Détection des deux roues par capteurs vidéos fixes

Nicolas Tronson

ECOLE CENTRALE DE NANTES
IFSTTAR

ECOLE DOCTORALE SPIGA

THÈSE DE DOCTORAT

Spécialité : GÉNIE CIVIL

Présentée et soutenue publiquement par:

NICOLAS TRONSON

le 28 mai 2013
à l'IFSTTAR de Nantes

DÉTECTION DES DEUX ROUES PAR CAPTEURS VIDÉO FIXES

Jury

Directeur :	Philippe Lepert	Directeur de Recherches, IFSTTAR Nantes
Co-directeur :	Laurent Trassoudaine	Professeur, Institut Pascal Clermont-Ferrand
Rapporteurs :	Sylvie Treuillet	Professeur, Université d'Orléans
	Pascal Vasseur	Professeur, INRA Rouen
	Majdi Khoudeir	Professeur, Université de Poitiers
Examinateurs :	Luce Morin	Professeur, INSA Rennes
	Didier Aubert	Directeur de Recherches, IFSTTAR Paris
	Jérôme Idier	Directeur de Recherches, IRCCyN Nantes

RÉSUMÉ

Les travaux de thèse présentés dans ce mémoire ont pour objectif la création d'un système optique permettant la détection des deux-roues en milieu urbain. Pour cela, nous avons fait le choix, à la vue des limites des systèmes actuels principalement lié aux ombres portées des véhicules, pénalisant ainsi leur détection, d'opter pour un système de prise de vue en stéréovision. L'originalité du système présenté ici réside dans le choix des optiques et la disposition des caméras. Nous utilisons en effet, deux caméras avec optique fisheye placées l'une au dessus de l'autre, alignées et orientées selon un axe vertical. Nous choisissons cette configuration pour des raisons de stabilité et de couverture de la scène. Nous étudions ensuite la calibration de ce système à partir d'un outil général de calibration de système de stéréovision, ainsi qu'à partir d'une approche développée spécifiquement et permettant de prendre en compte les particularités du système. En ce qui concerne le traitement des données pour effectuer la mise en correspondance entre les images et ainsi obtenir une carte 3D, nous proposons trois approches. Les deux premières sont largement inspirées de l'état de l'art. La troisième approche, que nous avons conçue, est assez originale et permet de cibler plus précisément la zone de recherche 3D en décomposant la scène en différentes couches correspondant à différentes hauteurs au dessus du niveau de la route. Pour pouvoir enfin détecter, classifier et suivre les véhicules, les informations 3D apportant la position et la dimension des véhicules sont intégrées dans un logiciel de suivi déjà existant.

Mots-clés stéréovision, optique fisheye, deux-roues, calibration, reconstruction 3D

Abstract

The objective of the work presented in this thesis is the creation of an optic system able to detect the two-wheeled vehicles in urban areas. To this prospect, we choose, owing to the limits of current systems primarily due to shadows cast by vehicles, thus penalizing their detection, to opt for a stereovision system. The originality of the system presented here is the optics and cameras placement choice. We use two cameras placed one above the other, oriented and aligned along a vertical axes. We choose this configuration for stability reasons, and because it makes it possible to cover the entire scene. Then, we study the system calibration, in one hand with a generic calibration tool for stereo systems, in the other hand with an approach specifically developed to consider the characteristics of the system. With regard to data processing, to perform the matching between images and thus obtain a 3D map, we propose three approaches. The first two are largely based on the state of the art. The third one, which was designed as a part of this thesis, is quite original and can target more precisely the search area by decomposing the 3D scene in different layers corresponding to different heights above the road level. To finally be able to detect, classify and track vehicles, 3D information providing the position and size of vehicles is integrated into an existing tracking software.

Keywords stereovision, fisheye lens, two wheeled vehicles, calibration, 3D reconstruction

REMERCIEMENTS

Je tenais tout d'abord à remercier Yann GOYAT pour m'avoir encadré durant ces trois années de thèse. Je remercie particulièrement Philippe LEPERT pour avoir dirigé ma thèse et ses précieux conseils en particulier lors de la rédaction de ce manuscrit. Je remercie également Laurent TRASSOUDAINE, Thierry CHATEAU, Jean-Philippe TAREL pour leurs conseils, et remarques avisées pendant la durée de cette thèse ainsi que pour la rédaction de ce manuscrit. J'adresse également mes remerciements à Alain RIOUALL pour son aide précieuse concernant l'installation du matériel, ainsi qu'à Olivier PITARD pour m'avoir aidé pour la prise en main et le paramétrage du logiciel d'acquisition et des caméras. Un grand merci à tous les membres de l'UR AGIT pour leur accueil et l'excellente ambiance qui y règne, notamment à Manuel ROURA, qui fût mon collègue de bureau durant ces trois ans. Je remercie également Dominique GRUYER du LIVIC pour m'avoir fourni le logiciel de simulation Sivic ainsi que pour son aide pour la prise en main de celui-ci, et pour le développement de modules qui m'ont été indispensables pour effectuer mes simulations. J'adresse également des remerciements à Damien EYNARD du laboratoire LITIS à Rouen pour son aide précieuse pour la calibration des caméras, et pour les vidéos qu'il a pu nous réaliser quand nos caméras étaient en panne.
Enfin, j'adresse mes profonds remerciements à ma famille et mes amis : à mes parents, ma sœur et mes frères pour m'avoir toujours soutenu, ainsi qu'à mes amis, Natacha et Pascal pour les nombreux weekends, Marie et Simi pour les soirées guitare, ainsi qu'à Anthony pour m'avoir permis de m'évader durant cette thèse en m'accueillant en Norvège durant les vacances.

TABLE DES MATIÈRES

LISTE DES FIGURES

Liste des tables

INTRODUCTION GÉNÉRALE

MALGRÉ une forte baisse du nombre de tués sur les routes ces dernières années, le nombre de morts reste élevé parmi les conducteurs de deux roues, ceci pour différentes raisons :

⇒ les conducteurs de deux roues sont beaucoup plus vulnérables
⇒ la cohabitation avec les autres véhicules pose parfois problème
⇒ l'aménagement routier n'est pas toujours bien adapté

Pour ces deux dernières raisons, il est important d'analyser en détails le comportement des deux roues vis à vis des autres véhicules ainsi que de la configuration routière.

Ces connaissances devraient être utiles à différents acteurs :

⇒ pour les **responsables de l'aménagement urbain** : elles doivent permettre d'aider à la prise de décision, concernant l'aménagement pour garantir une meilleure cohabitation des véhicules tout en préservant leur sécurité. Cette prise en compte est indispensable pour rester cohérent avec des politiques favorisant le développement du vélo en ville.
⇒ pour les **usagers de la route** : elles permettent un meilleur aménagement garantissant leur sécurité. Un sentiment de sécurité accru permet ainsi de favoriser la pratique du deux-roues.

L'objectif de cette thèse a donc été de concevoir un système, basé sur la vision, apportant un outil pour l'analyse du trafic des deux roues en milieu urbain, et ainsi aider à la prise de décision concernant l'aménagement de ces lieux. Cet objectif constitue ce que nous nommerons "notre problématique".

Des systèmes existent à l'heure actuelle pour le suivi et la détection de véhicules, mais ceux-ci ne se prêtent pas bien à la problématique des deux-roues, principalement pour des problèmes d'ombres et de proximité des véhicules (ce point est abordé en détail par la suite). L'idée est donc de s'affranchir de ce problème en introduisant une prise de vue en stéréovision. La stéréovision, approche relativement récente, n'a pas pas été jusque ici tellement utilisée pour le suivi de véhicules. Il est donc important d'en voir les spécificités.

Le *premier chapitre* présente la problématique, montre quels sont les problèmes posés par les deux-roues et explique l'intérêt d'un système de comptage de ceux-ci pour décider des aménagements. Il montre aussi

pourquoi ce comptage pose problème avec les systèmes de suivi de trajectoire existants. Partant de ce constat, on propose de réaliser un nouveau système de suivi et comptage par stéréovision.

Le *deuxième chapitre* présente un état de l'art des différents éléments utiles pour l'élaboration de l'observatoire de deux roues. Il recense, dans un premier temps, les différentes technologies de capteurs vidéo. Il décrit, en particulier, leur modélisation qui servira ensuite à leur calibration. Puis, des méthodes d'extraction fond forme répondant à la problématique de scène routière sont présentées. Enfin, un état de l'art des différentes méthodes de calcul de cartes de profondeur en stéréovision dense est rédigé. Cet état de l'art a pour but d'établir de quelles méthodes existantes on peut s'inspirer pour répondre à notre problématique.

Dans le *troisième chapitre*, l'extraction fond/forme, une étape restant indispensable au traitement, même en stéréovision, est approfondie. Nous présentons, dans un premier temps, les limites des méthodes usuelles d'extraction vis à vis de notre problématique et nous analysons quels sont les éléments pouvant détériorer la détection des véhicules. Face aux difficultés rencontrées pour comparer efficacement ces méthodes, nous proposons une base de vidéos simulées, avec leurs vérités associées. On peut alors effectuer une comparaison des méthodes, et ainsi choisir celle qui est la plus adaptée à notre besoin.

Dans le *quatrième chapitre*, l'adaptation de la stéréovision à notre problématique est étudiée. Dans un premier temps, on présente les facteurs gênant le traitement en stéréovision sur une scène routière. Nous étudions ensuite le dimensionnement du système de prise de vues stéréoscopiques ainsi que le traitement nécessaire à sa calibration. Enfin, nous recherchons le traitement stéréo le plus adapté à notre système. Pour cela, trois approches sont présentées : les deux premières sont assez proches de celles identifiées par l'état de l'art en y ajoutant quelques astuces pour les adapter ; la troisième, assez innovante est décrite en détail. Enfin, une comparaison à partir de données simulées est proposée selon le même principe que celui étudié dans le chapitre précédent pour l'extraction fond/forme. On conclut sur l'approche qui offre les meilleurs résultats.

Enfin, le *cinquième chapitre* présente la mise en œuvre du système et les résultats obtenus, dans un premier temps à partir de données simulées, puis à partir d'essais réalisés sur sites réels. Ces résultats permettent de voir quels sont les points à améliorer et les limites des approches choisies. Surtout, ils montrent que la stéréovision apporte réellement un plus pour la détection des deux-roues, et justifient ainsi l'approche choisie.

PROBLÉMATIQUE 1

SOMMAIRE

DANS ce chapitre, nous expliquons les enjeux de la thèse. Dans un premier temps, on fait un état des lieux de la circulation des deux-roues en milieu urbain, les véhicules concernés et leur place actuelle. Puis, nous abordons les problèmes de sécurité liés à la cohabitation avec les autres véhicules empruntant les mêmes voiries. Partant de là, on montre l'intérêt de créer des systèmes de comptage des deux-roues, et on analyse les limites des systèmes existants. On introduit les avantages espérés d'un traitement du problème par stéréovision, ce qui amène à ce travail de thèse.

1.1 LES DEUX-ROUES EN VILLE

1.1.1 Types de véhicules

En milieu urbain, une grand nombre de véhicules sont regroupés sous l'appellation "deux-roues". Ces véhicules peuvent êtres classés en différentes catégories sans ou avec moteur. Il est important de tous les prendre en compte pour pouvoir bien comprendre les enjeux liés à leur sécurité.

1.1.1.1 Les deux-roues non motorisés (ou vélos)

Les vélos sont très présents en milieu urbain. Leur faible coût, y compris d'utilisation, l'absence d'autorisation ou formation nécessaire pour les conduire, les rend particulièrement attractifs. De plus, ces véhicules peuvent circuler dans des zones interdites aux deux-roues motorisés (certains parcs, rues piétonnes ...), ce qui en fait des véhicules particulièrement bien adaptés à la circulation en centre ville. Certains vélos possèdent un système d'aide électrique pour réduire l'effort du conducteur, néanmoins cette motorisation très faible ne change pas réellement leur comportement et ne permet pas de considérer ces vélos comme des deux-roues motorisés.

1.1.1.2 Les deux-roues motorisées (2RM)

Catégories

Les deux-roues motorisées (2RM) représentent la part des deux-roues la plus variée. Ils sont, du fait de leur motorisation, capables de vitesses relativement importantes. On peut citer les catégories les plus nombreuses :

- ⇒ Le **vélomoteur** (solex) est relativement peu courant, mais subsiste dans des déplacement urbains, d'où l'intérêt de le citer. Ce véhicule de très faible cylindrée se situe entre le cyclomoteur et la bicyclette. Le moteur est placé sur la roue avant, et permet de propulser le véhicule à faible vitesse ou d'aider le conducteur dans le pédalage.
- ⇒ Les **cyclomoteurs** sont des deux-roues de faible cylindrée, ne pouvant, en temps normal, pas dépasser les 45km/h et pouvant être conduits par des conducteurs jeunes ou des conducteurs non formés. Leur place dans le trafic des deux-roues est en baisse depuis quelques années. Ils ne sont pas autorisés à circuler sur certaines zones telles que les quatre voies et les périphériques. Ils ont une dimension du même ordre qu'un vélo.
- ⇒ Les **scooters** (figure 1.1) sont très populaires en ville. Ils se caractérisent par des roues assez petites d'un diamètre ne dépassant pas 36 cm et une forme de châssis assez particulière formant un plancher permettant de poser les pieds. Ces véhicules motorisés sont très populaires du fait de leur maniabilité. Ils prennent la place, au fil des années, des cyclomoteurs pour les modèles de faible cylindrée. Ils se déclinent suivant différentes puissances et cylindrées.
- ⇒ Les **motocyclettes**, plus communément appelées motos, sont les deux-roues les plus utilisés pour les déplacements sur grande distance du fait de leur vitesse. Elles se déclinent sous différentes formes et différentes cylindrées. En milieu urbain, elles sont utili-

FIGURE 1.1 – *Scooters de petite (à gauche) et grosse cylindrée (à droite)*

sés principalement pour des trajets mixtes (urbain et péri urbain) leur permettant de circuler à vitesse normale dans une circulation fluide.

⇒ Les motos peuvent posséder un **side-car** afin de transporter une personne supplémentaire. Ce type de véhicule reste néanmoins très marginal dans le trafic routier. Dans notre cas, il n'est donc pas nécessaire de s'y attarder, d'autant plus que, du fait de la largeur supplémentaire apportée par le side-car, le comportement de ce véhicule s'apparente plus à celui d'un VL.

⇒ D'autres véhicules sont apparus récemment comme par exemple des **tricycles à moteur** (figure 1.2). Ces véhicules peuvent être classé, malgré leurs trois roues, dans la catégorie des deux-roues motorisés, du fait de leur comportement et leur capacité à se faufiler dans le trafic urbain comme les autres deux-roues. Les trois roues augmentent la stabilité, ce qui donne un certain succès auprès de conducteurs peu habitués aux deux-roues.

FIGURE 1.2 – *Scooter à 3 roues*

Législation

Les 2RM sont classés en quatre catégories principales auxquelles s'appliquent les mêmes règles :

⇒ Les cyclomoteurs de moins de 50 cm^3 ne peuvent circuler au delà de 45 km/h. La conduite de ces véhicules est autorisée dès 14 ans avec un brevet de sécurité routière (BSR). Les véhicules concernés sont les cyclomoteurs, les solex, et les scooters de faible cylindrée.

⇒ Les *motocyclettes légères* (MTL) possèdent une cylindrée pouvant aller jusqu'à 125 cm^3 ainsi qu'une puissance ne dépassant les 11 kW. Pour conduire ces véhicules, le conducteur doit posséder un permis A1 pour une conduite dès 16 ans, un permis A à partir de 18 ans, ou le permis B validé depuis quelques années. Cette catégorie regroupe principalement des scooters et motos.

⇒ Les *motocyclettes de grosse cylindrée* (MTT) concernent les véhicules d'une cylindrée supérieure à 125 cm^3. Cette catégorie se décompose de la manière suivante :
- la catégorie MTT1 concerne les véhicules de 25 kW de puissance maximale. Leur conduite requiert un âge d'au moins 18 ans avec le permis A.
- la catégorie MTT2 concerne les véhicules dont la puissance va jusqu'à 73,6 kW. La réglementation pour pouvoir conduire ces engins est plus restrictive puisque le conducteur doit posséder le permis A et 21 ans, ou moins mais alors avec ce permis validé depuis au moins deux ans.

⇒ Les *tricycles à moteur* (TM) : Les constructeurs ont utilisé l'absence de législation pour lancer ce type de produit. En effet, pour pouvoir les conduire, la loi ne demande que d'avoir le permis B et 18 ans, comme pour les véhicules légers (VL). Cette souplesse a fait le succès de ces véhicules ces dernières années, permettant à des titulaires du permis B de circuler en deux-roues sans besoin d'une formation spécifique.

1.1.2 Place des deux-roues dans le trafic urbain

On remarque depuis quelques années une augmentation de la circulation des deux-roues en milieu urbain. Par exemple, le nombre de motocyclettes a augmenté de 25 % entre 1998 et 2008. A Paris, la tendance est la même puisque, à partir de données provenant de 6 sites, l'observatoire de la ville a remarqué une augmentation de 38 % de ce trafic.

En ce qui concerne le vélo, une étude provenant de l'observatoire des déplacements de l'agglomération lyonnaise [2] fait un état des lieux de son importance dans le trafic urbain en 2006. De cette étude, quelques points marquant sont à retenir :

⇒ la part modale du vélo est de 1,7% dans le grand Lyon, ce chiffre montant à 2,8% dans le centre ville.

⇒ entre 1995 et 2006 l'utilisation des différents moyens de transport à évolué de la façon suivante :
- voiture : -14%
- transport en commun : +11%
- marche à pied : -2%
- vélo : +148% (évolution supérieure à +280% dans le centre ville)

⇒ le trafic des vélos présente de fortes fluctuations mensuelles avec des périodes creuses l'hiver et en août (trafic global faible quelles que soient les catégories).

L'étude montre également que les vélos libre service (VLS), apparus depuis 2005 sur l'agglomération, ont pris rapidement une place importante pour

atteindre plus de 30% du trafic total de vélos, preuve d'une réelle réponse apportée par cette offre.

De façon plus générale les VLS se développent dans toutes les agglomérations françaises. À Rennes, ville pionnière, ce service est apparu en 1998. Son succès a amené l'offre à se développer partout en France comme à Paris en 2007 (voir figure 1.3). Fin 2008 une vingtaine de villes françaises possédaient leur offre de vélos libre service. Les villes comme les usagers trouvent leur compte dans cette offre. Pour les usagers, elle permet un transport adaptable en termes de trajet et d'heure, idéale pour les petits déplacements, pour une somme modique. Pour les villes, cet investissement permet de promouvoir l'utilisation du vélo, les VLS ayant un effet d'entraînement sur l'usage du vélo en général. De plus, ils permettent une offre complémentaire de transport dans la ville. Enfin, malgré un coût assez élevé puisqu'on l'estime à 100 millions d'euros en France en 2008, les avantages en termes de gain pour les cyclistes, de décongestion des transport en commun et de la voie routière, et de respect de l'environnement rendent le bilan plutôt positif [3]. Etant donné le coup de pouce donné au développement des vélos via les VLS, il est cohérent pour ces villes d'étudier la problématique des deux-roues pour garantir la sécurité des usagers et ne pas en freiner le développement.

FIGURE 1.3 – *Station de vélo libre service à Paris*

1.1.3 Peut-on aller plus loin ?

Comme on l'a dit, on constate une augmentation du trafic des deux-roues ces dernières années. En effet, ceux-ci répondent parfaitement à différentes problématiques de déplacement urbain. Les villes peuvent avoir tout intérêt à favoriser le développement de ce moyen de transport, mais avant tout, il est important de se poser la question : en favoriser le développement est-il réaliste et envisageable ? Il faut aussi se demander, si une

ville peut accepter un trafic de deux-roues nettement plus important sans rencontrer de gros problèmes de sécurité.

Pour répondre à ces questions, nous partons de deux exemples provenant de pays étrangers où la pratique des deux-roues est plus développée qu'en France et de façon nettement différente :

⇒ Les Pays-Bas avec un trafic de vélos faisant référence

⇒ Le Vietnam et son impressionnant trafic de 2RM

1.1.3.1 Cas des Pays-Bas

Aux Pays-bas, le vélo est presque un emblème national. Une étude provenant du ministère des transports néerlandais dresse un état de la place du vélo dans ce pays en 2009 [4] : elle constitue un record européen. A titre de comparaison, cette même étude indique que 5% des déplacements en France sont réalisés à l'aide d'une bicyclette, alors qu'au Pays-Bas, ce chiffre monte à 26%. Ceci illustre bien la marge de progression possible en France. On pourrait se demander si un trafic trop important n'est pas facteur d'une hausse de l'accidentologie. L'étude, effectuée pour quelques pays européens, semble démontrer le contraire (voir figure 1.4) puisqu'elle montre qu'une hausse du trafic de vélos diminue le risque de décès par kilomètre parcouru, ceci étant dû en partie à une meilleure attention de tous les usagers de la route envers les vélos. Les conducteurs de voiture étant également des cyclistes, ils sont sensibles à la problématique. Cette première raison est éventuellement transposable aux 2RM puisque, plus ils sont présents, plus les autres usagers doivent être vigilants quant à leur présence possible.

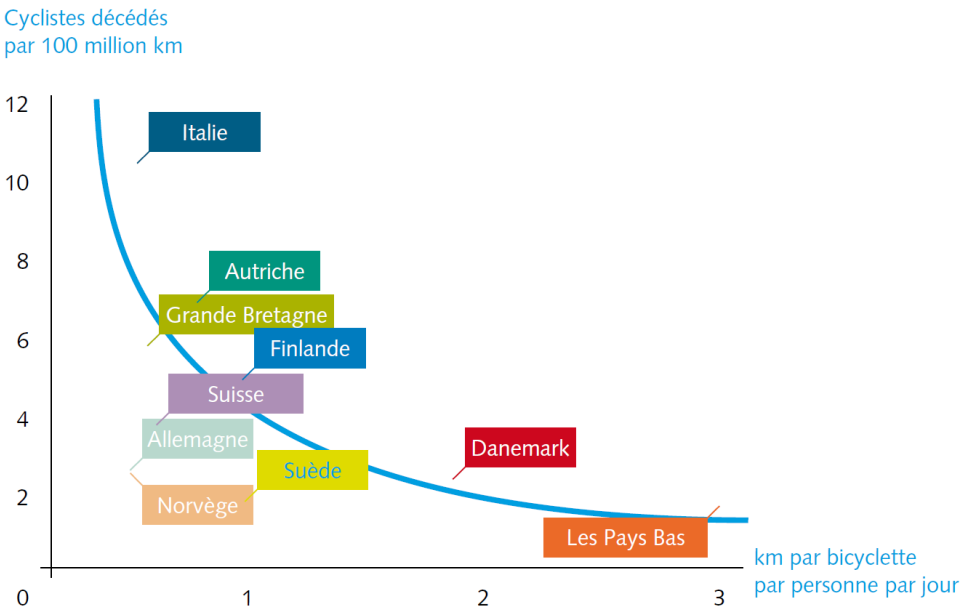


FIGURE 1.4 – *Comparaison du risque en vélo pour différents pays européens*

La problématique du vélo étant prise en compte depuis de très nombreuses années, de nombreux aménagement comme les pistes cyclables ou les parkings à vélo (voir figure 1.5) ont été réalisés. Il peut donc être intéressant de s'en inspirer. Citons par exemple Houten, une ville nouvelle

pour laquelle tout a été pensé dès la conception pour favoriser les déplacements à pied ou à vélo [5]. Cette ville est entourée par un périphérique que les automobilistes sont obligés d'emprunter pour se rendre d'un quartier à l'autre, et dans la ville elle même la circulation en voiture n'est facile que pour l'accès aux quartiers. Les autres voies à l'intérieur de la ville sont volontairement aménagées pour rendre le déplacement très compliqué en voiture, et ainsi décourager de son utilisation. Différentes règles de circulation sont également mises en place comme par exemple : la priorité est donnée aux vélos sur les intersections ; dans les lignes droites les voitures n'ont pas le droit de les dépasser ; de nombreuses zones de la ville sont interdites aux voitures ... Grâce à ceci, toute la place est donnée à l'intérieur de la ville aux pistes cyclables et la tranquillité y est assurée.

FIGURE 1.5 – *Parking à vélo devant la gare centrale d'Amsterdam*

1.1.3.2 Exemple du Vietnam

Le Vietnam est un autre exemple intéressant à analyser. Dans ce pays, le mode de déplacement roi dans les villes est le 2RM, au point de rendre le trafic impressionnant (voir figure 1.6). A Hô-Chi-Minh-Ville, par exemple, un état des lieux [6] indique qu'en 2006, il y avait environ 300 000 voitures pour 3 millions de deux-roues (principalement des 2RM). Cette situation est assez problématique puisque les pollutions sonores et aériennes générées par les deux-roues sont très importantes (phénomène accentué par le pétrole qui est en général de mauvaise qualité). Ce développement des 2RM vient du faible coût d'achat des motos grâce à l'importation de modèles venant de Chine, qui rend ce mode de transport abordable pour une bonne partie de la population, contrairement aux voitures. En outre, pendant de longues années le bus était très peu développé dans la ville, l'usage de la moto s'y est donc bien implanté et il semble aujourd'hui difficile d'inverser le processus. Enfin, les gens ont perdu l'habitude de marcher ou de

rouler en vélo au profit des 2RM qui sont utilisés même pour les trajets courts. Tout ceci empêche le développement des transports en commun et le développement socio-économique de la ville, et contribue à donner une mauvaise image de celle-ci.

FIGURE 1.6 – *Circulation en 2RM au Vietnam*

L'exemple du Vietnam, où le développement du parc de 2RM s'est fait de façon anarchique et sans être accompagné d'un effort d'adaptation des infrastructures, montre bien les dérives que peuvent amener la circulation des 2RM si aucune politique de gestion du trafic urbain n'est mise en place. Pour ces raisons, un 2RM prenant la place d'une voiture mono-occupant est une bonne chose, mais en aucun cas un 2RM ne doit se substituer à un vélo, ou un piéton pour les petits trajets. Il est donc important de favoriser chaque moyen de transport selon les déplacements et les zones de la ville. Cet enjeu est primordial dans la politique d'aménagement urbain.

1.1.4 Solutions apportées par les deux-roues

Les deux-roues sont particulièrement bien adaptés pour la circulation en milieu urbain et ceci pour des raisons qui diffèrent selon le type.

De manière générale, la ville est le lieu d'embouteillages et de congestions fréquents, en particulier aux heures de pointes (entrée et sortie de travail). Les deux-roues, grâce à leur faible dimension, peuvent se faufiler dans ces ralentissements de trafic et ainsi en réduire l'ampleur et l'impact. L'augmentation de trafic de deux-roues est donc intéressante pour les villes (au même titre que celle des transports en commun). Les aménagements supprimant des voies de circulations au profit de voies bus ou de pistes cyclables vont dans ce sens.

Les deux-roues non motorisés ont l'avantage de ne pas nécessiter de carburant, ce qui en fait un mode de déplacement économique, non polluant et peu bruyant du fait de l'absence de motorisation. Ils sont très utiles en milieu urbain ainsi qu'en centre ville où ils peuvent avoir un comportement se rapprochant de celui du piéton. Leur vitesse en situation non

embouteillée est inférieure à celle des autres véhicules, mais devient supérieure en cas de trafic saturé.

Les deux-roues motorisés ont l'avantage de pouvoir se déplacer comme tout autre véhicule dans les conditions non ralenties de trafic mais aussi de se faufiler quand celui-ci devient plus saturé. C'est pourquoi ces véhicules sont très populaires pour les déplacements domicile-travail, où les distances sont encore importantes. De plus, la motorisation rendant la progression plus facile qu'avec un vélo, ce confort est également un critère de choix par rapport aux deux-roues non-motorisés.

Enfin, un autre avantage des deux-roues vient d'une plus grande facilité de stationnement en centre ville et d'une certaine tolérance vis à vis du stationnement gratuit (bien qu'illégal) pour les 2RM.

1.1.5 Frein au développement des deux-roues

Malgré les avantages listés précédemment, le développement des deux-roues est freiné pour différentes raisons.

Les derniers chiffres de l'ONISR (Observatoire national interministériel de la sécurité routière) [7] attestent que malgré une forte baisse de la mortalité des usagers de la route en France, la mortalité des usagers de deux-roues reste très importante, le risque d'être tué par kilomètre parcouru étant très nettement supérieur pour ces usagers (voir table 1.1). On peut également remarquer (voir table 1.2) qu'en milieu urbain, les conducteurs tués sont principalement des usagers de deux-roues (toutes catégories confondues), contrairement au milieu rural. Et ceci pour différentes raisons :

- ⇒ la cohabitation avec les autres usagers de la route n'est pas toujours simple.
- ⇒ les infrastructures sont parfois inadaptées aux deux-roues.
- ⇒ les usagers des deux-roues (ainsi que les piétons) sont beaucoup plus vulnérables. La faible vitesse en ville, réduit le nombre de morts usagers de VL, mais a nettement moins d'impact en ce qui concerne les conducteurs de deux-roues.

Catégories d'usagers	Conducteurs et passagers tués dans le véhicule	Conducteurs et passagers tués par milliard de véhicules × km	Risque relatif
Cyclomotoristes	248	109,5	20,7
Motocyclistes	704	103,9	19,6
Usagers VL	2 117	5,3	1
Usagers PL	65	2,7	0,5
Usagers TC	4	1,4[1]	0,3[1]

TABLE 1.1 – *Estimation du risque par catégories d'usagers en 2010 (conducteurs et passagers)*

La faible dimension des véhicules ainsi que leur vitesse généralement différente du reste du flux de circulation, les rendent peu visibles. Pour ces raisons, le sentiment d'insécurité est très présent chez les conducteurs de deux-roues. Ce phénomène est accentué par le fait que les infrastructures sont souvent mal adaptées à la circulation de véhicules évoluant à vitesses

	Milieu urbain	Rase campagne	Total
Piétons	346	139	485
Cyclistes	59	88	147
Cyclomotoristes	123	125	248
Motocyclistes	272	432	704
Véhicules légers	288	1829	2117
Autres	45	246	291
Total	**1133**	**2859**	**3992**

TABLE 1.2 – *Nombre de personnes tuées par catégories d'usagers selon le milieu en 2010*

différentes. Ce sentiment d'insécurité tout à fait réel se confirme au niveau de la mortalité.

La baisse de mortalité pour les conducteurs de véhicules à quatre roues s'explique par un changement de comportement des conducteurs, mais aussi par une meilleure protection des usagers d'une voiture suite aux efforts menés par les constructeurs automobiles pour améliorer la sécurité de leurs produits. Ces améliorations ne profitent pas aux deux-roues. Ceux-ci restent très vulnérables et le moindre choc, même à faible vitesse, peut leur être fatal. La législation impose le port du casque aux conducteurs de deux-roues motorisés, mais ce port est facultatif pour les cyclistes. Ainsi un choc, même à vitesse très réduite peut être mortel.

1.2 LES FACTEURS D'INSÉCURITÉ DES DEUX-ROUES EN VILLE

Les statistiques sur l'accidentologie des deux-roues font ressortir plusieurs situations dangereuses dues au comportement des usagers :

⇒ L'inexpérience de certains conducteurs
⇒ Les remontées de files
⇒ Les problèmes de cohabitation avec les autres véhicules

1.2.1 Manque de maîtrise des jeunes conducteurs

Du fait de la moins bonne stabilité des deux-roues, la perte de contrôle est également une cause d'accident. Ceci peut-être accentué par le fait que la conduite de certains 2RM ne demande pas de formation préalable, d'où des conducteurs pouvant moins bien maitriser leur véhicule. Pour illustrer ceci, des statistiques indiquent que 16% des accidents de 2RM n'impliquent qu'un 2RM seul. Ces accidents sont en général lourds de conséquence, puisqu'en Ile-de-France, par exemple, 38% des conducteurs de 2RM qui se sont tués en 2009, ont été victimes d'un accident où ils étaient seuls.

Une autre étude [8] va également dans ce sens. Elle fait ressortir deux paramètres :

⇒ L'âge du véhicule : les deux premières années du véhicule, en Ile-de-France, les conducteurs de 2RM sont beaucoup plus impliqués dans les accidents corporels (voir figure 1.7).
⇒ Le nombre d'années de permis : les conducteurs de 0 à 3 ans de permis sont impliqués dans 49% des accidents corporels pour les 2RM de cylindrée supérieure à 125 cm^3.

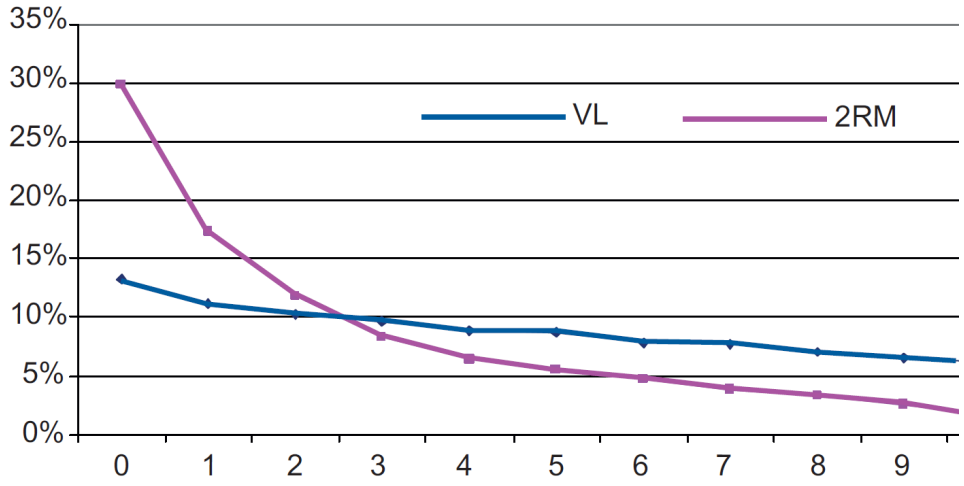


FIGURE 1.7 – *Accidents corporels selon l'âge du véhicule*

1.2.2 Comportement des motocyclistes (remontée de file)

Les remontées de file interviennent quand la circulation est ralentie, principalement sur les grandes routes à plusieurs voies comme les périphériques. Les 2RM, profitent de leur faible dimension pour se faufiler entre les files de voitures (voir figure 1.8). Des accidents se produisent alors quand les véhicules changent de voie et ne voient pas la moto qui arrive et à laquelle ils coupent la route. Cette pratique, longtemps interdite mais encore très fréquente, a été depuis peu intégrée au code de la route afin de la cadrer.

FIGURE 1.8 – *Remontée de file par des 2RM sur le périphérique parisien*

1.2.3 Cohabitation

Les deux-roues, motorisés ou non, sont particulièrement soumis aux angles morts. Ceux-ci sont accentués pour les véhicules de dimension importante comme les poids lourds (voir figure 1.9). 40% des cyclistes accidentés par un poids lourd se trouvaient dans son angle mort. C'est pourtant la collision avec ce type de véhicule qui est la plus préjudiciable pour un cycliste. Des situations typiques de dangers liés aux angles morts ont été identifiées principalement lorsque un véhicule tourne (voir figure 1.10) ou change de voie dans le cas des remontées de file.

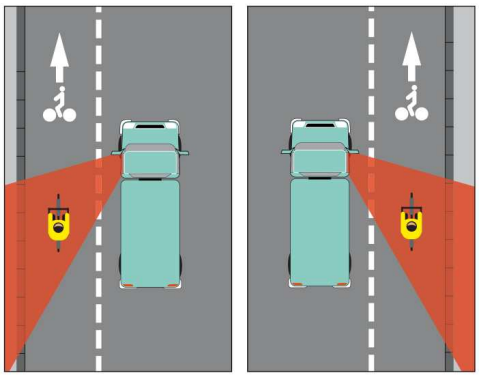


FIGURE 1.9 – *Cyclistes dans l'angle mort*

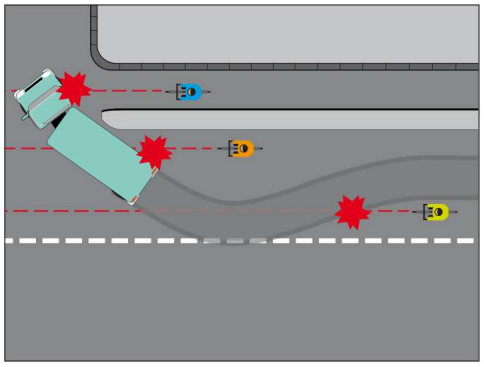


FIGURE 1.10 – *Situations d'accident causés par l'angle mort*

Dans certaines rues à sens unique, la circulation est autorisée dans les deux sens pour les cyclistes. Dans ce cas, un nombre important d'accidents ont lieu entre cyclistes et piétons qui ne regardent que d'un coté avant de traverser. Du fait de leur faible dimension, les deux-roues sont, en général, moins bien perçus que les autres véhicules. C'est encore plus le cas pour les vélos : ils ne possèdent pas de moteur, sont donc silencieux, ce qui est un avantage pour le confort de vie en ville mais un inconvénient les rendant plus difficilement détectables par les autres usagers de la route. Enfin, toujours pour les vélos, le manque de visibilité peut se retrouver accentué en conditions nocturnes avec des lumières faibles voir en leur absence totale. Pour ces raisons, la loi a récemment imposé le port du gilet réfléchissant pour les cyclistes hors agglomération. En agglomération ce dernier n'est pas imposé par la loi en raison de l'éclairage mais il est vivement conseillé.

Les dernières études statistiques [8] mettent bien en avant ce problème de cohabitation en ville pour les accidents entre 2RM et VL. En effet, comme le montre la figure 1.11, si la responsabilité est à peu près égale en rase campagne, celle-ci relève plus des VL dans les grandes agglomérations, ce qui traduit bien un problème de perception des 2RM par les autres véhicules.

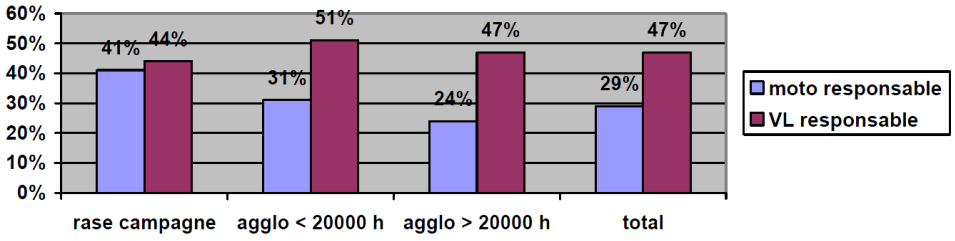


FIGURE 1.11 – *Responsabilité dans les accidents moto - véhicule léger*

Dangers pour les vélos en intersection

Les intersections sont des zones particulièrement problématiques pour les vélos en ville, en raison de leur faible visibilité et leur vitesse réduite. Les cas les plus accidentogènes recensés [9] sur les intersections sont (voir figure 1.12) :

1. La collision à angle droit
2. Les manœuvres de tourne à gauche des VL
3. Les manœuvres de tourne à droite des VL
4. Les manœuvres de tourne à gauche des vélos

Pour résumer, les problèmes liés au comportement viennent principalement :

⇒ d'un manque de vigilance quant à la présence possible de deux-roues

⇒ d'un manque de prise de conscience pour les deux-roues des situations dangereuses dans lesquelles ils peuvent se trouver

Pour prévenir les risques d'accidents, il est important que les usagers (deux-roues et autres) préviennent longtemps à l'avance d'une manœuvre qu'ils vont effectuer en mettant leur clignotant ou en tendant le bras, chose parfois négligée. Toutefois, si ces efforts de bonne conduite de la part des usagers sont importants, ils ne sauraient se substituer à une politique d'aménagement des infrastructures, comme celle décrite pour les Pays-Bas, par exemple.

1.3 LES AMÉNAGEMENTS POUR LA SÉCURITÉ DES DEUX-ROUES

Pour améliorer la sécurité et permettre une meilleure cohabitation des différentes catégories de véhicules, différents aménagements s'avèrent efficaces lorsqu'ils sont bien choisis et bien conçus. Nous en donnons quelques exemples ci-dessous.

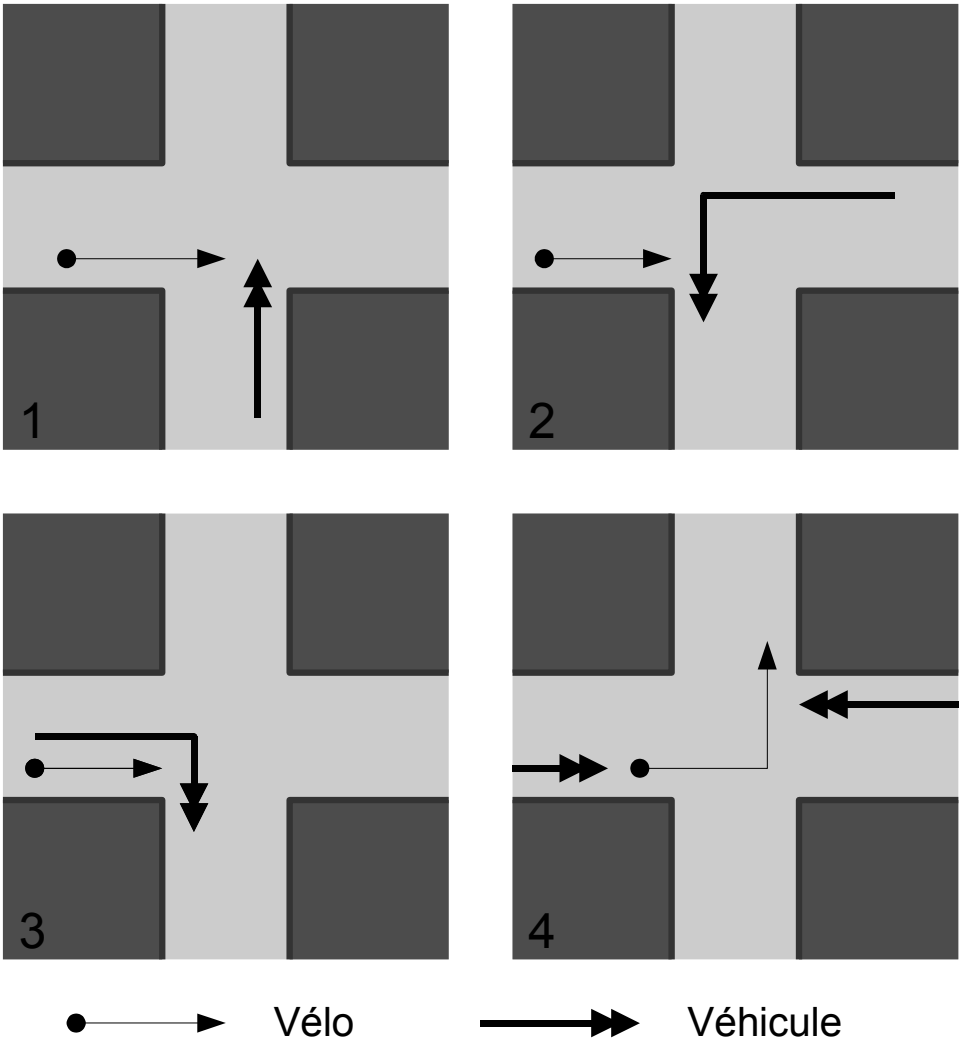


FIGURE 1.12 – *Situations dangereuses pour les vélos sur les carrefours*

1.3.1 Pistes cyclables

Les vélos ne circulant pas à la même vitesse que les voitures, leur cohabitation n'est pas forcément aisée sur route en milieu urbain. La différence de vitesse crée un danger. En cas de rue étroite, le dépassement n'est pas toujours simple.

Pour améliorer cette situation, de nombreux aménagements effectués en ville consistent à séparer le flux de vélos des autres véhicules. Ces aménagements permettent également de donner un confort de conduite aux cyclistes du fait de la sensation de sécurité accrue.

En fonction de l'aménagement possible, deux types de solutions sont envisageables :

⇒ La piste cyclable est une voie de circulation à part entière réservée aux vélos ; elle permet de préserver le cycliste de la cohabitation avec d'autres véhicules plus rapides que lui. Elle est bien adaptée pour des grandes sections comportant peu d'intersections, et donc, en premier lieu, aux zones péri-urbaines. En revanche, en fin d'aménagement, la situation peut devenir dangereuse car les conducteurs

peuvent ne pas s'attendre à retrouver des cyclistes sur la même voie qu'eux.

⇒ La bande cyclable est incluse dans la voie, elle est réalisée par un marquage au sol indiquant une zone de la route réservée au cycliste. Elle a l'avantage de permettre une mise en place assez simple dans des endroits où la place n'est pas suffisante pour une piste à part entière (par exemple les centres-villes) tout en gérant la mixité des modes de transport sur la même chaussée. En revanche, rien n'empêche un véhicule de circuler dessus. Certains véhicules profitent de cette bande pour stationner illégalement, ce qui rend la circulation dangereuse pour le cycliste qui doit alors la quitter.

Malgré l'apport positif de ces aménagements, les pistes et bandes cyclables posent des problèmes sur les intersections et ronds-points, les vélos risquant de se faire couper la route. Ceci a conduit les villes à penser d'autres aménagements.

1.3.2 Aménagements des intersections

Les SAS à vélos

Afin de répondre aux problèmes d'angles morts, dangereux pour les cyclistes dans les intersections, de nombreux carrefours sont équipés de SAS [10] (voir figure 1.13). Ceux-ci permettent aux cyclistes de se placer en tête de file, devant les autres véhicules. Ces aménagements, très répandus dans différents pays européens, furent introduits dans de nombreuses grandes villes de France avant d'être intégrés officiellement au code de la route en 1998. Le cycliste est positionné devant les autres véhicules, ce qui lui permet donc de :

⇒ pouvoir démarrer avant les autres véhicules
⇒ mieux être identifié par les véhicules
⇒ mieux voir la situation pour mieux anticiper les déplacements au démarrage.
⇒ pouvoir se placer plus facilement pour tourner à gauche tout en étant vu
⇒ être moins soumis aux gaz d'échappement lors du démarrage

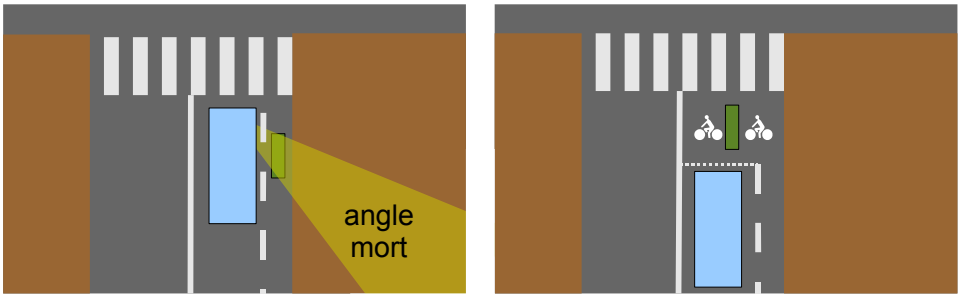


FIGURE 1.13 – *Intersection avec (à gauche) et sans SAS (à droite)*

Tourne à gauche (TAG)

Comme nous avons pu le voir, tourner à gauche sur une intersection est très dangereux pour un deux-roues, ceci étant accentué par la densité de trafic et le nombre de files à l'intersection. Grâce au SAS, cette opération se retrouve simplifiée et davantage sécurisée. On peut alors effectuer un TAG suivant deux approches différentes, plus ou moins utilisées en fonction des pays ou des cyclistes (voir figure 1.14).

⇒ Le tourne à gauche **direct** est effectué quand le cycliste se place sur la gauche du SAS, et emprunte la voie à gauche après avoir laissé passer le véhicules venant de face. L'emploi du SAS sécurise la manœuvre mais contraint le cycliste à rester immobile au centre du carrefour un certain moment.

⇒ Le tourne à gauche **indirect** revient à supprimer l'opération de tourne à gauche grâce à l'utilisation de deux SAS. L'usager du deux-roues se place sur le SAS comme pour tourner à droite, puis se place sur le SAS de la voie à sa droite. Ainsi quand le feu passera au vert il n'aura qu'à rouler tout droit. Cette approche est utilisée dans quelques pays, notamment au Danemark. Néanmoins, elle oblige le cycliste à attendre deux fois aux feux.

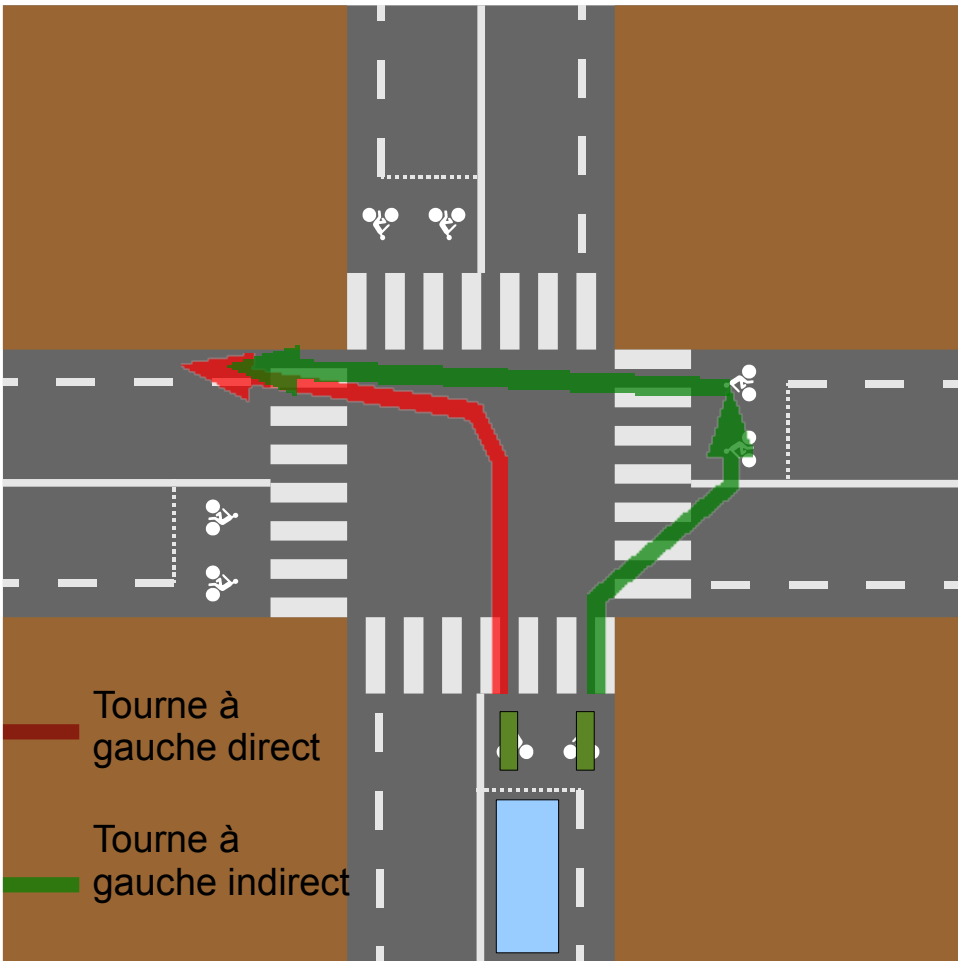


FIGURE 1.14 – *Tourne à gauche direct et indirect*

Tourne à droite (TAD)

Lors des démarrages aux intersections, des accidents se produisent quand un véhicule veut tourner à droite et coupe ainsi la route d'un 2 roues qu'il n'avait pas vu, placé sur une piste cyclable. Pour éviter ce risque, la solution est de faire en sorte que les véhicules et les deux-roues ne puissent démarrer au même moment sur l'intersection. C'est dans ce but que, dans certains carrefours, le tourne à droite a été instauré. Si un cycliste veut tourner à droite à un feu, une signalisation lui indique qu'il peut démarrer même si le feu est rouge. Le cycliste doit néanmoins s'assurer que la voie sur laquelle il s'engage est libre comme pour un "cédez le passage". Cette mesure, d'abord en essai dans quelques villes françaises, a été officiellement intégré au code de la route (article R. 415-15, décret du 12 novembre 2010).

1.3.3 Limites d'efficacité des aménagements

Difficultés d'aménagement

La problématique de prise en compte des cyclistes dans l'aménagement urbain est assez récente. Par conséquent, dans certaines zones, l'adaptation consistant à ajouter des pistes ou bandes cyclables et autres signalisations n'est pas forcément possible. La configuration des lieux peut également contraindre à un aménagement assez complexe et difficilement compréhensible pour le cycliste.

Aménagement discontinu

Les pistes et les bandes cyclables sont bénéfiques pour limiter les risques d'accidents, néanmoins, ils peuvent avoir un effet négatif, notamment en cas d'aménagement partiel de certaines zones. Dans ce cas, les automobilistes peuvent oublier la présence possible de cyclistes. De plus, le cycliste peut se sentir en sécurité du fait des aménagements et baisser sa vigilance quand il revient sur une voie non aménagée.

Aménagements des ronds-points

Les carrefours giratoires sont souvent assez dangereux pour les cyclistes. En effet, dans le cas où la route donnant sur un rond-point est longée par des pistes cyclables, l'automobiliste prête moins attention au vélos. Mais sur le rond-point les pistes cyclables disparaissent, le cycliste se retrouve donc "lâché" dans une situation demandant une vigilance toute particulière. Il reste sur l'extérieur du rond-point mais s'expose au risque d'être renversé par une voiture voulant prendre une sortie que le cycliste n'emprunte pas.

1.3.4 Limites des connaissances sur l'efficacité des aménagements

La connaissance du comportement des deux-roues en ville n'est pas parfaite, un exemple le montre bien : le code de la route n'impose pas aux cyclistes d'utiliser les pistes ou bandes cyclables (sauf dans les cas où cela est explicitement indiqué), d'où un doute sur leur réelle efficacité dans tous

les cas. Les exemples listés précédemment parlent largement des aménagements urbains pour les cyclistes. Les 2RM, eux, sont très peu pris en compte. Certains utilisent les aménagements pour cyclistes mais comme ils ne leur sont pas destinés, cela cause un problème de cohabitation avec les cyclistes. D'une manière générale, les différentes études portant sur ce sujet montrent bien que celui-ci est très complexe et lié à de nombreux facteurs.

Les études actuelles, menées seulement sur quelques sites du fait de la complexité de leur mise en place (comptage manuel), ou à partir des statistiques venant des procès verbaux en cas d'accident, ne sont pas d'une grande fiabilité et ne permettent pas une étude plus poussée de l'influence de tous les facteurs intervenants dans l'accidentologie des deux-roues sur les différentes configuration des carrefours en ville. Il est donc nécessaire de disposer de moyens plus systématiques et plus faciles à mettre en œuvre pour élargir les bases expérimentales de ces études.

1.4 Le comptage des deux-roues en ville

1.4.1 Intérêt

Comme expliqué précédemment, l'usage des deux-roues présentant un réel intérêt en termes de décongestion de trafic, les agglomérations ont tout intérêt a en favoriser le développement. Néanmoins, afin d'aménager au mieux les voies urbaines pour garantir la sécurité de tous, ces agglomérations ont besoin d'en savoir plus en ce qui concerne le trafic des deux-roues (heures de circulation, part du trafic des deux-roues, manœuvres effectuées par ceux-ci, situations dangereuses ...). Citons en exemple les villes de Nantes et Paris qui sont très demandeuses d'outils pour élaborer des statistiques servant à l'aménagement urbain.

1.4.2 Systèmes existants et limites

A l'heure actuelle, une multitude de systèmes existent pour la surveillance du trafic. Nous allons les présenter et voir s'ils répondent ou non à notre problématique de détection et de comptage des deux-roues. Pour cela nous partons d'une étude listant les systèmes existants [11], et nous examinons si leur transposition au comptage des 2RM est possible.

1.4.2.1 Systèmes de comptage usuels

Les tubes pneumatiques placés perpendiculairement à la voie et comprimés par le passage de véhicules, sont utilisés depuis longtemps pour effectuer des comptages de véhicules. Grâce à deux tubes, il est possible de connaitre la vitesse du véhicule. Cependant, ce système, largement utilisé en raison de sa facilité de mise en œuvre, n'est pas bien adapté à notre problème de deux-roues qui se faufilent entre les voitures, sur des zones que ne couvrent pas toujours ces tubes.

Les boucles magnétiques sont également largement utilisées pour la mesure de la circulation. Leur fonctionnement est simple : les boucles sont

insérées dans la chaussées et génèrent un champ magnétique, celui-ci est modifié par le passage d'un véhicule, qui peut ainsi être détecté. Ces systèmes ont de sérieux inconvénients : ils sont destructifs envers la chaussée, lourds à mettre en place, et ils ne détectent pour le moment pas efficacement les 2RM à cause de leur faible dimension et la grande variabilité de leur trajectoires.

Les radars peuvent également servir à la détection des véhicules, ils ont l'avantage de ne pas être destructifs envers la chaussée et peu sensibles aux conditions météorologiques. Ils fonctionnent en utilisant une onde électromagnétique qui permet d'estimer la vitesse du véhicule à partir du changement de fréquence du signal par effet doppler lors de la réflexion de l'onde sur le véhicule. Cependant ces systèmes ne permettent d'observer la situation que ponctuellement, sans aucune possibilité d'analyse de leur trajectoire. En outre, la discrimination entre les deux-roues et les autres véhicules n'est pas fiable.

1.4.2.2 Systèmes optiques

Les systèmes utilisant la vision sont de plus en plus utilisés, ils présentent en effet de nombreux avantages puisqu'ils sont :

⇒ peu coûteux,
⇒ facilement montables et démontables,
⇒ non destructifs pour la chaussée.

Un système existe déjà permettant de suivre les véhicules sur les virages. Il est composé de deux caméras filmant la courbe, ainsi que d'un télémètre à balayage laser permettant d'obtenir des points d'échos correspondant aux véhicules (voir figure 1.15). Un autre système, dérivé du pre-

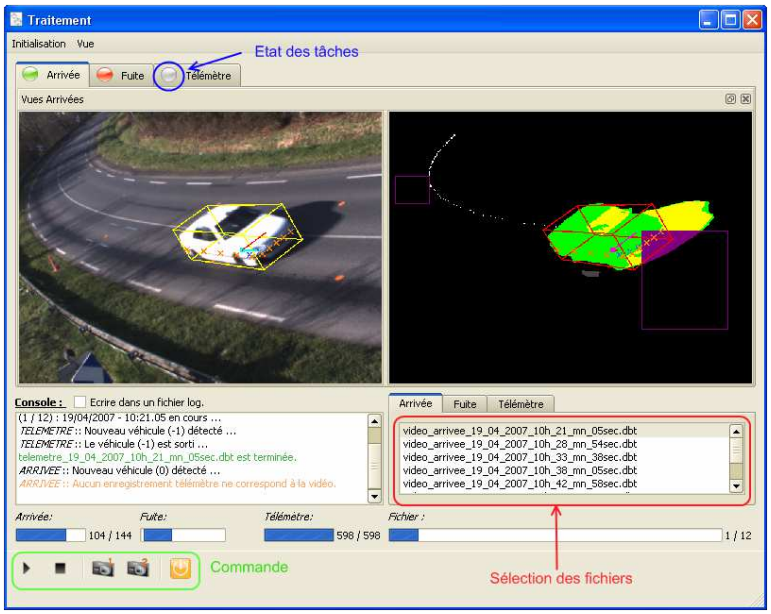


FIGURE 1.15 – *Système de suivi à base de vision + télémètre*

mier, a été développé pour suivre les véhicules sur un rond-point, et ainsi

être capable de fournir des informations sur l'écoulement du trafic, sous la forme de matrices origine-destination. Ce système fonctionne sur le même principe que le précédent mais en n'utilisant qu'une seule caméra fisheye montée sur un mat dressé au centre du rond-point. Ces deux systèmes se basent sur une vision monoculaire de la scène, et utilisent des algorithmes d'extraction fond/forme ainsi que des algorithmes de suivi par filtre à particule pour reconnaître les véhicules. Cette solution est parfaitement fonctionnelle pour des véhicules de taille importante tels que les voitures, mais pose de gros problèmes pour la détection des deux-roues, dont celui posé par les ombres portées (voir paragraphe 3.1).

Une approche alternative [12] permet de détecter et classifier les véhicules, et offre des taux de reconnaissance montant à 98% pour les motos. Toutefois, elle oblige à placer la caméra perpendiculairement à l'axe de la route pour ne pas subir l'influence de l'ombre (voir figure 1.16), les masquages y sont donc très fréquents. En outre, la position transversale réelle du véhicule est difficilement mesurable et, le champ de vision est très réduit. A noter également d'autres approches utilisées pour la détection de piétons par stéréovision [13] permettant, elles aussi, de s'affranchir du problème des ombres portées.

Une autre approche [14] [15] prétend à la détection des voitures et des 2RM en utilisant les dimensions des véhicules et les contours. Néanmoins, l'article n'annonce aucun résultat en termes de qualité de la classification, et tous les exemples sont issus de scènes où il n'y a pas d'ombre.

D'autres travaux [16] annoncent une détection des deux-roues, mais ils ne portent que sur le cas autoroutier (seulement des 2RM), ou dans les intersections où la circulation n'est pas dense avec différentes catégories de deux-roues [1] [17]. Dans tous ces cas, les exemples donnés ne contiennent pas d'ombres (voir figure 1.17). Les deux derniers cas insistent d'ailleurs sur les problèmes rencontrés face aux ombres.

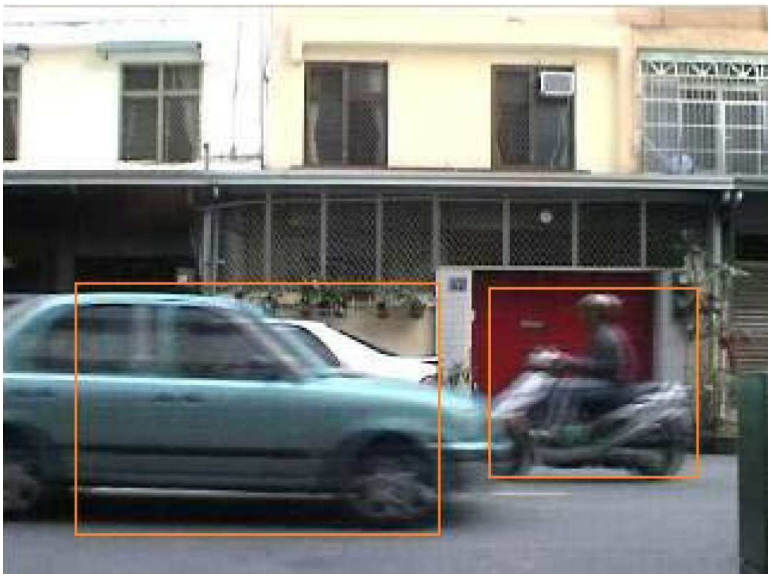


FIGURE 1.16 – *Système de détection avec caméra perpendiculaire*

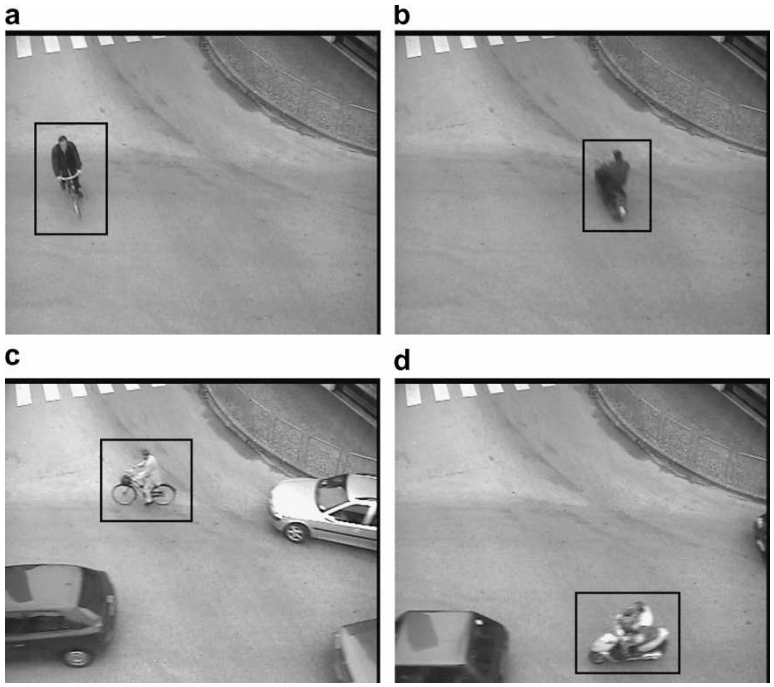


FIGURE 1.17 – *Détection des deux-roues sur une intersection d'après [1]*

Les systèmes industriels sont presque exclusivement des systèmes basés sur la vidéo. A l'heure actuelle, on peux citer différentes approches développées par les sociétés Optelecom nkf, ACIC - Intelligent Vision Systems, Capflow, Visioway,... Ces systèmes, basés sur des caméras de surveillance, permettent le comptage des véhicules sur des voies prédéfinies, principalement sur autoroutes. Mais, aucun de ces systèmes n'annonce de résultats pour la détection des 2RM.

1.4.3 Attentes envers les systèmes optiques

On a pu voir que les systèmes optiques sont maintenant très utilisés pour la surveillance du trafic. Ces systèmes sont assez récents et, par conséquent, encore à perfectionner. Ils ont, comme avantage par rapport aux autres technologies, de percevoir une information très riche qui devrait en théorie être suffisante pour différencier les types de véhicules, faire du comptage, et suivre leurs trajectoires, puisque cette information est suffisante pour qu'un œil humain puisse discriminer et suivre les véhicules.

La jeunesse de ces systèmes est la raison de leurs limitations actuelles et laisse des perspectives quant aux améliorations possibles. Partant de ce constat, la technologie de capteurs basés sur la vision semble la plus judicieuse pour traiter notre problématique. On a pu voir qu'à l'heure actuelle, et malgré les applications possibles grâce à la vision, la détection des deux-roues reste délicate en raison des problèmes d'ombres que les méthodes actuelles ont du mal à séparer du véhicule. La stéréovision pourrait permettre de détecter, donc d'éliminer, les zones d'ombres. La piste d'un système basé sur la vision stéréo pour détecter les deux-roues semble donc prometteuse.

1.5 MOTIVATION ET OBJECTIF DE LA THÈSE

On a vu dans ce chapitre que les deux-roues ont toute leur place en ville ; cette place va d'ailleurs continuer à s'accroître pour contribuer à la résolution des problèmes de pollution aérienne et sonore (par le développement des deux-roues non motorisés), d'engorgement de trafic et de stationnement. Mais, la conduite en deux-roues reste assez dangereuse, ce qui contribue à freiner le développement de ce mode de transport. La difficulté réside principalement dans la cohabitation avec les autres catégories de véhicules, qui circulent à des vitesses différentes, et qui peuvent avoir du mal à bien voir les deux-roues. En outre, un accident qui serait minime dans le cas de deux voitures, peut être beaucoup plus lourd de conséquences pour les conducteurs de deux-roues, les statistiques de la sécurité routière l'attestent. Pour apporter des solutions à cette problématique, on manque encore de connaissances sur le comportement des deux-roues et sur les aménagements susceptibles de procurer le maximum de sécurité pour les usagers, et ainsi de favoriser le développement du deux-roues en ville.

Un besoin crucial aujourd'hui est donc d'observer davantage et mieux le comportement des deux-roues en ville et leur cohabitation avec les autres véhicules dans le trafic urbain, en particulier dans les zones typiquement accidentogènes (intersections, ronds-points, ligne droite saturée avec remontée de file). Or peu de solutions existent pour procéder à ces observations.

C'est pour cette raison qu'a été créée le projet ANR METRAMOTO[1], dont l'objectif est d'apporter des solutions pour la détection et le suivi des deux-roues motorisés, en étudiant différentes technologies intrusives :

⇒ capteurs piézo et boucle éléctromagnétique

⇒ magnétomètre

et non intrusives :

⇒ télémétrie laser

⇒ analyse d'images.

Le travail de thèse s'est donc naturellement inscrit dans ce projet débuté en 2010 pour traiter cette dernière technologie.

Dans ce contexte, la thèse vise à permettre de créer des outils d'analyse du trafic des véhicules (deux-roues et autres) pour mieux connaitre leurs interactions. Étant donnée la multitude de situations dangereuses, on ne peut tout traiter. On se focalisera donc sur une zone particulièrement problématique et assez complexe, à savoir les intersections urbaines, tout en cherchant à créer un système utilisable dans d'autres configurations. Étant donné l'absence actuelle de solutions efficaces pour détecter les deux-roues, l'objectif que nous nous sommes fixés est d'être capable de distinguer ces deux-roues (motorisés ou non) des autres véhicules et des piétons. En analysant la vitesse et les dimensions des véhicules, on devrait être capable de différencier un vélo d'une moto, mais en ce qui concerne les catégories intermédiaires, telles les cyclomoteurs ou les scooters, ce point devient plus complexe, on fait donc le choix de ne pas chercher à les différencier. Le but de ce travail est donc d'apporter un outil basé sur la vision avec utilisation des méthodes de stéréovision pour distinguer efficacement les véhicules en vue de leur classification.

1. http ://metramoto.pagesperso-orange.fr/

1.6 Principe de l'approche proposée

Au cours des dernières années, des systèmes appelés "observatoires de trajectoires" ont été développés pour suivre les trajectoires des véhicules circulant sur des virages dangereux ou sur des giratoires. Or ces systèmes, basés sur la vision monoculaire et, dont les images sont traitées à l'aide de méthodes d'extraction fond/forme, ont pour limite de ne pouvoir détecter les deux-roues efficacement. Le but visé par cette thèse est donc d'adapter ce concept, en y apportant des modifications nécessaires pour suivre tous les véhicules, dont les deux-roues, et être capable de les classifier, principalement dans une intersection en milieu urbain.

La difficulté pour détecter les deux-roues dans les précédents observatoires vient de leurs caractéristiques géométriques : ils sont plus étroits, plus hauts et plus "informes" que les voitures. L'ombre portée au sol est, en général, confondue par les systèmes monovisions, avec le véhicule, qui peut donc sembler beaucoup plus gros. Ainsi l'erreur de classification d'un deux-roues est très courante, surtout en cas de circulation dense. Le fait de pouvoir différencier le véhicule de son ombre est donc indispensable pour détecter efficacement les deux-roues.

De plus, du fait de la densité du trafic urbain, il faut être capable de discriminer des véhicules circulant en flot dense (roulant côte à côte, ou se suivant de près dans les congestions à faible vitesse) et qui pourraient être confondus par une approche monovision.

Partant de ce constat, l'approche de stéréovision semble parfaitement adaptée pour obtenir des informations spatiales sur la position des véhicules. Le principal inconvénient de la stéréovision, qui la rend difficile à mettre en œuvre dans certains cas, vient de sa lourdeur de calcul. Toutefois, dans notre cas, le traitement n'a pour but que d'apporter des informations, et notamment des statistiques, pour aider les gestionnaires à prendre des décisions sur l'aménagement routier. Il ne doit pas nécessairement être effectué en temps réel.

L'inconvénient que rencontrent les méthodes d'extraction fond/forme (identification des ombres) devant être résolu par cette approche, il reste néanmoins indispensable de la conserver pour :

- ⇒ Soulager le calcul de stéréovision en présélectionnant les zones à traiter
- ⇒ Différencier les véhicules des objets de l'environnement

Par définition, un système stéréo possède deux caméras, qui doivent être positionnées à une hauteur importante pour permettre une vue plongeante sur le carrefour à observer. De plus, dans le carrefour, les véhicules pouvant provenir de différentes voies, il est important de choisir des caméras ayant un champ de vision très large.

Il est apparu intéressant, notamment pour résoudre des problèmes de stabilité, de positionner les deux caméras sur un même axe vertical. C'est une solution qui a été peu étudiée jusqu'ici. Nous verrons donc, dans les chapitres suivants comment cette mise en œuvre est réalisable, et quelles sont les adaptations nécessaires à ce traitement, qui est schématisé dans la figure 1.18.

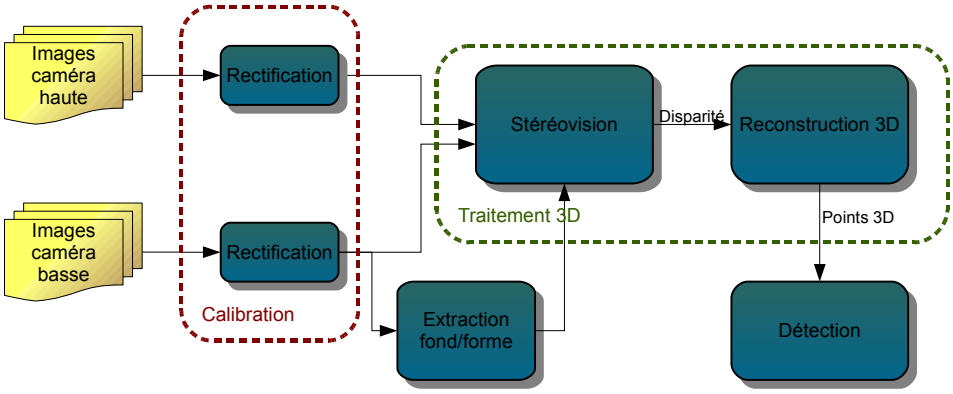


FIGURE 1.18 – *Chaine globale de traitement envisagée*

1.7 ENJEUX

Le travail effectué dans cette thèse doit répondre à deux enjeux :

⇒ un enjeu pratique : créer un système permettant d'effectuer un observatoire de trajectoire capable de détecter les deux-roues en intersections urbaine. Pour cela, le système doit être capable de catégoriser les véhicules et les suivre dans des situations complexes de circulation dense et lente. Il doit, *in fine*, permettre d'aider à une meilleure compréhension des dangers liés aux deux-roues en ville, et ainsi d'apporter de meilleures solutions pour leur sécurité.

⇒ un enjeu scientifique : développer un système permettant l'acquisition et le traitement des séquences en stéréovision. Pour cela, il faut choisir le type de caméras à utiliser, le positionnement relatif de celles-ci pour une prise de vue en stéréovision et la calibration spécifique à mettre en place pour ce type de système. L'emplacement idéal du système sur la zone à acquérir doit aussi être analysé. Il faut enfin réaliser le traitement des images tenant compte du système choisi pour arriver à répondre aux besoins cités dans l'enjeu pratique.

1.8 CONCLUSION

Nous avons pu voir en détail dans cette partie la problématique liée à l'usage des deux-roues en ville, et à son intérêt grandissant. Ce mode de déplacement a l'avantage de porter réponse à de nombreux problèmes de circulation urbains (congestion, non pollution pour les vélos) ; les agglomérations y portent donc le plus grand intérêt. Néanmoins, à cause des problèmes de sécurité et de cohabitation avec les autres véhicules, ce mode de déplacement ne se développe pas aussi vite qu'il le pourrait. A partir de ces enjeux de sécurité, et en mettant en lumière le réel besoin d'une meilleure compréhension des comportements des usagers, nous avons montré l'intérêt de développer des systèmes capables d'analyser le trafic des deux-roues en milieu urbain. La problématique est vaste ; notre choix s'est donc porté sur le contexte correspondant aujourd'hui aux enjeux les plus forts, à savoir les intersections.

Partant de ce constat, nous avons étudié les systèmes existants et leurs

limites, dans le but de comprendre pourquoi ces systèmes ne répondaient pas correctement à la problématique des deux-roues. Nous en sommes arrivés à l'idée de développer un système basé sur la stéréovision.

Le chapitre suivant fait un état de l'art sur les méthodes d'extraction/forme existantes, les types de capteurs vidéo ainsi que les méthodes de traitement en stéréovision pour voir lesquelles sont les mieux adaptées à notre problématique. Nous pourrons nous en inspirer pour la conception du système, conception que nous étudierons dans un chapitre ultérieur.

ÉTAT DE L'ART 2

SOMMAIRE

NOUS avons explicité dans le chapitre précédent la problématique liée aux deux-roues en ville. De cette étude, les besoins en systèmes permettant d'analyser leur trafic ainsi que les lacunes des systèmes actuels ont été mis en évidence. Ceci a permis de conclure à l'intérêt d'un système basé sur la vision, utilisant un traitement en stéréovision après une phase d'extraction fond/ forme. Pour pouvoir élaborer ce système, l'étape préalable que nous rapportons dans ce chapitre est la réalisation d'un état de l'art sur les différents capteurs vidéo existants permettant la prise de vue, pour ensuite aborder la problématique de prise de vue et de traitement en stéréovision. Enfin, nous étudions des méthodes d'extraction fond/forme applicables au domaine routier. Cette étude permettra de voir, quelles sont les méthodes existantes qu'on peut utiliser ou dont on peut s'inspirer pour notre application.

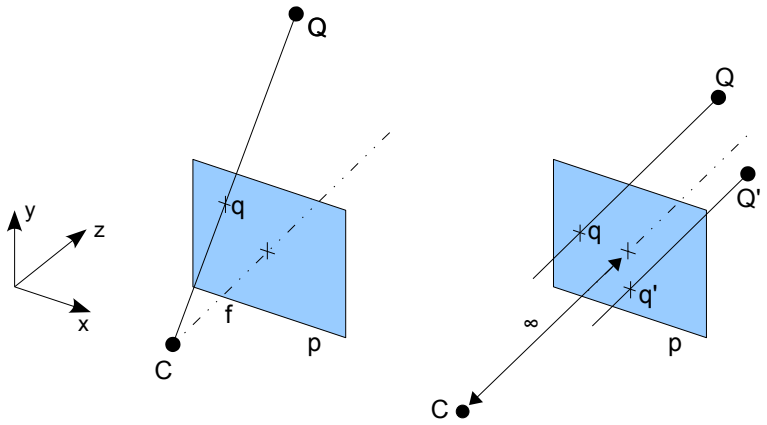


FIGURE 2.1 – *Modes de projection : sténopé (à gauche) et orthographique (à droite)*

2.1 PRISE DE VUE VIDÉO

2.1.1 Modes de projection

Avant de parler des différents types et caractéristiques des capteurs, il est intéressant de voir les modes de projection utilisés en vision pour représenter la projection d'un point de l'espace 3D sur un plan d'image 2D.

2.1.1.1 Projection sténopé

Ce type de projection est le plus utilisé puisque qu'il modélise le fonctionnement de la plupart des optiques en vision où tous les rayons lumineux provenant de la scène sont projetés de façon à converger sur un point appelé point focal. Son principe est le suivant : Soit C le centre de projection (ou point focal), et p le plan des cellules sensibles, situé à une distance f (appelée longueur focale) de C. Soit Q un point de l'espace 3D, sa projection sera le point q se situant à l'intersection de la droite (CQ) et du plan p (voir figure 2.1).

2.1.1.2 Projection orthographique

Dans ce modèle, il n'y a pas de point focal, ou alors, il peut être considéré comme situé à une distance infinie du capteur C. Ceci a pour conséquence de projeter tous les rayons selon des droites parallèles, et de restreindre considérablement le champ de vision. La distance des objets au plan p n'influe pas sur leur taille projetée. Ce type de projection ne permet pas de représenter la majorité des optiques, mis à part les optiques dites *télécentriques* (présentées plus loin en figure 2.8).

2.1.2 Capteurs directionnels

Les capteurs les plus couramment utilisés et les moins coûteux utilisent des longueurs focales situés entre 4 et 100mm. Ils fournissent une image de bonne qualité et très peu déformée. Ces capteurs peuvent être très bien

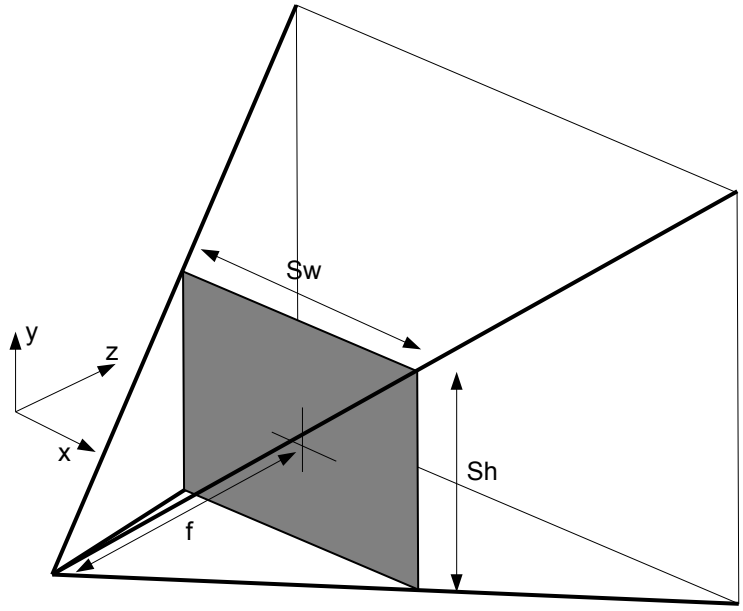


FIGURE 2.2 – *modélisation d'un capteur vidéo classique*

modélisés par le modèle de projection sténopé. En revanche, dans la pratique, les longueurs focales des différentes optiques existantes conduisent à un champ de vision assez réduit qui ne permet de filmer qu'une zone bien précise. L'angle d'ouverture de ce type d'optique dépend de plusieurs paramètres dont, la longueur focale f provenant de l'optique, et la taille du capteur $S_w \times S_h$, tous exprimées en millimètre. Dans ce cas, ce type de capteur peut se modéliser comme le montre la figure 2.2.

L'angle d'ouverture ouv est alors calculé de la façon suivante :

$$ouv_x = 2 \times \arctan \frac{S_w}{2 \times f} \tag{2.1}$$

$$ouv_y = 2 \times \arctan \frac{S_h}{2 \times f} \tag{2.2}$$

L'utilisation de capteurs classiques oblige donc à filmer un carrefour sous différents angles, en positionnant les capteurs de chaque coté de la voie. De plus, tous ces capteurs doivent être synchronisés pour pouvoir traiter toutes les données avec le même référentiel temporel. La contrainte n'étant pas de réaliser un traitement temps réel, ce regroupement de données est faisable. Il reste qu'on souhaite réaliser un système d'acquisition facile à installer et à désinstaller pour des campagnes d'essai de courte durée. L'utilisation de capteurs permettant de couvrir intégralement la zone désirée parait donc préférable pour la détection de véhicules en intersection.

2.1.3 Capteurs omnidirectionnels

2.1.3.1 Capteurs catadioptriques

Principe

Les capteurs catadioptriques sont composés d'une caméra classique ainsi que d'un miroir d'une forme pouvant être parabolique, hyperbolique, conique, plane... Ces capteurs ont pour avantage de pouvoir couvrir de très grandes zones. C'est la raison pour laquelle ils sont utilisés dans de nombreuses applications de robotique [18] [19] [20] où l'on souhaite avoir une vision omnidirectionnelle de la scène sans pour autant avoir plusieurs caméras. La modélisation d'un miroir relativement plat permet de couvrir intégralement la zone souhaitée. En revanche ce type de capteurs a deux inconvénients principaux :

⇒ Les miroirs doivent être parfaits, sans aucune aspérité afin de rendre une image de bonne qualité. De plus, ils doivent posséder une forme bien définie, et donc, être réalisés sur mesure. Cette fabrication demande un travail de grande précision qui a un coût important.

⇒ La caméra et son miroir doivent être parfaitement alignés pour que le résultat soit satisfaisant. Dans le cas d'un traitement en stéréovision les deux caméras doivent également être alignées, augmentant ainsi la complexité de mise en œuvre et de calibration sur site.

Familles de capteurs

Les capteurs à miroir catadioptriques peuvent être classés selon deux familles (voir figure 2.3) :

⇒ **Les capteurs centraux** : Tous les rayons réfléchis vers le point focal de la caméra (*effective pinhole*) passent par un point appelé point de vue unique (*effective viewpoint*) [21], simplifiant grandement le calcul.

⇒ **Les capteurs non-centraux** : Le point de vue unique n'existe pas ici, il faut donc tenir compte de la forme du miroir et de sa dérivée en chaque point pour calculer la réflexion des rayons lumineux vus par la caméra.

L'existence du point de vue unique dépend donc de la forme du miroir, ainsi que du positionnement de la caméra par rapport à celui-ci. Nous allons voir ceci avec quatre familles de miroir existantes.

Type de miroir

En fonction du besoin, différents types de miroirs sont possibles. Leur géométrie ainsi que la position de la caméra vis-à-vis de ceux-ci influent sur le champ de vision du capteur global. Les plus couramment utilisés sont :

⇒ **Le miroir plan** (figure 2.4) : Ce miroir est le plus simple en termes de réalisation, cependant il n'offre aucun avantage dans notre cas par rapport à un capteur classique puisque le champ de vision n'est pas augmenté. L'inconvénient de ce type de capteur est que la caméra est visible dans l'image et qu'elle masque une partie du champ de vue. Quelques travaux utilisent néanmoins ce type de miroir en

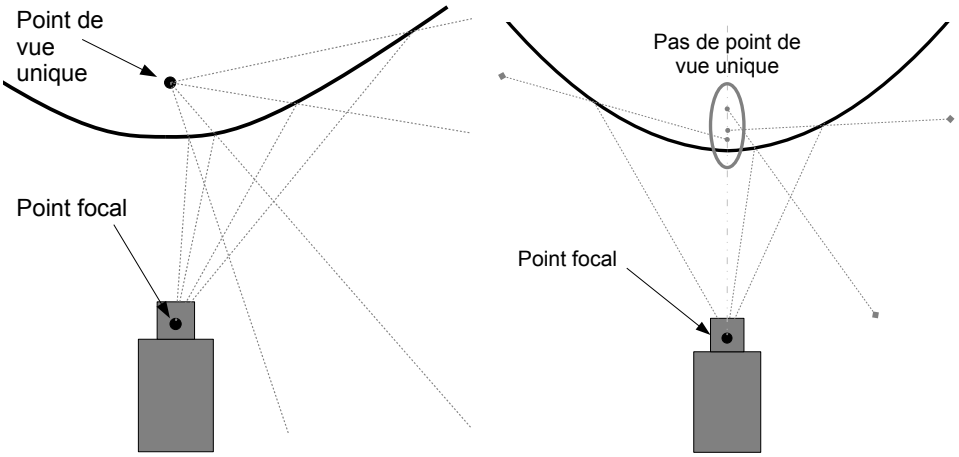


FIGURE 2.3 – *Capteur central et non central*

combinant plusieurs miroirs plans pour augmenter le champ de vision [22] et [23] et ainsi s'adapter à l'observation d'un rond-point et de ses voies d'accès.

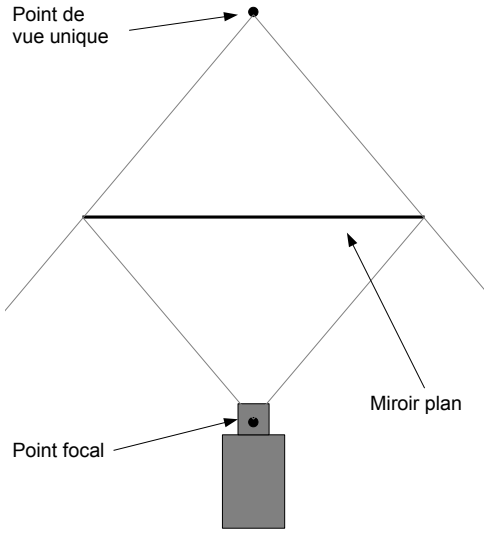


FIGURE 2.4 – *Miroir plan*

⇒ **Le miroir conique** (figure 2.5) : Du fait de sa géométrie, ce miroir permet de n'acquérir que la zone autour du capteur, ce qui est adapté et très utilisé dans de nombreuses applications de robotique [24] [25] [26]. Dans notre cas, il n'est pas adapté car il présente une zone aveugle sous forme d'un cône dans l'axe du capteur. A noter qu'un point de vue unique théorique peut apparaître confondu avec le point focal de la caméra au sommet du cône, mais cela ne présente aucun intérêt à cet emplacement.

⇒ **Le miroir hyperbolique** (figure 2.6) : Une hyperbole possède deux nappes ayant chacune un foyer. Si le miroir correspond à une des nappes, son foyer sera alors le point de vue unique à la condition que le point focal de la caméra soit placé dans le foyer de la seconde nappe de l'hyperbole. Cette propriété rend ce type de capteur assez populaire [18] [19] [20].

⇒ **Le miroir parabolique** (figure 2.7) : Du fait de sa géométrie, il pos-

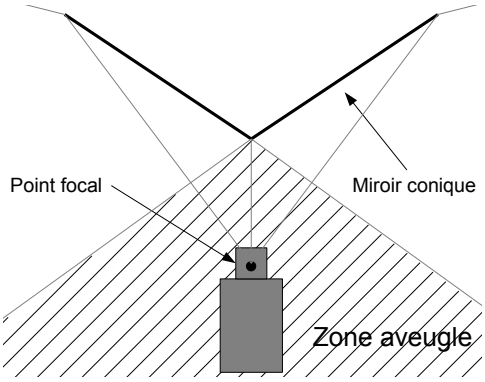


FIGURE 2.5 – *Miroir conique*

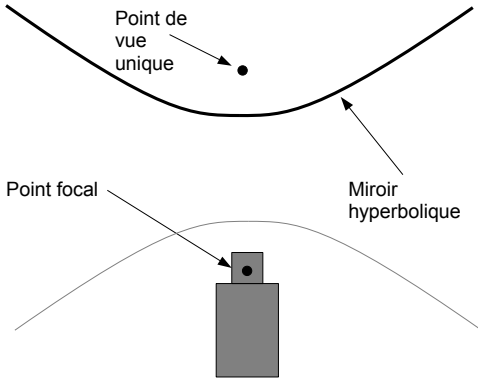


FIGURE 2.6 – *Miroir hyperbolique*

sède un seul foyer qui correspond au point de vue unique. Le point focal pourrait être considéré comme présent mais à une distance infinie. Dans le cadre d'une projection orthographique, la contrainte de distance de la caméra n'existe pas. Ceci simplifie grandement la mise en place puisque l'image obtenue ne dépend pas de l'éloignement au miroir [27] [28], mais contraint à l'utilisation d'une optique télécentrique (projection orthographique, voir partie 2.1.1) de la largeur du miroir, voir figure 2.8.

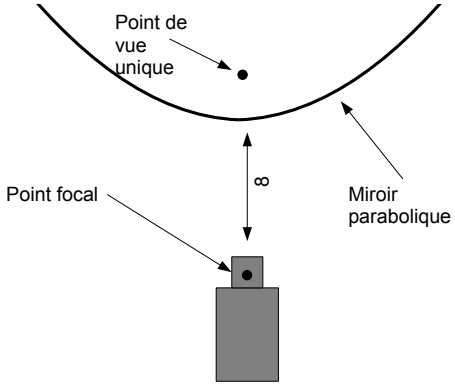


FIGURE 2.7 – *Miroir parabolique*

A noter, que d'autres approches utilisent des combinaisons de plusieurs miroirs pour s'adapter précisément à une scène routière [23], ou pour permettre un traitement en stéréovision en utilisant le point de vue unique de

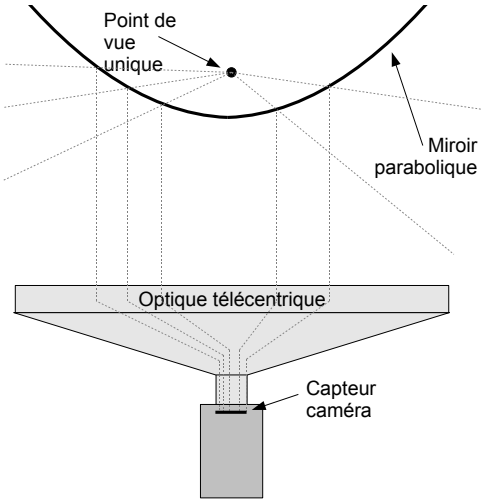


FIGURE 2.8 – *Miroir parabolique avec optique télécentrique*

chaque miroir d'un jeu de miroirs, et ainsi avoir différents points de vue de la scène.

2.1.3.2 Capteurs fisheye

Principe

Les capteurs fisheye sont constitués d'une caméra classique sur laquelle se place une optique dite "fisheye" (figure 2.9) permettant de filmer la scène avec un large champ de vision. Ce dernier est obtenu grâce à une série de lentilles intégrées à l'optique (voir figure 2.10) qui sont dimensionnées de manière à faire converger tous les rayons lumineux provenant du champ d'ouverture vers le capteur de la caméra. L'image obtenue est assez similaire à celle obtenue par les capteurs catadioptriques. Ces optiques ont un large champ de vision (voir figure 2.11), généralement 185°, permettant de couvrir toute la zone située dans une demi-sphère (voir figure 2.12). Les paramètres de l'optique sont :

⇒ L'angle d'ouverture ouv

⇒ La longueur focale f

Type de capteur fisheye

Soit f la longueur focale, θ l'angle d'un rayon lumineux arrivant sur le capteur, et r sa distance au centre sur la surface sensible (le capteur de la caméra) (voir figure 2.13). Selon le type de capteur, différentes fonctions de représentation sont possibles :

⇒ fonction linéaire

$$r = f \times \theta \tag{2.3}$$

⇒ fonction orthographique

$$r = f \times \sin\theta \tag{2.4}$$

⇒ fonction equisolide

$$r = 2 \times f \times \sin\frac{\theta}{2} \tag{2.5}$$

FIGURE 2.9 – *Optique fisheye de marque fujinon*

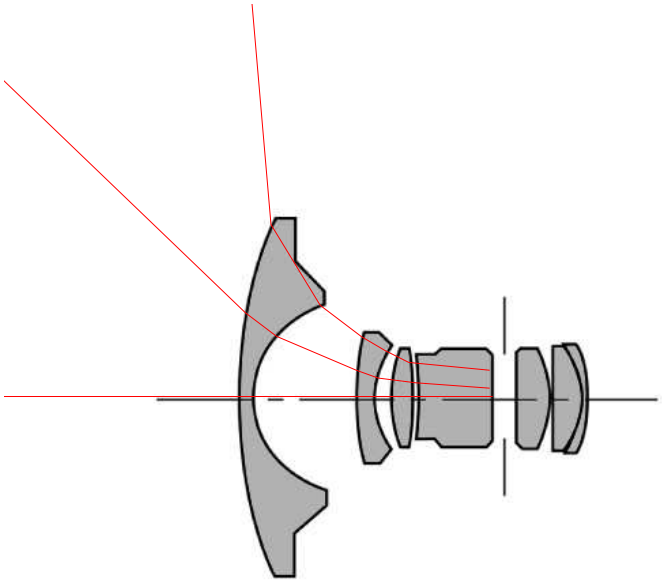


FIGURE 2.10 – *Vue en coupe d'une optique fisheye*

$\Rightarrow$ fonction stéréographique

$$r = 2 \times f \times \tan\frac{\theta}{2} \tag{2.6}$$

Malheureusement ces informations, ne sont généralement pas communiquées par les constructeurs, et la fonction n'est pas nécessairement aussi parfaite que celles décrites ci-dessus. Cette fonction $r = f(\theta)$ sera donc obtenue lors de la phase de calibration du capteur. Pour mémoire, la fonction de représentation d'un capteur optique classique (partie 2.1.2) appelée projection perspective (figure 2.14) se note :

$$r = f \times \tan\theta \tag{2.7}$$

Dans notre cas, si l'on souhaite visualiser les véhicules ayant une hauteur maximale H, à une distance D de 25 mètres autour d'un capteur positionné à une hauteur H_{cam}, l'angle d'ouverture minimal à calculer sera

FIGURE 2.11 – *Image prise par un capteur fisheye*

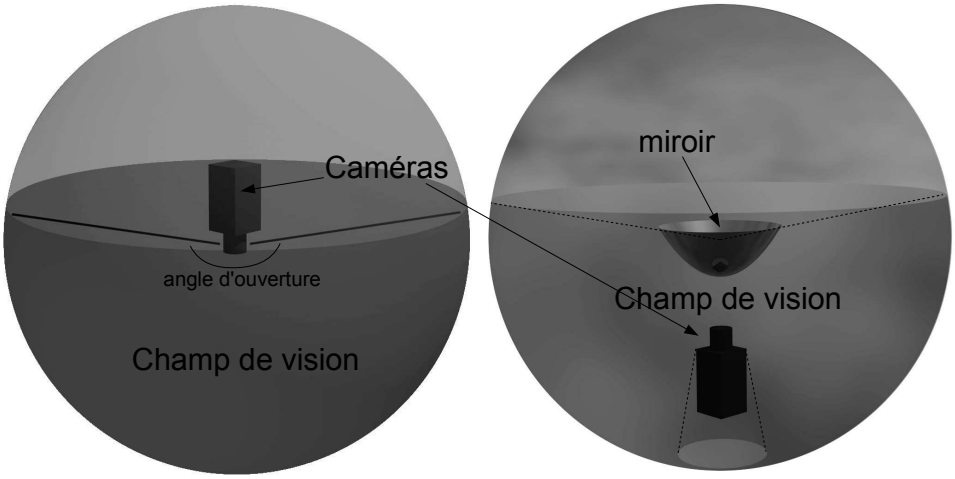


FIGURE 2.12 – *Champ de vision d'un capteur fisheye (à gauche) et à miroir catadioptrique (à droite)*

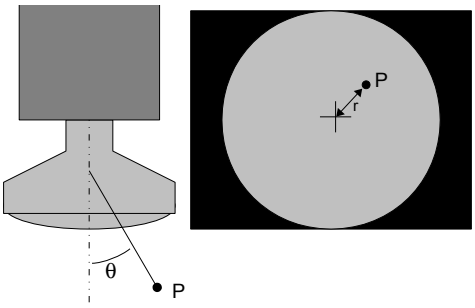


FIGURE 2.13 – *Illustration de r et θ*

donc :

$$ouv_{min} = 2 \times \arctan \frac{D}{H_{cam} - H} \tag{2.8}$$

Ces capteurs ont néanmoins une précision qui varie selon la position sur l'image, la zone centrale étant beaucoup plus précise que la périphérie [28].

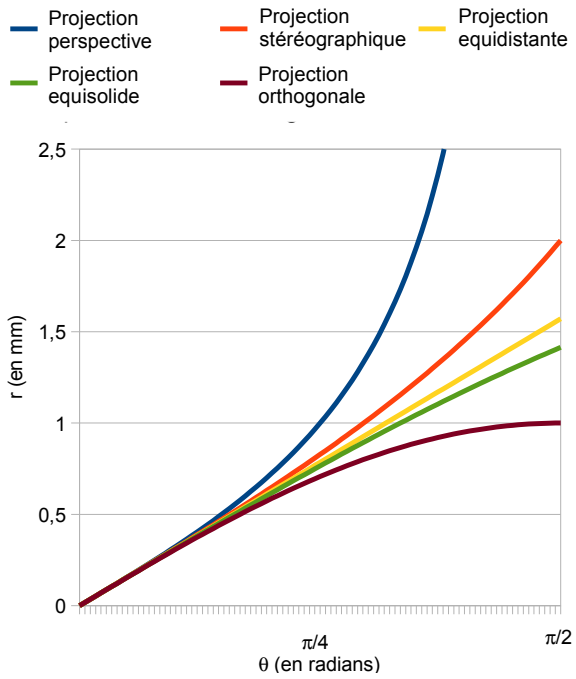


FIGURE 2.14 – *fonctions de projection fisheye : r=f(θ) (avec f=1)*

2.1.4 Modélisation de l'optique

2.1.4.1 Projection sténopé et orthographique

Un capteur vidéo peut être modélisé de la manière suivante pour représenter le passage des coordonnées dans le monde réel (X,Y,Z) aux coordonnées sur le plan de l'image (u,v) :

$$\begin{vmatrix} u \\ v \\ 1 \end{vmatrix} = \underbrace{\begin{vmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}}_{M_I} \underbrace{\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}}_{M_P} \underbrace{\begin{vmatrix} \ddots & & & \vdots \\ & R_{3\times3} & & T_{1\times3} \\ & & \ddots & \vdots \\ 0 & 0 & 0 & 1 \end{vmatrix}}_{M_E} \begin{vmatrix} X \\ Y \\ Z \\ 1 \end{vmatrix} \tag{2.9}$$

avec R une matrice de rotation et T un vecteur de translation

Les paramètres extrinsèques sont inclus dans la matrice M_E. Il s'agit d'un vecteur de translation et d'une matrice modélisant une rotation selon trois dimensions. Il peuvent donc être exprimés par 6 valeurs.

Dans le cas d'une optique standard les paramètres intrinsèques sont α_u, α_v, u_0, et v_0. Ici, les tailles du capteur, la résolution de l'image et la focale sont incluses dans les paramètres, en posant la valeur de la focale à 1 (voir figure 2.15).

Le traitement peut donc être décomposé en trois étapes :

⇒ La rectification extrinsèque grâce à la matrice de changement de base M_E

⇒ La rectification intrinsèque à l'aide de la matrice M_I

⇒ Une étape intermédiaire (matrice M_P dans l'équation 2.9) pour mo-

déliser la projection liée au type de capteur. Cette matrice est différente selon la projection utilisée :

– Pour la projection sténopé par exemple, une matrice identité de taille 4x4 :

$$\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \tag{2.10}$$

– Pour la projection orthographique, la distance des objets selon z ne rentrant pas en compte dans leur représentation, cette matrice est donc remplacée par :

$$\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \tag{2.11}$$

– Pour les optiques omnidirectionnelles à miroir catadioptrique ou à optique fisheye, une simple matrice ne peut suffire. Une approche de modélisation différente est donc nécessaire. Les modèles listés dans le paragraphe suivant permettent de prendre en compte ce type d'optique.

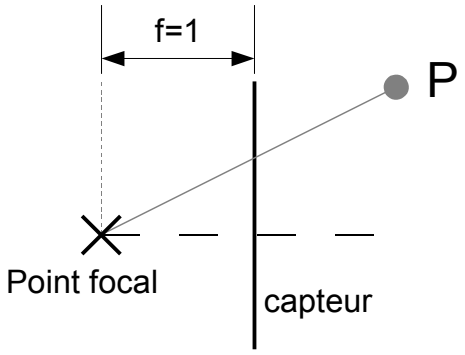


FIGURE 2.15 – *Modèle de caméra standard*

2.1.4.2 Modélisation des capteurs omnidirectionnels

Dans les capteurs omnidirectionnels à miroir catadioptrique ou à objectif fisheye, les distortions radiales sont à prendre en compte. Pour cela, deux types de modélisation existent et sont couramment utilisées pour modéliser les paramètres intrinsèques du capteur.

⇒ Le modèle polynomial
⇒ Le modèle unifié

Modèle polynomial

Comme étudié précédemment, un ensemble caméra et optique fisheye a une relation $r = f(\theta)$. Le but est donc de trouver cette relation sachant

que pour un point de coordonnées X,Y,Z on a :

$$\theta = \arctan\left(\frac{\sqrt{X^2+Y^2}}{Z}\right) \tag{2.12}$$

Une approche consiste à estimer la loi $r = f(\theta)$, par un polynôme d'ordre n tel que :

$$r(\theta) = k_1\theta + k_2\theta^2 + k_3\theta^3 + ... + k_n\theta^n \tag{2.13}$$

Cette approche est largement utilisée [29] [30] [31] [32] [33].

Modèle unifié

Depuis quelques années, un autre modèle introduit dans [34] est couramment utilisé [35] [36].

Dans ce modèle, les paramètres intrinsèques sont ramenés à une valeur unique appelée ξ, qui repose sur l'équivalence entre la projection sur une surface quadratique et la projection sur une sphère. Il permet de modéliser tous les capteurs catadioptriques ayant un point de vue unique, ainsi qu'une grande majorité des optiques fisheye [37].

On considère deux repères : le repère sphérique R_s centré sur O_s, centre de la sphère unitaire S, et le repère caméra R_c dont l'origine est son centre optique, O_c. Ces deux repères sont orientés selon les mêmes axes, mais le repère R_s est décalé d'une valeur ξ selon l'axe z par rapport au repère R_c (voir figure 2.16).

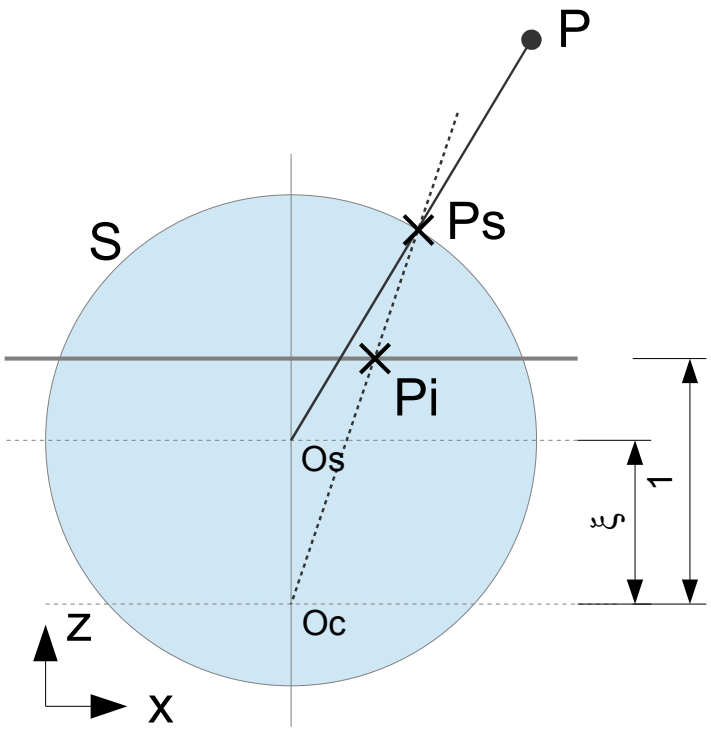


FIGURE 2.16 – *Modèle de caméra sphérique*

Soit un point P dans la scène, ses coordonnées dans R_s s'écrivent :

$$P^{Rs} = \begin{vmatrix} X \\ Y \\ Z \end{vmatrix} \tag{2.14}$$

Ce point est projeté sur la sphère unitaire. On appelle P_s le point d'intersection du segment $[P\ Os]$ avec la sphère S. Les coordonnées de ce point dans R_s s'écrivent donc :

$${P_s}^{Rs} = \begin{vmatrix} X/\lambda \\ Y/\lambda \\ Z/\lambda \end{vmatrix} \text{ avec } \lambda = \sqrt{X^2 + Y^2 + Z^2} \tag{2.15}$$

On exprime ce point P_s dans le repère caméra R_c décalé de R_s d'une valeur ξ selon l'axe z :

$${P_s}^{Rc} = \begin{vmatrix} X/\lambda \\ Y/\lambda \\ Z/\lambda + \xi \end{vmatrix} \tag{2.16}$$

D'où le point P_i, la projection de P_s sur le plan de l'image dans R_c :

$${P_i}^{Rc} = \begin{vmatrix} \frac{X/\lambda}{Z/\lambda+\xi} \\ \frac{Y/\lambda}{Z/\lambda+\xi} \\ 1 \end{vmatrix} = \begin{vmatrix} \frac{X}{Z+\xi\lambda} \\ \frac{Y}{Z+\xi\lambda} \\ 1 \end{vmatrix} = \begin{vmatrix} u \\ v \\ 1 \end{vmatrix} = \begin{vmatrix} \frac{x-u_0}{\alpha_u} \\ \frac{y-v_0}{\alpha_v} \\ 1 \end{vmatrix} \tag{2.17}$$

Avec u et v les coordonnées du point projeté sur le capteur, et x et y les coordonnées correspondantes en pixels sur l'image obtenue, u_0 et v_0 les coordonnées du centre du capteur, et α_u α_v, des facteurs d'agrandissement.

Dans le modèle unifié, on inverse l'approche qui vient d'être décrite : soit un pixel de coordonnées (x, y) sur l'image, les coordonnées des vecteurs associés u et v se déduisent avec :

$$u = \frac{x - u_0}{\alpha_u} \tag{2.18}$$

$$v = \frac{y - v_0}{\alpha_v} \tag{2.19}$$

Le point de l'image P_i dans le repère caméra sur le capteur est donc :

$${P_i}^{Rc} = \begin{vmatrix} u \\ v \\ 1 \end{vmatrix} \tag{2.20}$$

Ce point P_i est projeté sur la sphère au point P_s qui s'exprime donc :

$${P_s}^{Rc} = \begin{vmatrix} u\Delta \\ v\Delta \\ \Delta \end{vmatrix} \tag{2.21}$$

On repasse dans le repère R_s en retirant une valeur ξ sur l'axe z, ce qui donne donc :

$$P_s{}^{Rs} = \left| \begin{array}{c} u\Delta \\ v\Delta \\ \Delta - \xi \end{array} \right| \tag{2.22}$$

avec Δ correspondant au rapport $\frac{P_s\, O_c}{P_i\, O_c}$ calculé tel que :

$$\sqrt{(u\Delta)^2 + (v\Delta)^2 + (\Delta - \xi)^2} = 1 \tag{2.23}$$

soit

$$\Delta^2(u^2 + v^2 + 1) - 2\Delta\xi + \xi^2 - 1 = 0 \tag{2.24}$$

En posant $d = u^2 + v^2 + 1$, on peut exprimer deux solutions pour Δ :

$$\Delta = \frac{1}{d}\left(\xi + \sqrt{\xi^2 - d\xi^2 + d}\right) \tag{2.25}$$

$$\Delta = \frac{1}{d}\left(\xi - \sqrt{\xi^2 - d\xi^2 + d}\right) \tag{2.26}$$

$$\tag{2.27}$$

Les deux solutions correspondent aux deux intersections possibles avec la sphère. Dans notre cas, la solution qui nous intéresse est celle pour laquelle Δ est maximal. Sachant que $d \geq 0$:

$$\Delta = \frac{1}{d}\left(\xi + \sqrt{\xi^2 - d\xi^2 + d}\right) \tag{2.28}$$

On ne peut recalculer la position du point P selon X, Y, Z, ne connaissant pas la distance de ce point. En revanche, il est possible de ne calculer que X/Z et Y/Z.

$$\frac{X}{Z} = \frac{u}{1 - \frac{\xi}{\Delta}} \tag{2.29}$$

$$\frac{Y}{Z} = \frac{v}{1 - \frac{\xi}{\Delta}} \tag{2.30}$$

On peut également recalculer θ :

$$\theta = \arctan\left(\frac{\sqrt{u^2 + v^2}}{1 - \frac{\xi}{\Delta}}\right) \tag{2.31}$$

Calibration de ces modèles

Nous avons vu deux modélisations possibles pour les capteurs omnidirectionnels, la modélisation par un polynôme ou la modélisation unifiée plus récemment apparue. Chacune de ces méthodes a ses avantages :

- ⇒ La modélisation par polynôme peut s'adapter à n'importe quel type de capteur mais demande de calculer beaucoup de paramètres pour être précise.
- ⇒ La modélisation unifiée à l'avantage de ne calculer qu'un paramètre ξ au lieu des n coefficients du polynôme. Mais elle ne s'adapte pas à tous les capteurs [37].

Différentes méthodes utilisent ces modélisations pour calibrer les capteurs à miroir catadioptrique ou fisheye. Les plus courantes sont :

⇒ La toolbox de Scarramuzza [1] [33] utilisant un modèle polynomial

⇒ La toolbox de Mei [2] [19] avec un modèle unifié

L'approche de ces deux méthodes consiste à fournir des images d'une mire de type damier, prise sous différents angles de façon à couvrir tout le champ de vision du capteur à calibrer (voir figure 2.17). La détection des coins du damiers (figure 2.18) sur plusieurs images permet d'estimer les paramètres du modèle sachant que les point d'une même image sont sur le même plan et espacés d'une distance connue.

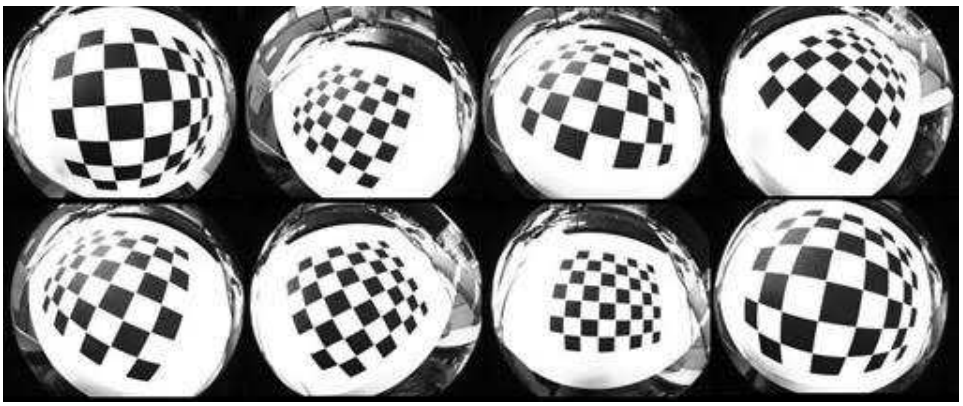


FIGURE 2.17 – *Mires pour la calibration*

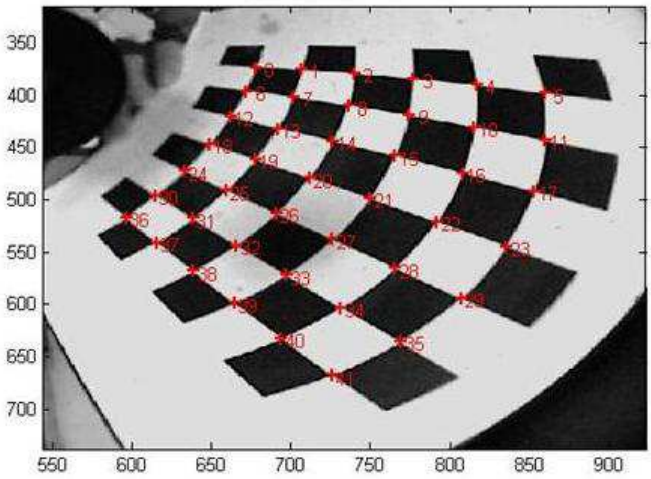


FIGURE 2.18 – *Détection des coins d'un damier*

2.2 STÉRÉOVISION

2.2.1 Principe général

Une caméra unique filmant une scène fournit une image 2D d'une scène du monde réel qui est en 3D. En d'autres termes, tous les points du monde

1. https ://sites.google.com/site/scarabotix/ocamcalib-toolbox
2. http ://www-sop.inria.fr/icare/personnel/Christopher.Mei/ChristopherMeiPhDStudentToolbox.html

réel 3D sont projetés sur le plan 2D du capteur. On ne perçoit pas l'information de profondeur des objets présents sur l'image. Le principe de la stéréovision permet d'obtenir cette profondeur en utilisant un principe dont nous nous servons tous. L'être humain, par exemple, perçoit la distance de ce qu'il voit grâce à ses deux yeux. Notre cerveau analyse les images perçues par chaque œil pour en déduire la distance. La stéréovision ne fait que reprendre ce principe en substituant aux yeux des caméras, et au cerveau un ordinateur.

Si l'on remplace les deux yeux de l'être humain par deux caméras, positionnées côte à côte, orientées vers la même direction et espacées d'une certaine distance, les images perçues au même instant ne seront pas les mêmes. Les points de vue étant différents, un objet proche des caméras sera perçu à une position très différente sur leurs images, alors qu'un objet très lointain, sera en revanche perçu de manière quasi-identique par les deux caméras.

La figure 2.19 le montre bien. Le cube vert est lointain, il y a peu de différence entre sa perception par la caméra gauche et sa perception par la caméra de droite. En revanche, le cube bleu est beaucoup plus proche des caméras. Il n'est donc pas du tout perçu au même endroit sur les images provenant de ces caméras. La différence d'emplacement, appelée disparité, est alors très importante.

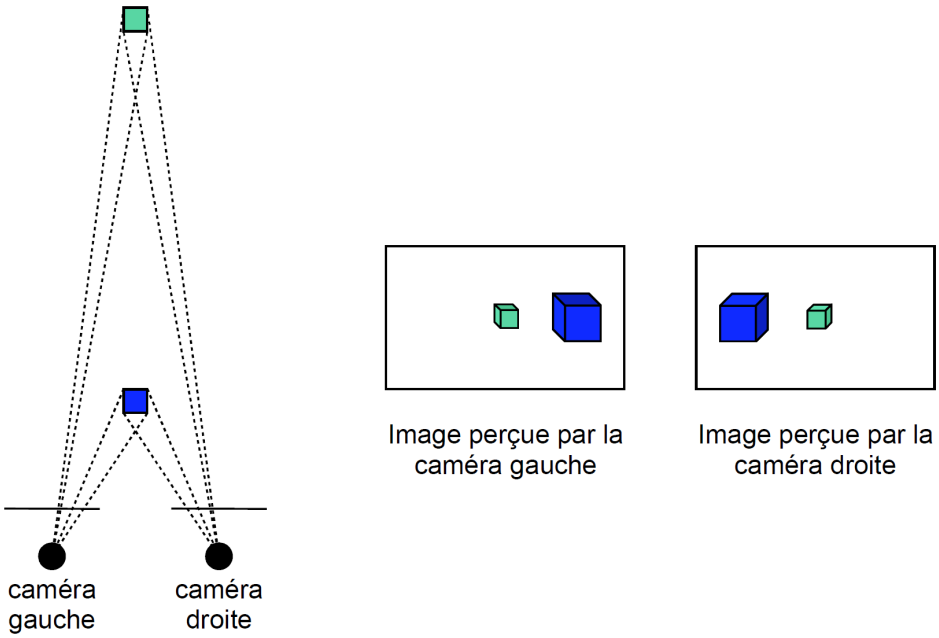


FIGURE 2.19 – *Principe de la stéréovision*

Dans le cas standard, deux caméras ayant des optiques classiques sont disposées côte à côte. Il est alors possible de modéliser le système obtenu comme le montre la figure 2.20. En considérant que les caméras sont identiques avec des capteurs de taille $S_w \times S_h$ générant des images de taille $W \times H$ avec une longueur focale f ainsi qu'un écartement (horizontal uniquement) entre les deux caméras noté E, la relation entre la disparité obtenue et la distance de l'objet observé peut être calculée :

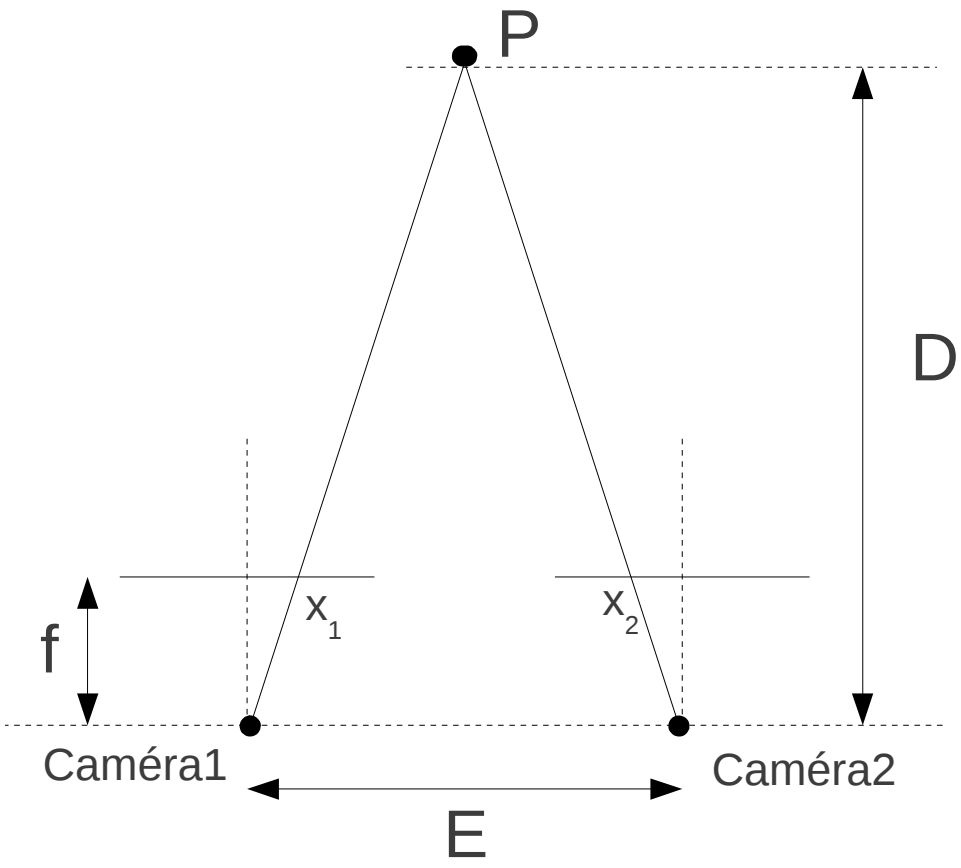


FIGURE 2.20 – *Modélisation d'un système stéréo standard*

$$E = \frac{x_1 D}{f} - \frac{x_2 D}{f}. \tag{2.32}$$

d'où

$$D = \frac{Ef}{x1 - x2} \tag{2.33}$$

x_1 et x_2 correspondent aux coordonnées du rayon sur le capteur de la caméra pour chacune des deux caméras. Pour convertir cette mesure en pixels u_1 et u_2, on applique la relation :

$$x_1 = \frac{u_1 Sw}{W} - \frac{Sw}{2} \tag{2.34}$$

d'où :

$$x_1 - x_2 = (u_1 - u_2)\frac{Sw}{W} = disp\frac{Sw}{W} \tag{2.35}$$

La relation entre la disparité $disp$ et la distance D a donc la forme :

$$D = \frac{A}{disp} = \frac{A}{u_1 - u_2} \tag{2.36}$$

Avec A une constante prenant en compte l'écartement entre les caméras, la focale, la taille des capteurs, ainsi que la résolution de l'image.

2.2.2 Format des images

Pour effectuer une recherche d'appariement par stéréovision, les images utilisées sont en niveau de gris. En effet, la couleur n'apporte pas beaucoup plus d'informations utilisables pour cette étape de calcul. Des travaux ont essayé de montrer l'apport d'un calcul avec une image couleur, mais le calcul se retrouve complexifié, pour un gain en termes de résultats qui n'est pas réellement significatif.

2.2.3 Géométrie épipolaire

On considère, comme illustré sur la figure 2.21, deux capteurs gauche et droit, ayant chacun pour foyer respectifs les points O_G et O_D. Le point P est projeté sur le capteur de la caméra gauche au point x_G, et au point x_D sur celui de la caméra droite. Tous les points appartenant au plan passant par les trois points O_G, O_D, et P se projettent respectivement pour chaque capteur sur les droite l_g et l_d. On appelle l_g la droite épipolaire du point x_D et l_d, la droite épipolaire du point x_G. Les points e_G et e_D sont appelés épipôles des capteurs gauche et droit. Dans notre problématique de mise en correspondance stéréoscopique, le but est donc d'apparier les pixels des droites épipolaires l_g et l_d.

2.2.4 Principe du traitement

Tout d'abord les deux images doivent être, si nécessaire, rectifiées de sorte que les lignes épipolaires deviennent des lignes horizontales sur les images.

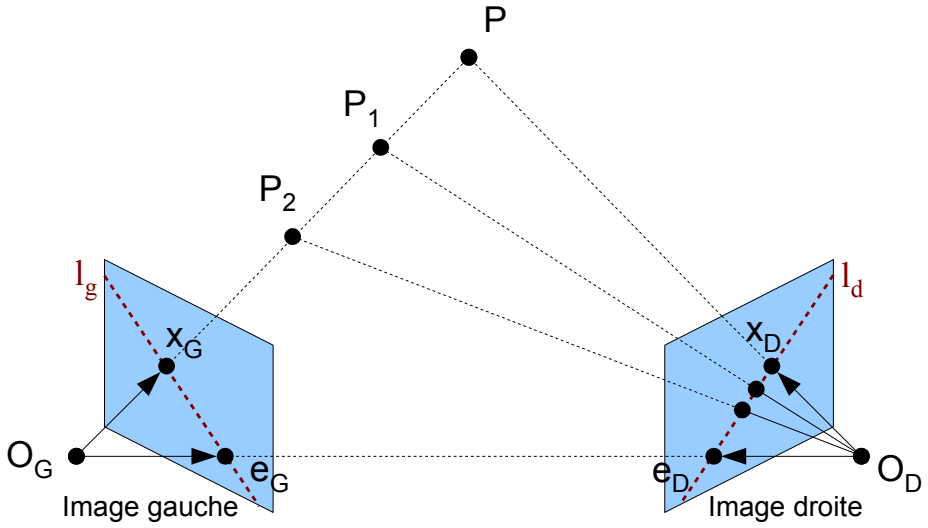


FIGURE 2.21 – *Géométrie épipolaire pour un système stéréoscopique*

Ensuite, dans le cas de calcul de stéréo dense, on recherche pour chaque pixel d'une image, le pixel correspondant, c'est-à-dire le pixel le plus ressemblant sur l'autre image. La recherche est effectuée sur une seule dimension, généralement selon une ligne horizontale. La rectification doit permettre de corriger les images pour pouvoir effectuer efficacement cette recherche 1D. On conçoit qu'une recherche sur un seul pixel ne donnera pas des résultats satisfaisants. Pour parer à ce problème, la recherche sera donc effectuée avec une fenêtre de plusieurs pixels centrée sur le pixel désiré. Le problème revient donc à rechercher la fenêtre de l'image de droite qui ressemble le plus à la fenêtre de l'image de gauche (voir figure 2.22).

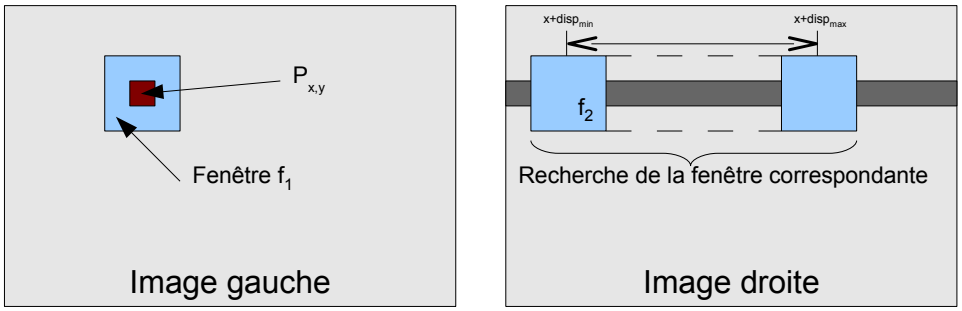


FIGURE 2.22 – *Recherche de disparité sur une dimension*

Pour chaque pixel $P_{x,y}$ de l'image référence :

1. une fenêtre f_1 centrée sur $P_{x,y}$ est extraite de l'image
2. pour une disparité $disp$ allant de $disp_{min}$ à $disp_{max}$, une fenêtre f_2 centrée sur $P_{x+disp,y}$ est extraite de la seconde image. Pour chaque disparité on calcule la distance $score_{disp}$ entre f_1 et f_2 soit $score_{disp} = Dist(f_1, f_2)$. Les bornes de l'intervalle de recherche : $disp_{min}$ et $disp_{max}$, sont définies en fonction de la configuration des capteurs et de la scène à observer.
3. La valeur attribuée au pixel de la carte de disparité $Disp_{x,y}$ corres-

pondant à $P_{x,y}$, correspond à la disparité $disp$ telle que $score_{disp}$ est le meilleur.

Pour estimer cette ressemblance, différentes méthodes mesurent la distance entre deux fenêtres $Dist(f_1, f_2)$. Nous en présenterons quelques-unes ci-après :

⇒ SAD *(Sum of Absolute Difference)*
⇒ SSD *(Sum of Squared Difference)*
⇒ ZSAD *(Zero mean Sum of Absolute Difference)*
⇒ ZSSD *(Zero mean Sum of Squared Difference)*
⇒ NCC *(Normalized Cross-Correlation)*
⇒ ZNCC *(Zero mean Normalized Cross-Correlation)*
⇒ CENSUS

2.2.4.1 Invariances

Soit deux fenêtre notées f_1 et f_2 de taille identique contenant N valeurs, provenant de deux images différentes. On note $Dist(f_1, f_2)$ le résultat du calcul de distance entre ces deux fenêtres par une méthode quelconque. En fonction de cette mesure, différentes invariances sont définies avec $a, b, c, d \in \Re^*$ (ensemble des réels privé de zéro) :

⇒ Invariance de gain :

$$Dist(af_1, bf_2) = Dist(f_1, f_2). \tag{2.37}$$

⇒ Invariance de biais :

$$Dist(f_1 + c, f_2 + d) = Dist(f_1, f_2). \tag{2.38}$$

⇒ Invariance de gain et biais :

$$Dist(af_1 + c, bf_2 + d) = Dist(f_1, f_2). \tag{2.39}$$

formules dans lesquelles on note $af+b$ la fenêtre dérivée de f dans laquelle les valeurs des pixels sont modifiées par multiplication par le réel a et par addition du réel b.

Ces invariances sont utiles pour ne pas prendre en compte des différences de gain ou de biais lors du calcul de distance. Ces phénomènes peuvent être causées par des différences de sensibilité entre les capteurs ou la luminosité perçue par chacun.

2.2.4.2 Méthode de mesure de distance entre deux fenêtres

Parmi les mesures de distance entre deux fenêtres, il existe deux principaux types :

⇒ les mesures de **similarité** utilisent la ressemblance entre deux fenêtres ; cette similarité est principalement utilisée sur les mesures utilisant la corrélation croisée comme la NCC et la ZNCC, ou la méthode CENSUS.
⇒ les mesures de **dissimilarité** se basent sur la différence de niveau de gris entre deux pixels pour calculer la distance, c'est le cas des mesures basées sur les P-normes comme la SAD, SSD, ZSAD, et ZSSD.

Notations préalables Dans les mesures de distance entre deux fenêtres, nous utilisons les notations suivantes :

⇒ Le nombre de pixels de la fenêtre f est noté N_f

⇒ Soit deux fenêtres f_1 et f_2 de taille identique N_f dont chaque élément (l'intensité d'un pixel en niveau de gris) d'index i variant de 1 à N_f est noté ${f_1}^i$ et ${f_2}^i$. Le produit scalaire entre ces deux fenêtres correspond à :

$$f_1 \cdot f_2 = \sum_{i=1}^{Nf} {f_1}^i \times {f_2}^i. \tag{2.40}$$

⇒ D'une façon générale la P-norme d'une fenêtre f est notée :

$$\|f\|_P = \left(\sum_{i=1}^{Nf} |f^i|^P \right)^{\frac{1}{P}} \tag{2.41}$$

Soit dans les cas que nous allons voir :

– La norme d'ordre 1

$$\|f\|_1 = \sum_{i=1}^{Nf} |f^i| \tag{2.42}$$

– La norme d'ordre 2

$$\|f\|_2 = \sqrt{\sum_{i=1}^{Nf} (f^i)^2} \tag{2.43}$$

⇒ La valeur moyenne d'une fenêtre f est notée $\overline{f}$

$$\overline{f} = \frac{1}{N_f} \sum_{i=1}^{Nf} f^i. \tag{2.44}$$

2.2.4.3 SAD *(Sum of Absolute Difference)*

Ici, la distance est la somme des valeurs provenant de la différence absolue entre deux matrices de taille identique représentant chacune une fenêtre. Le résultat obtenu est compris dans l'intervalle $[0; +\infty[$, avec un score idéal de 0 dans le cas de deux fenêtres identiques.

$$SAD(f_1, f_2) = \sum_{i=0}^{N_f - 1} |f_{1i} - f_{2i}| = \|f_{1i} - f_{2i}\|_1 \tag{2.45}$$

Cette mesure est largement utilisée [38] [39] [40] [41].

La figure 2.23 montre l'allure d'une carte de disparité calculée avec la mesure SAD, à partir d'images provenant d'une base de donnée [3] couramment utilisée pour évaluer les approches stéréo.

3. http ://vision.middlebury.edu/stereo/

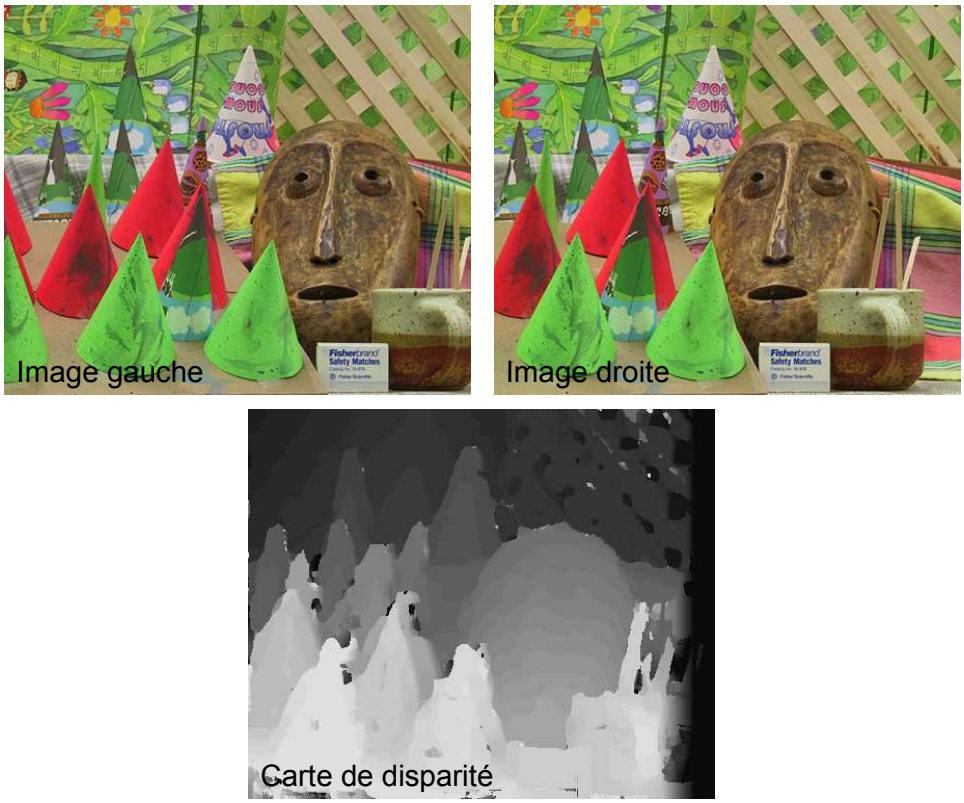


FIGURE 2.23 – *Exemple d'une carte de disparité obtenue avec la mesure SAD*

2.2.4.4 **SSD** *(Sum of Squared Difference)*

Ici, la distance est la somme des valeurs résultant de la différence au carré, terme à terme, entre deux matrices représentant les fenêtres. Comme pour la SAD, le résultat obtenu est compris dans l'intervalle $[0; +\infty[$, deux fenêtres identiques auront donc une distance de 0 entre elles.

$$SSD(f_1, f_2) = \sum_{i=0}^{N_f - 1} (f_{1i} - f_{2i})^2 \tag{2.46}$$

Cette mesure est également très utilisée [42] [43] [44] [45].

2.2.4.5 **ZSAD** *(Zero mean Sum of Absolute Difference)*

La méthode ZSAD reprend le principe de la SAD, mais centre chaque fenêtre par rapport à sa valeur moyenne. Le traitement ne différant de la SAD que par cette phase de centrage de la fenêtre, l'intervalle et le meilleur score restent le même, à savoir, un score compris dans $[0; +\infty[$, et un résultat parfait à 0.

$$ZSAD(f_1, f_2) = \left\| (f_{1i} - \overline{f_1}) - (f_{2i} - \overline{f_2}) \right\|_1 \tag{2.47}$$

Le centrage a pour avantage de ne pas prendre en compte des variations de luminosité entre deux images en donnant une invariance de type biais sur la mesure (équation 2.38). Ce cas est fréquent en environnement extérieur où la luminosité de la scène ne peut être contrôlée.

2.2.4.6 **ZSSD** *(Zero mean Sum of Squared Difference)*

Le principe est le même que pour la méthode SSD, en ajoutant également un centrage des fenêtres comme dans la méthode ZSAD et pour les mêmes raisons. La plage des score obtenue sera donc comme précédemment $[0; +\infty[$, avec 0 pour un résultat parfait.

$$ZSSD(f_1, f_2) = \sum_{i=0}^{N-1} ((f_{1_i} - \overline{f_1}) - (f_{2_i} - \overline{f_2}))^2 \tag{2.48}$$

De manière identique à la mesure ZSAD, celle-ci possède une invariance de type biais (équation 2.38).

2.2.4.7 **CC***(Cross-Correlation)*

La corrélation croisée est une mesure de similarité correspondant à :

$$CC(f_1, f_2) = f_1 \cdot f_2 \tag{2.49}$$

Elle n'est en réalité pas exploitable telle qu'elle, car l'intervalle des résultats est $[0; +\infty[$ avec un score idéal le plus élevé possible. Mais le score maximum à obtenir dans le cas de deux fenêtres identiques sera plus élevé si deux fenêtres ont des niveaux de gris élevés. Pour utiliser cette corrélation, des adaptations menant aux deux mesures listées ci-dessous sont nécessaires.

2.2.4.8 **NCC** *(Normalized Cross-Correlation)*

La distance calculée correspond à la corrélation croisée, mais normalisée, pour fixer l'intervalle des résultats à $[0; 1]$, avec un score idéal devant tendre vers 1.

$$NCC(f_1, f_2) = \frac{f_1 \cdot f_2}{\|f_1\|_2 \|f_2\|_2} \tag{2.50}$$

Cette mesure utilisée par exemple dans [46] [47] [48] possède une invariance de type gain (équation 2.37).

2.2.4.9 **ZNCC** *(Zero mean Normalized Cross-Correlation)*

Ici, la distance correspond à la mesure précédente à ceci près que les fenêtres sont centrées par rapport à leur valeur moyenne. Le résultat obtenu est compris dans l'intervalle $[-1; 1]$, avec un score idéal devant tendre vers 1.

$$ZNCC(f_1, f_2) = \frac{(f_{1_i} - \overline{f_1}) \cdot (f_{2_i} - \overline{f_2})}{\left\|f_{1_i} - \overline{f_1}\right\|_2 \left\|f_{2_i} - \overline{f_2}\right\|_2} \tag{2.51}$$

Cette mesure est très couramment utilisée [49] [50] [51]. Elle possède une invariance de type gain et biais (équation 2.39).

2.2.4.10 CENSUS

Cette approche, introduite par [52], est assez différente. En effet, on ne prend pas en compte ici la valeur du pixel mais une signature calculée au préalable. La signature R_τ de chaque pixel de l'image est calculée de la façon suivante :

$$R_\tau(f) = \otimes_{k \in [1;N_f]-\{N_f/2\}} \xi(f^{N_f/2}, f^k) \tag{2.52}$$

L'opérateur $\otimes$ est un opérateur de concaténation permettant d'effectuer plusieurs comparaisons pour le calcul de chaque bit de la signature. Chaque pixel d'une fenêtre f_k est comparé à la valeur du pixel central de la fenêtre $f^{N_f/2}$. La comparaison, notée $\xi(f^{N_f/2}, f^k)$ permet d'obtenir les bits de la signature, ceux-ci prenant comme valeur :

$$\xi(f^{N_f/2}, f^k) \begin{cases} 1 & \text{si} f_k > f^{N_f/2} \\ 0 & \text{sinon.} \end{cases} \tag{2.53}$$

Dans la pratique, on ne prend en compte que des fenêtres de taille 3×3 ou 5×5 autour du pixel désiré, pour effectuer cette transformation. L'image transformée est ainsi créée en remplaçant les valeurs des pixels par leur signature de taille 8 ou 24 bits, respectivement, en fonction de la taille du voisinage choisi (voir figure 2.24).

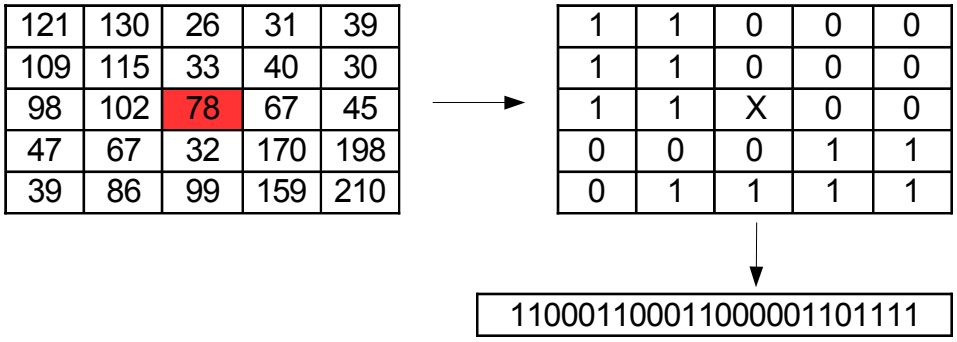


FIGURE 2.24 – *Transformation de CENSUS*

Le calcul se fait sur des signatures comprenant des résultats de comparaisons par rapport aux pixels voisins. L'avantage est donc d'être peu sensible aux variations de luminosité ambiante sur la scène, comme c'est le cas pour les méthodes centrées.

Puis le calcul de mise en correspondance se fait en prenant une fenêtre sur les images transformées. Le calcul de distance entre deux fenêtres comprenant les bits résultants des signatures est réalisé par une distance de Hamming.

$$CENSUS(f_1, f_2) = D_{Ham}(R_\tau(f_1), R_\tau(f_2)) \tag{2.54}$$

Cette distance de Hamming correspond à une mesure de distance entre deux mots binaires de taille identique. Le score obtenu correspond donc à la somme des bits du mot résultant de l'opération de ou-exclusif (XOR) entre ces deux mots à comparer, soit le nombre de valeur différentes entre ces deux valeurs à comparer. La figure 2.25 donne un exemple de calcul de cette distance.

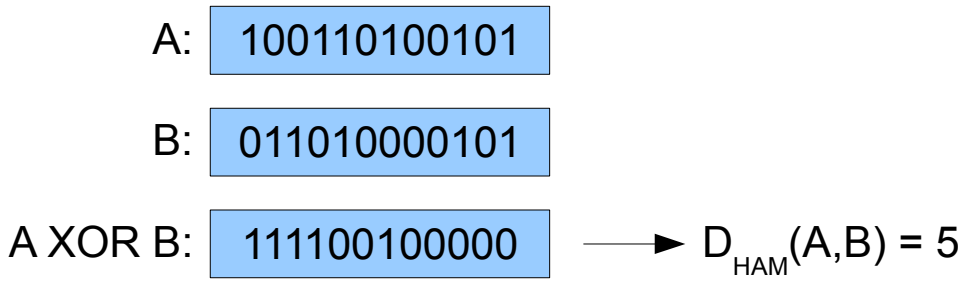


FIGURE 2.25 – *Distance de Hamming entre deux mots binaires*

La valeur de $CENSUS\,(f_1, f_2)$ appartient à l'intervalle $[0; N_f]$, le score idéal correspond à une distance de Hamming nulle. Cette méthode est très utilisée et elle possède l'avantage de présenter une invariance de type gain et biais (équation 2.39), et de plus, du fait du type de calcul assez simple utilisé (comparaison, distance de hamming) d'être parfaitement adaptée pour une implémentation sur un FPGA (Field-Programmable Gate Array), circuit qui contient des portes logiques dont les interconnexions sont programmables [53] [54] [55] [56] [57] [58] [59].

2.2.5 Autres approches

2.2.5.1 Approches multi-résolutions

Dans d'autres approches [60] [61] [62], une solution pour réduire les effets liés aux tailles de fenêtre et diminuer le temps de calcul consiste à décomposer les images en une pyramide d'images ayant différentes échelles (voir figure 2.26).

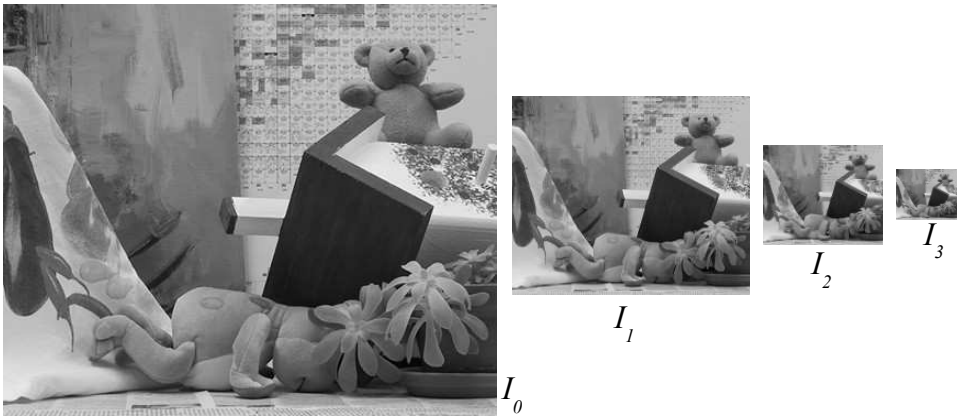


FIGURE 2.26 – *Création d'une pyramide d'images de différentes échelles*

Ainsi, le calcul de recherche d'appariement s'effectue au préalable sur l'image ayant la moins bonne résolution afin de cerner le résultat à obtenir, puis la recherche est effectuée dans les images successives de plus en plus précises afin d'affiner le résultat.

Différentes approches existent pour réaliser les pyramides, à savoir principalement :

⇒ Pyramides moyennes : La valeur du pixel calculée est la moyenne des pixels dans une zone de taille 2×2 [63].

⇒ Pyramides gaussiennes : On effectue un filtrage gaussienne sur l'image puis on ne prend qu'un pixel sur quatre afin de sous-échantillonner l'image.

Une propagation des résultats est ensuite effectuée en déduisant à partir du résultat obtenu pour une image $n+1$ la zone de recherche pour l'image n. Lors de cette étape, les erreurs peuvent se propager. En effet, si un détail n'a pas été perçu sur une couche élevée, il se peut que la zone de recherche sur les couches inférieures ne corresponde pas.

2.2.5.2 Contraintes

Plusieurs contraintes peuvent être testées afin de vérifier la cohérence des résultats obtenus lors de la recherche de mise en correspondance. Pour exprimer ces différentes contraintes, on considère, ${p_i}^k$ un pixel de l'image d'indice i à la position k, et $C({p_i}^k)$ le pixel correspondant trouvé sur l'autre image.

Unicité

La contrainte d'unicité permet de vérifier que deux pixels différents sur une image n'ont pas le même correspondant dans l'autre (voir figure 2.27).

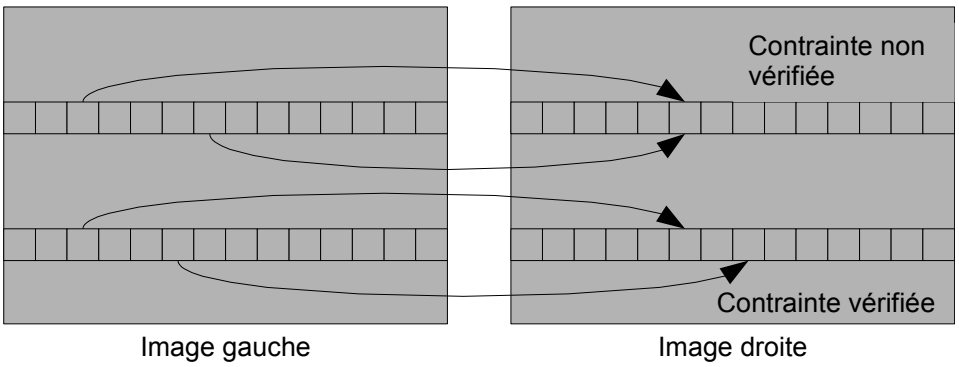


FIGURE 2.27 – *Contrainte d'unicité*

Cette contrainte utilisée dans [64] [65] [66] [40] se vérifie donc par :

$$\text{Si } C(p_1^{k1}) = p_2^{k2} \text{ alors } C(p_1^{k3}) \neq p_2^{k2} \text{ avec } k1 \neq k3. \tag{2.55}$$

A noter que si le plan est incliné par rapport à une caméra, cette contrainte peut ne pas être vérifiée alors que le résultat est correct comme c'est le cas dans la figure 2.28.

Ordre

Cette contrainte consiste à vérifier que l'ordre des pixels d'une image est le même que celui des pixels correspondants sur l'autre image (voir figure 2.29).

Cette contrainte, utilisée par exemple dans [67] [68] [69], se vérifie par :

$$\text{Si } C(p_1^{k1}) = p_2^{k2} \text{ et } C(p_1^{k3}) = p_2^{k4} \text{ alors } (k1-k2)(k3-k4) \geq 0. \tag{2.56}$$

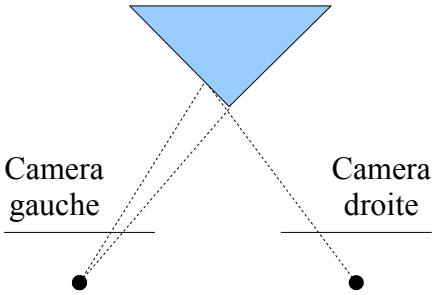


FIGURE 2.28 – *Appariement correct ne respectant pas la contrainte d'unicité*

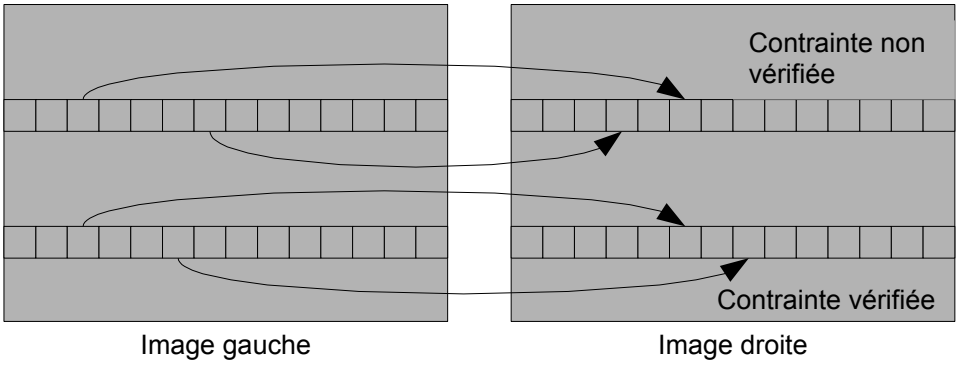


FIGURE 2.29 – *Contrainte d'ordre*

A noter que si un objet se situe devant un autre avec un espacement suffisant entre eux, cette contrainte peut ne pas être vérifiée malgré un appariement correct comme le montre la figure 2.30.

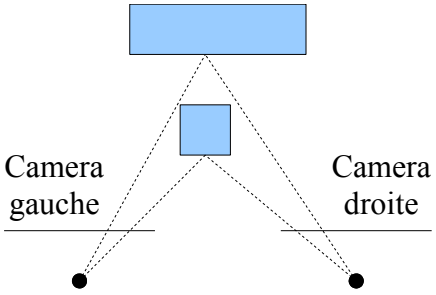


FIGURE 2.30 – *Appariement correct ne respectant pas la contrainte d'ordre*

Symétrie

Comme expliqué précédemment, du fait de la disposition des caméras par rapport à la scène à filmer, on ne peut empêcher les occultations. Celles-ci fourniront donc des résultats erronés, l'appariement ne pouvant s'effectuer correctement. Pour pallier à ce problème, une solution, illustrée dans la figure 2.31, est de détecter les appariements erronés par la contrainte de symétrie. Cette contrainte est vérifiée si :

$$C(p_1^{k1}) = p_2^{k2} \text{ et } C(p_2^{k2}) = p_1^{k1} \tag{2.57}$$

Une contrainte variante de celle-ci appelée contrainte de symétrie faible

existe [70]. Elle diffère en y ajoutant un seuil de tolérance T sur la symétrie. Elle se vérifie si :

$$C(p_1^{k1}) = p_2^{k2} \text{ et } C(p_2^{k2}) = p_1^{k3} \text{ avec } |k3 - k1| < T \tag{2.58}$$

La contrainte de symétrie est l'une des plus employées [71] [72] [73] [51], elle a l'avantage de détecter les zones erronées du fait des occultations. Elle peut également détecter des zones erronées sur les surfaces lisses. En revanche, elle a l'inconvénient que la seconde recherche double le temps de calcul. La figure 2.32 illustre l'influence de cette contrainte en signalant en rouge les zones ne la respectant pas. Celles-ci correspondent majoritairement aux parties masquées de la scène.

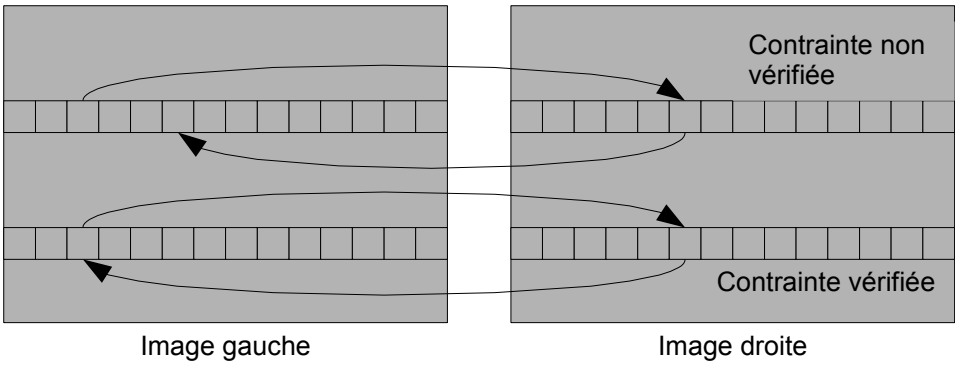


FIGURE 2.31 – *Contrainte de symétrie*

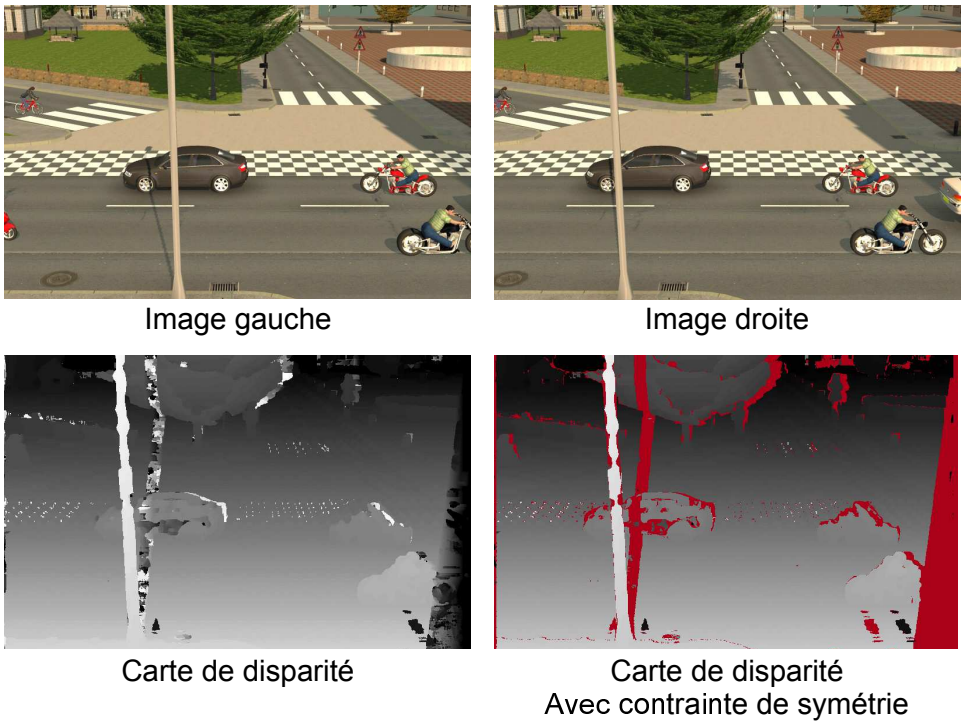


FIGURE 2.32 – *Exemple de contrainte de symétrie*

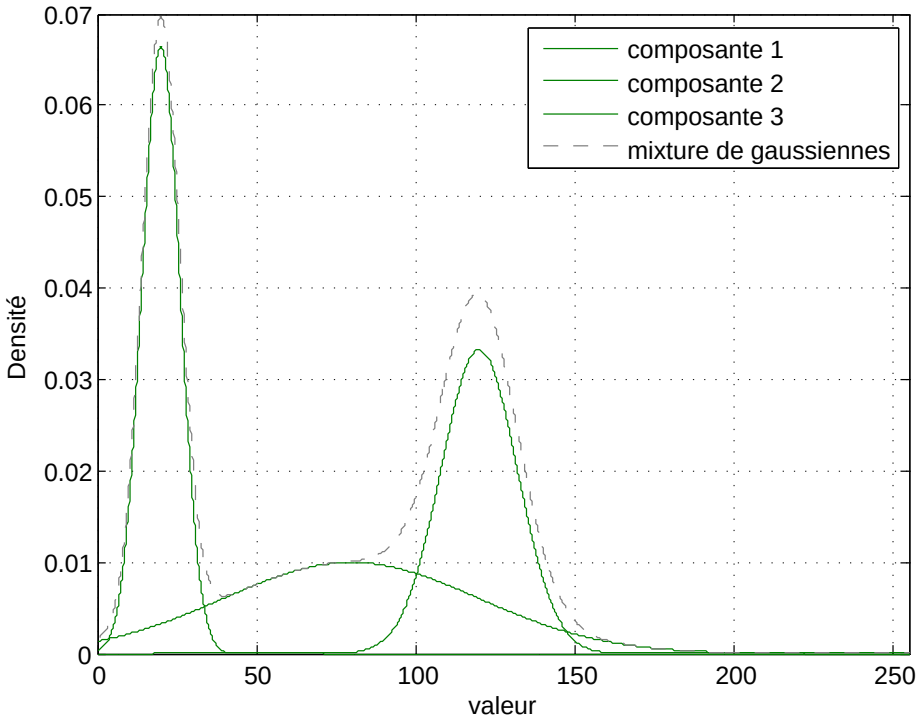


FIGURE 2.33 – *modèle à base de mixture de gaussienne*

2.3 EXTRACTION FOND-FORME

De nombreuses applications vidéos cherchent à distinguer les objets mobiles d'une séquence d'images enregistrée à l'aide de caméras fixes installées sur le bord de la route. Cette segmentation fond/forme peut être effectuée avec de simples méthodes calculant la différence entre deux images successives, ou encore en calculant une image moyenne du fond. Ces algorithmes ont des performances très limitées en environnement extérieur (variation de la luminosité ambiante, ombre des objets, ...).

Depuis quelques années, les méthodes les plus populaires sont les méthodes dites "probabilistes" où l'on modélise et met à jour le modèle du fond en fonction de probabilité de récurrence des valeurs des pixels. Trois de ces méthodes sont détaillées ci-après.

2.3.1 Mixture de gaussiennes

Dans cette approche, chaque pixel est modélisé par une mixture de N gaussiennes, $2 \leq N \leq 5$ [74] (voir figure 2.33).

Pour $n = 1, \ldots, N$, chaque élément de la mixture de gaussiennes est représenté par :

- ⇒ une moyenne $\boldsymbol{\mu_n}$
- ⇒ une matrice de covariance Σ_n avec Σ_n=${\sigma_n}^2 I$, σ_n étant l'écart type de la gaussienne.
- ⇒ un poids α_n ($\sum_{n=1}^{N} \alpha_n = 1$)

On peut remarquer que σ_n est réduit à un scalaire, comme discuté dans [74].

À chaque nouvelle image traitée, la mixture de gaussiennes (pour tous les pixels) est mise à jour pour expliquer correctement les couleurs affichées par chaque pixel. Pour faire ceci, à un instant t, on considère que le modèle $\mathbf{M_t}$ généré pour chaque pixel à partir des mesures $\{\mathbf{Z_0}, \mathbf{Z_1}, ..., \mathbf{Z_{t-1}}\}$ est correct. La vraisemblance pour qu'un pixel appartienne au fond est :

$$P\left(\mathbf{Z_t}|\mathbf{M_t}\right) = \sum_{n=1}^{n=N} \alpha_n \mathcal{N}\left(\boldsymbol{\mu_n}, \boldsymbol{\Sigma_n}\right) \tag{2.59}$$

$$\mathcal{N}\left(\boldsymbol{\mu_n}, \boldsymbol{\Sigma_n}\right) = \frac{1}{(2\pi)^{d/2}\left|\boldsymbol{\Sigma_n}\right|^{1/2}} e^{-\frac{1}{2}(\mathbf{Z_t}-\boldsymbol{\mu_n})^T \boldsymbol{\Sigma}^{-1}(Z_t-\boldsymbol{\mu_n})} \tag{2.60}$$

où d est la dimension de l'espace de couleurs de la mesure $\mathbf{Z_t}$.

Pour mettre à jour le modèle, on associe d'abord la mesure $\mathbf{Z_t}$ à une gaussienne n' si

$$||\mathbf{Z_t} - \boldsymbol{\mu_n}|| < K\sigma_n \tag{2.61}$$

où K vaut 2 ou 3. L'opérateur $<$ est vrai si toutes les composantes du vecteur à gauche sont inférieures à $K\sigma_n$.

Cette mesure représente le fond si la gaussienne n' explique le fond de la scène. En fait, le poids $\alpha_{n'}$ est élevé. Cette gaussienne est alors mise à jour :

$$\alpha'_n \leftarrow (1-\delta)\alpha'_n + \delta \tag{2.62}$$

$$\boldsymbol{\mu'_n} \leftarrow (1-\delta)\boldsymbol{\mu'_n} + \delta\mathbf{Z_t} \tag{2.63}$$

$$\sigma^{2'}_n \leftarrow (1-\delta)\sigma^{2'}_n + \delta(\mathbf{Z_t} - \boldsymbol{\mu'_n})^T(\mathbf{Z_t} - \boldsymbol{\mu'_n}) \tag{2.64}$$

avec δ le coefficient d'apprentissage. Il représente la vitesse d'adaptation du modèle. Pour toutes les autres gaussiennes $n \neq n'$, la moyenne et la variance ne sont pas modifiées, mais :

$$\alpha_n \leftarrow (1-\delta)\alpha_n \tag{2.65}$$

Si le test 2.61 échoue, le pixel est associé à la forme. La gaussienne ayant le plus petit poids est réinitialisée avec la mesure actuelle :

$$\alpha_n = \delta \tag{2.66}$$

$$\boldsymbol{\mu_n} = \mathbf{Z_t} \tag{2.67}$$

$$\sigma_n^2 = \bar{\sigma}^2 \tag{2.68}$$

avec $\bar{\sigma}^2$ une variance élevée. Ces affectations sont aussi appliquées pour l'initialisation de la mixture.

2.3.2 Codebook 2 layers

Cette méthode [75] est très largement inspirée de celle du codebook [76]. Elle en diffère en utilisant deux codebooks, bibliothèques de données pour chaque pixel contenant des informations pour modéliser le fond. Ceci a été réalisé de manière à pouvoir retenir des valeurs de pixels

qui ont appartenu au fond, et qui pourraient redevenir du fond. C'est typiquement le cas avec les mouvements de branches d'arbre.

Chaque codebook contient des éléments appelés codeword (CW) pour modéliser le fond de l'image. Chacun des CW contient ces informations :

⇒ v_i : valeur moyenne du pixel (R,V,B)
⇒ I_{max} : limite maximale d'intensité du CW
⇒ I_{min} : limite minimale d'intensité du CW
⇒ f : fréquence du CW (nombre d'occurrences)
⇒ λ : nombre maximal d'images où le CW ne correspond à aucun pixel
⇒ p : première occurrence du CW
⇒ q : dernière occurrence du CW

Le principe est le même qu'avec le codebook simple, mais avec deux codebooks par pixel : un principal appelé M, et un secondaire appelé H.

Le traitement se fait en 2 phases :

1. une **phase d'apprentissage** qui sert à créer les codebooks principaux initiaux
2. une **phase de soustraction** pour extraire le fond de la forme.

2.3.2.1 Phase d'apprentissage

Cette phase sert à initialiser le modèle de fond à partir des premières images. Elle ne fait donc pas la distinction entre le fond et la forme. Son fonctionnement est le suivant :

L'intensité I_t de chaque pixel $x_t = (R, V, B)$, est calculée par

$$I_t = \sqrt{R^2 + V^2 + B^2} \tag{2.69}$$

La distorsion de couleur δ entre ce pixel $x_t = (R, V, B)$ et un codeword c_i caractérisé par $v_i = (\bar{R}_i, \bar{V}_i, \bar{B}_i)$ peut être calculé par :

$$\langle x_t, v_i \rangle^2 = (\bar{R}_i R + \bar{V}_i V + \bar{B}_i B)^2 \tag{2.70}$$

$$\|v_i\|^2 = \bar{R}_i^2 + \bar{V}_i^2 + \bar{B}_i^2 \tag{2.71}$$

$$\|x_t\|^2 = R^2 + V^2 + B^2 \tag{2.72}$$

$$colordist(x_t, v_i) = \delta = \sqrt{\|x_t\|^2 - \frac{\langle x_t, v_i \rangle^2}{\|v_i\|^2}} \tag{2.73}$$

Un pixel x_t avec une intensité I_t correspond à un codeword c_i si I_t est dans l'intervalle $[I_{min}, I_{max}]$ et si la distorsion de couleur δ respecte $\delta < \epsilon$

En phase d'apprentissage, seule M est construit, H reste vide. Pour un nouveau pixel x_t, on cherche un CW dans M correspondant à x_t. Si on en trouve un, il est mis à jour avec x_t, sinon un nouveau CW est créé à partir de la valeur de x_t de la façon suivante :

$$v_i \leftarrow (R, V, B) \tag{2.74}$$

$$I_{min} \leftarrow max\{0, I_t - \alpha\} \tag{2.75}$$

$$I_{max} \leftarrow min\{255, I_t + \alpha\} \tag{2.76}$$

$$f \leftarrow 1;\ \lambda \leftarrow t - 1;\ p \leftarrow t;\ q \leftarrow t \tag{2.77}$$

avec α une valeur représentant une tolérance d'intensité.

Pendant la phase d'apprentissage, un codeword est mis à jour par un pixel x_t comme ceci :

$$\bar{R} \leftarrow \frac{\bar{R} \times f + R}{f+1} \text{ (de même pour V et B)} \tag{2.78}$$

$$I_{min} \leftarrow \frac{I - \alpha + f \times I_{min}}{f+1} \tag{2.79}$$

$$I_{max} \leftarrow \frac{I + \alpha + f \times I_{max}}{f+1} \tag{2.80}$$

$$f \leftarrow f+1;\ \lambda \leftarrow max\{\lambda, t-q\};\ p \leftarrow p;\ q \leftarrow t \tag{2.81}$$

2.3.2.2 Phase de soustraction

En phase de soustraction, pour un nouveau pixel x_t, on cherche un CW dans M correspondant à x_t. Si on en trouve un, il est mis à jour avec x_t et le pixel est considéré comme appartenant au fond. Sinon, on cherche un CW correspondant à ce pixel dans H. Si on un trouve un, on le met à jour avec x_t, sinon, on en crée un nouveau dans H avec la valeur de x_t.

Un CW est mis à jour comme précédemment dans la phase d'apprentissage à l'exception de I_{min} et I_{max} qui sont mis à jour de la façon suivante, avec β un coefficient pour changer la vitesse d'adaptation :

$$I_{min} \leftarrow (1-\beta)\,(I_t - \alpha) + \beta \cdot I_{min} \tag{2.82}$$

$$I_{max} \leftarrow (1-\beta)\,(I_t + \alpha) + \beta \cdot I_{max} \tag{2.83}$$

Ensuite, les modèles M et H sont affinés avec ces règles :

⇒ Supprimer les CWs de H ayant $\lambda > T_H$

⇒ Déplacer les CWs restant plus de T_{add} dans H vers M

⇒ Supprimer les CWs de M n'apparaissant pas plus longtemps que T_{delete}

Cette méthode garde beaucoup de données dans la couche H qui ne sont pas importantes au moment où elles arrivent mais qui pourraient l'être plus tard. Ceci est donc très utile pour permettre par exemple une bonne détection des mouvements de branches d'arbre. En revanche, ajouter des données pendant le traitement crée une occupation mémoire variable, ce qui peut se répercuter sur la vitesse d'exécution pour trouver le codeword correspondant dans une très grande quantité de codewords.

2.3.3 VuMètre

Le VuMètre développé par Goyat *et al.* [77] est un modèle non-paramétrique basé sur une estimation discrète de la probabilité de distribution. Il s'agit d'une approche probabiliste pour définir le modèle du fond. Pour cela, il utilise trois paramètres :

⇒ un **coefficient d'apprentissage** α, permettant de modifier la vitesse d'adaptation du modèle.

⇒ un **seuil fond/forme** appelé T, permettant de sélectionner à partir de quel niveau on considère le pixel comme fond ou forme

⇒ un **nombre de classes** N, qui permet de définir sur combien de valeurs sera discrétisée la probabilité de distribution

Soit I_t une image à l'instant t. Le vecteur $y_t(u)$ donne la valeur RVB du pixel u. Un pixel peut prendre deux états, (ω_1) s'il appartient au fond, (ω_2) s'il appartient à la forme. Cette méthode essaye d'estimer $p(\omega_1 \mid y_t(u))$. Avec 3 composantes de couleur i (Rouge, Vert, Bleu), la fonction de densité de probabilité peut être approximée par :

$$p(\omega_1 \mid y_t(u)) = \prod_{i=1}^{3} p(\omega_1 \mid y_t^i(u)) \tag{2.84}$$

avec

$$p(\omega_1 \mid y_t^i(u)) \approx K_i \sum_{j=1}^{N} \pi_t^{ij} \delta(b_t^i(u) - j) \tag{2.85}$$

où δ est le symbole de Kronecker, $b_t(u)$ donne le vecteur d'index de la classe associée à $y_t(u)$, j est un index de classe, et K_i est une constante de normalisation permettant de garder à chaque instant :

$$\sum_{j=1}^{N} \pi_t^{ij} = 1 \tag{2.86}$$

π_t^{ij} est une fonction de masse discrète représentée par une classe.

A la première image $(t = 0)$, les valeurs des classes sont mises à $\pi_0^{ij} = 1/N$ pour garder une somme égale à 1 comme dans l'équation 2.86. A chaque nouveau pixel, sa valeur correspond à une classe π_t^{ij} et son niveau est mis à jour selon :

$$\pi_{t+1}^{ij} = \pi_t^{ij} + \alpha \cdot \delta(b_{t+1}^i(u) - j) \tag{2.87}$$

Après un certain nombre d'images, les classes modélisant le fond ont une valeur plus élevée. Pour pouvoir décider à quel moment un pixel appartient au fond ou non, un seuil T (voir figure 2.34) est défini. Chaque nouveau pixel ayant une classe au dessus de ce seuil sera détecté comme appartenant au fond.

En mode RVB, chaque pixel est modélisé par 3 VuMètres (un par composante couleur). Pour considérer un pixel comme appartenant au fond, il doit être calculé comme appartenant au fond pour chaque VuMètre.

Pour obtenir un bon apprentissage et une bonne adaptation de l'algorithme, il est nécessaire de bien choisir les paramètres (taux d'apprentissage α et seuil T). Ces valeurs peuvent être changées en fonction de la luminosité ou de la vitesse des véhicules suivis. Par défaut, la valeur du taux d'apprentissage est 0.01 et celle du seuil est de 0.2.

2.4 Conclusion

Nous avons pu avoir dans ce chapitre un aperçu des technologies et méthodes existantes aussi bien pour les capteurs vidéo et leurs optiques, que leur modélisation respective permettant de les calibrer. Puis, nous avons rédigé un état de l'art sur les méthodes existantes pour un traitement en stéréovision dense dans le cas classique de deux caméras côte à côte. Enfin, nous avons étudié les méthodes d'extraction fond/forme en montrant en

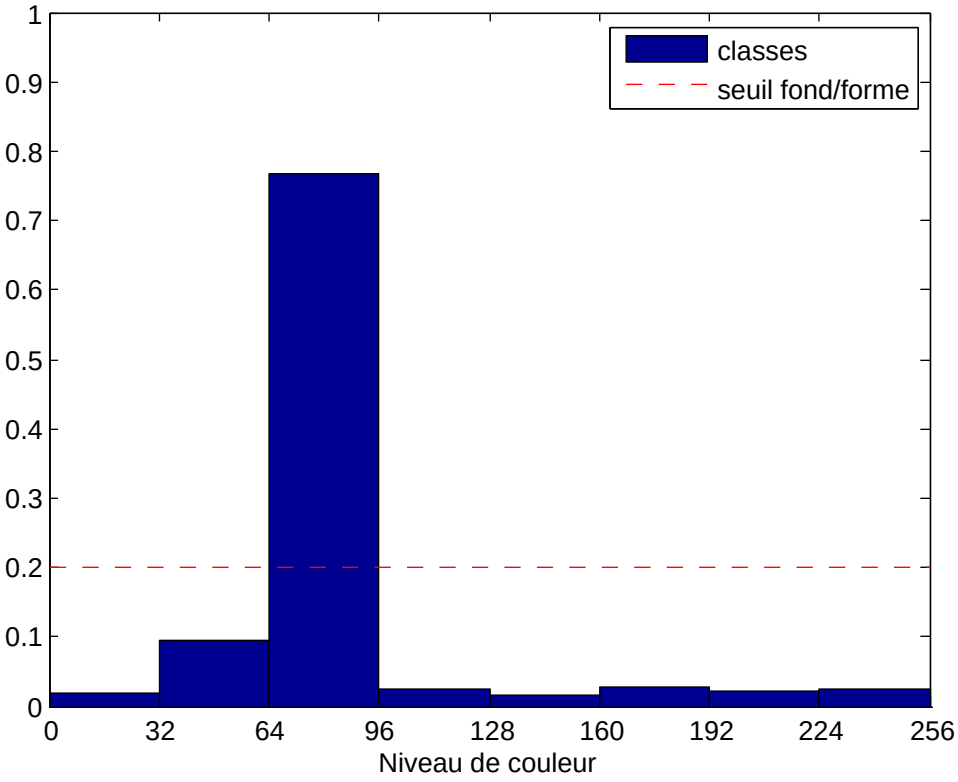


FIGURE 2.34 – *Vue des niveaux de classes et du seuil du vumètre*

particulier l'intérêt d'utiliser une approche dite probabiliste pour l'observation de scènes routières. Les chapitres suivants vont donc, à partir de cet inventaire, permettre de décrire la démarche envisagée pour bâtir un observatoire répondant aux besoins, de voir l'application des méthodes à une scène routière en proposant un banc de test, et d'étudier le choix du capteur vidéo, de la disposition des caméras et du traitement stéréo associé.

Application des méthodes d'extraction fond / forme 3

Sommaire

Dans ce chapitre, nous partons de l'état de l'art sur l'extraction fond forme réalisé dans la partie 2.3, pour approfondir l'application de ces méthodes dans notre cas concret de surveillance de scène routière. Après une présentation des caractéristiques de ce type de scène, et des facteurs pouvant perturber la détection des formes, nous présentons un modèle de comparaison ainsi que toute une base de séquences mise en place à partir de données simulées pour pouvoir évaluer les méthodes d'extraction quantitativement, et ainsi, en déduire celle qui est la plus adaptée à notre cas de scène routière.

3.1 Problématique sur scène routière

Dans une scène routière, l'acquisition se fait à partir d'une caméra fixe et il existe de nombreuses causes pouvant altérer la qualité de l'extraction.

Bruit sur les images

Ce bruit est inhérent à tout capteur vidéo et est accentué par une faible luminosité et une faible sensibilité du capteur vidéo. Il se caractérise par des valeurs de pixel changeant légèrement. Étant donné que les méthodes d'extraction se basent sur la couleur du pixel, elles doivent donc avoir une certaine tolérance pour ne pas être trop sensible au bruit (voir figure 3.1). Un autre phénomène de bruit peut aussi être lié à la compression des images, il est donc important d'utiliser un encodage ayant le moins possible de pertes.

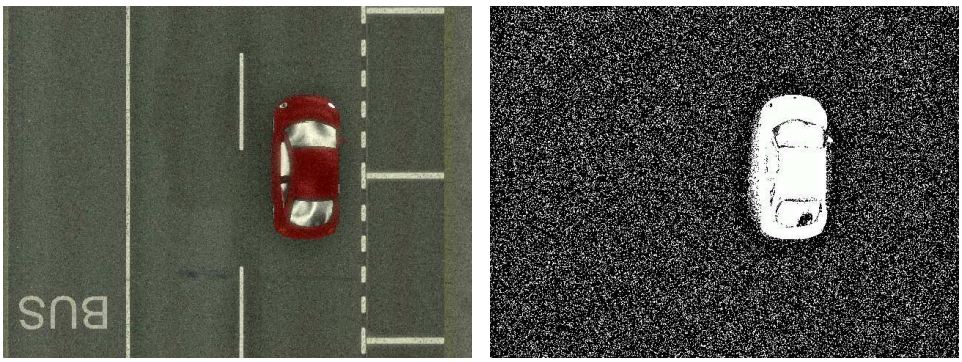


FIGURE 3.1 – *Influence du bruit fort sur l'extraction*

Mouvement d'objets appartenant au fond

Ce phénomène est relativement présent. On pense, par exemple, aux mouvements des branches d'arbre liés au vent. Le but est de détecter ces objets comme appartenant au fond de l'image. Ce mouvement est, de façon générale, assez répétitif. La méthode d'extraction doit donc être capable d'apprendre les valeurs des pixels les plus récurrents sur un laps de temps à définir, et de considérer que ces pixels correspondent au fond. C'est pourquoi les méthodes dites probabilistes semblent plus adaptées ici.

Changement de luminosité

Ce phénomène est assez courant quand on observe des véhicules sur la route, et peut être lié à un passage de nuage. Les couleurs sur l'image varient donc et l'algorithme peut détecter qu'il s'agit d'un objet. L'idéal serait d'être assez peu sensible aux changements de luminosité, ou d'être capable de s'adapter rapidement pour limiter les mauvaises détections.

Scènes sombres - faible contraste

En cas de faible luminosité, les images captées ont des niveaux de couleur assez faibles, les différences entre les couleurs sont moins marquées. Il

peut donc être plus difficile de détecter le fond de la forme dans ce type de scène.

Brouillard

Tout comme avec les scènes sombres, le brouillard réduit considérablement le contraste sur l'image. Plus la distance des objets croît, plus le phénomène s'accentue, la couche de brouillard à traverser par les rayons lumineux étant plus épaisse.

Différences de couleurs

Les véhicules ayant une couleur proche du fond peuvent être plus difficilement détectés étant donné que les algorithmes se basent sur la couleur du pixel pour identifier le fond de la forme.

Ombres

Les ombres sont des zones qui suivent les véhicules, avec des changements de valeur de pixel par rapport au fond. Il est donc assez logique que les algorithmes puissent classer par erreur ces zones comme forme. Ce point peut-être très gênant car, en se basant sur l'extraction fond/forme seule, un petit véhicule peut, du fait de son ombre importante, être vu comme un gros véhicule. La différenciation de deux véhicules proches peut également être assez délicate comme le montre la figure 3.2 qui illustre bien les problèmes liés aux deux-roues.

Changement du fond : Stationnement

Dans une rue, des véhicules stationnés qui sont donc considérés comme du fond peuvent sortir de leur place et doivent donc être considérés comme de la forme. Le raisonnement inverse se produit pour un véhicule qui vient se stationner sur une place. La méthode d'extraction fond/forme doit donc être capable de mettre à jour son modèle de fond pour prendre en compte ces modifications.

FIGURE 3.2 – *Problème de detection lié à l'ombre*

FIGURE 3.3 – *vidéo IPPR*

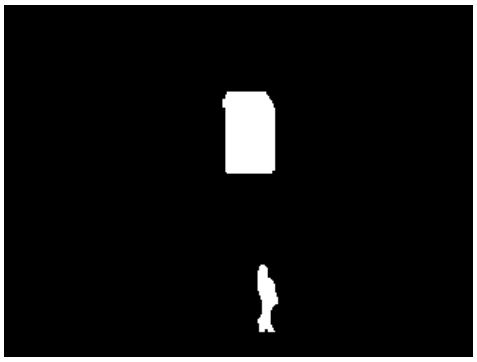


FIGURE 3.4 – *vérité terrain*

Densité de circulation

Les méthodes de détection doivent apprendre le fond, et pour cela, elles considèrent que les valeurs des pixels les plus fréquentes sont des valeurs de fond. Ceci peut poser problème dans des situations urbaines à circulation très dense où le fond n'est pas souvent visible en raison du flux dense et continu de véhicules. Par exemple, en cas de congestion de trafic, les véhicules et les 2RM s'arrêtent et risquent d'être détectés comme du fond.

3.2 SÉQUENCES DE TEST

Pour pouvoir tester l'efficacité des algorithmes d'extraction fond/forme vis à vis des causes d'altération, il faut appliquer le traitement sur des vidéos pour lesquelles la vérité terrain est connue. Ces vidéos avec vérité terrain sont particulièrement rares. En effet, elles sont réalisées de façon manuelle en cliquant sur chaque pixel représentant une forme dans chaque image d'une vidéo, travail très fastidieux. De plus, les rares vidéos trouvées ne traitent pas forcément les scènes routières, et ne mettent pas forcément en évidence tous les aspects présentés précédemment.

Afin de résoudre ce problème, une autre approche envisagée consiste à utiliser un simulateur générant des scènes vidéo en 3D, et à partir desquelles on peut obtenir la vérité terrain.

3.2.1 Vidéos réelles

La seule vidéo réelle trouvée (figure 3.3) et montrant une scène routière avec sa vérité terrain associée (figure 3.4), est la séquence appelée "data3" de l'IPPR contest 2006 [1]. Cette séquence d'une minute à 5 images/sec et d'une taille de 320×240 pixels montre une circulation dans une rue avec un bus et des piétons. Malheureusement, une vidéo de quelques dizaines de secondes, avec des images de taille très réduite, n'est pas suffisante pour analyser efficacement les algorithmes.

Bien évidemment, une génération manuelle de la vérité terrain introduit des erreurs et, aussi précise qu'elle soit, ne peut garantir une vérité terrain parfaite. Par conséquent, elle introduit un léger bruit dans les comparaisons. D'où l'intérêt d'utiliser une approche différente pour générer les séquences avec vérité terrain.

1. http ://media.ee.ntu.edu.tw/Archer_contest

3.2.2 Vidéos simulées

Des vidéos de synthèse ont été réalisées à l'aide du simulateur Prosivic [78], qui présente deux avantages :

⇒ Possibilité de tester séparément les facteurs perturbants pour une même scène afin d'isoler les influences de chacun sur le résultat du traitement.

⇒ Les vidéos sont générées avec la vérité terrain associée, ce qui permet de pouvoir comparer les résultats.

Bien évidemment, les vidéos obtenues ne sont pas réelles, et le simulateur utilisé doit être suffisamment réaliste et de bonne qualité pour pouvoir représenter le plus fidèlement possible les différents facteurs gênants. Le simulateur utilisé permet de modéliser tous les facteurs décrits en 3.1 pour générer les séquences visibles sur la figure 3.5 :

⇒ **vidéo 2** : carrefour, véhicules sur différentes voies, 3 voitures et piétons, bruit assez fort, changement de luminosité faible (la vidéo 1 correspond à la vidéo réelle décrite précédemment).

⇒ **vidéo 3** : rue, scène assez sombre, 1 bus, véhicules variés et des piétons, changement brusque de luminosité à mi-séquence.

⇒ **vidéo 4** : rond-point, un seul véhicule qui ne génère pas d'ombre, un arbre au milieu du rond-point avec ombre, l'arbre bouge, mouvement du soleil.

⇒ **vidéo 5** : circulation dense, voitures de tailles et couleurs différentes et 1 bus, ombres venant des véhicules et des bâtiments, masquages entre véhicules.

⇒ **vidéo 6** : identique à la vidéo 5 mais filmée sous un angle différent : de l'autre côté de la route.

⇒ **vidéo 7** : vue de dessus, bruit assez fort, automobiles de couleurs et tailles variées et 1 bus, ombres des véhicules.

⇒ **vidéo 8** : identique à la vidéo 7, bruit faible.

Les caractéristiques de ces séquences sont regroupées dans le tableau 3.1.

Vidéo	Durée	Images	Taille	img/sec
1 (IPPR)	$00:59,80$	299	320×240	5
2	$00:28,96$	724	640×480	25
3	$00:29,00$	725	640×480	25
4	$00:39,24$	981	640×480	25
5	$00:23,04$	576	640×480	25
6	$00:23,04$	576	640×480	25
7	$00:26,84$	671	640×480	25
8	$00:26,48$	662	640×480	25

TABLE 3.1 – *Caractéristiques des séquences de test*

3.2.3 Méthodes de paramétrage d'extraction testés

Trois méthodes d'extraction sont testées ici, à savoir :

⇒ La mixture de gaussiennes (GMM)

⇒ Le codebook 2 layers (CB2)

⇒ Le Vumetre (VUM)

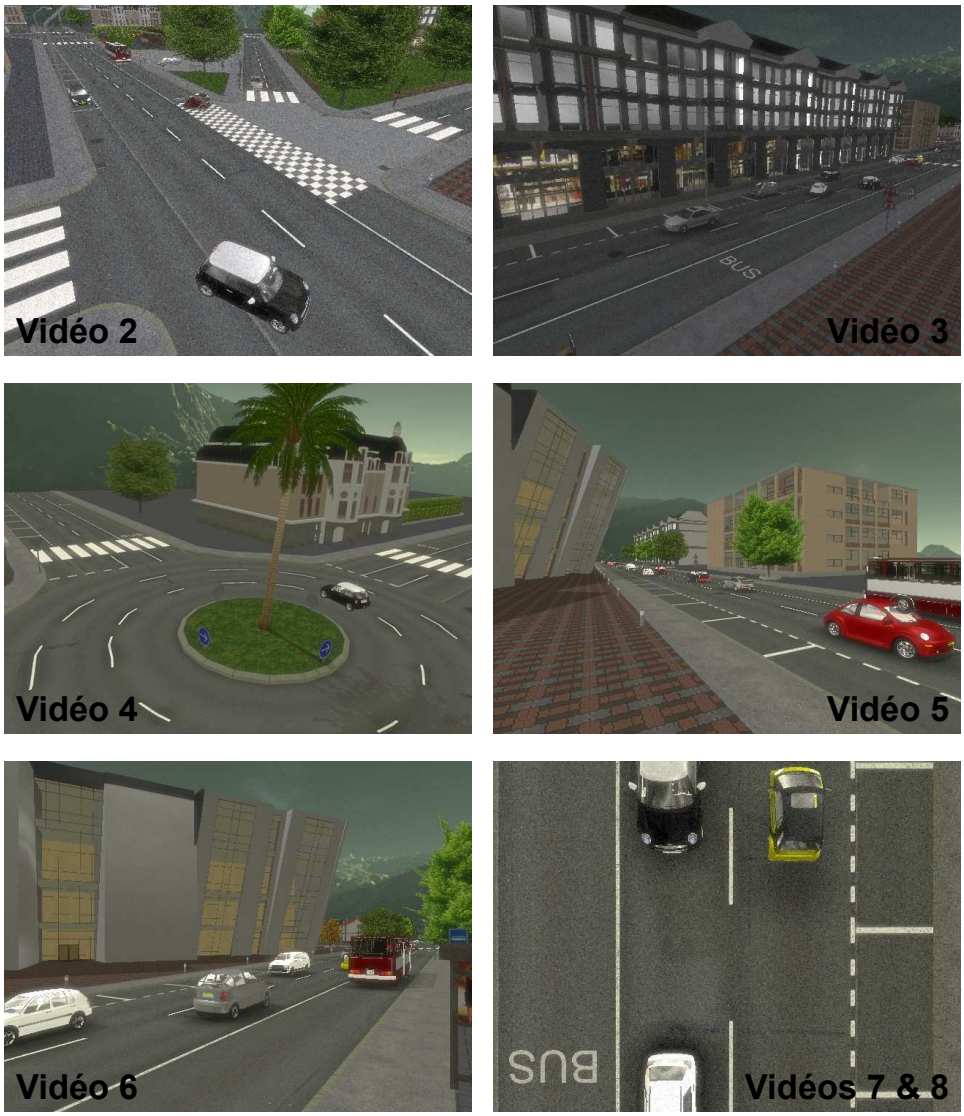


FIGURE 3.5 – *Vidéos simulées*

Étant donné que le paramétrage de l'algorithme d'extraction fond/forme lors des tests a une réelle importance pour la qualité de l'extraction, chaque séquence est testée avec les mêmes paramètres. On définit des configurations possibles pour chacune des méthodes (voir table 3.2), et on choisit les paramètres qui permettent d'obtenir les meilleurs résultats.

3.3 CRITÈRES DE COMPARAISON

À partir des résultats de l'extraction fond/forme, on obtient une image de forme binaire (pixels détectés comme fond ou forme). La vérité terrain est également de ce type. Par conséquent, il semble judicieux d'utiliser des critères existants et adaptés à ce type de comparaison.

	GMM		CB2		VUM	
config.	δ	K	α	β	α	T
1	0,9	0,01	0,4	1,1	0,01	0,2
2	0,8	0,01	0,4	1,3	0,01	0,4
3	0,5	0,01	0,4	1,5	0,01	0,1
4	0,9	0,005	0,55	1,1	0,005	0,2
5	0,8	0,005	0,55	1,3	0,005	0,4
6	0,5	0,005	0,55	1,5	0,005	0,1
7	0,9	0,02	0,7	1,1	0,02	0,2
8	0,8	0,02	0,7	1,3	0,02	0,4
9	0,5	0,02	0,7	1,5	0,02	0,1

TABLE 3.2 – *Paramètres des tests avec les trois méthodes*

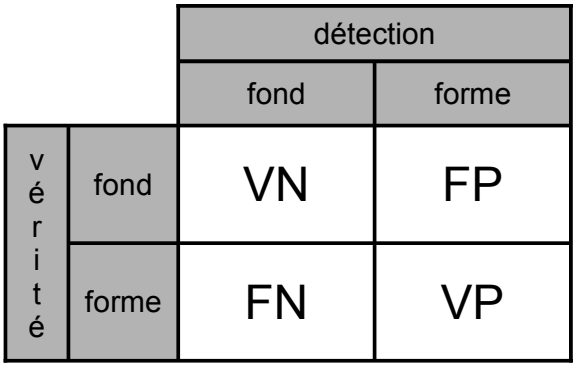


FIGURE 3.6 – *classification des pixels*

3.3.1 Classification

Tout d'abord, pour chaque pixel, en fonction de son état détecté (fond ou forme) et de la vérité terrain (fond ou forme), quatre classifications sont possibles (voir figure 3.6) :

⇒ vrai positif (VP) : correspond à un pixel détecté correctement comme forme (zone en jaune sur la figure 3.7)

⇒ vrai négatif (VN) : correspond à un pixel détecté correctement comme fond (zone en noir sur la figure 3.7)

⇒ faux positif (FP) : correspond à un pixel détecté par erreur comme forme (zone en rouge sur la figure 3.7)

⇒ faux négatif (FN) : correspond à un pixel détecté par erreur comme fond (zone en vert sur la figure 3.7)

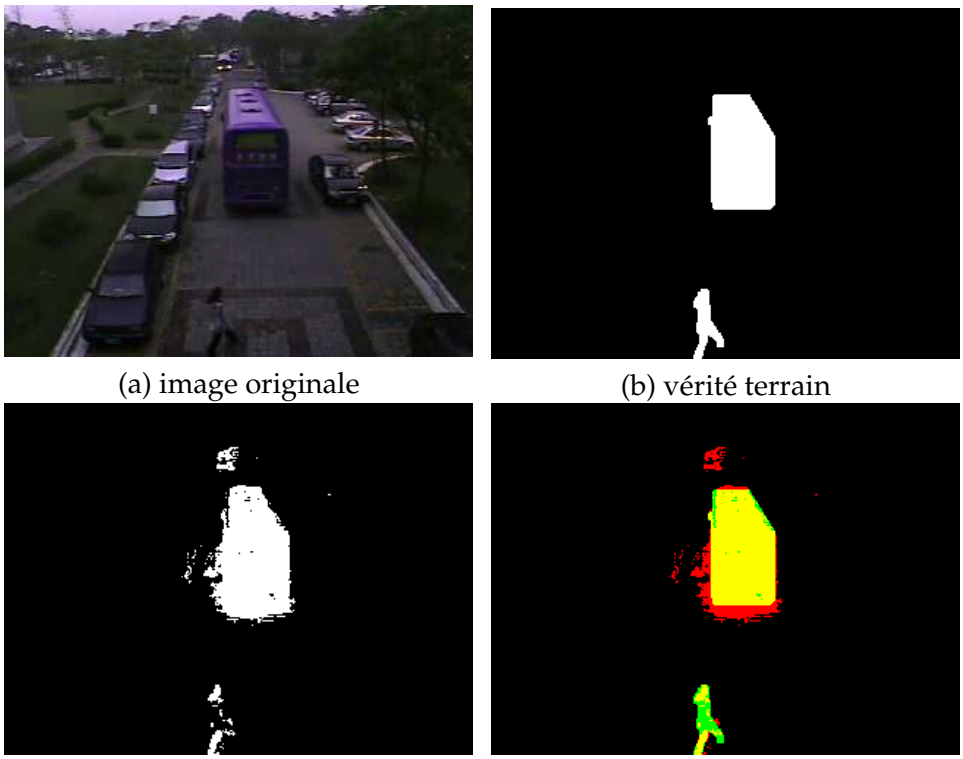
Pour chaque image i, on calcule le nombre de pixels de chaque catégorie, à savoir : VP_i, VN_i, FP_i, et FN_i.

3.3.2 Précision et Rappel

Le principe revient à calculer deux taux :

⇒ La précision (Prec) ou sensibilité (Se) :

$$Se_i = \frac{VP_i}{VP_i + FP_i} \tag{3.1}$$



(a) image originale (b) vérité terrain

(c) résultat extraction (d) classification zones

FIGURE 3.7 – *Classification des zones à partir de la vérité terrain*

⇒ Le rappel (Rap) ou spécificité (Sp) :

$$Sp_i = \frac{VN_i}{VN_i + FP_i} \tag{3.2}$$

La sensibilité reflète une bonne détection d'un objet, alors que la spécificité met plutôt en valeur la bonne détection du fond. L'idéal est d'avoir ces deux valeurs à 1.

3.3.3 Mesure Δ

Pour analyser la qualité de l'extraction à partir des pixels classés, on crée une mesure s'inspirant des classifications ROC (Receiver Operating Characteristic) appelée Δ [79].

Pour chaque image i d'une séquence, on calcule sa sensibilité et sa spécificité puis on place un point correspondant à cette image dans le repère ROC, soit Se_i en fonction de $1 - Sp_i$ (voir figure 3.8). Une détection parfaite est caractérisée par le point de coordonnées (0,1). Plus on sera proche de ce point idéal, plus l'extraction pourra être considérée comme bonne. La droite de non discrimination passant par les points (0,0) et (1,1) correspond aux cas où on ne parvient pas à différencier le fond de la forme.

Pour calculer la qualité de l'extraction, on mesure la distance Δ sur l'axe des ordonnées entre le point parfait (0,1) et la droite parallèle à la droite de non discrimination passant par le point à tester. On effectue cette mesure pour chaque image, puis on calcule la moyenne de ces distances. Pour une séquence de N images, soit pour N points de coordonnées x_i et y_i, cela

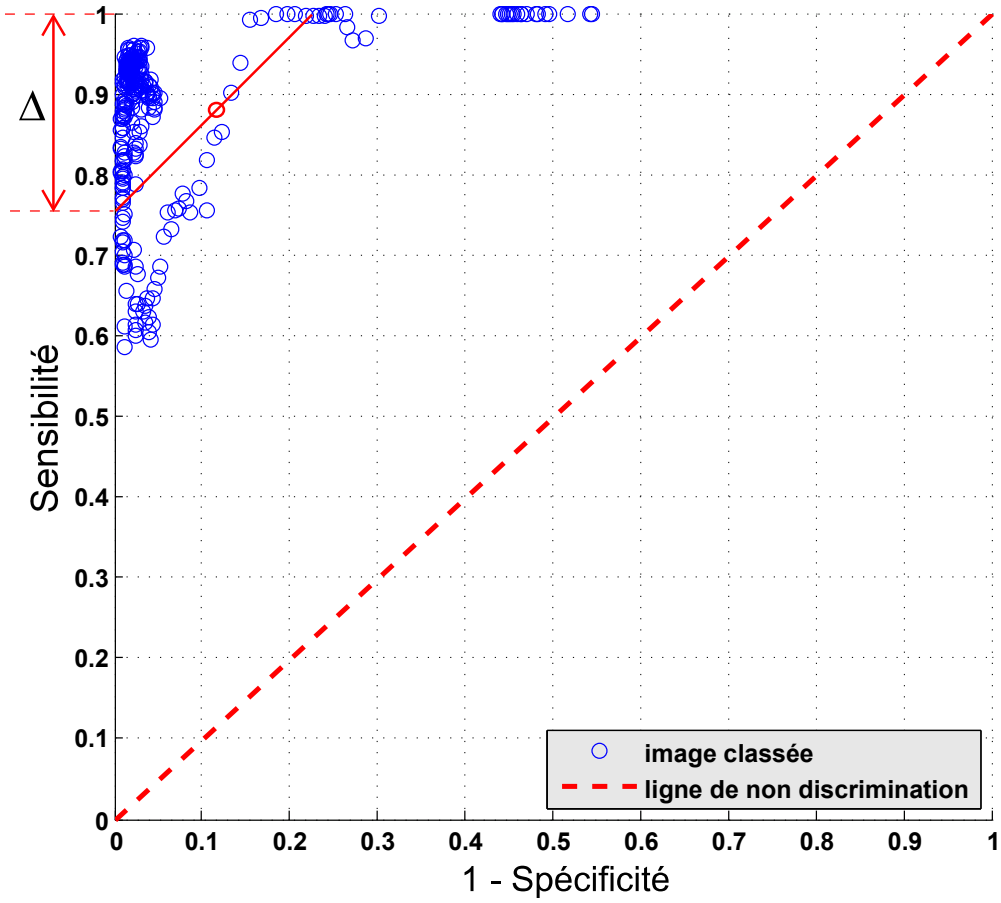


FIGURE 3.8 – *images dans un repère sensibilité/spécificité*

revient à calculer :

$$\Delta = \frac{1}{N} \sum_{n=1}^{n=N} 2 - Sp_i - Se_i \tag{3.3}$$

Le résultat de cette mesure est compris dans l'intervalle $[0; 2]$. Meilleure sera l'extraction, plus ce score devra s'approcher de 0. En revanche, un résultat proche de 1 est révélateur d'une mauvaise discrimination entre le fond et la forme. Un résultat supérieur à 1, montre quant à lui, une inversion des détections : le fond aura plus tendance à être détecté comme de la forme et inversement. Un score de 2 est révélateur d'une détection parfaite à ceci près qu'elle est inversée (le fond est détecté comme de la forme et vice - versa).

Limite : Le résultat de la mesure Δ a pour inconvénient de dépendre de la proportion de zones de forme dans l'image. Ainsi, une image ayant très peu de formes peut avoir plus de faux-positifs que de vrais positifs même si l'extraction semble correcte. L'exemple en figure 3.9 le montre bien : si on prend deux images de 100 pixels pour lesquelles 80% des pixels sont correctement détectés, on a alors deux notes très différentes selon que la forme représente 50% ou 10% de l'image.

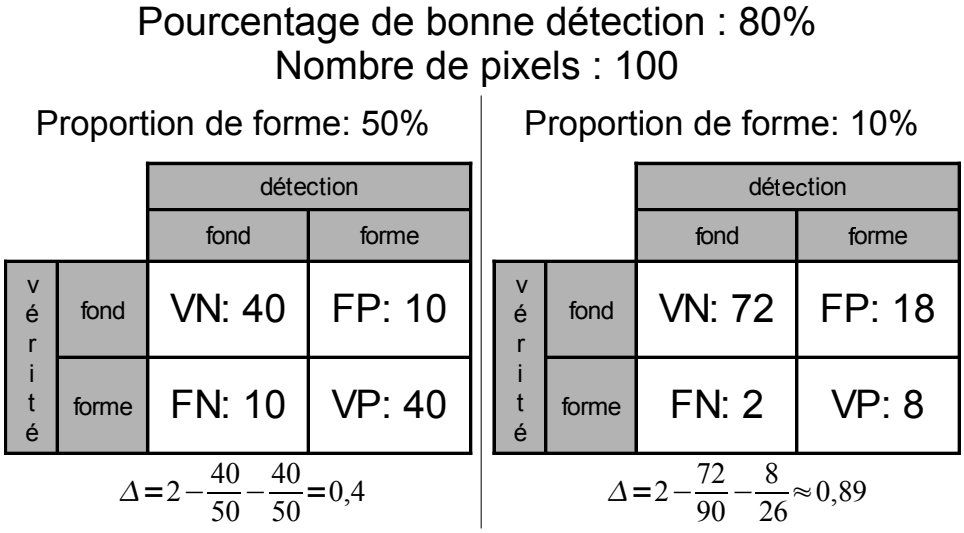


FIGURE 3.9 – *Calcul de* Δ

3.3.4 Mesure F

Un autre critère très intéressant et largement utilisé en statistique dans de nombreux domaines d'application [80] [81] [82] [83] est la mesure F. Pour cela, on calcule la précision et le rappel d'après les pixels classés suivant les quatre catégories décrites précédemment. On a donc :

$$Prec_i(P) = \frac{VP_i}{VP_i + FP_i} \tag{3.4}$$

$$Prec_i(N) = \frac{VN_i}{VN_i + FN_i} \tag{3.5}$$

$$Rap_i(P) = \frac{VP_i}{VP_i + FN_i} \tag{3.6}$$

$$Rap_i(N) = \frac{VN_i}{VN_i + FP_i} \tag{3.7}$$

$$Prec_i = (Prec_i(P) + Prec_i(N))/2 \tag{3.8}$$

$$Rap_i = (Rap_i(P) + Rap_i(N))/2 \tag{3.9}$$

$$F_i = 2 \times \frac{Prec_i \times Rap_i}{Prec_i + Rap_i} \tag{3.10}$$

Cette mesure donne une note comprise dans l'intervalle $[0; 1]$, elle correspond à la moyenne harmonique de la précision et du rappel. Plus son score se rapproche de 1, plus la classification pourra être considérée comme bonne. La note F globale attribuée à la séquence sera la moyenne de tous les F_i correspondant à chaque image de celle-ci. Ici, on utilise comme précision la moyenne entre la précision sur les positifs et les négatifs. Le rappel est également la moyenne des rappels positifs et négatifs. Ceci permet donc de prendre en compte les faux négatifs dans la mesure.

Remarque : Cette mesure, aussi appelée F_1, est un cas particulier de F_β dont l'équation est :

$$F_\beta = \frac{\left(1+\beta^2\right) \times \text{Précision} \times \text{Rappel}}{\beta^2 \times \text{Précision} + \text{Rappel}} \tag{3.11}$$

Le terme β permet donc de régler la pondération entre la précision et le rappel. Dans le cas où β=1, on donne la même pondération aux deux termes.

3.3.5 PSNR

Le PSNR (*Peak Signal to Noise Ratio*) est une mesure très utilisée en traitement d'images et particulièrement pour évaluer la dégradation de l'image suite à une compression. Il peut donc être utilisé pour comparer la dégradation du résultat de l'extraction par rapport à la vérité. Toutefois, il ne prend en compte la qualité visuelle de l'extraction et ne peut être considéré comme une mesure objective. Il est calculé de la façon suivante :

$$PSNR = 10 \times \log\left(\frac{d^2}{EQM}\right) \tag{3.12}$$

avec d l'écart maximal entre deux valeurs de pixel, soit pour une image binaire d=1. EQM correspond à l'erreur quadratique moyenne entre la vérité et le résultat de l'extraction, des images de taille $m \times n$, calculée de la façon suivante :

$$EQM = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left[extraction(i,j) - verite(i,j)\right]^2 \tag{3.13}$$

3.4 RÉSULTATS

3.4.1 Qualité d'extraction

On calcule la qualité de l'extraction comme décrit précédemment. Cette mesure est faite pour chacune des séquences avec chaque méthode et chaque configuration (voir table 3.2). On garde la configuration qui obtiendra les meilleurs résultats globalement pour toutes les méthodes. Les résultats obtenus avec ce réglage de paramètres optimums pour les différentes vidéos sont rapportés dans la table 3.3.

Étant donné que l'idéal est d'avoir Δ=0 et F=1, on remarque que la qualité est globalement plus mauvaise avec la mixture de gaussiennes, mais en revanche, la qualité d'extraction est à peu près identique de façon générale entre le VuMètre et le codebook 2 layers. Le VuMètre est meilleur sauf dans les deux scènes où le bruit est très exagéré. Ceci est dû à la largeur d'une classe qui ne peut pas contenir les variations liées au bruit.

3.4.2 Vitesse d'exécution

Les tests ont été réalisés sur un processeur Intel Xeon® E5520 cadencé à 2,26 GHz, la vitesse d'exécution dépend donc de ces caractéristiques. Pour que ces vitesses soient comparables, tous les algorithmes testés ont été codés en langage C.

	GMM *(config 3)*		CB2 *(config 9)*		VUM *(config 6)*	
vid	Δ	F	Δ	F	Δ	F
1 (IPPR)	0,317	0,813	0,152	0,884	0,188	0,914
2	0,517	0,615	0,528	0,643	0,517	0,613
3	0,570	0,724	0,393	0,796	0,501	0,828
4	0,366	0,705	0,315	0,766	0,317	0,786
5	0,364	0,850	0,396	0,854	0,298	0,883
6	0,433	0,816	0,475	0,822	0,357	0,859
7	0,293	0,798	0,195	0,870	0,266	0,728
8	0,244	0,904	0,144	0,928	0,103	0,919
moy	0,388	0,778	0,325	0,820	0,318	0,816

TABLE 3.3 – *Mesure de Δ et F*

Le temps moyen de calcul pour chaque séquence avec les différents paramètres possibles a été mesuré. On peut en déduire la vitesse moyenne de traitement d'une image avec chacune des méthodes (voir table 3.4).

La vitesse d'exécution est plus rapide avec le VuMètre, puis le codebook 2 layers et enfin la mixture de gaussiennes. Mais dans nos conditions de tests, aucune de ces trois méthodes ne permet de faire un traitement en temps réel à 25 images/s sur des images de taille 640×480.

	images/sec		
Séquence	GMM	CB2	VUM
vidéo 2	5,08	10,31	11,90
vidéo 3	5,21	10,31	13,70
vidéo 4	5,18	11,24	13,51
vidéo 5	5,24	10,64	13,51
vidéo 6	5,21	8,40	12,66
vidéo 7	5,13	9,80	13,33
vidéo 8	5,18	10,64	12,20
moyenne	5,13	9,90	12,66

TABLE 3.4 – *Comparaison de la rapidité d'exécution*

Bien évidemment les résultats présentés ici en termes de vitesse d'exécution sont dépendants de la qualité de l'implémentation réalisée. Néanmoins, les implémentations donnant ces résultats étant celles qui seront utilisées par la suite pour la chaine globale, comparer les méthodes selon ce critère secondaire est important pour départager des approches semblant donner des résultats similaires sur le plan qualitatif.

3.5 CHALLENGE VISAGE

3.5.1 Présentation

Suite aux essais de validation des méthodes d'extraction présentés précédemment, une base de vidéos simulées est créée. L'idée est alors de la compléter et de la rendre disponible à la communauté scientifique [79] [84]. Le travail de thèse a contribué au challenge Visage de RFIA 2012 par la

réalisation de la base de donnée de vidéos [2] permettant de comparer différentes méthodes. Le challenge Visage proposait un benchmark pour pouvoir tester des algorithmes d'extraction fond/forme vis à vis des différents facteurs dégradants dans des scènes routières. On disposait de 20 vidéos de 60 secondes permettant de tester l'influence de chacun de ces facteurs sur les différentes méthodes. La base de séquences vidéos a été créée afin de voir les résultats sur différents lieux, avec chacun des facteurs influents pris indépendamment, et avec ou non un délai laissé pour l'apprentissage.

Ces vidéos sont notées sur 3 chiffres pour les différencier :

⇒ le type d'événement se produisant sur la séquence :

1. pas d'événement (non bruité)
2. pas d'événement (avec bruit)
3. forte luminosité (avec bruit)
4. brouillard (avec bruit)
5. mouvement de branches d'arbres (avec bruit)

⇒ le type de scène (figure 3.10)

1. rue
2. rond-point

FIGURE 3.10 – *Scènes du benchmark visage*

⇒ le cas (voir figure 3.11)

1. événement toujours présent, pas de circulation pendant les 15 premières secondes (permet de vérifier le bon apprentissage de l'algorithme pendant les 15 premières secondes des conditions extérieures).
2. apparition de l'événement à 20 secondes et disparition de l'événement à 40 secondes, circulation pendant toute la séquence (permet d'évaluer la capacité à s'adapter à un événement arrivant en cours de séquence).

Ainsi chaque participant peut télécharger les séquences sur le site, les traiter avec son propre algorithme, et renvoyer les vidéos obtenues. Les vidéos sont alors comparées avec la vérité terrain pour évaluer l'efficacité du traitement. Les critères choisis pour les évaluer sont le rappel, la précision, la mesure F, et le PSNR.

2. http ://visage.univ-bpclermont.fr/ ?q=node/2

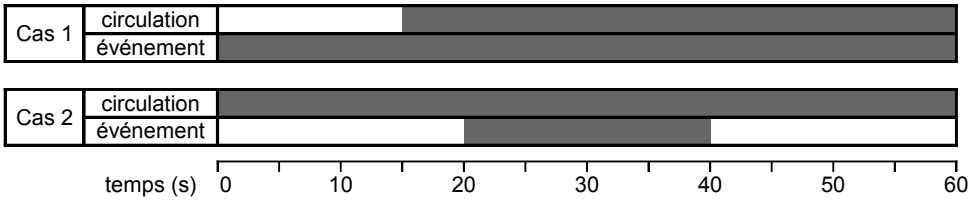


FIGURE 3.11 – *Cas utilisés pour le benchmark visage*

Les résultats obtenus sont présentés selon quatre zones intéressantes à analyser pour chaque séquence :

⇒ séquence totale (voir table 3.5)
⇒ zone "dynamic1" : 18 à 22s (voir table 3.6) correspond à la période d'apparition d'un événement.
⇒ zone "static" : 22 à 38s (voir table 3.8) correspond à la partie de la séquence sur laquelle l'événement est présent.
⇒ zone "dynamic2" : 38 à 42s (voir table 3.7) correspond à la période de disparition de l'événement.

Ces tests permettent donc d'analyser la capacité d'adaptation de l'algorithme face à différents événements.

Remarque : La combinaison du cas n°2 et de l'événement de type brouillard donnent une situation qui n'est pas réaliste, puisque le brouillard ne peut apparaitre en seulement quelques secondes.

3.5.2 Résultats de l'algorithme VuMètre

Compte tenu des résultats obtenus sur les séquences de test (voir paragraphe 3.4) qui donne un léger avantage à la méthode du VuMètre, nous avons axé notre contribution au challenge VISAGE sur l'évaluation plus fine de cette méthode. Les résultats obtenus sont présentés dans les tableaux 3.5 à 3.8 ci-dessous.

On constate que les résultats sont aussi bons avec ou sans bruit, ceci étant probablement dû à la largeur de la classe qui permet une tolérance d'erreur. Les résultats sont quasiment aussi bons avec la forte luminosité et le brouillard (voir figure 3.12), à l'exception des périodes d'adaptation. En revanche, on remarque des difficultés à faire face efficacement aux mouvements de branches d'arbres (événement 5), entrainant beaucoup de faux positifs (voir figure 3.13). Enfin les ombres entrainent beaucoup de faux positifs, comme on peut le voir par exemple sur cette même figure.

Séquence					Résultats			
idx	Evt	Bruit	Scène	Cas	Rappel	Précision	F-Mesure	PSNR
111	∅		rue	1	0.917451	0.621398	0.740946	40.9957
112	∅		rue	2	0.909645	0.611547	0.731388	41.1063
121	∅		rdpt	1	0.904069	0.657444	0.761281	38.2963
122	∅		rdpt	2	0.903566	0.650853	0.756667	38.5328
211	∅	√	rue	1	0.907914	0.573074	0.702642	35.3424
212	∅	√	rue	2	0.899416	0.566112	0.694862	35.397
221	∅	√	rdpt	1	0.89583	0.599242	0.718118	34.8495
222	∅	√	rdpt	2	0.895332	0.594716	0.7147	34.9209
311	lumin.	√	rue	1	0.890291	0.569462	0.69462	35.1928
312	lumin.	√	rue	2	0.873818	0.543181	0.669925	31.3443
321	lumin.	√	rdpt	1	0.877682	0.597791	0.71119	35.0574
322	lumin.	√	rdpt	2	0.868929	0.569917	0.688354	32.1592
411	brouil.	√	rue	1	0.81393	0.574271	0.673413	37.8291
412	brouil.	√	rue	2	0.854516	0.525669	0.650917	26.1576
421	brouil.	√	rdpt	1	0.772962	0.597859	0.674227	37.5835
422	brouil.	√	rdpt	2	0.832464	0.533943	0.650594	25.1701
511	arbres	√	rue	1	0.862995	0.517428	0.646958	20.4522
512	arbres	√	rue	2	0.881802	0.529271	0.661499	∞
521	arbres	√	rdpt	1	0.886719	0.564046	0.689499	30.1815
522	arbres	√	rdpt	2	0.89386	0.580381	0.703792	∞

TABLE 3.5 – *Résultats du VuMètre sur la séquence totale*

Séquence					Résultats			
idx	Evt	Bruit	Scène	Cas	Rappel	Précision	F-Mesure	PSNR
111	∅		rue	1	0.924736	0.921417	0.923074	90.1142
112	∅		rue	2	0.890228	0.921038	0.905371	94.1601
121	∅		rdpt	1	0.888881	0.929307	0.908644	86.8133
122	∅		rdpt	2	0.906269	0.91236	0.909304	82.9291
211	∅	√	rue	1	0.92204	0.924082	0.92306	90.1732
212	∅	√	rue	2	0.891208	0.922543	0.906605	94.3112
221	∅	√	rdpt	1	0.886994	0.932811	0.909326	91.3486
222	∅	√	rdpt	2	0.904217	0.91725	0.910687	87.7351
311	lumin.	√	rue	1	0.902743	0.917795	0.910207	89.3813
312	lumin.	√	rue	2	0.820318	0.513354	0.63151	57.314
321	lumin.	√	rdpt	1	0.875249	0.930647	0.902098	91.1464
322	lumin.	√	rdpt	2	0.834494	0.543486	0.658262	62.238
411	brouil.	√	rue	1	0.822407	0.996578	0.901154	91.1673
412	brouil.	√	rue	2	0.749992	0.505558	0.603982	49.0375
421	brouil.	√	rdpt	1	0.757442	0.99502	0.860127	91.2105
422	brouil.	√	rdpt	2	0.76077	0.510813	0.611224	47.2493
511	arbres	√	rue	1	0.877127	0.536798	0.666004	59.3407
512	arbres	√	rue	2	0.842033	0.516271	0.640088	58.8394
521	arbres	√	rdpt	1	0.876808	0.610234	0.719627	74.8374
522	arbres	√	rdpt	2	0.898423	0.660947	0.761602	75.2492

TABLE 3.6 – *Résultats du VuMètre sur la zone dynamique 1 (18 à 22s)*

Séquence					Résultats			
idx	Evt	Bruit	Scène	Cas	Rappel	Précision	F-Mesure	PSNR
111	∅		rue	1	0.914452	0.920442	0.917437	95.277
112	∅		rue	2	0.891195	0.894034	0.892612	95.8383
121	∅		rdpt	1	0.86399	0.921976	0.892042	87.7173
122	∅		rdpt	2	0.916133	0.917643	0.916887	82.8404
211	∅	√	rue	1	0.907063	0.922385	0.91466	95.1708
212	∅	√	rue	2	0.881734	0.902129	0.891815	96.0604
221	∅	√	rdpt	1	0.861846	0.926806	0.893146	90.8769
222	∅	√	rdpt	2	0.914496	0.922868	0.918663	88.1107
311	lumin.	√	rue	1	0.878399	0.924169	0.900703	94.497
312	lumin.	√	rue	2	0.838316	0.512263	0.635932	58.6561
321	lumin.	√	rdpt	1	0.824082	0.93234	0.874875	90.6666
322	lumin.	√	rdpt	2	0.876284	0.545543	0.672446	61.5091
411	brouil.	√	rue	1	0.827779	0.998277	0.905068	96.6337
412	brouil.	√	rue	2	0.800084	0.506946	0.620642	53.2528
421	brouil.	√	rdpt	1	0.630102	0.996541	0.772047	89.2715
422	brouil.	√	rdpt	2	0.812053	0.518625	0.632987	52.6145
511	arbres	√	rue	1	0.869563	0.522782	0.652987	59.8234
512	arbres	√	rue	2	0.85412	0.522092	0.648053	64.6579
521	arbres	√	rdpt	1	0.853784	0.598922	0.703997	74.8635
522	arbres	√	rdpt	2	0.911887	0.725409	0.808029	79.0566

TABLE 3.7 – *Résultats du VuMètre sur la zone dynamique 2 (38 à 42s)*

Séquence					Résultats			
idx	Evt	Bruit	Scène	Cas	Rappel	Précision	F-Mesure	PSNR
111	∅		rue	1	0.935194	0.894678	0.914487	60.6395
112	∅		rue	2	0.933778	0.890537	0.911645	67.2529
121	∅		rdpt	1	0.917871	0.910272	0.914056	51.2781
122	∅		rdpt	2	0.913491	0.909431	0.911456	54.4835
211	∅	√	rue	1	0.932972	0.89952	0.91594	60.8659
212	∅	√	rue	2	0.932298	0.898409	0.91504	67.5794
221	∅	√	rdpt	1	0.915723	0.915002	0.915362	57.5676
222	∅	√	rdpt	2	0.911346	0.913749	0.912546	60.1421
311	lumin.	√	rue	1	0.912525	0.8936	0.902963	60.2172
312	lumin.	√	rue	2	0.915897	0.89473	0.90519	67.1678
321	lumin.	√	rdpt	1	0.89738	0.913563	0.905399	57.2741
322	lumin.	√	rdpt	2	0.883237	0.877428	0.880323	58.4628
411	brouil.	√	rue	1	0.84718	0.996753	0.9159	63.8026
412	brouil.	√	rue	2	0.839102	0.589505	0.6925	48.5269
421	brouil.	√	rdpt	1	0.798077	0.993977	0.88532	58.5127
422	brouil.	√	rdpt	2	0.773164	0.647747	0.70492	48.569
511	arbres	√	rue	1	0.888365	0.537563	0.669812	30.78
512	arbres	√	rue	2	0.885851	0.518383	0.654036	29.8594
521	arbres	√	rdpt	1	0.906694	0.670106	0.770651	44.8869
522	arbres	√	rdpt	2	0.904984	0.645603	0.753599	46.0042

TABLE 3.8 – *Résultats du VuMètre sur la zone static (22 à 38s)*

FIGURE 3.12 – *Détection avec du brouillard et du bruit*

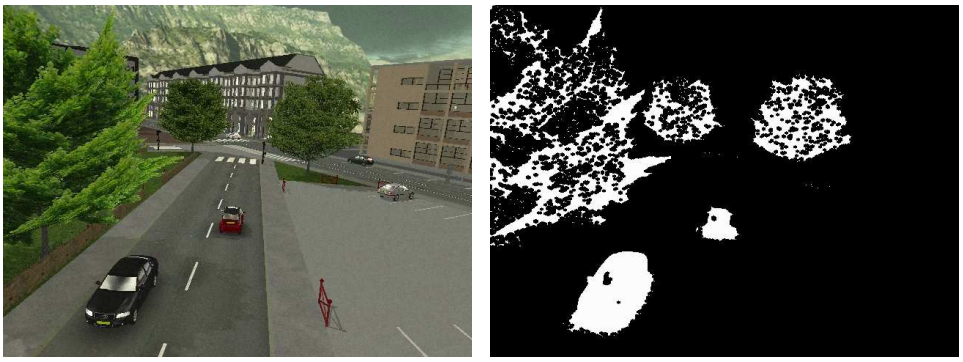


FIGURE 3.13 – *Détection avec des mouvements d'arbres et du bruit*

3.6 Discussion

Les différents essais montrent des résultats sensiblement identiques entre le codebook 2 layers et le VuMètre, ainsi qu'une qualité d'extraction légèrement en dessous en ce qui concerne la mixture de gaussiennes. Cependant, ces résultats sont à analyser avec précaution.

La mixture de gaussienne, du fait de sa précision requiert un paramétrage précis pour bien fonctionner. Ce paramétrage idéal peut varier en fonction de la scène, et d'autres conditions.

Les deux autres méthodes (codebook 2 layers et VuMètre) sont plus approximatives, en ce qu'elles définissent une certaine tolérance (liée à la largeur d'une classe pour le VuMètre ou à la taille du codebook). Cette tolérance est efficace contre les phénomènes de bruit mais peut dégrader la précision de la détection dans certains cas.

De plus, les essais ont été réalisés avec un temps d'apprentissage de la scène relativement court. Les mouvements de branches d'arbre étant assez longs, ils n'ont donc pas pu être complètement assimilés dans le modèle de fond des algorithmes. Un temps d'apprentissage plus long n'est pas nécessairement meilleur pour la détection, car, ces mouvements de branches sont rarement purement répétitifs.

Du fait de la tolérance liée à la discrétisation ainsi que de son temps de calcul, la méthode du VuMètre semble donc la mieux adaptée à notre cas.

3.7 Conclusion

Nous avons pu étudier ici les enjeux de l'extraction fond/forme sur une scène de type routière. Pour cette étude, trois méthodes furent sélectionnées de l'état de l'art et testées par l'intermédiaire d'une base de vidéos simulées avec leurs vérités associées, permettant ainsi d'analyser précisément les résultats des algorithmes face aux différents facteurs susceptibles d'altérer l'extraction. Cette étude a conduit à choisir la méthode appelée VuMètre.

La méthode choisie n'est pas parfaite. Des problèmes restent présents notamment en ce qui concerne l'identification et l'élimination des ombres. Cet inconvénient n'est cependant pas rédhibitoire puisque l'extraction fond/forme ne sert qu'à sélectionner une zone de l'image qui sera ensuite traitée par une approche de stéréovision qui, elle, gérera les problèmes d'ombre. En revanche, pour éviter que trop de zones de la forme soient détectées comme fond, un filtrage de type dilatation sera intéressant.

Cette partie "en mouvement" extraite, seulement 10% de l'image, voire moins, va devoir être exploitée. Nous allons voir au chapitre suivant le traitement effectué sur ces formes pour obtenir une carte 3D via une méthode innovante de stéréovision.

APPLICATION DES MÉTHODES DE STÉRÉOVISION

4

SOMMAIRE

DANS ce chapitre, nous étudions l'application de la stéréovision à la détection des deux-roues sur une intersection, en milieu urbain. Nous regardons tout d'abord les facteurs susceptibles de dégrader un traitement stéréo sur une scène routière et nous analysons leurs impacts sur la détection des deux-roues. Puis, le système est dimensionné pour une prise de vue stéréo optimale de la scène. Ensuite, la problématique de calibration de

ce système est exposée. Enfin, nous regardons les adaptations nécessaires aux méthodes issues de l'état de l'art pour permettre un traitement stéréo, étant donnée la disposition particulière et le type des caméras.

4.1 Limites de la stéréovision sur des scènes routières

Différentes situations sont susceptibles de poser problème dans la mise en œuvre d'un système de stéréovision et surtout dans le traitement de ses images. Nous les analysons dans ce paragraphe.

4.1.1 Surfaces homogènes

Le calcul de la carte de disparité peut être en partie dégradé par la présence de surfaces homogènes. Prenons l'exemple d'une partie de la scène qui se traduit par une surface homogène sur les images prises par les deux caméras. Toutes les fenêtres définies sur cette partie des images sont sensiblement identiques (au bruit près). Il est difficile d'apparier une fenêtre de l'image droite à une autre fenêtre de l'image gauche. Il est donc impossible d'apparier sans ambiguïté les pixels qui représentent le même point de la zone homogène dans les deux images. C'est en fait le bruit qui pilotera cet appariement, et conduira à des résultats sans aucune réalité physique.

Dans notre cas, ces phénomènes liées aux surfaces lisses s'observent principalement sur les murs des bâtiments et sur les carrosseries des gros véhicules. Les deux roues y sont très peu soumis du fait de leur petite taille et de l'absence de grandes surfaces de carrosserie.

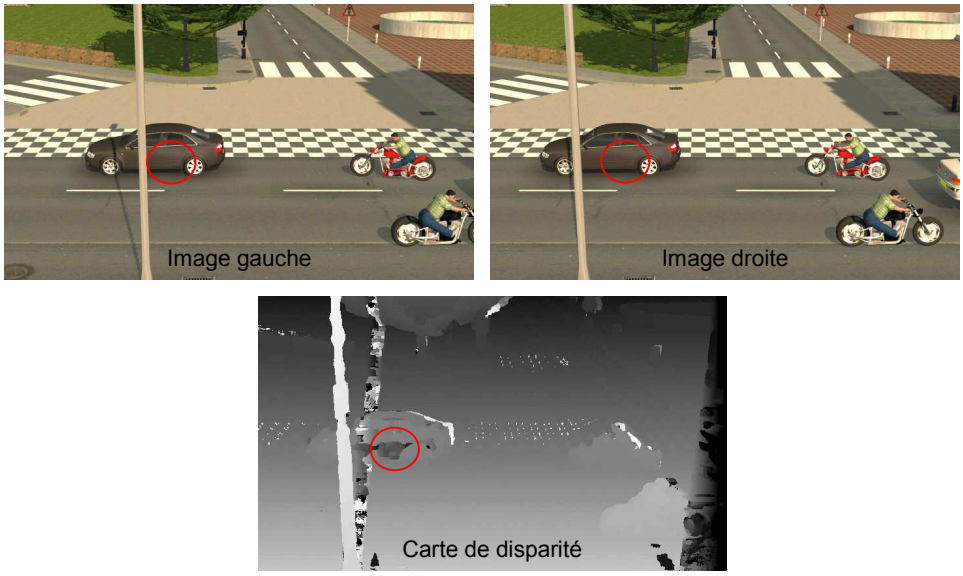


FIGURE 4.1 – *Erreurs de mises en correspondances sur surfaces lisses*

Sur les images simulées, on remarque également un phénomène plus marginal de mauvaise détection lié à un problème très similaire aux surfaces homogènes, à l'emplacement des damiers au sol. À cet endroit, la texture est répétitive et une erreur de mise en correspondance est également possible.

4.1.2 Reflets

Un autre phénomène qui perturbe l'analyse stéréoscopique est lié aux surfaces réfléchissantes. En effet, sur ce type de surfaces, les caméras n'étant pas positionnées au même endroit, il est tout à fait possible qu'un reflet soit perçu de manière complètement différente par les deux capteurs, rendant la mise en correspondance impossible. Cet effet survient principalement sur les carrosseries des véhicules (voir figure 4.2), les pare-brises, et finalement sur toutes les grandes surfaces lisses. Il est dépendant de la position de l'objet réfléchi, des surfaces réfléchissantes et des caméras. Là encore, les deux-roues et les piétons y sont peu soumis du fait de l'absence de pare-brise et de carrosserie. Notons que le problème des reflets étant en partie lié au fait que les points de vue ne sont pas les mêmes, écarter davantage les caméras risque d'accentuer ce problème.

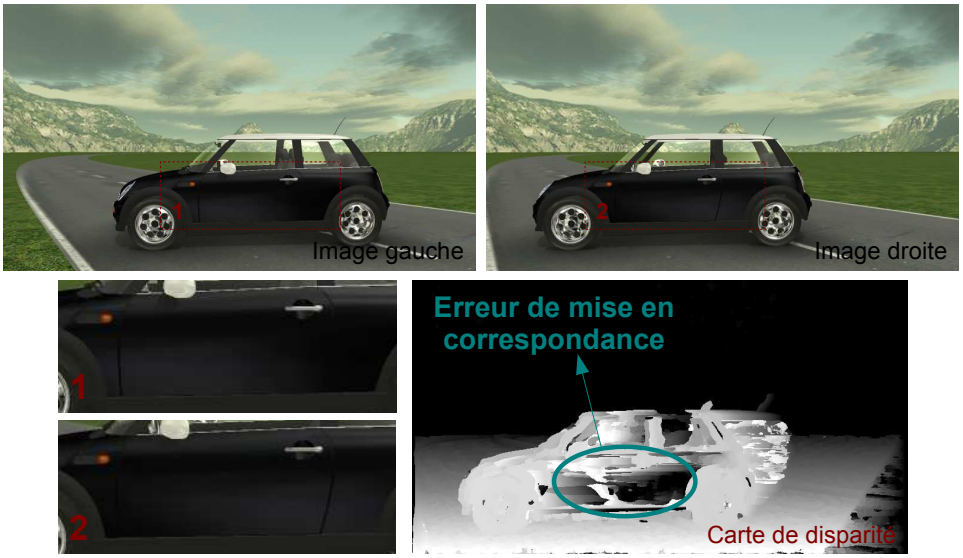


FIGURE 4.2 – *Influence des reflets sur la mise en correspondance*

4.1.3 Occultations, masquage

Les objets ne sont pas perçus de la même manière selon deux points de vue différents, et en particulier, des parties peuvent être visibles sur une des caméras, et ne pas l'être sur l'autre. Il devient alors impossible de trouver, pour une zone visible sur une caméra, la zone correspondante perçue par l'autre caméra. C'est typiquement le cas pour les bords de la scène (voir figure 4.1). Ces erreurs d'appariement sont aussi illustrées sur la figure 4.3, où des zones masquées par le panneau ne peuvent pas être retrouvés sur l'autre image. Plus on écarte les caméras, plus ce phénomène sera présent car les points de vue seront de plus en plus différents.

Par ailleurs, un grand véhicule, de type bus, peut en cacher un petit, de type vélo. On peut différencier les occultations entres véhicules en deux catégories (voir figure 4.4) :

⇒ Les masquages de véhicules côte à côte (cas 1 sur la figure 4.4)
⇒ Les masquages de véhicules se suivant (cas 2 sur la figure 4.4)

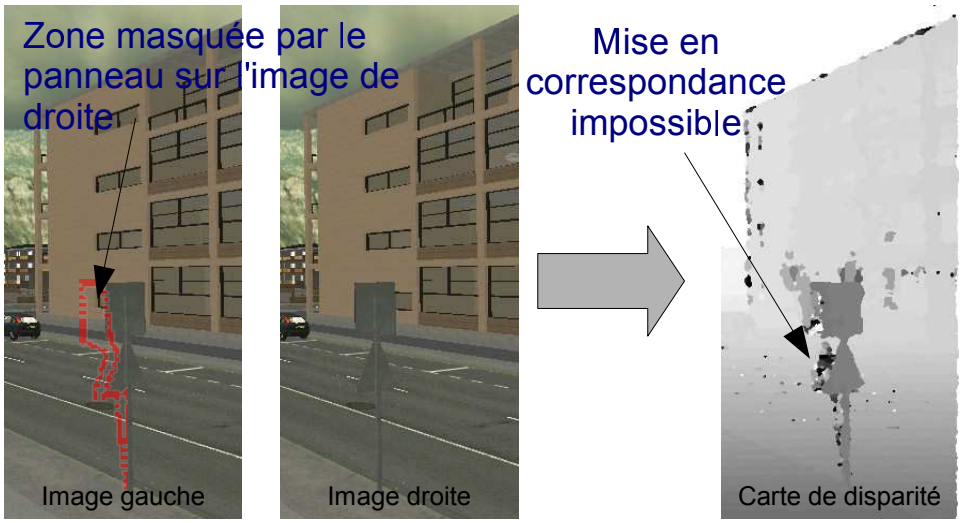


FIGURE 4.3 – *Conséquence des occultations sur le calcul de carte de disparité*

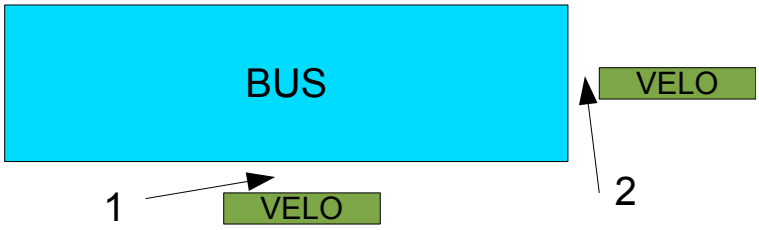


FIGURE 4.4 – *Masquages entre véhicules (vue de dessus)*

Lorsque les véhicules sont côte à côte, la distance qui les sépare peut être relativement faible. La faible largeur du véhicule masqué (cas d'un deux roues) peut accentuer ce phénomène. Si les deux véhicules roulent à la même vitesse, il peut arriver que l'un d'eux ne soit pas du tout visible sur la scène. Lorsque les véhicules se suivent, la distance inter-véhiculaire est généralement plus grande, sauf si ces véhicules sont très proches dans un flot à très faible vitesse ou à l'arrêt, dans un embouteillage ou à un stop par exemple. Mais, dans le cas d'un stop, les véhicules finiront par circuler, la distance deviendra donc plus importante. On voit donc très clairement que le cas le plus critique à prendre en compte est le masquage des véhicules circulant l'un à côté de l'autre.

4.1.4 Bruit

Tout comme pour l'extraction fond/forme (voir partie 3.1), il y a toujours un bruit dans l'image quelle que soit la caméra. Celui-ci étant aléatoire et propre à chaque capteur, les deux images ont donc un bruit différent, ce qui nuit à la recherche de mise en correspondance.

4.1.5 Différence de luminosité

Deux caméras de modèle identique n'ont pas nécessairement la même sensibilité. De plus, n'étant pas situées exactement au même endroit, elles peuvent ne pas percevoir la même luminosité. Pour contrer ce pro-

blème, les méthodes de mesure présentant des invariances de biais (décrit en 2.2.4.1), comme ZSAD, ZSSD, ZNCC et CENSUS sont mieux adaptées (voir l'état de l'art en partie 2.2.4).

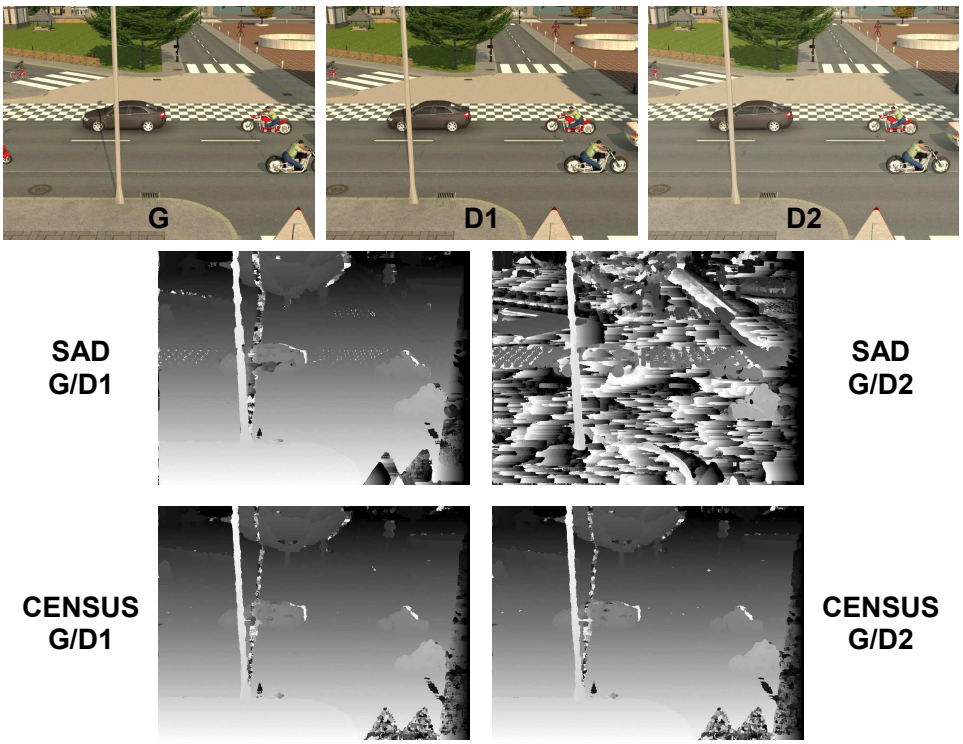


FIGURE 4.5 – *Comparaison de robustesse de deux méthodes stéréo pour une différence de luminosité*

La figure 4.5 montre un test faisant ressortir ce phénomène. Une image gauche notée G est comparée avec deux images droites identiques à ceci prêt qu'une d'entre elle (notée D1 sur la figure 4.5) présente la même luminosité que l'image gauche, et la seconde (notée D2 sur la figure 4.5) est légèrement plus claire. On remarque bien que, grâce à ses invariances de type gain et biais, la méthode CENSUS est très nettement plus robuste vis à vis de cette variation de luminosité que la méthode SAD.

4.2 DIMENSIONNEMENT DU SYSTÈME

4.2.1 Configuration de la scène

Notre problématique principale consiste à détecter les deux roues dans les intersections en milieu urbain. Pour cela, il est important de définir les dimensions de la zone à observer et ses caractéristiques :

⇒ La configuration d'une intersection est très variable.

⇒ Elle comporte au moins 3 accès, généralement 4 ; les véhicules peuvent donc venir de différents angles.

⇒ De nombreux véhicules sont à l'arrêt ou circulent de façon très lente ; ils peuvent donc être très proches les uns des autres, d'où un risque accru de masquages (voir ci-dessus).

Le fait de vouloir observer un carrefour impose donc d'avoir un large champ de vision. Notons que la prise de vue devant être assez générale, elle peut, de ce fait, convenir à d'autres types de zones urbaines à observer.

4.2.2 Choix de l'emplacement du système de vision

Pour suivre les véhicules sur un carrefour en milieu urbain, on a choisi de réaliser une détection sur un rayon de 25 mètres. La solution consistant à placer les capteurs en hauteur devient alors inévitable, pour limiter au maximum les effets de masquage des véhicules entre eux. De plus, la position du capteur doit être la plus centrale possible dans l'intersection, pour couvrir au mieux les différentes voies. Un positionnement du système au centre de l'intersection paraît néanmoins très difficile à mettre en œuvre. En revanche, un placement sur un angle de l'intersection est tout à fait envisageable (voir figure 4.6). Cet emplacement a également l'avantage de pouvoir positionner les éléments du système masquant une partie de la scène dans une zone sans intérêt du champ de vision des caméras. C'est donc lui que nous avons choisi.

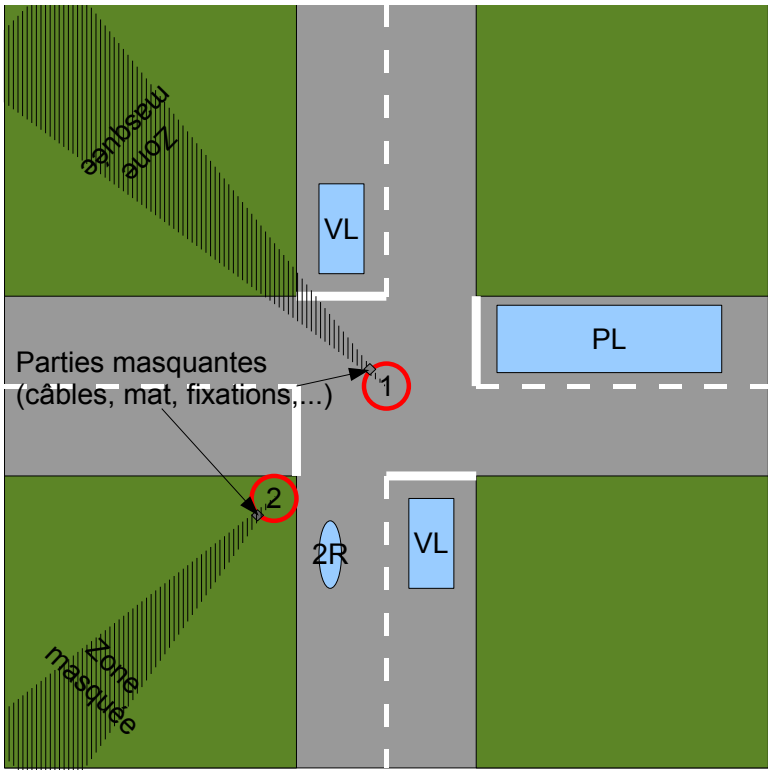


FIGURE 4.6 – *Placement du système sur une intersection*

Bien entendu, malgré la hauteur de la caméra, on ne peut garantir totalement un non masquage des deux roues dans le cas où ceux-ci circulent à côté d'un véhicule de grande dimension, de type bus ou poids-lourd. Cette difficulté peut, dans certains cas, être surmontée lors d'une étape ultérieure effectuée par le logiciel de suivi de véhicules (hors du champ de cette thèse).

4.2.2.1 Positionnement relatifs des caméras pour la stéréovision

En ce qui concerne le placement des caméras l'une par rapport à l'autre, deux approches peuvent être envisagées :

⇒ De nombreuses applications de stéréovision dans divers domaines utilisant des caméras fisheye existent [85] [86] mais la plupart d'entre elles utilisent deux capteurs côte à côte sur un support horizontal. En théorie, cette solution est possible. Étant donnée la distance des objets à suivre, il est indispensable d'utiliser un écartement important entre les deux caméras, de l'ordre du mètre. Un système stéréo placé à une hauteur d'environ 8m avec un tel écartement des caméras devient difficile à mettre en place pour des raisons évidentes de stabilité, de prise au vent et de mise en œuvre (voir la partie droite de la figure 4.7).

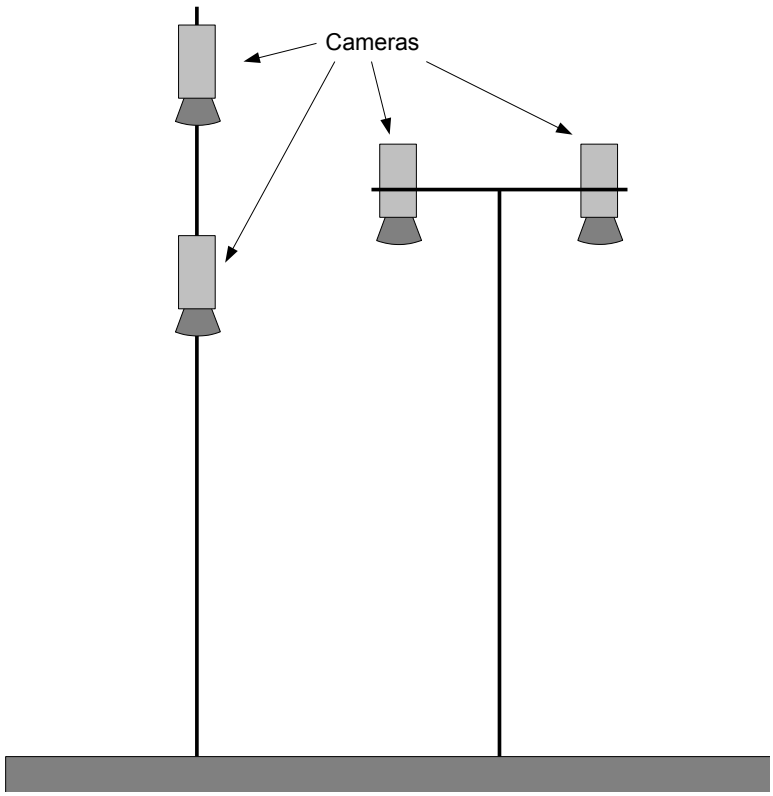


FIGURE 4.7 – *Dispositions stéréovision fisheye*

⇒ Une autre approche, inspirée des travaux que Ragot [87] a effectués avec des capteurs catadioptriques, consiste à placer les deux caméras alignées et orientées selon le même axe vertical (voir la partie gauche de la figure 4.7). Cette disposition à l'avantage d'être beaucoup plus stable et simple à installer à une hauteur importante. Néanmoins, avec cette disposition, la disparité angulaire (écart entre les deux angles α d'un même objet vu par chacune des caméras, illustré sur la partie gauche de la figure 4.8) est nulle au centre des images, c'est à dire sur l'axe passant par les deux caméras, et variable selon la distance par rapport à cet axe. Pour un objet donné,

plus la différence de disparité est grande, meilleure est la précision sur la mesure de sa hauteur.

In fine, nous avons retenu la seconde configuration : caméras alignées l'une au dessus de l'autre.

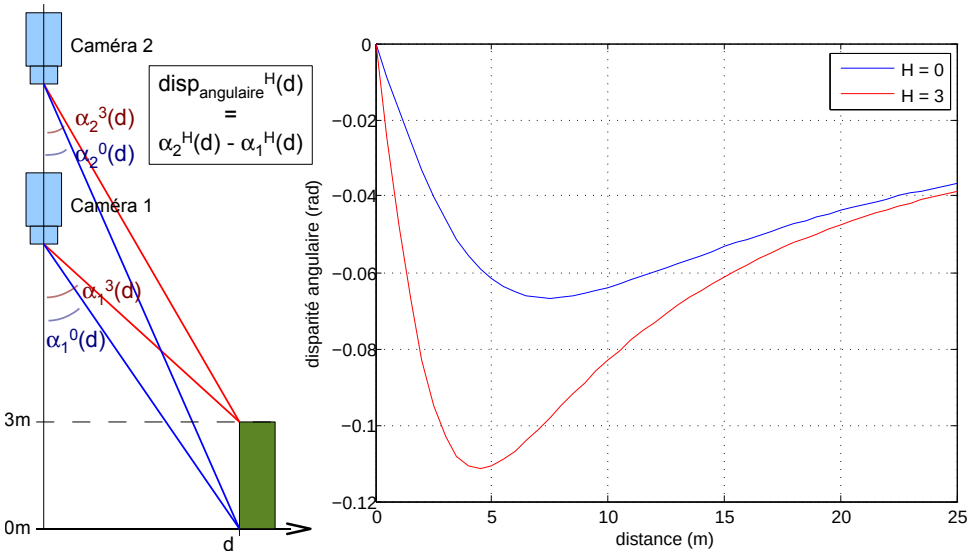


FIGURE 4.8 – *Disparité angulaire autour du système à 0 et 3 m de haut*

4.2.2.2 Écartement des caméras

Dans la pratique, plus l'écartement est important, plus la disparité obtenue est importante. Néanmoins, un écartement trop important est pénalisant pour différentes raisons :

⇒ Le système devient beaucoup plus complexe à réaliser et à installer sur site.

⇒ La différence de point de vue accentue les phénomènes d'occultations (zones visibles par une caméra mais pas par l'autre). Ainsi, plus l'écartement augmente, plus cette différence risque de se traduire par des erreurs lors du traitement en stéréovision.

L'écartement retenu (1m) réalise un bon compromis entre ces avantages et ces inconvénients.

4.2.3 Choix du type de capteurs optiques

La position du système de vision au-dessus de la scène, et l'écartement entre les caméras (les capteurs optiques) ayant été choisis, il reste à définir plus précisément le type des capteurs optiques qui doit être utilisé. Compte-tenu des éléments issus de l'état de l'art et de la configuration de la scène à observer (§ 4.2.1), ce choix ne pouvait se faire qu'entre des caméras directionnelles, des capteurs à miroir catadioptrique et des caméras à optique fisheye. Des simulations d'observations ont permis de trancher.

4.2.3.1 Simulations

Afin de tester les différents types de capteurs envisageables, ainsi que les différentes dispositions de caméras pour que la prise de vue en stéréo-

vision soit optimale, on a effectué des essais à l'aide d'un simulateur. Nous avons opté pour le simulateur appelé Sivic [78], qui a été développé dans le but de tester les différents capteurs dans le domaine routier. Il modélise également différents types d'environnements routiers, dont des scènes urbaines, sur lesquelles il est possible de faire circuler différents types de véhicules selon des scénarios prédéfinis. On peut simuler l'acquisition de cette circulation à l'aide de différents modèles de systèmes optiques. Ce simulateur nous a permis de tester en bureau les différentes technologies de capteurs et leurs dispositions telles qu'elles ont été présentées précédemment. L'environnement testé est un carrefour en milieu urbain.

4.2.3.2 Test des capteurs

Comme expliqué précédemment, un **capteur directionnel** a pour inconvénient de couvrir un champ de vision très réduit. Si bien que si l'on souhaite utiliser ce type de capteurs et couvrir une zone maximale, il doit être orienté de manière oblique comme cela est montré sur la figure 4.9. Différents problèmes apparaissent alors :

- ⇒ Les voies donnant sur le carrefour ne sont pas toutes visibles ;
- ⇒ Pour voir toute la zone du carrefour, on se retrouve obligé de placer la caméra avec beaucoup de recul. L'angle de prise de vue accentue les problèmes de masquage entre véhicules
- ⇒ La définition des objets sur l'image est très variable en fonction de leur position sur la scène

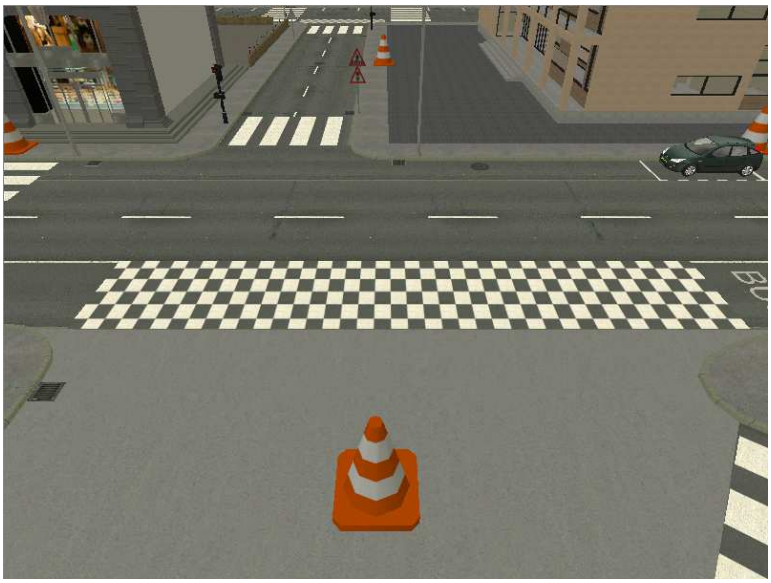


FIGURE 4.9 – *Prise de vue avec capteur classique*

En revanche, en plaçant un capteur de vision omnidirectionnel (fisheye ou à miroir catadioptrique) à une hauteur de l'ordre de 10 mètres, on remarque que la zone peut-être couverte sans aucun problème du fait du grand champ de vision.

Les figures 4.10 et 4.11 présentent un exemple de **capteur à miroir catadioptrique** parabolique associé à une caméra dirigée vers le haut. La figure 4.12 illustre les simulations de prise de vue associées. L'équation du

miroir est :

$$z = \frac{x^2 + y^2}{0,09} \tag{4.1}$$

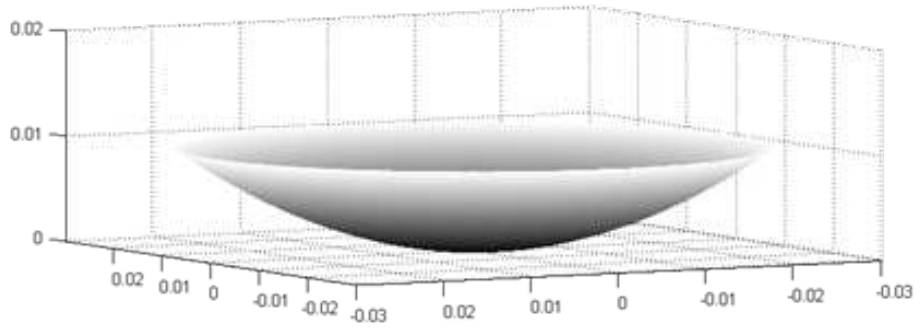


FIGURE 4.10 – *Miroir modélisé (vue 3D)*

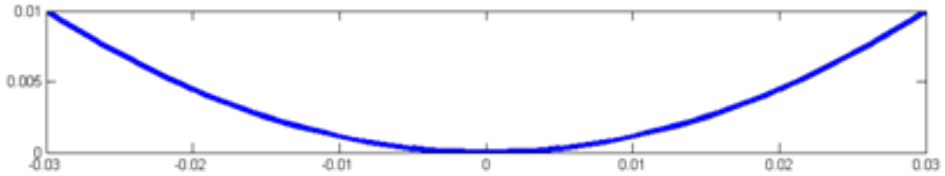


FIGURE 4.11 – *Miroir modélisé (vue en coupe)*

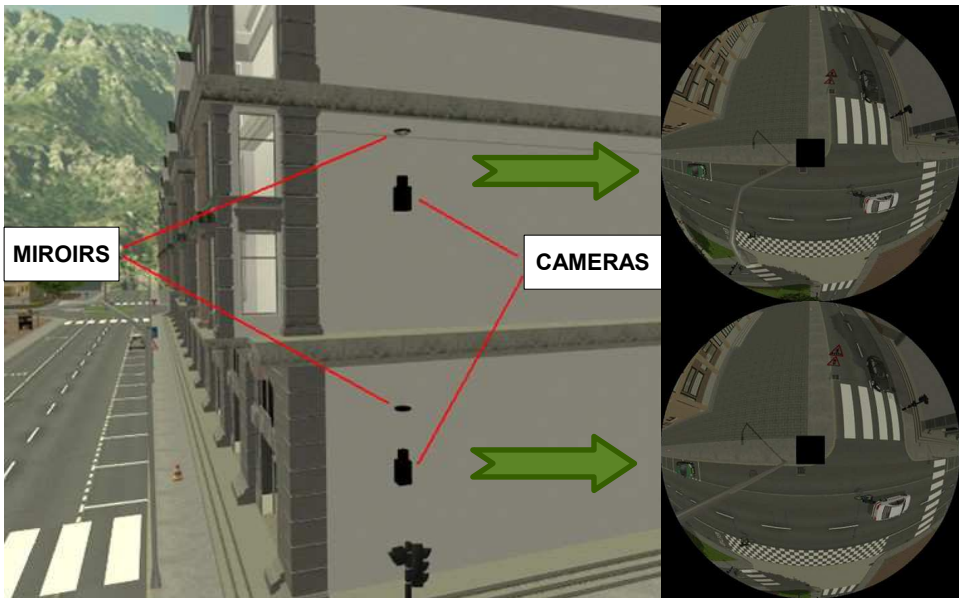


FIGURE 4.12 – *Simulation de prise de vue stéréo avec miroir*

La prise de vue est également modélisée avec une **optique fisheye** placée à différentes hauteurs mais, contrairement au cas précédent, avec une caméra orientée vers le bas de façon à filmer le sol (voir figure 4.13). Ici l'optique fisheye est simulée avec un angle d'ouverture de 185°. On remarque que la scène est parfaitement couverte par le capteur, et que toutes les voies

donnant sur l'intersection sont visibles. Par conséquent les optiques fisheye peuvent parfaitement convenir à notre besoin en termes de champ de vision avec des capteurs orientés vers le bas.

FIGURE 4.13 – *Simulation d'un capteur fisheye d'ouverture 185° à 7 m du sol*

Pour des raisons évidentes de faible champ de vision, telles qu'expliquées au dessus, la solution des capteurs classiques est écartée. En revanche, les capteurs omnidirectionnels sont parfaitement adaptés. Pour la réalisation du système final, le traitement stéréo demande une grande précision lors de la mise en place des caméras pour garantir un bon alignement permettant une mise en correspondance efficace. Un système stéréo à miroir catadioptrique est plus délicat à installer car les caméras ainsi que les miroirs doivent être correctement alignés. Pour cette raison, notre choix pour la réalisation du système se porte sur des caméras à optique fisheye. Ce choix est conforté par le fait que les optiques fisheye sont plus faciles à obtenir et moins coûteuses que les miroirs catadioptriques.

4.2.4 Modélisation et prototypage du système, choix finaux

Dans la solution retenue (deux caméras à optique fisheye placées l'une au dessus de l'autre), l'alignement des axes optiques des deux caméras est une contrainte forte, qui doit être assurée dès la construction du système et maintenue durant tout son fonctionnement.

Afin de répondre à ces contraintes d'alignement et de garantir un grand champ de vision, deux solutions technologiques sont envisageables :

⇒ fixer les caméras le long d'une **barre métallique** qui sera hissée d'un seul bloc à la hauteur désirée (voir figure 4.14) : ceci permet de fixer simplement les caméras et de les aligner avant de les mettre en place. Néanmoins, il est indispensable d'utiliser des caissons

FIGURE 4.14 – *Prototype avec barre métallique et images associées*

étanches pour protéger les caméras de l'humidité et autres intempéries. En outre, cette disposition génère, du fait de la section de la barre nécessaire à sa rigidité, le masquage d'une partie de la zone observée.

⇒ placer les caméras à l'intérieur d'un **tube transparent** : cette approche a l'avantage de garantir une bonne étanchéité du système. Les caméras sont tenues par le tube, ce qui réduit considérablement les masquages en ne laissant que les câbles reliant une caméra à l'autre. Néanmoins, toute l'image est vue à travers le tube, ce qui peut poser deux types de problèmes :
- Le tube doit être géométriquement quasi-parfait pour éviter les déformations de la scène observée ;
- Des phénomènes de réfraction peuvent se produire étant donné que les rayons lumineux parvenant aux capteurs ne traversent pas l'épaisseur du tube avec le même angle d'incidence. De plus, cet angle d'incidence fait également varier l'épaisseur de tube traversée.
- Des reflets de l'image sur la zone opposée du tube sont possibles.

Les deux prototypes sont modélisés (voir figure 4.15) pour les intégrer dans les simulations, mais ces simulations ne permettent pas d'analyser le risque de reflets. Pour cela, des essais réels sont nécessaires. Après essais sur le prototype avec un tube (voir figure 4.16), des reflets sont bien présents, créant des objets fantômes qui s'avèrent être fortement pénalisants pour

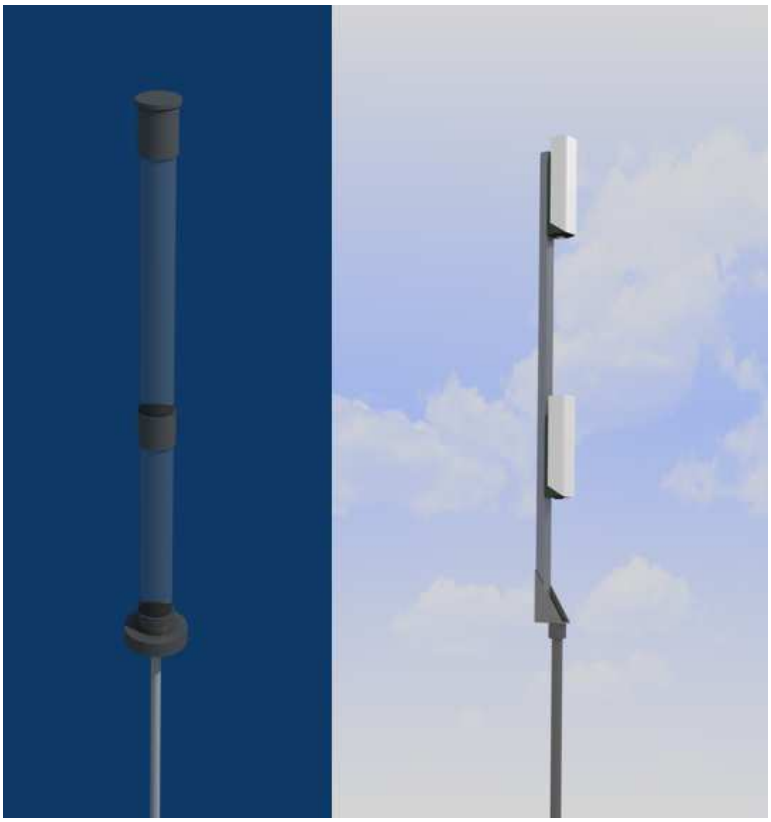


FIGURE 4.15 – *Modélisation des prototypes*

la qualité du traitement. De plus, la variation d'angle d'incidence, combinée aux inévitables imperfections du tube, déforme l'image, principalement dans sa partie centrale. Ce phénomène pourrait peut-être être amélioré par l'utilisation d'un meilleur matériau pour la conception du tube, mais cela n'aurait pas d'effets sur les reflets. Cette piste a donc été écartée pour la suite de l'étude.

4.2.5 Influence de la disposition du système sur les occultations

On a décrit précédemment le phénomène lié, dans la stéréovision, à la différence de points de vue des deux caméras. On a présenté une solution (voir paragraphe 2.2.5.2) permettant de détecter les zones erronées par ce principe. Il est donc important d'étudier comment les occultations apparaissent dans la configuration du système finalement retenu.

Compte tenu de la disposition l'une au dessus de l'autre des caméras, quasiment tous les points vus par la caméra basse seront également vus par la caméra haute (à l'exception de zones minimes sous le châssis d'un véhicule, comme le montre la figure 4.17, et en limite de scène). Il est donc préférable de choisir la caméra basse en référence, et de chercher les points correspondant sur la caméra haute, car le nombre d'occultations est alors diminué.

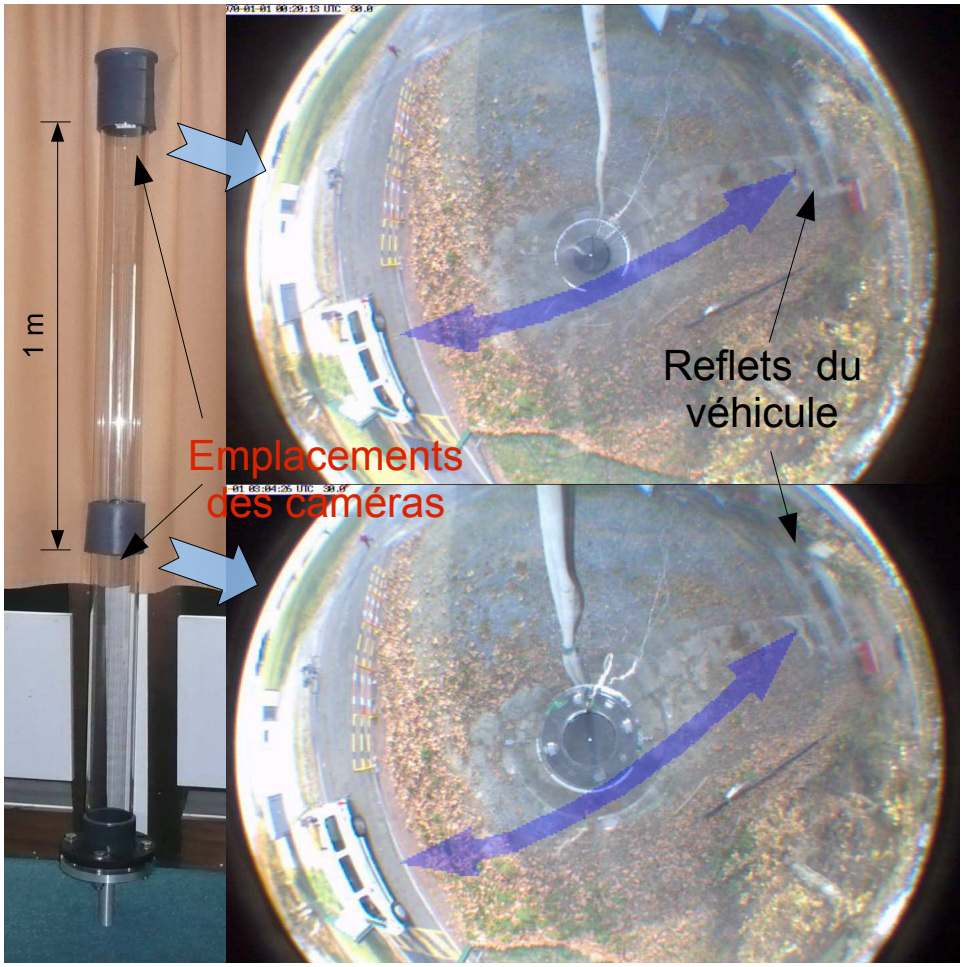


FIGURE 4.16 – *Prototype avec tube et images associées*

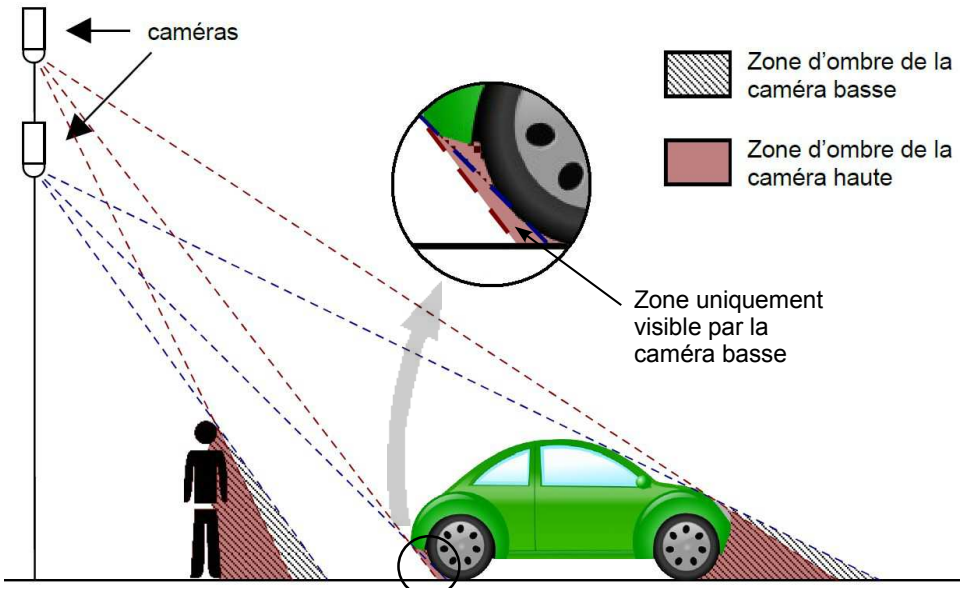


FIGURE 4.17 – *Zones non visibles par les caméras*

4.3 CALIBRAGE DU SYSTÈME

Pour que le système puisse fonctionner et que l'on puisse traiter efficacement les vidéos provenant des deux caméras, il faut effectuer une étape préalable consistant à effectuer une calibration, et ceci pour pouvoir :

⇒ Prendre en compte les paramètres intrinsèques de chacun des ensembles caméra + optique

⇒ Corriger les erreurs d'alignement résiduelles entre les deux caméras

4.3.1 Approche globale

Des approches génériques existent pour calibrer un système stéréo quel que soit le type de capteurs, comme le logiciel Hyscas [36] [88], illustré par la figure 4.18. Son principe consiste à prendre des images d'une mire visible simultanément par les deux caméras. Une paire d'images (provenant de la caméra haute et la caméra basse) est capturée pour différents emplacements de la mire couvrant différentes zones du champ de vision des deux caméras. Cette approche permet d'estimer en même temps les paramètres intrinsèques et extrinsèques en utilisant le modèle unifié pour modéliser l'optique fisheye. Les mires utilisées sont des mires à points, le logiciel cherchera ainsi le centre de chaque point.

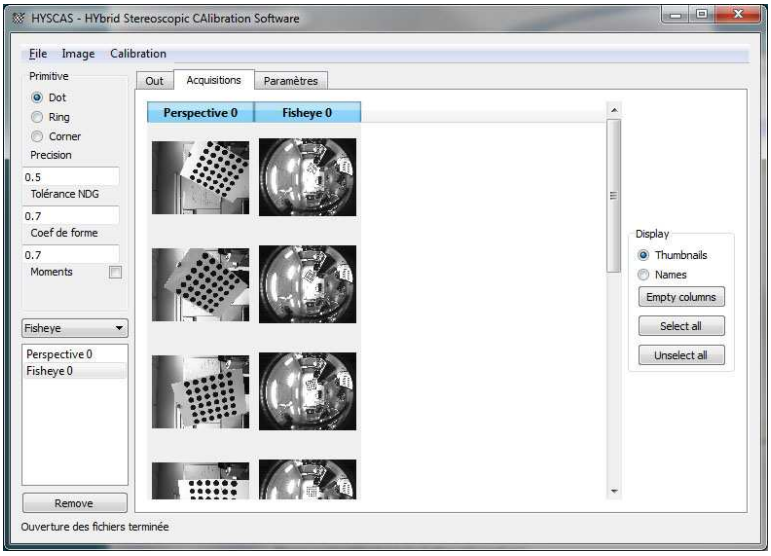


FIGURE 4.18 – *Logiciel de calibration Hyscas*

Durant les expérimentations, cette approche n'a pas donné les résultats souhaités. La disposition particulière de notre système ne facilite pas cette calibration. Il s'est avéré difficile d'obtenir des bonnes images des mires avec les deux caméras, une mire visible par la caméra basse étant forcément plus éloignée sur l'autre caméra (voir figure 4.19). Une solution, consisterait à augmenter la taille des mires. Une taille A0 pourrait répondre au problème, mais la mire serait alors difficile à manipuler, et à garder plane. De plus, du fait qu'on est obligé de tenir la mire éloignée des caméras, des problèmes d'éblouissement liés aux reflets sur les parties blanches de la mire sont assez fréquents. Pour ces raisons, cette approche peut sembler

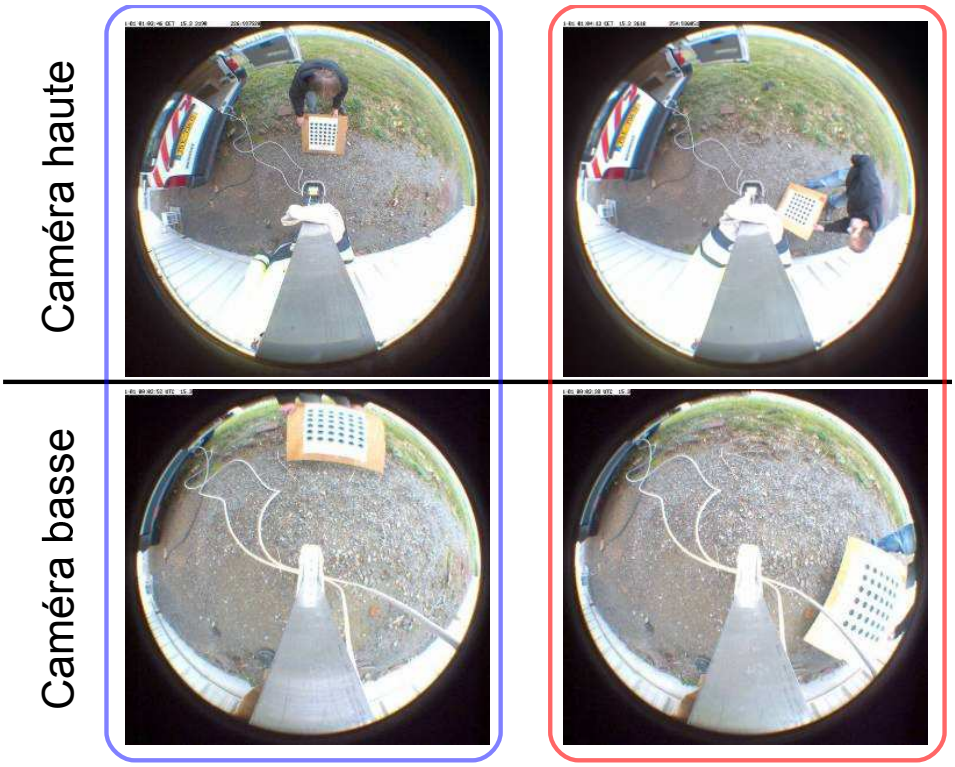


FIGURE 4.19 – *Problème de placement des mires pour la calibration pour Hyscas*

réalisable en intérieur mais très délicate à mettre en œuvre sur site extérieur.

4.3.2 Approche proposée

Partant de ce constat, une autre approche a été appliquée, consistant à calibrer individuellement chaque ensemble [caméra + optique] pour en obtenir les paramètres intrinsèques. Ensuite (une fois les caméras assemblées sur le support), on relève plusieurs points visibles par chacune des deux caméras pour estimer les erreurs d'alignement à corriger.

La *calibration intrinsèque* est propre à chaque capteur (chaque ensemble caméra + optique fisheye) et permet de mesurer les paramètres permettant de passer du monde réel 3D (la scène filmée) au monde 2D (l'image de cette scène sur la plaque de pixels). Elle règle notamment les problèmes de distorsion due à l'optique.

La *calibration extrinsèque* est propre au système de vision stéréoscopique global (l'association des deux capteurs) et permet de mesurer les paramètres permettant de passer du monde 3D à deux images 2D, cohérentes et donc comparables. Elle règle les problèmes de positionnement relatif, et notamment les problèmes résiduels d'alignement, des deux capteurs.

4.3.2.1 Outil pour la calibration intrinsèque

Dans un premier temps, on effectue le calibrage intrinsèque de chaque caméra. Pour cela, une méthode utilisée pour la vision monoculaire est suffisante. Le capteur ayant une optique fisheye, deux modélisations sont possibles comme décrit en 2.1.4. Deux boites à outils sont également disponibles pour la calibration. Nous avons choisi d'utiliser la méthode polynomiale avec la boite à outils de Scarramuzza [33], parce qu'elle était déjà maitrisée en interne.

On acquière les images d'une mire prise sous différents angles couvrant l'ensemble du champ de vision de l'optique. Dans la pratique, une dizaine d'images est suffisant. La taille et le nombre de cases inscrites sur la mire ne sont pas importants en théorie, mais la pratique permet de constater la plus grande simplicité de manipulation d'une petite mire d'une dizaine de centimètres tenant dans la main et placée très proche du capteur, pour différentes raisons :

- ⇒ Une petite mire est plus facilement plane, et risque moins de se courber sous l'influence du vent par exemple
- ⇒ Cette mire étant très proche du capteur, les risques de reflets sur les zones blanches sont réduits
- ⇒ La manipulation de la mire est plus simple et plus précise.

La boite à outils permet d'estimer les paramètres de la caméra, connaissant la position des coins sur l'image, l'espacement entre eux, et sachant que tous ces points se situent sur le même plan. Après calcul, on obtient :

- ⇒ La position du centre optique dans l'image pixelisée.
- ⇒ Les coefficient du polynôme de distorsions radiales $\theta(r)$ ainsi qu'une visualisation de cette fonction correspondant à :

$$\theta\left(r\right) = \arctan\left(\frac{p_0 + p_1 r + p_2 r^2 + \cdots + p_n r^n}{r}\right) \tag{4.2}$$

  Dans la pratique, un polynôme d'ordre 4 est suffisant pour estimer avec précision les distorsions radiales.
- ⇒ Un fichier en sortie contenant tous ces paramètres et pouvant être réutilisé dans un programme

Connaître les paramètres intrinsèques de chaque caméra n'est pas suffisant pour pouvoir effectuer le traitement. Il faut maintenant effectuer une seconde étape produisant les paramètres extrinsèques du système constitué par les deux caméras.

4.3.2.2 Calibration extrinsèque

En théorie, pour pouvoir comparer les images produites par les deux caméras, C_1 et C_2, du système de vision stéréoscopique, il faut que ces deux caméras soient strictement alignées. Cela signifie que les axes optiques de ces caméras soient confondus avec la ligne (en l'occurrence, verticale) passant par leurs centres optiques, comme illustré sur la figure 4.20a.

Dans la réalité, les caméras ne sont pas parfaitement alignées. Cette réalité est illustrée par la figure 4.20b.

La calibration extrinsèque vise donc à déterminer les 6 paramètres de rotation qui amèneraient les deux caméras dans un alignement parfait. En

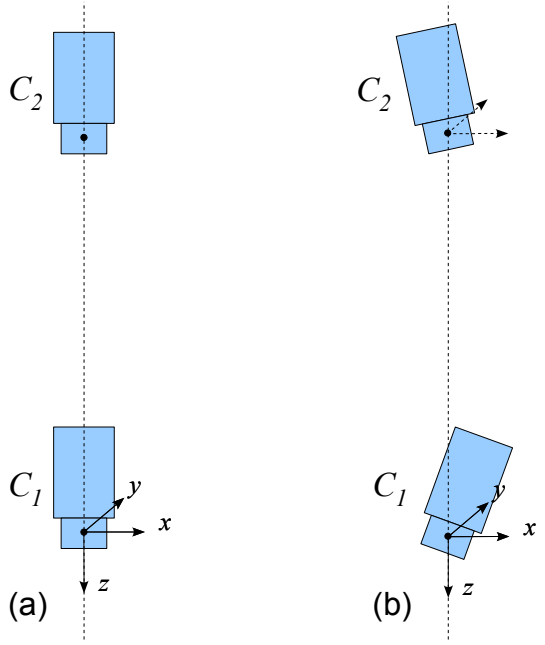


FIGURE 4.20 – *Système stéréoscopique parfaitement aligné (a) et imparfaitement aligné (b)*

réalité, la rotation autour de l'axe des centres (autour de z) ne concerne que l'une des caméras, l'autre fournissant la référence. Ce sont donc 5 paramètres, ϕ_{x1}, ϕ_{y1}, ϕ_{x2}, ϕ_{y2} et ϕ_{z2} qui sont à déterminer, avec :

$\Rightarrow$ ϕ_{x1} : Angle de rotation de la caméra C_1 autour de l'axe des x
$\Rightarrow$ ϕ_{y1} : Angle de rotation de la caméra C_1 autour de l'axe des y
$\Rightarrow$ ϕ_{x2} : Angle de rotation de la caméra C_2 autour de l'axe des x
$\Rightarrow$ ϕ_{y2} : Angle de rotation de la caméra C_2 autour de l'axe des y
$\Rightarrow$ ϕ_{z2} : Angle de rotation de la caméra C_2 autour de l'axe des z

Ces paramètres pourront alors être utilisés pour modifier les images reçues par le système stéréoscopique en vue d'obtenir celles qui auraient été reçues par le même système parfaitement aligné.

Pour pouvoir déterminer ces cinq paramètres, trois approches sont testées.

La *première approche* consiste à matérialiser deux points, P_1 et P_2, sur l'axe du système ; pour cela on peut utiliser le support vertical des deux caméras, et on positionne sous chaque caméra un point matériel, comme le montre la figure 4.21.

Pour chaque caméra d'index j, connaissant ses paramètres intrinsèques, on peut déterminer les deux angles de rotations ϕ_{xj} et ϕ_{yj} qui ramèneront l'image du point (P_1 pour C_1, P_2 pour C_2) au centre du capteur de la caméra. La rotation ϕ_{z2} est simplement obtenue en mettant en correspondance au moins un point vu par les deux caméras dans la scène, en dehors de l'axe.

Cette approche présente des inconvénients puisque le point de l'axe visible par C_2 ne peut pas être le même que celui visible par C_1, ce dernier étant caché à C_2 par C_1. En outre, une petite imprécision sur la position

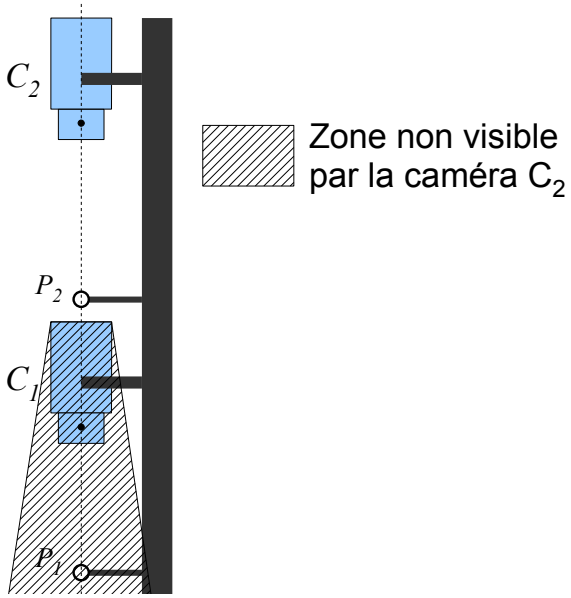


FIGURE 4.21 – *Dispositif de calibrage extrinsèque*

des points P_1 et P_2, situés sur l'axe, conduit à une imprécision plus forte sur des points plus éloignés de l'axe.

Pour la *seconde approche*, on commence par identifier des points facilement reconnaissables dans la scène, et visibles par les deux caméras. Pour l'exemple, considérons l'un des points, P_i, ainsi identifié. En utilisant les paramètres intrinsèques de la caméra C_1, on peut projeter l'image de ce point sur une sphère unitaire centrée sur le centre optique de C_1. On obtient, dans l'espace 3D, un point théorique ${P_{i1}}'$. On fait de même à partir de l'image de P_i dans la caméra C_2, et on obtient ${P_{i2}}'$. Si les caméras sont bien alignées, l'axe des centres des caméras et la droite $[{P_{i1}}'; {P_{i2}}']$ doivent être coplanaires. Le point P_i doit d'ailleurs être sur le même plan. En général, ce n'est pas le cas.

On détermine les angles ϕ_{x1}, ϕ_{y1}, ϕ_{x2}, ϕ_{y2} et ϕ_{z2} des rotations à appliquer aux caméras en écrivant que l'image ${P_{ij}}''$ du point P_i sur la sphère unitaire associée à la caméra C_j, une fois les corrections d'alignement réalisées, respectera au mieux cette coplanarité. Cela se fait en minimisant la fonctionnelle :

$$F = \sum_i \left| \arctan\left(\frac{{Y_{i2}}''}{{X_{i2}}''}\right) - \arctan\left(\frac{{Y_{i1}}''}{{X_{i1}}''}\right) \right| \tag{4.3}$$

Dans cette fonctionnelle, on a :

$$\begin{vmatrix} {X_{i1}}'' \\ {Y_{i1}}'' \\ {Z_{i1}}'' \end{vmatrix} = R_y(\phi_{y1}) \times R_x(\phi_{x1}) \times \begin{vmatrix} {X_{i1}}' \\ {Y_{i1}}' \\ {Z_{i1}}' \end{vmatrix} \tag{4.4}$$

et

$$\begin{vmatrix} {X_{i2}}'' \\ {Y_{i2}}'' \\ {Z_{i2}}'' \end{vmatrix} = R_z(\phi_{z2}) \times R_y(\phi_{y2}) \times R_x(\phi_{x2}) \times \begin{vmatrix} {X_{i2}}' \\ {Y_{i2}}' \\ {Z_{i2}}' \end{vmatrix} \tag{4.5}$$

équations dans lesquelles $\begin{vmatrix} {X_{i1}}' \\ {Y_{i1}}' \\ {Z_{i1}}' \end{vmatrix}$ et $\begin{vmatrix} {X_{i2}}' \\ {Y_{i2}}' \\ {Z_{i2}}' \end{vmatrix}$ sont respectivement les coordonnées des points ${P_{i1}}'$ et ${P_{i2}}'$ dans les repères locaux liés aux sphères unitaires, l'axe des z étant l'axe du système, l'axe des x de C_1 étant coplanaire avec l'axe des x de C_2, et de même pour les axes des y. Les $R_p(\phi_{p_i})$ sont les matrices de rotation d'angles ϕ_{p_i} autour de l'axe p (en l'occurrence : x, y ou z) de la caméra C_j ($j = 1$ ou 2).

Dans la mise en œuvre de cette seconde approche, il convient toutefois de prendre certaines précautions. Ainsi, les points P_i ne doivent pas être tous choisis dans un même plan de la scène, orthogonal à l'axe du système, sinon la convergence est difficile, et *in fine* on risque d'obtenir un sol incliné (voir figure 4.22). Il faut donc choisir des points régulièrement distribués autour de l'axe du système et à différents niveaux le long de cet axe.

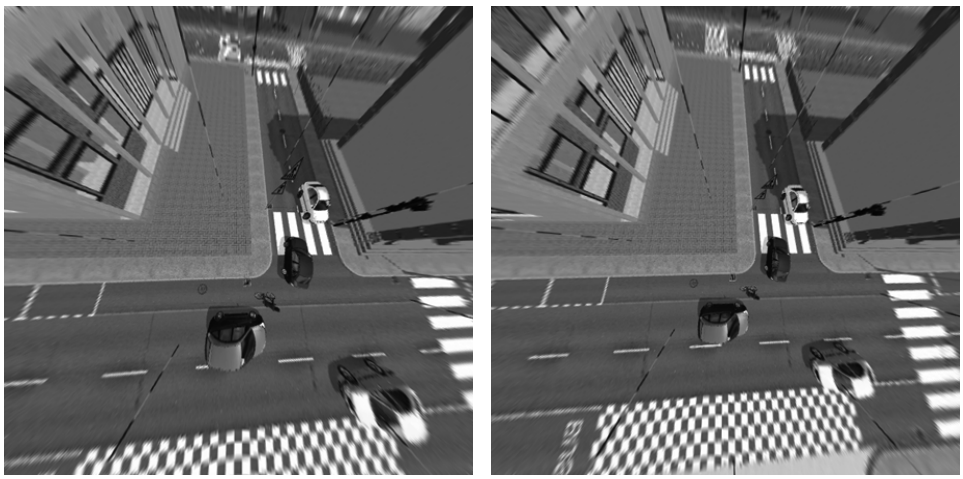


FIGURE 4.22 – *Images avec erreur de rectification extrinsèque, caméra basse (à gauche) et haute (à droite) : le sol se retrouve incliné*

La *troisième approche* permet de tirer parti des avantages des deux précédentes. Elle est réalisée en deux phases successives :

⇒ La première consiste à relever un point sur l'axe du système pour la caméra basse C_1. On détermine les angles de correction à appliquer à C_1, ϕ_{1x} et ϕ_{1y}, lors de cette phase.

⇒ La seconde consiste à calculer les rotations à appliquer sur l'autre caméra en prenant des points visibles sur les deux caméras et en cherchant à minimiser une fonctionnelle F par rapport aux trois angles ϕ_{2x}, ϕ_{2y} et ϕ_{2z}, puisque les deux autres angles sont connus. Cette minimisation converge plus facilement.

In fine, les deux caméras sont bien alignées sur l'axe du système et entre elles.

4.4 Application du système stéréoscopique fisheye avec des caméras alignées

4.4.1 Introduction

La nature des capteurs (optique fisheye) et leur position est assez particulière. On ne peut pas effectuer un traitement comme ceux décrits précédemment (voir paragraphe 2.2.4) avec les images prises par une optique fisheye. En effet, les mises en correspondance ne peuvent être effectuées sur une ligne horizontale, ceci étant lié au fait que :

⇒ les caméras placées l'une au dessus de l'autre, et alignées sur le même axe, ne permettent pas cette mise en correspondance

⇒ l'image provenant des capteurs fisheye est très distordue ; ce dernier point ne serait pas critique si les caméras étaient placées l'une à coté de l'autre ; une étape pour distordre les images pourrait suffire à permettre une recherche selon une ligne horizontale, comme c'est le cas dans différentes approches [85] [86].

Des adaptations sont donc nécessaires. Trois approches ont été testées et sont présentées ici :

⇒ **Approche dépliée** : les images sont transformées pour pouvoir utiliser les méthodes "habituelles" appliquées aux caméras situées côte à côte.

⇒ **Approche non dépliée** : l'étape de mise en correspondance ne s'effectue plus par ligne horizontale, mais par rayon.

⇒ **Approche multi-couches** : l'image I_H provenant de la caméra du haut est modifiée pour créer une suite d'images I_h, telles que, sur chacune de ces images, les points de la scène réelle se trouvant à la hauteur h soient caractérisés par une disparité nulle entre I_h et I_B (image de la caméra du bas).

4.4.2 Approche dépliée

L'appariement des pixels se fera non pas par une recherche le long de lignes horizontales, mais par une recherche le long de rayons. Pour appliquer les méthodes citées auparavant, il faut transformer l'image pour que les rayons de l'image d'origine deviennent des lignes horizontales sur l'image transformée comme le montre la figure 4.23.

L'idée est donc de transformer chaque image en effectuant un dépliage de celle-ci, de sorte que les rayons deviennent des lignes horizontales. Ce qui, comme le montre la figure 4.24, revient à étirer l'image obtenue sur la zone centrale.

Pour une image originale dont les coordonnées des pixels sont notées selon x et y, et dont le centre a pour coordonnées x_c et y_c, les coordonnées u et v des pixels de l'image dépliée sont alors :

$$u = \sqrt{(x - x_c)^2 + (y - y_c)^2} \times K_u \tag{4.6}$$

$$v = \arctan \frac{y - y_c}{x - x_c} \times K_v \tag{4.7}$$

Où K_u et K_v sont des constantes définissant la taille de l'image dépliée.

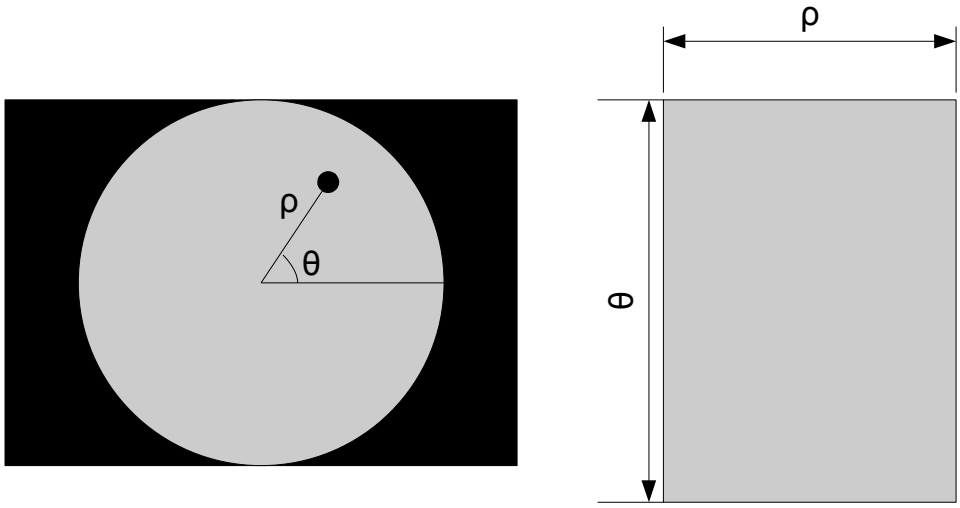


FIGURE 4.23 – *Principe du dépliage*

Cette solution à l'avantage de ramener le problème à un cas simple puisque le calcul de mise en correspondance peut se faire comme s'il s'agissait de deux caméras placées côte à côte. Néanmoins, la zone centrale de l'image est très étirée alors que les zones lointaines se retrouvent comprimées. La charge de calcul devient très importante pour une zone centrale étirée qui n'apporte en fait pas beaucoup d'informations supplémentaires. Une taille de fenêtre constante n'est pas idéalement adaptée étant données les zones de compressions et d'étirements.

Enfin, la relation entre disparité et distance n'est pas aussi évidente que dans le cas simple de caméras ayant des optiques classiques.

$$D = \frac{H_1 - H_2}{\frac{1}{\tan \alpha_1} - \frac{1}{\tan \alpha_2}} \tag{4.8}$$

Pour éviter ces différents problèmes, nous proposons ci-dessous deux approches se passant de cette étape de dépliage.

4.4.3 Approche non dépliée

Une autre approche envisageable est d'effectuer la mise en correspondance directement sur l'image brute prise par l'optique fisheye, sans effectuer de dépliage. Il n'y a alors pas de zones étirées comme cela peut être le cas avec le traitement déplié.

En revanche, contrairement au traitement standard, la ligne sur laquelle se fait la recherche de mise en correspondance (un rayon de l'image) ne coïncide pas avec une ligne de pixels. Une interpolation devient alors indispensable pour effectuer le traitement.

On souhaite obtenir la disparité selon un rayon de chaque pixel de l'image de la caméra basse, I_1, ceci pour associer une hauteur à chacun de ces pixels. Dans ce but, on calcule les coordonnées polaires (θ, ρ) de chaque pixel de départ.

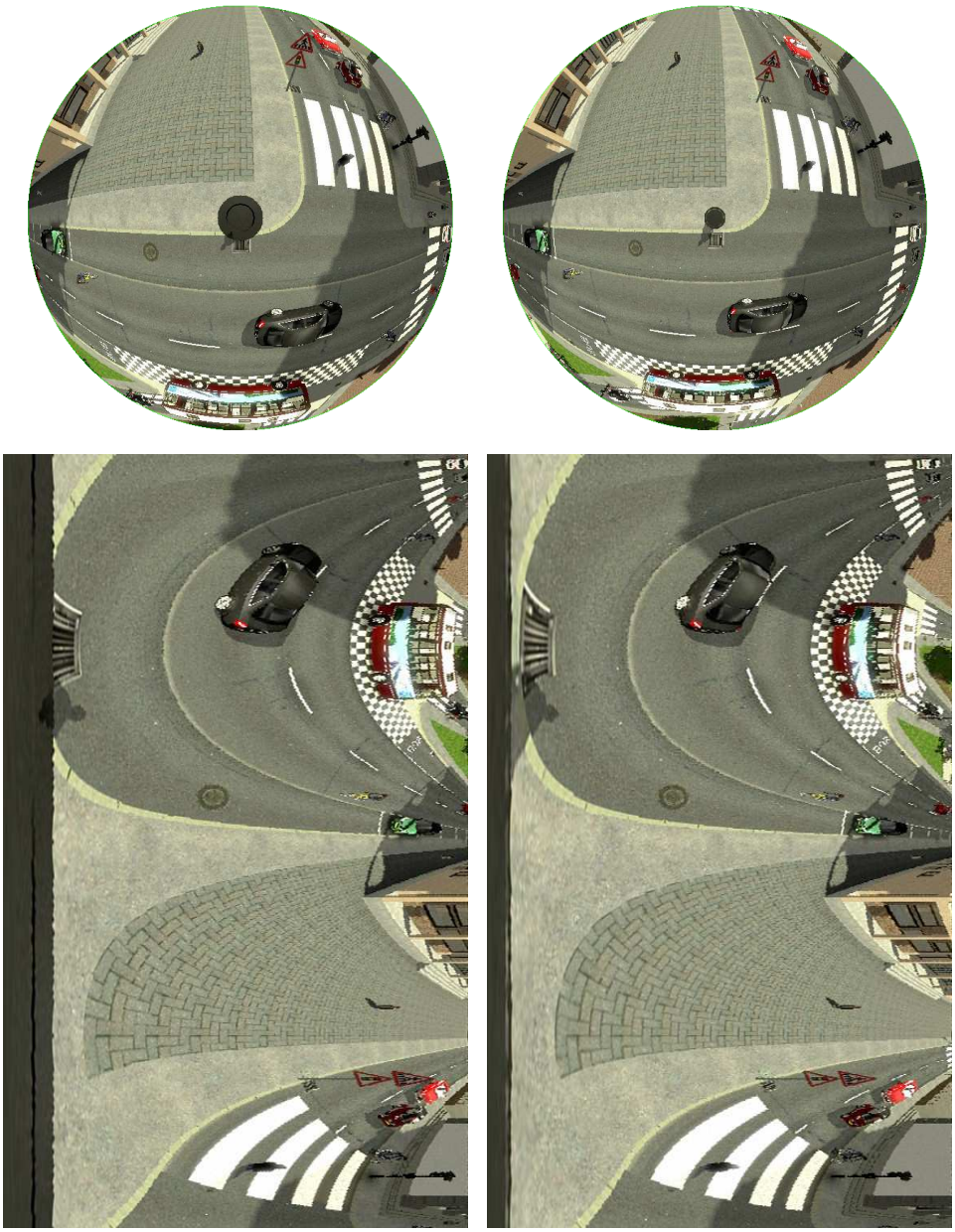


FIGURE 4.24 – *Images fisheye et images dépliées*

$$\theta = \arctan\left(\frac{y - y_c}{x - x_c}\right) \tag{4.9}$$

$$\rho = \sqrt{(y - y_c)^2 + (y - y_c)^2} \tag{4.10}$$

Ceci étant fait, on peut appliquer la procédure suivante, pour chaque pixel $P_{x,y,1}$ de l'image I_1 :

1. On associe à ce pixel une fenêtre f_1, centrée sur celui-ci et orientée selon le rayon qui le traverse. On peut définir de bien des façons la forme et la taille de cette fenêtre qui, le plus souvent, sera carrée ou angulaire (voir figure 4.25). Notons que la taille d_{pix} des pseudo-pixels de la fenêtre est identique à celle des pixels de l'image.

2. Sur l'image provenant de l'autre caméra, I_2, on définit de la même façon une fenêtre f_2 de même forme et de même taille que f_1, qui est centrée sur un pixel positionné sur le même rayon que $P_{x,y,1}$ mais à une distance $d = disp \times d_{pix}$ de celui-ci. En termes plus mathématiques :

$$\rho_2 = \rho_1 \tag{4.11}$$
$$\theta_2 = \theta_1 + disp \times d_{pix} \tag{4.12}$$

   $disp$ est un entier variant de $disp_{min}$ à $disp_{max}$, bornes dépendant de la scène à observer et de la configuration des caméras.
3. On calcule, par la méthode retenue (SAD, SSD, ..., CENSUS) une distance entre les fenêtres f_1 et f_2.
4. On associe au pixel $P_{x,y,1}$ la disparité $disp$ qui correspond à la plus petite distance entre f_1 et f_2.

En recommençant les étapes 1 à 4 pour tous les pixels de l'image initiale I_1, on obtient une carte de disparité associée à I_1, donc une carte d'altitude des points représentés par chacun des pixels.

La fenêtre carrée est la plus simple au vu des traitements standards, mais elle présente un risque d'aberrations lors de la recherche de mise en correspondance.

La fenêtre angulaire permet de supprimer ce risque, en prenant des pixels par interpolation sur le même rayon.

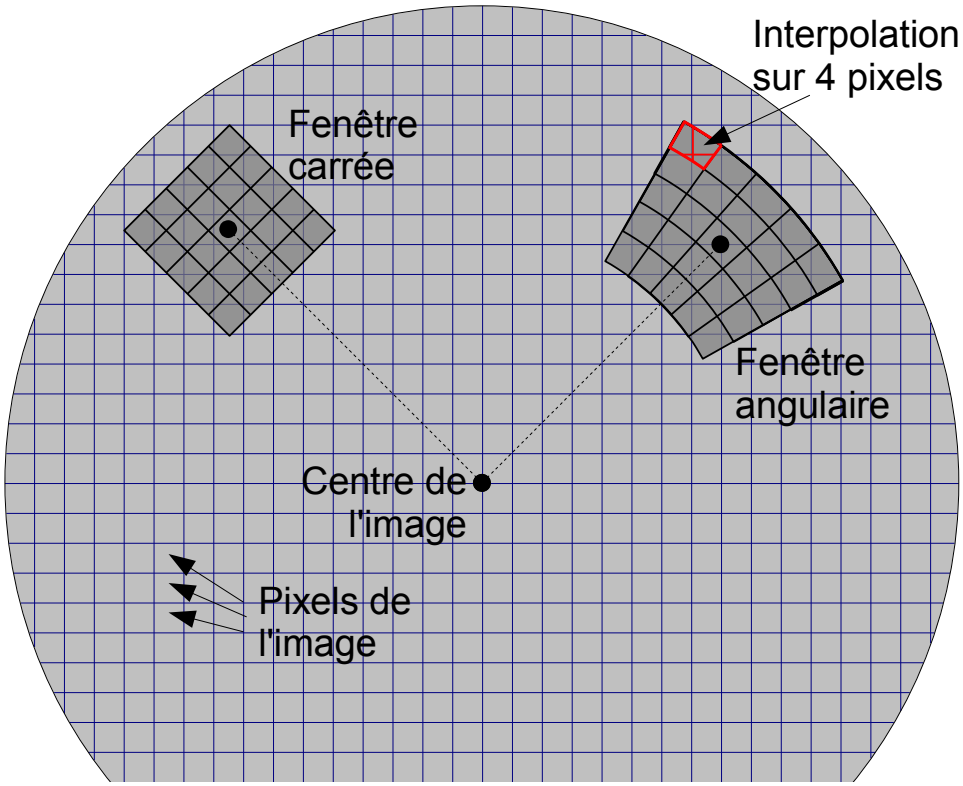


FIGURE 4.25 – *Fenêtre carrée et angulaire*

Dans cette approche, une fenêtre relevée ne correspondant pas avec les pixels de l'image, elle sera calculée par interpolation. Cela a pour conséquence d'augmenter considérablement le temps de calcul. En revanche,

l'avantage de cette approche est que l'on ne sur-traite pas la zone centrale comme avec l'approche précédente.

Il reste que ce type de traitement n'est pas du tout adapté pour la méthode CENSUS. En effet, chaque fenêtre devant être interpolée, il faut, dans le cas de cette méthode, lire une fenêtre plus grande que nécessaire, puis effectuer la transformée de CENSUS localement sur cette fenêtre, et enfin, effectuer le calcul de distance. Le traitement devient alors excessivement lourd.

4.4.4 Approche multi-couches

4.4.4.1 Principe de l'approche

Les deux approches présentées ci-dessus ont pour inconvénient de ne pas pouvoir limiter la plage de recherche de disparité à un intervalle correspondant à des objets situés entre 0 et 3 mètres, comme le sont généralement les véhicules à suivre. Pour que cela soit possible, il faudrait que cette plage de recherche soit adaptable à chaque zone de l'image. Une approche alternative a donc été imaginée qui consiste à découper la scène en « tranches » de hauteur.

Considérons une scène filmée par les deux caméras, basse C_1 et haute C_2. On définit dans cette scène une suite de N plans orthogonaux à l'axe du système (parallèles au sol), superposés les uns aux autres, et également espacés de δH entre une hauteur minimale H_{min} (généralement 0 m) et une hauteur maximale H_{max} (généralement 3 m). On aura donc l'écart entre deux plans successifs :

$$\delta H = \frac{H_{max} - H_{min}}{N - 1} \tag{4.13}$$

On appelle I_1 et I_2 les deux images provenant des caméras C_1 et C_2. A partir de ces deux images, on se propose de construire N images artificielles (qui ne correspondent pas à ce que verrait une caméra réelle, qu'elle que soit sa position), dites "couches", chacune associée à un plan. Ainsi, la couche I_i sera associée au plan situé à "l'altitude" H_i au-dessus du sol comme le montre la figure 4.26.

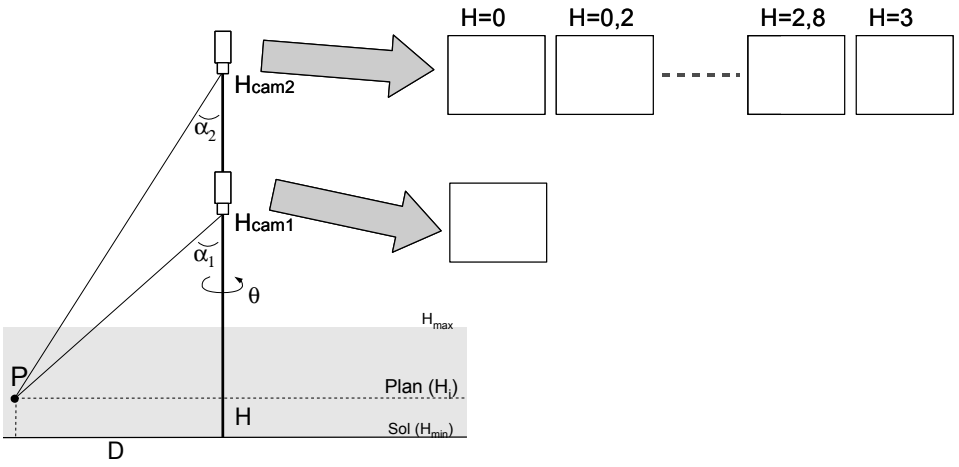


FIGURE 4.26 – *Principe de création des couches*

Soit P un point matériel qui est situé dans la scène à une distance D de l'axe du système et une hauteur H. On dit que ce point est à la hauteur du plan H_i le plus proche (on dira qu'il "est dans le plan"). Le point P est vu par la caméra C_1 sous l'angle α_1, par la caméra C_2 sous l'angle α_2, avec :

$$D = (H_2 - H) \times \tan \alpha_2 = (H_1 - H) \times \tan \alpha_1 \tag{4.14}$$

Dont on tire la relation :

$$\alpha_1 = \arctan \left(\frac{H_2 - H}{H_1 - H} \times \tan \alpha_2 \right) \tag{4.15}$$

Ce point matériel P correspond, après application des corrections de calibration, au pixel de coordonnées x_1 et y_1 dans l'image I_1 (respectivement x_2 et y_2 dans l'image I_2). En coordonnées polaires, ce pixel a la position définie par θ_1 et ρ_1 dans l'image I_1 (respectivement θ_2 et ρ_2 dans l'image I_2). Les angles θ_1 (= θ_2) définissent la position de P autour de l'axe du système stéréoscopique, alors que ρ_1 et ρ_2 sont directement liés à α_1 et α_2.

Par construction, tous les points du plan i, tels que P, doivent être vus dans la couche $I_i{}'$ avec un angle égal à α_1, celui avec lequel ils sont vus par la caméra basse. Le processus de construction de la couche $I_i{}'$ est donc le suivant (voir figure 4.27) :

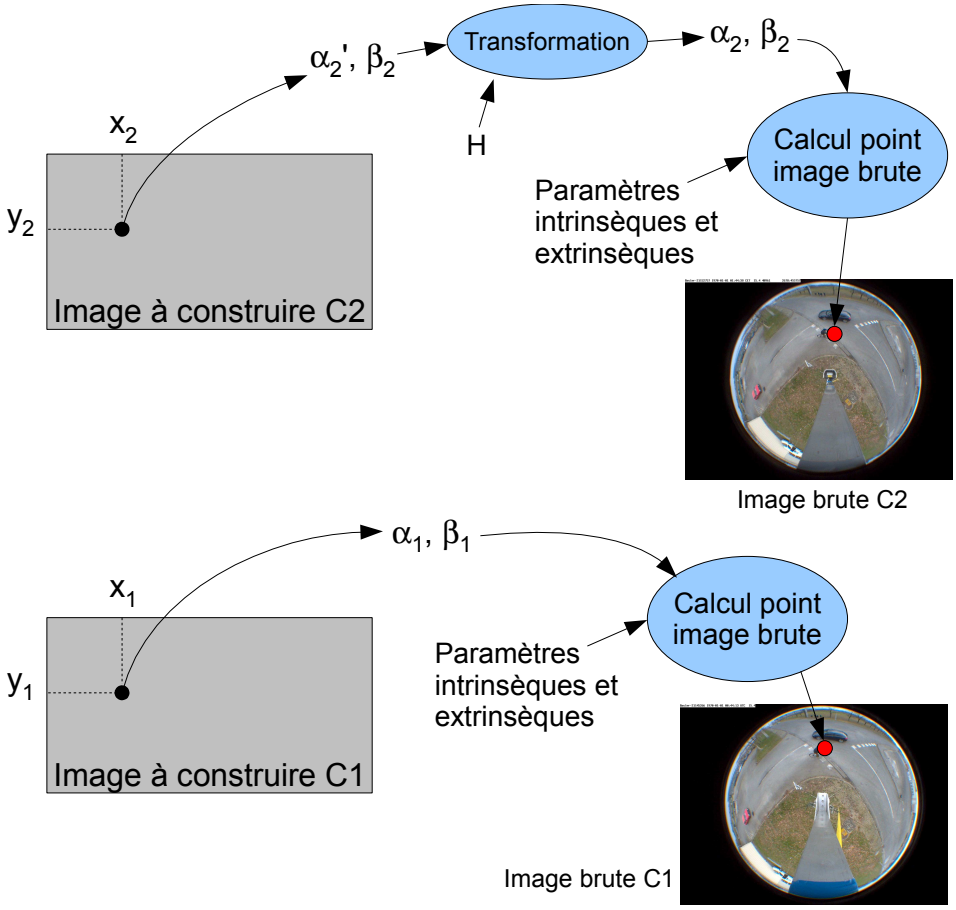


FIGURE 4.27 – *Principe de construction des images pour le traitement multi-couches*

1. Soit $V_{x,y,2}$ la valeur d'un pixel de l'image I_2 de la caméra haute. Ce pixel représente un point P de la scène réelle, dont la hauteur, à ce stade, n'est pas connue ; en revanche, les angles θ et α_2 correspondant à ce point (voir figure 4.26) sont accessibles grâce aux paramètres de calibration de la caméra C_2.
2. On fait alors l'hypothèse que ce point P est dans le plan i, donc à la hauteur H_i. Il est alors totalement identifié dans le repère physique de la scène à partir de θ, α_2 et H_i. On peut maintenant calculer grâce à (4.15) α_1, l'angle sous lequel le voit la caméra basse.
3. La valeur du pixel $V_{x,y,2}$ est alors attribuée, dans la couche $I_i{}'$, au pixel correspondant aux coordonnées polaire ρ_1 (calculé à partir de α_1 en utilisant les paramètres de calibration de C_1) et θ.
4. On répète ce processus pour tous les pixels de l'image I_2, et on crée ainsi complètement la couche $I_i{}'$.

Pour un couple d'images I_1 et I_2 donné (soit un instant donné de la séquence), il n'est plus nécessaire de partir d'une fenêtre f_1 dans I_1 et de rechercher la fenêtre la plus semblable dans l'ensemble des fenêtres possibles de I_2. Dans la présente approche, pour ladite fenêtre f_1, il suffit de chercher sur les N couches la fenêtre positionnée au même endroit dans les couches $I_i{}'$ qui est la plus proche de f_1 (voir figure 4.28). La hauteur H_i de la couche donne directement la hauteur de l'objet dans la scène.

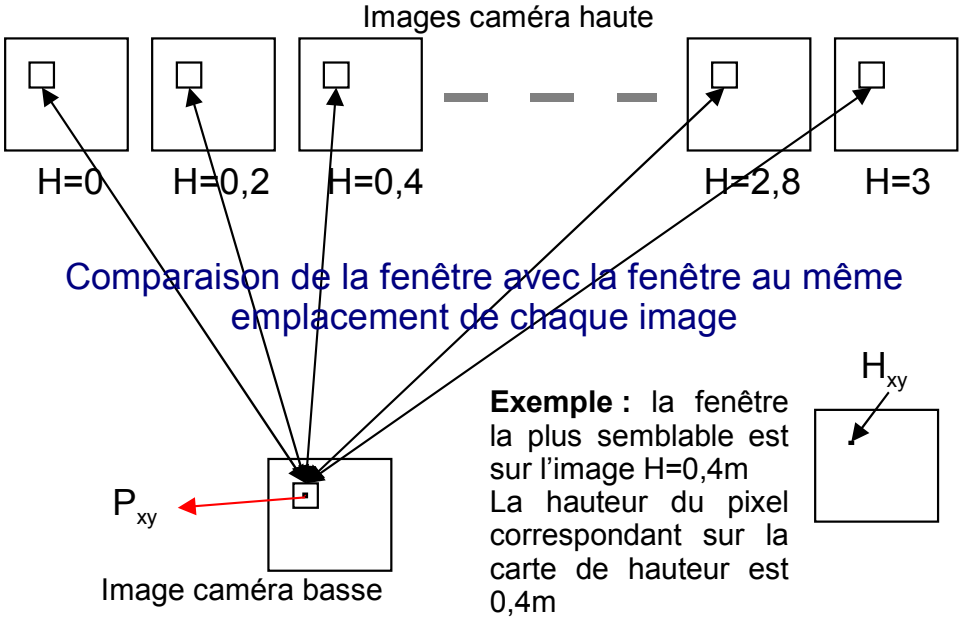


FIGURE 4.28 – *Recherche de mise en correspondance avec les couches*

Ainsi, les résultats obtenus sont directement sous forme de hauteur et non pas de disparité, qu'il aurait fallu reconvertir par la suite pour obtenir cette hauteur. Le calcul de distance entre deux fenêtres s'effectue alors avec les mesures décrites précédemment (SAD, ZNCC,...) dans la partie 2.2.4.

4.4.4.2 Transformation des images

Dans l'approche multi-couches, le format de l'image n'est pas déterminant sur la mise en correspondance. Il est important que toutes les images soient dans le même référentiel. On peut ainsi transformer celles-ci selon une infinité de modes. Nous en étudions trois en détails (figure 4.29 et table 4.1) :

⇒ mode **déplié** : les images sont transformées pour être dépliées (à ne pas confondre avec l'approche dépliée, voir paragraphe 4.4.2)

⇒ mode **fisheye** : les images sont rectifiées pour obtenir des images fisheye avec représentation linéaire

⇒ mode **distordu** : les images sont distordues pour obtenir une image équivalente à une projection sténopé (voir paragraphe 2.1.1.1). L'avantage réside dans le fait qu'un objet sur la route à la même résolution en pixels quelle que soit sa distance par rapport au système. Dans ce mode, la différence entre les couches n'est en fait qu'un niveau de zoom différent sur l'image.

FIGURE 4.29 – *Les trois modes de transformation différents pour un traitement multi-couches*

Mode	Avantages	Inconvénients
Déplié	⇒ Possibilité de ne déplier que la zone intéressante de l'image	⇒ Taille de fenêtre non homogène
Fisheye	⇒ Résolution proche de l'image d'origine ⇒ Partie centrale pas trop étirée	⇒ Taille de fenêtre non homogène ⇒ Partie de l'image non utilisée
Distordu	⇒ Taille de fenêtre homogène (correspond à une même dimension)	⇒ Zone lointaine très étirée

TABLE 4.1 – *Avantages et inconvénients des modes de transformation pour le traitement multi-couches*

4.4.4.3 Adaptations des méthodes de calcul de hauteur

Dans l'approche multi-couches, le traitement de disparité revient à comparer des zones identiques sur différentes images. Par conséquent aucun balayage sur l'image n'est nécessaire. De ce fait, des simplifications deviennent possibles pour appliquer les mesures de distance entre deux fenêtres listées en 2.2.4.

Application des méthodes SAD et SSD

Soit une image provenant de la caméra basse appelée I_1 et une image venant de la caméra haute appelée I_2. I_2 est transformée en N couches d'indice n où n varie de 1 à N. Chaque couche est donc appelée ${I_n}'$.

L'opération de calcul de la carte pour la méthode SAD peut s'écrire, après avoir définit une fenêtre de convolution W (voir ci-dessous) :
Pour chaque couche n allant de 1 à N,

1. On calcule une seconde liste d'images ${I_n}''$ telles que :

$$ {I_n}'' = \left\| {I_n}' - I_1 \right\|_1 * W \tag{4.16} $$

2. On crée une image, dont chaque pixel de coordonnées (x,y) prend pour valeur n tel que ${I_n}''_{x,y}$ est minimum

On peut ainsi attribuer à chaque pixel de l'image de la caméra du bas sa hauteur.

Le calcul avec la mesure SSD est très proche de la mesure SAD. Par conséquent, le traitement diffère simplement pour le calcul de ${I_n}''$ qui est alors obtenu par :

$$ {I_n}'' = \left\| {I_n}' - I_1 \right\|_2 * W \tag{4.17} $$

Application de la méthode CENSUS

Pour effectuer ce traitement avec la méthode CENSUS, il faut tout d'abord effectuer la transformée de CENSUS de chacune des couches. Ensuite, la distance de Hamming entre deux fenêtres f_1 et f_2 de taille N_f contenant des signatures binaires peut se décomposer en une somme de distances de Hamming prises élément par élément :

$$ D_{HAM}(f_1, f_2) = \sum_{i=1}^{N_f} D_{HAM}({f_1}^i, {f_2}^i) \tag{4.18} $$

La distance calculée peut donc l'être par une convolution d'une fenêtre d'une image contenant des distances de Hamming prises pixel par pixel.

Ainsi, on calcule T_1, la transformée de Census de I_1, puis pour chaque couche n allant de 1 à N :

1. T_n la transformée de Census de I'_n est calculée
2. On calcule ${T_n}'$ une image contenant la distance de Hamming entre T_n et T_1 calculée pour chaque pixel, soit

$$ {T_{nx,y}}' = D_{HAM}(T_{nx,y}, T_{1x,y}) \tag{4.19} $$

3. Grâce à la propriété de la distance de Hamming décrite dans l'équation 4.18, on effectue l'opération de fenêtrage par convolution soit :

$$T_n'' = T_n' * W \tag{4.20}$$

Enfin, on calcule la carte de sortie, dont chaque pixel de coordonnées (x,y) prend pour valeur n tel que $T_{nx,y}''$ est minimum.

Application des méthodes centrées (ZSAD, ZNCC, ...)

Dans le cas des méthodes dites centrées, chaque fenêtre est centrée par rapport à la valeur moyenne de ses pixels. Par conséquent, il devient plus complexe de réaliser cette transformation en traitant directement les images comme présenté précédemment. On peut, en effet, continuer à utiliser les méthodes telles quelles mais des optimisations deviennent impossibles. Une approche alternative consiste à modifier légèrement ces mesures de distances pour pouvoir les appliquer. Dans ce cas, on considère l'opération de centrage de la façon suivante : Soit une image A et une fenêtre W utilisée pour la convolution dont la somme des termes est égale à 1. L'opération de centrage réalisée localement sur chaque fenêtre f :

$$f_{centree} = \left(f - \overline{f}\right) \tag{4.21}$$

devient en traitant l'image A globalement :

$$A_{centree} = (A - A * W) \tag{4.22}$$

De ce fait, le calcul de distance (ZSAD modifiée) entre deux images A et B devient :

$$\begin{aligned} ZSAD2(A,B) &= \|A_{centree} - B_{centree}\|_1 && (4.23)\\ &= \|(A - A*W) - (B - B*W)\|_1 && (4.24)\\ &= \|(A-B) - (A-B)*W\|_1 && (4.25) \end{aligned}$$

Notation : Dans les exemples suivants, on considère, pour deux images A et B, le produit $A.B$ comme étant le produit terme à terme. Ainsi, $A^2 = A.A$ en terme à terme.

Dans ce cas, les valeurs de pixels sont centrées non pas par rapport à la fenêtre dans laquelle le pixel est présent, mais par rapport au voisinage de ce pixel.

De manière analogue pour la ZSSD

$$\begin{aligned} ZSSD2(A,B) &= (A_{centree} - B_{centree})^2 && (4.26)\\ &= ((A - A*W) - (B - B*W))^2 && (4.27)\\ &= ((A-B) - (A-B)*W)^2 && (4.28) \end{aligned}$$

Application des méthodes de corrélation

Les mesures de corrélation (NCC et ZNCC) ne pouvant s'appliquer directement au traitement sur les images prises globalement, nous réalisons également des adaptations de ces méthodes qu'on appelle ici NCC2 et ZNCC2, celles-ci étant définies de la façon suivante :

$$NCC2(A,B) = \frac{(A.B) * W}{\sqrt{(A.A) * W}.\sqrt{(B.B) * W}} \tag{4.29}$$

$$= \frac{(A.B) * W}{\sqrt{(A.A) * W.(B.B) * W}} \tag{4.30}$$

$$ZNCC2(A,B) = NCC2(A_{centree}, B_{centree}) \tag{4.31}$$

$$= \frac{(A_{centree}.B_{centree}) * W}{\sqrt{(A_{centree}.A_{centree}) * W}.\sqrt{(B_{centree}.B_{centree}) * W}} \tag{4.32}$$

$$= \frac{((A - A * W).(B - B * W)) * W}{\sqrt{C}} \tag{4.33}$$

Avec :

$$C = ((A - A * W).(A - A * W)) * W.((B - B * W).(B - B * W)) * W \tag{4.34}$$

Choix de la fenêtre de convolution W

Pour appliquer les méthodes précédentes, on utilise usuellement des fenêtres W carrées ayant une valeur uniforme. La fenêtre est idéalement d'une taille impaire pour pouvoir être centrée sur un pixel. Dans une fenêtre de taille $n \times n$, chaque élément aura pour valeur $\frac{1}{n^2}$.

Dans nos calculs de distance, la fenêtre étant appliquée par convolution, il devient possible, et de manière assez simple, d'appliquer d'autres fenêtres que la fenêtre carrée. Il est donc intéressant de voir l'incidence d'une fenêtre gaussienne sur le traitement :

$$gaussienne(x,y) = A\mathrm{e}^{-\frac{(x-x_0)^2+(y-y_0)^2}{2\sigma^2}} \tag{4.35}$$

avec σ l'écart type de la gaussienne et A l'amplitude de la gaussienne définie de sorte que :

$$\sum_{x,y} gaussienne(x,y) = 1 \tag{4.36}$$

Dans le cas de l'opération de centrage d'une image appelée A, $A_{centree}$ devient :

$$A_{centree} = (A - A * W) = A * W_C \tag{4.37}$$

avec

$$W_C = W_1 - W \tag{4.38}$$

où W_1, une matrice carrée, de taille identique à W (avec n impaire), est définie de la façon suivante :

$$W_1(x,y) = \begin{cases} 1 & \text{si } x = y = \frac{n-1}{2} \\ 0 & \text{sinon} \end{cases} \tag{4.39}$$

avec x et y allant de 0 à $n-1$.

4.4.4.4 Vitesse de calcul

L'approche multi-couches présente un atout majeur en termes de rapidité de calcul. Le traitement peut se faire directement sur les images entières sans avoir à prendre des fenêtres que l'on décale. La partie coûteuse ici est le calcul des différentes couches. Cette transformation étant toujours identique, on effectue une longue étape de pré-calcul des cartes de transformation des images qu'il n'est plus nécessaire de renouveler par la suite. Les seules opérations nécessaires pour le calcul sont donc :

⇒ des convolutions
⇒ des soustractions ou des multiplications, terme à terme, entre deux images
⇒ des transformations d'images à partir d'une carte pré-calculée

Ces opérations étant assez courantes en traitement d'images, elles sont généralement implémentées et optimisées dans les librairies de développement utilisées en traitement comme OpenCV [1]. De plus, l'absence de balayage sur l'image pour la recherche de mise en correspondance à l'avantage de permettre à l'approche multi-couches de se prêter plus facilement à un calcul parallélisé.

Grâce à ceci, dans le cas d'un calcul simple de SAD, le calcul implémenté en langage C fonctionne, sur des images de 600×600 pixels, à des vitesses de l'ordre de 6 images par secondes. Pour les autres méthodes, le calcul est plutôt de l'ordre de 30 secondes par image. Bien sûr, ce temps est fonction :

⇒ de la taille de l'image
⇒ du nombre de couches
⇒ de la taille de la fenêtre
⇒ de la mesure de distance utilisée

4.4.5 Comparaison des approches et des méthodes de calcul de hauteur

4.4.5.1 Principe de la comparaison

Afin de choisir la meilleure méthode à utiliser dans notre cas, il est important de trouver un critère permettant une mesure quantitative de la mise en correspondance stéréoscopique.

Pour pouvoir réaliser cette comparaison, il faut bien entendu, posséder :

⇒ La paire d'images à comparer
⇒ La vérité associée que l'on compare ensuite au résultat du traitement sur la paire d'images.

De nombreuses mesures existent pour mesurer l'efficacité d'un algorithme de stéréovision. Elles comparent une carte de disparité avec celle

1. http ://opencv.willowgarage.com/wiki/

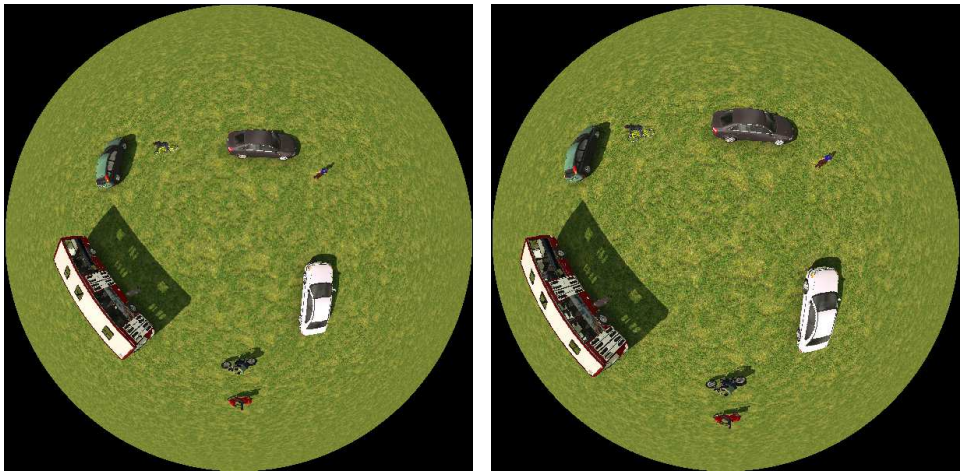


FIGURE 4.30 – *Images simulées pour comparaison*

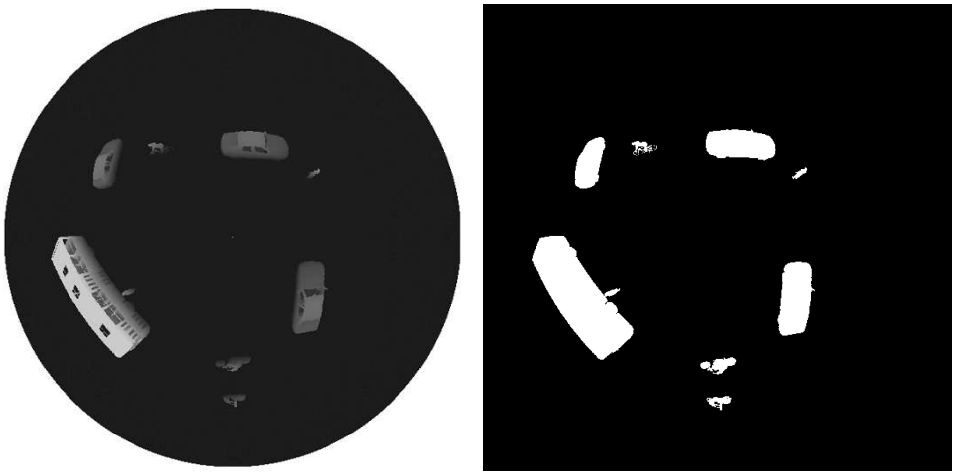


FIGURE 4.31 – *Vérité (à gauche) et masque de comparaison (à droite)*

à obtenir en théorie. La disposition des caméras que nous avons retenue (l'une au dessus de l'autre) apporte une relation disparité-distance assez particulière. L'importance n'est donc pas de savoir si la disparité obtenue est la bonne mais plutôt si la position (X,Y,Z) de l'objet est correcte.

Le logiciel de simulation SIVIC [78] construit des scènes en 3D. Il est donc capable de restituer la vérité terrain concernant le positionnement des objets sur cette scène (figure 4.31) avec une précision millimétrique, en même temps qu'une paire d'images captées par les caméras simulées (figure 4.30). Cette vérité est convertie sous forme de hauteur "vraie" au dessus du niveau de la route. L'objectif est donc de comparer si la hauteur au dessus du niveau de la route, calculée par l'approche stéréoscopique, représente bien la hauteur "vraie".

4.4.5.2 Critère d'évaluation

Pour mesurer l'efficacité d'un traitement, il est important de bien définir les zones à comparer. Pour cela des masques sont générés, qui permettent de sélectionner la zone d'intérêt sur laquelle effectuer la comparaison (voir figure 4.31). On peut donc comparer les résultats sur différentes zones en utilisant plusieurs masques. L'idée est ensuite de comparer les hauteurs obtenues dans la zone sélectionnée. Pour cela, on définit le critère suivant :

$$P(u) = \frac{N(V, R, M, u)}{Nbpix(M)} \tag{4.40}$$

avec V la vérité terrain, R le résultat à comparer, M le masque de comparaison, $N(V, R, M, u)$ le nombre de pixels dans la zone décrite par M tel que $|V - R| < u$, $Nbpix(M)$ le nombre total de pixels délimités par le masque, et u un niveau d'erreur.

Ce critère se présente sous la forme d'une courbe de répartition des erreurs (voir figure 4.32) permettant de mesurer le degré de précision du calcul en stéréovision. Ainsi, on peut connaître pour chaque écart avec la vérité terrain, le pourcentage de pixels dont l'erreur est inférieure à cette valeur. Une comparaison parfaite donnerait une droite horizontale d'ordonnée 1 (100% des pixels ont une erreur inférieure à u, quelque soit u).

Les tests effectués consistent donc, pour la paire d'image présentée en figure 4.30, à comparer les résultats obtenus par traitement avec la vérité terrain grâce à ce critère.

4.4.5.3 Résultats

Taille de la fenêtre

Si l'on compare les résultats obtenus selon la taille de la fenêtre utilisée lors, par exemple, de l'approche non dépliée, on remarque l'importance du choix de cette taille. En fonction de la scène, et de la résolution des objets que l'on souhaite observer, la fenêtre idéale doit être ni trop petite, pour capter suffisamment de détails afin de bien mettre en correspondance les zones de l'image, ni trop grande, pour ne pas ignorer les petits détails qui se retrouvent noyés dans la totalité des informations comprises dans une fenêtre. De plus, une fenêtre trop grande ralentit considérablement le

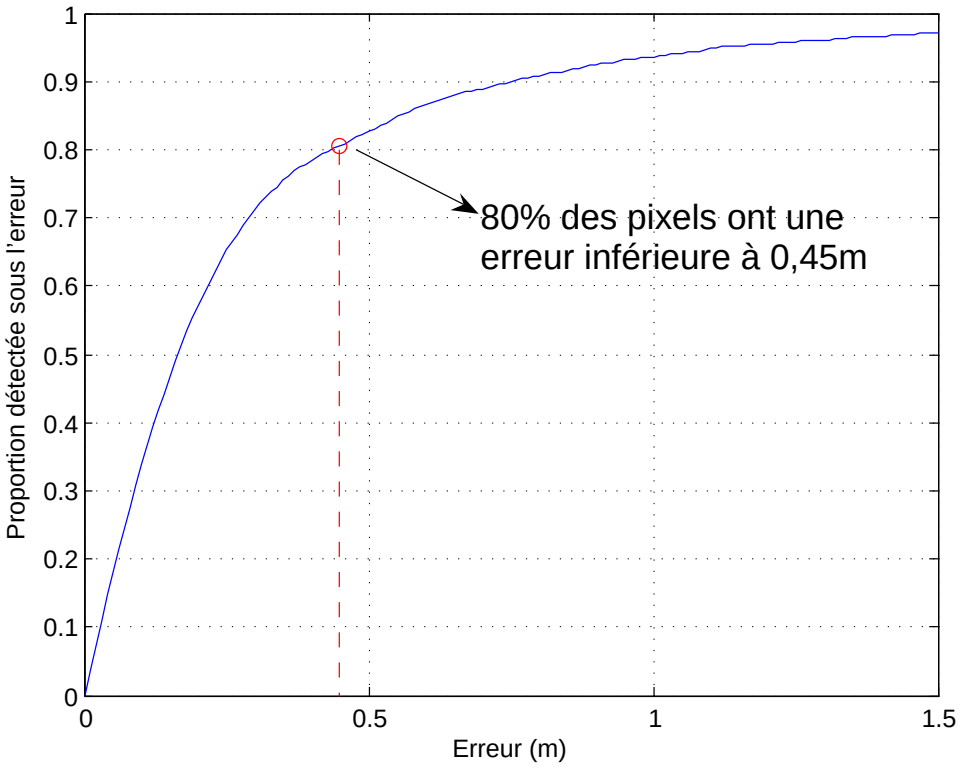


FIGURE 4.32 – *Courbe d'erreur*

temps de calcul. Pour toutes ces raisons, dans ce test (figure 4.33), la fenêtre de taille 11×11 semble être le meilleur compromis, mais ceci ne peut être généralisé.

Un résultat similaire est obtenu pour les autres approches.

Caméra de Référence

Dans la partie 4.2.5, nous avions conclu que, pour limiter les occultations, il serait plus intéressant de prendre la caméra basse en référence pour le traitement de mise en correspondance. Pour vérifier cette conclusion, le test est effectué en prenant successivement en référence la caméra basse (à 6m de hauteur) puis la caméra haute (à 7m de hauteur). Le résultat obtenu, visible dans la figure 4.34, confirme la conclusion en donnant de meilleurs résultats avec la caméra basse comme référence. Ceci se vérifie quelque soit l'approche de calcul utilisée. La différence n'est pas aussi flagrante que dans le test précédent, mais elle est bien réelle, en particulier pour un affinage des résultats à faible erreur.

Interpolations

Des interpolations sont utilisées lors des différentes étapes de transformation telles que :

⇒ Le dépliage de l'image
⇒ La rectification de l'image pour la calibration
⇒ Le calcul des couches
⇒ ...

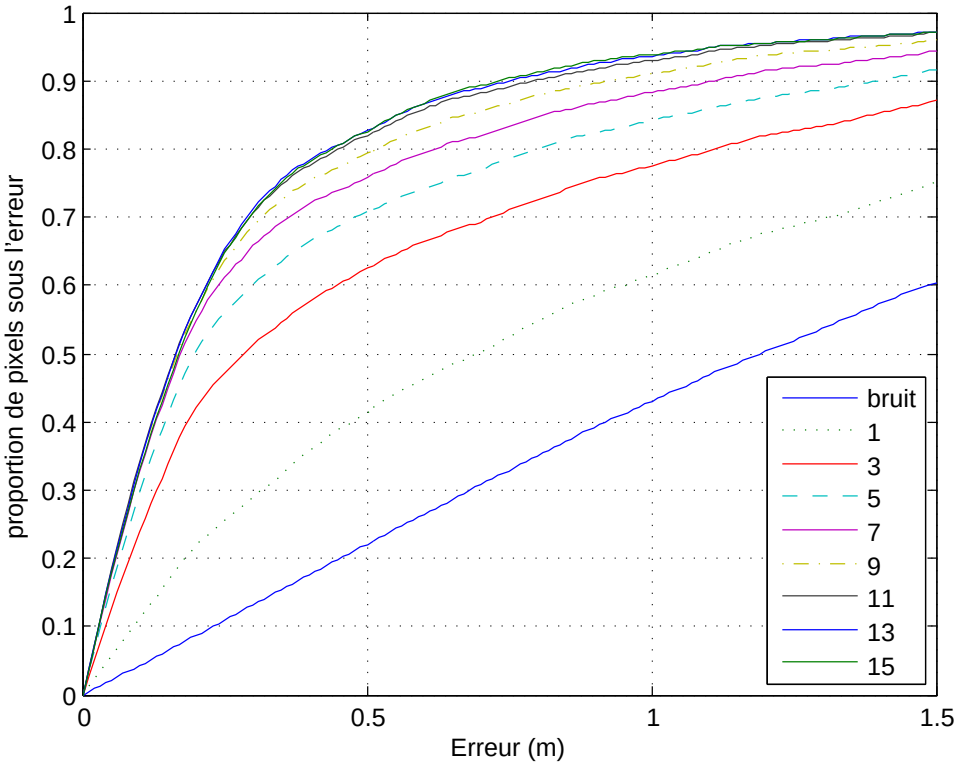


FIGURE 4.33 – *Comparaison des résultats suivant taille de la fenêtre*

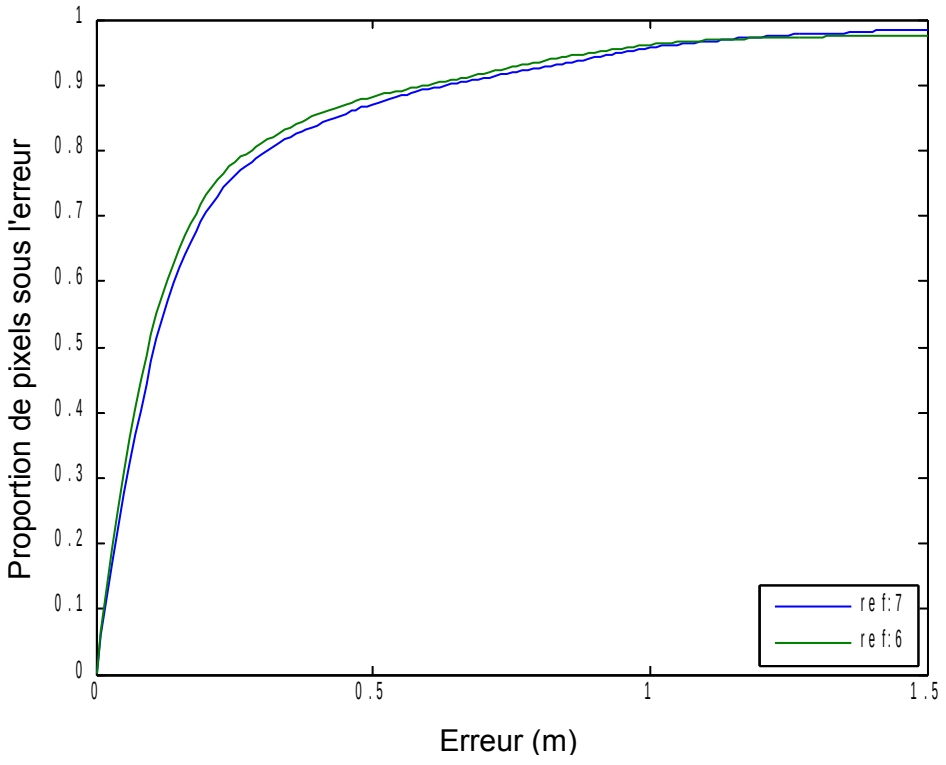


FIGURE 4.34 – *Comparaison des résultats selon le choix de la caméra de référence*

Cette opération introduit inévitablement une perte de précision sur l'image même si celle-ci peut sembler minime. Deux méthodes principales sont utilisées pour l'interpolation sur une image, à savoir :

⇒ L'interpolation bilinéaire

⇒ L'interpolation bicubique

L'interpolation bicubique donne généralement des images qui semblent à l'œil plus belles et moins dégradées. Mais il est important d'évaluer plus quantitativement l'influence du type d'interpolation sur le calcul. Cette évaluation, menée ci-dessous selon l'approche multi-couches, s'applique aussi aux autres approches.

Pour cela, on effectue deux calculs identiques où seul le type d'interpolation change. Les résultats ne permettent pas de différencier nettement l'une où l'autre interpolation, même si l'interpolation bilinéaire donne des résultats très légèrement supérieurs lors de tous les tests effectués (voir figure 4.35). De plus, l'interpolation bilinéaire semble préférable du fait que son temps de calcul est légèrement inférieur.

Quoi qu'il en soit, chaque opération d'interpolation dégradant l'image, il est important d'effectuer toutes les corrections comme la calibration et le calcul de couches ou le dépliage en une seule et même étape.

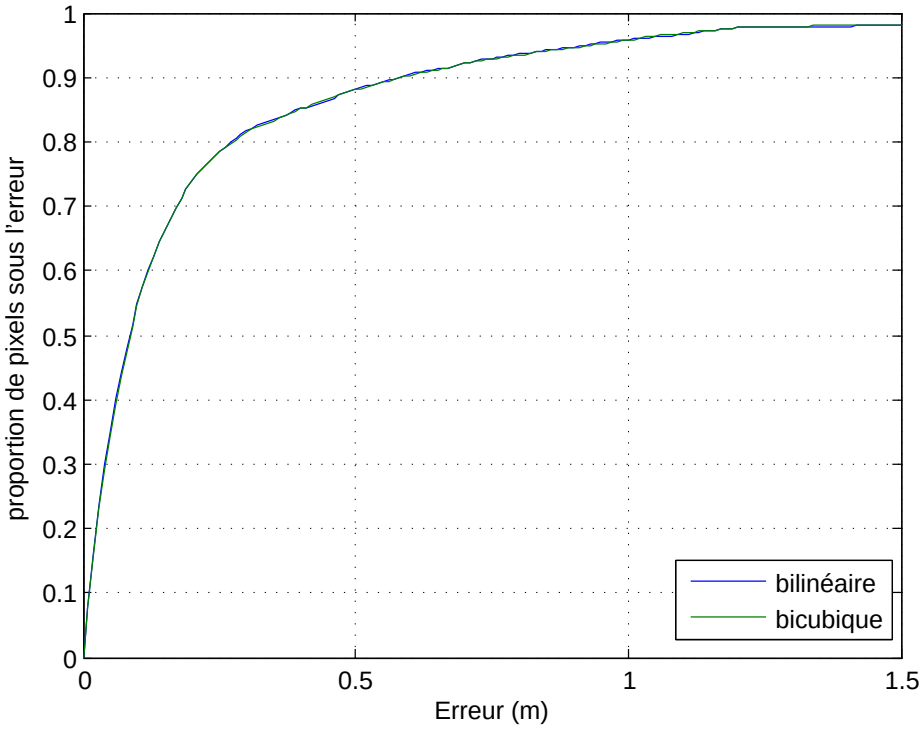


FIGURE 4.35 – *Comparaison des résultats suivant l'interpolation choisie*

Fenêtre angulaire ou carrée pour l'approche non dépliée

Dans l'approche de traitement non déplié, deux types de fenêtres ont été présentés. Nous effectuons alors les tests avec chacune d'entre elles (taille 11×11) pour vérifier si, comme supposé, la fenêtre angulaire donne de meilleurs résultats.

Comme le montre la figure 4.36, la fenêtre angulaire est plus efficace. Il n'y a pas de véhicules au centre de l'image mais, si cela avait été le cas, les différences entre les deux types de fenêtres auraient été beaucoup plus marquées.

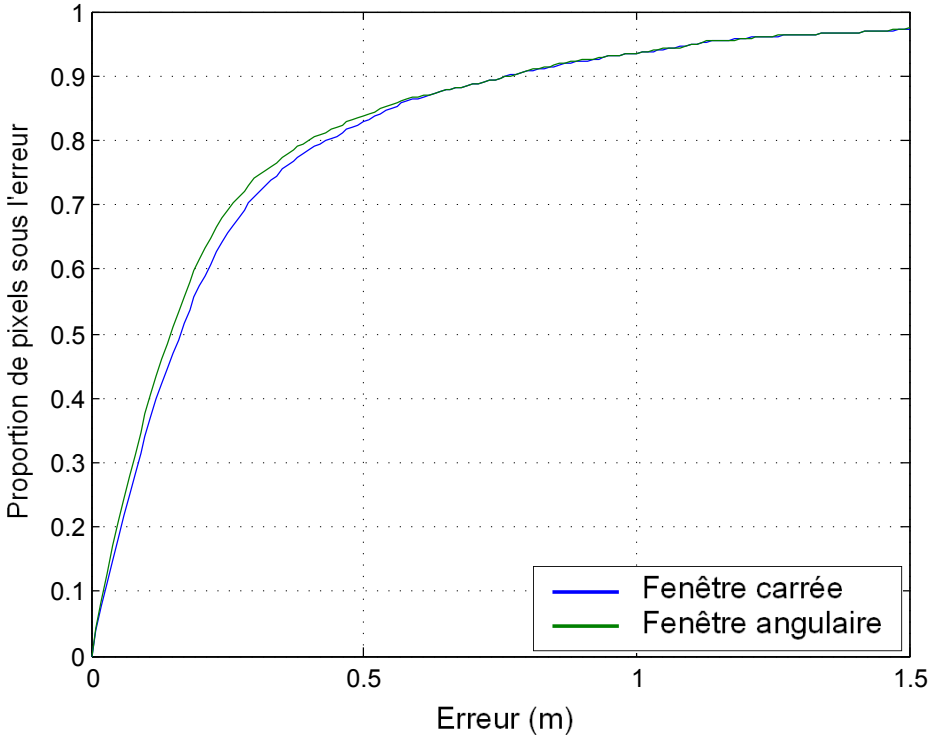


FIGURE 4.36 – *Comparaison des résultats suivant le type de fenêtre pour l'approche non dépliée*

Catégories de véhicules

Les phénomènes liés aux carrosseries sont souvent problématiques pour la qualité de l'extraction, comme expliqué en 4.1. Dans cette partie, l'hypothèse consiste à dire que les deux roues et les piétons ayant moins de carrosseries ou autres surfaces homogènes seront mieux détectés par la stéréovision. Pour vérifier ceci, nous sélectionnons manuellement les deux roues et les piétons dans un cas, les autres véhicules dans l'autre (voir figure 4.37). Les différents essais, quelle que soit l'approche, la taille de fenêtre, ou la mesure utilisée conduisent à deux conclusions fortes (voir figure 4.38) :

⇒ La qualité globale de la détection de l'ensemble des véhicules, 2R, piétons, est fortement influencée par la qualité de détection des VL et des bus, en raison du nombre de pixels plus importants occupés par leur forme.

⇒ La qualité de la détection de 2R et piétons est significativement supérieure à la qualité obtenue pour les autres véhicules.

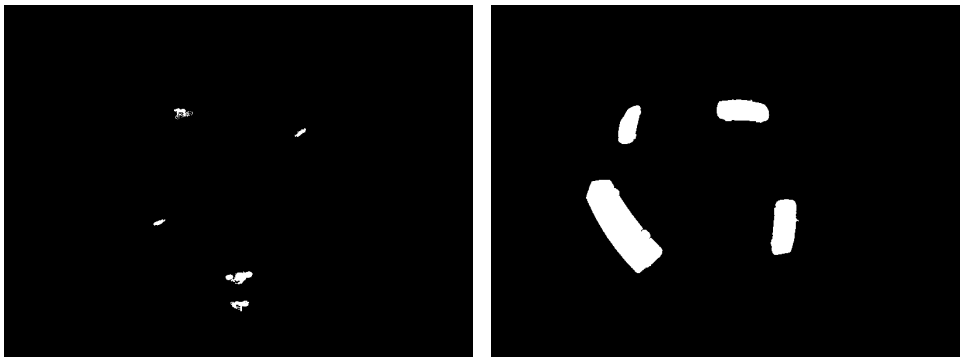


FIGURE 4.37 – *Masque de sélection : 2 roues et piétons (à gauche), VL et bus (à droite)*

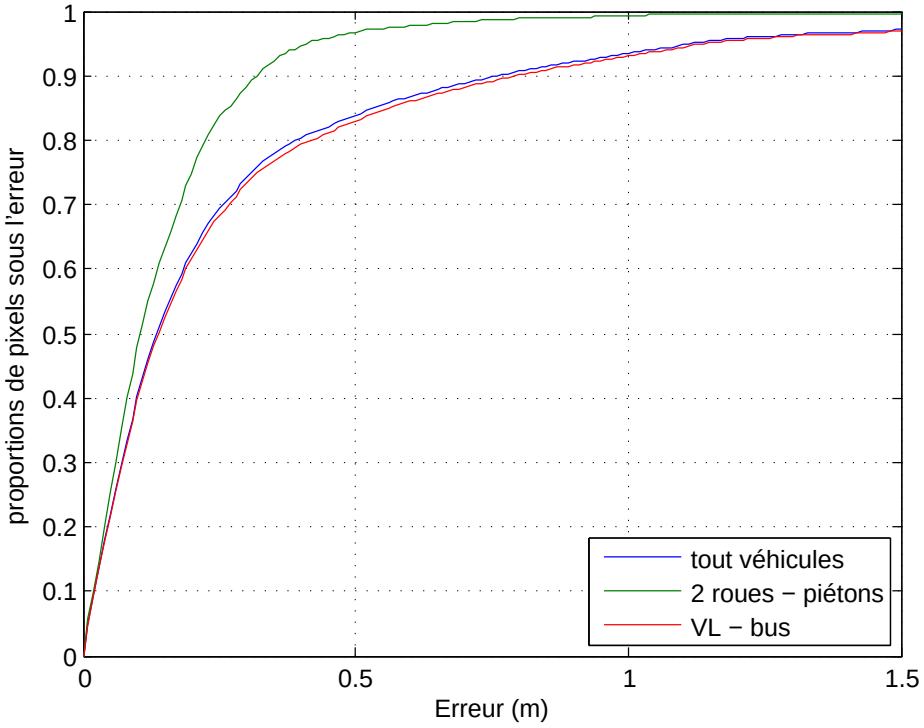


FIGURE 4.38 – *Comparaison des résultats suivant la catégorie de véhicules*

Approche pour le traitement stéréo

Afin de mesurer laquelle des trois approches utilisées pour le traitement stéréo (dépliée, non-dépliée, ou multi-couches) est la meilleure, nous les comparons avec une fenêtre de taille 11×11 et une mesure de distance SAD. Pour pouvoir évaluer la méthode dépliée, une adaptation est réalisée en dépliant la vérité terrain, et en appliquant une pondération sur les résultats pour tenir compte de l'étirement causé par le dépliage (voir figure 4.39). Les résultats, visibles sur la figure 4.40, montrent une nette amélioration pour des faibles erreurs avec l'approche multi-couche. En revanche, sur des plus grosses erreurs, l'approche n'est pas meilleure. Ces erreurs importantes représentent principalement des zones de carrosserie à grande surface homogènes.

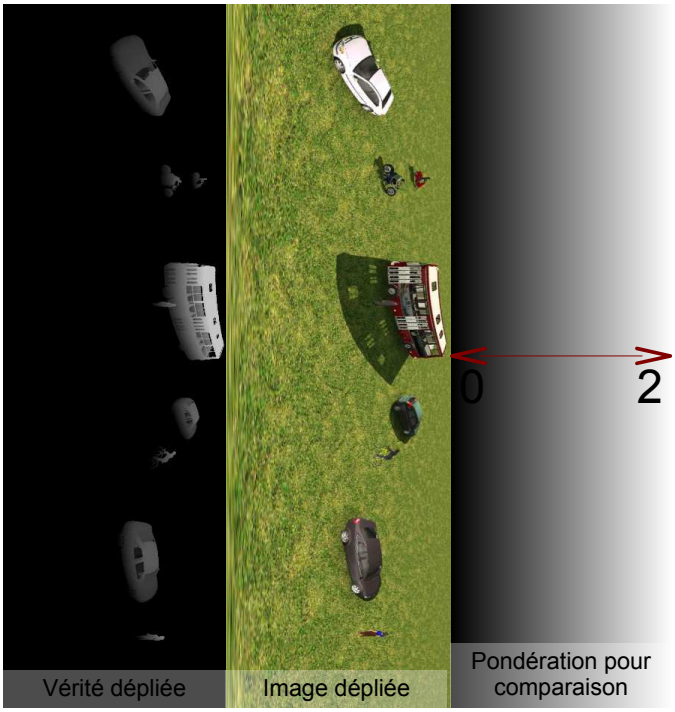


FIGURE 4.39 – *Vérité et pondération utilisées pour comparer l'approche dépliée avec les deux autres approches*

Ces résultats, vérifiés également quelle que soit la mesure de distance, confirment la supériorité de l'approche multi-couches aussi bien d'un point de vue de qualité de résultat, que de rapidité de calcul. Celle-ci demande un temps inférieur à une seconde pour comparer deux images, contre un temps de l'ordre de 20 secondes pour l'approche dépliée et encore plus élevé pour l'approche non dépliée [2].

4.4.5.4 Optimisation de l'approche multi-couches

Type de fenêtre de convolution

Deux types de fenêtrages possibles pour l'approche multi-couches ont été présentés, à savoir une fenêtre classique carrée, et une fenêtre gaus-

2. tests effectués sur un processeur Intel Xeon E5520 cadencé à 2,26 GHz

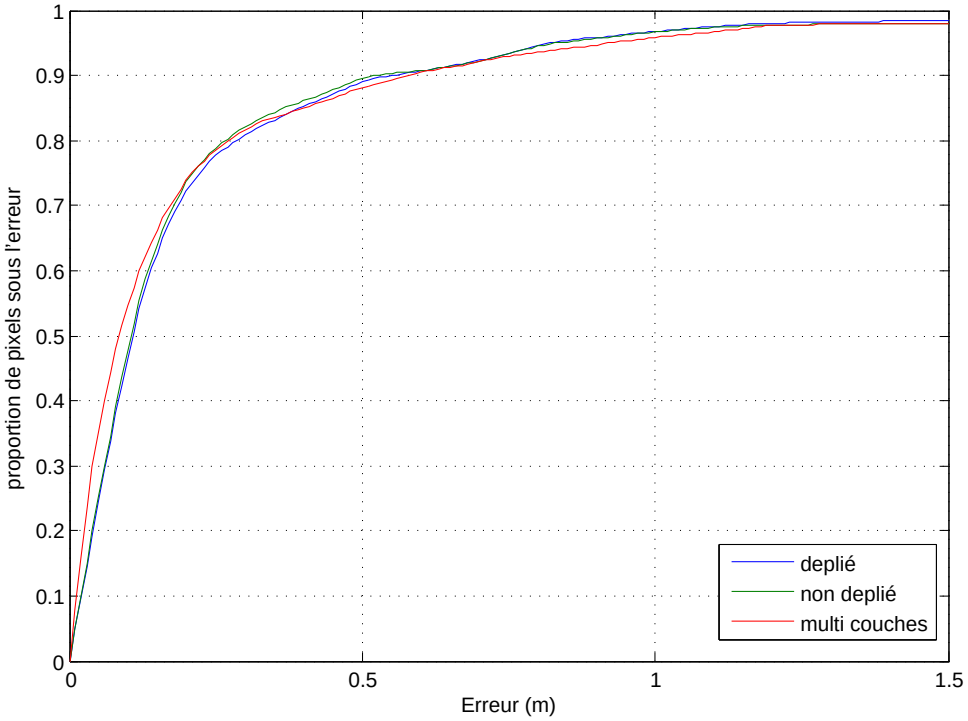


FIGURE 4.40 – *Comparaison des résultats suivant l'approche avec SAD et fenêtre de taille 11*

sienne. Nous avons donc testé différentes tailles de fenêtres. Les résultats obtenus sont présentés en figure 4.41. À titre de comparaison, la fenêtre carrée de taille 11×11 (courbe bleue) est la même que celle obtenue pour la méthode multi-couches lors de la comparaison des approches. Sur ces tests, les meilleurs résultats sont obtenus avec l'approche gaussienne, en particulier pour les erreurs faibles. Ensuite, la fenêtre gaussienne de taille 15 avec un écart type de 3 est meilleure ici, mais cette taille idéale varie en fonction de la résolution de l'image et de la taille des objets sur celle-ci.

Méthode de mesure de distance

Pour identifier la méthode donnant les meilleurs résultats, nous effectuons le test en choisissant la configuration qui a donné les meilleurs résultats jusque là : une approche multi-couches avec une fenêtre gaussienne de taille 15 × 15 et un écart type de 3, en faisant varier la mesure de distance utilisée. Les premiers résultats (voir figure 4.42) montrent de faibles différences entre les différentes méthodes de mesures de distance choisies, avec néanmoins un résultat légèrement moins bons pour la méthode CENSUS. En y regardant de plus près, les meilleurs résultats sont obtenus avec la SAD.

Toutefois, et comme expliqué précédemment, la SAD comme la SSD ne peuvent pas être utilisées sur les scènes en extérieur du fait de leur sensibilité à la variation de luminosité ou de contraste entre les images des deux caméras. Pour vérifier ceci, le test suivant est réalisé à partir des mêmes images, à ceci près que l'une des images a été modifiée de sorte qu'elle soit légèrement assombrie. Les résultats obtenus (voir figure 4.43) montrent

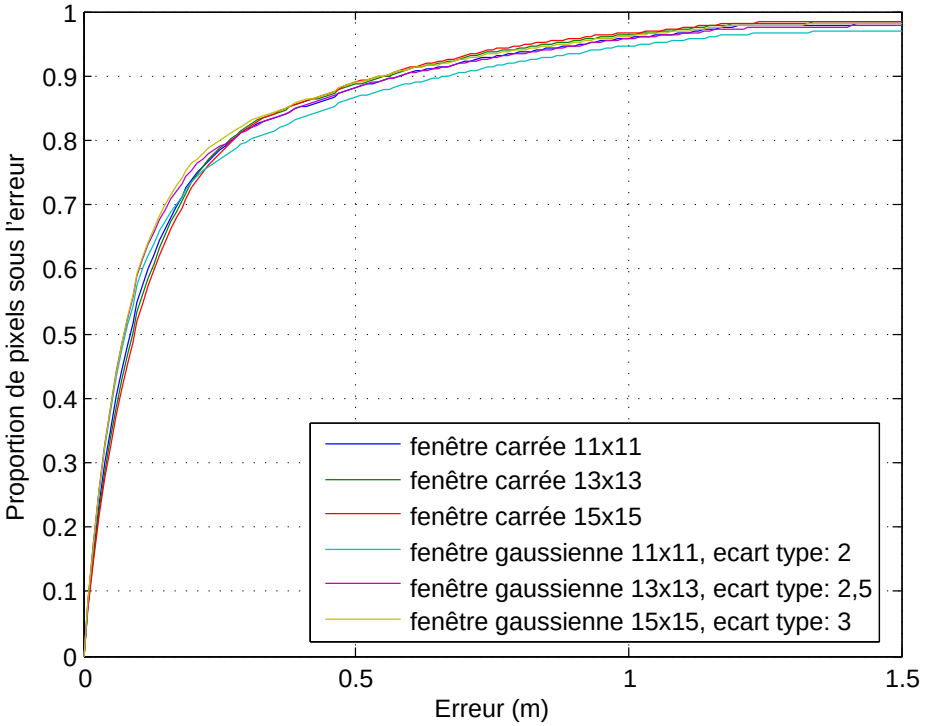


FIGURE 4.41 – *Résultats de l'approche multi-couche en fonction du type de fenêtre*

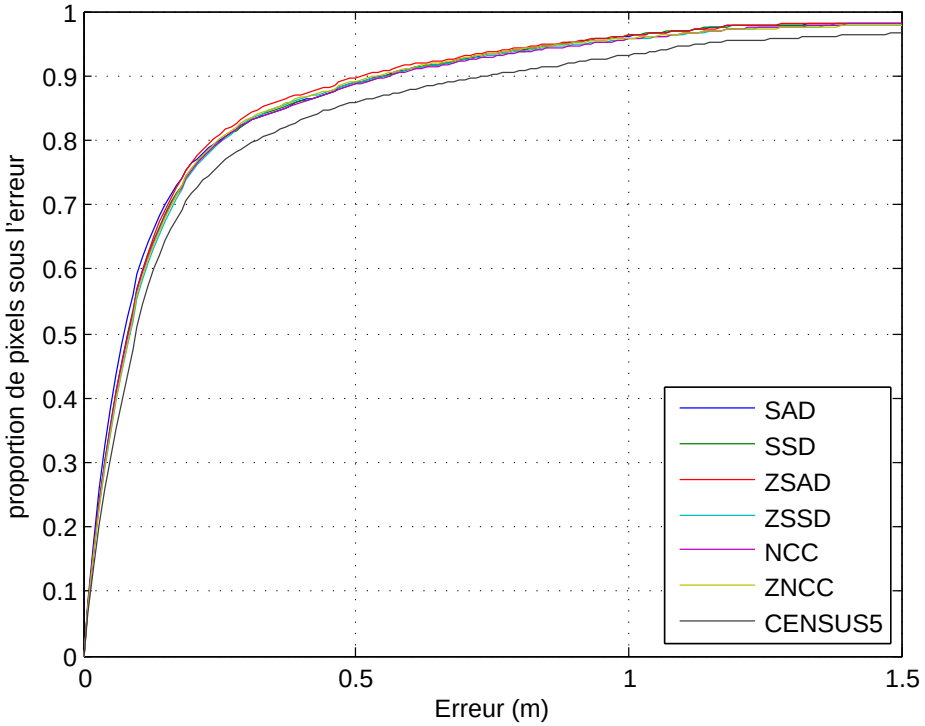


FIGURE 4.42 – *Résultats de l'approche multi-couches en fonction de la méthode*

bien les limites des SAD et SSD ce qui nous permet de les écarter définitivement pour la suite de l'étude. Les scores obtenus pour les autres méthodes restent sensiblement les mêmes.

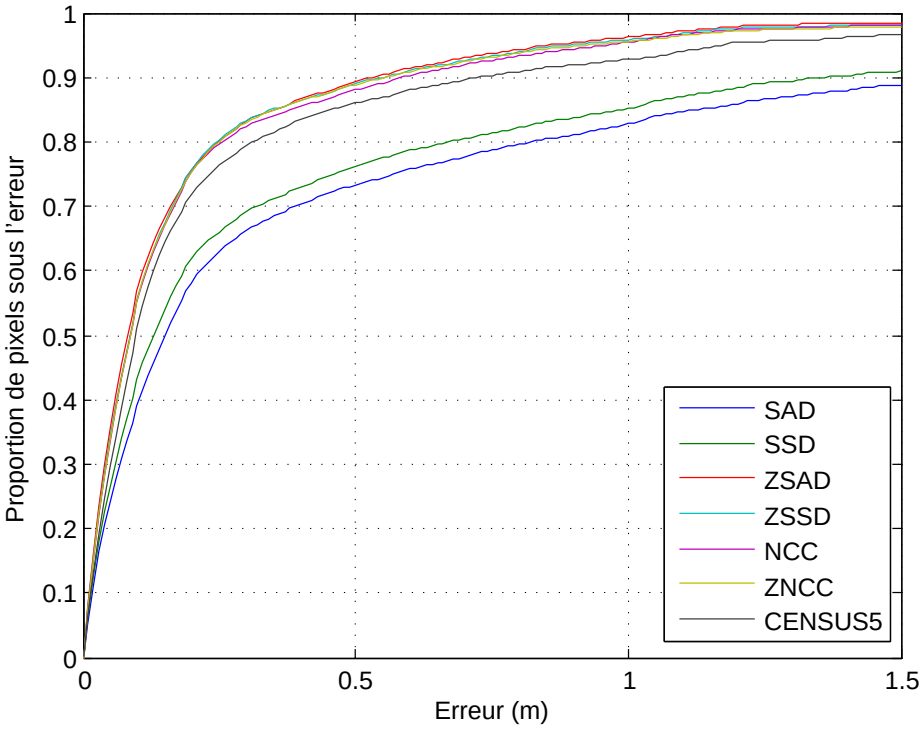


FIGURE 4.43 – *Résultats de l'approche multi-couches en fonction de la méthode, avec différence de luminosité entre les images*

Pour la méthode CENSUS, les résultats sont toujours faiblement inférieurs mais ceci est à prendre avec précaution. En effet, du fait de la taille de la transformée, la taille de fenêtre n'est pas réellement comparable avec les autres méthodes. Pour la suite de l'étude nous choisissons donc d'utiliser principalement les méthodes NCC, ZNCC, ainsi que la méthode CENSUS pour des tests face à des conditions réelles.

4.4.5.5 Analyse

Les résultats présentés ci-dessus ne se basent que sur une seule scène simulée. Par conséquent, ils n'ont qu'une valeur indicative. On peut néanmoins retenir plusieurs choses :

- ⇒ La meilleure approche pour le calcul stéréo est le calcul multi-couches. Cette approche donne des résultats meilleurs et un temps de calcul significativement réduit.
- ⇒ Dans cette approche, le choix d'une fenêtre de type gaussienne est préférable, mais la taille idéale ainsi que l'écart type dépendent de la scène.
- ⇒ La détection est meilleure pour les petits véhicules (piétons et deux-roues)
- ⇒ Il faut mieux choisir la caméra basse en référence pour le calcul

4.5 CONCLUSION

Nous avons étayé le choix du système stéréo adapté au suivi des véhicules sur une intersection. Un prototype a été réalisé pour acquérir des données. Ce système demande une adaptation particulière des approches et des méthodes de calibrage et de traitement de la stéréovision, comme nous avons pu le voir. D'après cette étude, l'approche de la stéréovision dite "multi-couches" associée à une fenêtre gaussienne semble donner les meilleurs résultats. Ce travail nous donne des indications sur les choix de paramètres à opérer. Nous allons voir dans la partie suivante les résultats de l'application de ce traitement sur des séquences simulées réelles.

RÉSULTATS 5

SOMMAIRE

DANS ce chapitre, nous présentons les résultats obtenus à partir d'une part des vidéos simulées et d'autre part, et en majeure partie, des vidéos réelles réalisées sur site. Ces essais ont pour but de voir dans un premier temps si la calibration est fonctionnelle, puis si l'acquisition des données permet un traitement stéréoscopique et enfin si le système de stéréovision permet d'apporter des informations suffisantes à la détection des deux-roues.

Pour cela, nous avons dans un premier temps, réalisé ces essais à l'aide de vidéos produites par le simulateur Sivic [78], puis à l'aide de vidéos issues de caméras réelles et acquises dans le cadre des essais du projet METRAMOTO, avec des passages de véhicules bien définis. Enfin, nous avons réalisé d'autres essais à l'aide du système prototype sur le site de Nantes, avec une circulation plus "libre" en intersection.

En analysant les limites observées lors de ces essais, nous proposons par la suite, des pistes permettant d'améliorer la détection des deux-roues.

5.1 SIMULATION SIVIC

Pour valider le système, l'étape préalable fût de tester celui-ci à l'aide de vidéos simulant deux types de situations : des passages simples de véhicules et un trafic plus complexe en intersection. Dans tous les cas simulés, la fonction de représentation $r(\theta)$ est parfaitement connue. Néanmoins, nous faisons comme si elle ne l'était pas, et modélisons une mire à damier virtuelle que nous plaçons devant le capteur afin de simuler la calibration de celui-ci et, ainsi, de vérifier le bon fonctionnement global de l'approche.

5.1.1 Essais 1

Scénario

Dans un premier temps, nous réalisons des essais en simulant une file de vélos circulant sur une rue et dépassée par des voitures (visible en figure 5.1).

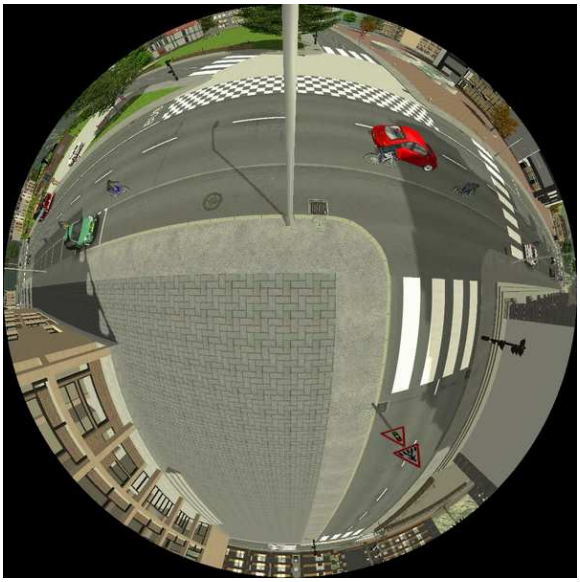


FIGURE 5.1 – *Essais sous Sivic*

Ces essais visent d'abord à évaluer la capacité du système à différencier des objets. L'intérêt de ces essais est également de pouvoir tester des situations que l'on ne pourrait vérifier lors d'essais réels pour des raisons évidentes de sécurité, comme des passages de véhicule à une très grande proximité des cyclistes : les trajectoires ont été définies pour que la distance entre le cycliste et la voiture soit de moins de 50cm.

Résultats

L'extraction fond forme combinée à la stéréovision permet bien d'isoler les véhicules. Le volume des véhicules est bien détecté, et dans le cas d'une voiture dépassant un vélo, la différence de hauteur des pixels voisins permet de différencier les véhicules. En ce qui concerne le traitement, celui-ci donne de bons résultats quel que soit le mode de transformation de l'image (voir paragraphe 4.4.4.2) choisi dans l'approche multi-couche.

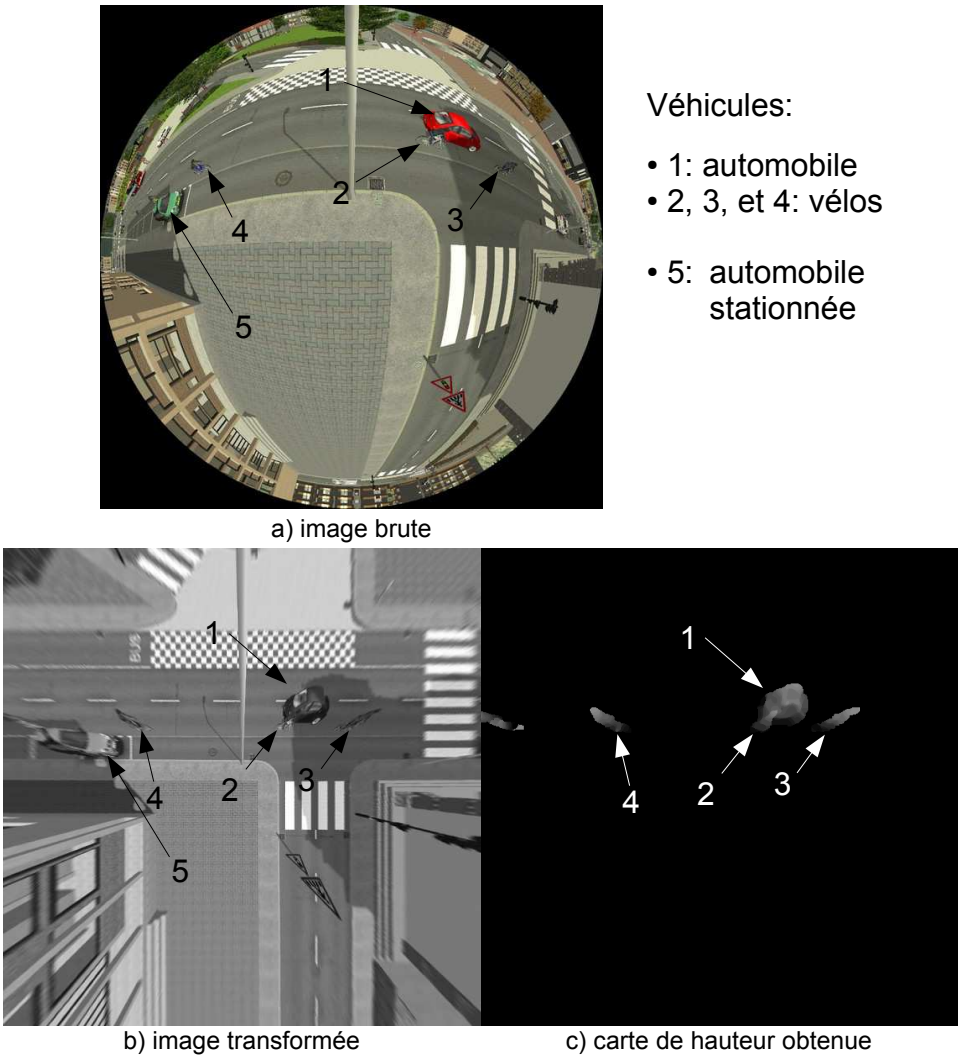


FIGURE 5.2 – *Résultats des essais sous Sivic*

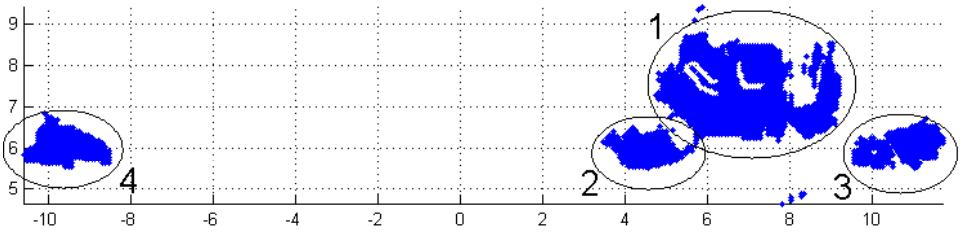


FIGURE 5.3 – *Résultats des essais sous Sivic (reprojection des points)*

Les résultats obtenus (voir figures 5.2 et 5.3) montrent également que l'empreinte des deux-roues est très différente de celle des VL. De plus, ces données projetées dans un repère 3D mettent en évidence la différenciation des véhicules malgré leur grande proximité.

En intégrant ces données dans le logiciel de comptage et de suivi déjà existant [91], nous pouvons recenser les différents types de véhicules comme le montre la figure 5.4. Dans ce logiciel, deux zones de comptage (une entrée et une sortie de l'intersection) sont définies. Les véhicules passant dans ces deux zones sont comptabilisés. En outre, ils sont suivis entre

les zones entrantes et sortantes, ce qui permet d'associer chaque entrée et chaque sortie.

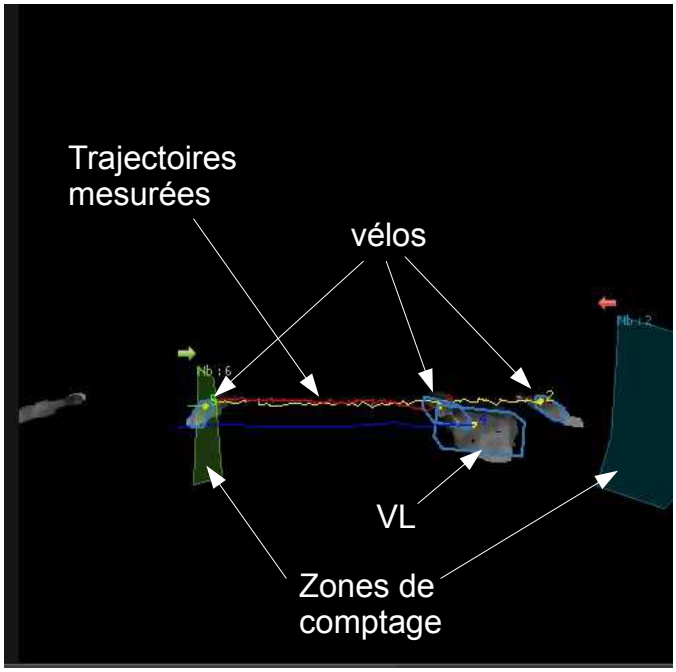


FIGURE 5.4 – *Suivi et comptage à partir des données extraites de la stéréovision*

5.1.2 Essais 2

Scénario

Nous avons réalisé une séquence d'essais sous Sivic simulant un trafic plus varié sur une intersection urbaine (figure 5.5). Cette séquence possède différents éléments caractéristiques :

⇒ Voitures stationnées
⇒ Traversées de piétons
⇒ Trafic varié (voitures, vélos, piétons, motos, scooter…)
⇒ Arrêt et démarrage de véhicules à un feu
⇒ Dépassements
⇒ Démarrage simultané des vélos et voitures aux feux.

Ces essais sont réalisés avec deux configurations distinctes :

⇒ Les deux caméras sont parfaitement alignées
⇒ Une rotation légère est appliquée à l'une des deux caméras

À chaque fois, Sivic fournissait des vérités terrain afin de les comparer aux résultats obtenus, et de mesurer ainsi l'influence d'une correction d'alignement des caméras sur les résultats du traitement.

Résultats

Ici, le but est d'effectuer une comparaison avec la vérité terrain sur l'ensemble d'une séquence. Le critère introduit dans la partie 4.4.5 (une courbe) est adapté à une paire d'images mais pas à une séquence totale. Pour utiliser ce critère au niveau de la séquence, nous attribuons un score à chaque

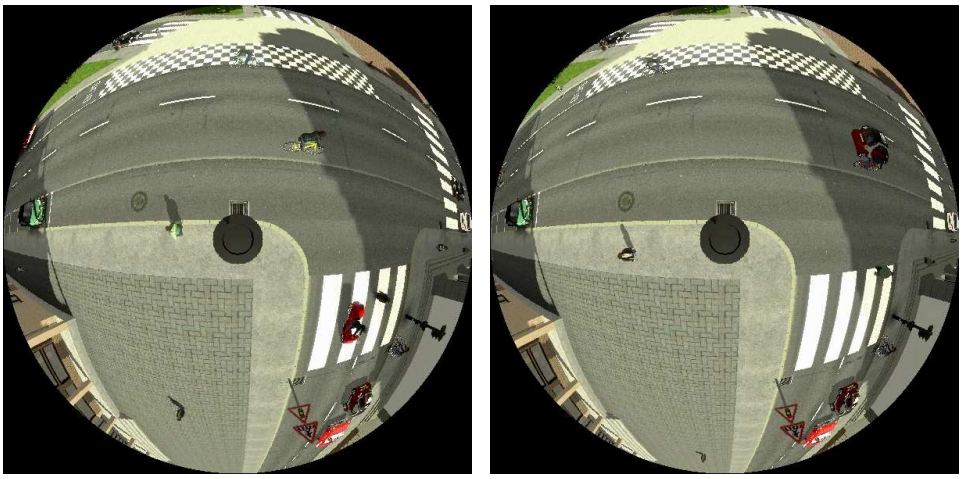


FIGURE 5.5 – *Séquence d'essai en simulation*

paire d'images correspondant au rapport entre l'aire sous la courbe de répartition des erreurs et l'aire maximum possible (voir figure 5.6) pour l'intervalle d'erreurs allant de 0 à 1,5m.

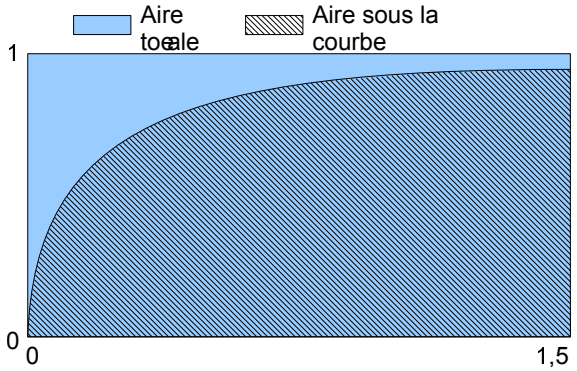


FIGURE 5.6 – *Critère de calcul du score d'une séquence*

En prenant la notation précédente (voir equation 4.40), on peut noter ce critère S comme :

$$S(i) = \frac{1}{1,5} \int_{u=0}^{u=1,5} P_i(u) \tag{5.1}$$

avec i l'index de l'image traitée.

Il devient alors possible de tracer une courbe d'évolution de S tout au long de la séquence et d'en tirer des enseignements. Ceci permet de constater que les scores sont globalement élevés tout au long de la séquence (voir le graphique supérieur de la figure 5.7). On constate des scores plus faibles dans les images comprenant plus de pixels forme (voir le graphique inférieur de cette même figure), car lorsqu'il y a des gros véhicules dans l'image, et qu'ils comportent des surfaces homogènes, ceux-ci sont moins bien détectés en stéréovision.

En outre, en comparant les scores issus des résultats obtenus lors de cette seconde série d'essais (figure 5.8), on remarque que, dans la grande majorité des cas, les résultats avec ou sans correction d'alignement sont quasi-identiques sur la majorité de la séquence. Les rares zones où la différence est plus marquée correspondent à des passages de la séquence

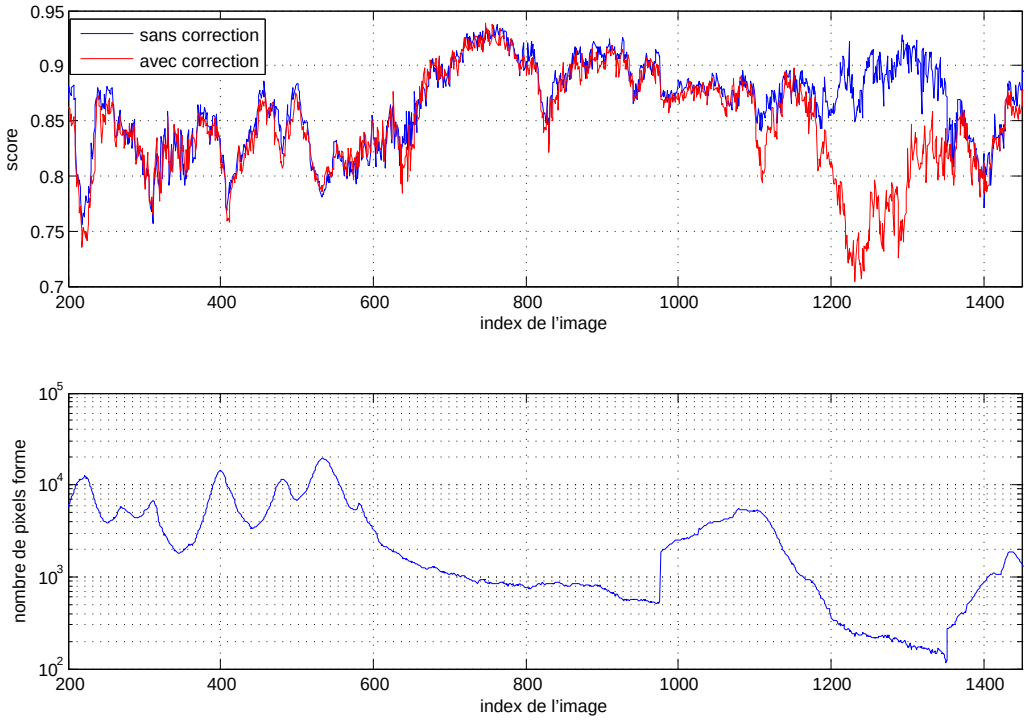


FIGURE 5.7 – *Comparaison avec ou sans rotation (en haut) et nombre de pixels correspondant(en bas)*

contenant très peu de pixels forme qui décrivent des véhicules assez lointains, comme le montre le graphique inférieur de cette même figure.

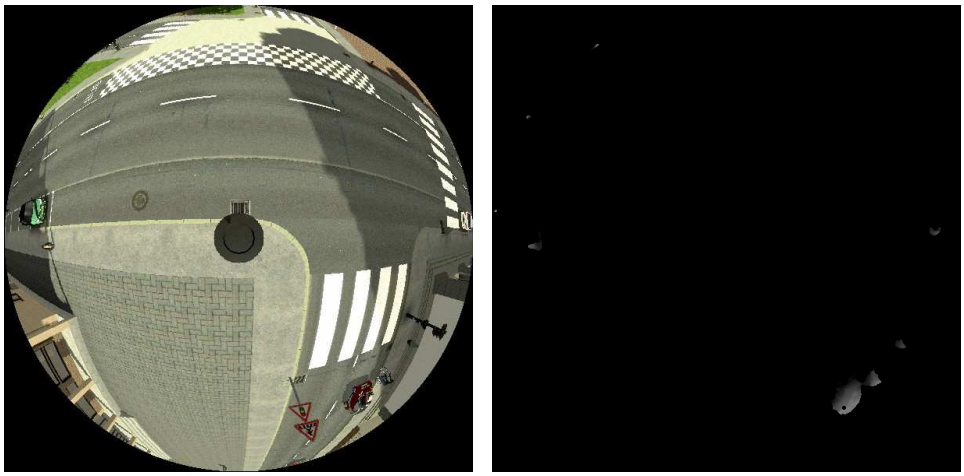


FIGURE 5.8 – *Résultats obtenus lors des essais simulés*

Si l'on prend un cas assez complexe avec une circulation mixte (2RM et VL) avec des ombres portées au sol rendant problématique la détection par la seule extraction fond/forme, on constate, d'après les résultats obtenus (figure 5.9) par le traitement stéréoscopique et la reprojection des points dans un repère réel (figure 5.10), que le calcul de carte de hauteur permet bien :

⇒ de s'affranchir des ombres portées au sol

⇒ d'obtenir la position et les dimensions de chacun des véhicules
⇒ de différencier les VL des 2RM
⇒ de différencier deux véhicules proches (la moto 2 et et le VL 5 par exemple)
⇒ *in fine* d'effectuer une bien meilleure détection que l'extraction fond/forme seule

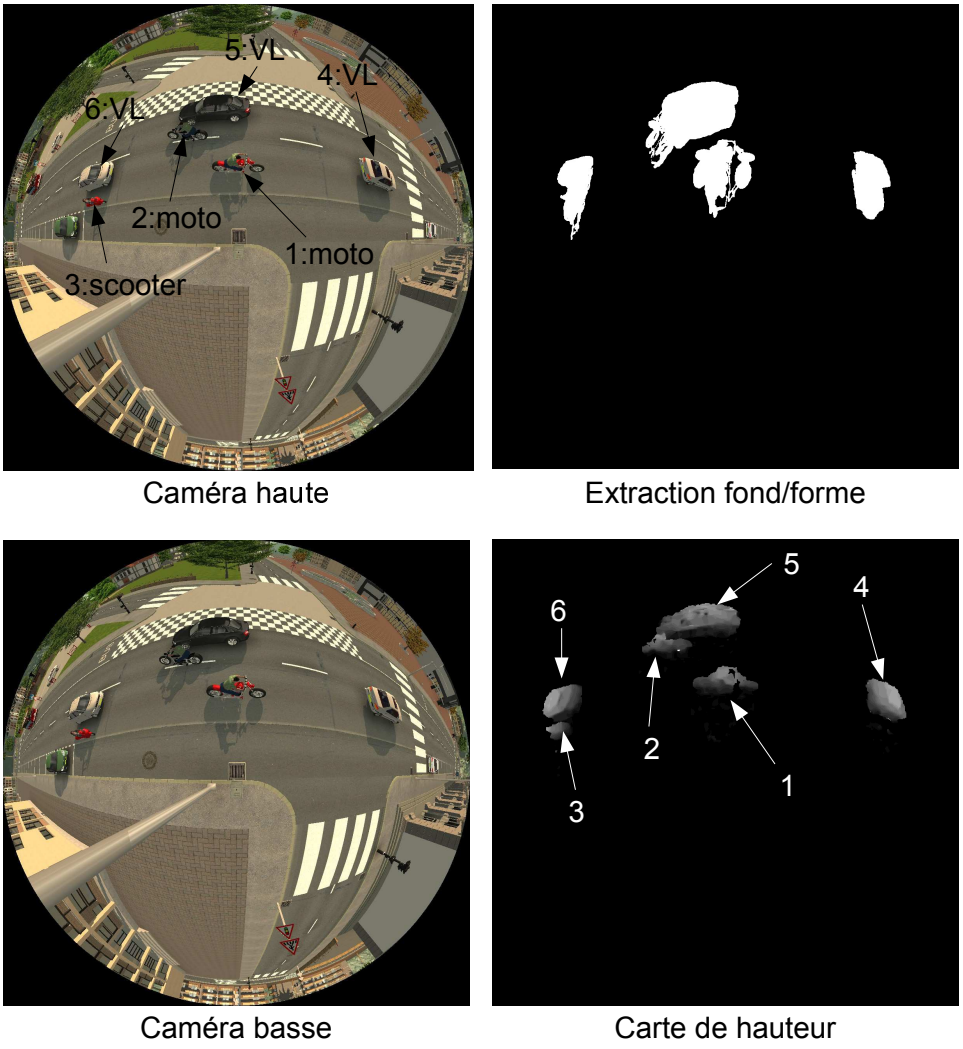


FIGURE 5.9 – *Résultats obtenus lors des essais simulés avec ombres*

Ces résultats répondent parfaitement à la problématique de détection des deux-roues.

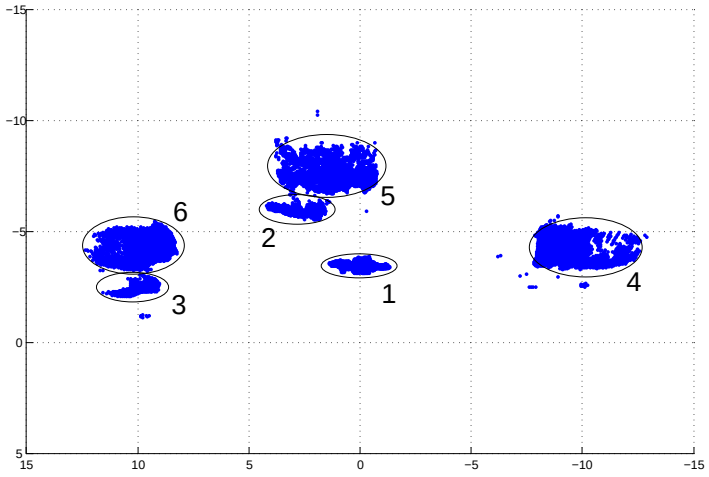


FIGURE 5.10 – *Reprojection des points de l'essai avec ombres, vue de dessus*

5.2 ESSAIS RÉELS

Les essais réels permettent de valider le choix des capteurs, l'approche de traitement utilisée, la mise en œuvre du système, et d'identifier tous les autres problèmes qui auraient pu être oubliés.

5.2.1 Essais sur une section droite

Scénario

Dans le cadre du projet METRAMOTO, des essais ont été réalisés avec des passages de véhicules, dont des 2RM, selon des scénarios très précis proposant diverses situations de circulation, à différentes vitesses (illustrées dans la figure 5.11) :

⇒ Moto en interfile
⇒ Moto seule (voir figure 5.12)
⇒ Voiture seule
⇒ Moto devant une voiture
⇒ Moto derrière une voiture
⇒ Moto entre deux voitures
⇒ Moto qui double des voitures

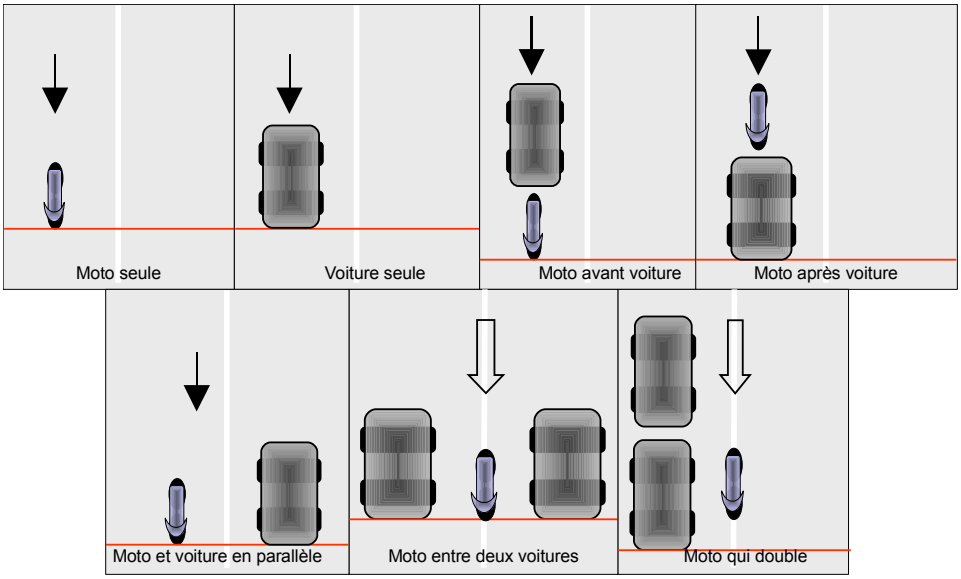


FIGURE 5.11 – *Situations testées dans les essais METRAMOTO*

Résultats

Lors des essais METRAMOTO, la calibration mise en œuvre utilisait le logiciel Hyscas (voir paragraphe 4.3.1).

On peut remarquer que la distinction se fait bien entre deux véhicules proches grâce à la 3D, comme le montre la projection des points issus de la carte de hauteur (voir figures 5.13 et 5.14). En revanche, comme expliqué lors de la partie 4.1, la détection est plus problématique sur les voitures en raison de leurs surfaces lisses (accentué par le blanc) et des reflets sur le pare-brise. Ce phénomène lié aux reflets se constate également très

FIGURE 5.12 – *Essais réalisés dans le cadre de du projet METRAMOTO - Image brute*

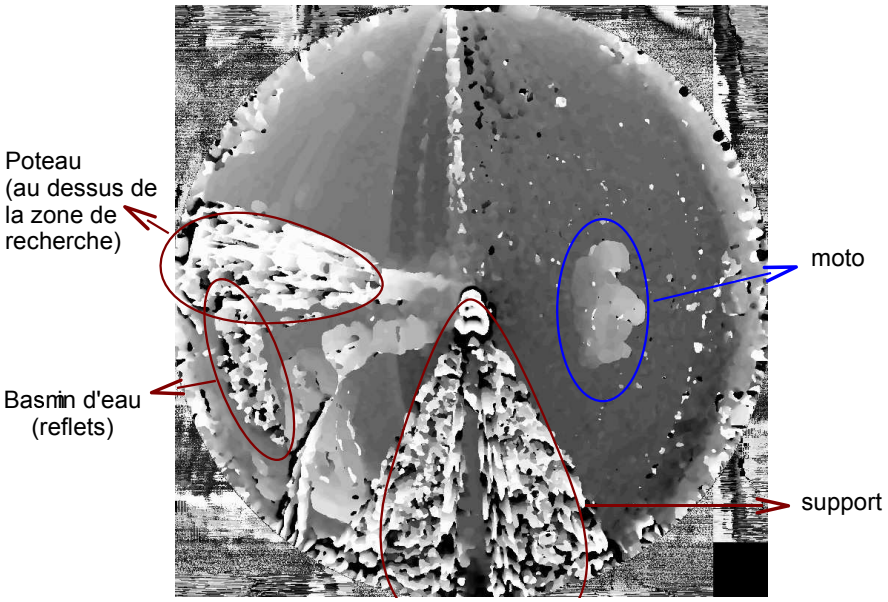


FIGURE 5.13 – *Résultats des essais réalisés dans le cadre de METRAMOTO - Carte de hauteur obtenue par l'application du traitement proposé*

bien dans le bassin d'eau à gauche de la scène. Néanmoins, ce problème de mauvaises détections peut être en grande partie supprimé, les valeurs obtenues dans ces zones se situant, pour une grande majorité d'entre elles, aux bornes de la zone de recherche en hauteur. À noter que, comme on peut le remarquer sur la figure 5.14, l'extraction fond/forme détecte les zones du support des caméras comme de la forme. Ceci est dû au fait que la réflexion des véhicules blancs sur ce support modifie brièvement sa luminosité. Néanmoins, ce point n'est pas problématique puisqu'il suffit de bien sélectionner la zone d'intérêt pour le traitement des images, et ainsi, ignorer cette partie.

Dans la situation de circulation en interfile, tout l'apport du système est démontré. Les résultats de l'extraction fond/forme montrent (voir fi-

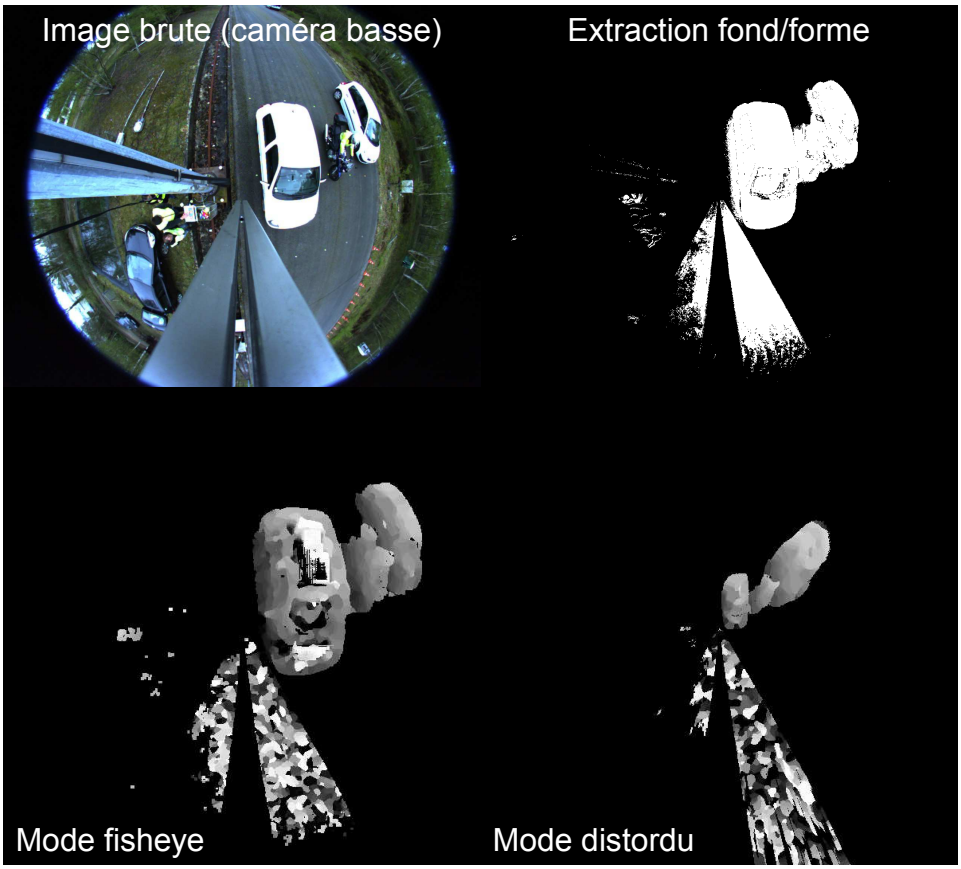


FIGURE 5.14 – *Résultats des essais métramoto avec une moto en interfile*

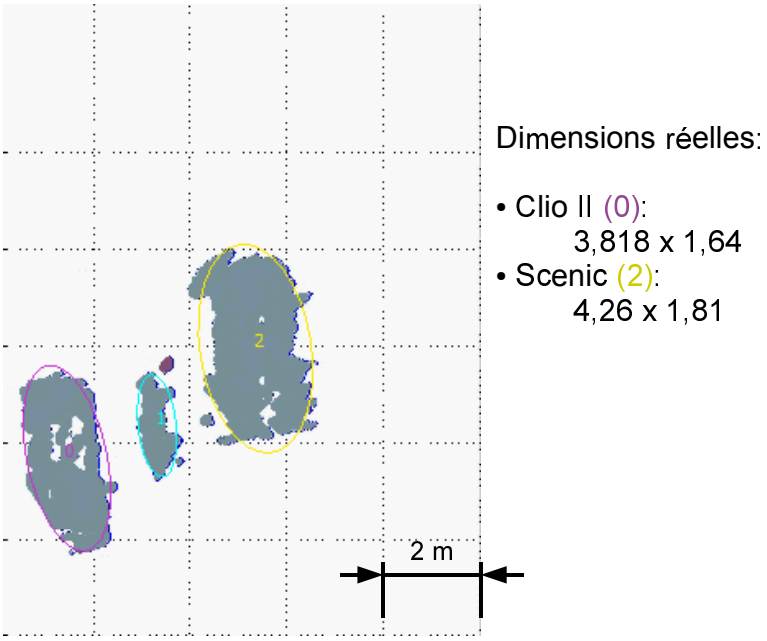


FIGURE 5.15 – *Projection des points en vue de dessus*

gure 5.14) bien l'incapacité de cette approche à détecter et à isoler les deux-roues en interfile. La stéréovision, quant à elle, résout ce problème car tous

les points formes étant calculés en 3D dans le repère réel, il est alors possible de séparer les véhicules (voir figures 5.15 et 5.16).

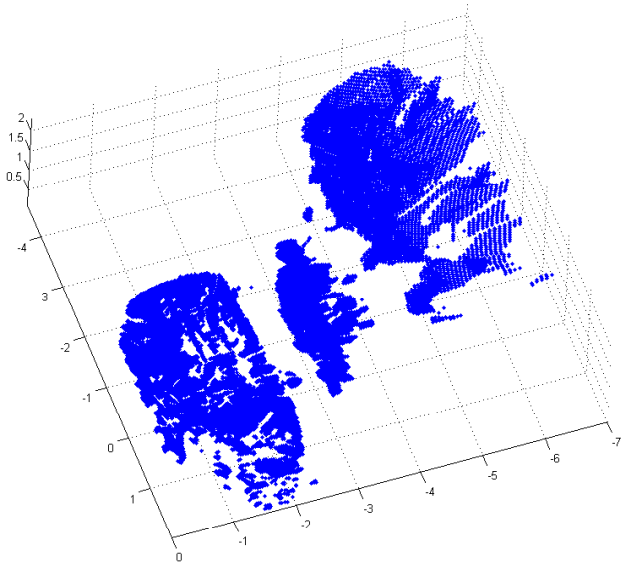


FIGURE 5.16 – *Représentation sous forme de points à partir des données calculées en mode fisheye*

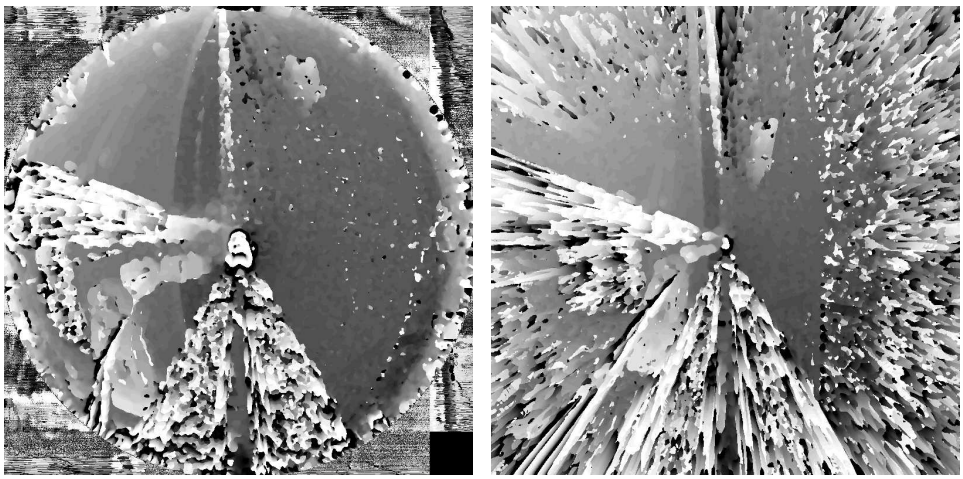


FIGURE 5.17 – *Comparaison de l'influence du mode utilisé : fisheye (à gauche) et distordu (à droite)*

La figure 5.17 montre que le résultat du calcul de carte 3D est beaucoup plus bruité en mode distordu. Ceci s'explique par le fait que les zones extérieures sont très étirées dans ce mode. Dans ce cas, une légère imprécision de calibration à davantage d'impact. L'autre inconvénient du mode distordu réside dans le fait que l'on réduit la zone centrale où il y a beaucoup d'informations, et qu'on agrandit des zones lointaines, donnant plus d'importance à ces zones, qui sont pourtant les moins précises.

Les résultats transformés en nuages de points dans les figures 5.16 et 5.19 semblent confirmer également cette remarque. Avec, le mode distordu, la détection de la voiture lointaine reste approximative (mais néanmoins suffisante) alors que, pour la voiture proche, on ne tire pas parti de la précision de la caméra.

FIGURE 5.18 – *Image des essais METRAMOTO en mode distordu*

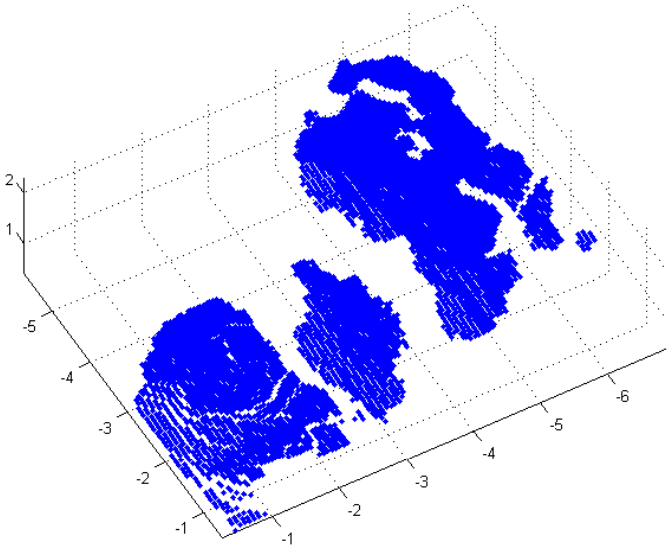


FIGURE 5.19 – *Représentation sous forme de points à partir des données calculées en mode distordu*

Enfin, les dimensions des nuages de points observés sont parfaitement cohérents avec les dimensions de véhicules fournies par les constructeurs (voir figure 5.15).

Sur les images distordues, on peut remarquer que les zones droites de la scène restent courbées (figure 5.18). Ceci est révélateur d'une imperfection dans l'estimation des paramètres intrinsèques. C'est très probablement lié aux difficultés rencontrées lors de l'essai dans la phase de calibrage du système, les mires n'ayant pas couvert entièrement le champ de vision des optiques. Malgré cette imperfection, la détection est tout à fait possible. De

meilleurs résultats peuvent donc encore être espérés lorsque les procédures de calibration seront rigoureusement mises en œuvre.

5.2.2 Essais sur intersection

Scénario

Afin de voir la capacité de traitement du système avec une circulation plus complexe, nous avons placé le système stéréo sur une intersection en site fermé. Puis différents véhicules ont circulé librement, sans consigne particulière, tout en respectant le code de la route. Ainsi, nous avons pu réaliser des vidéos contenant :

⇒ des piétons
⇒ des vélos
⇒ des motos
⇒ un bus
⇒ des voitures (différents coloris et tailles)

Du fait de la circulation "libre", ces vidéos contiennent différentes situations de masquage et sont donc plus fidèles à une circulation réelle (voir figure 5.20).

Le but de ces essais est également de tester la mise en œuvre de l'approche de calibration sur site réel. Cette calibration est réalisée en deux étapes :

⇒ La première consiste à calibrer chaque capteur séparément. Pour cela, la calibration est réalisée à partir de mires à damier filmées de façon à couvrir l'ensemble du champ de vision d'après la méthode de Scarramuzza. Les capteurs sont calibrés une fois installés sur le support, mais avant que ce support ne soit fixé au sommet du mat.
⇒ Une fois le support des caméras placé en hauteur, la seconde étape concernant les paramètres extrinsèques est réalisée comme décrit dans la partie 4.3.2.2

À noter que ces essais devaient également permettre de valider la synchronisation des deux caméras. Celle-ci est réalisée à l'aide d'un circuit externe envoyant un signal aux deux caméras au même instant.

FIGURE 5.20 – *Images issues de l'essai sur intersection, caméra haute (à gauche) et caméra basse (à droite)*

Résultats

Dans les essais réalisés sur site, des véhicules circulent librement. Les résultats obtenus à partir de la méthode multi-couches confirment que le rayon de couverture est fortement réduit en utilisant le mode distordu (par rapport aux autres modes, par exemple le fisheye) comme le montre la figure 5.21.

La calibration n'ayant pas été effectuée de manière satisfaisante, deux problèmes ont été mis en évidence :

⇒ La détection correcte ne dépasse pas 15m de rayon autour du système

⇒ Les hauteurs des véhicules détectés ne sont pas toujours correctes

Á nouveau, on fait le même constat. Le passage des mires devant les caméras n'est pas aussi simple qu'il n'y parait, et une zone du champ de vision peut facilement ne pas être couverte si la procédure n'est pas suffisamment rigoureuse. Une attention particulière devra donc se porter sur ce point lors des prochains travaux.

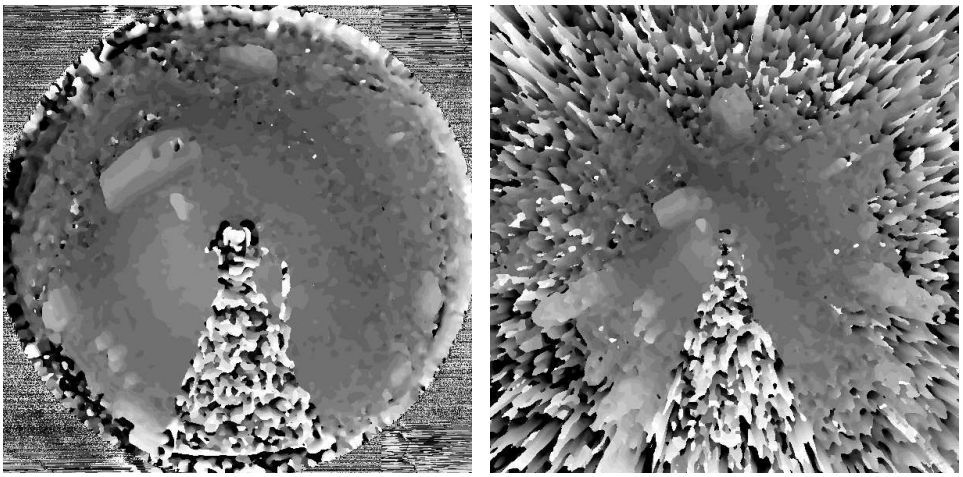


FIGURE 5.21 – *Comparaison de l'influence du mode de calcul, fisheye (à gauche) ou distordu (à droite)*

En regardant en détail la projection des points obtenus dans un repère 3D (figure 5.22), on constate que la détection permet de différencier les voitures, des piétons et des deux-roues. La notion de profondeur est bien distinguée et permet donc une différentiations d'objets semblants proches sur un plan 2D, comme cela est le cas avec le piéton derrière la voiture sur la partie de droite de cette même figure. En revanche, la différenciation entre les deux roues et les piétons est plus délicate. Elle est facilitée à proximité du centre de la scène, mais, plus complexe à grande distance comme cela peut se voir sur la partie de droite de la même figure.

La meilleure détection avec les vidéos prises dans le cadre de Metramoto peut s'expliquer également par le fait que, lors de ces essais, les caméras utilisées n'étaient pas les mêmes. Les images provenant des essais en intersection sont de même définition mais plus bruitées. Ceci explique en partie les erreurs de détection à grande distance, plus importantes dans les essais sur intersection.

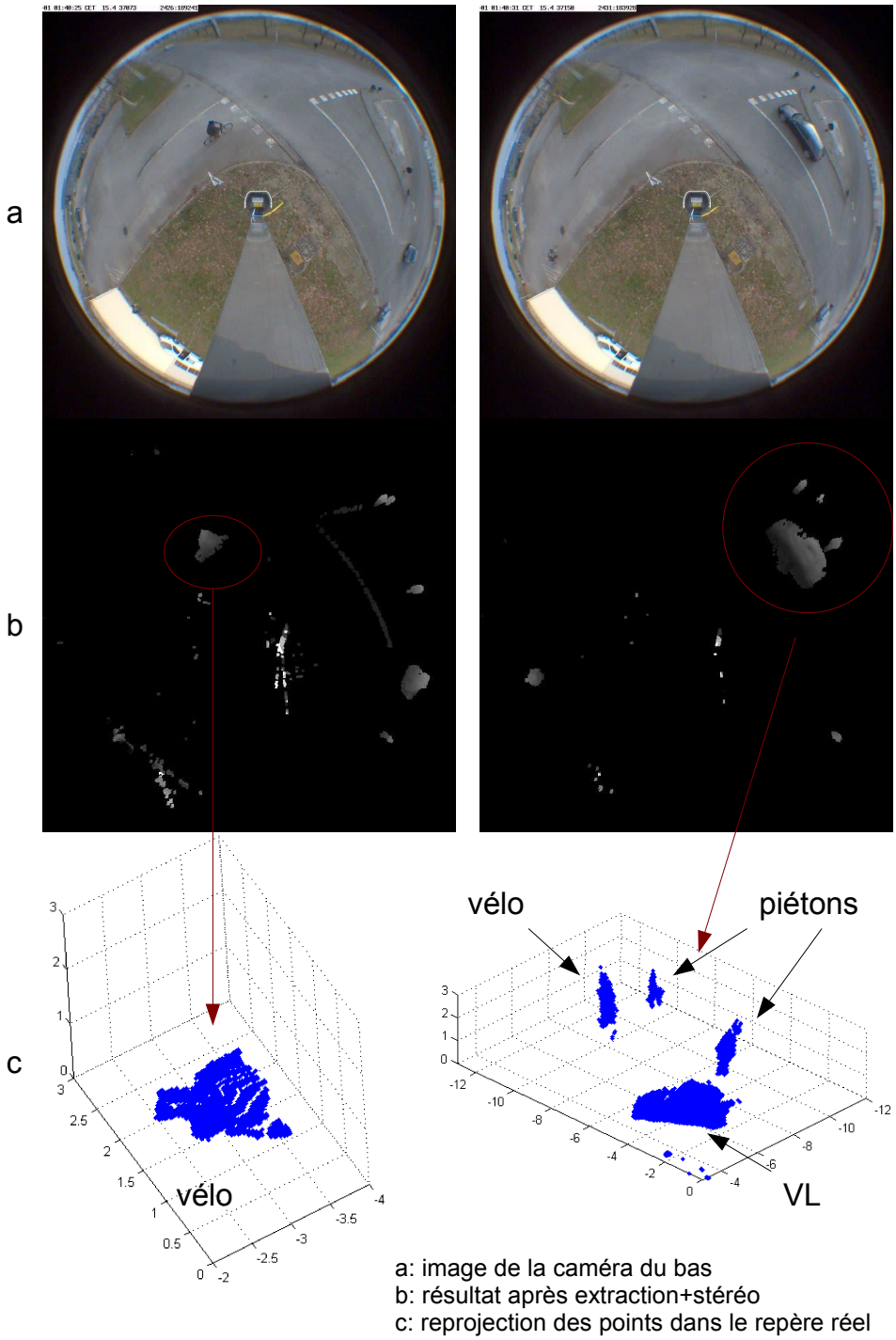


FIGURE 5.22 – *Résultats des essais libres*

5.3 AMÉLIORATIONS POSSIBLES

En ce qui concerne les vidéos réelles, nous n'avons pas pu mener à bien tous les essais que nous aurions souhaités en raison de problèmes divers sur les caméras (retard de livraisons, pannes, retours en réparation,...). Pour ces raisons, la procédure de calibration n'a pu être affinée. Pour valider complètement la démarche, il faudrait mener à bien les travaux suivants :

⇒ Recalibrer le système avec l'approche proposée en veillant à bien couvrir tout le champ des caméras.
⇒ Vérifier si cette nouvelle calibration améliore la détection, si la zone de couverture du système après traitement est agrandie, en particulier dans le cas du mode distordu.
⇒ Améliorer le traitement des données en appliquant des post-filtrages sur celles-ci.
⇒ Regrouper les données reçues sous forme de nuages de points pour délimiter et catégoriser les véhicules.
⇒ Intégrer ces données dans un logiciel de suivi déjà existant.

Enfin, en ce qui concerne la synchronisation des caméras, point indispensable pour pouvoir effectuer un traitement en stéréovision, malgré un bon fonctionnement de celle-ci, il a été constaté sur des vidéos longues de quelques dizaines de minutes, de rares cas de perte d'une image pour l'une des caméras. Par conséquent, il faut prendre en compte ces décalages et améliorer le circuit de synchronisation, ou choisir d'autres caméras plus facilement synchronisables.

5.4 Conclusion

Nous avons pu voir, dans ce chapitre, les résultats obtenus pour la détection des deux-roues à l'aide de notre système. Les essais réalisés en simulation semblent permettre de bonnes détections. Les résultats obtenus à partir des essais réels montrent que la détection fonctionne bien à faible distance, mais pose problème pour des distances plus lointaines. Les essais réels ont également permis de confirmer que le choix du mode fisheye est plus judicieux que celui du mode distordu. Il offre, en effet, une meilleure robustesse aux imprécisions de calibration.

Malgré quelques points à améliorer, l'approche permet donc de différencier un deux-roues d'un autre véhicule, précisément grâce à l'apport de la stéréovision dont l'intérêt est donc bien démontré pour la détection des deux-roues.

Enfin, même si les résultats ne sont pas encore parfaits pour une détection en intersection, dans le cas d'une observation de deux-roues en interfile la détection est d'ores et déjà bien meilleure qu'avec un système en mono-vision, répondant ainsi en bonne partie aux attentes du projet METRAMOTO.

Conclusions et perspectives

Conclusions

Les travaux présentés ici s'inscrivaient en partie dans le cadre du projet METRAMOTO dont l'enjeu était de fournir des systèmes non intrusifs permettant de détecter les deux-roues en milieu urbain.

Aujourd'hui, le trafic de deux-roues en milieu urbain tend à s'intensifier, mais leur circulation reste plus dangereuse que celle des autres véhicules, la problématique des deux-roues ayant été, pendant longtemps, très peu prise en compte dans les décisions d'aménagements urbains. Pour pouvoir apporter des solutions nouvelles et efficaces, et s'adapter au mieux à la circulation des différentes catégories de véhicules, une meilleure connaissance de la problématique des deux-roues est donc indispensable. A l'heure actuelle, cela n'est pas réellement possible en raison de la difficulté de prise en compte de cette famille de véhicules par les systèmes d'analyse de trafic existants.

Le verrou qui, jusqu'ici, empêchait la détection des deux roues par les systèmes basés sur la vision était principalement lié aux ombres : on avait des difficultés à différencier le véhicule de son ombre portée, et cela gênait la différenciation des véhicules. La solution envisagée a été de créer un système basé sur la vision en stéréo débouchant sur la reconstruction d'images 3D de la scène, et permettant ainsi de s'affranchir des ombres, qui sont exclusivement au niveau du sol.

Nous avons fait le choix de dimensionner notre système pour pouvoir effectuer des détections sur une intersection urbaine, où les enjeux en termes de sécurité des deux-roues sont très forts. Compte-tenu de ce choix, un système en stéréovision possédant des optiques fisheye s'est révélé comme le plus adapté. Pour des raisons de stabilité du système, celui-ci étant placé en hauteur, l'idée, très peu utilisée jusque là, d'aligner les caméras selon un axe vertical est apparue fournir une solution intéressante.

L'étude des méthodes actuelles de traitement de stéréovision nous a permis de voir dans quelle mesure elles étaient adaptées à un traitement des images acquises avec notre disposition particulière de caméras. Elle a pointé les limites de ces méthodes et permit de proposer une approche innovante apportant un plus en termes de qualité de traitement et de vitesse d'exécution. Cette approche diffère des précédentes en ce qu'elle crée plu-

sieurs images intermédiaires appelées couches, ayant pour chacune une disparité nulle avec l'image de référence à une hauteur définie. Ceci a l'avantage :

- ⇒ de sélectionner plus précisément la zone de recherche
- ⇒ de ne pas effectuer de balayage sur l'image
- ⇒ de discrétiser les résultats
- ⇒ de fournir directement des résultats en termes de hauteur au dessus du niveau de la route

L'approche s'étant révélée meilleure en termes de précision et de rapidité de calcul, son choix s'impose. Le seul inconvénient de cette approche est que, du fait du nombre de couches à calculer, le traitement demande beaucoup de mémoire pour stocker les informations intermédiaires en cours de calcul. Cela n'est pas gênant pour notre application.

Tout au long de cette thèse, le simulateur Sivic a été utilisé dans le but de valider et de sélectionner les algorithmes à utiliser, aussi bien en ce qui concerne les méthodes d'extraction fond/forme, les approches de stéréovision, les méthodes de calibration, que le traitement global du process. L'énorme avantage de ce simulateur venait de la variété de scènes routières, de conditions et de scénarios qui pouvait être générés, et au fait qu'à chaque fois, les vérités terrain étaient aisément et précisément accessibles.

Les essais sur site réel n'ont pu être entièrement menés à bien en raison de problèmes matériels. Néanmoins, les premiers résultats obtenus montrent que le traitement permet bien de lever le verrou principal qui empêchait la détection des deux-roues, à savoir la suppression des ombres. Les informations obtenues par ce traitement sont également suffisamment précises pour pouvoir catégoriser les véhicules. Toutefois le traitement et la mise en œuvre sont à améliorer pour accroitre la portée du système.

PERSPECTIVES

Les essais réels ont été réalisé dans des conditions qui n'ont pas permis une calibration parfaite du système, ce qui a amené quelques imprécisions dans les résultats. Les lacunes au niveau de la mise en œuvre de cette calibration ont été pointées, et peuvent être corrigées par une procédure plus rigoureuse. Par conséquent, d'autres essais doivent être réalisés afin d'affiner ce point.

L'algorithme de traitement stéréo multi-couches à l'avantage d'être considérablement plus rapide que les autres. Il a, en particulier, été codé en langage C et optimisé pour utiliser les différents cœurs du processeur et ainsi gagner en vitesse d'exécution. A l'heure actuelle, il fonctionne avec une approche CENSUS à des vitesses de l'ordre de 1,5 images par seconde. Aucune prise en compte du GPU n'est pourtant utilisée ici. De plus, pour une très grande partie des opérations, le traitement est effectué sur toute l'image sans prendre en compte la ou les zones décrites par l'extraction fond/forme. Il paraît donc probable qu'une optimisation du codage de

cet algorithme, combinée à une augmentation de la puissance de calcul des nouvelles machines, pourrait permettre d'envisager la possibilité d'un calcul en temps réel, dont nous ne sommes déjà pas si loin.

Les résultats du traitement des séquences d'images sont disponibles sous forme de nuages de points. Un système opérationnel de suivi de véhicules a été développé au sein de l'IFSTTAR, permettant de réaliser la détection et le suivi de véhicules à partir de données mono-caméra de ce type. Ce système n'est toutefois pas conçu pour traiter la problématique des deux-roues, faute de données *ad hoc*. Il reste donc à réaliser la passerelle permettant d'intégrer les données produites par notre système stéréoscopique dans ce logiciel de suivi.

Toute la démarche présentée ici repose sur un système basé sur la vision. Or cette approche présente des limites en situation nocturne. Effectuer une analyse en fonction des heures de la journée est impossible. Un travail intéressant à effectuer serait donc d'étudier la possibilité d'utiliser des caméras thermiques ou infra-rouge pour pouvoir réaliser cette analyse sur des vidéos prises à toute heure de la journée.

Comme nous avons pu le voir, le système d'acquisition utilisé est un prototype, qu'un grand nombre de problèmes matériels n'ont pas permis d'optimiser. Par exemple, sa portée ne dépasse pas les 15 mètres. Un travail est donc à mener pour concevoir un système plus fonctionnel, plus simple à mettre en place et à calibrer, en ayant une portée accrue.

Le système de prise de vue en stéréovision et le principe de traitement sont assez originaux. Ils ont l'avantage, du fait de la disposition des caméras, d'être beaucoup plus stables que les systèmes utilisés généralement. Dans notre cas, nous les avons utilisés pour détecter les deux-roues, mais cette approche peut être utilisée pour d'autres applications, allant bien au-delà du domaine routier et nécessitant la perception en 3D d'une scène sur un champ de vision très large. Des applications sont donc possibles dans divers domaines comme en vision pour la robotique, en vidéo-surveillance, etc.

On peut également imaginer des applications du système de traitement en le transposant à une caméra placée sur un véhicule et orientée vers l'avant. La dimension permettant le traitement en stéréovision n'est alors pas l'écartement entre deux caméras, mais la différence de position de la caméra causée par le déplacement du véhicule. Les images prises entre deux instants proches peuvent donc être traitées de la même façon. Des applications d'aide à la conduite par vision embarquée, par exemple, sont alors envisageables.

Bibliographie

[1] Messelodi (S.), Modena (C.) et Cattoni (G.), « Vision-based bicycle/motorcycle classification », *Pattern recognition letters*, vol. 28, n° 13, 2007, p. 1719–1726. (Cité pages xi, 23 et 24.)

[2] Agence d'urbanisme pour le développement de l'agglomération lyonnaise, « Le vélo dans nos déplacements quotidiens », septembre 2010. (Cité page 7.)

[3] Commissariat général au développement durable, « Les coûts et les avantages des vélos en libre service », mai 2010. (Cité page 8.)

[4] Ministère des transports Néerlandais, « Le vélo aux pays-bas », 2009. (Cité page 9.)

[5] Centre d'études sur les réseaux, les transports, l'urbanisme et les constructions publiques (CERTU), « Pays-bas : Houten, la ville nouvelle favorable aux modes doux », juillet 2007. (Cité page 10.)

[6] Tien (P. M.), « L'explosion des deux roues à hcmv : un vrai défi pour les transports urbains », dans *CODATU XIII*. (Cité page 10.)

[7] Observatoire national interministériel de la sécurité routière (ONISR), « La sécurité routière en france - bilan de l'année 2010 », 2011. (Cité page 12.)

[8] Riou (D.) et Verrier (D.), « Sécurité routière et usage des deux-roues motorisés en île-de-france », 2009. (Cité pages 13 et 16.)

[9] Centre d'études sur les réseaux, les transports, l'urbanisme et les constructions publiques (CERTU), « Guide des carrefours urbains », 1999. (Cité page 16.)

[10] Centre d'études sur les réseaux, les transports, l'urbanisme et les constructions publiques (CERTU), « Fiche vélo : Les sas à vélo », août 2009. (Cité page 18.)

[11] Centre d'études sur les réseaux, les transports, l'urbanisme et les constructions publiques (CERTU), « La détection des deux-roues motorisés : quels systèmes, quels outils ? », décembre 2010. (Cité page 21.)

[12] Lin (H.) et Wei (J.), « A street scene surveillance system for moving object detection, tracking and classification », dans *Intelligent Vehicles Symposium, 2007 IEEE*, p. 1077–1082. IEEE, 2007. (Cité page 23.)

[13] Fakhfakh (N.), *Détection et localisation tridimensionnelle par stéréovision d'objets en mouvement dans des environnements complexes, Application aux passages à niveau*. Thèse de doctorat, École Centrale de Lille, 2011. (Cité page 23.)

[14] TAI (J.), TSENG (S.), LIN (C.) et SONG (K.), « Real-time image tracking for automatic traffic monitoring and enforcement applications », *Image and Vision Computing*, vol. 22, nº 6, 2004, p. 485–501. (Cité page 23.)

[15] TAI (J.) et SONG (K.), « Automatic contour initialization for image tracking of multi-lane vehicles and motorcycles », dans *Intelligent Transportation Systems, 2003. Proceedings. 2003 IEEE*, vol. 1, p. 808–813. IEEE, 2003. (Cité page 23.)

[16] RAD (R.) et JAMZAD (M.), « Real time classification and tracking of multiple vehicles in highways », *Pattern Recognition Letters*, vol. 26, nº 10, 2005, p. 1597–1607. (Cité page 23.)

[17] MESSELODI (S.), MODENA (C.) et ZANIN (M.), « A computer vision system for the detection and classification of vehicles at urban road intersections », *Pattern Analysis & Applications*, vol. 8, nº 1, 2005, p. 17–31. (Cité page 23.)

[18] COMBY (F.), CADERAS DE KERLEAU (C.) et STRAUSS (O.), « Étalonnage de caméras catadioptriques hyperboloïdes », 2005. (Cité pages 33 et 34.)

[19] MEI (C.) et RIVES (P.), « Single view point omnidirectional camera calibration from planar grids », dans *Robotics and Automation, 2007 IEEE International Conference on*, p. 3945–3950. IEEE, 2007. (Cité pages 33, 34 et 44.)

[20] SVOBODA (T.), « Central panoramic cameras design, geometry, egomotion », *PhD Theses, Center of Machine Perception, Czech Technical University in Prague*, 1999. (Cité pages 33 et 34.)

[21] BAKER (S.) et NAYAR (S.), « A theory of single-viewpoint catadioptric image formation », *International Journal of Computer Vision*, vol. 35, nº 2, 1999, p. 175–196. (Cité page 33.)

[22] NALWA (V.), « A true omnidirectional viewer ». Rapport technique, technical report, Bell Laboratories, 1996. (Cité page 34.)

[23] GHORAYEB (A.), POTELLE (A.), DEVENDEVILLE (L.) et MOUADDIB (E. M.), « Capteur omnidirectionnel optimal pour le diagnostic de la circulation dans les carrefours urbains », *Orasis 2009*, 2009. (Cité pages 34 et 35.)

[24] YAGI (Y.) et YACHIDA (M.), « Real-time generation of environmental map and obstacle avoidance using omnidirectional image sensor with conic mirror », dans *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, p. 160–165. IEEE, 1991. (Cité page 34.)

[25] BRASSART (E.), DELAHOCHE (L.), CAUCHOIS (C.) *et al.*, « Experimental results got with the omnidirectional vision sensor : Syclop », dans *Omnidirectional Vision, 2000. Proceedings. IEEE Workshop on*, p. 145–152. IEEE, 2000. (Cité page 34.)

[26] CAUCHOIS (C.), BRASSART (E.), MARHIC (B.) et DROCOURT (C.), « An absolute localization method using a synthetic panoramic image base », dans *Omnidirectional Vision, 2002. Proceedings. Third Workshop on*, p. 128–135. IEEE, 2002. (Cité page 34.)

[27] NAYAR (S. K.), « Catadioptric omnidirectional camera », dans *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, coll. « CVPR '97 », p. 482–, Washington, DC, USA, 1997. IEEE Computer Society. (Cité page 35.)

[28] GONZALEZ-BARBOSA (J.), *Vision panoramique pour la robotique mobile : Stéréovision et localisation par indexation d'images*. Thèse de doctorat, Université Paul Sabatier - Toulouse III, 2004. (Cité pages 35 et 38.)

[29] SHAH (S.) et AGGARWAL (J.), « Intrinsic parameter calibration procedure for a (high-distortion) fish-eye lens camera with distortion model and accuracy estimation* », *Pattern Recognition*, vol. 29, n° 11, 1996, p. 1775–1788. (Cité page 41.)

[30] KANNALA (J.) et BRANDT (S.), « A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses », *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, n° 8, 2006, p. 1335–1340. (Cité page 41.)

[31] SHAH (S.) et AGGARWAL (J.), « A simple calibration procedure for fish-eye (high distortion) lens camera », dans *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, p. 3422–3427. IEEE, 1994. (Cité page 41.)

[32] MICUSIK (B.) et PAJDLA (T.), « Estimation of omnidirectional camera model from epipolar geometry », dans *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 1, p. I–485. IEEE, 2003. (Cité page 41.)

[33] SCARAMUZZA (D.), MARTINELLI (A.) et SIEGWART (R.), « A flexible technique for accurate omnidirectional camera calibration and structure from motion », dans *Computer Vision Systems, 2006 ICVS'06. IEEE International Conference on*, p. 45–45. IEEE, 2006. (Cité pages 41, 44 et 102.)

[34] GEYER (C.) et DANIILIDIS (K.), « A unifying theory for central panoramic systems and practical implications », *Computer Vision ?ECCV 2000*, 2000, p. 445–461. (Cité page 41.)

[35] COURBON (J.), MEZOUAR (Y.), ECK (L.) et MARTINET (P.), « A generic fisheye camera model for robotic applications », dans *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, p. 1683–1688. IEEE, 2007. (Cité page 41.)

[36] CARON (G.) et EYNARD (D.), « Multiple camera types simultaneous stereo calibration », dans *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, p. 2933–2938. IEEE, 2011. (Cité pages 41 et 100.)

[37] YING (X.) et HU (Z.), « Can we consider central catadioptric cameras and fisheye cameras within a unified imaging model », *Computer Vision-ECCV 2004*, 2004, p. 442–455. (Cité pages 41 et 43.)

[38] ARCARA (P.), DI STEFANO (L.), MATTOCCIA (S.) *et al.*, « Perception of depth information by means of a wire-actuated haptic interface », dans *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 4, p. 3443–3448. IEEE, 2000. (Cité page 50.)

[39] STEFANO (L.), MARCHIONNI (M.) et MATTOCCIA (S.), « A fast area-based stereo matching algorithm », *Image and vision computing*, vol. 22, nº 12, 2004, p. 983–1005. (Cité page 50.)

[40] KIM (J.), LEE (K.), CHOI (B.) et LEE (S.), « A dense stereo matching using two-pass dynamic programming with generalized ground control points », dans *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 2, p. 1075–1082. IEEE, 2005. (Cité pages 50 et 55.)

[41] MAYER (H.), « Analysis of means to improve cooperative disparity estimation », *International archives of photogrammetry remote sensing and spatial information sciences*, vol. 34, nº 3/W8, 2003, p. 25–32. (Cité page 50.)

[42] KANADE (T.), YOSHIDA (A.), ODA (K.) *et al.*, « A stereo machine for video-rate dense depth mapping and its new applications », dans *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR'96, 1996 IEEE Computer Society Conference on*, p. 196–202. IEEE, 1996. (Cité page 51.)

[43] KU (J.), LEE (K.) et LEE (S.), « Multi-image matching for a general motion stereo camera model », dans *Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on*, vol. 2, p. 608–612. IEEE, 1998. (Cité page 51.)

[44] MANDUCHI (R.) et TOMASI (C.), « Distinctiveness maps for image matching », dans *Image Analysis and Processing, 1999. Proceedings. International Conference on*, p. 26–31. IEEE, 1999. (Cité page 51.)

[45] OKUTOMI (M.) et KANADE (T.), « A multiple-baseline stereo », *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 15, nº 4, 1993, p. 353–363. (Cité page 51.)

[46] FERRARI (V.), TUYTELAARS (T.) et GOOL (L.), « Wide-baseline multiple-view correspondences », dans *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 1, p. I–718. IEEE, 2003. (Cité page 52.)

[47] DI STEFANO (L.) et MATTOCCIA (S.), « Fast template matching using bounded partial correlation », *Machine Vision and Applications*, vol. 13, nº 4, 2003, p. 213–221. (Cité page 52.)

[48] TSAI (D.), LIN (C.) et CHEN (J.), « The evaluation of normalized cross correlations for defect detection », *Pattern Recognition Letters*, vol. 24, nº 15, 2003, p. 2525–2535. (Cité page 52.)

[49] SUN (C.), « A fast stereo matching method », dans *Digital Image Computing : Techniques and Applications*, p. 95–100. Citeseer, 1997. (Cité page 52.)

[50] ZHANG (Z.) et SHAN (Y.), « A progressive scheme for stereo matching », *3D Structure from Images ?SMILE 2000*, 2001, p. 68–85. (Cité page 52.)

[51] LHUILLIER (M.), QUAN (L.) *et al.*, « Reconstruction quasi-dense et modèles 3d à partir d'une séquence d'images », 2004. (Cité pages 52 et 57.)

[52] ZABIH (R.) et WOODFILL (J.), « Non-parametric local transforms for computing visual correspondence », *Computer Vision ?ECCV'94*, 1994, p. 151–158. (Cité page 53.)

[53] IBARRA-MANZANO (M. A.), ALMANZA-OJEDA (D.-L.), DEVY (M.) *et al.*, « Stereo vision algorithm implementation in fpga using census transform for effective resource optimization », dans *Proceedings of the 2009 12th Euromicro Conference on Digital System Design, Architectures, Methods and Tools*, coll. « DSD '09 », p. 799–805, Washington, DC, USA, 2009. IEEE Computer Society. (Cité page 54.)

[54] CLAUS (C.), LAIKA (A.), JIA (L.) et STECHELE (W.), « High performance fpga based optical flow calculation using the census transformation », dans *Intelligent Vehicles Symposium, 2009 IEEE*, p. 1185–1190. IEEE, 2009. (Cité page 54.)

[55] JIN (S.), CHO (J.), DAI PHAM (X.) *et al.*, « Fpga design and implementation of a real-time stereo vision system », *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 20, n° 1, 2010, p. 15–26. (Cité page 54.)

[56] PORTER (R.) et BERGMANN (N.), « A generic implementation framework for fpga based stereo matching », dans *TENCON'97. IEEE Region 10 Annual Conference. Speech and Image Technologies for Computing and Telecommunications., Proceedings of IEEE*, vol. 2, p. 461–464. Ieee, 1997. (Cité page 54.)

[57] STEIN (F.), « Efficient computation of optical flow using the census transform », *Pattern Recognition*, 2004, p. 79–86. (Cité page 54.)

[58] KONOLIGE (K.), « Small vision systems : Hardware and implementation », dans *ROBOTICS RESEARCH-INTERNATIONAL SYMPOSIUM-*, vol. 8, p. 203–212. MIT PRESS, 1998. (Cité page 54.)

[59] MURPHY (C.), LINDQUIST (D.), RYNNING (A.) *et al.*, « Low-cost stereo vision on an fpga », dans *Field-Programmable Custom Computing Machines, 2007. FCCM 2007. 15th Annual IEEE Symposium on*, p. 333–334. IEEE, 2007. (Cité page 54.)

[60] IOCCHI (L.) et KONOLIGE (K.), « A multiresolution stereo vision system for mobile robots », dans *AIIA (Italian AI Association) Workshop, Padova, Italy*, 1998. (Cité page 54.)

[61] GU (C.) et WU (L.), « Structural matching of multiresolution for stereo vision », dans *Pattern Recognition, 1990. Proceedings., 10th International Conference on*, vol. 1, p. 243–245. IEEE, 1990. (Cité page 54.)

[62] GONG (M.) et YANG (Y.), « Multi-resolution stereo matching using genetic algorithm », dans *Stereo and Multi-Baseline Vision, 2001.(SMBV 2001). Proceedings. IEEE Workshop on*, p. 21–29. IEEE, 2001. (Cité page 54.)

[63] CROUZIL (A.) et MASSIP PAILHES (L.), « Perception du relief et du mouvement par analyse d'une séquence stéréoscopique d'images », 1997. (Cité page 54.)

[64] YANG (Y.), YUILLE (A.) et LU (J.), « Local, global, and multilevel stereo matching », dans *Computer Vision and Pattern Recognition, 1993. Proceedings CVPR'93., 1993 IEEE Computer Society Conference on*, p. 274–279. IEEE, 1993. (Cité page 55.)

[65] KOSCHAN (A.), « Using perceptual attributes to obtain dense depth maps », dans *Image Analysis and Interpretation, 1996., Proceedings of the IEEE Southwest Symposium on*, p. 155–159. IEEE, 1996. (Cité page 55.)

[66] MÜHLMANN (K.), MAIER (D.), HESSER (J.) et MÄNNER (R.), « Calculating dense disparity maps from color stereo images, an efficient implementation », *International Journal of Computer Vision*, vol. 47, n° 1, 2002, p. 79–88. (Cité page 55.)

[67] GONG (M.) et YANG (Y.), « Near real-time reliable stereo matching using programmable graphics hardware », dans *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, p. 924–931. IEEE, 2005. (Cité page 55.)

[68] SZELISKI (R.) et SCHARSTEIN (D.), « Symmetric sub-pixel stereo matching », *Computer Vision ECCV 2002*, 2002, p. 657–659. (Cité page 55.)

[69] DHOND (U.) et AGGARWAL (J.), « Stereo matching in the presence of narrow occluding objects using dynamic disparity search », *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 17, n° 7, 1995, p. 719–724. (Cité page 55.)

[70] CROUZIL (A.), MASSIP-PAILHES (L.) et CASTAN (S.), « A new correlation criterion based on gradient fields similarity », dans *Pattern Recognition, 1996., Proceedings of the 13th International Conference on*, vol. 1, p. 632–636. IEEE, 1996. (Cité page 57.)

[71] HIRSCHMÜLLER (H.), INNOCENT (P.) et GARIBALDI (J.), « Real-time correlation-based stereo vision with reduced border errors », *International Journal of Computer Vision*, vol. 47, n° 1, 2002, p. 229–246. (Cité page 57.)

[72] ZHANG (Y.) et KAMBHAMETTU (C.), « Stereo matching with segmentation-based cooperation », *Computer Vision ?ECCV 2002*, 2002, p. 521–522. (Cité page 57.)

[73] GUTIERREZ (S.) et MARROQUI ?N (J.), « Robust approach for disparity estimation in stereo vision », *Image and Vision Computing*, vol. 22, n° 3, 2004, p. 183–195. (Cité page 57.)

[74] STAUFFER (C.) et GRIMSON (W. E. L.), « Adaptative background mixture models for a real-time tracking », *Conference on Computer Vision and Patern Recognition*, 1999. (Cité pages 58 et 59.)

[75] SIGARI (M. H.) et FATHY (M.), « Real-time background modeling/subtraction using two-layer codebook model », *International MultiConference of Engineers and Computer Scientists*, 2008. (Cité page 59.)

[76] KIM (K.), CHALIDABHONGSE (T. H.), HARWOOD (D.) et DAVIS (L.), « Real-time foreground-background segmentation using codebook model », *Real-time Imaging*, vol. 11(3), 2005, p. 167–256. (Cité page 59.)

[77] GOYAT (Y.), CHATEAU (T.), MALATERRE (L.) et TRASSOUDAINE (L.), « Vehicle trajectories evaluation by static video sensors », *9th IEEE International Conference on Intelligent Transportation Systems*, 2006. (Cité page 61.)

[78] GRUYER (D.), ROYERE (C.), DU LAC (N.) *et al.*, « Sivic and rtmaps, interconnected platforms for the conception and the evaluation of driving assistance systems », *World Congress and Exhibition on Intelligent Transport Systems and Services*, 2006. (Cité pages 70, 94, 120 et 131.)

[79] TRONSON (N.), GOYAT (Y.) et GRUYER (D.), « Comparaison de méthodes d'extraction fond/forme pour des scènes de circulation routière », *CORESA*, 2010. (Cité pages 73 et 77.)

[80] HRIPCSAK (G.) et ROTHSCHILD (A.), « Agreement, the f-measure, and reliability in information retrieval », *Journal of the American Medical Informatics Association*, vol. 12, nº 3, 2005, p. 296–298. (Cité page 75.)

[81] MAKHOUL (J.), KUBALA (F.), SCHWARTZ (R.) et WEISCHEDEL (R.), « Performance measures for information extraction », dans *Proceedings of DARPA Broadcast News Workshop*, p. 249–252, 1999. (Cité page 75.)

[82] HUANG (Y.), POWERS (R.) et MONTELIONE (G.), « Protein nmr recall, precision, and f-measure scores (rpf scores) : structure quality assessment measures based on information retrieval statistics », *Journal of the American Chemical Society*, vol. 127, nº 6, 2005, p. 1665–1674. (Cité page 75.)

[83] EVANS (L.) et GARIEPY (R.), *Measure theory and fine properties of functions*. CRC, 1992. (Cité page 75.)

[84] DHOME (Y.), TRONSON (N.), VACAVANT (A.) *et al.*, « A benchmark for background subtraction algorithms in monocular vision : a comparative study », dans *Image Processing Theory Tools and Applications (IPTA), 2010 2nd International Conference on*, p. 66–71. IEEE, 2010. (Cité page 77.)

[85] HERRERA (P.), PAJARES (G.), GUIJARRO (M.) *et al.*, « Combination of attributes in stereovision matching for fish-eye lenses in forest analysis », dans BLANC-TALON (J.), PHILIPS (W.), POPESCU (D.) et SCHEUNDERS (P.), éditeurs, *Advanced Concepts for Intelligent Vision Systems*, vol. 5807 (coll. *Lecture Notes in Computer Science*), p. 277–287. Springer Berlin / Heidelberg, 2009. (Cité pages 92 et 106.)

[86] GEHRIG (S.), RABE (C.) et KRUEGER (L.), « 6d vision goes fisheye for intersection assistance », dans *Computer and Robot Vision, 2008. CRV '08. Canadian Conference on*, p. 34 –41, may 2008. (Cité pages 92 et 106.)

[87] RAGOT (N.), *Conception d'un capteur de stéréovision omnidirectionnelle : architecture, étalonnage et applications à la reconstruction 3D*. Thèse de doctorat, Université de Rouen, 2009. (Cité page 92.)

[88] CARON (G.) et EYNARD (D.), « Etalonnage simultané de systèmes stéréoscopiques hybrides », dans *Congrès des jeunes chercheurs en vision par ordinateur, ORASIS*, Praz-sur-Arly, France, June 2011. (Cité page 100.)

[89] CHAMBON (S.), *Mise en correspondance stéréoscopique d'images couleur en présence d'occultations*. Thèse de doctorat, Université Paul Sabatier - Toulouse III, 2005.

[90] ABRAHAM (S.) et FÖRSTNER (W.), « Fish-eye-stereo calibration and epipolar rectification », *ISPRS Journal of photogrammetry and remote sensing*, vol. 59, nº 5, 2005, p. 278–288.

[91] GOYAT (Y.), *Estimation precise des trajectoires de vehicule par un systeme optique*. Thèse de doctorat, Université de Clermont II, 2008. (Cité page 134.)

Notations

VL	Véhicule léger
PL	Poids lourd
2RM	Deux roues motorisé
VLS	Vélo en libre service
TAD	Tourne à droite
TAG	Tourne à gauche
MOG	Mixture of Gaussian
CB2	CodeBook 2 Layers
VUM	VuMètre
VP	Vrais positifs
VN	Vrais négatifs
FP	Faux positifs
FN	Faux négatifs
SAD	Sum of Absolute Difference
ZSAD	Zero mean Sum of Absolute Difference
SSD	Sum of Squared Difference
ZSSD	Zero mean Sum of Squared Difference
NCC	Normalized Cross-Correlation
ZNCC	Zero mean Normalized Cross-Correlation

Titre Détection des deux roues par capteurs vidéo fixes

Résumé Les travaux de thèse présentés dans ce mémoire ont pour objectif la création d'un système optique permettant la détection des deux-roues en milieu urbain. Pour cela, nous avons fait le choix, à la vue des limites des systèmes actuels principalement lié aux ombres portées des véhicules, pénalisant ainsi leur détection, d'opter pour un système de prise de vue en stéréovision. L'originalité du système présenté ici réside dans le choix des optiques et la disposition des caméras. Nous utilisons en effet, deux caméras avec optique fisheye placées l'une au dessus de l'autre, alignées et orientées selon un axe vertical. Nous choisissons cette configuration pour des raisons de stabilité et de couverture de la scène. Nous étudions ensuite la calibration de ce système à partir d'un outil général de calibration de système de stéréovision, ainsi qu'à partir d'une approche développée spécifiquement et permettant de prendre en compte les particularités du système. En ce qui concerne le traitement des données pour effectuer la mise en correspondance entre les images et ainsi obtenir une carte 3D, nous proposons trois approches. Les deux premières sont largement inspirées de l'état de l'art. La troisième approche, que nous avons conçue, est assez originale et permet de cibler plus précisément la zone de recherche 3D en décomposant la scène en différentes couches correspondant à différentes hauteurs au dessus du niveau de la route. Pour pouvoir enfin détecter, classifier et suivre les véhicules, les informations 3D apportant la position et la dimension des véhicules sont intégrées dans un logiciel de suivi déjà existant.

Mots-clés stéréovision, optique fisheye, deux-roues, calibration, reconstruction 3D

Title Two-wheeled vehicles detection by static video sensors

Abstract The objective of the work presented in this thesis is the creation of an optic system able to detect the two-wheeled vehicles in urban areas. To this prospect, we choose, owing to the limits of current systems primarily due to shadows cast by vehicles, thus penalizing their detection, to opt for a stereovision system. The originality of the system presented here is the optics and cameras placement choice. We use two cameras placed one above the other, oriented and aligned along a vertical axes. We choose this configuration for stability reasons, and because it makes it possible to cover the entire scene. Then, we study the system calibration, in one hand with a generic calibration tool for stereo systems, in the other hand with an approach specifically developed to consider the characteristics of the system. With regard to data processing, to perform the matching between images and thus obtain a 3D map, we propose three approaches. The first two are largely based on the state of the art. The third one, which was designed as a part of this thesis, is quite original and can target more precisely the search area by decomposing the 3D scene in different layers corresponding to different heights above the road level. To finally be able to detect, classify and track vehicles, 3D information providing the position and size of vehicles is integrated into an existing tracking software.

Keywords stereovision, fisheye lens, two wheeled vehicles, calibration, 3D reconstruction