outils intellectuels et professionnels indispensables à un jeune étudiant pour s'insérer dans la vie active.

Enfin, je tiens à remercier ma famille, ma belle famille et mes amis pour leurs soutiens et encouragements durant ces trois années de thèse. Je remercie en particulier Stéphanie King qui m'a soutenu quotidiennement pour les moments les plus difficiles et sans qui ce travail n'aurais pas été aussi abouti. Ses conseils, son énergie et sa bonne humeur m'ont été précieux tout au long de ces trois années.

Table des matières

Glossaire i

Table des figures iv

Liste des tableaux v

Introduction vii

1 La navigation autonome des drones et la vision 1
- 1.1 L'histoire des drones . . . 1
- 1.2 L'autonomie . . . 5
- 1.3 Les projets . . . 6
- 1.4 Dynamique et paramètres d'état d'un quadri-rotor . . . 8
- 1.5 Les capteurs . . . 10
- 1.6 La vision . . . 13
  - 1.6.1 Les mécanismes de formation d'une image . . . 13
  - 1.6.2 Estimation des paramètres d'état . . . 16
- 1.7 Conclusion . . . 25

2 Architecture du capteur hybride et du matériel 27
- 2.1 Capteurs et modèles . . . 27
  - 2.1.1 Choix des capteurs . . . 28
  - 2.1.2 Choix géométrique . . . 30
- 2.2 Modélisation du capteur . . . 31
  - 2.2.1 Projection perspective . . . 31
  - 2.2.2 Projection sphérique . . . 31
- 2.3 Étalonnage de caméras . . . 32
  - 2.3.1 Étalonnage stéréo hybride . . . 33
  - 2.3.2 Implémentation logicielle . . . 34
- 2.4 Matériel . . . 35
  - 2.4.1 Les capteurs . . . 35
  - 2.4.2 Processeurs de traitement . . . 41
  - 2.4.3 Interfaces de communication . . . 45
  - 2.4.4 Les quadri-rotors . . . 46
  - 2.4.5 Solution retenue . . . 47
- 2.5 Conclusion . . . 49

3 Estimation des paramètres de navigation **51**
3.1 Approche systémique . . . 52
3.2 Attitude . . . 54
3.3 Altitude . . . 55
3.3.1 Principe . . . 56
3.3.2 Cas des vues hybrides . . . 57
3.3.3 Homographie entre une sphère et un plan . . . 58
3.3.4 Algorithme . . . 58
3.4 Estimation de mouvement . . . 62
3.4.1 Mouvement du banc stéréo . . . 64
3.4.2 Mouvement à partir des points du sol . . . 65
3.4.3 Mouvement à partir des points de l'environnement . . . 67
3.4.4 Fusion des points 3D issus des vues mixtes . . . 67
3.4.5 Filtre de Kalman . . . 68
3.4.6 Algorithme . . . 69
3.5 Conclusion . . . 70

4 Résultats **73**
4.1 Étalonnage . . . 74
4.1.1 Comparaison avec les méthodes existantes . . . 75
4.1.2 Résultats pour un banc stéréo hybride . . . 75
4.1.3 Performances . . . 76
4.2 Attitude . . . 77
4.2.1 Estimation d'attitude à partir des points de fuite . . . 77
4.2.2 Estimation d'attitude basée sur la ligne d'horizon . . . 78
4.2.3 Influence de l'attitude sur l'altitude . . . 79
4.3 Altitude . . . 80
4.3.1 Comparaison avec une vérité terrain . . . 81
4.3.2 Résultats sur drone . . . 83
4.3.3 Performances . . . 84
4.4 Estimation du mouvement . . . 85
4.4.1 Résultats de simulation . . . 85
4.4.2 Comparaison avec une vérité terrain . . . 87
4.4.3 Résultats sur drone . . . 89
4.5 Conclusion . . . 90

Conclusion et perspectives **i**

Publications **vii**

Algorithme d'estimation du mouvement **ix**

Bibliographie xiii

Glossaire

ALTO	*Automatic Landing and Take-Off*, Décollage et atterrissage automatique. Projet financé par le FEDER et la région de Picardie.	vii, 6, 8
ARM	*Advanced Risc Machine*	iv, 39, 43, 44
CCD	*Charge-Coupled Device*, Dispositif à transfert de charge.	i, 13, 14
CMOS	*Complementary metal oxide semi-conductor*.	13
CPU	*Central Processing Unit*, Unité centrale de traitement.	iv, v, 42–44
DGPS	*Differencial Global Positioning System*.	12
DSP	*Digital Signal Processor*, Processeur de signal numérique.	iv, v, 43, 44
GPGPU	*General-Purpose computation on Graphics Processing Units*, Calcul générique sur un processeur graphique.	41, 42, 44
GPS	*Global Positioning System*.	vii, 12, 25
GPU	*Graphics Processing Unit*, Processeur graphique.	v, 41, 43, 44
HDRI	*High Dynamic Range Imaging*, Imagerie à grande gamme dynamique.	i, iv, 14
ISP	*Camera Interface Subsystem*	44
LDR	*Low Dynamic Range*, Faible gamme dynamique.	i, 14
MEMS	*Micro Electro-Mecanichal Systems*, Microsystèmes électromécaniques.	36, 37
MIMD	*Multiple Instruction Multiple Data*, Plusieurs processeurs gérés indépendamment.	43

RAM	*Random Access Memory*, Mémoire à accès non séquentiel.	43, 45
SDK	*Software Development Kit*, Trousse de développement logiciel.	38
SfM	*Structure from Motion*, Structure à partir du mouvement.	22, 23
SIFT	*Scale-invariant feature transform*, Transformation de caractéristiques visuelles invariante à l'échelle.	21, 22, 29
SIMD	*Single Instruction Multiple Data*, Une instruction est appliquée simultanément à plusieurs données.	41, 43, 44
UAV	*Unmanned Aerial Vehicle*, Drone.	1
USB	*Universal Serial Bus*, Bus de données série universel.	38, 45
VVS	*Virtual Visual Servoing*, Asservissement visuel virtuel.	i, 33, 49, 74

Table des figures

1.1 L'Aerial Target - 1916 - Source : [Aer]. 2

1.2 Différents drones pour des applications ciblées. (a) Drone militaire firebee. (b) Drone militaire pioneer. (c) Drone militaire predator. (d) Drone de reconnaissance lors de la catastrophe de Fukushima. (e) AR-Drone de Parrot à but ludique. - Sources : [Museum , Pio , Pre a, Fuk , Par] 3

1.3 Gamme d'application des drones [Onera 2005]. 4

1.4 Niveaux d'autonomie d'un drone. - Sources : [Onera 2005] 5

1.5 Thématiques de recherche sur les drones. - Sources : [GDR-GT2 2011] 5

1.6 Représentation des projets en fonction de leurs problématiques. . . 6

1.7 Situation des travaux présentés dans le plan de recherche du laboratoire. 8

1.8 6 degrés de liberté d'un drone. 10

1.9 (a) Centrale inertielle 3DM de Microstrain. (b) Capteur télémètre laser. (c) Capteur radar longue portée. (d) Capteur GPS. (e) Capteur de pression. (f) Scanner laser. - Sources : [Microstrain , Las , Rad , GPS , Pre b, URG] 11

1.10 (a) *Bracketing* ou prises de vue en fourchette - images LDR [1]. (b) Image rendue en HDRI [2]. (c) Capteur CCD [3] de 140 mega pixels de Dalsa. - Source : [HDR] 14

1.11 Différents capteurs. (a) Vue catadioptrique. (b) Vue fisheye. (c) Vue perspective. Sources : [Fis , Omn] 15

1.12 Formes géométrique des mirroirs. - Source : [Mouaddib 2005] . . 15

1.13 Système embarqué pour l'estimation d'altitude. - Source : [Sharp 2001] 17

1.14 (a) Estimation de l'attitude à partir de caméra catadioptrique et des points de fuite. (b) Estimation d'attitude à partir de caméra catadioptrique et de la ligne d'horizon. - Sources : [Bazin 2008, Demonceaux 2006b] 19

1.15 (a,b) SfM à partir de caméra catadioptrique. - Sources : [Geyer 2001, Lhuillier 2008] 21

1. *Low Dynamic Range*, Faible gamme dynamique.
2. *High Dynamic Range Imaging*, Imagerie à grande gamme dynamique.
3. *Charge-Coupled Device*, Dispositif à transfert de charge.

1.16 (a) Système composé d'une caméra et de trois pointeurs laser. (b) Disposition des capteurs de souris d'ordinateur. (c) Drone utilisé pour tester ces capteurs. - Sources : [Beyeler 2009, Sanahuja 2010b] 24
1.17 Phénomène de tavelure. - Source : [Spe] 25

2.1 (a) Drone MD4-1000 équipé du capteur de stéréovision hybride. . 29
2.2 (a) Modèle perspectif. (b) Modèle sphérique de projection centrale unifiée. - Source : [Barreto 2001] 32
2.3 Architecture du logiciel Hyscas. 34
2.4 (a) Déclinaison du champ magnétique. (b) Champ magnétique terrestre et nord géographique. - Source : [Nor] 36
2.5 Élément vibrant des gyroscopes. - Source : [One] 36
2.6 (a) Centrale inertielle MTi de XSens. (b) Centrale compacte 3DM-GX3-25 de Microstrain. - Sources : [MTI , Microstrain] 38
2.7 (a) Lecture et écriture dans une mémoire tampon circulaire. (b) Accès concurrent en lecture/écriture d'une même zone mémoire. . . 39
2.8 (a) Algorithme de lecture dans une mémoire tampon circulaire. (b) Algorithme d'écriture dans une mémoire tampon circulaire. 40
2.9 Schéma des connections pour la synchronisation des caméras IDS μEye LE12xx. - Source : [IDS] 41
2.10 Comparaison des architectures CPU et GPU. 42
2.11 (a) Gumstix composé d'une puce OMAP 3530. (b) Puce OMAP 3530. - Sources : [Pixhawk 2010, OMA] 43
2.12 Carte à base de processeur Atom d'Intel. 45
2.13 (a) Pellican - Ascending Technologies. (b) MD4-1000 - Microdrones. (c) Drone quadri-rotor - Heudiasyc, UTC. 47
2.14 Solution matérielle retenue. 48

3.1 Approche encaplusée. 52
3.2 Système global d'estimation des paramètres de vol à partir du capteur de stéréovision hybride. 53
3.3 Estimation des points de fuite à partir des lignes dans l'image en environnement clos. 55
3.4 Estimation de l'attitude basée sur la ligne d'horizon (a) Estimation à partir d'une vue catadioptrique. (b) Estimation à partir d'une vue fisheye. (c) Image de l'horizon projetée sur la sphère équivalente. . 56
3.5 Plane-sweeping entre deux vues hybrides. 58
3.6 Homographie sphere/plan. 59
3.7 (a) Mise à jour du masque en fonction de l'itération (erreur en fonction de l'altitude en mm). (b) Tableau de valeurs correspondant. . 60

3.8 (a) Vue perspective. (b) Vue catadioptrique (c) Homographie de la vue omnidirectionnelle sur la vue perspective finale. (d) Masque final 62
3.9 (a) Algorithme standard - Homographie de la vue courante vers la référence pour chaque pixel référence. (b) Algorithme optimisé - Calcul de la zone de traitement pour la LUT. (c) Algorithme optimisé - Homographie de la vue courante vers la référence par LUT. 63
3.10 LUT entre l'image fisheye et la sphère. 64
3.11 Principe général de la méthode d'estimation du mouvement. . . . 65
3.12 Points 3D et mouvement. 67

4.1 (a) Drone MD4-1000 avec système embarqué. (b) Pellican avec système embarqué. 74
4.2 (a) Version windows avec reprojection des poses en 3D. (b) Version mac avec miniatures des images des mires. 75
4.3 Des points d'intérêt sont sélectionnés dans une image perspective (a) et les courbes épipolaires correspondantes sont calculées et tracées dans l'image fisheye (b). (c), (d), (e) et (f) montrent plus précisément la partie de conique épipolaire correspondant aux points sélectionnés dans l'image perspective. 76
4.4 (a) Détection des contours pour la détection de droites. (b) Estimation des droites. 77
4.5 Comparaison, de haut en bas, du roulis, tangage et lacet par vision (bleu) et par centrale inertielle (rouge). 78
4.6 (a) Vue fisheye du drone. (b) Segmentation sol/air à partir de la ligne d'horizon. 79
4.7 Influence de l'erreur d'attitude sur l'estimation de l'altitude. . . . 80
4.8 Les croix représentent l'altitude estimée par *plane-sweeping* et la ligne représente les mesures idéales issues du télémètre laser. . . . 81
4.9 Altitude et segmentation du sol - 4.8% d'*inliers* - (a) Vue fisheye, (b) Vue perspective, (c) Segmentation du sol, (d) Homographie sphère/plane, (e) Comparaison vue de référence et homographie. . 82
4.10 (a,b) Vue embarquée du Pelican (de gauche à droite : la vue fisheye et perspective) - Altitude estimée : 1912mm. (c) Homographie de la vue fisheye vers la vue perspective. (d) Erreur. (e) Segmentation du sol $\mathbf{G}$. (f) Altitude de la trajectoire estimée par *plane-sweeping*. 83
4.11 Erreur angulaire entre la translation réelle (en mm) et estimée en degrés. 86
4.12 Erreur de distance entre la translation réelle et estimée. 87
4.13 Boucle (d) du robot (c) à partir des vues hybrides et fusionnées, (a) vue fisheye du robot, (b) vue perspective du robot. 88

4.14 (a) Trajectoire 3D d'un drone en milieu clos. (b) Données brutes et données filtrées par Kalman. 89
4.15 Mouvement en intérieur avec le Pellican. 90

Liste des tableaux

1.1 Tableau récapitulatif des principaux travaux de recherche menés sur les drones. 7
1.2 Comparaison des méthodes basées sur la vision pour l'estimation de l'altitude. 18
1.3 Comparaison des méthodes basées sur la vision pour l'estimation de l'attitude. 20
1.4 Comparaison des méthodes basées sur la vision pour l'estimation du mouvement. 23

4.1 Temps de calcul pour l'étalonnage en utilisant le logiciel Hyscas. Chaque pose est une image pour une caméra, deux images pour deux caméras, etc. Trente-six points sont utilisés pour chaque image. 76
4.2 Comparaison de l'attitude entre une centrale inertielle et la méthode basée sur la vision. 78
4.3 Altitude et segmentation du sol avec et sans obstacles - Paramètres de l'algorithme : $s = 6$, $thres = 25$. 82
4.4 Répartition du nombre de points en fonction de leur classification et des vues. 86

Introduction

Ces dix dernières années, l'intérêt porté par la communauté scientifique à la thématique des drones et des micro-drones n'a cessé de s'accroitre.

Ces engins sans pilote ont la capacité de voler en milieu extérieur ou intérieur (pour les plus compacts). Ils sont utilisés à la fois pour des applications militaires et civiles. En effet, le domaine militaire les utilise principalement comme relais de communication, comme sentinelles et pour l'évaluation de zones sensibles. Le domaine civil, quant à lui, les emploie en majorité pour des missions de cartographie, de surveillance du trafic et des missions de recherche et de sauvetage.
Ces applications militaires et civiles nécessitent une navigation précise et indépendante de l'intervention humaine, alors que les drones sont aujourd'hui en partie radio-commandés. D'une part, cette navigation quasi autonome est réalisée par la mise en place de points de passage (*check-points*) définis avant un vol par GPS [4]. D'autre part certaines méthodes proposent d'utiliser des cibles ou d'autres dispositifs nécessitant la connaissance du terrain ou un équipement spécifique au sol.
Les travaux actuels du domaine en rapport avec les drones visent majoritairement l'amélioration de leur autonomie liée à l'environnement. Les aspects les plus contraignants à résoudre tant pour les pilotes humains que les pilotes automatiques sont les phases critiques de décollage et d'atterrissage.
Ainsi, l'objet de ce travail de recherche porte sur les axes d'amélioration de l'autonomie de ces aéronefs lors d'un vol complet, c'est à dire le décollage, la navigation et l'atterrissage.

Définition du problème

Pour augmenter l'autonomie de navigation, les contraintes sont les suivantes : la compacité, la faible consommation en énergie et la détermination des paramètres de navigation comme la vitesse, l'altitude et l'assiette en temps réel. Ces informations se destinent à être intégrées dans la boucle de commande et mises à jour rapidement pour maintenir la stabilité de l'engin et le déplacer dans l'espace. Ainsi, ce sont ces paramètres de navigation que les travaux de recherche actuels ou passés visent à déterminer en temps réel à partir de capteurs dédiés ou de la fusion de capteurs.

La problématique du projet ALTO [5] dans lequel s'inscrit cette thèse est l'esti-

4. *Global Positioning System*.
5. *Automatic Landing and Take-Off*, Décollage et atterrissage automatique. Projet financé par le FEDER et la région de Picardie.

mation des paramètres de vol en particulier pour les phases critiques de décollage et d'atterrissage des drones compacts. Les contraintes liées à cette problématique sont à la fois matérielles et scientifiques.
D'une part au niveau matériel, il est nécessaire d'élaborer un capteur compact, peu consommateur d'énergie et capable de déterminer un nombre important de paramètres en temps réel. De plus, l'électronique liée doit répondre à des conditions de vibration et de perturbation électromagnétique. L'une des solutions consiste à utiliser plusieurs capteurs. Cependant, cette solution présente les inconvénients de limiter leur fonction à la tâche à laquelle ils sont destinés, d'augmenter la consommation énergétique, l'encombrement et d'accroître le risque de panne. Une seconde possibilité est d'utiliser des capteurs plus complexes : les systèmes de vision. Ces systèmes permettent en effet d'apporter une information riche d'un environnement tout en étant compacts et économes en énergie.
D'autre part, au niveau scientifique, les données collectées par les caméras sont complexes. En effet, les images obtenues nécessitent un traitement de l'information brute afin de déterminer les paramètres de navigation. En parallèle, ces informations peuvent être exploitées pour d'autres tâches comme des missions d'observation. De plus, pour obtenir une mise à jour en temps réel des paramètres estimés, des méthodes rapides et optimisées doivent être implémentées.
Ces systèmes de vision se distinguent en deux catégories :

- les caméras omnidirectionnelles disposent d'un large champ de vue mais la résolution n'est ni constante, ni homogène.
- les caméras perspectives disposent d'une résolution plus importante et constante mais le champ de vue est restreint.

Ces deux types de capteurs et leurs caractéristiques peuvent par ailleurs être combinés pour former un banc stéréoscopique "hybride". Cependant, une telle configuration pose des difficultés de mise en œuvre et nécessite une modélisation particulière. Nous proposons dans cette thèse de modéliser le capteur omnidirectionnel par une sphère unitaire et le capteur perspectif par un plan. Cette dernière modélisation conserve l'aspect discret du voisinage planaire et autorise les traitements classiques. Ainsi, le calcul des paramètres de navigation est rendu plus rapide à condition d'éviter des méthodes de mise en correspondance. La modélisation sphérique et planaire des deux capteurs nécessite des traitements adaptés.

Pour répondre à la problématique posée, l'hypothèse proposée dans cette thèse est le développement d'un capteur de vision hybride omnidirectionnel et perspectif capable de fournir des informations de navigation comme l'altitude, l'attitude et le mouvement.

Les contributions liées aux travaux proposés peuvent être présentés en quatre points.

- La contribution générale est la conception d'un capteur de stéréovision hy-

bride. À partir de ce capteur, une approche systémique est proposée.
D'abord, les méthodes implémentées sont encapsulées ; c'est à dire que pour chaque paramètre estimé au plus haut niveau est réemployé par le niveau inférieur. L'ensemble des interactions au sein de ce système encapsulé est ensuite illustré par l'aspect collaboratif du système. En effet, trois modes de fonctionnement le caractérisent :
- séparé : chaque capteur est utilisé pour une tâche propre ;
- stéréoscopique : les deux capteurs sont utilisés simultanément pour déterminer un ou plusieurs paramètres ;
- coopératif : destiné à l'échange de données entre les caméras pour enrichir et améliorer la perception de certains paramètres comme le mouvement.

Enfin, les méthodes proposées permettent de s'affranchir de l'environnement exploré (urbain, intérieur ou naturel) et fournissent l'ensemble des paramètres nécessaires à la navigation autonome d'un drone.

– La deuxième contribution porte sur l'estimation de l'altitude. Les travaux actuels se distinguent en deux catégories : les méthodes mono et stéréovision. D'une part, les travaux mono-caméra sont basés sur le suivi de cible ou sur l'appréciation de la proximité du sol. D'autre part, les méthodes de stéréovision déterminent l'altitude grâce à la mise en correspondance de primitives. La méthode de stéréovision proposée dans cette thèse s'affranchit de toute mise en correspondance. De plus, elle détermine l'altitude à l'échelle métrique et segmente le plan du sol sans utiliser de marqueur artificiel. L'algorithme ainsi obtenu semble être précis et en temps réel.
– La troisième contribution porte sur l'estimation du mouvement en exploitant l'aspect collaboratif et stéréoscopique du capteur. Ce mouvement est d'abord découplé en rotation et translation. La rotation est déterminée soit par vision, soit par centrale inertielle. La translation est ensuite déterminée à partir de deux types de points suivis indépendamment dans chacune des caméras. Parmi ces points, ceux localisés sur le sol fournissent l'échelle métrique du mouvement alors que les autres améliorent la robustesse au bruit. Ainsi, les paramètres d'estimation d'attitude, d'altitude et de segmentation du sol sont directement encapsulés et utilisés par cette méthode pour déterminer la rotation mais aussi la profondeur des points localisés sur le sol.
  Enfin, cette méthode présente l'avantage d'être modulaire, adaptable à tout type de vues et adaptable pour n caméras.
– La quatrième contribution est l'élaboration d'un logiciel de calibration hybride. A notre connaissance, dans l'état de l'art, nous n'avons pas rencontré d'outils libres capables de déterminer les paramètres de calibrage d'un système hybride. Pour étalonner une telle configuration, nous avons choisi de développer un nouvel outil capable de calibrer n caméras représentées par n modèles.

Le plan de présentation de ce travail de recherche s'articule en quatre chapitres. Un premier chapitre présente les drones, leur évolution dans un environnement 3D et les capteurs capables de fournir les informations de vol. Le deuxième chapitre traite du capteur hybride, de sa modélisation et sa calibration. Cette partie propose une architecture matérielle adaptée à l'application embarquée. Le troisième chapitre développe les méthodes proposées pour l'estimation d'altitude, d'attitude et de mouvement par vision hybride. Enfin, les résultats des méthodes proposées dans le troisième chapitre sont détaillés dans le quatrième chapitre. Ce chapitre valide expérimentalement, en embarqué, l'ensemble des méthodes proposées et met en avant les avantages et perspectives des algorithmes.

CHAPITRE 1

La navigation autonome des drones et la vision

Sommaire

1.1 L'histoire des drones 1
1.2 L'autonomie 5
1.3 Les projets 6
1.4 Dynamique et paramètres d'état d'un quadri-rotor 8
1.5 Les capteurs 10
1.6 La vision 13
1.6.1 Les mécanismes de formation d'une image 13
1.6.2 Estimation des paramètres d'état 16
1.7 Conclusion 25

1.1 L'histoire des drones

Les drones, ou encore UAV [1] sont des aéronefs sans pilote. Les premiers drones ont vu le jour en 1916 aux États-Unis avec l'*Aerial Target* (figure 1.1), un avion cible à vocation militaire. Les drones ont ensuite évolué. Ils ont été massivement utilisés pour des applications militaires. Pendant la première guerre mondiale, des avions sans pilote étaient radio commandés et équipés d'un gyroscope. En France, il faut attendre 1960 pour voir l'apparition des premiers drones avec le *R 20* de *Nord-Aviation*, engin dérivé de l'engin cible *CT 20* [Onera 2009a]. Durant la guerre 39-45, les fusées *V2*, considérées comme des bombes volantes, sont assimilées à une classe d'aéronefs sans pilotes. Ces engins sont par la suite améliorés et utilisés pour les guerres du Vietnam, Corée, Kosovo. Les principales missions étaient la localisation de rampes de missiles sol-air (Vietnam) fig.1.2(a), la surveillance et le réglage d'artillerie (guerre du Golfe) fig.1.2(b). Leur essor et leurs capacités opérationnelles furent démontrés pendant les conflits de l'ex-Yougoslavie, en Irak et en Afghanistan. Ils sont aujourd'hui principalement déployés lors des missions

1. *Unmanned Aerial Vehicle*, Drone.

de reconnaissance ou d'attaque fig.1.2(c).

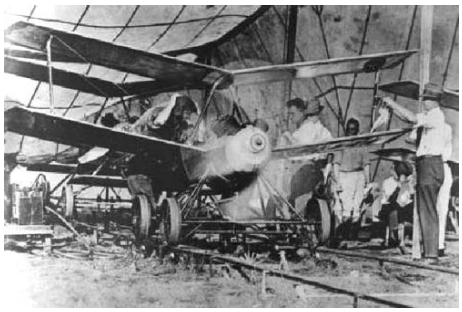


FIGURE 1.1 – L'Aerial Target - 1916 - Source : [Aer].

D'un point de vue civil, l'emploi des drones a considérablement évolué notamment grâce à la miniaturisation, à l'augmentation de puissance et au développement à faible coût des capteurs et unités de traitement. Ils sont de plus en plus utilisés pour des missions civiles de reconnaissance ou d'observation comme les études scientifiques de l'atmosphère, la surveillance d'incendies, le sauvetage, l'évaluation des dégâts en cas de catastrophe naturelle, la surveillance maritime, le trafic routier, le transport, la cartographie, etc. Ils ont un rôle prépondérant dans les missions de sauvetage comme en 2008 lors d'ouragans qui ont frappé la Louisiane et le Texas ou encore en 2011 dans l'explosion de la centrale nucléaire de Fukushima où la présence humaine était alors impossible à cause des fortes radiations fig.1.2(d). Ces aéronefs ont commencé à entrer sur le marché professionnel avec les Microdrone, les Novadem, le Pelican et même dans les foyers avec l'AR.Drone de Parrot fig.1.2(e). Les vocations de ces aéronefs sont la recherche, la photographie aérienne, et l'utilisation ludique. Toutefois, l'utilisation civile pose certaines contraintes opérationnelles comme la navigation et l'intégration dans l'espace aérien, la législation, la vulnérabilité au brouillage et aux ruptures de transmissions et certaines contraintes techniques comme la transmission de données, la discrétion, la motorisation, les pannes de vol, etc.

Les drones sont donc utilisés pour une large gamme d'applications (fig. 1.3, [Onera 2005]). Ils sont caractérisés par la taille et les performances [Onera 2009b, Onera 2005]. Ces performances sont définies par le rayon d'action, l'endurance et l'altitude.

- L'endurance se définit par le temps de vol d'un drone. Elle est comprise entre 20 à 40 heures.
- L'altitude est segmentée en trois zones : les basses altitudes (inférieures à 5000m), les moyennes altitudes (entre 5000m et 15000m, fig. 1.2(c)) et hautes altitudes (supérieures à 15 000m).

(a)

(b)

(c)

(d)

(e)

FIGURE 1.2 – Différents drones pour des applications ciblées. (a) Drone militaire firebee. (b) Drone militaire pioneer. (c) Drone militaire predator. (d) Drone de reconnaissance lors de la catastrophe de Fukushima. (e) AR-Drone de Parrot à but ludique. - Sources : [Museum , Pio , Pre a, Fuk , Par]

En prenant en compte ces critères de performance, les drones sont classifiés en trois catégories :

- Drones MALE (Moyenne Altitude, Longue Endurance) charge <=500kg (ex. le predator fig. 1.2(c))
- Drones HALE (Haute Altitude, Longue Endurance)
- Drones tactiques TUAV

D'un point de vue tactique, en fonction de la mission à réaliser, on distingue cinq types de drones :

- **micro-drones MAV (Micro Air Vehicles)**
- drones de très courte portée (TCP)
- drones moyenne portée lents (MCMM lents)
- drones rapides basse altitude (MCMM rapides)
- drones maritimes tactiques (DMT).

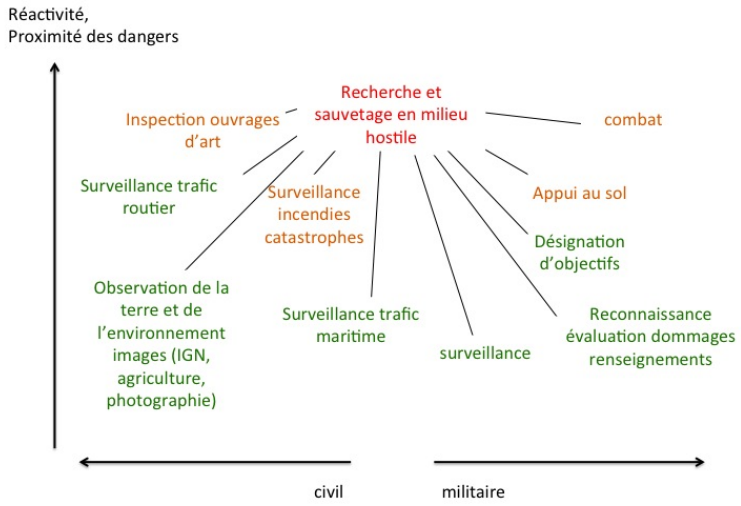


FIGURE 1.3 – Gamme d'application des drones [Onera 2005].

Les travaux présentés dans ce travail de recherche s'inscrivent dans la catégorie MAV, première catégorie énoncée ci-dessus.

Malgré les avancées technologiques en terme de miniaturisation et de développement, les drones sont majoritairement dépendants d'une présence humaine pour un pilotage déporté complet ou assisté. C'est la raison pour laquelle les axes de recherche actuels tendent à améliorer la navigation autonome. Par exemple, les travaux de [Andert 2010] proposent une solution utilisant des repères sur le sol ; alors que ceux de [Fabiani 2007] proposent une méthode qui permet de s'affranchir de ce type de contrainte.

1.2 L'autonomie

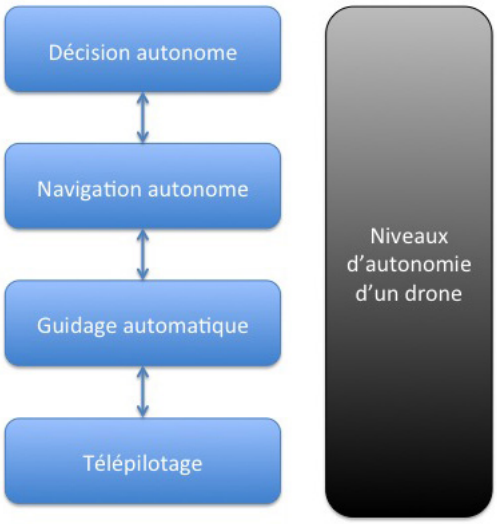


FIGURE 1.4 – Niveaux d'autonomie d'un drone. - Sources : [Onera 2005]

Les différents niveaux d'autonomie des drones (fig. 1.4, [Onera 2005]) sont l'objet d'étude de la part des roboticiens et des automaticiens. Les problématiques liées à cette autonomie sont regroupées en cinq thématiques (fig.1.5, [GDR-GT2 2011]).

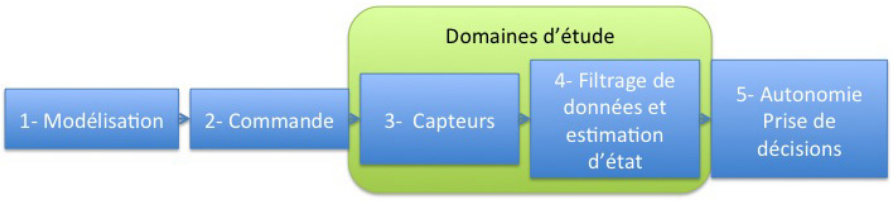


FIGURE 1.5 – Thématiques de recherche sur les drones. - Sources : [GDR-GT2 2011]

- la modélisation du système mécanique définit le modèle aérodynamique
- la synthèse de la commande : la partie commande doit faire face aux problématiques classiques de l'automatique mais également à la robustesse aux phénomènes aérologiques et à l'économie d'énergie
- la partie capteurs : cette partie vise à estimer les paramètres de vol nécessaires à l'estimation d'état. Les paramètres à déterminer sont définis dans la phase de modélisation

– le filtrage des données et l'estimation d'état : cette partie vise à développer de nouvelles techniques d'estimation d'état basée sur le modèle aérodynamique du drone ou à partir des capteurs et de leur dynamique
– la phase de prise de décision : on parle alors d'autonomie décisionnelle

1.3 Les projets

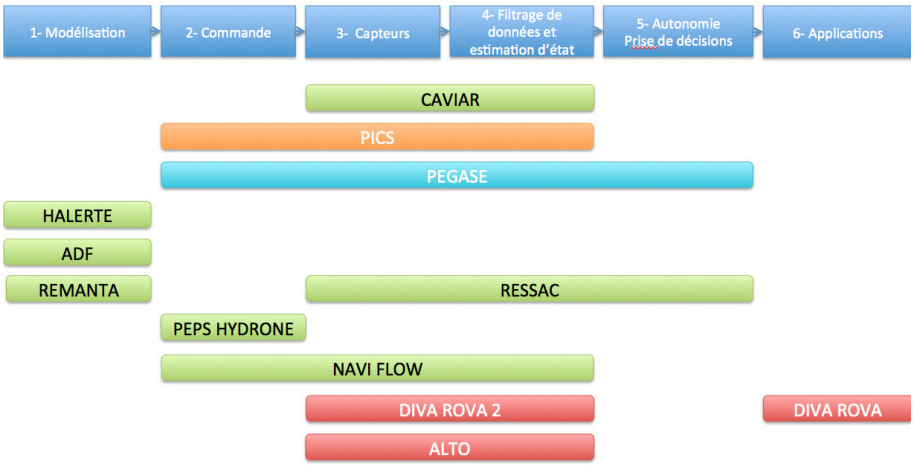


FIGURE 1.6 – Représentation des projets en fonction de leurs problématiques.

Depuis une dizaine d'années, plusieurs projets sont menés sur l'amélioration de l'autonomie des drones. Ils visent à trouver des solutions pour l'élaboration de systèmes aéronautiques fonctionnels et autonomes [Onera 2005]. Le tableau 1.1 présente les différents projets, leur envergure et la problématique abordée. Les principaux projets présentés sont ensuite classifiés dans la figure 1.6 en fonction du domaine d'étude qu'ils abordent.

Ces projets ont pour objectif commun l'augmentation de l'autonomie. Cependant, une problématique principale se pose : la stabilisation. La stabilisation est possible grâce à l'étude des caractéristiques nécessaires au système pour évoluer dans son environnement : l'attitude du drone, son altitude et son déplacement.

Le projet ALTO, auquel cette thèse s'intéresse, vise à développer le décollage et l'atterrissage automatique tout terrain d'un robot volant. Il se dissocie en deux parties principales. La première partie, objet de cette thèse, consiste à élaborer un capteur hybride omnidirectionnel/perspectif, le calibrer et développer des méthodes originales et efficaces pour l'estimation de l'attitude, l'altitude, des zones libres et des vitesses de déplacement. L'enjeu majeur est d'aboutir au développement de méthodes rapides (entre 30 Hz et 40 Hz) avec de bonnes précisions et robustesse

Projet	Envergure	Problématique	Année
CAVIAR	National	Apport de la vision omnidirectionnelle dans la robotique aérienne	06-08
PICS	International	Asservissement visuel de drone miniature	06-08
PEGASE	Européen	Asservissement d'avion et d'hélicoptère pour l'atterrissage automatique	06-09
ROBOA	National	Modélisation et commande par asservissement visuel du drone HoverEye.	03-04
ROBVOLINT	National	Autonomie d'un micro drone évoluant en environnement intérieur.	03-06
SCUAV	National	Commande de drones référencée multi capteurs.	07-09
ONERA			
PEA Action	National	Étude de coopération de multi-véhicules hétérogènes	07-14
HALERTE	National	Conception de drone pour le haute altitude et longue endurance	-
ADF	National	Simulation et modélisation	-
CITHAR	National	Nouvelles architectures informatiques, traitement distribué de l'information	-
SICODIS	National	Simulation distribuée	-
RESSAC	National	Augmentation de l'autonomie : contrôle de vol, acquisition et traitement d'informations, décisions embarquées	-
REMANTA	National	Dynamique de vol, aérodynamisme, matériaux et actionneurs	-
PEPSHYDRONE	National	Automatique, commande	-
NAVI FLOW	National	Navigation par flot optique	-
DIVA-ROVA	Régional	Surveillance	01-04
DIVA-ROVA2	Régional	Calcul de l'attitude par vision omnidirectionnelle	04-07
ALTO	Régional	Décollage et atterrissage tout terrain par vision	08-11

TABLE 1.1 – Tableau récapitulatif des principaux travaux de recherche menés sur les drones.

pour envisager des manœuvres automatiques. La seconde partie a pour objectif de contrôler l'engin dont le problème majeur est l'obtention d'une assiette parfaitement stable. Dans cette partie, qui concerne un autre travail de thèse, des méthodes

de fusion de données sont proposées et visent à améliorer cette stabilité.

L'objectif est donc de pouvoir remplacer les capteurs d'altitude, d'attitude et de vitesse utilisés dans la boucle de commande par le capteur de vision hybride.

Ce projet ALTO s'inscrit au sein des axes de recherche de l'équipe de vision du laboratoire MIS et est réalisé en collaboration avec le laboratoire MIS et Heudiasyc. Au sein du laboratoire Heudiasyc, les projets menés s'intéressent à la commande des drones mais également la perception pour la navigation autonome. Au laboratoire MIS, l'équipe Perception-Robotique traite des problématiques de vision omnidirectionnelle pour la navigation autonome des robots terrestres et aériens. La figure 1.7 donne un aperçu de ces projets et des thématiques de recherche abordées.

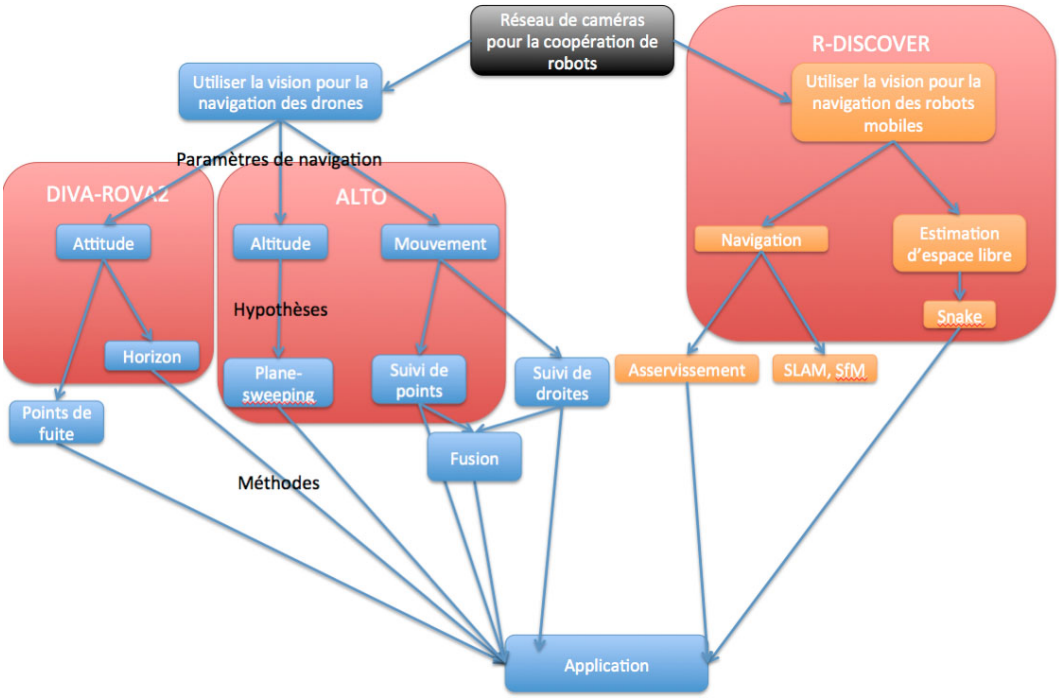


FIGURE 1.7 – Situation des travaux présentés dans le plan de recherche du laboratoire.

1.4 Dynamique et paramètres d'état d'un quadrirotor

Le modèle dynamique du drone permet de définir les paramètres d'état nécessaires à sa stabilité. Nous cherchons à les déterminer par l'intermédiaire de cap-

teurs.

Afin d'écrire les équations de dynamique du drone, définissons d'abord un repère fixe, lié au monde et noté $\mathcal{R}_m = (X_m, Y_m, Z_m)$. L'origine du repère du drone est noté $\mathcal{R}_t = (X_t, Y_t, Z_t)$ (fig.1.4). L'évolution de ce repère varie au cours du temps t en fonction du repère monde $\mathcal{R}_m$. Le passage du repère monde au repère du drone s'effectue par une matrice ${}^t\mathbf{R}_m$ définie dans l'eq.1.4, avec α, β, γ les angles d'Euler :

- rotation α autour de X_m, avec $\alpha \in [-\pi, \pi]$,
- rotation β autour de Y_m, avec $\beta \in [-\frac{\pi}{2}, \frac{\pi}{2}]$,
- rotation γ autour de Z_m, avec $\gamma \in [-\pi, \pi]$.

Ces angles représentent respectivement les angles de roulis, tangage et de lacet. Les matrices de rotation associées sont :

$$\mathbf{R}_\alpha = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \tag{1.1}$$

$$\mathbf{R}_\beta = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \tag{1.2}$$

$$\mathbf{R}_\gamma = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1.3}$$

$${}^t\mathbf{R}_m = \mathbf{R}_\gamma \mathbf{R}_\beta \mathbf{R}_\alpha \tag{1.4}$$

La matrice de changement de repère ainsi que les rotations définies précédemment constituent l'attitude du drone par rapport au repère monde.

Ensuite, le mouvement du système est composé d'une translation et d'une rotation. L'équation (1.5) caractérise le déplacement relatif entre le repère monde et le repère drone.

$${}^t\mathbf{t}_m = \begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} - {}^t\mathbf{R}_m \begin{bmatrix} X_m \\ Y_m \\ Z_m \end{bmatrix} = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}. \tag{1.5}$$

Le déplacement relatif entre une position et la position précédente est obtenu dans l'équation 1.6.

$$\mathbf{t}_{t+1} = {}^{t+1}\mathbf{t}_m - {}^t\mathbf{t}_m \quad \text{et} \quad \mathbf{R}_{t+1} = {}^{t+1}\mathbf{R}_m \, {}^t\mathbf{R}_m^{-1} \ . \tag{1.6}$$

Dans les équations d'état, le drone peut être caractérisé par son attitude (α, β, γ) et son altitude d par rapport au repère terrestre. La rotation $\mathbf{R}_{t+1}$ et la translation

FIGURE 1.8 – 6 degrés de liberté d'un drone.

$\mathbf{t}_{t+1}$ caractérisent le mouvement entre deux positions successives du drone, à deux instants donnés. Dans cette partie, les inconnues du drone sont identifiées (l'attitude, l'altitude, le mouvement). Ces inconnues sont déterminées à l'étape suivante à l'aide de capteurs.

1.5 Les capteurs

Les capteurs constituent l'une des parties cruciales pour la stabilité d'un aéronef car ils permettent de déterminer les paramètres d'état. La défaillance de l'un de ces capteurs peut mener à un dysfonctionnement dans la boucle de commande, à une instabilité et un accident. Leur rôle est de convertir une grandeur physique en un signal analogique, numérique ou logique. Ils sont caractérisés par quatre critères : l'étendue de mesure, la sensibilité, la rapidité et la précision. D'un point de vue énergétique, il est intéressant de discerner les capteurs actifs des capteurs passifs. La première catégorie émettent et reçoivent un signal, converti ensuite en énergie électrique. De ce fait, il sont consommateurs d'énergie et potentiellement détectables. Les radars, les laser, les ultrasons appartiennent à cette famille. Puis, la deuxième catégorie se contente de convertir un signal perçu en énergie électrique. Ainsi ce type de capteur consomme peu d'énergie, est discret mais dépend de la quantité d'énergie émise par un environnement. Ils peuvent être illustrés par les caméras, les baromètres, les capteurs inertiels (fig. 2.6(b)), etc. Pour un aéronef, les plus utilisés sont les capteurs barométriques (fig. 1.9(e)), les capteurs à ultrason, les capteurs inertiels (fig. 2.6(b)), les lasers (fig. 1.9(b),1.9(f)), les radars (fig. 1.9(c)) et les récepteurs GPS (fig. 1.9(d)).

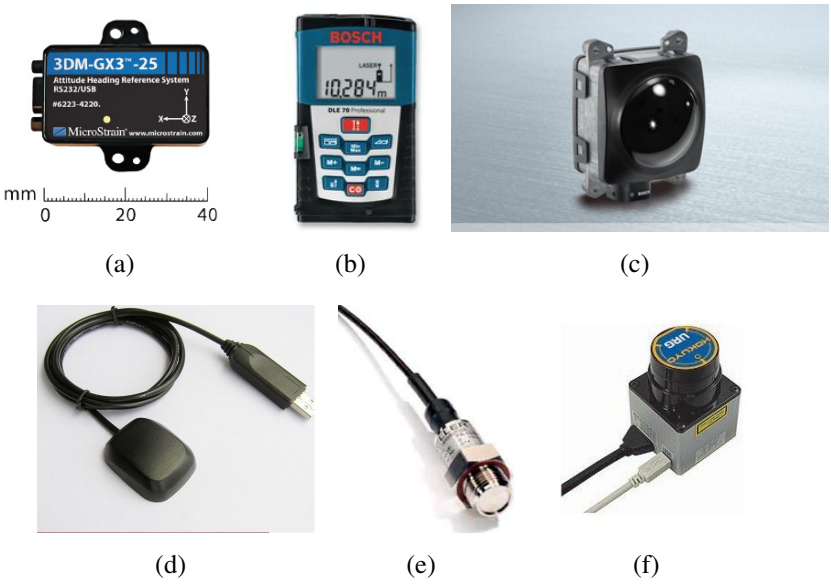


FIGURE 1.9 – (a) Centrale inertielle 3DM de Microstrain. (b) Capteur télémètre laser. (c) Capteur radar longue portée. (d) Capteur GPS. (e) Capteur de pression. (f) Scanner laser. - Sources : [Microstrain , Las , Rad , GPS , Pre b, URG]

Estimation des paramètres d'état

Cette partie propose de classifier les capteurs en fonction des paramètres à estimer. Certains capteurs peuvent déterminer un seul paramètre à la fois comme l'altitude, l'attitude ou le mouvement tandis que d'autres peuvent en déterminer plusieurs simultanément.

L'altitude

Pour déterminer l'altitude, les altimètres barométriques sont largement utilisés sur les avions mais dépendent des variations de pression et de température. Cette variation peut impliquer entre 6% et 7% d'erreur en précision.

Les altimètres laser sont d'autres capteurs précis mais leur fonctionnement requiert des conditions de réflexion spécifiques de la surface observée. [Vasconcelos 2010] propose d'utiliser un laser pour déterminer la distance au sol durant les phases critiques de décollage et d'atterrissage. Ce travail est utilisé sur hélicoptère à moteur thermique. Cependant, l'inconvénient de ce système réside dans l'encombrement, la consommation et le poids du système. La limite principale de la méthode est la tâche unique à laquelle le laser est destiné.

L'attitude

Les magnétomètres, gyroscopes, gyromètres et accéléromètres sont des capteurs embarqués dans les centrales inertielles. En utilisant ce type de capteur, il est possible de retrouver l'orientation en fonction du nord magnétique (fig.2.6(b)) mais aussi de déterminer l'attitude à partir des mesures d'accélération et de vitesse angulaire. Les avantages sont une mise à jour rapide des données (1000 Hz) et précise (erreur <0,5 °).
Néanmoins ces capteurs présentent une sensibilité aux perturbations magnétiques du drone, une accumulation d'erreurs et une dérive au cours du temps. C'est pourquoi [Euston 2008] propose d'utiliser un couple GPS/inertiel pour estimer l'attitude d'un drone où la sortie brute de l'accéléromètre participe à l'estimation de l'attitude à basse fréquence tandis que le gyromètre contribue à déterminer l'attitude à haute fréquence. L'information gravitationnelle quant à elle permet une mesure de l'orientation. Ce système permet d'exploiter des capteurs à faible coût, à faible encombrement et dispense d'une électronique dédiée.

Le mouvement

Les GPS sont peu précis à la fois verticalement (environ 25m) et horizontalement (environ 15m). Ils sont sensibles aux interruptions de transmission dans les environnements urbains et au brouillage. D'un point de vue stratégique, ce système est conçu par les État-Unis et demeure sous leur contrôle, d'où la réponse par la mise en place du système européen Galileo ou russe Glonass. Ils ont la capacité d'estimer l'altitude (de façon peu précise), le mouvement et les vitesses. Leur avantage est la possibilité de connaître la position d'un récepteur GPS capable de recevoir le signal d'au moins quatre satellites. La qualité de réception et la quantité de satellites définissent la précision de l'estimation de position.
D'autres systèmes, DGPS [2], beaucoup plus précis, réduisent l'erreur d'un GPS traditionnel grâce à l'ajout d'une station au sol. Toutefois, un tel système dépend d'une station fixe ce qui rend son déploiement contraignant.

Capteurs multi-paramètres

Les radars quant à eux sont capables d'estimer à la fois l'altitude, la vitesse et le plan du sol avec une bonne précision. Néanmoins, ces capteurs sont actifs, potentiellement détectables, consommateurs d'énergie et peu compacts contrairement aux capteurs de vision.

2. *Differencial Global Positioning System*.

Nous venons de faire un bilan des différents capteurs utilisables pour estimer les paramètres d'état d'un aéronef. Nous avons vu que certains capteurs déterminent un paramètre unique alors que d'autres estiment plusieurs paramètres simultanément. Utiliser plusieurs capteurs augmente l'encombrement et la consommation énergétique. Pour pallier ces contraintes, nous proposons d'exploiter des capteurs de vision plus complexes en terme de traitement de l'information, compacts, passifs et économes en énergie. Ces capteurs peuvent être utilisés seul ou couplés à une centrale inertielle. Ils font l'objet d'une étude plus approfondie dans la prochaine partie.

1.6 La vision

1.6.1 Les mécanismes de formation d'une image

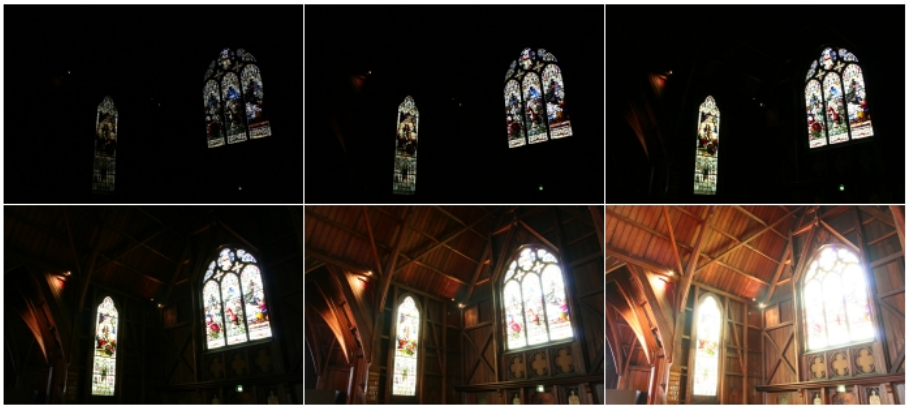
La vision est un moyen d'obtenir une information brute beaucoup plus riche de l'environnement observé. Ces systèmes se composent d'une surface sensible, les photo-sites, partie propre du capteur et d'une électronique de numérisation. Ensuite, une optique et/ou miroir est jointe à la zone de photo-sites. La qualité, la construction et les traitements de surface définissent la qualité de l'image formée. L'obtention d'une image de qualité est donc le résultat d'un choix minutieux d'un couple capteur/optique. Enfin, lorsque l'image est acquise, l'information dénaturée ou manquante nécessitera une étape de traitement d'image ou de restauration. Cependant, dans la plupart des cas, l'information non capturée ne pourra pas être retrouvée. Pour la navigation, il est important d'obtenir une information la plus précise possible avec un équilibre résolution/temps de traitement le meilleur possible.

Les matrices de photo-sites

On distingue deux types de matrices : les CCD et les CMOS [3]. Les CCD possèdent une bonne sensibilité, délivrent un courant assez important et sont plus rapides que les CMOS. Ils requièrent peu d'amplification et sont donc peu sensibles au bruit. Toutefois, ces capteurs sont consommateurs d'énergie et coûteux à fabriquer. Les CMOS en revanche sont beaucoup plus économiques à fabriquer et sont économes en énergie. Ils nécessitent une plus forte amplification et sont donc plus sensibles au bruit. Mais si les capteurs CCD ont longtemps dominé le marché grâce à leur qualité supérieure, la tendance s'inverse depuis peu en faveur des capteurs CMOS. L'évolution du traitement du bruit, de la qualité et du coût de construction en font aujourd'hui des capteurs incontournables.

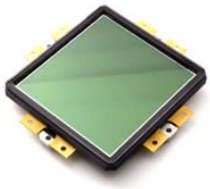
3. *Complementary metal oxide semi-conductor*.

Ces photo-sites sont caractérisés par leur profondeur de couleur et leur résolution. En référence, la profondeur de couleur d'un écran est de 8bits par canal de couleur. Les capteurs sont aujourd'hui capables de fournir un échantillonnage sur 14bits. Ces capteurs sont dit HDRI (figures 1.10(a) et 1.10(b)). Ils permettent entre autre d'obtenir une meilleure dynamique dans les zones sombres et claires que sur un échantillonnage en 8bits. Concernant la quantité d'informations perçues, pour une même taille de capteur, plus la résolution est importante, plus la taille des photo-sites est petite. De ce fait, les photo-sites perçoivent une quantité de lumière plus faible et deviennent alors plus sensibles au bruit lié à l'amplification. Aujourd'hui les capteurs peuvent atteindre jusqu'à 140MP fig.1.10(c).



(a)

(b)



(c)

FIGURE 1.10 – (a) *Bracketing* ou prises de vue en fourchette - images LDR. (b) Image rendue en HDRI. (c) Capteur CCD de 140 mega pixels de Dalsa. - Source : [HDR]

L'augmentation de la résolution pour des capteurs de taille constante pose une problématique : le pouvoir séparateur des lentilles, également appelé piqué en photographie. Certains constructeurs comme Canon étudient de nouveaux traitements de surfaces et de verres pour pouvoir répondre à un meilleur pouvoir séparateur.

Les objectifs

Les objectifs sont des systèmes optiques convergents formés de plusieurs lentilles permettant de projeter une partie d'une scène 3D sur un plan. En fonction de la combinaison des lentilles, la définition, le contraste, la transmission lumineuse, la distorsion et l'angle de vue seront différents. Les objectifs peuvent être également couplés à des miroirs. On distingue ainsi deux familles d'objectifs avec ou sans miroirs :

- les objectifs perspectifs ou classiques.
- les objectifs omnidirectionnels comme les lentilles fisheye, les miroirs hyperboloïdes, paraboloïdes, sphériques ou encore coniques (fig. 1.12).

(a) (b) (c)

FIGURE 1.11 – Différents capteurs. (a) Vue catadioptrique. (b) Vue fisheye. (c) Vue perspective. Sources : [Fis , Omn]

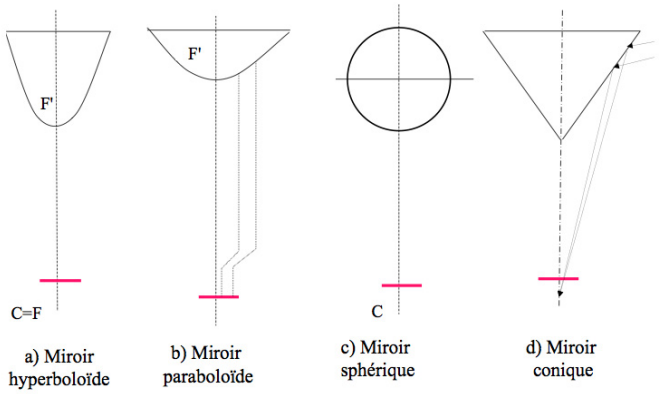


FIGURE 1.12 – Formes géométrique des mirroirs. - Source : [Mouaddib 2005]

La première catégorie est la plus répandue. Les objectifs perspectifs de 24mm de focale sont équivalents à la perception humaine. Ils présentent peu de distorsions, une résolution importante et constante.

La deuxième catégorie, les objectifs omnidirectionnels (fig. 1.11(a), fig. 1.12), sont présentés dans les travaux de [Mouaddib 2005] où leur modélisation, intérêts et inconvénients sont détaillés. Ces capteurs présentent l'avantage d'avoir un champ de vue large. Néanmoins ils possèdent une résolution non constante et une géométrie particulière. Les avantages des capteurs fisheye comparés aux caméras catadioptriques sont la réduction de vibrations et la suppression du trou noir au centre de l'image dues à l'absence de tige. Néanmoins, les miroirs catadioptriques possèdent un bien meilleur champ de vue : 180° avec une perception de 360° autour de l'axe optique contre 180° en diagonale pour un fisheye.

Les temps de traitement et d'acquisition des images ont longtemps été un argument en défaveur de la vision pour des applications temps réel. Cependant, l'amélioration de la puissance de calcul au fil du temps favorise aujourd'hui l'utilisation de ces capteurs pour des applications temps réel. Ils sont en effet capables d'estimer rapidement, précisément les paramètres de vol et peuvent être utilisés pour d'autres tâches. De plus les caméras sont compactes, consomment peu d'énergie, fournissent de fortes cadences (jusqu'à 200 images par seconde) et sont capables de recueillir un nombre important d'informations. Elles présentent également l'avantage d'avoir une mesure absolue et de ne pas être sensibles (une fois blindées) aux perturbations électromagnétiques générées par les propulseurs de l'aéronef et les batteries.

1.6.2 Estimation des paramètres d'état

La complexité des capteurs de vision leur permet de collecter une information riche de l'environnement. Des méthodes de traitement d'images exploitent ces informations pour estimer l'état des drones. Les travaux cités dans cette partie proposent de déterminer soit un paramètre d'état comme l'altitude, l'attitude ou le mouvement, soit plusieurs paramètres d'état simultanément.

L'altitude

La vision, mono ou stereo, peut être utilisée afin d'estimer l'altitude.
Les travaux de [Garcia-pardo 2000, Saripalli 2002, Sharp 2001] proposent de balayer le sol avec une caméra perspective (fig. 1.13) et de déterminer l'altitude à partir d'une mire fixée sur le sol. Ce type d'approche est intéressante puisqu'elle nécessite l'utilisation d'une seule caméra, fournit une pose complète et peut être utilisée en temps réel. Elle est cependant limitée à un environnement spécifique, muni de marqueurs artificiels au sol.
D'autres systèmes basés sur le flot optique emploient une caméra monoculaire perspective comme [Barrows 2001, Beyeler 2006, Chahl 2004, Green 2003]. Ces systèmes, inspirés des abeilles, permettent de déduire l'altitude, à partir du flot

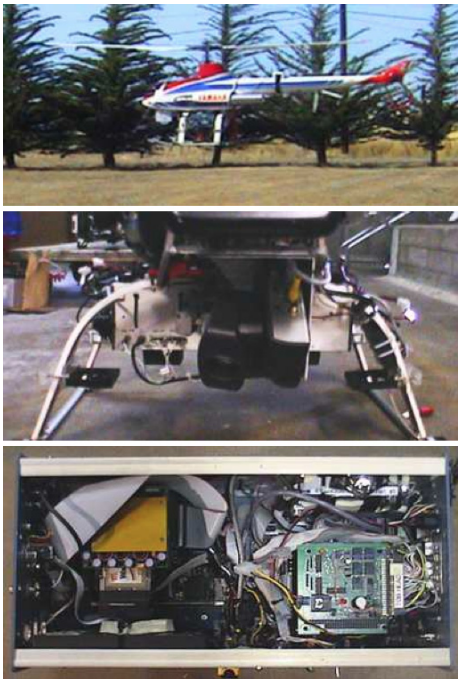


FIGURE 1.13 – Système embarqué pour l'estimation d'altitude. - Source : [Sharp 2001]

optique, en connaissant la vitesse de déplacement de la caméra.
[Beyeler 2006] propose d'estimer le tangage pour corriger le flot optique. Néanmoins ce travail ne propose pas de déterminer l'information du roulis. En l'absence de cette information, l'altitude estimée peut être biaisée, ce qui pose certaines limites de la méthode.
[Zufferey 2010] propose de simplifier le principe du flot optique appliqué aux caméras en utilisant plusieurs capteurs de souris d'ordinateur (fig. 1.16(b)). Ce dispositif évalue le déplacement et régule l'altitude en utilisant une faible puissance de calcul. Ainsi, ce système présente l'avantage d'être compact et rapide en traitement mais la résolution de l'environnement perçu est limitée au nombre de capteurs employés.
Dans les travaux de [Cheriany 2009], les auteurs utilisent une technique d'apprentissage. Le principe consiste à associer les informations contenues dans la texture d'une image à une valeur d'altitude possible.
D'autres travaux utilisent plus d'une caméra.
[Jung 2003, Meingast 2005] proposent d'utiliser des capteurs stéréoscopiques plutôt qu'une caméra unique. Dans ce travail, les auteurs développent une approche basée sur la mise en correspondance de points d'intérêt afin d'extraire la carte d'élévation du sol.
[Sanfourche 2003] proposent un algorithme multi-vues dans le but de calculer une

carte numérique du sol. Cet algorithme utilise une séquence d'images issue d'une seule caméra.

Ainsi, certains travaux proposent d'utiliser une seule caméra pour donner une appréciation de la proximité du sol. En revanche, d'autres méthodes proposent d'ajouter une seconde caméra pour retrouver l'information 3D par la mise en correspondance de primitives entre les deux vues. Cependant, ces travaux ne sont pas adaptés à une application en temps réel et embarquée.
Pour résumer cette partie, le tableau 1.2 récapitule les principales techniques d'estimation d'altitude par vision.

Méthode	Nombre de caméras	Principe
[Garcia-pardo 2000] [Saripalli 2002] [Sharp 2001]	1	Marqueur au sol
[Barrows 2001] [Beyeler 2006] [Chahl 2004] [Green 2003]	1	Flot optique (inspiré des abeilles)
[Beyeler 2006]	1	Flot optique
[Jung 2003] [Meingast 2005]	2	Mise en correspondance
[Cheriany 2009]	1	Apprentissage
[Sanfourche 2003]	1	Stereovision à partir du mouvement. Construction d'une carte numérique du sol
[Zufferey 2010]	>2	Flot optique basé sur les capteurs de souris

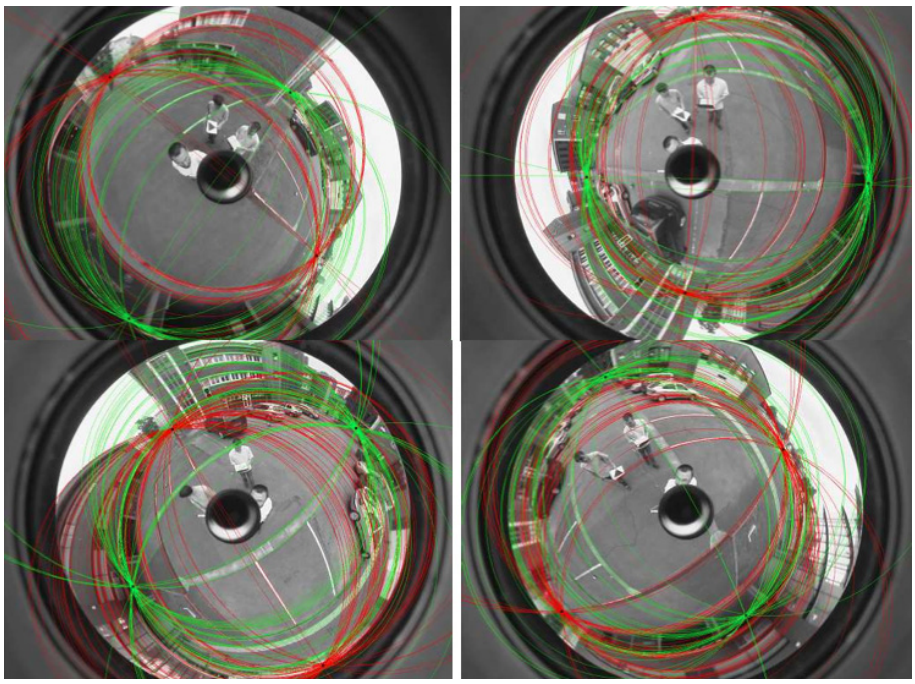
TABLE 1.2 – Comparaison des méthodes basées sur la vision pour l'estimation de l'altitude.

L'attitude

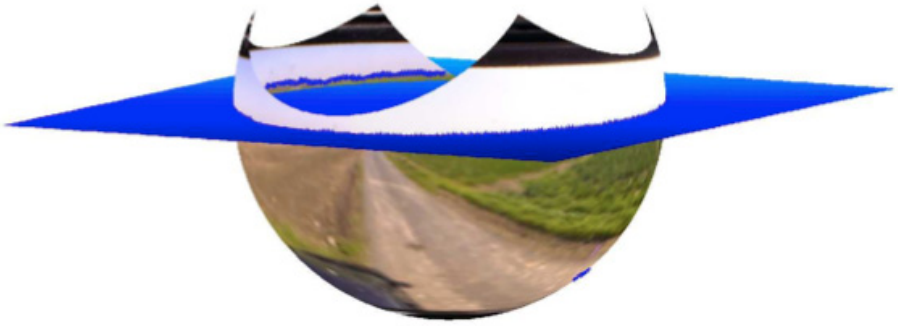
L'attitude peut être déterminée par différentes approches basées sur la vision. Deux cas se présentent.

Dans le premier cas, [Bazin 2008, Demonceaux 2007] présentent des méthodes s'appuyant sur les points de fuite en environnement urbain (fig. 1.14(a)). Ces points de fuite sont déterminés à partir de l'intersection des lignes horizontales et verticales massivement présentes dans ce type d'environnement. Ils apportent à

la fois une information sur l'attitude (assiette) et l'orientation (cap) à partir d'une caméra omnidirectionnelle. Néanmoins, cette méthode est difficilement applicable en milieu naturel.



(a)



(b)

FIGURE 1.14 – (a) Estimation de l'attitude à partir de caméra catadioptrique et des points de fuite. (b) Estimation d'attitude à partir de caméra catadioptrique et de la ligne d'horizon. - Sources : [Bazin 2008, Demonceaux 2006b]

Dans le deuxième cas, [Dusha 2007] propose une méthode basée sur la ligne d'horizon pour estimer l'attitude à partir d'une caméra perspective et son suivi par flot optique.

[Demonceaux 2006b] propose également une méthode basée sur la ligne d'horizon pour estimer l'attitude à partir d'une caméra omnidirectionnelle (fig. 1.14(b)). Ces méthodes présentent un maximum de 3° d'erreur en comparaison à des centrales inertielles.
[Thurrowgood 2009] propose d'estimer l'attitude à partir de tout type de caméra. Cette méthode se base sur l'extraction ciel/terre puis permet de trouver le plan formé par l'intersection de la ligne d'horizon et de la sphère de projection. Cependant, elle ne propose pas d'estimer l'orientation. Les avantages de ces travaux sont la possibilité d'adaptation à tout type de caméra (omni ou perspective) et l'implémentation temps réel.

L'estimation d'attitude par vision nécessite de considérer deux cas. Le cas urbain, où les lignes, largement présentes dans ce type d'environnement, permettent de déterminer les points de fuite et donc l'attitude. Dans l'environnement naturel, ces lignes sont assez rares. L'exploitation de la ligne d'horizon permet de déterminer l'attitude dans ce deuxième cas. Enfin, le tableau 1.3 synthétise les principales méthodes d'estimation d'attitude par vision.

Méthode	Type de vue	Principe	Environnement
[Demonceaux 2007] [Bazin 2008]	Catadioptrique	Points de fuite	Urbain, intérieur
[Dusha 2007]	Perspective	Ligne d'horizon	Naturel
[Demonceaux 2006b]	Catadioptrique	Ligne d'horizon - partition par Mahalanobis	Naturel
[Thurrowgood 2009]	Toutes	Ligne d'horizon - méthode basée histogramme	Naturel

TABLE 1.3 – Comparaison des méthodes basées sur la vision pour l'estimation de l'attitude.

Le mouvement

Les approches d'estimation de mouvement utilisent des primitives communes entre deux prises de vues successives pour quantifier un déplacement. Dans les plupart des travaux, ces primitives sont caractérisées par des points qui peuvent être soit suivis par flot optique [Singh 2005, Sanahuja 2010b, Garcia-Carrillo 2011] ou

soit mis en correspondance par des méthodes de SIFT [4] proposée par [Lowe 1999] par exemple. Le déplacement de ces points donne une information du mouvement à partir de la décomposition des matrices fondamentale, essentielle ou d'homographie [Hartley 2004, Ma 2003] ou à partir du découplage entre la rotation et la translation [Bazin 2010, Daniilidis 1993].

Nous avons distingué précédemment deux principales familles de caméras : les caméras perspectives et les caméras omnidirectionnelles. A partir de ces deux catégories, trois cas de figure sont rencontrés dans les travaux pour estimer le mouvement.

(a)

(b)

FIGURE 1.15 – (a,b) SfM à partir de caméra catadioptrique. - Sources : [Geyer 2001, Lhuillier 2008]

D'abord l'utilisation de caméras perspectives permet d'obtenir une résolution dense et constante. Par exemple, les travaux de [Kanatani 1985, Daniilidis 1993,

4. *Scale-invariant feature transform*, Transformation de caractéristiques visuelles invariante à l'échelle.

Shakernia 1999] utilisent ce type de caméras pour estimer le mouvement à partir de points localisés sur le sol. Cependant, le champ de vue restreint peut mener à un manque de primitives et causer des ambiguïtés de rotation/translation. De plus, cette faible couverture du champ de vue peut engendrer la perte des primitives suivies lors d'un déplacement.

Ensuite, l'emploi de caméras omnidirectionnelles donne accès à un champ de vue plus large. Elles peuvent ainsi recueillir un nombre plus important d'informations dans l'image mais sont moins précises à cause de la résolution limitée et non constante [Lee 2000, Lhuillier 2008] (fig. 1.15(b)).
[Geyer 2001] présente une méthode de SfM [5] à partir de points mis en correspondance dans des vues non calibrées (fig. 1.15(a)).
[Makadia 2007] propose une méthode sans mise en correspondance par une transformée de Radon. Cette méthode est robuste aux mauvaises corrélations de primitives mais nécessite un temps de calcul important.
[Bazin 2010] suggère une approche d'estimation de mouvement en découplant la rotation et la translation. L'orientation relative de deux caméras paracatadioptriques est déterminée à partir de points de fuite calculés à partir de lignes parallèles tandis que le mouvement est retrouvé à partir de la rotation connue et d'un minimum de deux points mis en correspondance. Ainsi, cette technique nécessite plusieurs types de primitives telles que des points et des lignes. Cependant, elle n'est pas adaptée à un environnement naturel.

Enfin certains travaux traitent de systèmes de vision hybrides. Le principe consiste à employer au moins deux caméras de différents types comme une vue perspective et une vue omnidirectionnelle. Par exemple, [Ly 2011] propose d'estimer le mouvement à partir de lignes dans les vues hybrides. Cette méthode générique peut s'appliquer à tout type de caméras modélisées par une sphère. [Bastanlar 2010] propose également une méthode de SfM basée sur des vues hybrides constituées d'une vue fisheye et perspective modélisées par deux sphères. Dans ce travail, les auteurs proposent de réaliser la mise en correspondance des points entre les deux vues à partir d'un SIFT adapté. Cependant, le temps de pré-traitement du SIFT proposé est beaucoup plus important qu'un SIFT classique. Par conséquent, cette méthode n'est pas adaptée à des applications compactes et embarquées.

Pour conclure, les travaux d'estimation de mouvement présentés dans cette section sont classés selon le type de caméra employé, c'est à dire perspective, omnidirectionnelle ou encore en fusionnant les caractéristiques de ces deux catégories de caméras en système hybride [Bastanlar 2010, Ly 2011]. L'ensemble de ces travaux proposent de déterminer le mouvement complet ou de le découpler en rota-

5. *Structure from Motion*, Structure à partir du mouvement.

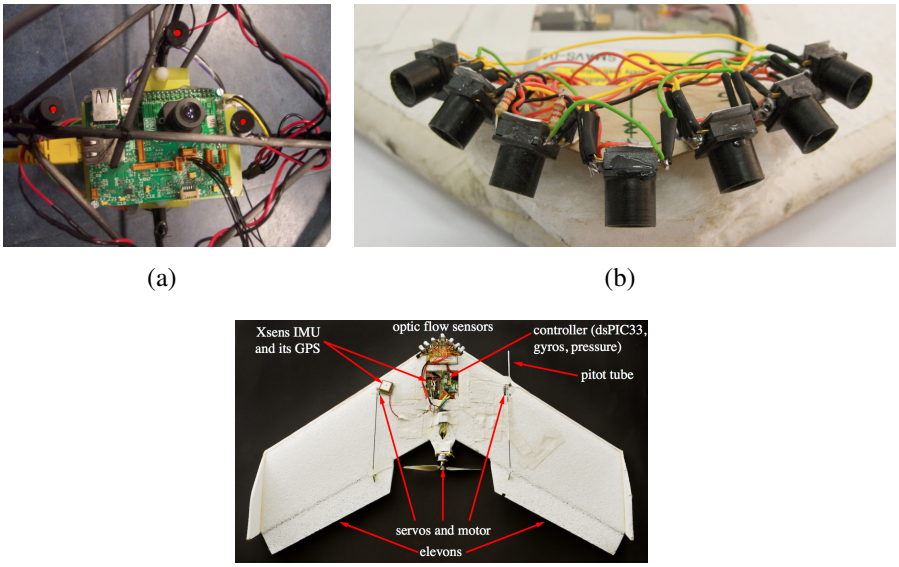
tion et en translation afin de répartir les estimations de paramètres entre différents capteurs [Daniilidis 1993, Bazin 2010]. Cependant dans les travaux présentés, peu proposent une estimation à l'échelle métrique. Les principaux travaux d'estimation de mouvement sont résumés dans le tableau 1.4 en fonction du type de caméra et des méthodes proposées.

Méthode	Caméra / Modèle	Principe
[Kanatani 1985]	Perspective / Planaire	Mouvement estimé à partir de lignes et de surfaces planaires. Pas de mise en correspondance.
[Shakernia 1999]	Perspective / Planaire	Estimation du mouvement à partir d'une surface planaire et d'une cible au sol (piste d'atterrissage).
[Lhuillier 2008]	Catadioptrique / central et non central	SfM et ajustement de faisceaux
[Singh 2005] [Sanahuja 2010b]	Perspective / Planaire	Flot optique
[Bazin 2010]	Catadioptrique / Sphérique	Découplage rotation et translation, suivi de points
[Geyer 2001]	Catadioptrique / Paraboloïde	SfM mise en correspondance de points à partir de vues non calibrées
[Makadia 2007]	Catadioptrique / Sphérique	SfM par transformée de Radon. Pas de mise en correspondance
[Garcia-Carrillo 2011]	Stéréo Perspective / Planaire	Flot optique
[Ly 2011]	Hybride / Sphérique	Dé-rotation, estimation de translation par fusion de points et lignes
[Bastanlar 2010]	Hybride / Sphérique	SfM à partir de SIFT

TABLE 1.4 – Comparaison des méthodes basées sur la vision pour l'estimation du mouvement.

Approches multi-paramètres

Les travaux cités précédemment proposent majoritairement d'estimer un paramètre de navigation. D'autres méthodes permettent, combinées ou non, de déterminer plusieurs paramètres de vol simultanément.



(a) (b)

(c)

FIGURE 1.16 – (a) Système composé d'une caméra et de trois pointeurs laser. (b) Disposition des capteurs de souris d'ordinateur. (c) Drone utilisé pour tester ces capteurs. - Sources : [Beyeler 2009, Sanahuja 2010b]

Les travaux de [Sanahuja 2010b] portent sur les étapes de modélisation, commande et capteurs (fig. 1.5). Ces travaux proposent d'utiliser comme capteur la vision pour l'asservissement visuel et d'estimer le déplacement 2D par flot optique. Il propose ensuite d'utiliser un capteur de stéréovision pour un asservissement 3D. Dans un contexte de traitement temps réel et embarqué, ces travaux montrent la difficulté des méthodes basées sur l'appariement de points entre deux vues. Une hypothèse de traitement déporté est ensuite proposée afin d'augmenter la puissance de calcul et de diminuer l'encombrement du matériel embarqué. Néanmoins, avec ce type de système d'autres problématiques se posent : la synchronisation, la latence et les perturbations de transmission. Pour palier ces inconvénients, un système mono caméra muni de trois pointeurs laser est développé (fig. 1.16(a)). Ce système, une fois calibré, permet d'estimer les angles de roulis, tangage et l'altitude du drone. Cependant, ces travaux se limitent à l'estimation de l'attitude et l'altitude, le mouvement 3D n'étant pas déterminé. De plus, les lasers présentent l'inconvénient d'être sujet à un phénomène de tavelure (*speckle* - fig. 1.17) [Goudail 2001], dû aux interférences générées par la réflexion de la lumière cohérente sur une sur-

face présentant des irrégularités. La tavelure génère un bruit granuleux gênant pour le traitement d'images.
[Moore 2010] proposent un capteur de stéréovision constitué de deux caméras fi-

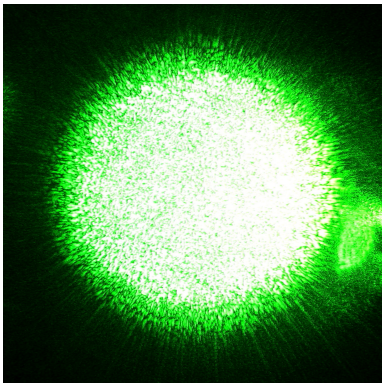


FIGURE 1.17 – Phénomène de tavelure. - Source : [Spe]

sheye placées l'une devant l'autre, pointées vers l'avant de manière à observer la projection du monde sur un cylindre. Ce système est capable de déterminer à la fois l'altitude, l'attitude, permet l'évitement d'obstacles et l'atterrissage d'un drone. Néanmoins, la limite de ces travaux réside dans l'utilisation de deux capteurs identiques fournissant des images fortement déformées. Les traitements traditionnels et l'interprétation humaine sont alors beaucoup plus difficiles.
[Rondon 2010] proposent d'estimer l'altitude, la position et la vitesse par vision, avec l'estimation des paramètres intégrée dans la boucle de commande. Toutefois, la limite de cette méthode est liée à la dépendance à des marqueurs au sol. L'évolution du drone est ainsi limitée à la position des marqueurs.
Enfin, [Beyeler 2009] proposent d'utiliser des capteurs de flot optique (fig. 1.16(b)) en coopération avec une centrale inertielle et un GPS pour les phases de décollage et atterrissage d'un drone (fig. 1.16(c)). L'étude du flot optique apporte une information sur la proximité des obstacles, sur l'altitude et l'attitude. Les avantages et inconvénients restent identiques à ceux des travaux de [Zufferey 2010].

1.7 Conclusion

Ce chapitre a abordé la problématique de la navigation autonome des drones, domaine largement étudié à travers différents projets internationaux et nationaux. A travers cette notion d'autonomie, on distingue la capacité d'un drone à percevoir les paramètres d'état, liés à sa dynamique et indispensables à sa stabilisation. Les paramètres d'état peuvent être estimés à l'aide de plusieurs capteurs basiques, capables d'estimer un seul état, ou à l'aide de capteurs plus complexes, capables de

percevoir plusieurs états simultanément. Cependant la multiplication du nombre de capteurs mis en œuvre augmente l'encombrement, la consommation énergétique et le risque de panne. C'est pourquoi les capteurs de vision et les méthodes associées constituent le choix de nos axes de recherche.
En effet, nous avons vu que les approches basées sur la vision peuvent estimer l'ensemble des paramètres de navigation tels que l'altitude [Eynard 2010], l'attitude [Bazin 2008, Demonceaux 2006b, Demonceaux 2006a, Demonceaux 2007], les vitesses, la proximité des obstacles en temps réel ou l'ensemble de ces paramètres simultanément [Mondragón 2010]. Les images issues des caméras peuvent également être réutilisées pour d'autres tâches comme des missions d'observation. Dans les approches présentées dans ce chapitre, deux types de caméras sont utilisées : les caméras perspectives et omnidirectionnelles. Lors de la navigation d'un drone, l'environnement à observer est assez large, il est important d'obtenir une information globale grâce à un capteur omnidirectionnel et précis grâce à une caméra perspective. Cette configuration est appelée système de stéréovision hybride. Certains travaux de la littérature propose certains de ces systèmes mais actuellement, aucune des méthodes n'est en temps réel.

Enfin, dans cette thèse, un capteur hybride est proposé. D'une part une vue fisheye, modélisée par une sphère, fournit l'orientation et l'attitude. D'autre part, la vue perspective, modélisée par un plan fournit une image précise de l'environnement. De ce fait, cette combinaison permet d'exploiter les avantages des deux vues. Ainsi, à partir de méthodes de traitement d'images sans mise en correspondance, il semble possible de répondre aux contraintes de temps réel et de consommation énergétique tout en déterminant l'ensemble des paramètres de navigation.

CHAPITRE 2

Architecture du capteur hybride et du matériel

Sommaire

2.1 Capteurs et modèles 27
2.1.1 Choix des capteurs 28
2.1.2 Choix géométrique 30
2.2 Modélisation du capteur 31
2.2.1 Projection perspective 31
2.2.2 Projection sphérique 31
2.3 Étalonnage de caméras 32
2.3.1 Étalonnage stéréo hybride 33
2.3.2 Implémentation logicielle 34
2.4 Matériel 35
2.4.1 Les capteurs 35
2.4.2 Processeurs de traitement 41
2.4.3 Interfaces de communication 45
2.4.4 Les quadri-rotors 46
2.4.5 Solution retenue 47
2.5 Conclusion 49

Ce chapitre propose dans une première partie de justifier le choix des capteurs et du modèle. Une deuxième partie, détaille ensuite la modélisation, puis une troisième partie présente la méthode d'étalonnage de ce nouveau système. Enfin, dans une dernière partie, le matériel utilisé est présenté.

2.1 Capteurs et modèles

Le chapitre 1 pose la problématique de l'autonomie des drones compacts. L'autonomie peut être améliorée à condition de connaitre les états du drone durant le vol. Ces états peuvent être déterminés à partir des capteurs de vision car ils sont capables de recueillir un grand nombre d'informations et d'estimer l'ensemble des

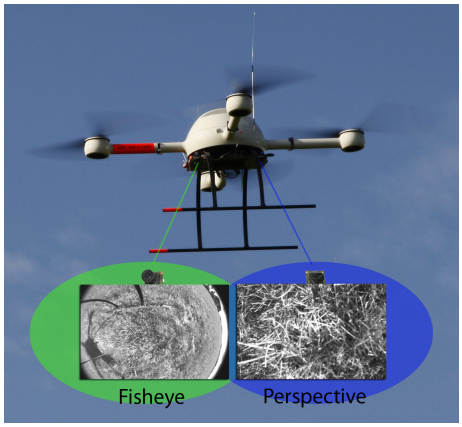
paramètres de navigation. Cependant l'ensemble des méthodes d'estimation des paramètres sont basées soit sur des caméras perspectives soit omnidirectionnelles et peu proposent des algorithmes en temps réels.

2.1.1 Choix des capteurs

Tout d'abord, l'utilisation de caméras omnidirectionnelles est adaptée à l'estimation de l'orientation et à l'observation globale d'un environnement. L'intérêt des caméras à large champ de vue a déjà été démontré dans différents travaux comme [Hrabar 2003] pour la navigation ou [Demonceaux 2007] pour l'estimation d'attitude.
Une telle image peut être obtenue à partir d'une combinaison de miroirs et de lentilles formant la catégorie des capteurs catadioptriques. Ils présentent une zone noire au centre de l'image, due à la présence d'une tige de liaison entre l'objectif et le miroir catadioptrique. Le champ de vue obtenu est alors supérieur à 180° avec une perception de 360° autour de l'axe optique. D'autres capteurs hypergones composés uniquement de lentilles permettent de couvrir un champ de vue de 180° dans toutes les directions. Ces capteurs sont les lentilles fisheye.
Les avantages de ces objectifs comparés à des miroirs catadioptriques est l'absence de zone noire au centre de l'image et une sensibilité réduite face aux vibrations grâce à l'absence de tige et de miroir. Ces vibrations rencontrées lors d'un vol sont également compensées par la courte focale. De plus, la lentille fisheye est la plus compacte de la catégorie objectifs/miroirs omnidirectionnels. En revanche, l'inconvénient face à un miroir catadioptrique est l'angle de vue un peu plus réduit et face à un objectif perspectif, la vue fisheye possède une résolution plus faible, non constante et présente de fortes distorsions.

Ensuite, les caméras perspectives sont employées pour des applications classiques comme l'estimation du mouvement ou l'observation d'une zone précise d'une scène comme le suivi de cible. L'avantage d'un objectif perspectif est la formation d'une image de résolution importante, uniforme et faiblement distordue. Néanmoins, l'inconvénient majeur est un champ de vue limité. D'un point de vue mécanique, la caméra perspective possède les mêmes qualités que la caméra fisheye. La sensibilité aux vibrations sera liée à la focale choisie : plus la focale est longue, plus le flou lié aux tremblements sera amplifié dans l'image.

Ainsi, l'élaboration d'un système hybride permet de combiner les caractéristiques de ces deux types de capteurs. Nous proposons d'utiliser une caméra fisheye pour la vue omnidirectionnelle en raison de ses propriétés mécaniques et une caméra perspective standard (fig. 2.1(a)). L'utilisation d'un tel système permet d'assurer un recouvrement total de la perspective. Il est possible d'obtenir une grande masse d'informations communes entre les deux images ce qui est plus difficile avec



(a)

FIGURE 2.1 – (a) Drone MD4-1000 équipé du capteur de stéréovision hybride.

deux vues perspectives où une seule petite zone est partagée.

Pour illustrer ce choix, certains travaux traitent des capteurs hybrides. Dans les travaux de [Sturm 2002], l'auteur formule les matrices fondamentales pour deux vues hybrides composées d'une caméra perspective, affine ou para-catadioptrique. [Puig 2008] utilise la mise en correspondance entre deux vues hybrides en utilisant la méthode SIFT. Les auteurs proposent ici d'appliquer une transformation polaire à la vue omnidirectionnelle pour la mise en correspondance des deux vues par la méthode SIFT. Cette méthode fonctionne sur des vues non calibrées. Toutefois, l'inconvénient est la difficulté de l'implémenter en temps réel sur un système embarqué.
[Bastanlar 2010] suggère une configuration semblable à celle proposée dans cette thèse : une caméra fisheye et une perspective. Les caméras sont modélisées par deux sphères et une méthode de mise en correspondance par SIFT est proposée pour estimer le mouvement (cf. section 1.6.2). Cependant, cette méthode présente le même inconvénient que la précédente.

Enfin, l'étendue des possibilités offertes par un capteur hybride laisse envisager de nouvelles applications. En effet, son utilisation permet de déterminer à la fois l'attitude, l'altitude, le plan du sol et le mouvement par différentes méthodes et les images acquises par ce capteur peuvent être utilisées pour d'autres tâches. Ce capteur présente également l'avantage de répondre aux contraintes de poids ($2 \times 12\,\mathrm{g}$), de vibration, d'encombrement (H : $36\,\mathrm{mm}$, W : $36\,\mathrm{mm}$, L : $20\,\mathrm{mm}$) et de faible consommation (entre $0.4\,\mathrm{W}$ et $1\,\mathrm{W}$ par caméra) requis par des drones compacts. Cependant, la difficulté réside dans l'application des méthodes classiques de géométrie et traitement d'images en stéréovision. L'adaptation et le développement de nouvelles méthodes constituent un défi scientifique et informatique afin d'obtenir

un capteur capable d'estimer en temps réel, de façon robuste et précise les paramètres de navigation d'un drone.

2.1.2 Choix géométrique

Les caméras centrales sont des caméras à point de vue unique (perspectives, catadioptriques) et peuvent être représentées par le modèle sphérique de [Barreto 2001] à double centre de projection.
Une caméra perspective et fisheye composent le capteur hybride proposé. Elles peuvent donc être modélisées par :

- deux modèles sphériques,
- un modèle hybride : planaire (fig. 2.2(a)) et sphérique (fig. 2.2(b)).

[Bastanlar 2010, Ly 2011] présentent des travaux basés sur le premier cas de figure. D'un point de vue géométrique, cette modélisation présente l'avantage d'utiliser des modèles identiques mais ne permet pas l'emploi des méthodes traditionnelles adaptées à une géométrie planaire. De plus, lors de l'étalonnage d'une caméra perspective par le modèle sphérique, les deux centres de projection tendent à être confondus et ce modèle devient ainsi assimilable au modèle planaire. La représentation sphérique de ce type de caméra ne semble donc pas adaptée.
D'un point de vue algorithmique, cette représentation permet d'utiliser des méthodes génériques, adaptés à la sphère. Néanmoins, le passage sur la sphère requiert une interpolation coûteuse en temps de traitement comme par exemple la recherche du voisinage d'un point image. Les applications en temps réel ne semblent alors pas possibles. C'est pourquoi cette combinaison nécessite l'étude de nouvelles méthodes appropriées et plus rapides.

Le deuxième cas de figure propose un modèle hybride, planaire et sphérique. D'un point de vue géométrique, chaque modèle apporte une géométrie différente et permet l'emploi d'une variété de méthodes adaptées soit à la géométrie sphérique soit à la géométrie planaire. Néanmoins, ce capteur hybride rend plus complexe les traitements photométriques entre les deux vues comme pour les méthodes de mise en correspondance.
D'un point de vue algorithmique il est possible d'appliquer les méthodes traditionnelles adaptées aux caméras perspectives. Ces méthodes conservent l'avantage du voisinage discret du plan image, connu pour sa rapidité en temps de traitement. En parallèle, d'autres travaux adaptés à la sphère peuvent être utilisées sur la vue omnidirectionnelle.

Avant de présenter les contributions de ce travail, le modèle géométrique du système hybride et la méthode d'étalonnage sont détaillés dans les parties suivantes.

2.2 Modélisation du capteur

2.2.1 Projection perspective

La modélisation planaire des caméras perspectives est représentée par la figure 2.2(a). Soit $\mathbf{X} = \begin{pmatrix} X & Y & Z \end{pmatrix}^\mathsf{T}$ un point 3D exprimé dans le repère caméra. Ce point est projeté dans le plan image suivant $\mathbf{x} = \begin{pmatrix} x & y & 1 \end{pmatrix}^\mathsf{T}$:

$$\mathbf{x} = pr(\mathbf{X}) \quad \text{avec} \quad \begin{cases} x = \frac{X}{Z} \\ y = \frac{Y}{Z} \end{cases} . \tag{2.1}$$

$\mathbf{x}$ est le point sur le plan image normalisé. Le point pixellique correspondant, $\mathbf{u} = \begin{pmatrix} u & v & 1 \end{pmatrix}^\mathsf{T}$, est obtenu par la relation $\mathbf{u} = \mathbf{K}\mathbf{x}$.

$$\mathbf{K} = \begin{pmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2.2}$$

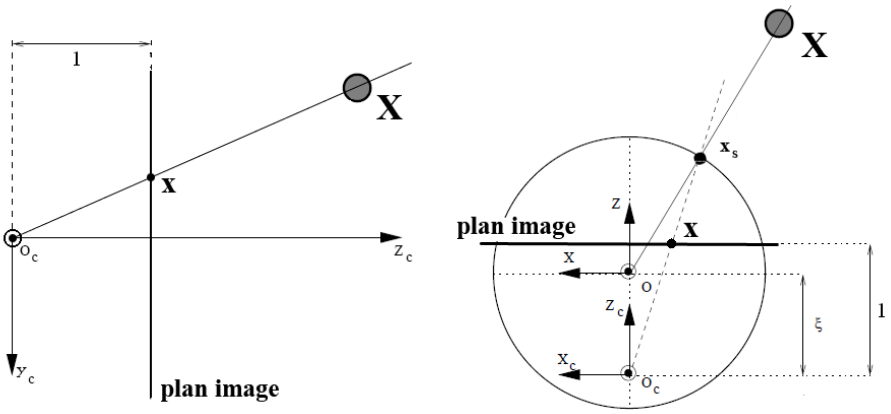
(α_u, α_v) représentent respectivement l'échelle de l'image dans les directions x et y. (u_0, v_0) représentent les coordonnées du point formé par l'intersection entre l'axe optique et le plan image. $\mathbf{K}$ est la matrice des paramètres intrinsèques, connaissant les paramètres $\gamma_1 = \{\alpha_u, \alpha_v, u_0, v_0\}$:

Ainsi, la projection perspective complète d'un point 3D vers le point image pixellique est donnée par l'eq.2.3.

$$pr_{\gamma_1}(\mathbf{X}) = \mathbf{K} pr(\mathbf{X}). \tag{2.3}$$

2.2.2 Projection sphérique

Les capteurs fisheye ne font pas partie des capteurs à point de vue unique [Baker 1999]. En effet, les rayons lumineux ne convergent pas en un point unique mais en une zone appelée diacaustique. [Ying 2004] montrent toutefois qu'il est possible d'utiliser la sphère unitaire pour ce type de capteur. Cette modélisation est possible si l'erreur de calibration est en adéquation avec l'application désirée. Sur cette proposition, [Mei 2007] présente une méthode de calibration basée sur la sphère unitaire. Cette méthode destinée aux capteurs à point de vue unique et aux fisheyes donne la possibilité de modéliser les distorsions radiales et tangentielles des lentilles. En suivant le modèle sphérique présenté dans la figure 2.2(b), un point 3D $\mathbf{X}$ est tout d'abord projeté sur une sphère unitaire, centrée en $\begin{pmatrix} 0 & 0 & 0 \end{pmatrix}^\mathsf{T}$. Le point obtenu, $\mathbf{x}_s$, est ensuite projeté de façon perspective sur le plan image suivant $\mathbf{x}$, à partir du second centre de projection placé en $\begin{pmatrix} 0 & 0 & -\xi \end{pmatrix}^\mathsf{T}$ avec $\rho = \sqrt{X^2 + Y^2 + Z^2}$. :



(a) (b)

FIGURE 2.2 – (a) Modèle perspectif. (b) Modèle sphérique de projection centrale unifiée. - Source : [Barreto 2001]

$$\mathbf{x}_s = \frac{\mathbf{X}}{\rho}, \quad \mathbf{x} = pr_\xi(\mathbf{X}) \quad \text{avec} \quad \begin{cases} x = \frac{X}{Z+\xi\rho} \\ y = \frac{Y}{Z+\xi\rho} \end{cases}, \tag{2.4}$$

Enfin un point image pixellique est obtenu à partir d'un point 3D connaissant les paramètres intrinsèques $\gamma_2 = \{\alpha_u, \alpha_v, u_0, v_0, \xi\}$ et en appliquant $pr_{\gamma_2}(\mathbf{X}) = \mathbf{K}pr_\xi(\mathbf{X})$ (eq. 2.1).

Une fois le modèle défini, il est nécessaire de calibrer le banc stéréo en estimant les paramètres intrinsèques et extrinsèques.

2.3 Étalonnage de caméras

L'étalonnage de caméra est un processus indispensable pour connaitre la relation entre un point 3D et un point image et surtout le lien entre chaque pixel et une ligne de vue 3D. Lors de la calibration, les différentes poses et les paramètres intrinsèques d'une caméra par rapport à une mire d'étalonnage sont estimées simultanément.

Plusieurs travaux et outils d'étalonnage sont proposés. Pour une seule caméra, [Scaramuzza 2006, Mei 2007] et d'autres auteurs proposent des outils adaptés aux caméras centrales. Dans le cas d'une ou deux caméras, [Bouguet 2004] propose un outil pour les caméras perspectives.

Bien que ces méthodes proposent de calibrer différents types de caméras, peu de logiciels libres proposent de calibrer des systèmes stéréoscopiques de même type. Pour le calibrage hybride, il semble qu'aucun outil n'est proposé. C'est la

raison pour laquelle une collaboration a été réalisée avec un autre doctorant pour étendre les travaux de [Caron 2009, Caron 2010] à l'étalonnage hybride multi-caméras. Cette collaboration a débouché sur l'élaboration d'un logiciel libre, multi plateforme (Mac, Windows, Linux) capable de calibrer N caméras à N modèles [Caron 2011] à partir d'asservissement visuel virtuel (AVV ou VVS [1]).

L'AVV (Asservissement Visuel Virtuel), ou technique d'asservissement visuel basée image, vise à déplacer virtuellement une caméra à la position et l'orientation désirée en minimisant l'erreur entre les primitives de l'image courante et les primitives de l'image de référence. Cette technique a été étendue à l'étalonnage de caméras perspectives dans le travail de [Marchand 2001] ou encore stéréoscopiques avec [Caron 2010].
Sa formulation permet de l'étendre à l'étalonnage de bancs stéréo hybrides composés d'une multitude de caméras de modèles hétérogènes. Ainsi il est possible de combiner par exemple une caméra perspective avec une caméra fisheye modélisées respectivement par un plan et une sphère.

La contribution dans ce travail est l'extension des travaux d'étalonnage de caméras perspectives et omnidirectionnelles par VVS à des vues hybrides composées de n caméras et de n modèles présenté dans la partie suivante.

2.3.1 Étalonnage stéréo hybride

L'étalonnage stéréoscopique peut se réaliser de deux manières : estimer les paramètres intrinsèques des caméras puis estimer la pose de l'une par rapport à l'autre [Boutteau 2009] ou alors estimer tous les paramètres simultanément [Bouguet 2004].
Notre choix se tourne vers l'estimation des paramètres simultanéement, car il est possible d'obtenir les paramètres de calibration les mieux ajustés. Ainsi, en étendant l'étalonnage générique d'une caméra à plusieurs caméras, le principe consiste à définir une vue de référence et à exprimer les points de la mire d'étalonnage dans chacune des vues. Chacune des pose est ensuite exprimée en fonction de la caméra de référence. De manière générale, pour un banc de N cameras, l'objectif est de minimiser un point $\mathbf{u} = (pr_{\gamma_{m_j}}({}^{c_j}\mathbf{M}_{c_1}\,{}^{c_1}\mathbf{M}_o\,{}^{o}\mathbf{X})$ et sa projection détectée dans l'image, notée $\mathbf{u}^*$. Le critère d'optimisation devient alors :

$$\Delta = \sum_{j=1}^{N}\sum_{k=1}^{k_j}(pr_{\gamma_{m_j}}({}^{c_j}\mathbf{M}_{c_1}\,{}^{c_1}\mathbf{M}_o\,{}^{o}\mathbf{X}_k) - {}^{c_j}\mathbf{u}_k^*)^2. \tag{2.5}$$

avec :

– $pr_{\gamma_{m_j}}$ la projection liée au modèle utilisé (cf. parties 2.2.1 et 2.2.2),

1. *Virtual Visual Servoing*, Asservissement visuel virtuel.

– ${}^{c_1}\mathbf{M}_o$ la position de la caméra de référence par rapport au repère observé (la mire),
– ${}^{c_j}\mathbf{M}_{c_1}$ la position de la caméra courante par rapport à la caméra de référence,
– ${}^{o}\mathbf{X}_k$ le k-*ième* point 3D exprimé dans le repère objet,
– ${}^{c_j}\mathbf{u}_k^*$ la projection du point détecté dans l'image.

Cette formulation générique de l'étalonnage stéréoscopique hybride simultané permet l'étalonnage d'un couple capteur/objectif avec tout modèle de projection adapté. Cet étalonnage permet par exemple de modéliser notre capteur hybride composé d'une caméra fisheye et d'une caméra perspective respectivement par une sphère et un plan.

2.3.2 Implémentation logicielle

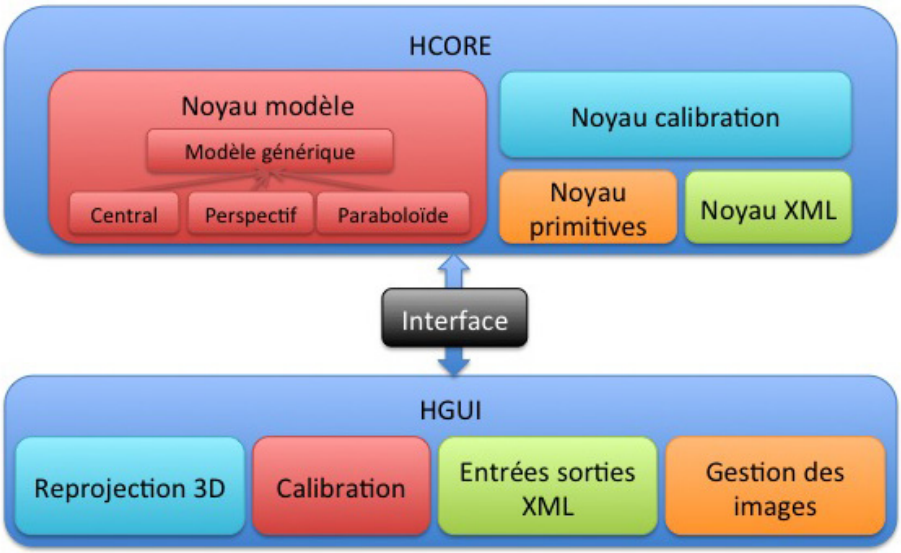


FIGURE 2.3 – Architecture du logiciel Hyscas.

Ce travail a ensuite donné lieu à la conception d'une architecture générique et modulaire en C++, à la réalisation de l'interface utilisateur et à une adaptation multi-plateforme. De plus, ce logiciel libre est mis à disposition de la communauté. Il utilise les librairies Qt, Visp et OpenCV et est indépendant de tout autre programme. L'architecture de ce logiciel est présentée à la figure 2.3. Cette contribution permet à la communauté scientifique d'étendre ses travaux sur la vision hybride.

Cette partie a justifié le choix du capteur hybride proposé, de sa modélisation

et propose une méthode d'étalonnage adaptée. La section suivante propose de détailler précisément les choix matériels pour l'acquisition et la partie traitements.

2.4 Matériel

Cette partie propose de faire la revue des principaux capteurs utilisés lors des expériences liées à ce travail de recherche.
Tout d'abord, l'étude de la centrale inertielle permet de mieux comprendre les limites d'utilisation de ce système. Une deuxième partie traite des capteurs de vision utilisés, puis dans une troisième partie, les différentes architectures mises en œuvre sont développées. Enfin, les plateformes volantes utilisées sont détaillées et permettent d'expérimenter les algorithmes proposés dans le chapitre 3 et les résultats associés sont présentés dans le chapitre 4.

2.4.1 Les capteurs

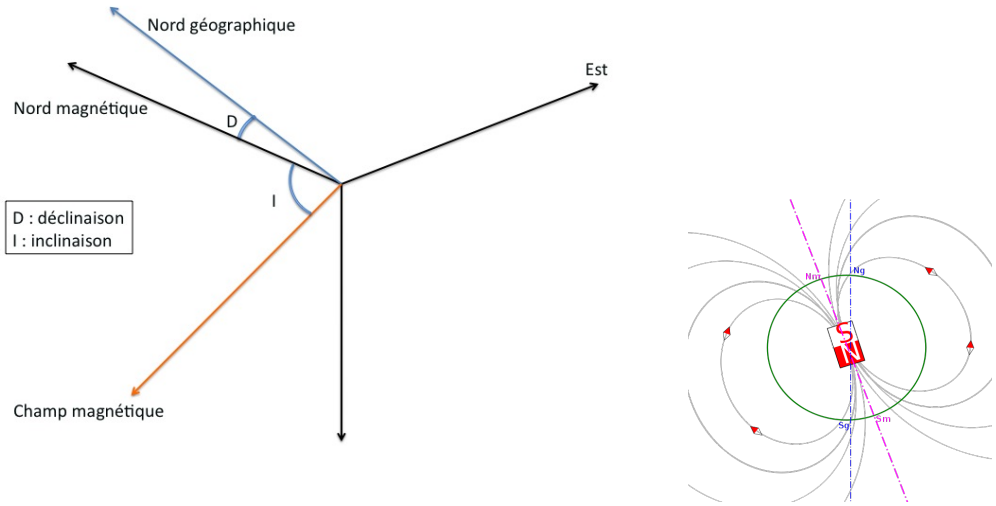
Centrale inertielle

Les centrales inertielles sont composées d'un magnétomètre, d'un gyroscope et gyromètre (capteur de position et vitesse angulaire), d'un accéléromètre et d'un capteur de température pour compenser le biais de mesure.

Magnétomètres

Tout d'abord, les magnétomètres ont pour rôle d'estimer l'orientation de la centrale par rapport au nord, orientation définie selon deux axes : l'axe objet-nord géographique et l'axe objet-nord magnétique (figure 2.4(b)). Pour cela, un vecteur dont le module est son intensité (figure 2.4(a)) représente le champ magnétique terrestre. Ce vecteur se décompose en deux composantes verticale et horizontale. D'une part, le méridien géographique et le méridien magnétique formant l'angle D, également appelé déclinaison. D'autre part, l'angle I formé entre le vecteur du champ magnétique et la composante horizontale est l'inclinaison. Le rôle d'un magnétomètre est donc d'estimer le nord magnétique à une position donnée dans l'espace.

Cependant, le champ magnétique varie au cours du temps. Ce phénomène est appelé déclinaison magnétique et dépend de la localisation géographique (1° par an en moyenne pour la zone 50.142146,1.828365). Afin de compenser les perturbations extérieures, il est nécessaire de calibrer le système. C'est pourquoi les magnétomètres sont des éléments sensibles, en particulier dans des applications de drone. En effet, les propulseurs électriques génèrent un champ magnétique important durant le fonctionnement. De plus, le matériel électronique embarqué est une source supplémentaire de perturbations électromagnétiques.



(a) (b)

FIGURE 2.4 – (a) Déclinaison du champ magnétique. (b) Champ magnétique terrestre et nord géographique. - Source : [Nor]

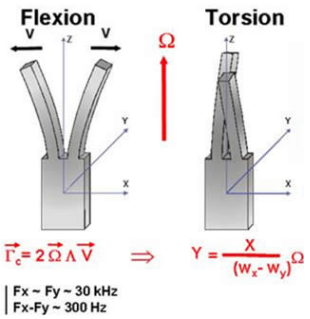


FIGURE 2.5 – Élément vibrant des gyroscopes. - Source : [One]

Technologiquement, les magnétomètres peuvent être formés d'un capteur à bobine inductive, à effet Hall, à variation de flux ou encore magnéto-résistifs. Dans les applications de drone, des capteurs magnéto-résistifs sont utilisés. Ils sont couplés à des gyroscopes et accéléromètres de type MEMS [2].

Gyroscopes et gyromètres

Ensuite, les gyroscopes déterminent la position angulaire et la vitesse angulaire de la centrale. Les types MEMS sont répandus du fait de leur faible poids et encombrement. On retrouve par exemple ce type de capteur dans les téléphones mobiles, disques dur ou encore dans les manettes de jeux. Il sont également utilisés dans les

2. *Micro Electro-Mecanichal Systems*, Microsystèmes électromécaniques.

drones. Leur fonctionnement se base sur la vibration d'une structure de quartz sensible aux rotations liées à la force de Coriolis (fig. 2.5). Ainsi, deux masses vibrent en opposition de phase. Lorsque le capteur tourne, l'effet de Coriolis crée une vibration entretenue par la propriété piezzo-électrique. Cette différence de potentiel génère un signal qui permet de connaitre la vitesse de rotation.

Ce capteur est caractérisé techniquement par la résolution, la température de fonctionnement, le nombre d'axes, le biais et l'échelle.

Accéléromètres

Puis, tout comme les gyroscopes, les accéléromètres sont le plus souvent de type MEMS. Leur fonction est la mesure de l'accélération d'un système. Le principe d'un accéléromètre repose sur la déformation ou le déplacement d'un corps pendant une accélération. Ils sont basés sur le principe fondamental de la dynamique (eq. 2.6) où $\vec{\mathbf{F}}$ est la somme des forces appliquées sur un objet. m correspond à sa masse inertielle et $\vec{\mathbf{a}_a}$ l'accélération de son centre d'inertie. L'accélération du centre d'inertie est composée de l'accélération de Coriolis $\vec{\mathbf{a}_c}$ (négligée), l'accélération gravitationnelle $\vec{\mathbf{a}_g}$ et l'accélération relative ou de mouvement $\vec{\mathbf{a}_r}$. L'accélération mesurée par la centrale est donnée par l'équation 2.7.

$$\sum \vec{\mathbf{F}_i} = m\vec{\mathbf{a}_a} = m(\vec{\mathbf{a}_g} + \vec{\mathbf{a}_r} + \vec{\mathbf{a}_c}) \tag{2.6}$$

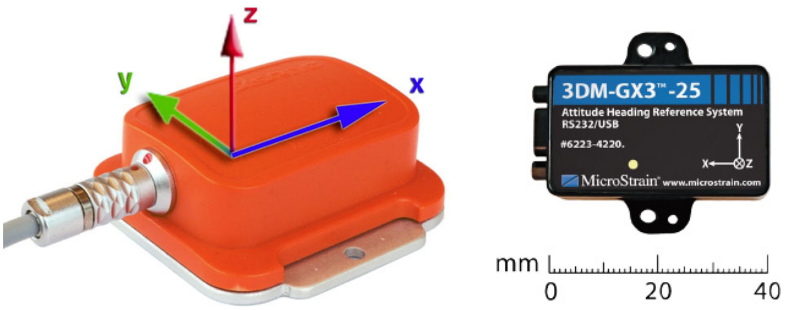
$$\sum \vec{\mathbf{F}}_{i \neq a_g} = m(\vec{\mathbf{a}_a} - \vec{\mathbf{a}_g}) \tag{2.7}$$

Leur fonctionnement, tout comme les gyroscopes, repose sur la vibration et la déformation d'une lame de quartz. Cette vibration et ces déformations produisent un effet piezzo-électrique : le changement de résonance est fonction de l'accélération.

Les accéléromètres sont caractérisés par le nombre d'axes, le biais, l'échelle, la température de fonctionnement, la précision et la sensibilité.

Centrales inertielles utilisées

Dans les travaux présentés, deux types de centrales inertielles sont employées : une centrale pour la validation des tests présentée sur la figure 2.6(a) et une centrale compacte embarquée présentée sur la figure 2.6(b).
La première centrale (fig. 2.6(a)) est capable de fournir l'attitude à une fréquence de 512Hz sans filtrage et 256Hz avec filtrage (filtrage de Kalman). L'erreur statique est inférieure à $0,5^\circ$ en roulis/tangage et inférieure à 1° en lacet. L'erreur dynamique est de 2° et la résolution angulaire de $0,05^\circ$. La consommation énergétique est de 0,35W.
La seconde centrale (fig. 2.6(b)) est capable de fournir l'attitude à une fréquence



(a) (b)

FIGURE 2.6 – (a) Centrale inertielle MTi de XSens. (b) Centrale compacte 3DM-GX3-25 de Microstrain. - Sources : [MTI , Microstrain]

comprise entre 1Hz et 1000Hz avec filtre numérique ajustable. Les erreurs statique et dynamique sont équivalentes à la centrale précédente et la résolution angulaire inféireure à $0,1^{\circ}$. Enfin, argument majeur pour une application embarquée, la consommation énergétique est de 80mA à 5V en USB (soit 0,4W).

Dans la suite, les informations d'orientation absolue et relative fournies par ces centrales seront exploitées.

Le capteur de stéréovision hybride

Utilisé comme système principal, complété ou appuyé par une centrale inertielle, le capteur de stéréovision hybride est employé pour estimer des paramètres d'état du drone. Les contraintes de vibration et de déplacement rapide d'un aéronef obligent une parfaite maîtrise de ces capteurs tant sur le plan électronique que logiciel. Cette maîtrise passe par la vitesse de prise de vue, mais aussi la synchronisation des capteurs. Pour répondre aux contraintes de vitesse, de synchronisation, de compacité et de faible consommation, deux caméras uEye d'IDS de modèle *UI-1226LE-C* sont utilisées. Ces caméras compactes (12g, H=36 mm, l=36 mm, L=20 mm), disposent d'une connectique USB [3] auto-alimentée, d'une synchronisation matérielle, d'une résolution 752x480 en couleur, et d'une cadence maximale de 87im/s. De plus, leur faible consommation, de 2 W maximum, permet une utilisation embarquée avec des aéronefs compacts. Leur monture M12 autorise des objectifs miniatures dont la norme répandue permet l'accès à un large choix de lentilles comme les fisheye. Les avantages sont leur facilité d'utilisation et la fourniture d'un SDK [4] complet.
Néanmoins, l'inconvénient des caméras de marque IDS est l'incompatibilité des

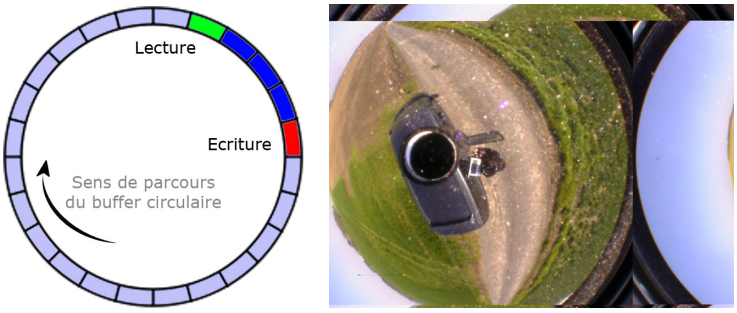
3. *Universal Serial Bus*, Bus de données série universel.
4. *Software Development Kit*, Trousse de développement logiciel.

pilotes récents avec des architectures non x86 (comme l'OMAP 3530 d'architecture ARM [5]) contrairement à d'autres caméras comme les Point Grey.

Les caméras sont capables de fortes cadences (environ $500 im/s$) mais il est nécessaire de les synchroniser et de gérer la mémoire de stockage des images. Ces aspects sont abordés dans la prochaine section.

Mémoire tampon circulaire

Il est fréquent d'associer à une caméra une seule zone mémoire tampon. Dans le cas d'un rafraichissement rapide de la lecture et/ou écriture il se produit un accès concurrent (fig. 2.7(b)). Pour pallier ce phénomène, une zone mémoire circulaire est créée (*ring buffer, ou circular buffer*) (fig.2.7(a)).



(a) (b)

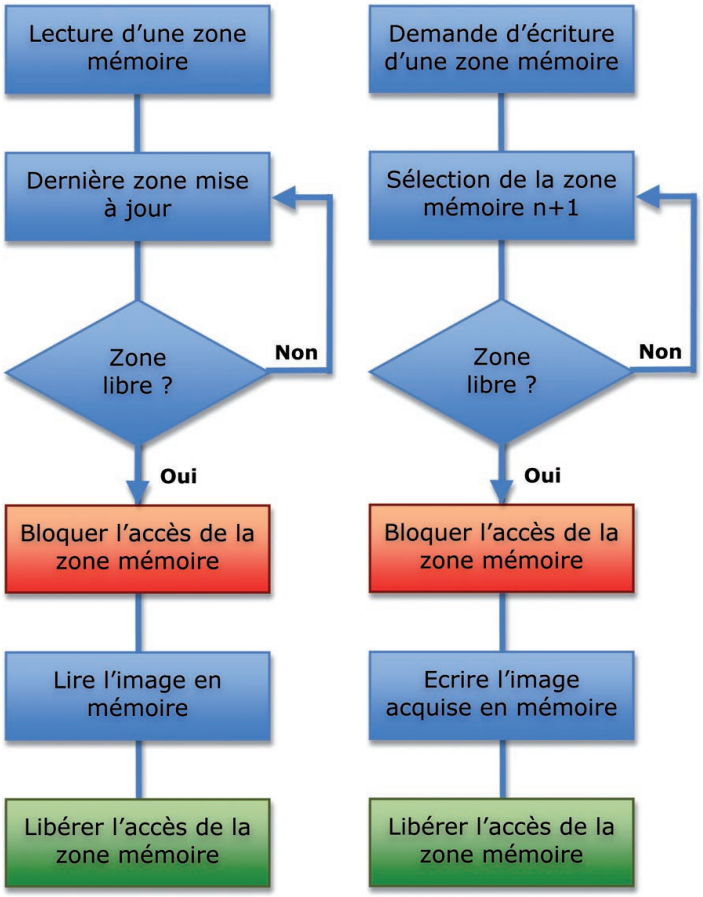
FIGURE 2.7 – (a) Lecture et écriture dans une mémoire tampon circulaire. (b) Accès concurrent en lecture/écriture d'une même zone mémoire.

Pour chacune des caméras, plusieurs zones mémoire sont attribuées. La figure 2.7(a) propose 24 images tampons. Chaque tampon peut se retrouver dans trois états distincts. Le premier état est la lecture, le second l'écriture et le troisième latent. Pour lire (fig.2.8(a)) ou écrire (fig.2.8(b)) dans un zone mémoire, sa disponibilité est vérifiée au préalable. En cas de disponibilité, la zone mémoire en cours d'accès est verrouillée afin d'y accéder en lecture ou écriture.

Synchronisation

Après l'étape d'acquisition des caméras, il se pose ensuite le problème de la synchronisation. Lorsque les caméras ne disposent pas de synchronisation matérielle, il est possible de les synchroniser par logiciel. Cette technique ne garantit pas une parfaite synchronisation des deux vues. Le logiciel donne l'instruction de débuter l'acquisition. Si l'une des caméras est au milieu de l'acquisition d'une image,

5. *Advanced Risc Machine*



(a) (b)

FIGURE 2.8 – (a) Algorithme de lecture dans une mémoire tampon circulaire. (b) Algorithme d'écriture dans une mémoire tampon circulaire.

l'interface logicielle attendra jusqu'à la prochaine trame complète et disponible. Par conséquent il peut y avoir un écart important entre la consigne et l'acquisition effective. En raison de son caractère non déterministe, la synchronisation logicielle n'est pas adaptée à des applications mobiles.

Au contraire, la synchronisation matérielle (fig. 2.9) est une méthode précise. Une caméra est définie maître, l'autre esclave. La première émet un signal électrique appelé *flash* lors de sa prise de vue. La seconde caméra attend ce signal en réception. Ce signal, appelé *trigger*, permet de déclencher l'acquisition au même moment. Ainsi les deux caméras observent le même instant d'une scène donnée. C'est pourquoi le choix se porte vers une synchronisation matérielle des deux systèmes de vision.

D'un point de vue matériel, les caméras doivent être cadencées à la même vitesse de prise de vue pour être synchronisées. De préférence, la caméra esclave

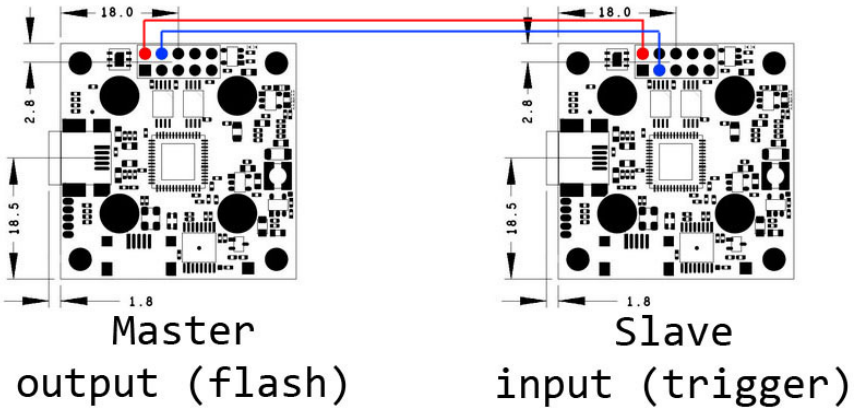


FIGURE 2.9 – Schéma des connections pour la synchronisation des caméras IDS μEye LE12xx. - Source : [IDS]

aura une cadence plus élevée (26 im/s) afin d'être à tout moment prête à recevoir un signal de la caméra maître (25 im/s).

2.4.2 Processeurs de traitement

Lorsque les informations issues des capteurs sont extraites, il est nécessaire de définir la partie matérielle capable d'exploiter et de traiter ces informations. Différentes pistes de traitements parallèles et séquentiels de l'information sont proposées.

Le GPGPU [6] est un type d'architecture massivement parallèle très en vogue pour certaines applications de traitement d'images. L'architecture des cartes ATI associée à la bibliothèque *Brook+* sera d'abord présentée. Ensuite, les processeurs séquentiels embarqués seront abordés. Ces processeurs beaucoup moins puissants ont l'avantage d'être compacts et faibles consommateurs d'énergie.

GPGPU

Contrairement aux CPU, dont la vocation est le traitement séquentiel d'informations complexes, le GPU [7] permet un traitement parallèle d'opérations simples SIMD [8]. L'exploitation des processeurs de cartes graphiques (GPU) s'effectue à partir du langage de programmation GLSL (basé sur OpenGL) et HLSL (basé sur DirectX). Ce sont les langages de programmation de *pipeline* (ou tuyau) les

6. *General-Purpose computation on Graphics Processing Units*, Calcul générique sur un processeur graphique.
7. *Graphics Processing Unit*, Processeur graphique.
8. *Single Instruction Multiple Data*, Une instruction est appliquée simultanément à plusieurs données.

plus bas niveau. Ils permettent d'utiliser la carte pour des applications graphiques mais également la détourner pour d'autres applications. Cette idée est la base du GPGPU : utiliser un processeur graphique pour des applications non graphiques. Chacun des principaux concepteurs de cartes graphiques a ensuite développé son propre langage : CUDA pour les cartes Nvidia et ATI Stream (basé sur Brook+) pour les cartes ATI. Un standard ouvert, OpenCL, a ensuite été mis en place en 2010. Ce langage a pour objectif d'unifier la programmation pour différentes architectures (x86, Cell, puces ATI, puces Nvidia...). Mac OS 10.6 Snow Leopard est l'un des premiers systèmes d'exploitation basé sur ce langage capable de tirer partie à la fois de la puce graphique et du CPU [9].

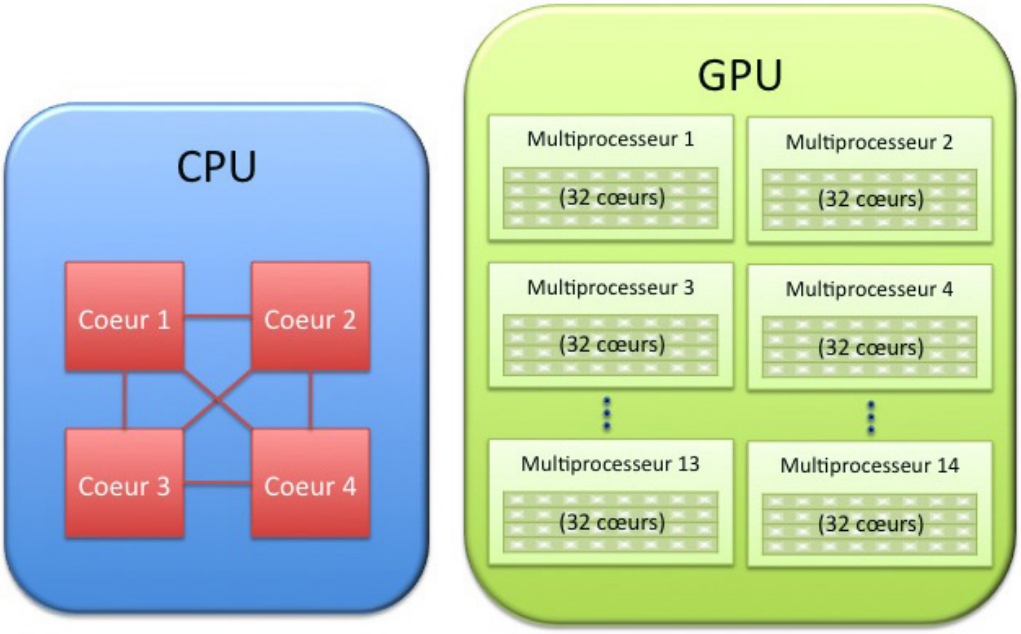


FIGURE 2.10 – Comparaison des architectures CPU et GPU.

Pour certaines expériences, il est proposé d'utiliser une carte ATI 4850 avec Brook+ comme langage de programmation. Ce langage permet d'implémenter l'algorithme d'estimation d'altitude présenté au chapitre 3. L'avantage de *Brook+* est la similitude au langage C. Toutefois, peu après l'adaptation de cet algorithme à partir de vues mixtes en GPGPU, ATI a mis de côté le développement de Brook+. La société oriente désormais son développement vers ATI Stream liée à la nouvelle norme ouverte : OpenCL. Pour des applications parallélisables, l'avantage de ce type d'architecture est le gain de temps de traitement face à une architecture CPU

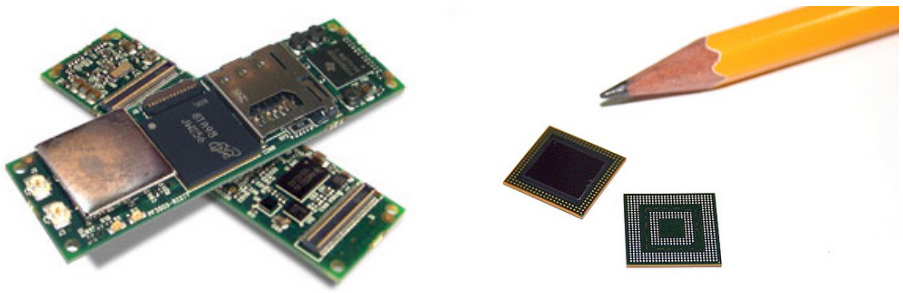
9. *Central Processing Unit*, Unité centrale de traitement.

MIMD [10] classique x86. En comparaison (fig.2.10) à un Intel Core 2 Duo cadencé à 3Ghz, pour le calcul de 50 homographies par exemple le gain est de 363 fois en faveur du GPU. Pour un algorithme d'estimation de l'altitude, le gain est de 45 fois toujours en faveur du GPU. Ces résultats donnent un aperçu de la possibilité d'équilibre de charge entre CPU et GPU. Cependant, le principal inconvénient de cette technologie reste la consommation énergétique, l'encombrement mais aussi les temps de transfert mémoire/GPU. Ainsi, cette architecture peut être utilisée uniquement dans des drones d'envergure importante.

La miniaturisation des CPU/GPU, et leur tendance à être fusionnés (projet AMD Fusion, Larrabee d'Intel), sont des arguments pour considérer cette technologie pour de futures applications embarquées. Si les GPU sont aujourd'hui inadaptés pour des aéronefs miniatures, d'autres processeurs de traitement SIMD sont capables de traiter un lot d'informations en parallèle comme les DSP [11].

Gumstix

Le Gumstix (fig.2.11(a)) est de la taille d'un *chewing-gum*. C'est un système complet basé sur une puce OMAP 3530 (fig.2.11(b)) composée d'un CPU d'architecture ARM et d'une mémoire RAM [12]. L'ARM est plus puissant qu'un processeur x86 équivalent. Pour les tests, la version Overo Fire a été utilisée.
Cette carte dispose en plus des composants cités précédemment, d'une interface



(a) (b)

FIGURE 2.11 – (a) Gumstix composé d'une puce OMAP 3530. (b) Puce OMAP 3530. - Sources : [Pixhawk 2010, OMA]

WiFi, Bluetooth, d'une partie graphique et d'un DSP. D'autres composants utilisent ces composants (BeagleBoard). Ces puces miniatures de nouvelle génération sont largement utilisées dans les consoles portables (Archos) mais aussi dans

10. *Multiple Instruction Multiple Data*, Plusieurs processeurs gérés indépendamment.
11. *Digital Signal Processor*, Processeur de signal numérique.
12. *Random Access Memory*, Mémoire à accès non séquentiel.

les téléphones mobiles (iPhone 3GS, Nokia N8xx, Palm Pre, Samsung Galaxy). L'avantage de telles cartes est la présence de CPU, GPU et DSP. Le GPU embarqué est peu puissant, bien trop faible pour rivaliser avec un CPU. Néanmoins, la présence du DSP ouvre la possibilité de traitements parallèles. Des algorithmes de traitement d'images peuvent être implémentés en temps réel. Par exemple, [Sanahuja 2010a] a implémenté le Lucas-Kanade de [Bradski 2000] sur Gumstix. Le résultat est un traitement temps réel d'un flux vidéo. L'inconvénient principal est la difficulté de prise en main de l'architecture ARM du processeur. En effet, la compatibilité avec les pilotes existants est assez restreinte. La plupart des constructeurs de caméras et de capteurs développent leurs pilotes pour des architectures x86. Pour ce type d'architecture, il convient de développer un système propre, allant de l'électronique du capteur au développement des pilotes. L'ensemble des programmes à implémenter et les librairies sont difficilement compilables sur une telle puce. Pour réduire le temps de compilation, il est indispensable d'utiliser la cross-compilation [Pixhawk 2010]. Enfin, pour la communication avec une caméra, un connecteur ISP [13] est spécialement prévu à cet effet. Ainsi, les informations brutes du capteur peuvent directement être intégrées dans le DSP ou le processeur évitant les délais de latence dus aux transferts entre les différentes mémoires.

Afin d'obtenir un système rapide à mettre en œuvre, l'OMAP 3530 du Gumstix n'est pas une solution idéale tant il nécessite diverses compétences et un temps d'investissement assez conséquent. L'hétérogénéité des architectures est un avantage dans la répartition des algorithmes de traitement mais également une difficulté technique à franchir. Mais si le Gumstix présente certains inconvénients, il reste l'un des processeurs de traitement miniature dont le rapport puissance/consommation est l'un des meilleurs du marché. Une fois maîtrisé et finement adapté c'est certainement un excellent choix pour l'application à des aéronefs compacts et robustes.

Atom

Pour rendre rapidement opérationnel un système, l'architecture x86 est une réponse à la plupart des contraintes précitées de pilotes propriétaires. L'implémentation d'une telle architecture permet l'utilisation de la plupart des caméras du marché et des différents capteurs. Toutefois, l'inconvénient principal comparé au Gumstix réside dans la faible puissance de traitement. En effet, ce type d'architecture dispose d'un CPU de type Atom d'Intel avec quelques instructions SIMD (SSE, MMX) et d'une carte graphique (GMA 950). La faible puissance de ces composants en font une puce quasiment inexploitable d'un point de vue GPGPU. Comparée à une puce OMAP 3530, le rapport puissance/consommation est bien moins

13. *Camera Interface Subsystem*

intéressant.

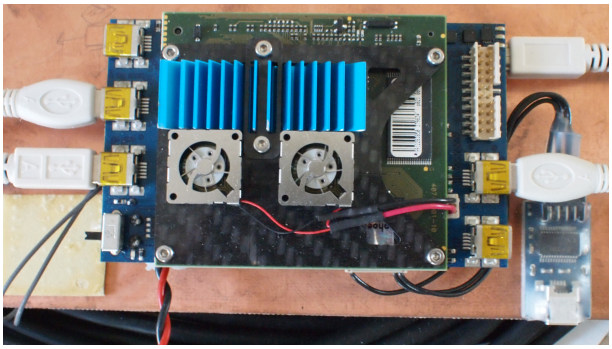


FIGURE 2.12 – Carte à base de processeur Atom d'Intel.

En raison de la facilité de déploiement et d'un besoin de réaliser des acquisitions embarquées, le choix s'est tourné vers une carte *Atom board* (fig.2.12) de la société *Ascending Technologies*. Ce choix est motivé par la faible consommation et la compatibilité avec les pilotes des caméras grand public. L'implémentation d'une telle carte est en effet rapide, et le développement d'algorithmes ne nécessite ni portage ni cross-compilation. Elle autorise l'utilisation d'un Linux standard de type Ubuntu. De plus, la consommation d'énergie est égale à celle du Gumstix (5 W) et son encombrement est un peu plus important (100 x 60 x 25 mm). Ce système dispose d'1 Go de RAM, de 7 ports USB 2.0, 2 ports série, 1 port LVDS (écran externe), 1 port Mini PCI Express et 1 port SDHC. Le port mini PCI Express permet l'intégration d'une carte Wifi et Bluetooth. Enfin le nombre de ports USB est un élément déterminant pour la connectique et l'évolutivité de la carte car l'absence de port éthernet autorise deux modes de communication : le wifi et le port série. Le premier à haut débit (54Mbps en wifi g et 300Mbps en wifi n) ouvre la possibilité au transfert d'images. Le port série quant à lui limite son utilisation à un contrôle de l'état et à une connexion console avec la carte.

2.4.3 Interfaces de communication

L'interface de communication est le moyen pour un système embarqué et autonome de communiquer avec l'extérieur. Dans ce travail, l'objectif est d'observer, suivre les évènements du drone et éventuellement transmettre les images. Deux pistes sont possibles : la communication câblée, et la communication sans fil.

Dans le premier cas, le rayon d'action est restreint. La liaison filaire est adaptée pour du débuggage, lancer une application ou contrôler le système de manière ponctuelle. Parfois, certaines expériences peuvent être réalisées avec des câbles. Ces câbles permettent de connecter les capteurs de vision et la centrale inertielle directement sur une station au sol. Cependant le rayon d'action est limité à environ

6 mètres et l'altitude à environ 3m. De plus, dans de telles conditions les risques d'accident sont élevés : le câble déstabilise fortement le drone et peut interrompre la rotation d'une hélice.

Dans le second cas, l'utilisation de liaison sans fil permet de communiquer sans lien physique. Cette solution semble être la plus sûre. La communication sans fil offre alors plusieurs applications.

- Le drone peut communiquer avec le sol pour transmettre son état.
- La station peut émettre des commandes.
- Le traitement déporté et une communication bidirectionnelle permettent de transmettre l'état et les données issues des capteurs et d'envoyer la commande résultante du traitement de ces données.

D'un point de vue matériel, deux principales solutions sans fils sont possibles :

- l'utilisation du wifi dans la bande des 2.4 GHz ou des 5 GHz permettant l'utilisation de la norme IEEE 802.11n à des cadences allant jusqu'à $300 Mbit/s$
- l'emploi d'un système de transmission vidéo adapté.

Par ailleurs, la transmission d'un flux de stéréovision, dont un couleur et un noir et blanc, complexifie fortement la transmission et force à diminuer la cadence vidéo. Par conséquent, deux méthodes d'exploitation des données sont retenues.

Tout d'abord les données sont traitées en différé. L'objectif est de sauvegarder les différents paramètres de vols et des flux vidéos grâce à la carte Atom puis de les traiter hors ligne. Pour cela deux alternatives sont proposées : l'enregistrement sur mémoire non volatile et l'enregistrement en déporté par wifi dont la portée en zone dégagée (le cas d'un drone) est en théorie de 300m (testé à plus de 100m d'altitude).

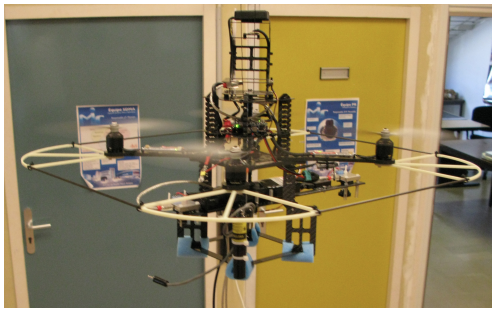
Ensuite, une deuxième solution vise à traiter les données en temps réel en utilisant un câble entre les capteurs du drone et le système de traitement. Cette solution est utilisée uniquement pour valider les algorithmes mais le déplacement du drone se limite à une zone restreinte.

2.4.4 Les quadri-rotors

Aéronefs à voilure tournante compacts et stabilisés, les quadri rotors disposent d'une capacité d'emport assez importante (jusqu'à 1kg) et provoquent peu de vibrations. Toutefois contrairement à des hélicoptères à voilure tournante, la stabilisation n'est pas mécanique mais réalisée par électronique.

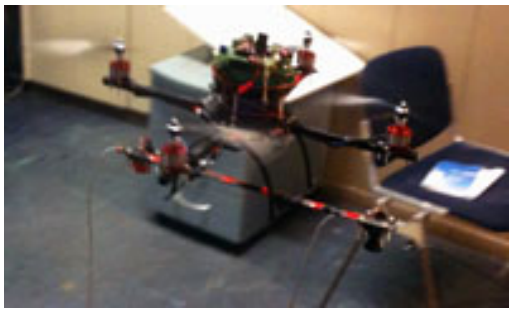
D'une part, la réalisation des prises de vue en extérieur s'effectue avec un drone MD4-1000 (fig. 2.13(b)) de la société Microdrone. Les informations issues des capteurs sont directement sauvegardées ou transmises à la station au sol par voie aérienne.

D'autre part, les vols en intérieur sont réalisés avec un aéronef conçu par le laboratoire Heudiasyc (fig. 2.13(c)) et avec un Pellican de la société *Ascending*



(a)

(b)



(c)

FIGURE 2.13 – (a) Pellican - Ascending Technologies. (b) MD4-1000 - Microdrones. (c) Drone quadri-rotor - Heudiasyc, UTC.

Technologies. Les capteurs sont reliés à la station au sol par câble USB. Enfin, le système final est intégré sur un Pellican (fig. 4.1(b)) dont la perspective de ces travaux est la commande à partir des algorithmes proposés en chapitres 3 et 4.

2.4.5 Solution retenue

Les capteurs d'attitude, de vision et les cartes de traitements constituent les éléments essentiels pour l'estimation des paramètres de navigation d'un drone. La compacité, la robustesse, la consommation d'énergie sont des critères importants lors d'une utilisation aérienne. La capacité d'emport des drones varie entre 500g et 1000g. La solution retenue doit considérer tous les éléments cités précédemment.

L'utilisation de la centrale 3DM de Microstrain répond à la contrainte de poids et de consommation d'énergie. Ensuite, les capteurs de vision LE12xx de la société IDS sont des capteurs légers, faibles consommateurs d'énergie. Il peuvent être synchronisés par matériel. Néanmoins ces capteurs sont sensibles aux perturbations électromagnétiques, en particulier celles générées par les moteurs d'un drone en action. Lors des premiers essais, ces caméras étaient sujettes à des "pompages".

Les perturbations agissent sur le *pixel clock*. L'effet est l'alternance d'images sur-exposées et sous-exposées. Ces caméras, d'un design le plus épuré possible, sont dépourvues de carcasse de protection. Un blindage électromagnétique a donc été réalisé afin d'éviter ces effets de pompage.

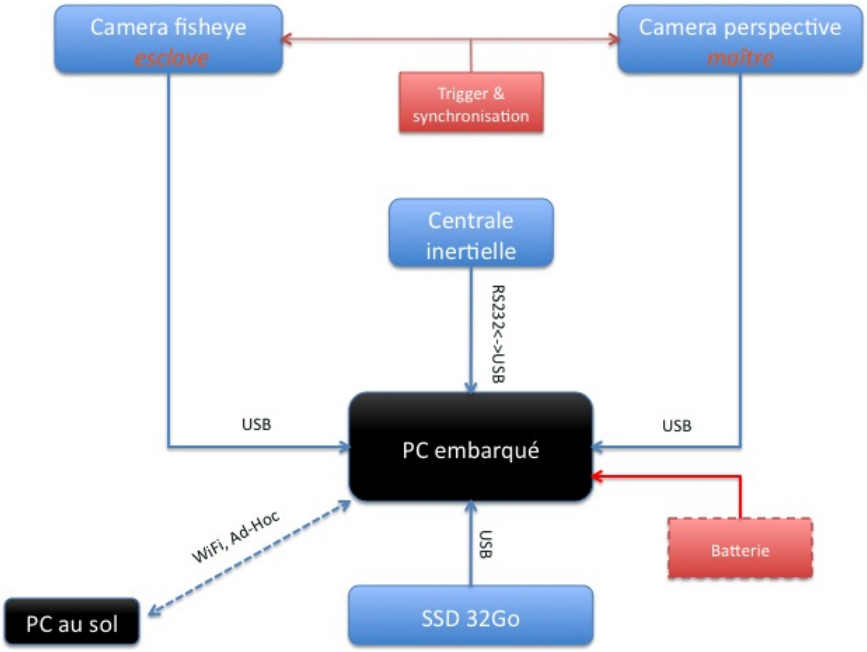


FIGURE 2.14 – Solution matérielle retenue.

Les capteurs et la carte embarquée (fig.2.14) sont testés pour évaluer la consommation en énergie. L'ensemble est alimenté par une batterie LiPo de 11.1 V, 1300 mA h soit 14.4 W/ h. La consommation totale du système est d'environ 9.4 W sans affichage extérieur et 12 W avec écran. L'autonomie du système complet est proche de 1 h30 min. Les caméras sont fixées sur une tige carbone avec la centrale inertielle. Elles sont orientées vers le bas pour une meilleure estimation de l'altitude, du mouvement et de l'attitude.

2.5 Conclusion

Ce chapitre à présenté un capteur de stéréovision hybride composé d'une caméra fisheye et d'une caméra perspective. Une centrale inertielle est ajoutée en complément ou en comparaison de ce système.

Le choix d'un tel système est tout d'abord justifié d'un point de vue mécanique, géométrique et algorithmique. En premier, la mécanique répond aux contraintes de vibration et de perturbation électromagnétique. En second, la géométrie est définie par une sphère pour la vue fisheye et un plan pour la vue perspective. Cette géométrie permet d'utiliser l'ensemble des algorithmes basés sur chacune des vues mais également de conserver l'aspect discret du voisinage planaire.

Ensuite, cette géométrie hybride a nécessité le développement d'un outil d'étalonnage spécifique, [Caron 2011]. Cet outil basé sur le VVS est capable d'étalonner n caméras à n modèles, et est valorisé en tant que logiciel libre et cross-plateforme.

Enfin, l'aspect matériel du capteur et des processeurs de traitement a été abordé. Deux configurations sont retenues : la première consiste à embarquer le processeur de traitement pour enregistrer les données et les traiter hors ligne. Une deuxième configuration propose de relier directement les capteurs du drone à une plateforme externe par l'intermédiaire de câbles. L'ensemble du système est embarqué grâce à l'une des trois plateformes quadri-rotor sélectionnées, de capacité d'emport d'environ $500\,\mathrm{g}$.

Par conséquent, le système retenu répond aux contraintes de compacité, de vibration et de faible consommation énergétique liées à un vol de drone. De plus, les drones sélectionnés sont définis pour des applications extérieures et intérieures ce qui nous permet de valider différents algorithmes dans des conditions variées.

À partir de la configuration du capteur proposé, des méthodes d'estimation des paramètres de navigation sont détaillées dans le chapitre 3. La configuration matérielle proposée dans ce chapitre est ensuite testée dans le chapitre 4.

CHAPITRE 3

Estimation des paramètres de navigation

Sommaire

3.1 Approche systémique 52
3.2 Attitude 54
3.3 Altitude 55
3.3.1 Principe 56
3.3.2 Cas des vues hybrides 57
3.3.3 Homographie entre une sphère et un plan 58
3.3.4 Algorithme 58
3.4 Estimation de mouvement 62
3.4.1 Mouvement du banc stéréo 64
3.4.2 Mouvement à partir des points du sol 65
3.4.3 Mouvement à partir des points de l'environnement 67
3.4.4 Fusion des points 3D issus des vues mixtes 67
3.4.5 Filtre de Kalman 68
3.4.6 Algorithme 69
3.5 Conclusion 70

Le chapitre précédent a proposé un capteur de stéréovision hybride destiné à estimer les paramètres de vol d'un drone. Ce système est composé d'une caméra fisheye, modélisée par une sphère et d'une caméra perspective, modélisée par un plan. L'apport de la vision hybride est un moyen de profiter des avantages de chacune des catégories de caméras mais aussi d'utiliser des méthodes basées sur le modèle perspectif ou sphérique.

Dans ce chapitre, nous proposons à partir de ce nouveau système de déterminer l'ensemble des paramètres de navigation d'un drone.

3.1 Approche systémique

Les paramètres d'état sont nécessaires au vol et à la stabilité d'un drone. Ils se caractérisent par l'attitude, définie par les angles de roulis α, tangage β, lacet γ, l'altitude d, la segmentation du sol, notée $\mathbf{G}$ et le mouvement composé d'une rotation $\mathbf{R}_{t+1}$ et d'une translation $\mathbf{t}_{t+1}$. Pour déterminer l'ensemble de ces paramètres par vision, une approche systémique est proposée afin d'étudier le capteur dans sa complexité.
D'abord, les méthodes employées sont encapsulées afin de déterminer l'ensemble

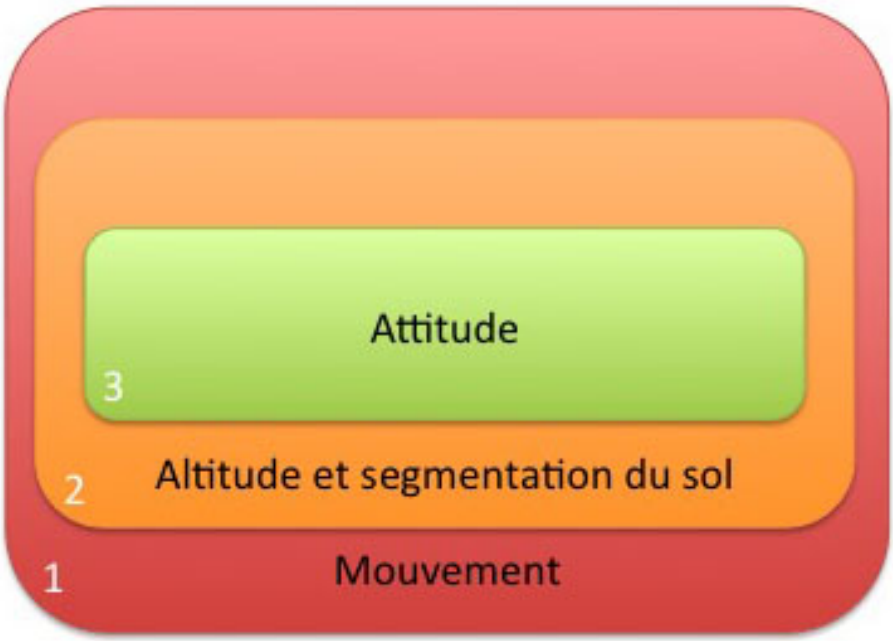


FIGURE 3.1 – Approche encaplusée.

de ces données (fig. 3.1) avec un seul capteur hybride. Dans cette démarche, chaque paramètre estimé au plus haut niveau est réemployé par le niveau inférieur. Cependant, un paramètre de niveau inférieur ne peut être déterminé tant que celui situé en amont n'a pas été traité. L'avantage d'un tel système est de pouvoir déterminer l'ensemble des paramètres avec un seul capteur.
Le système est tout d'abord supposé étalonné : $\mathbf{R}_c$, $\mathbf{t}_c$ sont connus. Puis, les paramètres d'attitude, estimés au niveau 3, permettent de déterminer à la fois l'assiette, la rotation relative entre deux mouvements notée $\mathbf{R}_{t+1}$ mais aussi la normale au sol notée $\vec{\mathbf{n}}$. Cette normale est ensuite utilisée conjointement avec les paramètres d'étalonnage ($\mathbf{R}_c$ et $\mathbf{t}_c$) pour rechercher l'altitude et segmenter le sol. À partir de cette altitude, estimée au niveau 2, il est possible de connaître, dans chaque image, les pixels appartenant au sol et donc la profondeur des points associés. Grâce à ce

postulat, le mouvement de niveau 1 peut être estimé en découplant la rotation $\mathbf{R}_{t+1}$ connue au niveau 3 et la translation $\mathbf{t}_{t+1}$. Cette translation est obtenue à partir du suivi des points localisés sur le sol dont la profondeur est connue et ceux dans l'environnement.

L'ensemble des interactions au sein de ce système hiérarchisé est ensuite illustré par l'aspect collaboratif du système. en effet, trois modes de fonctionnement le caractérisent :

- un aspect **séparé**, où chaque caméra peut être utilisée indépendamment et utiliser soit des méthodes adaptées aux caméras omnidirectionnelles grâce au modèle sphérique, soit des méthodes adaptées aux caméras perspectives grâce au modèle planaire ;
- un aspect **stéréoscopique**, où les deux vues permettent de retrouver une information 3D à partir de la géométrie particulière, hybride ;
- un aspect **coopératif**, où l'apport des informations issues d'un mode séparé ou stéréoscopique permet d'enrichir un autre mode.

L'avantage principal de ces trois modes de fonctionnement est l'échange possible des informations dans le but d'améliorer la robustesse, la précision et les capacités natives de chaque capteur. Ces modes de fonctionnement seront utilisés dans les différentes méthodes proposées dans ce chapitre.

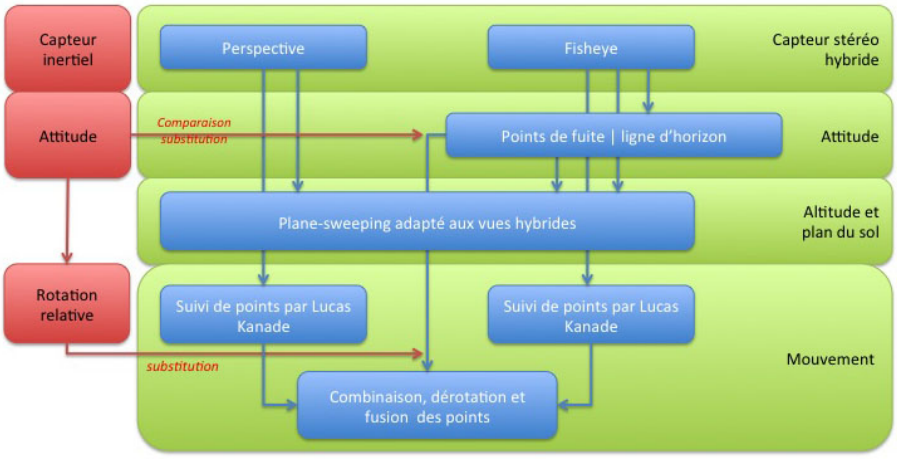


FIGURE 3.2 – Système global d'estimation des paramètres de vol à partir du capteur de stéréovision hybride.

Enfin, les méthodes proposées permettent à partir du même capteur hybride de déterminer seul, ou couplé à une centrale inertielle, trois paramètres d'état du drone. Cependant, l'utilisation de ce capteur est ouvert à l'emploi d'autres tâches telles que la surveillance ou l'observation. Les méthodes proposées s'articulent en

trois idées principales.
En premier, l'attitude et l'orientation sont déterminées soit par centrale inertielle, soit par vision [Demonceaux 2007] dans un environnement urbain / intérieur et [Demonceaux 2006b, Demonceaux 2006a] dans un environnement naturel. Dans ce cas, le capteur fonctionne en mode séparé.
En second, la connaissance de la calibration, de l'attitude et de l'homographie entre les deux vues permettent d'estimer l'altitude puis de segmenter le plan du sol [Eynard 2010]. Dans ce cas le capteur fonctionne en mode stéréoscopique et coopératif si l'attitude est estimée par vision.
En troisième, le mouvement est découplé en une rotation et une translation. D'abord la rotation est déterminée soit par vision, soit par centrale inertielle. Puis la translation est estimée à partir du suivi de points dans les vues hybrides [Eynard 2011]. Grâce à l'estimation de l'altitude et à la segmentation du plan par vision, la translation est calculée à l'échelle métrique. Dans ce cas, les trois modes sont utilisés. D'abord le mode coopératif : l'altitude fournit une information sur la segmentation du sol et sur la distance des points observés. De plus, l'attitude et l'orientation fournissent une information sur la rotation. Ensuite, les points suivis utilisent le mode séparé. Enfin, la fusion des points fait intervenir l'aspect stéréoscopique.
Pour finir, l'ensemble de ces méthodes ne nécessitent aucune mise en correspondance entre les paires stéréo grâce à l'utilisation d'une méthode dense et à la connaissance de la profondeur des points observés. L'implémentation des méthodes proposées sur le système coopératif est illustrée à la figure 3.2.

3.2 Attitude

Il convient de distinguer trois approches pour l'estimation de l'attitude en fonction de l'environnement considéré et/ou du capteur utilisé.

Tout d'abord, dans un environnement urbain, ou dans un environnement clos, l'attitude et l'orientation peuvent être déterminées à partir d'une vue omnidirectionnelle [Bazin 2008, Demonceaux 2007]. Les nombreuses lignes verticales et horizontales présentes dans l'environnement 3D sont détectées dans l'image omnidirectionnelles. Ces lignes sont ensuite projetées sur la sphère équivalente où chaque ensemble de droites parallèles s'intersectent en deux points opposés appelés points de fuite. Ainsi, il est possible de déterminer les trois directions correspondantes à l'attitude du drone (fig. 3.3).

Ensuite, en environnement naturel, les lignes sont peu présentes. Toutefois, dans les conditions définies par le vol à vue, il est possible d'utiliser la ligne d'horizon observée par une caméra omnidirectionnelle [Demonceaux 2006b]. Cette ligne d'horizon est projetée sur la sphère équivalente où la segmentation ciel terre est réa-

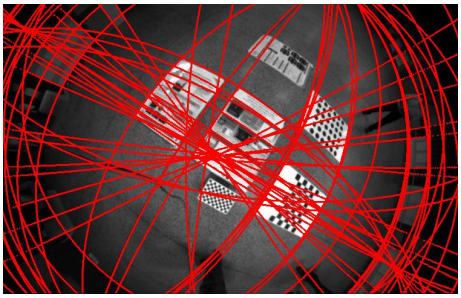


FIGURE 3.3 – Estimation des points de fuite à partir des lignes dans l'image en environnement clos.

lisée (fig. 3.4(c)). Cette méthode permet d'obtenir des résultats à la fois à partir de vues catadioptriques (fig. 3.4(a)) et fisheye (fig. 3.4(b)).

Enfin, en troisième approche, l'attitude peut être obtenue par centrale inertielle.

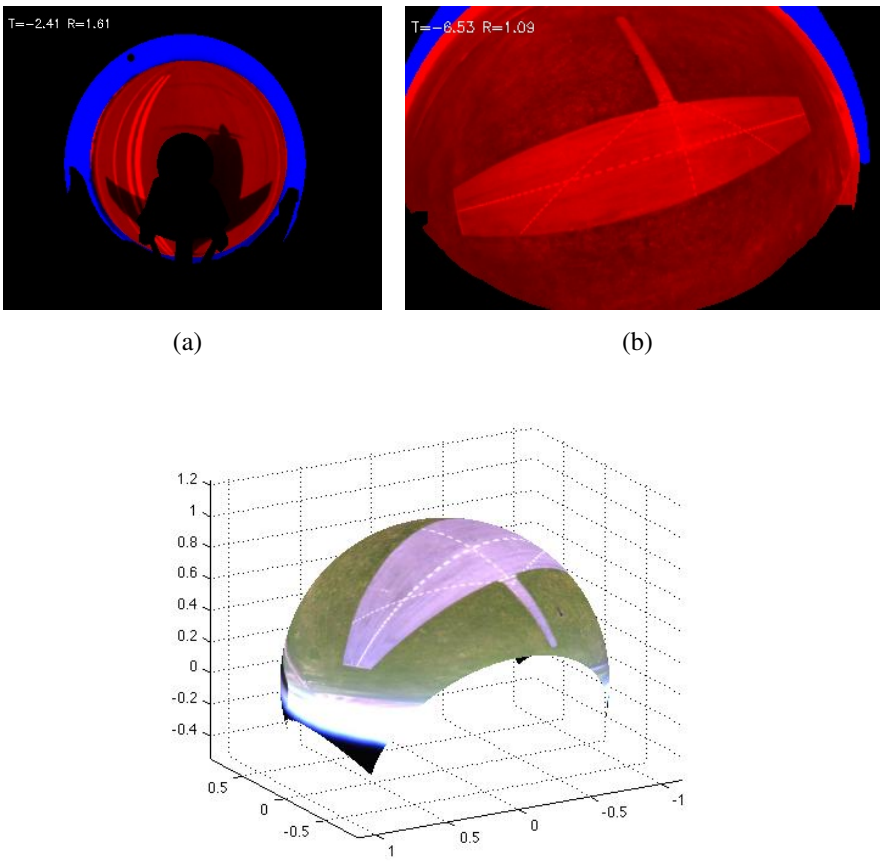
$\vec{\mathbf{n}} = (n_x, n_y, n_z)$ définit la normale au sol. Ce sol est supposé tangent à la surface de la terre. Cette normale est utilisée dans la section 3.3. Il convient d'utiliser l'équation 3.1 pour l'exprimer dans le repère caméra.

$$\begin{bmatrix} n_x \\ n_y \\ n_z \end{bmatrix} = {}^t\mathbf{R}_m \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} \tag{3.1}$$

Pour conclure, l'attitude peut être déterminée soit à partir de la vue fisheye ou soit à partir de la centrale inertielle et les méthodes présentées permettent d'estimer l'attitude dans différents types d'environnements.

3.3 Altitude

L'étape précédente a permis de déterminer l'attitude. En supposant le sol plan et dominant, cette attitude permet d'obtenir la normale au sol, notée $\vec{\mathbf{n}}$, observée par le système. À partir de cette information et de l'information de calibration, nous cherchons à déterminer l'altitude. L'une des méthodes possible consiste à balayer un ensemble d'altitudes et de déterminer celle pour laquelle l'erreur associée est minimale. Ainsi, nous proposons de déterminer l'altitude d grâce à une méthode de *plane-sweeping* en exploitant l'homographie entre deux vues. Cette méthode a été introduite par [Collins 1996] puis proposée en temps réel par [Gallup 2007].
Nous suggérons dans cette partie d'adapter cet algorithme au système hybride en vue d'estimer l'altitude et de segmenter le sol en temps réel. Cette altitude se définit par la distance du plan au centre optique de la vue fisheye (fig. 3.6).



(a) (b)

(c)

FIGURE 3.4 – Estimation de l'attitude basée sur la ligne d'horizon (a) Estimation à partir d'une vue catadioptrique. (b) Estimation à partir d'une vue fisheye. (c) Image de l'horizon projetée sur la sphère équivalente.

3.3.1 Principe

Le principe du *plane-sweeping* consiste à retrouver l'orientation et la distance de plans 3D observés par plusieurs caméras et reliés par une homographie. Pour ce faire, la méthode vise à tester un ensemble d'hypothèses de plans, défini par une normale $\vec{\mathbf{n}}$ et une distance d. Pour chaque pixel d'une vue de référence, le meilleur plan est obtenu à partir d'un score de ressemblance. Ce score est maximum lorsque la différence du voisinage d'un pixel de la vue de référence et de la vue transformée par homographie est minimum. L'avantage de cette méthode dense est de permettre la reconstruction d'un environnement 3D sans mise en correspondance entre les vues.

Soient $\mathbf{I}$ et $\mathbf{I}_{ref}$ respectivement la vue courante et la vue de référence. $\mathbf{I}(\mathbf{x})$ est

l'intensité du pixel normalisé $\mathbf{x}$. Notons, l'image $\mathbf{I}^*$ la transformation de l'image $\mathbf{I}$ par la matrice d'homographie $\mathbf{H}$. Ainsi, pour un pixel $\mathbf{x}$ situé à l'altitude d du plan Π, $\vec{\mathbf{n}}$ la normale à ce plan et $\mathbf{R}_c$, $\mathbf{t}_c$ les paramètres extrinsèques des deux vues, le calcul de l'homographie est défini par l'équation suivante :

$$\mathbf{I}^*(\mathbf{x}, d) = \mathbf{I}(\mathbf{H}_{\mathbf{x},d}) = \mathbf{I}((\mathbf{R}_c - \mathbf{t}_c\frac{\mathbf{n}^T}{d})\mathbf{x}) \tag{3.2}$$

Cependant, dans le cas hybride, le modèle sphérique pose la difficulté d'échantillonnage liée à la recherche du voisinage. C'est pourquoi il est préférable d'estimer l'erreur sur l'image globale et non sur un voisinage local. Ainsi, la meilleure estimation de $\mathbf{H}$ correspond à l'erreur globale ε minimale. Cette erreur est définie entre la transformation $\mathbf{I}^*$ par homographie et la vue de référence $\mathbf{I}_{ref}$ (eq. 3.3) sur l'ensemble des pixels appartenant au sol segmenté noté $\mathbf{G}$.

$$\varepsilon = \sum_{\mathbf{x}\in\mathbf{G}} |\mathbf{I}^*_{\mathbf{x}} - \mathbf{I}_{\mathbf{x},ref}| \tag{3.3}$$

3.3.2 Cas des vues hybrides

Nous proposons d'estimer l'altitude d et la segmentation du plan du sol par *plane-sweeping* hybride. $\mathbf{R}_c$, $\mathbf{t}_c$, sont connus par calibration. $\vec{\mathbf{n}}$ est estimée par centrale inertielle ou à partir des méthodes proposées en section 3.2. La vue perspective $\mathbf{I}_{ref}$ est définie comme référence. Dans cette vue, le plan du sol est supposé dominant. Ainsi le *plane-sweeping* s'effectue sur l'ensemble de l'image et sera affiné de manière itérative. En parallèle, lors de la segmentation du plan du sol, le voisinage de chaque pixel est utilisé. Grâce à la modélisation planaire du capteur de référence, les étapes d'échantillonnage et d'interpolation sont évitées. En effet, la projection de l'image fisheye par homographie vers l'image de référence permet d'obtenir directement une image planaire et évite toute recherche d'un voisinage sur la sphère.
Ensuite, les images $\mathbf{I}$ obtenues avec la caméra fisheye sont projetées sur la sphère puis projetées par homographie sur le plan de référence noté $\mathbf{I}^*$. En considérant le drone dans un état stable, les caméras sont supposées pointées vers le sol. Par conséquent, l'espace de travail de la vue fisheye est localisé au centre de l'image. Utiliser cette zone de travail présente plusieurs avantages : la surface exploitée présente moins de distorsions et une meilleure résolution comparée au reste de l'image. La méthode de *plane-sweeping* adaptée au capteur hybride est détaillée dans la figure 3.5 puis traitée dans la partie suivante.

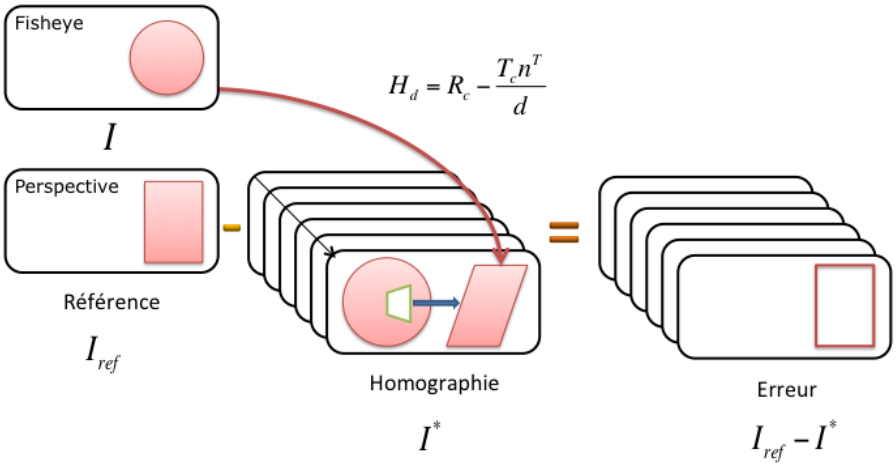


FIGURE 3.5 – Plane-sweeping entre deux vues hybrides.

3.3.3 Homographie entre une sphère et un plan

L'homographie relie la projection d'un plan 3D formé sur chacune des deux vues hybrides (fig. 3.6). Dans [Hartley 2004], l'homographie lie la projection d'un plan 3D sur deux plans. Dans le travail de [Mei 2006], l'homographie lie la projection d'un plan 3D sur deux sphères. De manière générale, une homographie est définie par l'équation suivante :

$$\mathbf{H} = \mathbf{R}_c - \mathbf{t}_c \frac{\mathbf{n}'}{d} \tag{3.4}$$

Pour un système à deux caméras perspectives, $\mathbf{x}^*$ est la projection de $\mathbf{x}$ sur la perspective par homographie $\mathbf{H}^{-1}$. Nous obtenons alors l'équation 3.5.

$$\mathbf{x}^* \sim \mathbf{H}^{-1}\mathbf{x} \tag{3.5}$$

Ensuite, ces deux projections planaires sont remplacées par une projection sphérique et une projection planaire obtenues dans l'équation 3.6 à l'échelle près.

$$\mathbf{x} \sim \mathbf{x}_s^* = \rho' \mathbf{H}\mathbf{x}_s \tag{3.6}$$

Ainsi, l'homographie $\mathbf{H}$ dépend de $\mathbf{R}_c$, $\mathbf{t}_c$, de l'attitude $\vec{\mathbf{n}}$ et de l'altitude d. Puisque $\mathbf{R}_c$, $\mathbf{t}_c$ et $\vec{\mathbf{n}}$ sont connues, la partie suivante traite de l'algorithme de segmentation du plan du sol et de l'estimation de l'altitude recherchée d.

3.3.4 Algorithme

Principe général

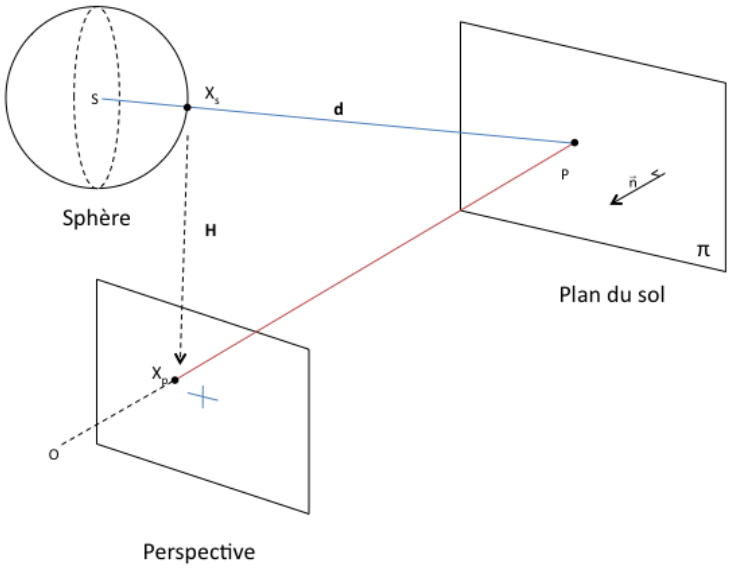


FIGURE 3.6 – Homographie sphere/plan.

Algorithme 1 Algorithme d'estimation d'altitude et segmentation du sol - initialisation

Estimation$(d_{min}, d_{max}, s, \Delta d, \mathbf{I}_{ref}, \mathbf{I}, thres, \mathbf{n}, \mathbf{R}_c, \mathbf{t}_c)$

%Initialisation

$a_0 = \widehat{d_{-1}} = d_{min}$

$b_0 = \widehat{d_0} = d_{max}$

$G = \{p \in \mathbf{I}_{ref}\}$

tant que $|\widehat{d_k} - \widehat{d_{k-1}}| > \Delta_d$ **faire**

%Estimation de la meilleure altitude

$\widehat{d_{k+1}} = \text{argmin}_{d \in \{\frac{b_k - a_k}{s-1} t + a_k; \in [0, s-1]\}} \sum_{\mathbf{x} \in G} |\mathbf{I}_{ref}(\mathbf{x}) - \mathbf{I}^*(\mathbf{x}, d_k)|$

%Transformation par homographie

$\mathbf{I}^*(\mathbf{x}, \widehat{d}) = \mathbf{I}((\mathbf{R}_c - \mathbf{t}_c \frac{\mathbf{n}^T}{\widehat{d}})\mathbf{x})$

%Estimation du masque d'inliers/outliers

$G = \{\mathbf{x}, \frac{\sum_{\mathbf{x} \in W_\mathbf{x}} |\mathbf{I}_{ref}(\mathbf{x}_1) - \mathbf{I}^*(\mathbf{x}_1, \widehat{d})|}{\sum_{\mathbf{x} \in W_\mathbf{x}} \mathbf{I}_{ref}(\mathbf{x}_1)} < thres\}$

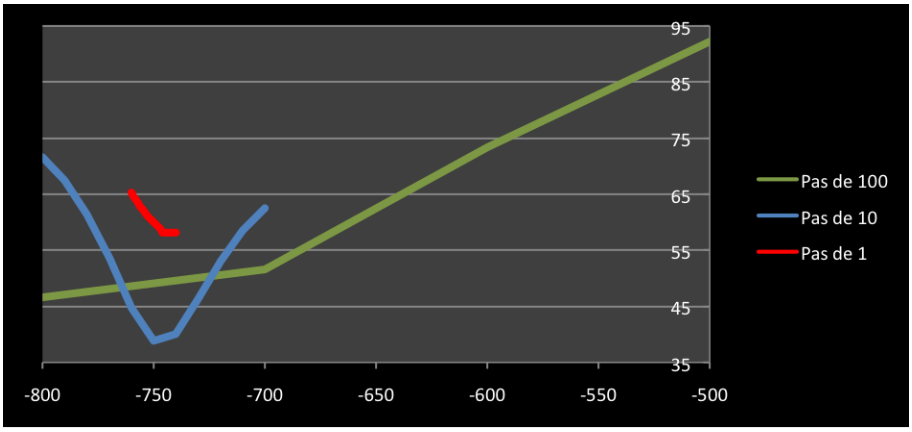
%Estimation du nouvel espace de recherche en fonction de l'échantillonnage

$a_{k+1} = \widehat{d_{k+1}} - \frac{b_k - a_k}{s-1}$

$b_{k+1} = \widehat{d_{k+1}} + \frac{b_k - a_k}{s-1}$

$k = k + 1$

Return $\widehat{d}$



(a)

Pas de 100mm		Pas de 10mm		Pas de 1mm	
Altitude	Erreur	Altitude	Erreur	Altitude	Erreur
-800	46,6103	-800	71,5969	-760	65,2918
-700	51,6341	-790	67,525	-759	64,6357
-600	73,4368	-780	61,4158	-758	63,9884
-500	92,2213	-770	53,6202	-757	63,3586
		-760	44,6512	-756	62,7516
		-750	38,8528	-755	62,1631
		-740	39,9866	-754	61,6178
		-730	46,2466	-753	61,1197
		-720	53,0472	-752	60,6927
		-710	58,4904	-751	60,3046
		-700	62,4665	-750	59,9255
				-749	59,4855
				-748	59,0647
				-747	58,7127
				-746	58,4557
				-745	58,2718
				-744	58,1823
				-743	58,1501
				-742	58,1788
				-741	58,2592
				-740	58,3918

(b)

FIGURE 3.7 – (a) Mise à jour du masque en fonction de l'itération (erreur en fonction de l'altitude en mm). (b) Tableau de valeurs correspondant.

Estimer l'altitude par *plane-sweeping* consiste à minimiser ε (éq. 3.3) par la recherche de d, réalisée de façon exhaustive. Pour diminuer le temps de calcul, l'exploration de d est limitée à un intervalle défini par $[d_{min}, d_{max}]$, avec d_{min} l'altitude minimale et d_{max} l'altitude maximale. Cette exploration s'effectue en deux temps, l'initialisation puis la mise à jour :

- l'initialisation est choisie en rapport avec la configuration du drone. L'algorithme est initialisé au décollage. L'intervalle initial est assez large (algorithme 1).
- Mise à jour : la meilleure altitude estimée à l'étape d'initialisation ou à l'instant $t-1$ est réutilisée afin d'obtenir un espace de recherche plus petit à l'instant t. Cet espace est déterminé à partir de $d_t \in [d_{t-1} - r_d, d_{t-1} + r_d]$ avec r_d dans l'équation 3.7.
  v_v représente la vitesse ascensionnelle maximale du drone et fps le nombre d'estimations par seconde dépendant du matériel utilisé. Cet ensemble dépend de la vitesse verticale maximale du drone (environ $\pm 7,5 m/s$ pour un MD4-1000) et de la puissance de calcul.
  Nous obtenons alors $\widehat{d}$ =**Estimation**$(d_{t-1} - r_d, d_{t-1} + r_d, step, \Delta d)$.

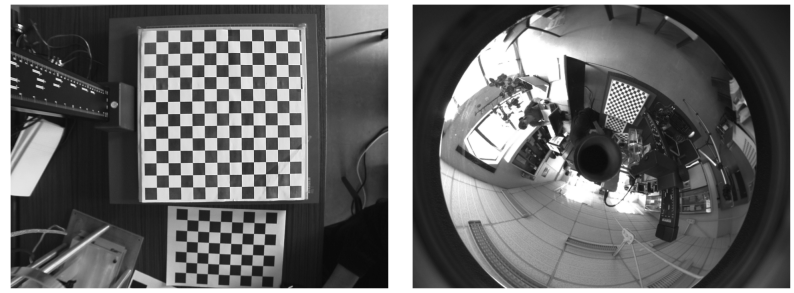
$$r_d = \frac{v_v}{fps} 10^3 \tag{3.7}$$

Pour distinguer le sol des obstacles, un seuil est appliqué sur la carte d'erreur calculée à chaque itération. La segmentation est représentée par un masque $\mathbf{G}$ déterminé à partir du seuillage $thres$ de la carte d'erreurs globales obtenue. Ce masque se compose de deux couleurs : noir pour les obstacles, blanc pour le plan du sol. Ce masque est ensuite mis à jour dans la figure 3.7(a) pour donner la figure 3.8(d) (voir algorithme 1). Le tableau 3.7(b) donne un aperçu des étapes de recherche de l'altitude. Cette recherche est effectuée de manière pyramidale, selon trois pas différents (1 mm, 10 mm et 100 mm).

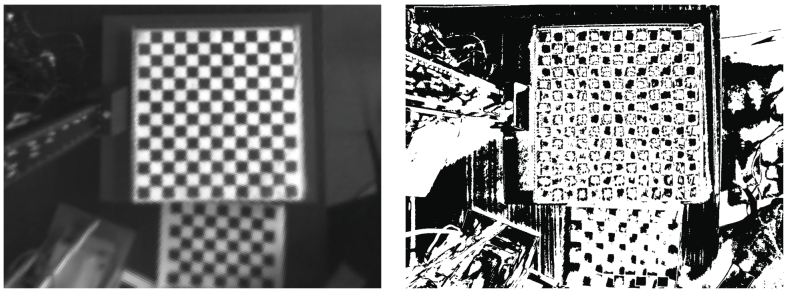
Optimisations

Deux algorithmes sont proposés. Tout d'abord, l'algorithme présenté précédemment (fig.3.9(a)) permet l'estimation de l'altitude et la segmentation du plan du sol. Pour l'intensité de chaque pixel de la vue de référence (la vue perspective), son correspondant dans la vue fisheye est comparé par homographie $\mathbf{H}$ en passant sur la sphère équivalente. Cet algorithme est implémenté sur GPU et permet d'avoir une mise à jour temps réel. Néanmoins, le GPU reste inadapté pour des applications embarquées. Le passage sur la sphère, le calcul du plus proche voisin et la recherche des points image associés aux deux vues constituent les étapes à optimiser.

L'utilisation de CPU et de cartes embarquées nécessitent le développement



(a) (b)



(c) (d)

FIGURE 3.8 – (a) Vue perspective. (b) Vue catadioptrique (c) Homographie de la vue omnidirectionnelle sur la vue perspective finale. (d) Masque final

d'une deuxième version de l'algorithme. Cette version se limite à l'estimation de l'altitude uniquement. A l'initialisation du programme, une LUT (*Look Up Table*, table de correspondance) est générée pour chaque pixel de la vue fisheye vers la sphère (fig. 3.10). Ensuite, pour chaque homographie, la surface correspondant à la zone observée par la perspective est définie dans la vue fisheye par $\mathbf{H}$ (fig. 3.9(b)). Les pixels situés dans la région commune perspective/fisheye sont ensuite projetés vers la vue de référence (perspective) par l'intermédiaire de la LUT (fig. 3.9(c)) par $\mathbf{H}^{-1}$. Cette optimisation permet un gain sur CPU de 7hz à 250hz entre l'algorithme standard et optimisé. Néanmoins, l'inconvénient de cette optimisation est la perte de l'information de segmentation. Cette information peut être estimée une fois l'altitude déterminée, au détriment d'une estimation moins précise comparée à l'algorithme proposé initialement.

3.4 Estimation de mouvement

Les sections précédentes ont proposé d'estimer l'attitude par vision ou centrale inertielle puis l'altitude et la segmentation du sol par vision. L'ensemble de ces paramètres est essentiel pour déterminer le mouvement, défini par le déplacement

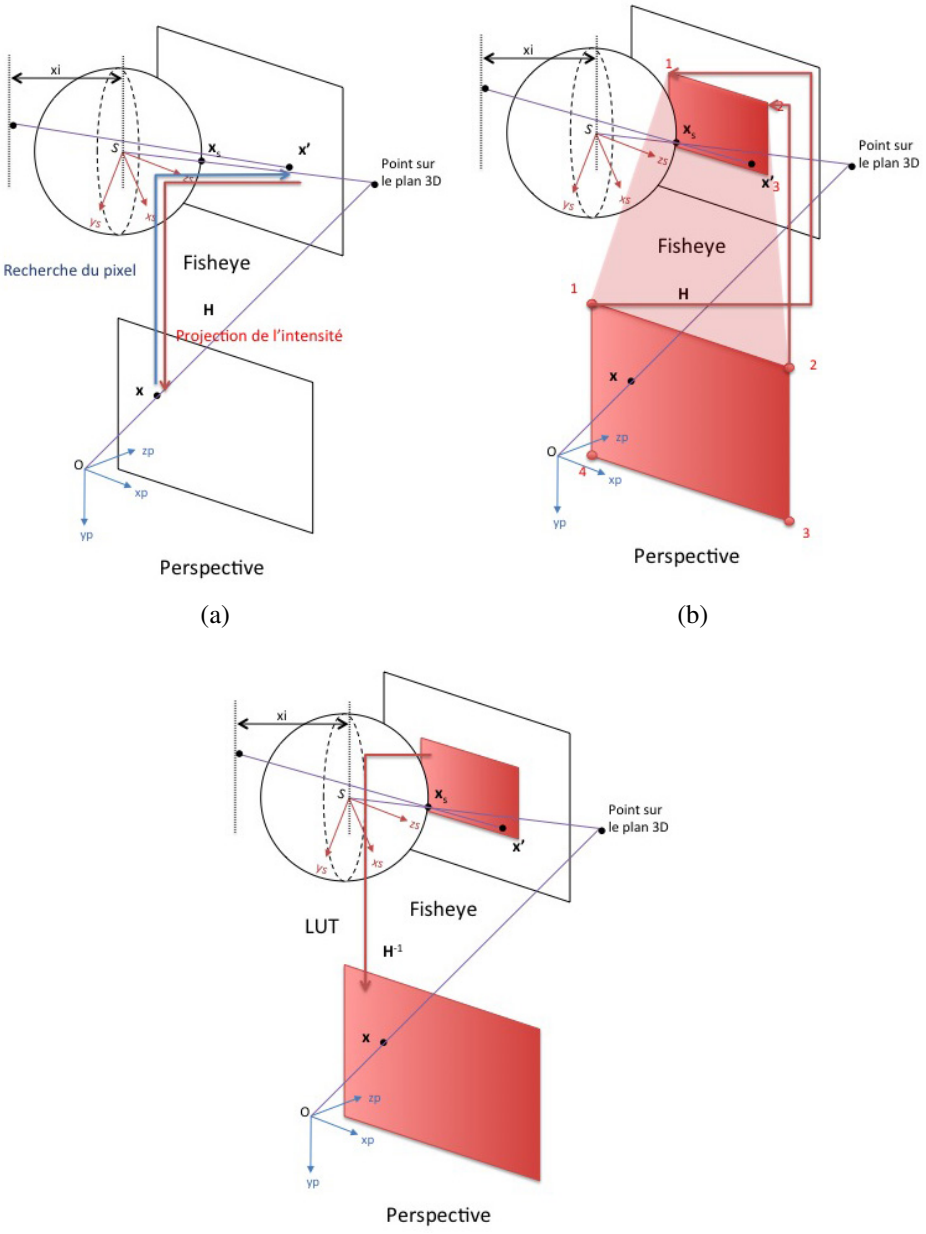


FIGURE 3.9 – (a) Algorithme standard - Homographie de la vue courante vers la référence pour chaque pixel référence. (b) Algorithme optimisé - Calcul de la zone de traitement pour la LUT. (c) Algorithme optimisé - Homographie de la vue courante vers la référence par LUT.

d'un drone entre deux instants. On peut découpler un mouvement entre l'instant $t+1$ et l'instant t, en une rotation $\mathbf{R}_{t+1}$ et une translation $\mathbf{t}_{t+1}$. Cette rotation est

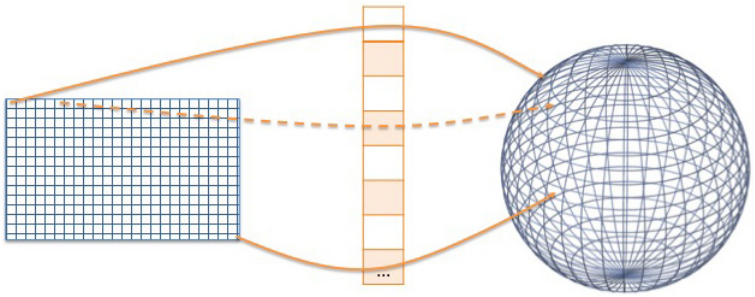


FIGURE 3.10 – LUT entre l'image fisheye et la sphère.

déterminée soit par centrale inertielle soit par vision grâce à l'une des méthodes précitée.

Ainsi, nous suggérons dans cette section de calculer la translation $\mathbf{t}_{t+1}$ à partir des points suivis indépendamment dans chacune des vues. Deux types de points (fig. 3.11) se distinguent :

- les points localisés sur le sol dont l'altitude et la segmentation sont supposés connues grâce aux travaux présentés précédemment : ils apportent l'échelle métrique et améliorent la précision
- les autres points de l'environnement dont la profondeur est inconnue : ils améliorent la robustesse au bruit

Ces deux types de points issus de chacune des vues sont ensuite fusionnés pour estimer la translation. Nous proposons de filtrer la translation estimée par un filtrage de Kalman.

3.4.1 Mouvement du banc stéréo

Dans chacune des images, on suppose les points suivis par flot optique.

Soit $\mathbf{x}_t$, un point suivi dans l'image acquise à l'instant t. Chaque point est lié relativement au repère monde $\mathbf{X}_m$ par une rotation ${}^t\mathbf{R}_m$ et une translation ${}^t\mathbf{t}_m$ dans l'équation 3.8. Pour un couple de points notés $\mathbf{x}_t$ et $\mathbf{x}_{t+1}$ correspondant à un mouvement rigide entre les instants t et $t+1$, on peut exprimer la rotation entre les deux repères image par l'équation 3.9. On obtient ainsi pour la rotation $\mathbf{R}_{t+1}$ liée au mouvement dans l'équation 3.10 ainsi que la translation $\mathbf{t}_{t+1}$. Enfin, à partir de l'altitude d nous déterminons d_t, la distance du point image à un point appartenant au sol $\mathbf{G}$ à l'instant t (voir fig.3.12) et obtenons l'équation 3.11.

$$\mathbf{x}_t = {}^t\mathbf{R}_m\mathbf{X}_m + {}^t\mathbf{t}_m \Longleftrightarrow \mathbf{X}_m = {}^t\mathbf{R}_m^{-1}(\mathbf{x}_t - {}^t\mathbf{t}_m) \tag{3.8}$$

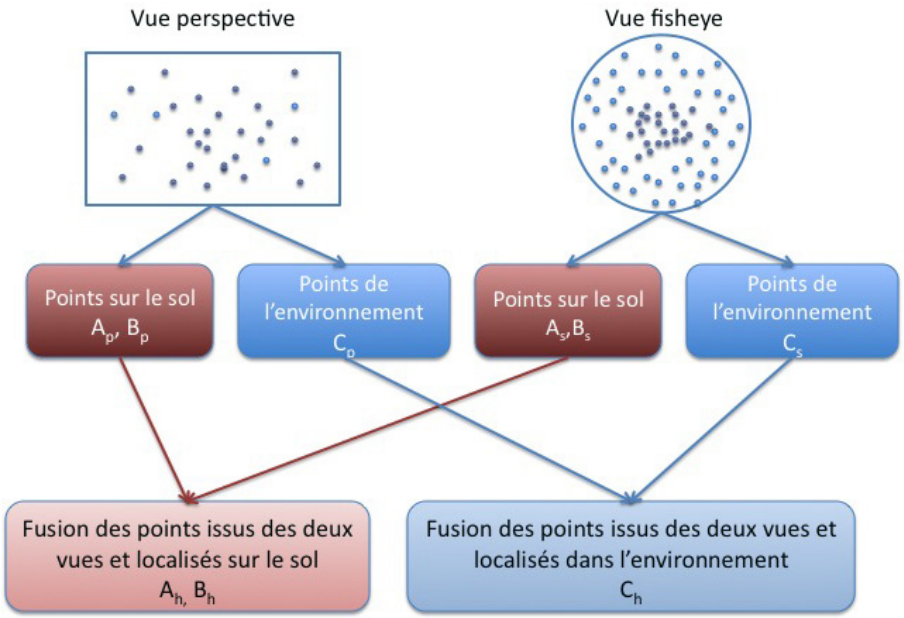


FIGURE 3.11 – Principe général de la méthode d'estimation du mouvement.

$$\mathbf{x}_{t+1} = {}^{t+1}\mathbf{R}_m \, {}^{t}\mathbf{R}_m^{-1} \mathbf{x}_t \tag{3.9}$$

$$\mathbf{R}_{t+1} = {}^{t+1}\mathbf{R}_m \, {}^{t}\mathbf{R}_m^{-1} \tag{3.10}$$

$$\mathbf{x}_{t+1} \sim d_t \mathbf{R}_{t+1} \mathbf{x}_t + \mathbf{t}_{t+1} \tag{3.11}$$

3.4.2 Mouvement à partir des points du sol

À partir du suivi des points appartenant au sol il est possible de déterminer le mouvement à l'échelle métrique (voir fig.3.12) à partir de l'équation 3.12. Ainsi, en écrivant cette équation sous forme matricielle, on obtient l'équation $\mathbf{At}_{t+1} = \mathbf{B}$ avec $\mathbf{x}_{t+1} = (x_{t+1}, y_{t+1}, z_{t+1})^T$, $\mathbf{A}$ exprimé dans l'équation 3.13 et $\mathbf{B}$ exprimé dans l'équation 3.14. Puis, pour le suivi d'au moins deux points, il est possible de déterminer $\mathbf{t}_{t+1}$ par une méthode des moindres carrés. La translation est alors exprimée à partir de la vue perspective puis de la vue fisheye et par conséquent à partir des deux vues hybrides.

- Dans la vue perspective, on obtient à partir des équations 3.13, 3.14 et des points localisés sur le sol, les matrices $\mathbf{A}_p$ et $\mathbf{B}_p$.
- Dans la vue fisheye, on obtient à partir des équations 3.13, 3.14 et des points localisés sur le sol, les matrices $\mathbf{A}_s$ et $\mathbf{B}_s$.

– Dans le cas hybride, on obtient $\mathbf{A}_h$ (eq. 3.19) en combinant $\mathbf{A}_s$ et $\mathbf{A}_p$. Puis on obtient $\mathbf{B}_h$ (eq. 3.20) en combinant $\mathbf{B}_s$ et $\mathbf{B}_p$. A partir de $\mathbf{A}_h$ et de $\mathbf{B}_h$ il est possible de déterminer $\mathbf{t}_{t+1}$ à partir des points situés sur le sol.
On note n le nombre de points perspectifs et m le nombre de points sphériques.

$$(\mathbf{x}_{t+1} \wedge \mathbf{t}_{t+1}) = -d_t(\mathbf{x}_{t+1} \wedge \mathbf{R}_{t+1}\mathbf{x}_t) \tag{3.12}$$

$$\mathbf{A} = \begin{bmatrix} 0 & -z_{t+1} & y_{t+1} \\ z_{t+1} & 0 & -x_{t+1} \\ -y_{t+1} & x_{t+1} & 0 \end{bmatrix} \tag{3.13}$$

$$\mathbf{B} = -d_t(\mathbf{x}_{t+1} \wedge \mathbf{R}_{t+1}\mathbf{x}_t)^T \tag{3.14}$$

$$\mathbf{A}_p = \begin{bmatrix} 0 & -z_{p0,t+1} & y_{p0,t+1} \\ z_{p0,t+1} & 0 & -x_{p0,t+1} \\ -y_{p0,t+1} & x_{p0,t+1} & 0 \\ \vdots & \vdots & \vdots \\ 0 & -z_{pn,t+1} & y_{pn,t+1} \\ z_{pn,t+1} & 0 & -x_{pn,t+1} \\ -y_{pn,t+1} & x_{pn,t+1} & 0 \end{bmatrix} \tag{3.15}$$

$$\mathbf{B}_p = \begin{bmatrix} -d_{p0}(\mathbf{x}_{p0,t+1} \wedge \mathbf{R}_{t+1}\mathbf{x}_{p0,t}^T)^T \\ \vdots \\ -d_{pn}(\mathbf{x}_{pn,t+1} \wedge \mathbf{R}_{t+1}\mathbf{x}_{pn,t}^T)^T \end{bmatrix} \tag{3.16}$$

$$\mathbf{A}_s = \begin{bmatrix} 0 & -z_{s0,t+1} & y_{s0,t+1} \\ z_{s0,t+1} & 0 & -x_{s0,t+1} \\ -y_{s0,t+1} & x_{s0,t+1} & 0 \\ \vdots & \vdots & \vdots \\ 0 & -z_{sm,t+1} & y_{sm,t+1} \\ z_{sm,t+1} & 0 & -x_{sm,t+1} \\ -y_{sm,t+1} & x_{sm,t+1} & 0 \end{bmatrix} \tag{3.17}$$

$$\mathbf{B}_s = \begin{bmatrix} -d_{s0}(\mathbf{x}_{s0,t+1} \wedge \mathbf{R}_{t+1}\mathbf{x}_{s0,t}^T)^T \\ \vdots \\ -d_{sm}(\mathbf{x}_{sm,t+1} \wedge \mathbf{R}_{t+1}\mathbf{x}_{sm,t}^T)^T \end{bmatrix} \tag{3.18}$$

$$\mathbf{A}_h = \begin{bmatrix} \mathbf{A}_s \\ \mathbf{A}_p \end{bmatrix} \tag{3.19}$$

$$\mathbf{B}_h = \begin{bmatrix} \mathbf{B}_s \\ \mathbf{B}_p \end{bmatrix} \tag{3.20}$$

3.4.3 Mouvement à partir des points de l'environnement

Après avoir déterminé la translation à partir des points localisés sur le sol, dont la profondeur est connue ; nous proposons de déterminer la translation $\mathbf{t}_{t+1}$ à partir d'un minimum de deux points localisés dans l'environnement dont la profondeur est inconnue. Cette méthode présentée dans [Bazin 2010] est représentée de façon générale par l'équation 3.21 (voir fig. 3.12), et pour plus de deux points, la translation est déterminée par SVD.

- Dans la vue fisheye, les points observés dans l'environnement forment $\mathbf{C}_s$.
- Dans la vue perspective, ces points observés forment $\mathbf{C}_p$.
- À partir des vues hybrides on obtient $\mathbf{C}_h$ (eq. 3.22) en combinant $\mathbf{C}_s$ et $\mathbf{C}_p$.

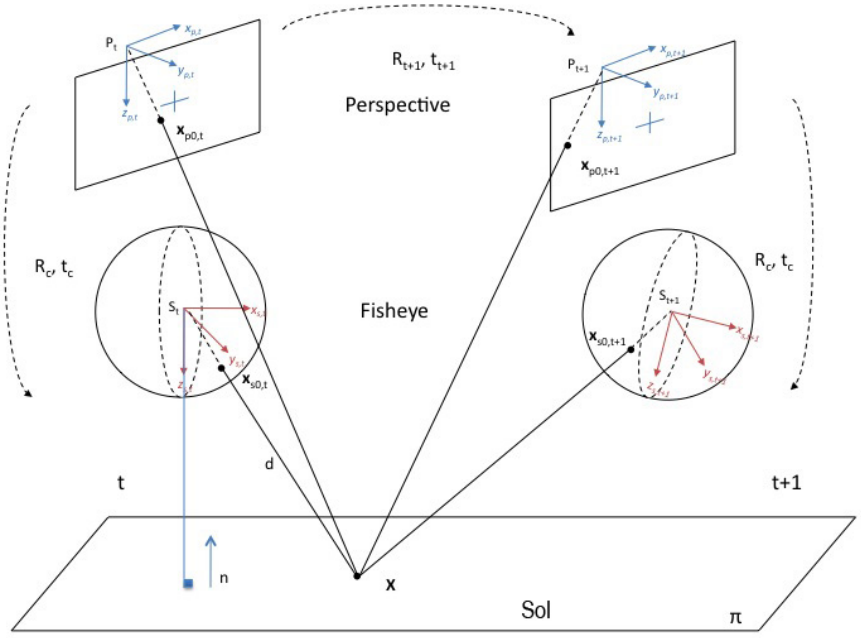


FIGURE 3.12 – Points 3D et mouvement.

$$\mathbf{t}_{t+1} \cdot (\mathbf{x}_{t+1} \wedge \mathbf{R}_{t+1}\mathbf{x}_t) = 0 \tag{3.21}$$

$$\mathbf{C}_h = \begin{bmatrix} \mathbf{C}_s \\ \mathbf{C}_p \end{bmatrix} \tag{3.22}$$

3.4.4 Fusion des points 3D issus des vues mixtes

En combinant les deux méthodes précédentes, il est possible de déterminer la translation à la fois à l'échelle métrique et de façon robuste.
La première méthode, basée sur les points du sol, est définie par l'équation 3.23.

$\mathbf{A}_h$ et $\mathbf{B}_h$ (eq. 3.23) représentent les points situés sur le sol et obervés par le système hybride.
La seconde méthode, basée sur les points de l'environnement, est définie par l'équation 3.25. $\mathbf{C}_h$ définie par l'équation 3.24, représente les points de l'environnement observés par le système hybride.
En fusionnant les matrices $\mathbf{A}_h$ et $\mathbf{C}_h$ en $\mathbf{A}_f$, $\mathbf{B}_h$ et 0 en $\mathbf{B}_f$ on obtient l'équation 3.26 résolue par la méthode des moindres carrés. Ces matrices $\mathbf{A}_f$ et $\mathbf{B}_f$ permettent d'obtenir la translation à partir de l'ensemble des points localisés dans l'environnement et sur le sol à partir des vues hybrides.

$$\mathbf{A}_h\mathbf{t}_{t+1} = \mathbf{B}_h \tag{3.23}$$

$$\mathbf{C}_h = \begin{bmatrix} \mathbf{R}_{t+1}\mathbf{x}_{s0,t} \wedge \mathbf{x}_{s0,t+1} \\ \vdots \\ \mathbf{R}_{t+1}\mathbf{x}_{sm,t} \wedge \mathbf{x}_{sm,t+1} \\ \mathbf{R}_{t+1}\mathbf{x}_{p0,t} \wedge \mathbf{x}_{p0,t+1} \\ \vdots \\ \mathbf{R}_{t+1}\mathbf{x}_{pn,t} \wedge \mathbf{x}_{pn,t+1} \end{bmatrix} \tag{3.24}$$

$$\mathbf{C}_h\mathbf{t}_{t+1} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}_{m+n,1} \tag{3.25}$$

$$\begin{bmatrix} \mathbf{A}_h \\ \mathbf{C}_h \end{bmatrix} \mathbf{t}_{t+1} = \begin{bmatrix} \mathbf{B}_h \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{3.26}$$

3.4.5 Filtre de Kalman

Le mouvement estimé présente parfois des discontinuités et des variations brutales. Ces variations sont dues à des perturbations dans l'acquisition des images. De plus, en cas de vol stationnaire, on se retrouve dans un cas singulier où le mouvement estimé est aberrant. Il est proposé un filtre de Kalman linéaire [Kalman 1960] dans le but de lisser la trajectoire et de prédire la translation si aucun mouvement n'a été estimé. L'état considéré correspond au vecteur de translation du mouvement i.e. $\mathbf{x}_k = \mathbf{t}_{t+1}$, modélisé par un système linéaire Gaussien donné par l'équation 3.27 :

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{E}\mathbf{x}_k + \mathbf{v}_k \\ \mathbf{y}_k &= \mathbf{O}\mathbf{x}_k + \mathbf{w}_k \end{aligned} \tag{3.27}$$

où $\mathbf{E} = \mathbf{I}_3$ est la matrice de transition d'état et $\mathbf{O} = \mathbf{I}_3$ est le modèle d'observation, avec comme hypothèse une vitesse constante durant un intervalle échantillonné. Les vecteurs $\mathbf{v}_k$ et $\mathbf{w}_k$ correspondent respectivement à l'erreur du modèle et à l'erreur d'observation. Ces bruits sont supposés additifs, blancs Gaussiens de moyenne nulle et les matrices de covariance sont pré-définies. Dans le but de garder la dynamique des mesures, une plus grande incertitude est attribuée au modèle i.e. 10^6 qu'aux mesures 10^2. A partir de ces hypothèses, le filtre de Kalman consiste à prédire la translation $\mathbf{t}_{t+1}$. Il permet également d'obtenir une estimation lissée en utilisant une étape de mise à jour lorsqu'une observation est disponible.

3.4.6 Algorithme

Estimer le mouvement à partir des méthodes proposées consiste tout d'abord à suivre des points par la méthode de KLT (*Kanade-Lucas-Tomasi Feature Tracker*) proposée et implémentée par [Bouguet 2000] dans chacune des vues. En parallèle, la rotation est mise à jour à partir de l'une des méthodes précédente ou par centrale inertielle. À partir de ces informations, un Ransac adapté consiste ensuite pour la première classe de point appartenant au sol, à effectuer un tri afin d'écarter les points aberrants [Malis 2005]. À partir de ce résultat, un tri est effectué pour écarter les mauvais points de la deuxième classe de points dont la profondeur est inconnue. Puis, si un nombre suffisant de points est sélectionné, les deux classes de points sont fusionnées pour déterminer le mouvement à l'échelle métrique et de façon robuste au bruit. Ce processus est répété jusqu'à obtenir une certaine erreur ou on nombre maximum d'itérations fixés dans l'algorithme. Les algorithmes utilisés sont présentées en annexes (2, 3, 4).

La translation estimée est ensuite filtrée par le filtre de Kalman.

Enfin, au fur et à mesure des translations déterminées, chacune des positions est exprimée relativement à la caméra à l'instant $t = 0$. Le repère de la caméra $t = 0$ représenté par $\mathcal{R}_0$ est exprimé dans le repère caméra $t = 1$ par l'équation 3.28 noté ${}^1\mathcal{R}_0$, ${}^2\mathcal{R}_0$ pour $t = 2$,... et de manière générale ${}^t\mathcal{R}_0$ (eq. 3.33). Pour exprimer un point de $\mathcal{R}_2$ dans $\mathcal{R}_0$ on note ${}^0\mathcal{R}_2$ (eq.3.30) et de manière générale ${}^0\mathcal{R}_n$ (eq. 3.34).

$$ {}^1\mathcal{R}_0 = \mathbf{R}_1\mathcal{R}_0 + \mathbf{t}_1 \tag{3.28} $$

$$ {}^2\mathcal{R}_0 = \mathbf{R}_2\mathcal{R}_1 + \mathbf{t}_2 \Leftrightarrow \mathcal{R}_2 = \mathbf{R}_2\mathbf{R}_1\mathcal{R}_0 + \mathbf{R}_2\mathbf{t}_1 + \mathbf{t}_2 \tag{3.29} $$

$$ \mathbf{R}_1^{-1}\mathbf{R}_2^{-1}(\mathcal{R}_2 - \mathbf{t}_2 - \mathbf{R}_2\mathbf{t}_1) \tag{3.30} $$

$$ \mathbf{R}_1^{-1}\mathbf{R}_2^{-1}(\mathcal{R}_2 - \mathbf{t}_2 - \mathbf{R}_2\mathbf{t}_1) \tag{3.31} $$

$$^{3}\mathcal{R}_0 = \mathbf{R}_3\mathcal{R}_2 + \mathbf{t}_3 \Leftrightarrow \mathcal{R}_3 = \mathbf{R}_3\mathbf{R}_2\mathbf{R}_1\mathcal{R}_0 + \mathbf{R}_3\mathbf{R}_2\mathbf{t}_1 + \mathbf{R}_3\mathbf{t}_2 + \mathbf{t}_3 \tag{3.32}$$

$$^{n}\mathcal{R}_0 = \mathbf{R}_n \dots \mathbf{R}_1\mathcal{R}_0 + \mathbf{R}_n\mathbf{R}_{n-1} \dots \mathbf{R}_2\mathbf{t}_1 + [\dots] + \mathbf{R}_n\mathbf{t}_{n-1} + \mathbf{t}_n \tag{3.33}$$

$$^{0}\mathcal{R}_n = \mathbf{R}_1^{-1} \dots \mathbf{R}_n^{-1}(\mathcal{R}_n - \mathbf{t}_n - \mathbf{R}_n\mathbf{t}_{n-1} - [\dots] - \mathbf{R}_n\mathbf{t}_{n-1} \dots \mathbf{R}_2\mathbf{t}_1) \tag{3.34}$$

La méthode proposée permet de déterminer la translation du mouvement à l'échelle métrique et de façon robuste grâce à l'exploitation des points localisés sur le sol et dans l'environnement. Cette démarche est applicable à une ou plusieurs caméras d'un système hybride ou non. Enfin, l'algorithme proposé ouvre la perspective d'une multitude d'applications et de différentes configurations possibles.

3.5 Conclusion

Ce chapitre présente des méthodes basées sur la vision capables de déterminer l'altitude, la segmentation du plan, l'attitude et le mouvement d'un drone à partir d'un capteur de stéréo hybride. Un aspect collaboratif du capteur est proposé : la vue fisheye est capable d'estimer l'attitude et l'orientation du drone. Cette attitude est ensuite utilisée pour estimer l'altitude ainsi que pour segmenter le plan. L'ensemble de ces informations extraites est ensuite propagé pour l'estimation du mouvement découplé en rotation (connue par l'attitude) et en translation. La translation peut être estimée dans chacune des vues à partir de deux types de points : les points situés sur le sol et les points localisés dans l'environnement. Les translations estimées dans chacune des vues sont ensuite fusionnées pour déterminer le mouvement global du drone. Ce mouvement est ensuite filtré par un filtre de Kalman. Enfin, l'ensemble des méthodes proposées est basé sur un capteur collaboratif évitant toute mise en correspondance entre les vues hybrides. Les interactions entre les vues peuvent être considérées de trois manières.

- Un aspect séparé. La caméra fisheye permet de déterminer, seule, l'attitude et l'orientation du drone.
- Un aspect stéréoscopique sans mise en correspondance. Il est mis en évidence par le *plane-sweeping*. Les informations issues des deux capteurs permettent de déterminer la profondeur du plan du sol observé et de segmenter les objets extérieurs à ce plan.

– Un aspect coopératif. L'estimation du mouvement est réalisé en mode séparé, l'aspect stéréoscopique du capteur est exploité pour fournir l'information métrique. La coopération a lieu lors de la fusion des points suivis dans chacune des vues mais également lors de l'exploitation des paramètres issus des étapes réalisées en amont.

Le chapitre suivant présente les résultats obtenus avec les méthodes proposées dans les chapitres 2, 3 et proposent de valider les hypothèses avancées. Ces résultats concernent la calibration, l'attitude, l'altitude, la segmentation du sol et le mouvement.

CHAPITRE 4

Résultats

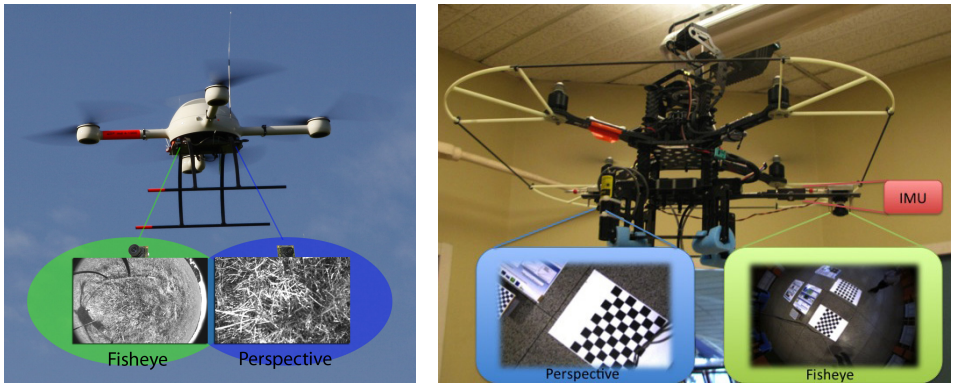
Sommaire

4.1 Étalonnage 74
4.1.1 Comparaison avec les méthodes existantes 75
4.1.2 Résultats pour un banc stéréo hybride 75
4.1.3 Performances 76
4.2 Attitude 77
4.2.1 Estimation d'attitude à partir des points de fuite 77
4.2.2 Estimation d'attitude basée sur la ligne d'horizon 78
4.2.3 Influence de l'attitude sur l'altitude 79
4.3 Altitude 80
4.3.1 Comparaison avec une vérité terrain 81
4.3.2 Résultats sur drone 83
4.3.3 Performances 84
4.4 Estimation du mouvement 85
4.4.1 Résultats de simulation 85
4.4.2 Comparaison avec une vérité terrain 87
4.4.3 Résultats sur drone 89
4.5 Conclusion 90

Ce nouveau chapitre présente les résultats des méthodes abordées aux chapitres 2 et 3.
Les expériences présentées sont réalisées sur deux types de drones : le MD4-1000 (fig. 4.1(a)) de la société Microdrones et le Pellican (fig. 4.1(b)) de la société Ascending Technologies. Chaque drone dispose du même équipement, composé d'une caméra perspective, d'une caméra fisheye et d'une centrale inertielle (fig. 4.1(a) et 4.1(b)). Les expériences sont réalisées en milieu intérieur et en environnement naturel. Les vols en environnement urbain étant actuellement interdits par la législation, il n'a pas été possible de tester cette situation.

Ce chapitre s'articule suivant quatre parties. Dans la première partie, la comparaison de la méthode de calibration aux méthodes existantes dans la littérature permet de valider l'approche proposée dans le logiciel de calibration ([Caron 2011]).

Dans la deuxième partie, les méthodes d'estimation d'attitude citées dans le chapitre 3 sont appliquées à la vue fisheye. Néanmoins, il est nécessaire de vérifier que les résultats d'estimation d'altitude par la méthode de *plane-sweeping* ne sont pas biaisés par l'attitude obtenue par vision. Dans la troisième partie, l'objectif est de vérifier la validité des résultats d'estimation de l'altitude par comparaison à une vérité terrain. Enfin, dans une quatrième partie, la méthode d'estimation de translation est comparée à une vérité terrain puis embarquée sur drone pour déterminer son mouvement.



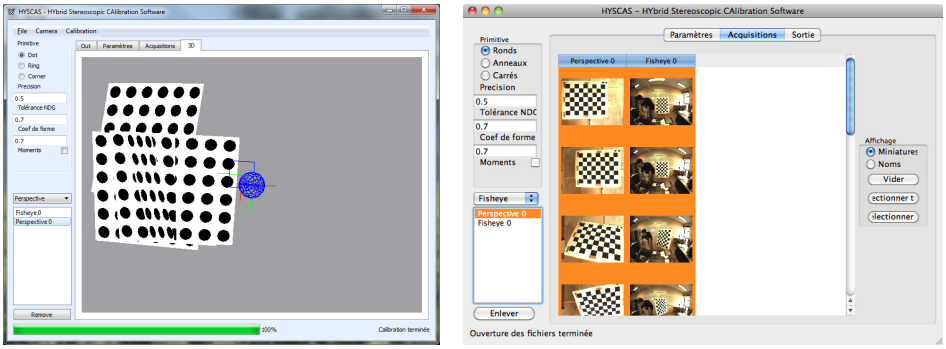
(a) (b)

FIGURE 4.1 – (a) Drone MD4-1000 avec système embarqué. (b) Pellican avec système embarqué.

4.1 Étalonnage

L'étalonnage ou calibrage est une étape importante dans l'estimation des paramètres intrinsèques et extrinsèques du système de stéréovision. La précision des résultats ainsi obtenus influe directement sur l'erreur des paramètres d'état du drone estimés par vision.

Les travaux de calibrage issus du VVS et adaptés au calibrage stéréo hybride ont abouti à l'élaboration d'un logiciel libre. Ce logiciel, nommé Hyscas, est mis à la disposition de la communauté scientifique. Il est multi-plateforme, disponible sous Windows fig. 4.2(a), Mac fig. 4.2(b), Linux, et est développé en C++ à l'aide de la librairie Qt. Il est capable d'estimer simultanément les paramètres intrinsèques et extrinsèques de caméras par différents modèles hybrides. Ces modèles sont les modèles perspectif, sphérique unitaire et paraboloïde.



(a) (b)

FIGURE 4.2 – (a) Version windows avec reprojection des poses en 3D. (b) Version mac avec miniatures des images des mires.

4.1.1 Comparaison avec les méthodes existantes

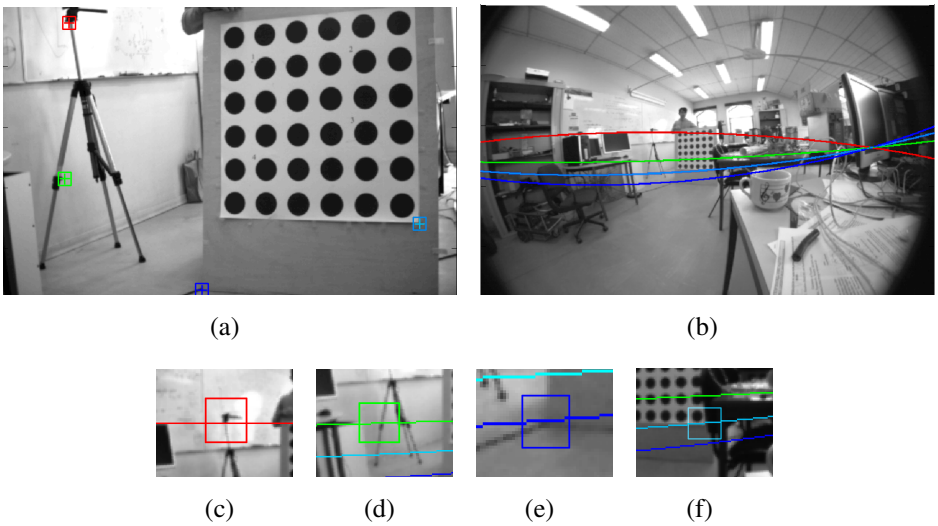
Afin de valider la méthode et le logiciel présentés, un banc stéréo composé de deux caméras perspectives d'une part, et d'une caméra omnidirectionnelle seule d'autre part, seront successivement calibrées. Ces résultats seront respectivement comparés aux outils de [Bouguet 2010] et de [Mei 2011]. Le premier outil permet d'étalonner une ou deux vues perspectives et le deuxième, développé sur la base du premier, est capable d'étalonner une caméra par le modèle sphérique.

D'après notre connaissance de la littérature, les méthodes d'étalonnage libres existantes permettent de calibrer des bancs stéréo composés de caméras c_1 et c_2 de même type. Pour deux caméras perspectives, les points issus des poses des mires observées dans chacune des vues sont utilisés par la méthode de [Bouguet 2010] et Hyscas. Les résultats obtenus sont semblables.
Ensuite, la même procédure est appliquée à une caméra omnidirectionnelle. Les points obtenus sont utilisés pour calibrer la caméra avec les méthodes de [Mei 2011] et Hyscas. Les résultats d'estimations sont identiques pour les deux outils avec une convergence plus rapide en faveur du logiciel Hyscas.
Dans cette partie, la méthode comparée aux outils existants semble être validée pour un système composé d'une ou de deux caméras.

4.1.2 Résultats pour un banc stéréo hybride

Après avoir comparé l'outil proposé à des méthodes existantes d'étalonnage, nous proposons d'évaluer la précision d'étalonnage directement dans les images. Cela permet de vérifier la précision des paramètres d'étalonnage. Pour ce faire, des points de l'environnement sont sélectionnés dans l'image perspective (fig. 4.3(a)).

Leurs coniques épipolaires sont obtenues à partir de la matrice essentielle calculée grâce aux résultats d'étalonnage. Ces coniques sont tracées sur la vue fisheye (fig. 4.3(b)). Les points d'intérêt sélectionnés sont grossis (fig. 4.3) et montrent la précision mais aussi la crédibilité des paramètres d'étalonnage.



(a) (b)

(c) (d) (e) (f)

FIGURE 4.3 – Des points d'intérêt sont sélectionnés dans une image perspective (a) et les courbes épipolaires correspondantes sont calculées et tracées dans l'image fisheye (b). (c), (d), (e) et (f) montrent plus précisément la partie de conique épipolaire correspondant aux points sélectionnés dans l'image perspective.

4.1.3 Performances

En utilisant Hyscas, l'étalonnage d'une caméra perspective à partir de 6 images s'effectue en 244 ms pour le calcul des paramètres intrinsèques (tab. 4.1). Ces résultats démontrent la rapidité du logiciel et offreny une possibilité de calibrage temps réel pour des applications futures et embarquées.

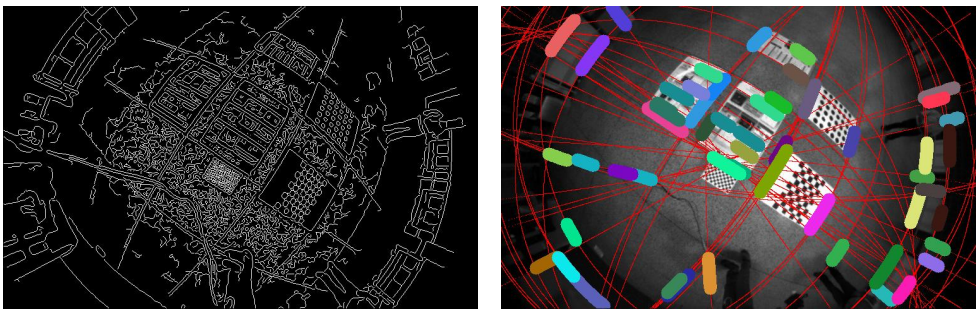
nombre de poses	perspective	fisheye	hybride
4	109 ms	115 ms	681 ms
6	244 ms	253 ms	1467 ms
8	469 ms	492 ms	2786 ms

TABLE 4.1 – Temps de calcul pour l'étalonnage en utilisant le logiciel Hyscas. Chaque pose est une image pour une caméra, deux images pour deux caméras, etc. Trente-six points sont utilisés pour chaque image.

4.2 Attitude

L'attitude peut être estimée soit par vision, soit par centrale inertielle. En utilisant la vision, deux méthodes sont proposées : une méthode adaptée au milieu urbain ou intérieur et une méthode adaptée au milieu naturel. La première consiste à utiliser les droites présentes dans l'image tandis que la seconde se base sur la projection de la ligne d'horizon sur la vue omnidirectionnelle. Les résultats obtenus sont comparés à une centrale inertielle, considérée comme vérité terrain.

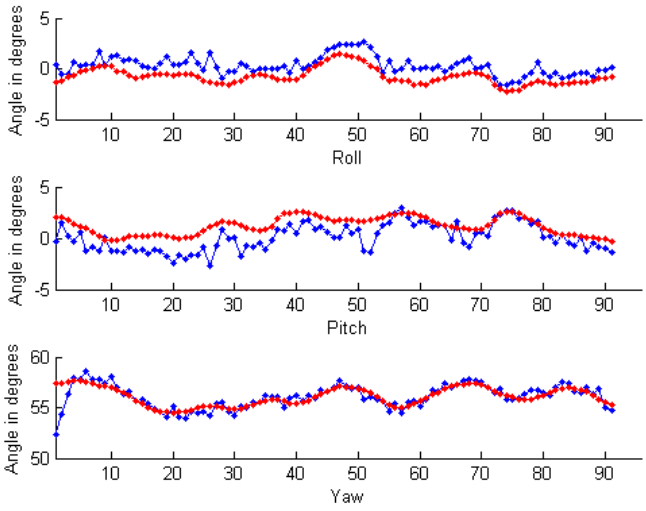
4.2.1 Estimation d'attitude à partir des points de fuite



(a) (b)

FIGURE 4.4 – (a) Détection des contours pour la détection de droites. (b) Estimation des droites.

En environnement intérieur, des expériences sont réalisées avec un quadri-rotor Pellican équipé du système hybride. Nous proposons de comparer l'attitude estimée par vision à celle estimée par centrale inertielle. Pour ce faire, la rotation entre la centrale inertielle et la vue fisheye est tout d'abord déterminée par calibrage à partir de la méthode [Lobo 2007] et de l'outil lié. Ces résultats sont illustrés par la figure 4.5(a) et le tableau 4.2 où la ligne bleue représente la méthode basée sur les points de fuite et la ligne rouge représente les valeurs issues de la centrale inertielle. Il est important de noter que l'estimation issue de la centrale inertielle est filtrée par un Kalman intégré à la centrale.
Pour le lacet, le décalage entre le nord magnétique de la centrale et le repère absolu de la caméra est centré par la formule suivante : $y_{i,vp} = y_{i,vp} - \bar{y}_{vp} + \bar{y}_{imu}$, avec $y_{i,vp}$ la i^{eme} valeur estimée par point de fuite et $\bar{y}_{imu}$ la moyenne des angles de lacet estimés par centrale inertielle. Ainsi, l'erreur d'attitude ε_i est définie par la différence entre la valeur obtenue par centrale inertielle et celle par vision de la i^{eme} valeur. Il est d'ailleurs nécessaire de préciser que dans le tableau 4.2 l'erreur du lacet n'est pas représentée à cause de la moyenne centrée précédemment.



(a)

FIGURE 4.5 – Comparaison, de haut en bas, du roulis, tangage et lacet par vision (bleu) et par centrale inertielle (rouge).

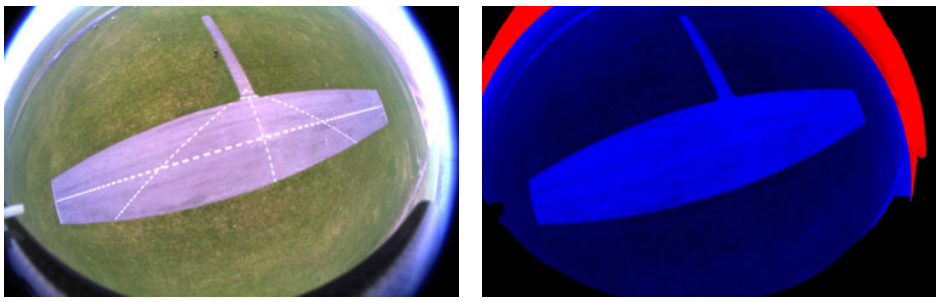
Angle	Roulis	Tangage
$\max(\varepsilon_i)$	3.97°	2.43°
$\overline{\lvert\varepsilon_i\rvert}$	1.32°	1.96°
σ_ε	1.65°	2.07°

TABLE 4.2 – Comparaison de l'attitude entre une centrale inertielle et la méthode basée sur la vision.

Enfin, l'erreur maximale introduite par la méthode de vision est inférieure à 5° (tab. 4.2) et l'erreur moyenne est inférieure à 2°. Ces résultats montrent la précision et la validité de l'approche. Cette méthode, testée sur lentille fisheye est donc capable d'estimer l'attitude de manière aussi efficace qu'un miroir catadioptrique. De plus, dans les travaux présentés, l'attitude est déterminée en temps réel (environ 30 Hz).

4.2.2 Estimation d'attitude basée sur la ligne d'horizon

En environnement naturel, nous proposons de présenter les résultats de la méthode exploitant l'information issue de la ligne d'horizon. Des tests ont été réalisés sur une vue catadioptrique embarquée sur ULM (*Ultra-léger motorisé*) et une implémentation en C++ de la méthode a permis d'estimer l'attitude sur une



(a) (b)

FIGURE 4.6 – (a) Vue fisheye du drone. (b) Segmentation sol/air à partir de la ligne d'horizon.

séquence de plus de 10000 images.
Ce programme a ensuite été adapté à une vue fisheye. Contrairement à une vue catadioptrique, la vue fisheye possède un champ de vue plus faible. Il est donc nécessaire d'utiliser un fisheye d'angle de vue de 180° pour observer la ligne d'horizon à chaque instant. Ces résultats obtenus sont comparés à une centrale inertielle. Pour ce faire, la rotation entre la vue fisheye et la centrale inertielle est étalonnée par la méthode de [Lobo 2007].
Enfin, pour l'image représentée figure 4.6(a), l'erreur est de 0.92°. Cette erreur correspond à l'angle formé entre la normale au sol estimée par centrale inertielle et celle déterminée par vision. La figure 4.6(b) représente la segmentation ciel/sol après la détection de la ligne d'horizon. De plus, l'algorithme en temps réel adapté en C++ permet d'obtenir une fréquence de mise à jour de 25 Hz. Cependant, une version améliorée de l'algorithme, basée sur le suivi des points contours permet d'obtenir une fréquence de mise à jour de 100 Hz.

Les méthodes présentées permettent de déterminer l'attitude mais également la normale au sol utilisée pour l'estimation d'altitude. C'est pourquoi, la partie suivante propose d'étudier l'influence de l'erreur d'attitude sur la précision de l'altitude estimée. Les erreurs maximales introduites dans ces tests sont issues de conditions réelles de vol à partir de l'attitude déterminée par vision.

4.2.3 Influence de l'attitude sur l'altitude

Cette partie propose d'étudier l'influence de l'attitude sur l'estimation de l'altitude. ‘

Lors d'un vol en milieu intérieur, les erreurs moyennes et maximales sont présentées dans le tableau 4.2. Ces résultats montrent une erreur maximale de 5° pour l'estimation d'attitude par vision. A partir de ces données extrêmes, des simula-

tions de 1000 itérations sont réalisées à partir d'une paire d'images acquises à une altitude de 1839 mm. Cette altitude est représentative de l'altitude moyenne de vol d'un drone en particulier dans les phases d'approche :

- dans la figure 4.7, l'influence du roulis est représentée en bleu : une erreur de 5° entraîne une erreur d'estimation d'altitude de 0.87%,
- la courbe rouge représente l'influence du tangage : une erreur de 5° entraîne 0.39% d'erreur sur l'altitude,
- le roulis et le tangage sont bruités indépendamment avec une erreur de 5°. L'erreur d'altitude engendrée est alors inférieure à 1%.

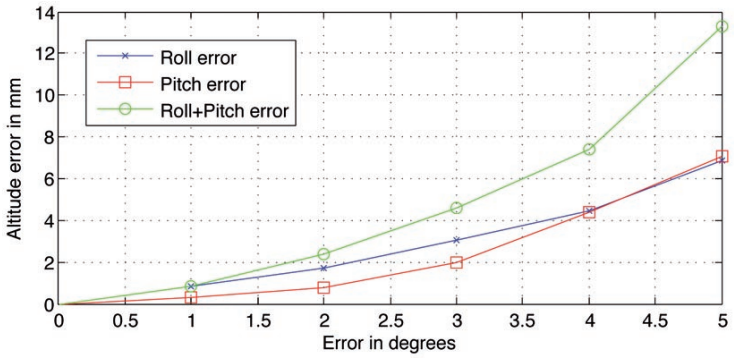


FIGURE 4.7 – Influence de l'erreur d'attitude sur l'estimation de l'altitude.

L'algorithme d'estimation d'altitude est donc peu sensible aux faibles erreurs d'attitude. Estimer l'attitude par vision est une piste possible pour la navigation des drones dans la mesure où l'erreur générée influence faiblement les algorithmes basés sur ces méthodes.

4.3 Altitude

Les méthodes précédentes montrent la validité des approches basées sur la vision. Ces méthodes permettent de déterminer l'attitude et donc la normale au sol. Cette normale et les paramètres de calibrage sont utilisés pour déterminer l'altitude.
Dans un premier temps, cette altitude calculée par *plane-sweeping*, est comparée à un télémètre laser. Ce dispositif est considéré comme une vérité terrain.
Dans un deuxième temps, des tests embarqués sur drone permettent de valider l'approche.
Dans un troisième temps, les performances de l'algorithme sont détaillées à la fois pour la méthode GPU et la méthode CPU.
Enfin, l'utilisation de la formulation suivante : $\varepsilon_d = \frac{(\widehat{d}-d_{gt})}{d_{gt}}$, avec d_{gt} la vérité ter-

rain permet de quantifier l'erreur ε_d entre l'altitude réelle et l'altitude estimée par *plane-sweeping*.

4.3.1 Comparaison avec une vérité terrain

La première expérience est réalisée dans un environnement clos avec un éclairage constant et maîtrisé. Le sol observé présente une texture importante. Puis, à chaque étape, un télémètre laser détermine la vérité terrain de l'altitude du banc de stéréovision et la centrale inertielle détermine la normale du sol. Enfin, sur la figure 4.8, l'altitude estimée présente une erreur moyenne de 2.41% pour un intervalle situé entre 500 mm et 2000 mm.

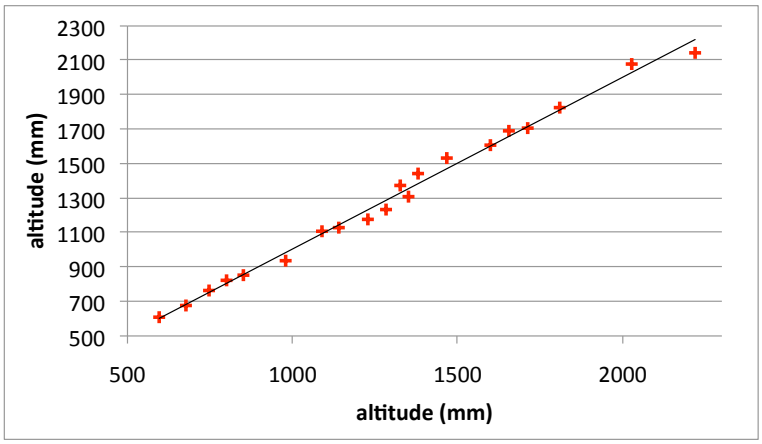


FIGURE 4.8 – Les croix représentent l'altitude estimée par *plane-sweeping* et la ligne représente les mesures idéales issues du télémètre laser.

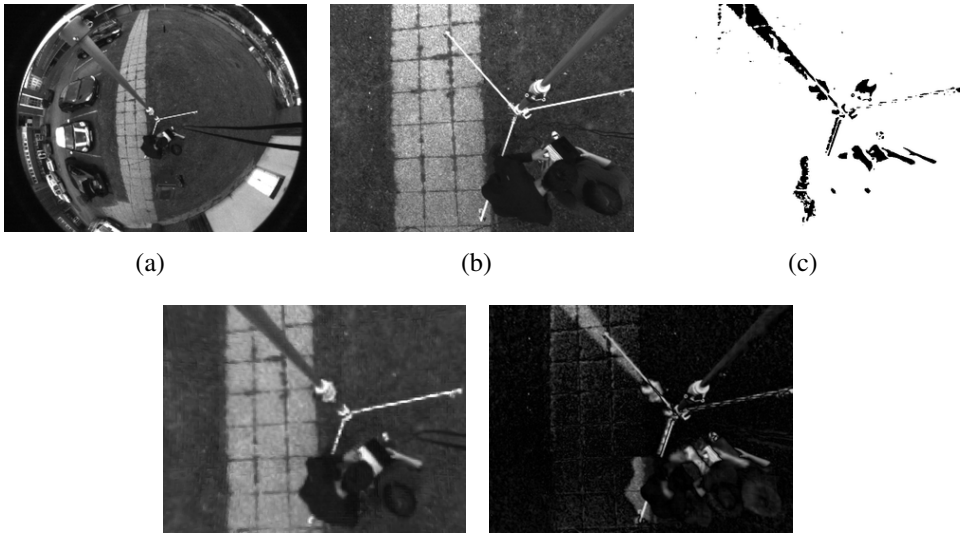
L'expérience suivante est réalisée en extérieur. Le sol observé se compose de gazon et de dalles texturées avec un éclairage naturel. Pour cette expérience, deux caméras séparées d'un empattement de 447mm sont montées sur un mat télescopique. Elle a pour objectif d'analyser le comportement de l'algorithme à des altitudes plus importantes : l'intervalle de travail se situe entre 2000 mm et 5200 mm.
L'estimation de l'altitude et la segmentation du plan du sol sont ensuite réalisées hors-ligne sur un GPU (algorithme 1). L'altitude est estimée précisément sur sol sans obstacle (tab. 4.3) avec une erreur située entre 0.18% et 3.14%. Dans le cas d'obstacles sur le sol (tab. 4.3), l'erreur observée semble plus importante, entre 7.52% et 8.82% comparée à la vérité terrain.

La segmentation du sol est correctement déterminée sur des surfaces contrastées. Par exemple, dans le cas d'un sol sans obstacles, le mat pneumatique, sur

Type	Vérité terrain	Altitude estimée	Erreur
Sol libre	2187mm	2200mm	0.59%
Sol libre	3244mm	3250mm	0.18%
Sol + obstacles (peu contrasté)	3244mm	3488mm	7.52%
Sol libre	4072mm	4200mm	3.14%
Sol libre	5076mm	5202mm	2.48%
Sol + obstacles	4080mm	4440mm	8.82%

TABLE 4.3 – Altitude et segmentation du sol avec et sans obstacles - Paramètres de l'algorithme : $s = 6$, $thres = 25$.

lequel les caméras sont fixées, est représenté par une zone aberrante (*outlier*, en noir) (fig. 4.9(c)). En revanche, pour une image composée d'un plan du sol dominant et de murs, le sol est segmenté en blanc et le mur en noir (pixels aberrants).



(a) (b) (c)

(d) (e)

FIGURE 4.9 – Altitude et segmentation du sol - 4.8% d'*inliers* - (a) Vue fisheye, (b) Vue perspective, (c) Segmentation du sol, (d) Homographie sphère/plane, (e) Comparaison vue de référence et homographie.

4.3.2 Résultats sur drone

Les résultats, validés sur mat pneumatique, et comparés à une vérité terrain sont ensuite testés sur un quadri-rotor Pelican. Deux caméras miniatures (uEye) d'empattement de 314mm sont embarquées sur un drone compact avec une centrale inertielle. La centrale détermine l'attitude du système. Les caméras reliées à un ordinateur externe par câble permettent de réaliser l'estimation en ligne (fig. 4.10(f)). L'estimation de l'altitude est réalisée par CPU (Méthode 3.3.4).

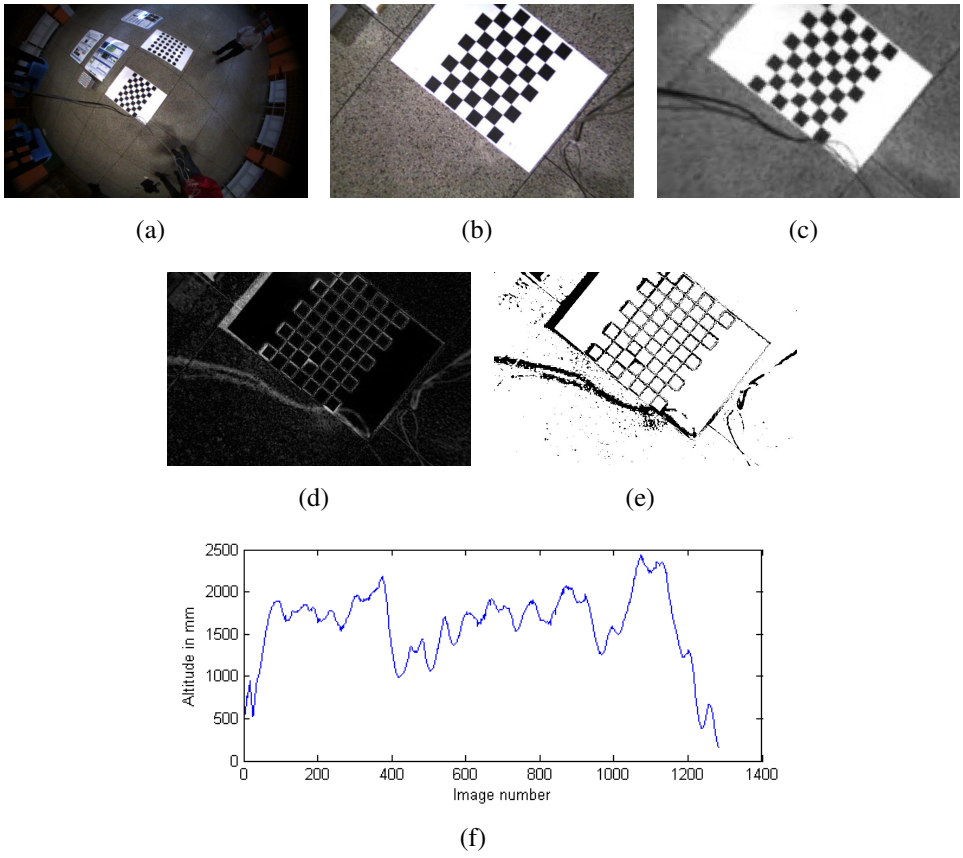


FIGURE 4.10 – (a,b) Vue embarquée du Pelican (de gauche à droite : la vue fisheye et perspective) - Altitude estimée : 1912mm. (c) Homographie de la vue fisheye vers la vue perspective. (d) Erreur. (e) Segmentation du sol $\mathbf{G}$. (f) Altitude de la trajectoire estimée par *plane-sweeping*.

En estimant l'altitude à partir des deux vues (fig. 4.10(a),4.10(b)) et de l'homographie (fig. 4.10(c)), le plan du sol (fig. 4.10(e)) est segmenté en utilisant l'erreur (fig. 4.10(d)) et devient le masque $\mathbf{G}$. La figure 4.10(e) montre la bonne segmentation d'objets planaires (mire blanche) et non planaires (cables en noir) après seuillage.

Pour conclure, les résultats observés, permettent de constater que la précision de l'estimation semble proportionnelle à l'altitude. En effet, plus l'altitude augmente, plus l'erreur augmente. Le contexte initial de ce travail de recherche est l'étude de méthodes basées sur la vision afin de rendre autonome les phases de décollage et d'atterrissage d'un drone. Les résultats obtenus par cette méthode sont adaptés à la problématique initiale. En effet, les phases d'approches constituent les étapes critiques d'un vol. Lors de ces approches, l'algorithme proposé est très précis pour des altitudes inférieures à 2000 mm.

D'un point de vue segmentation du sol, l'algorithme permet de réaliser cette tâche de façon approximative. Ces résultats paraissent suffisants pour la phase d'atterrissage et pour l'estimation du mouvement. Néanmoins, les aspects les plus évidents de la méthode à améliorer sont le cas de zones faiblement texturées, les surfaces spéculaires dont la réflexion dépend de l'orientation des caméras et de la position de la source d'éclairage.

4.3.3 Performances

Dans le premier cas, l'algorithme a été développé sur GPU en brook+ pour ATI puis testé sur une carte graphique ATI 4850 avec CPU Intel E8400 3Ghz. Cette implémentation permet un calcul en temps réel de l'altitude et de la segmentation du sol (30 Hz). Cependant, cette méthode n'est utilisable qu'hors-ligne car elle nécessite une unité de traitement volumineuse. Pour l'utiliser, la consommation d'énergie et l'encombrement du système en font un système impossible à intégrer sur micro-drone. La solution est alors l'adaptation de la méthode sur un processeur plus compact et économe en énergie : un CPU.

Dans le deuxième cas, l'algorithme est intégré sur CPU après optimisation. Ce portage permet d'estimer l'altitude sans segmenter le sol. Grâce à cette implémentation, la fréquence d'actualisation des données est de 180 Hz en moyenne. Ces résultats sont au dessus des capacités de rafraichissement d'une caméra standard et permettent un traitement embarqué.

Enfin, une version embarquée a été développée et portée sur l'ARM du Gumstix Overo Fire @600Mhz puis sur la carte ATOM. Cette implémentation permet un rafraichissement de 5 Hz sur ARM et 7 Hz sur carte Atom. L'aspect temps réel paraît assez limité mais reste exploitable pour des applications embarquées à faible vitesse ascensionnelle.

En adaptant cette méthode à la fois sur GPU, CPU et carte embarquée, les résultats obtenus sont mis à jour en temps réel et utilisables pour une application de drone. Pour estimer simultanément l'altitude et la segmentation du sol, il paraît nécessaire d'utiliser un GPU sur un aéronef de plus grande capacité d'emport.

Pour l'estimation d'altitude, la méthode proposée est utilisable directement sur drone compact comme le Pelican.

4.4 Estimation du mouvement

La calibration, l'altitude, la segmentation du sol et l'attitude sont connus grâce au capteur de stéréovision hybride proposé. Ces paramètres sont indispensables pour déterminer le mouvement, découplé en une rotation et une translation. La rotation est déterminée par vision ou par centrale inertielle et la translation est estimée à partir des deux catégories de points. Les points localisés sur le sol de profondeur connue et les points localisés dans l'environnement, de profondeur inconnue.

Cette section propose de présenter les résultats d'estimation du mouvement obtenus. Une première partie propose des résultats de simulations afin de valider la robustesse au bruit de l'approche proposée. Une deuxième compare ensuite l'estimation du mouvement à une vérité terrain. Cette comparaison valide la précision de l'approche proposée. Enfin, une troisième partie présente des expériences réalisées sur drone.

4.4.1 Résultats de simulation

Pour chaque scénario, la répartition des points s'effectue selon le tableau 4.4. L'erreur introduite est un angle aléatoire appliqué sur les lignes de vue quel que soit le modèle de projection utilisé.

- L'erreur d'angle de projection est calculé entre le vecteur réel de translation et le vecteur estimé.
- L'erreur métrique correspond à la norme de la différence entre le vecteur translation réel et celui estimé.

Pour l'ensemble de ces simulations, l'erreur en entrée varie de 0° à 1° par pas de 0.02° et l'erreur de sortie est la moyenne de 1000 itérations.

Méthode 1 : Mouvement à partir des points du plan - fig. 4.11 et fig. 4.12
L'estimation du mouvement à partir des points du sol est moins précise à partir de la vue sphérique que la vue perspective. Cette constatation se vérifie pour l'erreur angulaire et pour l'erreur de translation. Dans le cas hybride, le nombre de points est doublé grâce aux deux vues, l'estimation est alors plus robuste au bruit.

Vues	Nb. points sol	Nb. points env.
Hybride - fig. 4.11	60	60
Sphere - fig. 4.11,4.12	60	0
Perspective - fig. 4.11,4.12	60	0
Hybride - fig. 4.11,4.12	60	0
Sphere - fig. 4.11,4.12	30	30
Perspective - fig. 4.11,4.12	30	30
Hybride - fig. 4.11,4.12	60	60

TABLE 4.4 – Répartition du nombre de points en fonction de leur classification et des vues.

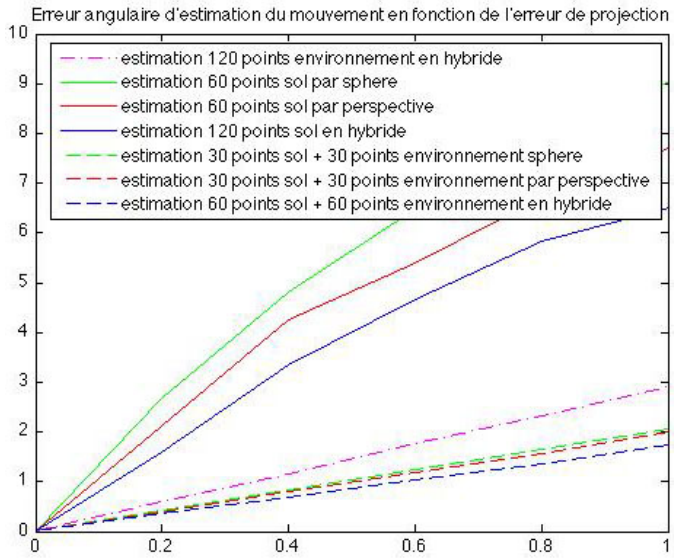


FIGURE 4.11 – Erreur angulaire entre la translation réelle (en mm) et estimée en degrés.

Méthode 2 : Mouvement à partir des points de l'environnement - fig. 4.11 La translation estimée à partir des points de l'environnement est obtenue à l'échelle près. L'analyse de l'erreur ne peut donc être réalisée qu'à partir du biais angulaire introduit par simulation (fig.4.11 - pourpre). Cette méthode est globalement plus robuste à l'erreur comparée à la méthode précédente mais n'apporte pas d'information métrique.

Méthode 3 : Fusion des points de l'environnement des vues mixtes - fig. 4.11 et fig. 4.12 Dans le dernier cas, la fusion des deux classes de points améliore de façon significative les résultats en comparaison aux méthodes

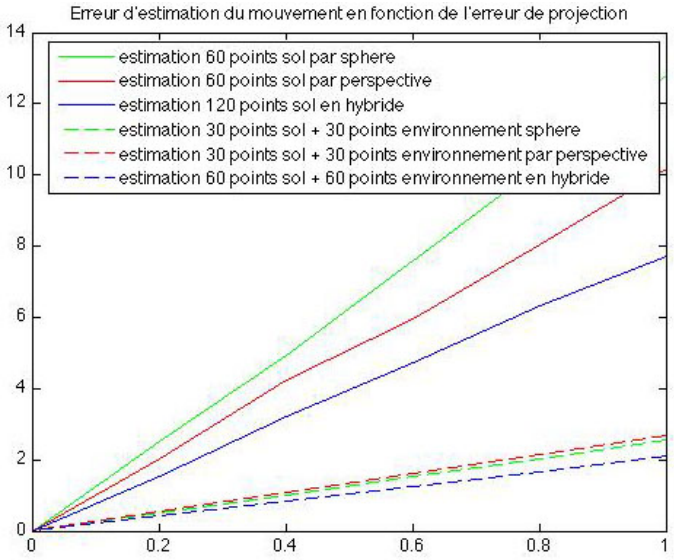


FIGURE 4.12 – Erreur de distance entre la translation réelle et estimée.

précédentes. La translation est déterminée à l'échelle métrique et la robustesse au bruit est améliorée. La fusion des points du sol avec ceux de l'environnement à partir des vues hybrides ou séparées amènent à la même conclusion. Cette méthode améliore donc la robustesse et la précision (voir fig.4.11 et fig.4.12).

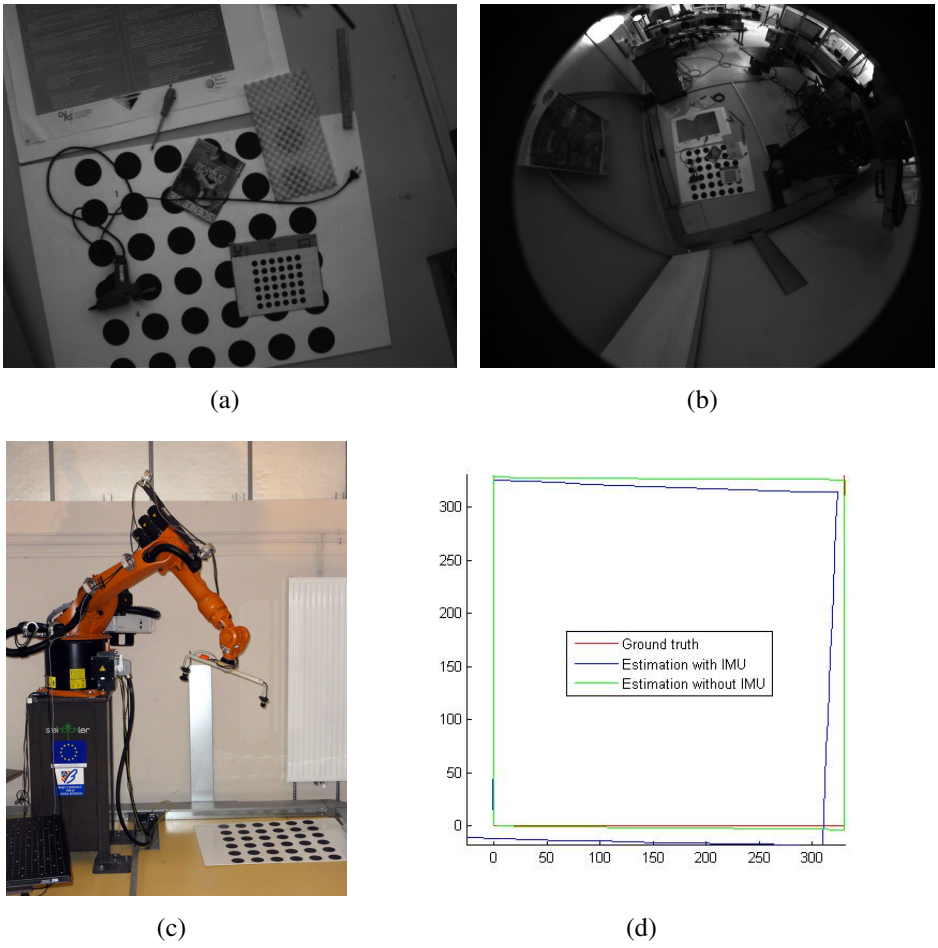
Dans le cas réel, la vue fisheye apporte une majorité de points localisés dans l'environnement. La vue perspective fournit plutôt des points localisés sur le sol. Dans la suite des expériences, nous proposons de suivre uniquement les points localisés sur le sol à partir de la vue perspective et de suivre les points de l'environnement uniquement à partir de la vue fisheye.

La méthode est ensuite validée par comparaison à un bras manipulateur effectuant une boucle de trajectoire carrée. L'estimation de trajectoire réalisée par le système du bras manipulateur est considérée comme vérité terrain.

4.4.2 Comparaison avec une vérité terrain

Le système hybride est fixé avec une centrale inertielle sur le bras d'un robot manipulateur industriel dont le système complet est illustré figure 4.13(c). La position du bras manipulateur est connue à chaque instant et la trajectoire déterminée par vision est comparée à celle issue du robot. Les images sont acquises dans un premier temps puis traitées hors ligne. Dans chacune de ces vues, le nombre de points suivis se situe entre 50 et 200. Dans la figure 4.13(d), l'algorithme est comparé à la vérité terrain, d'une précision proche du micron. La trajectoire du robot forme un carré de 300 mm de côté avec un déplacement total de 1320 mm,

exprimés dans les coordonnées du bras manipulateur.
Ensuite, l'altitude est estimée par *plane-sweeping*. À partir de cette information, le mouvement peut être déterminé et se décompose en rotation et translation. La rotation est déterminée par centrale inertielle (ligne bleue) ou à partir de la vérité terrain (ligne verte). Dans le premier cas, en bleu, l'erreur totale est de 19.084 mm en XYZ soit 1.45%. Dans le deuxième cas, en vert, l'erreur totale est de 15.865 mm en XYZ soit 1.20%.

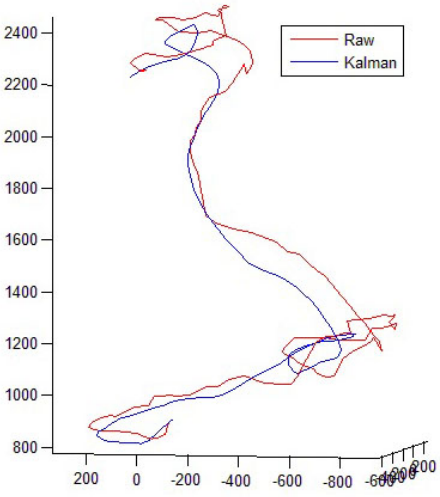


(a) (b)

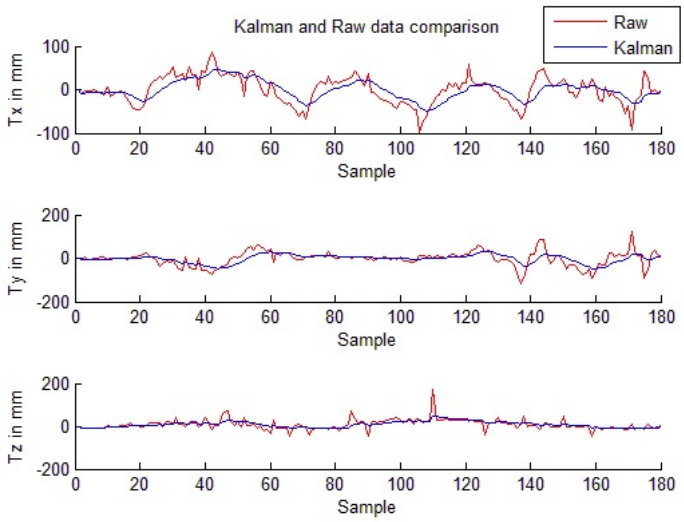
(c) (d)

FIGURE 4.13 – Boucle (d) du robot (c) à partir des vues hybrides et fusionnées, (a) vue fisheye du robot, (b) vue perspective du robot.

Ces résultats permettent de valider l'approche, appliquée sur drone dans la section suivante.



(a)



(b)

FIGURE 4.14 – (a) Trajectoire 3D d'un drone en milieu clos. (b) Données brutes et données filtrées par Kalman.

4.4.3 Résultats sur drone

Le système complet composé des deux caméras uEye et d'une centrale inertielle est testé sur un drone quadri-rotor dont les images sont traitées hors ligne. Les vols sont effectués en milieu intérieur. Dans chaque vue, le nombre de points suivis varie entre 50 et 200. La vue perspective est supposée observer le sol et la vue fisheye observer l'environnement dont la profondeur des points suivis est inconnue.

Tout d'abord, la centrale inertielle permet de déterminer l'attitude et les rotations liées aux mouvements entre deux instants. Ensuite, l'altitude, déterminée par *plane-sweeping*, permet de connaitre la profondeur des des points appartenant au sol en vue de connaître le mouvement. La figure 4.14(a) présente la trajectoire finale de ce mouvement estimé et la figure 4.14(b) la trajectoire selon t_x, t_y et t_z. Ces figures montrent en rouge les données brutes, sensibles au bruit et aux variations brutales. Enfin, en bleu, les données sont filtrées et lissées par un filtre de Kalman.

La figure 4.15 montre une deuxième expérience en milieu intérieur. Durant ce vol, pour des raisons de temps de traitement, l'attitude est estimée par vision sur une partie de la séquence et l'autre partie par centrale inertielle. Néanmoins, l'altitude est entièrement déterminée par *plane-sweeping*. Les informations ainsi extraites permettent d'obtenir le mouvement à partir du système hybride. Cette dernière expérience permet de valider l'ensemble des approches proposées et de répondre à la problématique posée initialement.

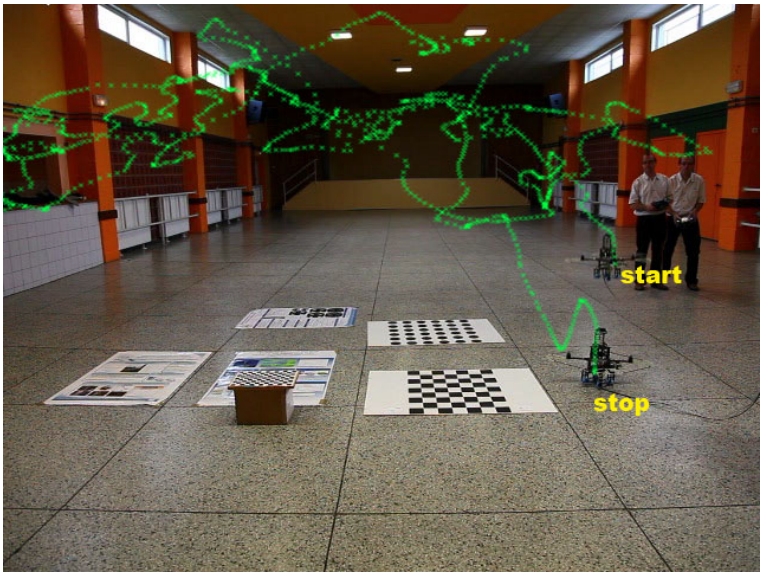


FIGURE 4.15 – Mouvement en intérieur avec le Pellican.

4.5 Conclusion

Ce chapitre a présenté les résultats issus des méthodes proposées dans le chapitre 3 et du logiciel d'étalonnage cité dans le chapitre 2.
En premier lieu, les paramètres d'étalonnage sont comparés à ceux de méthodes existantes. Ces résultats obtenus valident l'approche et les temps de traitement confirment la rapidité de l'algorithme développé. De plus, les paramètres extrinsèques et intrinsèques d'un banc stéréo calibré démontrent la précision directement

à partir des images et des lignes épipolaires.
En deuxième lieu, les résultats de ce calibrage sont ensuite utilisés pour déterminer l'attitude. Des méthodes de vision adaptées au milieu urbain ou au milieu naturel permettent d'obtenir l'attitude du drone. Une analyse de l'influence de l'erreur issue de ces paramètres sur l'estimation de l'altitude permet de vérifier que cette méthode peut être utilisée de manière collaborative avec la méthode de *plane-sweeping*.
En troisième lieu, les résultats d'estimation d'altitude montrent la précision de l'approche par *plane-sweeping*. Cet algorithme permet également de segmenter le plan pour dissocier les points appartenant au sol des points de l'environnement et déterminer la zone d'atterrissage.
En dernier lieu, l'ensemble de ces paramètres est propagé dans chacune des vues pour déterminer le mouvement. Ce mouvement est découplé en rotation et translation. La rotation est issue soit de l'une des méthodes présentées basées sur la vision soit de la centrale inertielle. La translation provient quant à elle du suivi de deux classes de points. L'avantage majeur de cette approche est de pouvoir obtenir une translation à l'échelle métrique et une mesure robuste au bruit. Un filtrage de Kalman permet de lisser cette trajectoire et de pallier les singularités de la méthode.

Les perspectives à améliorer concernent tout d'abord la détermination des paramètres de distorsions des caméras à partir de la méthode d'étalonnage. Puis, la méthode d'estimation d'attitude en milieu naturel offre des performances temps réel mais ne fournit pas l'orientation. Cet aspect serait à améliorer à partir de méthodes existantes par exemple. Ensuite, la méthode de *plane-sweeping* nécessite d'être améliorée dans le cas d'une zone observée peu texturée, comme du gazon suivi à haute altitude. Enfin, la précision de l'estimation du mouvement est directement liée aux données encapsulées. Cet aspect est à enrichir et à améliorer.

Conclusion et perspectives

Ce mémoire de thèse a présenté un nouveau capteur de stéréovision hybride composé d'une caméra fisheye et d'une caméra perspective modélisées respectivement par une sphère et un plan. À ce nouveau capteur sont associées de nouvelles méthodes de traitement d'images en temps réel et sans mise en correspondance dont l'objectif est de retrouver l'information 3D.

Dans le premier chapitre, nous avons situé notre travail au sein des nombreux projets menés autour de l'amélioration de l'autonomie de navigation des drones. Par accroître cette autonomie, une étude dynamique a d'abord été nécessaire afin de définir les paramètres à estimer. Un ensemble de capteurs sont ensuite classifiés en fonction de ces paramètres. Cependant, l'application compacte et embarqué visée a débouché sur l'étude des capteurs de vision pour laquelle un état de l'art a été réalisé. Cette revue de la littérature a mené vers un axe de recherche : l'étude d'un capteur de stéréovision hybride appuyé par une centrale inertielle.

Dans un second chapitre, l'idée de ce capteur hybride est analysée selon plusieurs axes.
Tout d'abord, ce capteur peut être représenté, d'un point de vue géométrique, par différents modèles. Notre choix s'est porté vers un modèle hybride car il semble mieux adapté aux traitements temps réels et au mode d'utilisation séparé du capteur. Néanmoins, cette modélisation a posé un nouveau problème : l'étalonnage. A notre connaissance, nous n'avons pas rencontré d'outils libres capables de déterminer les paramètres de calibrage d'un système hybride. Ainsi, pour étalonner une telle configuration, nous avons choisi de développer un nouvel outil basé sur le VVS. En effet, sa formulation est adaptée à l'étalonnage simultané des paramètres intrinsèques et extrinsèques d'un système de stéréovision hybride. Enfin, la troisième piste à explorer est matérielle. D'une part, pour répondre aux contraintes d'encombrement et de consommation énergétique, un large choix de capteurs de vision s'offrent à nous. Ce choix du couple caméra-objectif est effectué en fonction des contraintes liées l'utilisation d'un drone compact. Ainsi, cette décision s'est portée vers un capteur simple à mettre en œuvre, rapide et compact. D'autre part, l'ensemble du système est géré par une carte électronique. Cette carte est destinée à être embarquée et à transmettre ou exécuter les algorithmes proposés dans le chapitre 3.

Dans ce troisième chapitre, nous proposons des méthodes basées sur la vision qui visent à estimer les paramètres de navigation d'un drone à partir du capteur hybride proposé.
Tout d'abord, la modélisation sphérique de la vue fisheye permet d'appliquer des

méthodes d'estimation de l'attitude basées sur des vues omnidirectionnelles. Ensuite, le *plane-sweeping* permet l'estimation de l'altitude et la segmentation du plan sans mise en correspondance. Cet algorithme, adapté au modèle hybride a pu être implémenté sur différents processeurs de traitements en vue d'être traité en temps réel. Enfin, l'ensemble de ces paramètres est propagé dans chacune des deux vues afin d'estimer le mouvement. à l'échelle métrique et de façon robuste.

Le dernier chapitre présente les résultats obtenus à partir des méthodes proposées dans les chapitres 2 et 3. Ces méthodes sont comparées à une vérité terrain puis testées sur drone.
Tout d'abord, l'élaboration d'un outil d'étalonnage pour des caméras hybrides constitue une nouvelle approche. Pour la valider, nous avons proposé de comparer notre outil à ceux existants pour des configurations classiques. Les résultats sont semblables et valident l'approche. Puis, les méthodes d'estimation d'attitude sont utilisées à partir de la vue fisheye. La méthode basée sur la ligne d'horizon semble présenter certaines limites liées aux caractéristiques physiques du capteur dans certaines conditions de basse altitude. De plus, l'orientation n'est pas déterminée à partir de la ligne d'horizon. L'attitude estimée est ensuite utilisée pour le *plane-sweeping*. Il nous a semblé important d'étudier l'influence de l'erreur d'attitude sur la précision de l'altitude estimée. Les résultats sont intéressants et demandent à être confirmés sur des vols de longue durée. La méthode de *plane-sweeping* est ensuite comparée à une vérité terrain. Les résultats obtenus semblent confirmer la précision de l'approche. Cet algorithme est également testé en environnement intérieur et extérieur. Néanmoins, ces tests montrent parfois une certaine limite liée aux surfaces faiblement texturées. Enfin, pour valider la technique d'estimation de mouvement, le système de vision a été fixé sur un bras manipulateur. La trajectoire estimée par vision est comparée à celle issue du bras. Des résultats en milieu intérieur, couplés à la méthode de *plane-sweeping* et d'estimation de l'altitude ont ensuite validé l'approche sur drone. Les points de départ et d'arrivée sont cohérents par rapport à la réalité terrain.

La problématique posée initialement est l'estimation des paramètres de vol d'un drone compact durant les phases critiques de décollage et d'atterrissage. Pour répondre à cette problématique et aux contraintes liées, l'hypothèse choisie dans cette thèse est le développement d'un capteur de vision hybride omnidirectionnel et perspectif capable de fournir des informations sur l'altitude, l'attitude, le déplacement et les vitesses de déplacement.

Les travaux présentés s'appuient sur des méthodes d'estimation de l'attitude développées au sein du laboratoire MIS. Adapté à un milieu urbain ou naturel, cette méthode permet à un drone de connaitre son assiette dans tout type d'environnement. L'estimation du mouvement est également basée sur des travaux dé-

veloppés conjointement avec le laboratoire MIS et le laboratoire Coréen KAIST. Ils consistent à déterminer la translation du mouvement à partir d'une vue omnidirectionnelle et d'un minimum de deux points mis en correspondance. Le travail de recherche proposé permet d'étendre cette méthode.
Au sein des projets menés autour de la thématique des drones, ce travail de thèse propose un nouveau capteur hybride utilisé par une approche systémique.

Perspectives

D'un point de vue scientifique, les perspectives sont nombreuses.
Tout d'abord, concernant les paramètres de navigation déterminés par vision, la méthode d'attitude proposée dans un environnement naturel est capable d'estimer le roulis et le tangage. Cependant, le lacet n'est pas déterminé. Néanmoins, les travaux [Montiel 2006, Scaramuzza 2008, Kalantari 2009] proposent des solutions pour le déterminer. En parallèle, en milieu naturel, la perception de l'horizon à partir de la vue fisheye doit être améliorée. En effet, l'angle de vue limité à 180° amène parfois à des situations d'occultation de la ligne d'horizon à faible altitude. L'exploitation d'une seconde vue pourrait être envisagée pour augmenter le champ de vision.
Quant à l'algorithme d'estimation d'altitude, il présente certaines limites sur les textures homogènes comme l'herbe à haute altitude. La segmentation du plan souffre parfois des diversités de surfaces plus ou moins texturées. Pour connaitre la distance au sol, la méthode de *plane-sweeping* se base sur une minimisation de l'erreur formulée d'une manière où le changement d'illumination modifie et altère les résultats. L'étude d'un nouveau critère invariant aux changements de luminosité est l'une des voies à explorer. Une autre perspective d'utilisation du *plane-sweeping* serait de déterminer avec l'altitude, la normale du sol observé. Ainsi cette méthode deviendrait indépendante de tout autre algorithme et apporterait une information redondante à l'estimation de l'attitude.
Pour la perception du mouvement, des travaux réalisés au sein du laboratoire proposent d'utiliser des droites ou de fusionner les droites et les points. Ces travaux démontrent une meilleure robustesse au bruit. Une perspective de ce travail serait d'améliorer la précision et la robustesse à partir de la fusion des lignes et des points issus des vues hybrides en exploitant l'information de profondeur des points observés.
Enfin, l'approche systémique propose d'utiliser des méthodes encapsulées. Cependant l'encapsulation présente certaines limites comme lors de la défaillance d'un capteur ou encore l'estimation erronée d'un paramètre. Des méthodes de fusion de données et l'ajout de capteurs supplémentaires redondants sont des perspectives à envisager. De plus déterminer chacun des états séparément permet de rendre

modulaire l'estimation des paramètres. Néanmoins, l'erreur issue de chaque paramètre est cumulée puis répercutée sur les méthodes qui en dépendent. Une piste envisagée consisterait à déterminer l'ensemble des paramètres simultanément.

D'un point de vue technique, la calibration hybride permet de mélanger différents types de caméras et de les caractériser par des modèles tout aussi variés. L'idée d'hybridation des caméras pourrait être étendue aux capteurs en général. C'est pourquoi calibrer un système de vision avec une centrale inertielle ou un dispositif laser sont des pistes envisagées dans le développement futur du logiciel. De plus, les distorsions ne sont pas proposées dans l'outil présenté. Les distorsions radiales et tangentielles constituent donc les axes principaux de développement du logiciel.
Ensuite, les capteurs choisis permettent d'obtenir une résolution satisfaisante pour l'application requise. Le développement de caméras HDRI constitue une piste à prendre en considération pour pallier les phénomènes d'éblouissement et les zones d'ombre. En effet, ce type de caméra permet d'obtenir un contraste plus important et de collecter une information plus fidèle de la radiance d'une scène. Le contraste est un facteur d'amélioration pour le *plane-sweeping*.
Ensuite, l'arrivée récente des caméras à haute vitesse est également un aspect à considérer pour le développement des algorithmes. Ces nouveaux capteurs couplés à des méthodes rapides pourraient être une alternative à des capteurs haute fréquence comme les centrales inertielles.
Toutefois, ces pistes nécessitent un besoin de calcul toujours croissant. D'un point de vue capacité de traitement, la puissance de calcul évolue constamment. Il est aujourd'hui possible d'optimiser le système proposé en utilisant de nouveaux composants plus performants en terme de capacité de calcul et d'architecture parallèle. Par exemple, les nouveaux processeurs ARM sont désormais multi-processeurs et sont capables de traiter plusieurs tâches simultanément. Une perspective serait alors d'adapter les différents algorithmes sur ce type de configuration puis de répartir les tâches entre les CPU et le DSP d'un ou plusieurs Gumstix.

La thématique des drones et de la vision en général sont des domaines en constante évolution. Les idées proposées dans cette thèse ont pour objectif d'ouvrir de nouvelles pistes en terme de configuration de caméras, de configuration matérielle et de traitements d'images stéréo sans mise en correspondance. Le travail réalisé est ouvert à une implémentation et une intégration dans la boucle de contrôle afin de stabiliser le vol d'un drone. La perpétuelle progression des capacités de traitement ouvre de nouvelles perspectives.

Les méthodes proposées peuvent également être étendues à des applications de robotique sous-marine, terrestre et aérienne. L'estimation de la position et des

paramètres de déplacement d'un réseau de capteurs hybrides permettrait d'étendre l'idée de coopération à la navigation d'une flotte de robots hybrides. Nous proposons l'idée d'élaborer de nouveaux systèmes que nous appellerons "composites", composés d'éléments de nature très différentes, à savoir :

- des processeurs de traitements hybrides ou fusionnés (CPU,GPU,DSP),
- des capteurs de natures très différentes, calibrés par des méthodes hybrides,
- des robots hybrides,
- une collaboration et une complémentarité des systèmes utilisés.

La recherche, l'imagination et l'implémentation de nouvelles méthodes permettront d'explorer ces diverses perspectives.

Publications

Ce travail de thèse a donné lieu aux publications suivantes :

1. G. Caron et D. Eynard, "Etalonnage simultané de systèmes stéréoscopiques hybrides", Congrès des jeunes chercheurs en vision par ordinateur, ORASIS,2011.
2. D. Eynard, P. Vasseur, C. Demonceaux et V. Fremont, "Estimation de mouvement d'un drone à partir d'un capteur stéréo hybride", Congrès des jeunes chercheurs en vision par ordinateur, ORASIS,2011.
3. G. Caron et D. Eynard, "Multiple Camera Types Simultaneous Stereo Calibration", International Conference on Robotics and Automation, ICRA, 2011.
4. D. Eynard, P. Vasseur, C. Demonceaux et V. Fremont, "UAV Motion Estimation using Hybrid Stereoscopic Vision", Machine Vision Applications, MVA, 2011.
5. D. Eynard, P. Vasseur, C. Demonceaux et V. Fremont, "UAV Altitude Estimation by Mixed Stereoscopic Vision", International Conference on Intelligent Robots and Systems, IROS, 2010.
6. D. Eynard, P. Vasseur, C. Demonceaux et V. Fremont, "Estimation temps réel de l'altitude d'un drone à partir d'un capteur de stéréovision mixte", Reconnaissance des Formes et Intelligence Artificielle, RFIA, 2010.
7. G. Caron, D. Eynard, "Hybrid stereoscopic calibration", SPIE, 2011.

En soumission :

1. D. Eynard, P. Vasseur, C. Demonceaux, V. Fremont, "Real time UAV Altitude, Attitude and Motion Estimation from Hybrid Stereovision", Autonomous Robot Journal, Special Issue Micro-UAV Perception and Control.

Algorithme d'estimation du mouvement

Algorithme 2 Estimation de la translation à partir de points du sol

estimTparMC$(pointsPlan_t, pointsPlan_{t+1})$

$n = taille(pointsPlan_t)$
$A = zeros(n * 3, 3)$
$B = zeros(n * 3, 1)$
$pos = 1$
pour $i = 1 : n$ **faire**
%calcul de la distance au point a t=0
%normale exprimee ds la perspective
$n_{0p} = R_n n_0$
$d_p = d_s - (R_n^{-1} T_n^T)^T n_0$
$p0 = pointsPlan_t(i, :)^T$
$t = \frac{-d_p}{n'_{0p} p0}$
$d_p = ||\frac{p0}{||p0||} t||$
$A(pos + 0, :) = [0, -pointsPlan_{t+1}(i, 3), pointsPlan_{t+1}(i, 2)]$
$A(pos + 1, :) = [pointsPlan_{t+1}(i, 3), 0, -pointsPlan_{t+1}(i, 1)]$
$A(pos + 2, :) = [-pointsPlan_{t+1}(i, 2), pointsPlan_{t+1}(i, 1), 0]$
$c = -d_p(pointsPlan_{t+1}(i, :) \times R01 pointsPlan_t(i, :)^T)$
$B(pos + 0) = c(1)$
$B(pos + 1) = c(2)$
$B(pos + 2) = c(3)$
$pos = pos + 3$
$T = ((pinv(A)B))^T$

Algorithme 3 Estimation de la translation 0 partir de points du sol et 3D

estimTparMCFusion$(pointsPlan_t, pointsPlan_{t+1}, points3D_t, points3D_{t+1})$

$n = taille(pointsPlan_t)$
$A = zeros(n * 3, 3)$
$B = zeros(n * 3, 1)$
$pos = 1$
pour $i = 1 : n$ **faire**
%calcul de la distance au point a t=0
%normale exprimee ds la perspective
$n_{0p} = R_n n_0$
$d_p = d_s - (R_n^{-1} T_n^T)^T n_0$
$p0 = pointsPlan_t(i, :)^T$
$t = \frac{-d_p}{(n'_{0p} p0)}$
$d_p = ||\frac{p0}{||p0||} t||$
$A(pos + 0, :) = [0, -pointsPlan_{t+1}(i, 3), pointsPlan_{t+1}(i, 2)]$
$A(pos + 1, :) = [pointsPlan_{t+1}(i, 3), 0, -pointsPlan_{t+1}(i, 1)]$
$A(pos + 2, :) = [-pointsPlan_{t+1}(i, 2), pointsPlan_{t+1}(i, 1), 0]$
$c = -d_p(pointsPlan_{t+1}(i, :) \times R_{01} pointsPlan_t(i, :)^T)$
$B(pos + 0) = c(1)$
$B(pos + 1) = c(2)$
$B(pos + 2) = c(3)$
$pos = pos + 3$
$n = size(pts3D0, 1)$
pour $i = 1 : n$ **faire**
$C(i, :) = points3D_{t+1}(i, :)^T \times R_{01} points3D_t(i, :)^T$
$A = [A; C]$
$B = [B; zeros(taille(points3D_t, 1), 1)]$
$T = ((pinv(A)B))^T$

Algorithme 4 Algorithmes d'estimation du mouvement

Motion Main()

$Rcumul = Id_3$
%Calibration extrinseque
$GLOBAL : Rn, Tn$
%Calibration intrinseque
$GLOBAL : K_{fe}, K_p, xi$
pour $t = 0 : nImages$ **faire**
  %Points fisheye
  $X_{FE,t} = listeFisheye_t$
  $X_{FE,t+1} = listeFisheye_{t+1}$
  $X_{FE,t} = ImageVersSphere(K_{fe}^{-1}[X_{FE,t}, 1]^T, xi)$
  $X_{FE,t+1} = ImageVersSphere(K_{fe}^{-1}[X_{FE,t+1}, 1]^T, xi)$
  %Points perspective
  $X_{P,t} = listePerspective_t$
  $X_{P,t+1} = listePerspective_{t+1}$
  $X_{P,t} = K_p^{-1}[X_{P,t}1]^T$
  $X_{P,t+1} = K_p^{-1}[X_{P,t+1}1]^T$
  %Points mixtes
  $M_t = [X_{pt}; X_{st}]$
  $M_{t+1} = [X_{pt+1}; X_{st+1}]$
  $Ransac(15, 3, 2, taille(M_t)0.7, X_{P,t}, X_{P,t+1}, X_{FE,t}, X_{FE,t+1})$
  %On replace chaque translation dans le repère de la première image
  $Rcumul = RcumulR_{01,t}^{-1}$
  pour $i = 0 : t - 1$ **faire**
    $translation(i, :) = (R_{01,t}translation(i, :)^T)^T$
  $TGlobal = [0, 0, 0]$
  pour $i = 0 : t - 1$ **faire**
    $TGlobal = TGlobal + (translation(i, :)^T)^T$
  $cumulTranslation(t + 1, :) = (-Rcumul(TGlobal^T + Tmix^T))^T$

Bibliographie

[Aer] http ://www.ctie.monash.edu.au/hargrave/images/kettering_bug_1_500.jpg. (Cité en pages i et 2.)

[Andert 2010] F. Andert, F.-M. Adolf, L. Goormann et J.-S. Dittrich. *Autonomous Vision-Based Helicopter Flights Through Obstacle Gates*. J. Intell. Robotics Syst., vol. 57, pages 259–280, January 2010. (Cité en page 4.)

[Baker 1999] S. Baker et S.-K. Nayar. *A Theory of Single-Viewpoint Catadioptric Image Formation*. International Journal of Computer Vision, vol. 35, pages 175–196, 1999. 10.1023/A :1008128724364. (Cité en page 31.)

[Barreto 2001] Joao Barreto et Araujo H. *Issues on the Geometry of Central Catadioptric Image Formation*. International Conference On Pattern Recognition, 2001. (Cité en pages ii, 30 et 32.)

[Barrows 2001] G. Barrows, C. Neely et K. Miller. *Optic Flow Sensors for MAV Navigation*. vol. 195, pages 557–574, 2001. (Cité en pages 16 et 18.)

[Bastanlar 2010] Y. Bastanlar, A. Temizel, Y. Yardimci et P. Sturm. *Effective Structure-from-Motion for Hybrid Camera Systems.* In ICPR'10, pages 1654–1657, 2010. (Cité en pages 22, 23, 29 et 30.)

[Bazin 2008] J.C. Bazin, I.S. Kweon, C. Demonceaux et P. Vasseur. *UAV Attitude Estimation by Vanishing Points in Catadioptric Image*. In IEEE International Conference on Robotics and Automation 2008 (ICRA'08), pages 2743–2749, Pasadena, CA, May 2008. IEEE. (Cité en pages i, 18, 19, 20, 26 et 54.)

[Bazin 2010] J.C. Bazin, C Demonceaux, P. Vasseur et I.S. Kweon. *Motion Estimation by Decoupling Rotation and Translation in Catadioptric Vision*. Computer Vision and Image Understanding, vol. Volume 114, Issue 2, pages 254–273, February 2010. (Cité en pages 21, 22, 23 et 67.)

[Beyeler 2006] A. Beyeler, C. Mattiussi, J.-C. Zufferey et D. Floreano. *Visionbased altitude and pitch estimation for ultra-light indoor aircraft*. In in IEEE International Conference on Robotics and Automation ICRA'06, pages 2836–2841, 2006. (Cité en pages 16, 17 et 18.)

[Beyeler 2009] A. Beyeler, J.-C. Zufferey et D. Floreano. *optiPilot : control of take-off and landing using optic flow*. In Proceedings of the 2009 European Micro Air Vehicle conference and competition (EMAV '09), 2009. (Cité en pages ii, 24 et 25.)

[Bouguet 2000] J.-Y. Bouguet. *Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the algorithm*, 2000. (Cité en page 69.)

[Bouguet 2004] J.-Y. Bouguet. *Camera Calibration Toolbox for Matlab*, 2004. (Cité en pages 32 et 33.)

[Bouguet 2010] Bouguet, 2010. http ://www.vision.caltech.edu/bouguetj/. (Cité en page 75.)

[Boutteau 2009] R. Boutteau, X. Savatier, Ertaux J.-Y. et Mazari B. *An Omnidirectional Stereoscopic System for Mobile Robot Navigation*. volume 5, pages 3–17, 2009. (Cité en page 33.)

[Bradski 2000] G. Bradski. *The OpenCV Library*. Dr. Dobb's Journal of Software Tools, 2000. (Cité en page 44.)

[Caron 2009] G. Caron, E. Marchand et E. Mouaddib. *Model Based Pose Estimation For Omnidirectional Stereovision*. In IEEE Int. Conf. on Intelligent Robots and Systems, IROS'09, pages 5228–5233, St Louis, USA, October 2009. (Cité en page 33.)

[Caron 2010] G. Caron, E. Marchand et E.M. Mouaddib. *Single Viewpoint Stereoscopic Sensor Calibration*. In In International Symposium on I/V Communications and Mobile Networks, Rabat, Morocco, September 2010. ISIVC. (Cité en page 33.)

[Caron 2011] G. Caron et D. Eynard. *Hybrid Stereoscopic Calibration Software*. In http ://www.hyscas.com, 2011. (Cité en pages 33, 49 et 73.)

[Chahl 2004] Javaan S. Chahl, Mandyam V. Srinivasan et Shao-Wu Zhang. *Landing Strategies in Honeybees and Applications to Uninhabited Airborne Vehicles*. I. J. Robotic Res., vol. 23, no. 2, pages 101–110, 2004. (Cité en pages 16 et 18.)

[Cheriany 2009] A. Cheriany, J. Andersh, V. Morellas, N. Papanikolopoulos et B. Mettler. *Autonomous altitude estimation of a UAV using a single onboard camera*. In IEEE Int. Conf. on Intelligent Robots and Systems, IROS'09, IROS'09, pages 3900–3905, Piscataway, NJ, USA, 2009. IEEE Press. (Cité en pages 17 et 18.)

[Collins 1996] R. Collins. *A Space-Sweep Approach to True Multi-Image Matching*. In IEEE Computer Vision and Pattern Recognition, pages 358–363, June 1996. (Cité en page 55.)

[Daniilidis 1993] K. Daniilidis et H.-H. Nagel. *The Coupling of Rotation and Translation in Motion Estimation of Planar Surfaces*. In IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'93), pages 188–193, New-York, USA, June 1993. (Cité en pages 21, 22 et 23.)

[Demonceaux 2006a] C. Demonceaux, P. Vasseur et C. Pégard. *Omnidirectional vision on UAV for attitude computation*. In IEEE International Conference on Robotics and Automation 2006 (ICRA'06), pages 2842–2847, Orlando, FL, US, May 2006. (Cité en pages 26 et 54.)

[Demonceaux 2006b] C. Demonceaux, P. Vasseur et C. Pégard. *Robust Attitude Estimation with Catadioptric Vision*. In IEEE/RSJ International Conference on Intelligent Robots and Systems 2006 (IROS'06), pages 3448–3453, Beijing, China, October 2006. (Cité en pages i, 19, 20, 26 et 54.)

[Demonceaux 2007] C. Demonceaux, P. Vasseur et C. Pégard. *UAV Attitude Computation by Omnidirectional Vision in Urban Environment*. In IEEE International Conference on Robotics and Automation 2007 (ICRA'07), pages 2017–2022, Roma, Italy, April 2007. (Cité en pages 18, 20, 26, 28 et 54.)

[Dusha 2007] Damien Dusha, Wageeh W. Boles et Rodney Walker. *Fixed-Wing Attitude Estimation Using Computer Vision Based Horizon Detection*. In 12th Australian International Aerospace Congress, pages 1–19, Melbourne Australia, 2007. (Cité en pages 19 et 20.)

[Euston 2008] Mark Euston, Paul William Coote, Robert E. Mahony, Jonghyuk Kim et Tarek Hamel. *A complementary filter for attitude estimation of a fixed-wing UAV*. In IROS, pages 340–345, 2008. (Cité en page 12.)

[Eynard 2010] D. Eynard, P. Vasseur, C. Demonceaux et V. Fremont. *UAV Altitude Estimation by Mixed Stereoscopic Vision*. In IEEE Int. Conf. on Intelligent RObots and Systems, IROS'10, Taipei, Taiwan, October 2010. (Cité en pages 26 et 54.)

[Eynard 2011] D. Eynard, P. Vasseur, C. Demonceaux et V. Fremont. *UAV Motion Estimation using Hybrid Stereoscopic Vision*. In IAPR Conf. on Machine Vision Applications, MVA'11, Nara, Japan, June 2011. (Cité en page 54.)

[Fabiani 2007] P. Fabiani, V. Fuertes, A. Piquereau, R. Mampey et F. Teichteil-Königsbuch. *Autonomous flight and navigation of VTOL UAVs : from autonomy demonstrations to out-of-sight flights*. Aerospace Science and Technology, vol. 11, no. 2-3, pages 183 – 193, 2007. (Cité en page 4.)

[Fis] http ://www.nikonweb.com/fisheye/. (Cité en pages i et 15.)

[Fuk] http ://www.telegraph.co.uk/news/picturegalleries/worldnews/8411783/Japan-earthquake-and-tsunami-the-Fukushima-Daiichi-nuclear-power-plant.html ?image=4. (Cité en pages i et 3.)

[Gallup 2007] D. Gallup, J.-M. Frahm, P. Mordohai, Y. Qingxiong et M. Pollefeys. *Real-Time Plane-Sweeping Stereo with Multiple Sweeping Directions*, 2007. (Cité en page 55.)

[Garcia-Carrillo 2011] L.-R. Garcia-Carrillo, A. Dzul, R. Lozano et C. Pégard. *Combining Stereo Vision and Inertial Navigation System for a Quad-Rotor UAV*. In In 2011 International Conference on Unmanned Aircraft Systems, ICUAS'11, Denver, CO, USA, May 2011. (Cité en pages 20 et 23.)

[Garcia-pardo 2000] Pedro J. Garcia-pardo, Gaurav S. Sukhatme et James F. Montgomery. *Towards Vision-Based Safe Landing for an Autonomous Helicopter*. "Robotics and Autonomous Systems", 2000. (Cité en pages 16 et 18.)

[GDR-GT2 2011] GDR-GT2, 2011. http ://www.gdr-robotique.org/gt.php ?id=3. (Cité en pages i et 5.)

[Geyer 2001] C. Geyer et K. Daniilidis. *Structure and Motion from Uncalibrated Catadioptric Views*. In IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'01), pages 279–286, 2001. (Cité en pages i, 21, 22 et 23.)

[Goudail 2001] F. Goudail et Philippe Réfrégier. *Algorithmes statistiques pour le traitement des images polarimétriques en lumière cohérente*. Traitement du Signal, vol. 18 (5-6), pages 297–319, 2001. (Cité en page 24.)

[GPS] http ://www.tradeage.com/fr/sale/antenne-gps-à-puce-gps-récepteur-gps-usb-de-la-souris-pour-ordinateur-portable-ce228430. (Cité en pages i et 11.)

[Green 2003] W.-E. Green, P.-Y. Oh, K. Sevcik et G. Barrows. *Autonomous Landing for Indoor Flying Robots Using Optic Flow*. In in ASME International Mechanical Engineering Congress and Exposition, pages 1347–1352, 2003. (Cité en pages 16 et 18.)

[Hartley 2004] R. I. Hartley et A. Zisserman. Multiple view geometry in computer vision. Cambridge University Press, ISBN : 0521540518, seconde édition, 2004. (Cité en pages 21 et 58.)

[HDR] http ://fr.wikipedia.org/wiki/Fichier :HDRI-Example.jpg. (Cité en pages i et 14.)

[Hrabar 2003] S.-E. Hrabar et G.-S. Sukhatme. *Omnidirectional Vision for an Autonomous Helicopter*. In IEEE International Conference on Robotics and Automation, pages 558 – 563, 2003. (Cité en page 28.)

[IDS] http ://www.ids-imaging.com/. (Cité en pages ii et 41.)

[Jung 2003] I.-K. Jung et S. Lacroix. *High resolution terrain mapping using low altitude aerial stereo imagery*. In Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2, ICCV '03, pages 946–, Washington, DC, USA, 2003. IEEE Computer Society. (Cité en pages 17 et 18.)

[Kalantari 2009] Mahzad Kalantari, Amir Hashemi, Franck Jung et Jean-Pierre Guédon. *A New Solution to the Relative Orientation Problem using only 3 Points and the Vertical Direction*. CoRR, vol. abs/0905.3964, 2009. (Cité en page iii.)

[Kalman 1960] R.-E. Kalman. *A New Approach to Linear Filtering and Prediction Problems*. Transactions of the ASME - Journal of Basic Engineering, 1960. (Cité en page 68.)

[Kanatani 1985] K.-I. Kanatani. *Detecting the motion of a planar surface by line and surface integrals*. Computer Vision, Graphics, and Image Processing, vol. 29, no. 1, pages 13 – 22, 1985. (Cité en pages 22 et 23.)

[Las] http ://uk.farnell.com/bosch/dle-70/laser-telemeter-bluetooth-sew/dp/1638123. (Cité en pages i et 11.)

[Lee 2000] J. W. Lee, S. You et U. Neumann. *Large Motion Estimation for Omnidirectional Vision*. pages 161–, 2000. (Cité en page 22.)

[Lhuillier 2008] M. Lhuillier. *Automatic scene structure and camera motion using a catadioptric system*. Computer Vision and Image Understanding, vol. 109, pages 186–203, 2008. (Cité en pages i, 21, 22 et 23.)

[Lobo 2007] Jorge Lobo et Jorge Dias. *Relative Pose Calibration Between Visual and Inertial Sensors*. The International Journal of Robotics Research, vol. 26, no. 6, pages 561–575, Juin 2007. (Cité en pages 77 et 79.)

[Lowe 1999] David G. Lowe. *Object Recognition from Local Scale-Invariant Features*. In Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2, ICCV '99, pages 1150–, Washington, DC, USA, 1999. IEEE Computer Society. (Cité en page 21.)

[Ly 2011] S. Ly, C. Demonceaux et P. Vasseur. *Motion estimation for hybrid cameras using point and line feature fusion.* In IAPR (International Association of Pattern Recognition, MVA'11, Nara, Japan, June 2011. (Cité en pages 22, 23 et 30.)

[Ma 2003] Y. Ma, S. Soatto, J. Kosecka et S. Sastry. An invitation to 3d vision : From images to geometric models. 2003. (Cité en page 21.)

[Makadia 2007] A. Makadia, C. Geyer et K. Daniilidis. *Correspondence-free Structure from Motion*. International Journal of Computer Vision, vol. 75, no. 3, pages 311–327, 2007. (Cité en pages 22 et 23.)

[Malis 2005] E. Malis et E. Marchand. *Méthodes robustes d'estimation pour la vision robotique*. In Journées nationales de la recherche en robotique, JNRR'05, Guidel, France France, 2005. (Cité en page 69.)

[Marchand 2001] E. Marchand et François Chaumette. *A new formulation for non-linear camera calibration using virtual visual servoing.* In Rapport technique 1366, IRISA, January 2001. (Cité en page 33.)

[Mei 2006] C. Mei, S. Benhimane, E. Malis et P. Rives. *Homography-based Tracking for Central Catadioptric Cameras*. In In Proc. Intl. Conf. on Robotics and Automation, IROS'06, 2006. (Cité en page 58.)

[Mei 2007] C. Mei et P. Rives. *Single View Point Omnidirectional Camera Calibration from Planar Grids*. In In Proc. Intl. Conf. on Robotics and Automation, IROS'07, April 2007. (Cité en pages 31 et 32.)

[Mei 2011] Mei, 2011. http ://homepages.laas.fr/ cmei/index.php/Toolbox. (Cité en page 75.)

[Meingast 2005] M. Meingast, C. Geyer et S. Sastry. *Vision Based Terrain Recovery for Landing Unmanned Aerial Vehicles*. In Proceedings of the Conference on Decision and Control, 2005. (Cité en pages 17 et 18.)

[Microstrain] Microstrain. http ://www.hellopro.fr/centrales-inertielles-2003811-fr-1-feuille.html. (Cité en pages i, ii, 11 et 38.)

[Mondragón 2010] I. Mondragón, M. Olivares-Méndez, P. Campoy, C. Martínez et L. Mejias. *Unmanned aerial vehicles UAVs attitude, height, motion estimation and control using visual systems*. Autonomous Robots, vol. 29, pages 17–34, 2010. (Cité en page 26.)

[Montiel 2006] J. M. M. Montiel et Davison A.-J. *A visual compass based on SLAM*. In In Proc. Intl. Conf. on Robotics and Automation, (IROS'06). Available : http ://pubs.doc.ic.ac.uk/visual-compass-slam, 2006. (Cité en page iii.)

[Moore 2010] R.-J. Moore, S. Thurrowgood, D.-P. Bland, D. Soccol et M.-V. Srinivasan. *UAV altitude and attitude stabilisation using a coaxial stereo vision system.* In IEEE Int. Conf. on Robotics and Automation, ICRA'10, pages 29–34, 2010. (Cité en page 25.)

[Mouaddib 2005] E. Mouaddib. *La Vision Omnidirectionnelle*. In Journées nationales de la recherche en robotique, JNRR'05, Guidel, France, October 2005. (Cité en pages i, 15 et 16.)

[MTI] http ://www.rimtai.com/chtr/Products/XSENS/images/MTI-28A33G35.jpg. (Cité en pages ii et 38.)

[Museum] White Sands Missile Range Museum. http ://www.wsmr-history.org/BQM-34A.htm. (Cité en pages i et 3.)

[Nor] http ://en.wikipedia.org/wiki/Earth%27s_magnetic_field. (Cité en pages ii et 36.)

[OMA] http ://www.flickr.com/photos/gbraad/3755933364/. (Cité en pages ii et 43.)

[Omn] http ://www.tuexperto.com/2008/07/14/olympus-crea-una-lente-para-grabar-videos-a-360-grados/. (Cité en pages i et 15.)

[One] http ://www.onera.fr/dmph/capteurs-inertiels/gyro-vig.php. (Cité en pages ii et 36.)

[Onera 2005] Onera, 2005. http ://www.onera.fr/dcsd/ressac/index.html. (Cité en pages i, 2, 4, 5 et 6.)

[Onera 2009a] Onera, 2009. http ://www.onera.fr/conferences/drones/drones-histoire.php. (Cité en page 1.)

[Onera 2009b] Onera, 2009. http ://www.onera.fr/conferences/drones/categories-de-drones.php. (Cité en page 2.)

[Par] http ://www.geekonaute.fr/to-become/parrot-ar-drone-une-selection-des-plus-belles-videos/. (Cité en pages i et 3.)

[Pio] http ://www.air-defense.net/forum/index.php ?topic=12066.30. (Cité en pages i et 3.)

[Pixhawk 2010] Pixhawk, 2010. http ://pixhawk.ethz.ch. (Cité en pages ii, 43 et 44.)

[Pre a] http ://mangaloreanrobots.blogspot.com/2011_07_01_archive.html. (Cité en pages i et 3.)

[Pre b] http ://www.onera.fr/conferences/mesures-aerodynamique/07-capteurs-pression-parietale.php. (Cité en pages i et 11.)

[Puig 2008] L. Puig, J. Guerrero et P. Sturm. *Matching of Omindirectional and Perspective Images using the Hybrid Fundamental Matrix*. In Proceedings of the Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras, Marseille, France, oct. 2008. (Cité en page 29.)

[Rad] http ://rb-kwin.bosch.com/fr/pool/de/sicherheit/CAPS/Radarsensor.jpg. (Cité en pages i et 11.)

[Rondon 2010] E. Rondon, L.-R. Garcia-Carrillo et I. Fantoni. *Vision-Based Altitude, Position and Speed Regulation of a Quadrotor Rotorcraft*. In 2010 International Conference on Intelligent Robots and Systems, IROS'10, Taipei, Taiwan, October 2010. (Cité en page 25.)

[Sanahuja 2010a] G. Sanahuja, 2010. http ://www.hds.utc.fr/ gsanahuj/dokuwiki/doku.php. (Cité en page 44.)

[Sanahuja 2010b] G. Sanahuja. *Commande et localisation embarquée d'un drone aérien en utilisant la vision*. PhD thesis, January 2010. (Cité en pages ii, 20, 23 et 24.)

[Sanfourche 2003] M. Sanfourche, G. Le Besnerais et S. Philipp-Foliguet. *Height estimation using aerial side looking image sequences*. In ISPRS Archives, volume vol. XXXIV, München, sept 2003. ISPRS. Proceedings of ISPRS Conference PIA'03 (Photogrammetric Image Analysis). (Cité en pages 17 et 18.)

[Saripalli 2002] S. Saripalli, J.-F. Montgomery et G.-S. Sukhatme. *Vision-based Autonomous Landing of an Unmanned Aerial Vehicle*. In IEEE International Conference on Robotics and Automation, ICRA'02, pages 2799–2804, 2002. (Cité en pages 16 et 18.)

[Scaramuzza 2006] D. Scaramuzza, A. Martinelli et Y.-Roland Siegwart. *A Toolbox for Easily Calibrating Omnidirectional Cameras*. In Iros, Benjing, Chine, 2006. (Cité en page 32.)

[Scaramuzza 2008] D. Scaramuzza et R. Siegwart. *Correcting Vehicle Heading in Visual Odometry by Using Image Appearance*. In Proc. of The First International Workshop on Omnidirectional Robot Vision, November 2008. (Cité en page iii.)

[Shakernia 1999] O Shakernia, Y Ma, T Koo et S Sastry. *Landing an unmanned air vehicle : Vision based motion estimation and nonlinear control*. Asian Journal Of Control, vol. 1, no. 3, pages 128–145, 1999. (Cité en pages 22 et 23.)

[Sharp 2001] Cory S. Sharp, Omid Shakernia et S. Shankar Sastry. *A Vision System for Landing an Unmanned Aerial Vehicle*. In IEEE International Conference on Robotics and Automation, ICRA'01, Seoul, Korea, pages 1720–1727, 2001. (Cité en pages i, 16, 17 et 18.)

[Singh 2005] S.-P.-N. Singh et K.-J. Waldron. *Motion Estimation by Optical Flow and Inertial Measurements for Dynamic Legged Locomotion*. In IEEE Conference on Robotics and Automation, 2005. (Cité en pages 20 et 23.)

[Spe] (Cité en pages ii et 25.)

[Sturm 2002] P. Sturm. *Mixing Catadioptric and Perspective Cameras*. In Workshop on Omnidirectional Vision, Copenhagen, Denmark, pages 37–44, 2002. (Cité en page 29.)

[Thurrowgood 2009] Saul Thurrowgood, Dean Soccol, Richard James Donald Moore, Daniel Peter Bland et Mandyam V. Srinivasan. *A Vision based system for attitude estimation of UAVS*. In IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS'09 , October 11-15, 2009, St. Louis, MO, USA, pages 5725–5730, 2009. (Cité en page 20.)

[URG] http ://www.me.vt.edu/uav/html/vehicle.html. (Cité en pages i et 11.)

[Vasconcelos 2010] J.F. Vasconcelos, C. Silvestre, P. Oliveira et B. Guerreiro. *Embedded UAV model and LASER aiding techniques for inertial navigation systems*. Control Engineering Practice, vol. 18, no. 3, pages 262 – 278, 2010. (Cité en page 11.)

[Ying 2004] X. Ying et Z. Hu. *Can We Consider Central Catadioptric Cameras and Fisheye Cameras within a Unified Imaging Model*. In Tomas Pajdla

et Jiri Matas, editeurs, Computer Vision - ECCV 2004, volume 3021 of *Lecture Notes in Computer Science*, pages 442–455. Springer Berlin / Heidelberg, 2004. (Cité en page 31.)

[Zufferey 2010] J.-C. Zufferey, A. Beyeler et D. Floreano. *Autonomous flight at low altitude using light sensors and little computational power*. International Journal of Micro Air Vehicles, vol. 2, no. 2, pages 107–117, 2010. (Cité en pages 17, 18 et 25.)