

N° d'ordre : 3567

THESE

En vue de l'obtention du : **DOCTORAT**

Centre de Recherche : CEDOC

Structure de Recherche : Laboratoire de Recherche en Informatique et Télécommunications

Discipline : Science de l'ingénieur

Spécialité : Informatique

Présentée et soutenue le 18/12/2021 par :

KHALD ANISSE

Détection et Suivi d'objet Dans une Séquence d'images Omnidirectionnelles

JURY

Mohammed El HAZITI	PES, Ecole Supérieur de Technologie, Université Mohammed V, Rabat	Président
Aouatif AMINE	PES, Ecole Nationale des Sciences Appliquées, Université Ibn Tofail, Kenitra	Rapporteur/Examineur
Fatima GUEROUATE	PH, Ecole Supérieur de Technologie, Université Mohammed V, Rabat	Rapporteur/Examineur
Faissal OUARDI	PH, Faculté des Sciences, Université Mohammed V, Rabat	Rapporteur/Examineur
Amina RADGUI	PH, Institut National des Postes et Télécommunications	Co-encadrante
Mohamed RZIZA	PES, Faculté des Sciences, Université Mohammed V, Rabat	Directeur de thèse

Année Universitaire : 2021/2022



Résumé

Les systèmes catadioptriques sont des systèmes actuellement omniprésents dans les applications de vidéo surveillance et la robotique grâce à leur particularité pour acquérir le maximum d'informations de l'environnement et la réduction de la mise en œuvre du matériel. Ceci engendre une implication minutieuse dans le traitement des images relatives. Pour permettre de prendre en considération la géométrie déformée des images omnidirectionnelle dans une application de suivi d'objet. Dans le cadre de ce travail, nous avons proposé plusieurs méthodes basées d'une part sur le calcul des invariants spatiaux en utilisant les moments invariants et les contours actifs, et d'autre part sur les approches statistiques à savoir les distributions gaussiennes et Von Mises-Fisher. Les résultats obtenus conviennent parfaitement à notre objectif.

L'application des méthodes tenant compte de la géométrie des images omnidirectionnelles a permis une résolution de problèmes de changement de taille et de forme de l'objet lors de son déplacement car les algorithmes utilisés se focalisent sur l'information de la région interne de l'objet et son contour.

Mots-clés : Système catadioptrique, Image Omnidirectionnelle, Suivi d'objet, Les invariants, Gaussienne, Von Mises Fisher, Les contours actifs.,

Abstract

Recently, Catadioptric systems are ubiquitous in video surveillance and robotics applications due to their particularity for our camera. Also, for reducing the use of equipment hardware. This generates careful involvement in the processing of images provided by this systems. To taking into account the distorted geometry of omnidirectional images in an object tracking application, several methods have been used to obtain performed results. In This work, we have chosen approaches based on in first hand the computing of spatial invariants using invariant moments named Zernike and Snake. In the other hand, based on statistical approaches named Gaussian and Von MisesFisher distributions. Results obtained are performed in term in many criterion and they are suitable to our aim. Constraints change of size and shape of the object during its move because the algorithms used focus on the information of the internal region of the object. object and its outline.

Key Words : Catadioptric system, Omnidirectional image, Object tracking, Invariants, Gaussian, VonMises Fisher, Active contours.



AVANT-PROPOS

Les travaux présentés dans ce mémoire sont effectués au Laboratoire de Recherche en Informatique et Télécommunications (LRIT), à la Faculté des Sciences de Rabat (FSR), sous la direction du Professeur Mohamed RZIZA et le co-encadrement de Professeur Amina RADGUI.

Ce travail s'inscrit dans le domaine de la vision omnidirectionnelle. Le sujet proposé et l'objet de ce travail intitulé **"La Détection et le suivi d'objet dans une séquence d'images omnidirectionnelles"**.



REMERCIEMENTS

Tout d'abord je tiens à remercier ALLAH le tout puissant de m'avoir donné la santé, la volonté, le courage et la patience pour mener à terme ma formation et pourvoir réaliser ce travail de recherche.

Je tiens à exprimer ma gratitude a mon directeur de thèse Mohamed RZIZA, Professeur à la Faculté des Sciences de Rabat, Responsable de département informatique d'avoir accepter de diriger mon travail de recherche , pour ses précieux conseils, pour l'intérêt constant qu'il a porté à ce travail ainsi que pour son soutien sans faille sur les plans académiques et administratifs.

Je remercie également le professeur Mohamed EL HAZITI, professeur à l'école Supérieur de Technologie de Salé de m'avoir fait l'honneur de présider mon jury de thèse.

Je tiens particulièrement à exprimer ma reconnaissance à ma co-encadrante de thèse professeur Amina RADGUI, Professeur à l'institut National des Postes et Télécommunications Rabat, pour sa disponibilité et pour son encouragement, de m'avoir fait profiter de ses compétences, la quantité d'idées, de suggestions et de conseils, qu'elle m'a prodiguée.

Je remercie aussi Mme. Aouatif AMINE, professeur à l'école nationale des sciences appliquées rabat, d'avoir accepté de rapporter ce travail et pour le temps qu'elle a consacré pour lire et évaluer mon travail.

Je tiens également à exprimer mes remerciements aux membres du jury, qui ont accepté d'évaluer mon travail de thèse. Merci à Mme Fatima GUEROUATE, professeur à l'école Supérieur de Technologie de Salé d'avoir accepte de rapporter mon travail et aussi pour ces remarques.

Un grand merci à Mr. OUARDI Faissal, professeur au sein de la faculté des sciences de rabat pour avoir accepter de rapporter mon travail. Ces conseils, son savoir faire qui m'ont beaucoup aidé dans l'organisation du manuscrit.

A l'issue de la rédaction de cette recherche, je suis convaincue que la thèse est loin d'être un travail solitaire. En effet, je n'aurais jamais pu réaliser ce travail doctoral sans le soutien d'un grand nombre de personnes dont la generosite, la bonne humeur. Je souligne le soutien amical et chaleureux de tous les doctorants du Laboratoire LRIT, qui ont croise ma route durant ce parcours doctoral et, plus particulièrement, à tous ceux qui m'ont soutenu et encourage. Un merci special à mes amis qui ont toujours ete presents lorsque j'avais besoin d'eux. Merci a mes chers amis Assouli Jaouad, Abdelali SAADI, Imad IALA, EL Arbi EL Alaouy et mounir OMARI pour les

bons moments qu'on a passés durant plus que 10 ans, leurs aides et leurs encouragements. Un grand merci à ma chère Halima JEQUAL pour les moments agréables que nous avons partagés, sa bonne ambiance qui m'a fait oublier les moments difficiles. Mes pensées vont aussi à mon fils Abderrahman de 2 ans, pour la joie et l'amour qui a apporté à ma petite famille qui m'ont toujours motivé et encouragé pour avancer.

Enfin, mais pas pour autant moins important à mes yeux, je voudrais témoigner tout mon amour et ma reconnaissance à ma chère famille, à qui je dédie cette thèse. Merci à mes parents, mes sœurs de m'avoir soutenu moralement et financièrement et sans eux je n'en serais pas là. Merci pour leur présence dans ma vie, pour leurs conseils, leurs encouragements, leur patience et leur soutien plein et inconditionnel, même dans les moments les plus difficiles.



RÉSUMÉ

Les systèmes catadioptriques sont des systèmes actuellement omniprésents dans les applications de vidéo surveillance et la robotique grâce à leur particularité pour acquérir le maximum d'informations de l'environnement et la réduction de la mise en œuvre du matériel. Ceci engendre une implication minutieuse dans le traitement des images relatives. Pour permettre de prendre en considération la géométrie déformée des images omnidirectionnelle dans une application de suivi d'objet.

Dans le cadre de ce travail, nous avons proposé plusieurs méthodes basées d'une part sur le calcul des invariants spatiaux en utilisant les moments invariants et les contours actifs, et d'autre part sur les approches statistiques à savoir les distributions gaussiennes et Von Mises Fisher. Les résultats obtenus conviennent parfaitement à notre objectif.

L'application des méthodes tenant compte de la géométrie des images omnidirectionnelles a permis une résolution de problèmes de changement de taille et de forme de l'objet lors de son déplacement car les algorithmes utilisés se focalisent sur l'information de la région interne de l'objet et son contour.

Mots-clés : Système catadioptrique, Image Omnidirectionnelle, Suivi d'objet, Les invariants, Gaussienne, VonMises Fisher, Les contours actifs.



ABSTRACT

Recently, Catadioptric systems are ubiquitous in video surveillance and robotics applications due to their particularity for our camera. Also, for reducing the use of equipment hardware. This generates careful involvement in the processing of images provided by this systems. To taking into account the distorted geometry of omnidirectional images in an object tracking application, several methods have been used to obtain performed results.

In This work, we have chosen approaches based on in first hand the computing of spatial invariants using invariant moments named Zernike and Snake. In the other hand, based on statistical approaches named Gaussian and Von MisesFisher distributions. Results obtained are performed in term in many criterion and they are suitable to our aim.

Constraints change of size and shape of the object during its move because the algorithms used focus on the information of the internal region of the object. object and its outline.

Keywords : Catadioptric system, Omnidirectional image, Object tracking, Invariants, Gaussian, VonMises Fisher, Active contours.

INTRODUCTION GÉNÉRALE

Contexte générale

Pour commencer à qualifier un système de vision d'intelligent, il lui faut une quantité d'informations importante de l'environnement qui l'entoure et un algorithme puissant de traitement de cette information. L'un des systèmes d'observation les plus efficaces est un système avec un champ de vision large. Ce système doit collecter le maximum d'informations de l'environnement et il sera nécessaire de lui associer des capacités de traitement de l'information matérielles et logicielles considérables. Ces moyens de traitement doivent être omniprésents dans plusieurs domaines comme principalement la vidéo surveillance, la robotique avec un coût relativement faible. Actuellement, les systèmes pouvant prendre place dans ce contexte sont les systèmes catadioptriques qui produisent les images dites omnidirectionnelles avec un champs de vision très large.

Notre contexte général est la vision par ordinateur et plus précisément celle qui se base sur le système catadioptrique. Ce système est assez particulier car il est composé en plus de la caméra perspective ordinaire d'un miroir de différentes formes. Ce système basé sur des miroirs permet d'obtenir un objectif avec un montage plus léger et moins long que les systèmes omnidirectionnelle classiques (fig. 1). Ce Système permet d'obtenir une image de 360 degré. Ces images présentent des distorsions radiales dont il faut tenir compte.

FIGURE 1 – Un Système Catadioptrique



En général, les systèmes catadioptriques est appliqué principalement dans les trois domaines suivants :

- La navigation autonome du robot mobile.
- La reconstruction de scène en 3D.
- La surveillance "intelligente" et suivi.

Spécifiquement, l'objectif de cette thèse est de concevoir des algorithmes pour la détection et le suivi d'objets et de personnes pour une application de surveillance.

Problématique

Les caméras conventionnelles -perspectives- ont une ouverture de champ de vision relativement étroit. Généralement, le champ de vision est de l'ordre de 40° et 62° et va jusqu'à 84° et 120° pour les caméras grand angle. A cause de cette contrainte, on ne peut pas obtenir une vision globale de l'environnement dans une seule image par un système classique. Cette contrainte est importante et intéressante et fait le sujet de cette thèse.

Afin de compenser ces défauts, plusieurs systèmes ont été proposés :

- Les caméras panoramiques.
- Les systèmes à caméras multiples.
- Les caméras Fish-eye.
- Les caméras catadioptriques.

En général, la vision panoramique est proposée afin d'obtenir un champ de vision de 360° en horizontal et de plus de 180° en vertical. Avec une caméra rotative, on obtient une image panoramique par la rotation d'une caméra conventionnelle ou d'une caméra à balayage linéaire. Cette solution permet d'obtenir une bonne résolution d'image panoramique mais demande un montage complexe et une synchronisation délicate.

Avec un système de caméras multiples incluant deux ou plusieurs caméras synchronisées, on capture théoriquement le maximum de la scène observée. Les deux problèmes qui se présentent dans ce type de système est la redondance et l'absence de certaines parties de la scène.

Une caméra fish-eye est une caméra spéciale ayant une distance focale très courte, permettant d'obtenir directement un grand champ de vision, pouvant atteindre 180° ou plus. Mais ce type d'objectif produit des images de géométrie déformée et non mesurable. Ce qui rend l'utilisation des caméras Fish-eye dans la surveillance un peu délicate.

Enfin, la caméra catadioptrique permet d'obtenir une image panoramique en utilisant un seul capteur et par une seule capture. Cette caméra n'a aucun problème de synchronisation même si le montage peut s'avérer délicat.

Les images qui proviennent d'un système catadioptrique ont une géométrie déformée. Cette déformation est due à la forme du miroir utilisé pour la formation des images. Cette dernière se base sur la projection sur le plan 2D via le miroir qui se fait en utilisant les paramètres de la caméra en question.

Dans le domaine de la vision catadioptrique pour des applications de surveillance, Le processus de suivi d'objet et de personnes rencontre plusieurs problématiques fortement liées : le problème lié à la difficulté de la localisation et le suivi d'objet. Et le problème lié au traitement des images omnidirectionnelles provenant de ce système.



FIGURE 2 – Les images Omnidirectionnelles.

Dans une application de suivi, le problème de localisation est lié directement au mouvement de l'objet. C'est à dire qu'il faut vérifier à tous point de l'image que notre objet bouge de la même manière. Dans les images omnidirectionnelles, il est impossible de trouver une position de la caméra qui satisfait le principe de l'invariance par translation. Et ceux ci est dû à la géométrie du miroir qui donne des distorsions importantes et une résolution non-homogène dans l'image. Ces distorsions sont beaucoup plus intenses dans la région centrale de l'image (la région la plus proche de la caméra) et dans les périphéries. Cette déformation entraine une perte de données et l'application des traitements classiques ne permet pas de donner une grande performance surtout les traitements de base comme la détection de contour et la segmentation.

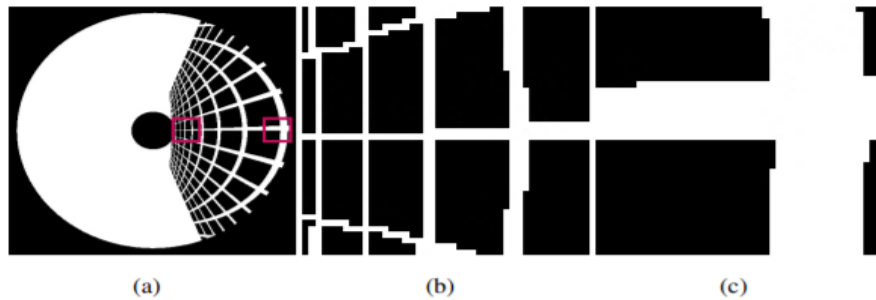


FIGURE 3 – (a) Image omnidirectionnelle d'une grille, (b) Zoom sur la zone centrale de l'image, (c) Zoom sur la zone périphérique de l'image.

Un traitement classique, quel qu'il soit, n'aura pas le même effet au centre et à la périphérie de l'image. En effet, la géométrie du miroir de révolution du système d'acquisition introduit des distorsions importantes dans l'image et crée une concentration de l'information au centre de l'image. Ce phénomène est mis en évidence dans la figure 3. Ce constat soulève un problème : Si les caméras omnidirectionnelles ont réussi à augmenter le champ de vision des caméras classiques, cela s'est fait au détriment de la simplicité des traitements.

Contributions

Dans un système catadioptrique, les scientifiques ont essayé de trouver une approche de traitement convenable pour les images omnidirectionnelles. C'est à dire, un traitement qui respecte la géométrie déformée des images et adapte le type de traitement sur chaque point de l'image. Par conséquent, certains ont traité les images en se basant sur la forme du miroir. D'autres se sont basés sur la projection des opérateurs du traitements sur des espaces équivalents.

A l'instar de ces chercheurs, la déformation existante dans les images qui proviennent d'un système catadioptrique nous a poussé à en tenir compte pour la localisation et le suivi d'objet dans les images omnidirectionnelles.

Dans ce cadre, nous avons proposé :

Le système de vision : Notre choix est évidemment le système catadioptrique avec un miroir parabolique. Ce choix est pris grâce à plusieurs raisons : i) Le champ de vision très large qui nous permet de ne pas perdre l'objet qui se déplace dans la scène environnante de la caméra. ii) Le traitement de cette quantité d'informations importante qui va se faire en lot et sans changement de perspective.

Le mouvement de l'objet dans un système catadioptrique est déterminé en fonction de sa distance de la caméra et de son déplacement autour. Ce qui entraîne un changement de taille et de forme.

Adaptation de traitement : Pour résoudre le problème de changement de taille et de forme d'objet, nous avons adopté deux approches : une première approche consiste à travailler sur la surface d'une demi sphère comme espace de traitement et de présentation des résultats. Dans cette approche nous allons adapter tous les algorithmes utilisés selon la géométrie sphérique et en utilisant les coordonnées sphériques.

La deuxième approche est basée sur une projection de la scène environnante sur la surface du miroir suivi d'une autre sur le plan catadioptrique 2D de l'image. Cette projection sur le plan 2D de l'image se fait en utilisant les paramètres de la caméra utilisés et calculés selon un modèle unifié de projection dites central.

Méthodes de localisation et suivi d'objet : Dans cette thèse, nous avons choisis plusieurs méthodes basées sur la contrainte de changement de taille et de forme lors du déplacement de l'objet. Pour cela, nous avons choisis les méthodes qui utilisent l'invariance par translation et rotation. Parmi ces méthodes, les contours actifs, qui se basent sur la minimisation d'énergie pour le suivi du contour de l'objet en mouvement. Nous avons adapté les contours actifs mathématiquement en utilisant les coordonnées sphériques.

Les invariants projectives sont appliqués ensuite pour contourner le problème de changement de forme et de taille. Les moments de ZERNIKE sont les plus utilisés dans ce domaine grâce à leur invariance à l'échelle et à l'orientation. Dans notre contexte, ces moments sont les plus convenables à utiliser dans une application de suivi d'objet dans un système catadioptrique car les moments calculés dépendent seulement de la région interne de l'objet. Pour déterminer cette région nous avons utilisé un voisinage sphérique pour chaque point de l'image. La taille du voisinage change selon la position du point dans l'image en fonction des angles sphériques θ et ϕ .

Ensuite, nous avons utilisé le même principe du voisinage sphérique combiné à des méthodes statistiques qui ne dépendent pas de la géométrie mais plutôt de la zone interne de l'objet. Ces méthodes sont caractérisées par une estimation des paramètres d'une distribution.

La distribution utilisée dans notre cas est la distribution gaussienne généralisée. Chaque région où se trouve notre objet est modélisée par un doublet (α, β) . L'objectif est de minimiser la distance entre les distributions pour déterminer la nouvelle position de notre objet.

Enfin, et comme la méthode statistique a donné des bons résultats, nous avons proposé une deuxième méthode statistique basée sur la *distribution de VonMises*. Cette distribution a pour paramètres κ et μ qui vont être estimés. La distribution de Von Mises a pour entrées des angles θ du filtre de gabor.

Les contributions proposées ont données des bons résultats en se basant sur les critère d'évaluations. Dans une application de suivi d'objet, nous avons appliqué les méthodes suivantes :

- Les contours actifs sphériques
- La distribution gaussienne généralisée
- La distribution de VonMises.
- Les moments de ZERNIKE.

Les critères d'évaluation que nous avons appliqué sont :

- Peak signal-to-noise ratio (PSNR)
- receiver operating characteristic (ROC)
- SOE Spatial Overlapping Estimation

Premièrement, Les résultats donnés par la méthodes des contours actifs ont été évalués par les critères : le PSNR et le ROC. Ces résultats ont été comparés avec la méthode classique non adaptée. Deuxièmement, en utilisant les distribution statistiques gaussienne et de VonMises, nous avons évalué les résultats en utilisant PSNR, ROC et le SOE. La comparaison a été effectuée avec la méthode de Meanshift adaptée sur la sphère. Finalement , ces critères ont été utilisés dans le cas de l'application des moment de ZERNIKE.

Plan du mémoire

Dans le premier chapitre, nous avons défini la vision omnidirectionnelle en illustrant les différents systèmes d'acquisition pour la formation des images. Les limites sont la raison de notre choix. par la suite, nous présentons les approches de traitement existantes dans le contexte de suivi d'objet. Finalement, nous expliquons la nécessité d'adapter le traitement classique à la géométrie particulière des images omnidirectionnelles.

Dans le deuxième chapitre, nous présentons un état de l'art des méthodes de suivi d'objet dans les images perspectives et aussi omnidirectionnelles. Ensuite, une classification des méthodes a été faite selon une représentation de l'objet dans l'images. Et à la fin, une étude comparative entre ces méthodes.

Le troisième chapitre explique l'utilisation de la méthode des contours actifs pour réaliser un suivi sphérique de l'objet. Dans cette contribution, nous avons utilisé une projection et représentation sphérique. Nous avons adapté tous les opérateurs mathématiques dans l'espace de la sphère.

Dans le quatrième chapitre, le suivi réalisé se base sur une modélisation statistique des blocs de l'image omnidirectionnelle en utilisant l'estimation des paramètres de la distribution. C'est un suivi de modèle. Dans cette thèse, nous avons utilisé la distribution gaussienne et la distribution de VonMises.

Le cinquième chapitre présente un état de l'art des types de descripteurs utilisés. Ensuite, nous avons réalisé une modélisation des images en utilisant les moments de ZERNIKE.

LA VISION OMNIDIRECTIONNELLE

1.1 Introduction

Ce chapitre présente des généralités sur le pilier de la formation des images omnidirectionnelles et de leurs systèmes d'acquisition en général, ainsi que les façons dont ces images ont été traitées avec des exemples concrets existants dans la littérature.

Nous allons souligner aussi les avantages et les inconvénients de chaque système d'acquisition. Nous invoquerons les approches de traitement des images omnidirectionnelles. A la fin de ce chapitre, nous allons présenter la nécessité de réaliser des adaptations des traitements directs pour remédier au problème des déformations existantes dans les images omnidirectionnelles.

Les ingénieurs, les scientifiques s'inspirent de la nature pour concevoir des systèmes de vision performants. La raison en est simple. La nature, pendant des milliards d'années d'essais et d'erreurs, a produit des solutions efficaces à des problèmes complexes du monde réel. Cette approche de la sélection des phénomènes naturels signifie que les échecs et l'inefficacité ne sont pas tolérés dans les systèmes naturels contrairement à la plupart des technologies conçues par les humains.

Lorsque les scientifiques ont voulu construire un appareil pour obtenir un champ de vision très large, ils ont commencé par poser la question : Y a-t-il des systèmes existants avec de telles capacités ? La réponse est oui. En fait, la nature offre un large choix d'exemples - en particulier dans le royaume des insectes (fourmis, araignées, papillons domestiques, etc). Par exemple, le système de vision des insectes équipés d'un oil composé est considéré comme un capteur omnidirectionnel (voir la figure 1.1).



FIGURE 1.1 – Oils composés des insectes

Les yeux composés se trouvent parmi les arthropathies et sont composés de nombreuses facettes simples, qui, selon les détails de l'anatomie, peuvent donner soit une seule image, soit des images multiples, par oeil. Chaque capteur possède son propre objectif et une cellule photosensible. Certains yeux ont jusqu'à 28 000 capteurs de ce genre, qui sont disposés de façon hexagonale, et qui peuvent donner un champ de vision complet de 360°. Par exemple, Grossberg et Nayar dans [105] comme montré dans la figure 1.2 ont inspiré la structure et le comportement de leur capteur de l'œil de libellule qui présente un système de vision complexe.

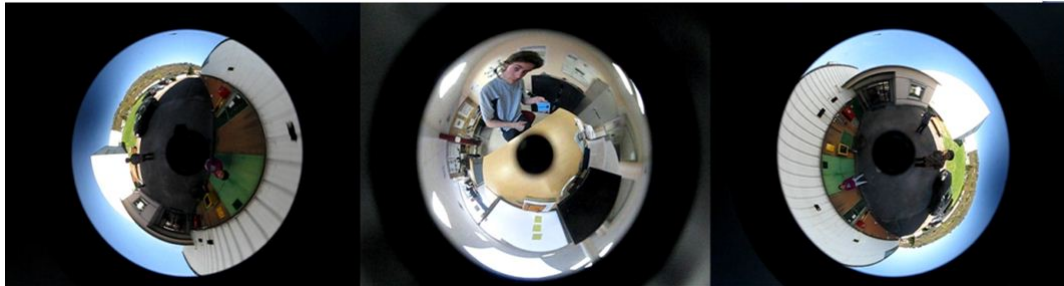


FIGURE 1.2 – Exemples image omnidirectionnelles

En science, le terme vision omnidirectionnelle se réfère à des capteurs de vision avec un très grand champ de vision (capteur avec un champ de vision horizontal de 360 degrés (figure 1.3).

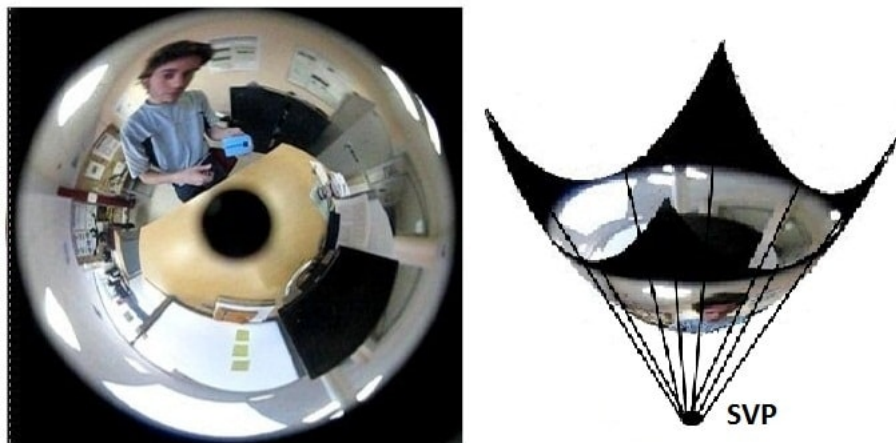


FIGURE 1.3 – Une vision omnidirectionnelle

Le champ de vision vertical se situe habituellement entre 60 et 150 degrés). La vision omnidirectionnelle implique la capture et l'interprétation automatique des images qui représentent habituellement des panoramas complets (panoramas horizontaux) de l'environnement. Parfois, le champ de vision peut être aligné verticalement ou dans n'importe quel autre plan, habituellement au moyen du déploiement d'un miroir à symétrie circulaire aligné dans la direction requise. Les champs de vision hémisphériques (360 degrés par 180 degrés) sont également utilisés, généralement en déployant une lentille de poisson. Enfin, des vues séparées peuvent être combinées, formant ainsi la projection sphérique complète idéale (Figure.1.4).

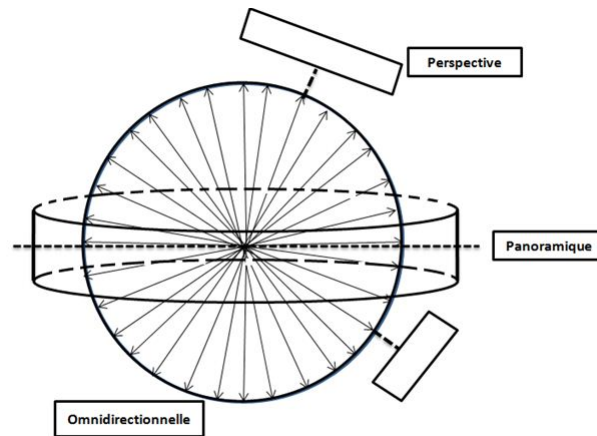


FIGURE 1.4 – Vision Omnidirectionnelle

1.2 Systèmes d'acquisition

1.2.1 Caméras panoramiques

Les caméras panoramiques sont des capteurs permettant de réaliser des vues panoramiques en perspective en utilisant uniquement une caméra unique. La plupart de ces capteurs déposent une caméra autour de l'axe vertical passant par le point focal de la caméra.

Ces caméras offrent des images panoramiques à haute définition. Cependant, un étalonnage et une synchronisation très précis des mouvements de la caméra dans le cas perspective sont nécessaires. Mais les capteurs panoramiques n'ont pas cette exigence. Au lieu de tourner autour de point focal, ils tournent autour d'un axe vertical à une certaine distance de ce dernier. Des images panoramiques à grande surface sont obtenues en faisant correspondre les vues de l'objet à partir de différentes positions (Figure 1.5).



FIGURE 1.5 – Caméra panoramique

Le principal inconvénient est que cette technique est très lente et ne convient pas à une utilisation dans un environnement dynamique. Dans tel circonstance, nous avons besoin de caméras capables

de capturer une vue globale à un seul coup afin d'avoir tous les objets en mouvement.

1.2.2 Caméras composées

Les caméras composites utilisent plusieurs caméras pour prendre des photos à un seul cliché de différentes directions. Puis, les associer pour créer une vue globale de l'environnement (figure 1.6).

Les avantages de ces caméras sont la haute résolution qui peut être obtenue et la possibilité d'attraper des images de différentes directions en même temps. Cependant, l'inconvénient est la complexité du système et la nécessité d'un processus d'étalonnage précis.

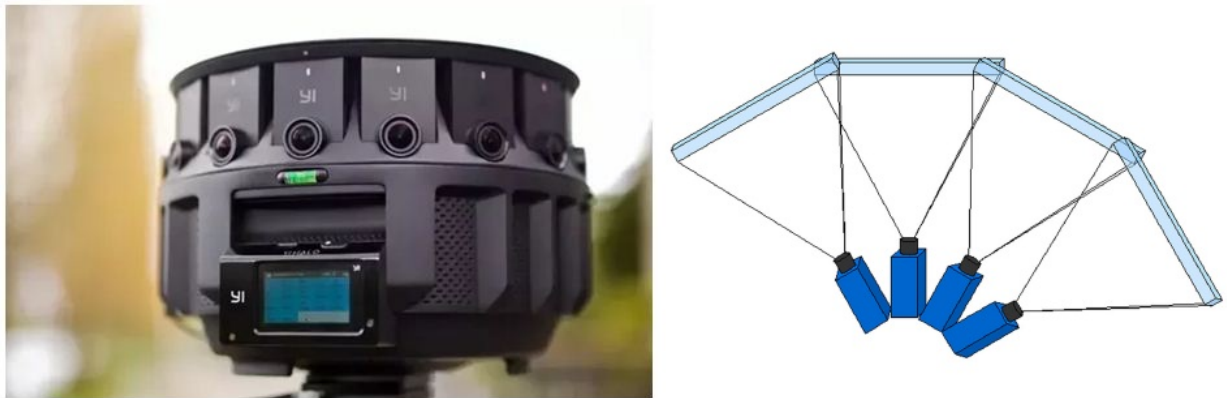


FIGURE 1.6 – Les caméras composées

Nanda et Cutler dans [84] ont proposé un système de caméras multiples de haute résolution, simple à réaliser et de coût très raisonnable permettant de créer des images panoramiques en temps réel. L'utilisation d'une unité centrale de traitement d'images (pour modifier le contraste, la luminosité, la balance des couleurs, etc.) permet d'améliorer la qualité de l'image.

Baker, Fermuller, Aloimonos et Pless dans [8] ont présenté un système de six caméras haute résolution qui prélèvent différentes parties de la sphère visuelle. Les quatre chercheurs ont développé des algorithmes pour l'étalonnage du système basés sur la relation entre les projections sur l'image de lignes parallèles dans l'espace et la rotation entre les caméras.

1.2.3 Caméras de lentilles spéciales.

Les caméras de lentilles spéciales (également appelées caméras à œil de poisson) sont des systèmes combinant une lentille de poisson et une caméra conventionnelle. Ils peuvent acquérir une vue presque hémisphérique (voir la figure 1.7). Leur principal avantage en ce qui concerne les capteurs de multiple caméras est qu'ils ne présentent pas une zone morte.

L'inconvénient est que la résolution des images est bonne au centre mais très faible à la périphérie. Ce n'est pas bon pour la navigation par robot, où les objets à localiser se trouvent sur le sol et apparaissent à l'horizon ou au-dessous. En d'autres termes, la résolution est très bonne au plafond mais pauvre à l'horizon.



FIGURE 1.7 – Caméra fisheye

Les lentilles fisheye ont été utilisées par Bakstein et Pajdla [80] pour acquérir des images $360^\circ \times 360^\circ$ à l'aide d'une technique de mosaïque. Cependant les images sont déformées : contrairement aux objectifs classiques, la distorsion n'est pas corrigée. Toutes les lignes sont fortement courbées à l'exception de celles passant par le centre de l'image. Pour retrouver les angles de ces images, elles doivent être projetées sur un écran sphérique. L'image perspective ne peut être reconstituée, ce qui rend l'analyse de ces images relativement difficile, principalement sur la périphérie, où la résolution est faible. De plus, ces lentilles sont complexes à réaliser et par conséquent très onéreuses.

1.2.4 Caméras catadioptriques

Les caméras catadioptriques ou les caméras à miroir convexe sont les plus utilisés en robotique pour obtenir des images omnidirectionnelles. Le capteur est composé d'une caméra perspective dirigée vers le haut du sommet d'un miroir convexe (voir la figure 1.8). L'axe optique de la caméra et l'axe géométrique du miroir sont alignés.

Ce système est généralement monté sur un robot mobile. Des différentes formes de miroirs peuvent être utilisées. Chaque forme présente des différentes propriétés qu'il faut tenir compte lors du choix du miroir pour une tâche particulière.

Parfois, les caméras catadioptriques ont tendance à être grandes par rapport aux caméras classiques. Ceci est dû au fait que la capture d'un champ de vision large exige que l'objectif et le miroir soient bien séparés l'un de l'autre. Pour surmonter cette limitation, des caméras catadioptriques pliées (caméras combinant plusieurs miroirs et un ensemble de lentilles) ont été développées. Ils utilisent la méthode de pliage optique pour plier le chemin optique entre le miroir courbé et le système de lentille. Le pliage avec un miroir courbé crée un pli de 180 degrés et peut réduire les effets optiques indésirables.

Les caméras catadioptriques sont des capteurs qui captent la lumière de toutes les directions couvrant une sphère complète. Dans la pratique, cependant, la plupart des caméras omnidirectionnelles ne couvrent que approximativement une demi-sphère, ou les 360 degrés complets le long de l'équateur de la sphère, mais excluent les pôles de la sphère.

Les caméras catadioptriques peuvent être centrales ou non centrales. Une caméra est centrale ou à point de vue unique, si tous les rayons se croisent en un seul point. Les rayons peuvent être



FIGURE 1.8 – Caméra à miroir convexe

complètement indépendants pour une caméra arbitraire, en particulier, ils ne doivent pas se croiser dans un seul point de vue. La raison pour laquelle un seul point de vue est si souhaitable est que les modèles de projection et les algorithmes deviennent simples.

Dans leur article ■ Single Viewpoint Catadioptric Cameras ■ [10] Nayar et Baker ont étudié trois critères essentiels pour les capteurs catadioptriques vérifiant la contrainte du point de vue unique : la forme des miroirs, la résolution des caméras et la position du point de vue unique. De plus, cette propriété permet de générer des images de perspective pure. Les caméras omnidirectionnelles qui utilisent uniquement l'effet de réfraction de l'objectif pour infléchir la lumière sont appelées caméras dioptriques¹ Les caméras qui utilisent les effets combinés de la réflexion d'un miroir et la réfraction d'une lentille sont appelées caméras catadioptriques.

Les caméras catadioptriques peuvent être regroupées en deux catégories : caméras qui utilisent une lentille spéciale et des caméras qui utilisent un miroir convexe et un ensemble de lentilles. Une question se pose : quel miroir utiliser ? La réponse est complexe. Chaque type de miroir a différentes caractéristiques, e.g. champ de vision, résolution spatiale.

Miroir plan

D'après les lois de Descartes, si un rayon lumineux rencontre un miroir plan, il est réfléchi et reste dans le plan d'incidence. L'angle d'incidence est égal à l'angle de réflexion. Le point de vue unique existe. Il est sur la bissectrice perpendiculaire à la droite reliant le point focal et son image virtuelle²(figure 1.9).

Zeng a proposé un système de mosaïque d'images vidéo basé sur un miroir planaire avec haute précision de conception. Ce capteur vérifie la contrainte du point de vue unique qui sera ajustée. Les distorsions existantes sont corrigées par une projection cylindrique et rassemblent des parties de l'image par une mise en correspondance³(Figure 1.10).

Miroir hyperbolique

Le miroir hyperbolique a deux points focaux. Tout rayon incident dont le prolongement passe par le foyer V d'une branche de l'hyperbole sera réfléchi en direction de l'autre foyer P . Le point

1. La Dioptrie est l'étude de la réfraction de la lumière par des lentilles. Les télescopes qui créent leur image avec une lentille convexe (réfracteurs) sont considérés comme des télescopes "dioptriques"..

2. Source de la figure : [51]

3. Source de la figure : [124]

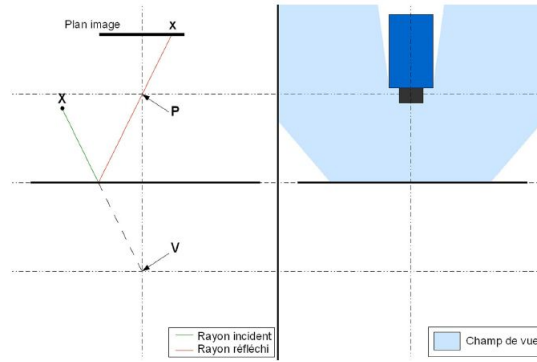


FIGURE 1.9 – Miroir plan

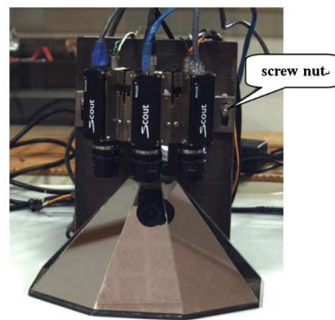


FIGURE 1.10 – Un système catadioptrique basé sur 3 miroirs plans trapezoid , avec 3 cameras de giga-ethernet

de vue unique se trouve en V . La réalisation des capteurs hyperboliques nécessite une coïncidence entre le centre optique P de la caméra et le deuxième foyer de l'hyperbole⁴(figure 1.11)

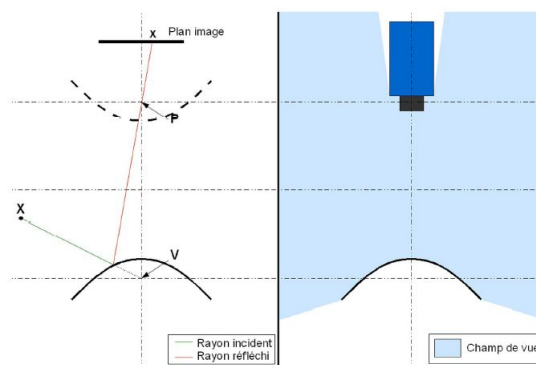


FIGURE 1.11 – Miroir hyperbolique

Ce type de miroir a été utilisé par Grassi et Okamoto dans [51] qui ont fabriqué à l'aide d'une machine CNC ultra-précise. Aussi un logiciel a été développé pour créer des images panoramiques et perspectives à partir de l'image acquise par le système.⁵(Figure 1.12).

4. Source de la figure : [51]

5. Source de la figure : [51]

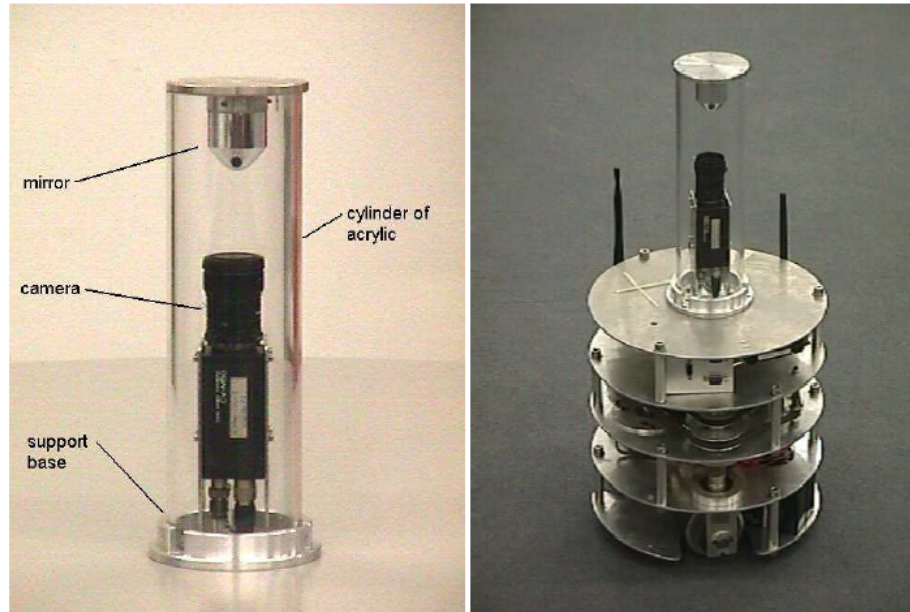


FIGURE 1.12 – Gauche : prototype du système de vision Omnidirectionnelle avec miroir hyperbolique. Droite : robot mobile avec ce système.

Miroir parabolique

Le miroir parabolique permet de contourner la contrainte de l'alignement des foyers lors de la fabrication. Une projection orthographique est réalisée : les points sont projetés suivant une direction déterminée sur un plan situé à une distance f de l'origine (la projection orthographique est parfois interprétée comme étant une projection perspective dans laquelle le point de perspective est rejeté à l'infini)⁶(figure 1.13)

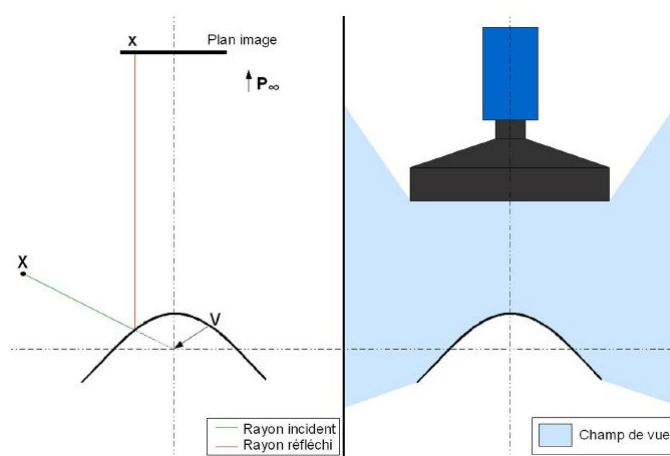


FIGURE 1.13 – Miroir parabolique

Nayar dans [[10],[9]] a réalisé une étude sur un capteur omnidirectionnel composé d'un miroir parabolique et d'une caméra perspective munie d'une lentille télécentrique. Ces lentilles spéciales

6. Source de la figure : [51]

permettent de sélectionner le faisceau lumineux des rayons parallèles à l'axe optique de la lentille.

Miroir conique

Un miroir conique associé à une caméra perspective satisfait la contrainte du point de vue unique si la focale de la caméra coïncide avec le sommet du cône⁷(figure 1.14).

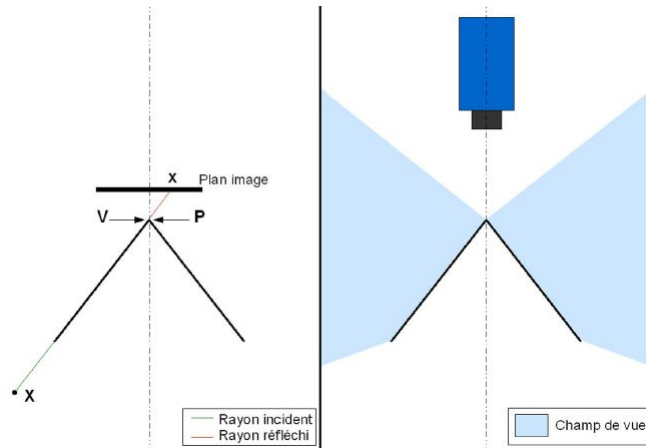


FIGURE 1.14 – Miroir conique

Une étude faite par Lin et Bajcsy [68] a montré qu'il est néanmoins possible sous certaines conditions de considérer ce capteur comme ayant un point de vue unique (une distance focale courte peut être approchée avec un point de vue unique).

Spacek dans a introduit le concept de point de vue dimensionnel séparable, grâce auquel un pseudo image perspective peut être reconstruite à partir d'une image conique.

Miroir sphérique

Le miroir sphérique, comme le miroir conique, ne peut être employé pour étendre le champ de vision : si le point de vue unique et le point focal de la caméra sont confondus, sur l'image acquise, seule la caméra serait visible (l'observateur ne peut voir que lui-même). Cependant, ces miroirs sont utilisés dans plusieurs applications de robotique et de vidéo surveillance (figure 1.15).

Un miroir sphérique est utilisé par Winters et Santos-Victor pour déterminer la position du robot. Matsui, Asoh et Thompson dans [23] ont montré ce type de capteur à l'avant d'un robot. L'image obtenue est dépliée et une mesure de corrélation est effectuée sur une base d'images de référence pour en déduire la localisation du robot.

Ishiguro [58] a effectué une synthèse pour comparer ces miroirs, avec leurs avantages et leurs inconvénients respectifs. Les miroirs hyperboliques et paraboliques qui sont SVP sont cependant difficiles à fabriquer et calibrer, et, de plus, pour le miroir parabolique, il faut utiliser une caméra télécentrique qui coûte relativement cher, et surtout qui est encombrante et lourde. En revanche, le miroir sphérique et le miroir conique sont plus faciles à fabriquer et sont moins chers, mais ne sont pas SVP. Une comparaison entre ces quatre miroirs est donnée dans le Tableau.1.1

7. Source de la figure : [51]

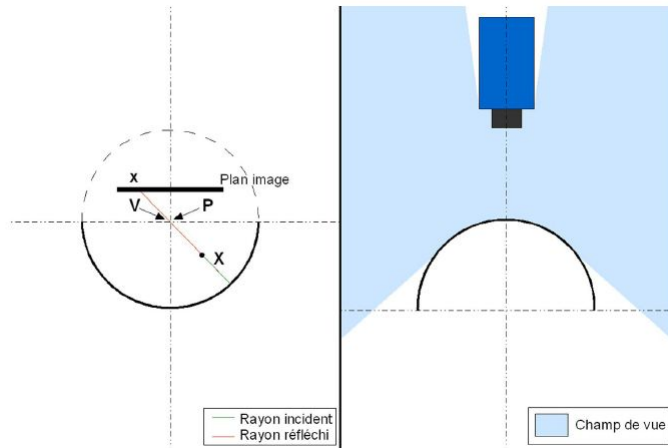


FIGURE 1.15 – Miroir sphérique

Type	Cout	Dis. focale	Champs de vis	SVP	Lentille
MH	élevé	long	$-45^\circ - -90^\circ$	oui	normale
MP	élevé	court	$-45^\circ - -90^\circ$	oui	télécentrique
MC	bas	court	$-10^\circ - -90^\circ$	non	normale
MS	bas	long	$-45^\circ - -45^\circ$	non	normale

TABLE 1.1 – Comparaison entre miroirs

1.3 Calibrage des systèmes catadioptriques

Dans cette section, nous présentons les méthodes existantes destinées à calibrer tout système omnidirectionnel central en citant les avantages et les inconvénients de chacun. Tout d’abord, nous présentons une classification de ces méthodes présentant les caractéristiques les plus pertinentes de chaque méthode particulière. Ensuite, nous passons à l’évaluation de la précision d’étalonnage considérant particulièrement la reconstruction métrique 3D. Le nombre de méthodes d’étalonnage de système catadioptrique central a augmenté ces dernières années. Ces méthodes diffèrent selon le modèle de la caméra : si le système sépare l’étalonnage de la caméra et l’estimation des paramètres du miroir [104], [83] ou s’il est considéré comme un système compact. La plupart des méthodes se réfèrent sur le modèle compact et utilisent soit des vues d’un modèle de test 2D ([99], [62], [77], [34], [42] et [76]), d’un modèle de test 3D ([89], [15]), d’un modèle cylindrique [106] et d’autres utilisent des images contenant des lignes caractéristiques ([46], [13], [120], [119], [25] et [118]). D’autres approches sont basées sur l’auto-calibration [100], [80], [93] et [40]. Outre la performance, l’ergonomie, la simplicité d’utilisation, l’algorithme et le type du modèle de test sont à considérer, car ils sont des éléments importants qui peuvent aider l’utilisateur à choisir la meilleure approche. Parmi toutes les approches mentionnées, il existe plusieurs méthodes d’étalonnage disponibles en ligne sous forme de ”Toolbox Open Source”. Ces méthodes permettent d’économiser en temps et en effort, en particulier lorsque l’utilisateur est plus intéressé par l’obtention des résultats de structure 3D que de traiter des modèles de projection complexes. Ces méthodes d’étalonnage sont classées selon 4 catégories :

- *Sphere-2D Pattern* : [77], qui utilise le modèle sphérique de la caméra et nécessite plusieurs images d’un modèle de test 2D.

- *DLT-like* :[89], qui utilise également le modèle sphérique de la caméra. Elle présente une solution analytique basée sur un ensemble de correspondances 3D-2D.
- *Sphere-Lines* :[13], qui utilise le modèle sphérique de la caméra et nécessite une image omnidirectionnelle unique contenant au minimum trois lignes.
- *Distortion-2D Pattern* : [99], qui considère les images omnidirectionnelles en tant que images déformées avec des paramètres de distorsion qui doivent être estimés.

L'étalonnage des caméras omnidirectionnelles n'est pas simplement différent des caméras classiques, elle est aussi plus difficile. Les caméras ordinaires présentent un problème linéaire car elles projettent une ligne en une ligne alors que les caméras omnidirectionnelles présentent un problème non-linéaire car elles transforment une ligne réelle en un cône sur l'image. Les travaux de recherche qui ont été réalisés sur l'étalonnage de telles caméras essaient donc de trouver une solution approchée linéaire pour résoudre le problème de cette non-linéarité. Plusieurs techniques ont été proposées dans la littérature, adaptées à différents modèles de formation d'images (Tableau 1.2⁸). Dans cette partie, nous présentons une taxonomie de classification des méthodes d'étalonnage existantes. On propose cinq catégories en fonction du mode d'étalonnage :

- Etalonnage basé sur des lignes caractéristiques ("Line based calibration") :

L'avantage de cette approche est qu'on a plus besoin d'un modèle de test et principalement que les lignes sont toujours présents dans de nombreuses scènes. Ces approches calculent les paramètres intrinsèques du système catadioptrique à partir de l'image de la conique absolue. Geyer et Daniilidis [46] utilisent seulement trois lignes de chaque image pour étalonner leurs caméras paracatadioptriques. Ying et Hu [123] utilisent la projection des lignes ainsi que la projection de la sphère pour analyser la relation entre les paramètres intrinsèques de la caméra et les contours de la sphère projetée sur l'image. Vasseur et Mouaddib [111] propose un algorithme, valable pour tout système catadioptrique central, qui détecte des lignes omnidirectionnelles à partir d'une scène 3D qui sont ensuite utilisés pour calculer les paramètres intrinsèques.

- Étalonage avec modèle 2D "2D pattern calibration" :

Ces méthodes utilisent un modèle d'étalonnage 2D avec des points caractéristiques (coins, points,...) faciles à extraire des images. Scaramuzza et al. [99] proposent une technique d'étalonnage pour les caméras omnidirectionnelles à point de vue unique. Ils supposent que la fonction de projection sur l'image peut être décrite par une écriture en série de Taylor, dont les coefficients sont estimés par la résolution d'un problème de minimisation. Mei et Rives [77] proposent, comme Scaramuzza, une approche souple pour l'étalonnage des caméras à point de vue unique, mais en se basant sur une fonction de projection théorique équivalente au modèle de la sphère. Deng et al. [34] utilisent le contour de l'ellipse de l'image catadioptrique et le champ de vision (FOV) pour obtenir les paramètres intrinsèques. Ensuite, ils utilisent la relation entre le catadioptrique central et le modèle sténopé de projection pour calculer les paramètres extrinsèques.

- Étalonage basé sur des points 3D ("3D Point-based calibration") :

Ces méthodes utilisent les positions 3D de points réels observés dans une seule image. Aliaga [4] propose une approche qui ne fonctionne que pour les systèmes paracatadioptriques. Elle permet l'estimation des paramètres intrinsèques et extrinsèques de la caméra, sachant que la projection du centre du miroir doit être déterminée manuellement.

Puig et al. [15] présentent une approche basée sur la transformation directe linéaire ("DLT") valable pour toute caméra catadioptrique centrale à travers une matrice de projection générique. A partir de cette matrice, les paramètres intrinsèques et extrinsèques sont extraits par optimisation non linéaire. Cette approche nécessite une image omnidirectionnelle unique contenant des points

8. source du tableau : [90]

Ref	Modèle de test	Nbre de vues	Type du miroir	Modèle caméra
[104]	—	unique	générique	Modèle hyperbolique central
[83]	3 images polarized	multiple	générique	Modélisation générique
[99]	plusieurs vues de modèle 2D	multiple	générique	Modèle de distorsion
[77]	plusieurs vues de modèle 2D	multiple	générique	Modèle sphérique
[34]	plusieurs vues de modèle 2D	multiple	générique	Modèle sphérique
[46]	3 lignes	unique	parabolique	Modèle sphérique
[13]	3 lignes	unique	générique	Modèle sphérique
[25]	1 ligne + contour miroir	unique	générique	Modèle catadioptrique non-central
[80]	9 correspondances	multiple	générique	Modèle de distorsion
[93]	2 ensemble de translations et rotations	multiple	générique	Modélisation générique
[40]	2 ensemble de rotations	multiple	générique	Modélisation générique
[4]	3D points	multiple	parabolique	Modèle catadioptric non-central
[15]	Modèle 3D (20 correspondances 3D-2D)	unique	générique	Modèle sphérique
[110]	3 lignes	unique	parabolique	Modèle sphérique

TABLE 1.2 – Classification des méthodes d'étalonnage du système catadioptrique

répartis dans au moins trois plans différents.

— Auto-calibration ("Self-calibration") :

Ce type de techniques utilise uniquement des points correspondants extraits de plusieurs points de vue, sans avoir besoin de connaître ni leurs localisations 3D ni leurs emplacements par rapport à la caméra. Micusik et Pajdla [80] proposent une méthode valable pour les objectifs fish-eye et les systèmes catadioptriques. Ils étudient la géométrie de la caméra, ses propriétés, la formation de l'image et aussi la possibilité de réaliser un auto-étalonnage à partir de correspondances de point. Ramalingam et al. [93] utilisent des translations, des rotations et des correspondances images pour calibrer les caméras centrales à partir des contraintes géométriques de projection.

— Image de polarisation ("Polarisation Imaging") :

Cette méthode est proposée par Morel et Fofi [83] qui utilisent l'imagerie de polarisation pour une reconstruction précise de la forme de la surface du miroir en se basant sur l'algorithme de Frankot-Chellappa [43].

Dans [66], Puig a réalisé une évaluation des différentes méthodes sur des systèmes omnidirectionnels. L'étude a montré que toutes les approches ont donné de bons résultats, à l'exception de l'approche "Sphere-Lines" appliquée sur un système "fish-eye". En particulier, pour ces lentilles, le meilleur étalonnage a été réalisé avec l'approche "DLT-like". Dans le cas du capteur catadioptrique de paramètres inconnus, l'approche "Distortion-2D Pattern" a fourni le meilleur résultat. Cette étude a également donné de l'importance à la surface occupée par les modèles d'étalonnage (points, lignes) dans l'image.

Pour notre banc de calibrage, nous avons utilisé les images provenant du système utilisé par Mouaddib et Vasseur dans [110]. Ce Système se base sur le modèle sphérique et utilise le miroir parabolique.

1.4 Traitement des images Omnidirectionnelles

Les systèmes catadioptriques ne sont pas sans failles : déformation de la géométrie de l'image, diminution de la résolution dans le centre de l'image (puisque la scène embrassée est beaucoup plus vaste, chaque pixel regarde donc un \blacksquare angle solide \blacksquare plus important), calibrage complexe du système...

Toutefois, grâce à ses nombreux facteurs, de plus en plus de chercheurs approfondissent les études sur le système catadioptrique. Pour le moment, il est appliqué principalement dans les trois

piliers suivants :

- La navigation du robot mobile ([45], [60], [41],[5],[11])
- La reconstruction de scène en 3D [[23], [96],[49],[85]]
- La surveillance "intelligente" et suivi d'objet [[87],[26],[57],[54],[114],[76],[79]]

La façon adoptée pour traiter ce type d'images n'est pas simple. Pourtant, dans la littérature il y a des chercheurs qui ont choisit d'appliquer les traitements classiques directement . Et d'autres qui ont adapté les traitements selon la forme du système utilisé (objectif et miroir) et faire des projections sur des espaces équivalents.

1.4.1 Traitements Classiques

Vlasis, Motomura, Hara et Asoh dans [112] ont travaillé sur la détection de contour. Leur objectif est d'extraire l'information de bas niveau d'une séquence d'images omnidirectionnelles pour la localisation et la navigation d'un robot mobile. La méthode proposée suit trois étapes : la détection de contour dans l'image. L'estimation de Parzen de la densité spatiale de ces contours et l'analyse des composantes principales. Pour la détection de contour, deux filtres de Sobel décalés de 90 degrés sont utilisés. La taille de ces filtres est toujours la même quelle que soit leur position dans l'image. L'image omnidirectionnelle est donc traitée de la même manière qu'une image perspective.

N'importe quel traitement classique n'aura pas le même effet au centre et à la périphérie de l'image. En effet, la géométrie du miroir du système introduit des distorsions importantes dans l'image et crée une concentration de l'information au centre de l'image.

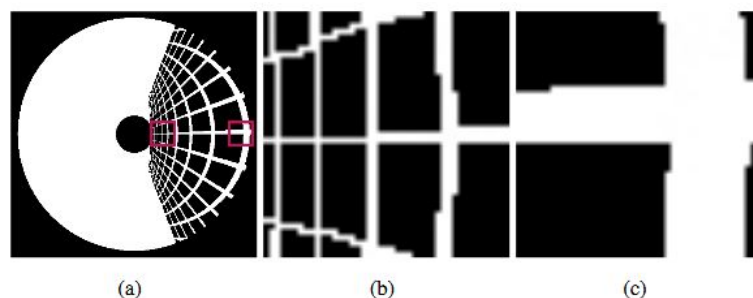


FIGURE 1.16 – (a) Image omnidirectionnelle d'une grille, (b) Zoom sur la zone centrale de l'image, (c) Zoom sur la zone périphérique de l'image.

L'application des traitements classiques tels qu'ils sont sur les images omnidirectionnelles peut donner dans certains cas de résultats acceptables, mais ceci ne pourra jamais être généralisé pour n'importe quelle application puisque dans la plupart des cas on est obligé d'adapter les traitements à la géométrie des images omnidirectionnelles pour obtenir de bon résultat. C'est ce que nous proposons dans ce manuscrit.

1.4.2 Adaptation des traitements

Les images échographiques (figure 1.17) est un exemple des images qui présentent des déformations et une résolution non uniforme. Ces images sont obtenues en coordonnées polaires ce qui va créer une anisotropie importante dans la résolution spatiale.

Habituellement, une transformation géométrique (ou reconstruction cartésienne) permet de passer des données en géométrie polaire, à une représentation cartésienne, par interpolation bilinéaire ou avec un algorithme de Bresenham [24]. Plusieurs chercheurs ont montré les limites de



FIGURE 1.17 – Image échographique

cette approche. ce qui rend encore les approches d'adaptation selon la géométrie plus robustes.

Les images omnidirectionnelles, comme les images échographiques présentent des déformations. Dans le cadre de l'imagerie omnidirectionnelle, plusieurs solutions ont été proposées afin d'obtenir des résultats significatifs. Cependant, les nouveaux algorithmes créés sont souvent partiellement adaptés aux images. On peut distinguer deux principales catégories d'adaptations :

1. La projection et le traitement de l'image dans un espace équivalent comme une sphère ou un cylindre.
2. L'utilisation de la géométrie du miroir afin d'obtenir des informations sur les distorsions de la scène observée sur l'image.

Approche 1 : Projection dans un espace équivalent

Dans la littérature, les espaces équivalents de projection les plus utilisés pour le traitement sont la sphère (Figure 1.18) et le cylindre (Figure 1.19) introduits respectivement par Daniilidis dans [47] et jacquey dans [59].



FIGURE 1.18 – Projection de l'image sur l'hémisphère.

L'espace du cylindre permet une vision panoramique de la scène, très proche de la vue perspective. C'est un support avantageux car il est possible de l'échantillonner avec une grille régulière dans un repère cylindrique. Cette projection est très lisible par l'œil humain. Cependant, l'image reste anamorphosée au sens où des lignes droites perpendiculaires à l'axe de rotation du miroir ne se projettent pas comme des droites sur l'image projetée. Les lignes horizontales dans la scène 3D sont donc déformées dans l'image cylindrique. De plus, comme pour l'image perspective, la reconstruction génère une interpolation des valeurs des niveaux de gris des pixels, liée au lissage et à l'échantillonnage.



FIGURE 1.19 – Projection de l'image sur le cylindre.

La projection de l'image omnidirectionnelle sur la sphère ou demi-sphère nécessite de savoir quelques paramètres de la caméra intrinsèques à savoir les coordonnées du centre optique et le paramètre α lié au miroir. un maillage suivi d'une interpolation est primordiale pour former une image sphérique. Ensuite, pendant le traitement, toutes les opérations suivent cette géométrie.

Approche 2 : Utilisation de la géométrie du miroir

Avec une caméra catadioptrique central, la contrainte du point de vue unique est vérifiée. A partir de l'image anamorphosée, il est donc possible de reconstituer n'importe quelle image perspective de la scène observée. Sur un plan choisi dans un champ de vision du capteur, on définit un maillage (échantillonnage de l'image), puis on calcule la valeur de niveau de gris de chaque point du maillage par interpolation.

Chaque image omnidirectionnelle prise par une caméra avec single view point (SVP) est modélisée par une image sphérique (Fig.3.9). Cette projection est appelée le modèle de projection unifié. C'est une projection sphérique qui tient compte de la résolution non linéaire causée par la forme du miroir utilisé.



FIGURE 1.20 – Géométrie sphérique de l'image Omnidirectionnelle

Les coordonnées sphériques d'un point P_s sont définies par Eq.(3.6) :

$$\begin{cases} X_s = \cos(\varphi) \sin(\theta) \\ Y_s = \sin(\varphi) \sin(\theta) \\ Z_s = \cos(\theta) \end{cases} \quad (1.1)$$

La projection stéréographique du point sphérique P_s sur le plan catadioptrique est exprimée par les coordonnées cartésiennes comme montré dans l'équation. (3.7) :

$$\begin{cases} u = \frac{X_s}{1 - Z_s} \\ v = \frac{Y_s}{1 - Z_s} \end{cases} \quad (1.2)$$

En utilisant les équations (3.6) et (3.7), nous obtenons le point $P_i(u, v)$ sur l'image exprimé par les coordonnées sphériques comme le montre l'équation (3.8) :

$$\begin{cases} u = \cot \frac{\theta}{2} \cos(\varphi) \\ v = \cot \frac{\theta}{2} \sin(\varphi) \end{cases} \quad (1.3)$$

Où $\theta \in 0 \text{ and } \pi$, et $\varphi \in 0 \text{ and } 2\pi$. Chaque point sur la sphère est caractérisé par deux paramètres (θ, φ) .

1.4.3 Exemples de traitement des images omnidirectionnelles

Nous allons citer les travaux qui ont été faits pour le traitement bas niveau. Parmi ces travaux, il y en a peu qui appliquent les différents opérateurs mathématiques directement sur l'image omnidirectionnelle. En revanche, les approches traitant la projection sur les espaces équivalents donnent des résultats logiques. Citons par exemple les traitements majoritairement utilisés dans les domaines cités au début de cette section :

- Détection de contour ([59],[64], [19])
- Lissage ([64]), [20]).
- Extraction des caractéristiques ([26], [91], [92], [82],[7]).
- Segmentation ([18], [53]).
- Estimation de probabilité ([94], [57], [54], [128], [75], [74])

Daniilidis, Makadia et Bulow ont travaillé dans [30] sur l'estimation de mouvement pour des images omnidirectionnelles pour définir l'opérateur gradient.

La détection de contour dans le cas perspective en 2D se fait en utilisant des fenêtres de filtre convolués avec l'image en question. Autrement, un traitement comme celui ci n'est pas applicable sur une image omnidirectionnelle de manière directe avant d'être adapté à la géométrie. Pour cela plusieurs chercheurs comme Daniilidis dans [47] ont fait un changement de repère en coordonnées sphériques pour appliquer ce filtre sur une image qui subit une projection sur la sphère.

Cette approche a été utilisée aussi par Khalid et al. dans [64] qui ont appliqué un filtre de Sobel sphérique pour la détection de contour afin de réaliser un suivi d'objet par la méthode des contours actifs. Stephanie et al. dans [19] ont adapté l'algorithme de Canny de détection de contour sur la sphère. Jacquy et al. dans [59] ont réalisé une projection cylindrique pour adapter les masques de convolution de filtre de Prewitt.

Le traitement de dé-bruitage des images en utilisant une convolution avec un filtre est appelé lissage. La gaussienne est le filtre le plus utilisé à ce propos. Il a été adapté dans le cas sphérique pour le pré-traitement des images omnidirectionnelles. Il a été utilisé par Daniilidis et al. dans [47] pour le lissage et la détection du contour. Stéphanie et al. dans [19] l'ont utilisé avec la fonction de Green.

L'estimation de mouvement est un vaste domaine d'application qui rassemble plusieurs travaux dans le cas omnidirectionnel. commençons par les travaux de Daniilidis dans [47], qui a estimé le flot optique. Cette approche est appliquée par Radgui et al. dans [92] qui a fait une adaptation sphérique de la méthode de Lukas et Kanade [71] en utilisant les ondelettes sphériques. Cette

adaptation a montré une amélioration qui mène à conclure d'adapter le traitement à la géométrie de l'image omnidirectionnelle. Marcovic dans [75] a estimé les paramètres du mélange de densité de probabilité de Von mises dans le but d'estimation de mouvement d'objet. Le même auteur dans [74] a travaillé sur la densité sphérique de Von mises Fisher dans une application d'estimation de pose. Cette densité sera l'objet de notre contribution dans le chapitre 5. Dans [94], [26] et [128] l'auteur a adapté le filtre particulaire sur la sphère dans le but de réaliser une localisation et un suivi d'objet.

1.5 Conclusion

L'application des traitements classiques tels qu'ils sont sur les images omnidirectionnelles peut donner dans certains cas de résultats acceptables au niveau pratique. Mais ceci ne pourra jamais être généralisé pour n'importe quelle application puisque dans la plupart des cas on est obligé d'adapter les traitements surtout au niveau théorique à la géométrie des images omnidirectionnelles déformée pour obtenir de bons résultats.

2.1 Introduction

Le Suivi d'objet est une tâche importante dans le domaine de la vision par ordinateur. L'utilisation d'ordinateurs de grande puissance et caméras vidéo de haute qualité couteuses a créé la nécessité croissante pour l'analyse vidéo. Ce besoin a généré beaucoup d'intérêt dans les algorithmes de suivi d'objet. Dans sa forme la plus simple, le suivi peut être défini comme le problème de l'estimation de la trajectoire d'un objet, dans le plan de l'image, qui se déplace dans une scène 3D.

Le Suivi d'objet peut être complexe en raison de :

- La perte d'information causée par la projection de l'espace 3D sur une image 2D.
- Le bruit dans les images.
- La complexité du mouvement d'un objet
- La non rigidité, la nature d'objet.
- Les occultations Partielles ou totales d'objet.
- Les formes d'objets complexes.
- Dans le contexte d'un système des images omnidirectionnelles, la complexité réside essentiellement dans le changement de forme et de taille de l'objet causé par la géométrie déformée.

2.2 Généralités sur le suivi d'objet

2.2.1 Contexte de suivi

Nous pouvons simplifier le suivi en imposant des contraintes sur le mouvement par exemple. Presque tous les algorithmes de suivi supposent que le mouvement d'un objet est lisse sans changements brusques.

Les réponses à ces questions dépendent de l'environnement et le contexte dans lequel le suivi est effectué et dépendent également de l'utilisation finale pour laquelle les informations de suivi sont demandées.

Dans l'environnement des systèmes catadioptriques, il est nécessaire de trouver des descripteurs invariants par rapport au changement de colorimétrie, de géométrie ou autres. Un grand nombre de méthodes de suivi ont été proposées. L'objectif de cette enquête est de regrouper les méthodes de suivi dans une large catégorie et de fournir des descriptions complètes des méthodes représentatives

de chaque catégorie. Par ailleurs, nous cherchons à identifier les nouvelles tendances et idées dans la communauté de suivi pour donner un aperçu du développement de nouvelles méthodes qui seront bien adéquates pour notre cas.

2.2.2 Représentations d'un objet

Dans un scénario de suivi, un objet peut être défini comme le centre d'intérêt de l'analyse effectuée sur l'image. Par exemple, les bateaux sur la mer, les poissons à l'intérieur d'un aquarium, des véhicules sur une route, avions dans les airs, les gens marchant sur une route, ou de bulles dans l'eau. Ils sont un ensemble d'objets important à suivre dans un domaine spécifique. Tout objet déjà cité appartient à un système de vision perspective. En revanche dans un système catadioptrique et plus que ces exemples, nous pouvons ajouter des objets variables de forme et de taille.

Pour distinguer entre méthodes de suivi, un objet peut être représenté par :

- Points d'intérêt : L'objet peut être représenté par un point, le centre de gravité [Veenman et al. 2001], ou par un ensemble de points [Serby et al. 2004]. En général, le point de la représentation est adapté pour les objets de suivi qui occupent de petites régions dans une image.

- Primitif des formes géométriques : La forme de l'objet est représenté par un rectangle, une ellipse [Comaniciu et al. 2003]. Bien que les formes géométriques simples sont plus appropriés pour une représentation simple des objets rigides, ils sont également utilisés pour le suivi des objets non rigides.

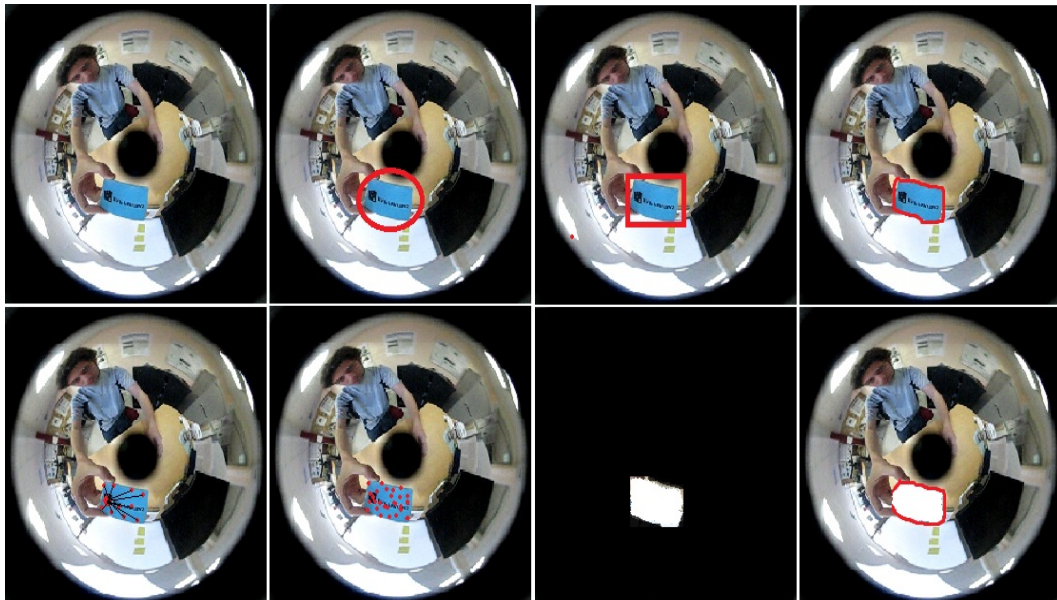


FIGURE 2.1 – Représentation de l'objet.

- Silhouette et contour : La représentation du contour définit la limite d'un objet. La région à l'intérieur du contour est appelée la silhouette de l'objet [Yilmaz et al. 2004].

- Modèles de forme articulation : Les objets articulés sont composés de parties du corps qui sont détenus avec des joints. Par exemple, le corps humain est un objet articulé avec le torse, jambes, mains, tête, pieds et reliés par des articulations. La relation entre les parties est régit par des modèles de mouvement cinématiques, par exemple, angle de l'articulation, etc.. Afin de représenter un objet articulé, on peut modéliser les éléments constitutifs en utilisant des cylindres ou des ellipses.

- Modèles Squelettiques : La squelette de l'objet peut être extraite en appliquant la transformé d'axe médian à la silhouette de l'objet [Ballard et Brown 1982, chap.8]. Ce modèle est communément utilisé comme une représentation de forme pour reconnaître des objets [Ali et Aggarwal 2001]. Il y a un certain nombre de façons de représenter les caractéristiques d'apparence des objets. Notons que les représentations de forme peuvent également être combinées avec les représentations d'apparence [Cootes et al. 2001] pour le suivi.

2.2.3 Taxonomie des méthodes de Suivi d'objet

D'après la représentation de l'objet, nous avons classifié les méthodes de suivi selon la façon dont nous avons représenté notre objet :

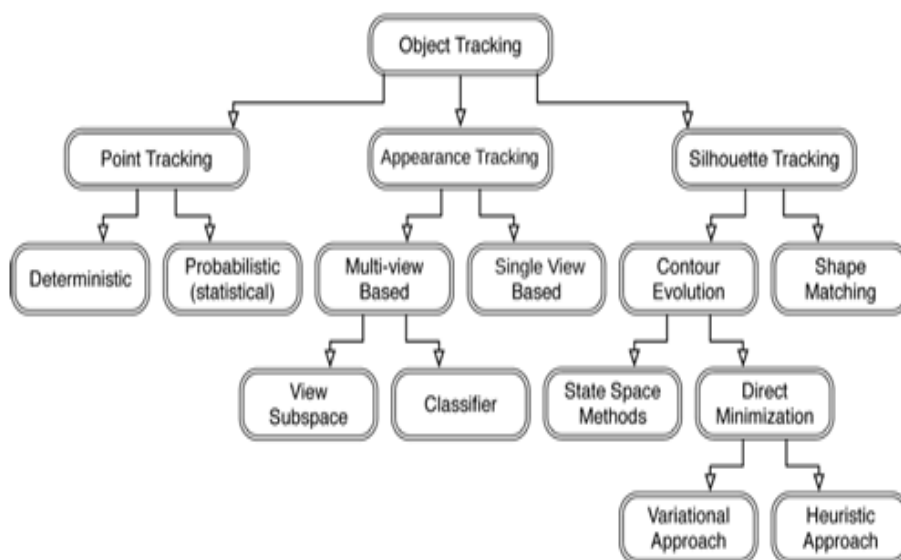


FIGURE 2.2 – Taxonomie des méthodes de suivi.

- Le suivi de points : ce sont des méthodes qui suivent les points d'intérêts liés à un objet ou des objets ponctuels. Il y a une multitude de techniques permettent de suivre un point dîtes des primitifs simples dans une séquence d'images à savoir la méthode SIFT, HARRIS, RANSAC, etc.

- Le suivi d'apparence : ce sont des méthodes utilisant une mesure de similarité entre des modèles des images par le biais de distance. Ces modèles d'images comme l'histogramme de couleur par exemple sont utilisés par la méthode Mean shift. Aussi l'utilisation des moment d'images pour la méthode CAM shift. Les distributions de probabilité sont aussi tenue en compte pour présenter une modélisation d'une image ou un bloc d'image.

- Le suivi de contour et silhouette : ce sont des méthodes se basant dans le suivi sur d'une part les caractéristiques d'objet comme le contour et la région intérieur de d'objet et d'autre part sur sa forme et la façon dont il se déplace dans la scène. La méthode des contours actifs ou Snake est la plus utilisée dans ce contexte.

2.3 Suivi d'objet

2.3.1 Méthodes basée sur points

Méthodes Déterministes

Les méthodes Déterministes sont caractérisées par des primitives ponctuelles qui sont détectées pour chaque image ainsi que chaque cible. Le suivi de cette cible est un procédé de mise en correspondance qui relie une image à une autre. Généralement, les primitives présentent une caractéristique discriminante, et invariante au court du temps (par exemple des coins ou des points à fort contraste). Ce qui assure qu'elles correspondent à un point réel des objets suivis, et que si elles sont détectées sur une image, elle le sont aussi sur les autres. Il est donc possible pour chaque primitive mise en évidence de calculer sa trajectoire le long de la séquence.

Ces méthodes (appelés aussi de points d'intérêt) représentent la cible par des descripteurs locaux calculés sur une petite région autour du point considéré. Les points sont déterminés sur l'objet par des méthodes de détection telles que SIFT (Scale-Invariant Feature Transform) [69], FAST (Features from Accelerated Segment Test) [97] et SURF (Speeded Up Robust Features) [16].

Dans [121], Yang et al. proposent un algorithme basé sur les points SIFT et des patches aléatoires pour modéliser l'objet suivi. Les auteurs utilisent conjointement deux sous-modèles d'apparence : un sous-modèle de couleur/texture et un autre de structure. Le sous-modèle de couleur/texture inclut des patches aléatoires décrits par leurs histogrammes de couleurs locaux RVB (Rouge, Vert, Bleu) et leurs descripteurs LBP (Local Binary Patterns). Le sous-modèle de structure est basé un histogramme spatial global encodant la disposition des points SIFT de la cible.



FIGURE 2.3 – Mise en correspondance entre deux images Omnidirectionnelles.

L'utilisation de ces méthodes a des limites à savoir : l'absence des points clés dans la région où se trouve notre cible rend le suivi impossible. Mais ce problème a été résolu par la sauvegarde des points d'intérêt au cours du traitement en utilisant la méthode Guo [52].

L'application nécessite plus du temps de calcul des descripteurs pour chaque image de la séquence surtout lorsque la région de recherche est importante.

Méthodes Probabilistes

La majorité des techniques de suivi d'objet de cette classe sont basées sur le filtrage Bayésien [[12], [22]]. Le filtrage Bayésien n'est autre que l'estimation récursive de l'état des objets à base d'observations acquises d'une manière séquentielle. L'état des objets ainsi que leurs observations sont considérées comme étant des variables aléatoires. Dans le cadre Bayésien, cela sous-entend une modélisation sous forme de densités de probabilités dont la propagation est basée sur l'inférence Bayésienne.

Dans le contexte des systèmes linéaires et Gaussiens, le filtre de Kalman est parmi les estimateurs les plus connus et utilisés pour le suivi d'objets. Il assure une estimation optimale de la densité de probabilité désirée. A chaque pas de temps k , le filtre de Kalman par exemple exécute deux étapes principales : une étape de prédiction et une étape de mise à jour [98]. Son principal inconvénient est sa restriction aux systèmes linéaires et Gaussiens. Généralement, la dynamique des objets ainsi que les modèles d'observation sont non linéaires. En terme de modélisation, cela se traduit par une fonction d'état f et une fonction d'observation h non linéaires. Afin de permettre le suivi d'objets dans le cas des modèles non linéaires, une autre technique de filtrage non linéaire se rajoute à la littérature du domaine du suivi d'objet, on parle notamment du filtrage PHD appliqué aux images omnidirectionnelles [57]. Contrairement, au filtre de Kalman étendu qui se base sur échantillonnage déterministe de la densité de probabilité $p(x(k)|z(1:k))$ et qui résolve la non-linéarité, les filtres particuliers proposent un échantillonnage aléatoire de cette densité de probabilité.

Un filtre à particules a pour but d'estimer la séquence de paramètres x_k pour $k = 0, 1, 2, 3...N$ en se basant seulement sur les données observées y_k pour $k = 0, 1, 2, 3...N$.

Tous les paramètres estimés bayésiens de x_k viennent de la distribution a posteriori $p(x_0, x_1, \dots, x_k | y_0, y_1, \dots, y_k)$ qui résulteraient d'une MCMC (Markov Chain Monte Carlo) usuelle ou d'un échantillonnage d'importance. Les méthodes à particule estiment la distribution de filtrage $p(x_k | y_0, y_1, \dots, y_k)$.

En général, l'algorithme est répété de manière itérative pour un nombre donné de valeurs k (que nous noterons N). Une fois ceci effectué, la moyenne sur toutes les particules $(\frac{1}{P} \sum_{L=1}^P x_k^{(L)})$ est approximativement la véritable valeur de x_k .

Dans le contexte omnidirectionnel, Bazin dans [17] a mis plus de nombre de particules au centre qu'aux périphéries car c'est difficile de faire l'échantillonnage à cause de la géométrie déformée. Il a réussis aussi à adapter le voisinage de points dans la technique d'histogramme.

2.3.2 Méthodes Basées sur l'apparence

Dans le cas du suivi d'objet non rigide en se basant sur les caractéristiques du contenu de l'image ou bloc plus que la forme. L'objet objectif de suivi se base sur deux étapes principales de haut niveau : la localisation et la correspondance. La façon dont les deux étapes sont combinées et pondérées dépend de l'application et joue un rôle décisif dans la robustesse et l'efficacité du suivi. Par exemple, le suivi du visage dans une scène encombrée repose davantage sur la représentation de la cible que sur sa dynamique, tandis que dans la vidéo surveillance aérienne, le mouvement de la cible et le mouvement de la caméra sont les plus importantes composantes.

La méthode KLT

L'algorithme KLT est une méthode de détection et de suivi [[71],[107]] basée sur la somme des carrés des différences d'intensité (SSD) dans la fenêtre de suivi entre des trames de séquences vidéo. En supposant qu'une image grise contient une fonctionnalité window W , qui comprend les

caractéristiques de la cible suivis. L'emplacement de la fenêtre de fonction dans l'intervalle $[t, t + \tau]$ devrait répondre à la relation suivante :

$$I(x, y, t + \tau) = I(x - \aleph, y - \beth, t) \quad (2.1)$$

C'est l'emplacement de pixels dans l'image $I(x, y, t + \tau)$ qui pourrait être obtenue par correspondance de pixels dans l'image $I(x, y, t)$ avec un déplacement de $\vec{d}(\aleph, \beth)$. Le but de l'algorithme KLT est de calculer ce déplacement.

Les auteurs dans [95] proposent d'utiliser un "compagnon" pour améliorer le suivi. Ce dernier correspond à une région avoisinant la cible en ayant un mouvement similaire. D'une manière semblable, Yang et al.[122] utilisent plusieurs "objets auxiliaires" assistants. Grabner et al. [50] proposent une approche plus générale pour le suivi orienté contexte. Ils introduisent la notion de "supporteurs" définis comme étant des "caractéristiques utiles pour la prédiction de la position de la cible". Le déplacement de ces points est statistiquement lié au mouvement de la cible.

Le Bloc Matching

L'algorithme "block matching" décrit et utilisé dans [102] par Spors et Rabenstein pour le suivi des yeux, consiste à rechercher pour chaque point d'intérêt de l'image I_t , le point de l'image I_{t+1} qui maximise un score de corrélation. La recherche est effectuée dans une fenêtre d'intégration.

Nous commençons par détecter les positions des points d'intérêt dans la première image de la séquence, et pour chaque point on extrait un masque de corrélation qui servira de référence. Dans les images suivantes de la séquence, les nouvelles positions des points sont estimées par la recherche des points qui minimisent la distance de block définie par :

$$\eta_{x,y} = \frac{3}{4}(p_{ref} - p_{x,y})^2 + \frac{1}{4}(p_{tmp} - p_{x,y})^2 \quad (2.2)$$

où p_{ref} désigne le masque de référence, p_{tmp} le masque extrait de l'image précédente et $p_{x,y}$ celui extrait de l'image traitée. Pour chaque point d'intérêt x , le point $x \in \omega$ qui donne la distance de block minimale est retenu comme la nouvelle position de x . On recherche donc les correspondants dans I_{t+1} , des points de I_t , en utilisant comme mesure de similarité une distance. La prise en compte d'un masque de référence dans le calcul de la distance, permet de limiter les effets dûs à un changement d'illumination pendant l'acquisition de la séquence d'images.

Le Mean Shift

Le modèle cible de référence est représenté par \mathbf{q} dans l'espace caractéristiques. Par exemple, le modèle de référence peut être choisi pour être la fonction de densité de probabilité du couleur de la cible. Le modèle cible peut être considéré comme centré à l'emplacement spatial.

Une cible candidate est définie à l'emplacement \mathbf{y} , et elle est caractérisée par $\mathbf{p}(\mathbf{y})$. Les deux fonctions de densité probabilité doivent être estimées à partir des données. Pour satisfaire un faible coût de calcul imposé par le traitement des densités discrètes, l'histogramme doit être utilisé. Ainsi, nous avons :

$$\hat{q} = \{\hat{q}_u\}_{u=1,\dots,m} \quad \sum_{u=1}^m \hat{q}_u = 1 \quad (2.3)$$

$$\hat{p}(y) = \{\hat{p}_u(y)\}_{u=1\dots m} \quad \sum_{u=1}^m p_u = 1 \quad (2.4)$$

L'histogramme n'est pas la meilleure densité non paramétrique pour l'estimation, mais il suffit aux besoins de suivi d'objet. D'autres estimations de la densité peuvent également être employées. Nous noterons :

$$\hat{\rho}(y) = \rho[\hat{p}(y), \hat{q}] \quad (2.5)$$

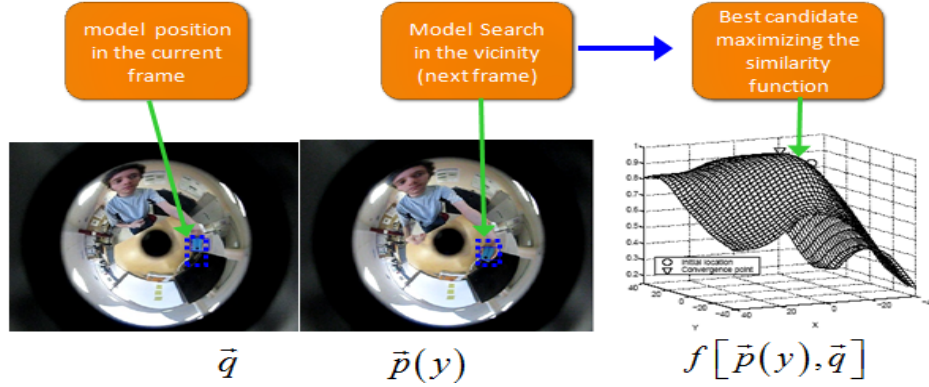


FIGURE 2.4 – Scénario de correspondance entre deux images dans une séquence par la méthode Mean Shift

La fonction de similarité définit une distance entre le modèle cible et les candidats. Pour faire face à des comparaisons entre les différentes cibles, cette distance devrait avoir une structure métrique. Nous définissons la distance entre deux distributions discrètes et l'estimation de l'échantillon du coefficient de Bhattacharyya entre \mathbf{p} et \mathbf{q} .

$$d(y) = \sqrt{1 - \rho[\hat{p}(y), \hat{q}]} \quad (2.6)$$

Dont on choisit :

$$\hat{\rho}(y) = \rho[\hat{p}(y), \hat{q}] = \sqrt{\rho[\hat{p}(y), \hat{q}]} \quad (2.7)$$

Le coefficient de Bhattacharyya est une mesure de type divergence qui a une interprétation géométrique simple. Il est le cosinus de l'angle entre les vecteurs unitaires de dimension \mathbf{m} , $(\sqrt{\hat{p}_1}, \sqrt{\hat{p}_2} \dots \sqrt{\hat{p}_m})$ et $(\sqrt{\hat{q}_1}, \sqrt{\hat{q}_2} \dots \sqrt{\hat{q}_m})$.

L'information de couleur a été choisie comme la fonction de la cible, Cependant, le même cadre peut être utilisé pour la texture et les arêtes, ou une combinaison entre eux. Dans l'algorithme du mean shift dans [28] dans le travail de D. Comaniciu, V. Ramesh, et P. Meer, la recherche de l'objet est effectuée en minimisant une distance entre histogrammes de couleur selon une méthode de type descente de gradient. Ceux ci conduit à une bonne précision de suivi. Dans le contexte omnidirectionnel, les auteurs dans [94] ont fait une adaptation du noyau mean shift dans un espace sphérique en adoptant une nouvelle distance géodésique [33].

2.3.3 Méthodes Basées sur Contour et silhouette

Parmi les méthodes les plus connues, le contour actif ou snake est une méthode de suivi de contour basée sur le calcul des énergies. Ce calcul est fait par le biais d'un détecteur de contour.

Ces énergies doivent être minimiser pour trouver le contour de l'objet à suivre. Cette méthode est caractérisée par un ensemble de points initialisés autour de la cible.

La minimisation de l'énergie revient alors à minimiser la distance entre ces points. Pour éviter la grande rétraction entre points Williams et Shah dans [116] ont proposé d'utiliser la différence de distance entre points par rapport à la distance moyenne.

les fonctions d'énergies est données par la formule suivante :

$$\mathbf{E}_{\text{Totale}} = \sum_{i=1}^N (\mathbf{a} * \mathbf{E}_{\text{continuit}}(\mathbf{P}_i) + \mathbf{b} * \mathbf{E}_{\text{courbure}}(\mathbf{P}_i) + \mathbf{c} * \mathbf{E}_{\text{gradient}}(\mathbf{P}_i) + \mathbf{d} * \mathbf{E}_{\text{ballon}}(\mathbf{P}_i)) \quad (2.8)$$

$\mathbf{P}_i = I(x_i, y_i)$ $i = 1..N$ sont les points du snake. a, b, c et d sont les coefficients attribués à chaque énergie. On distingue deux types d'énergies. Une énergie de continuité et une énergie de la courbure. L'énergie de continuité est définie par la formule suivante :

$$\mathbf{E}_{\text{continuit}} = \left| \text{distance}_{\text{moyenne}} - \sqrt{(x_i - x_{i-1})^2 - (y_i - y_{i-1})^2} \right| \quad (2.9)$$

Où $\text{distance}_{\text{moyenne}}$ est la moyenne des distances entre deux points successives du contour actif. Il est clair que si on minimise cette énergie, le point \mathbf{P}_i doit se positionner à une distance égale à $\text{distance}_{\text{moyenne}}$ du point \mathbf{P}_{i-1} .

L'énergie de la courbure est donnée par la formule suivante :

$$\mathbf{E}_{\text{courbure}} = \sqrt{(x_{i-1} + 2x_i + x_{i-1})^2 - (y_{i-1} + 2y_i + y_{i-1})^2} \quad (2.10)$$

la méthode des contours actifs appliqués dans un contexte de suivi a des inconvénients à savoir : L'initialisation des paramètres des énergies ce qui influe sur la convergence de l'algorithme. Ceux ci peut être résolue dans les perspectives par l'utilisation des algorithmes d'optimisation comme les Algorithmes génétiques.

2.4 Résultats & Synthèse

Dans notre synthèse, nous avons appliqué des méthodes de notre état de l'art dans l'objectif de déduire les quelles sont adéquates à notre contexte. Un suivi d'objet dans une séquence d'images omnidirectionnelles doit être réalisé par le biais d'une méthode qui respecte la géométrie déformée des images provenant d'un système catadioptrique. Nous avons fais le tour sur la littérature et nous avons classifié les méthodes selon la représentation de la cible (points, apparence et contour). Dans les méthodes basées sur points (SIFT, SURF, FAST), l'application directe du suivi d'objet a donné de bons résultat sur la sequence omnidirectionnelle. Mais les algorithmes sont loin d'être adaptés à la géométrie à cause de la concentration de points au centre. Aussi le temps de calcul est considérable. Pour cela, l'ajout d'un filtrage bayésien peut efficacement accélérer l'algorithme à savoir filtre de Kalman ou particulaire. Dans les méthodes basées sur l'apparence, les blocs utilisés dans les algorithmes de KLT (Figure. 2.5) , bloc matching, mean shift doivent être adaptés à la géométrie.

Dans le contexte omnidirectionnelle, les fenêtres utilisées dans la méthode mean shift doivent être adaptées (Figure 2.6)

Concernant la méthode des contours actifs, le résultat dans la figure 2.7 a montré une efficiente

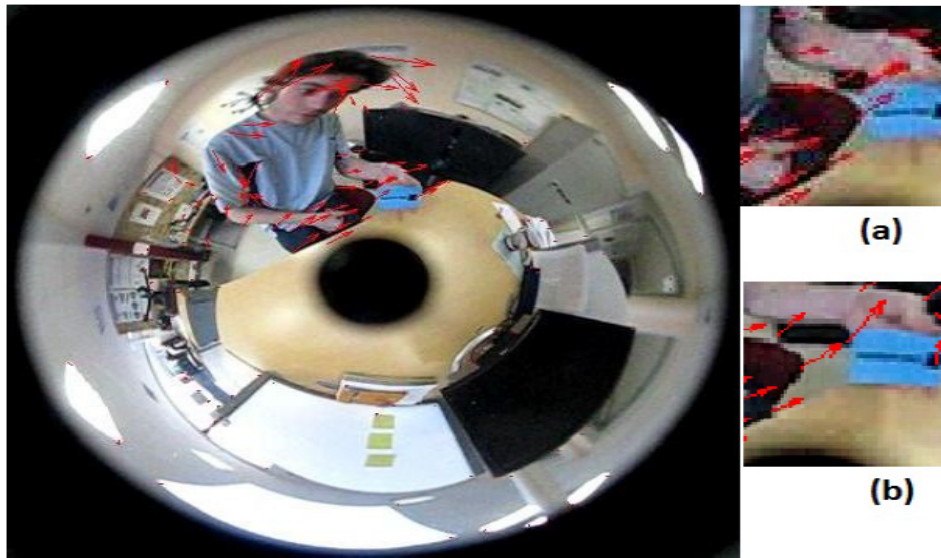


FIGURE 2.5 – Calcul du vecteur déplacement par la méthode KLT

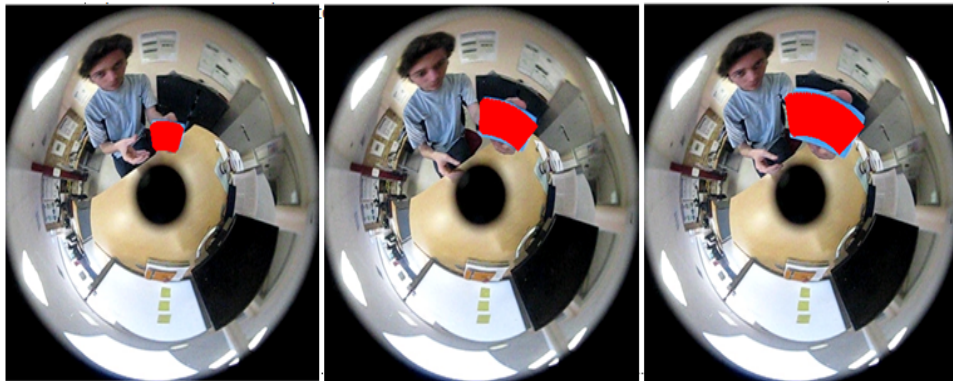


FIGURE 2.6 – Adaptation des noyaux mean shift.

de la méthode de manière pratique en terme de suivi de contour. Mais les opérateurs mis en jeux ne sont pas adaptés. Ce qui nous a amené à adapter la formulation mathématique de cette méthode à l'espace sphérique de l'image. Cette contribution sera bien détaillée dans le chapitre suivant.

2.5 Conclusion

Dans ce chapitre, nous avons fait une étude sur les méthodes les plus utilisées dans le suivi d'objet. Dans le contexte d'images omnidirectionnelle, nous avons cité les méthodes qui peuvent être adaptées et autres qui sont pratiquement efficaces mais théoriquement non adaptées. Dans les chapitres suivants, nous proposons des adaptations de suivi d'objet.

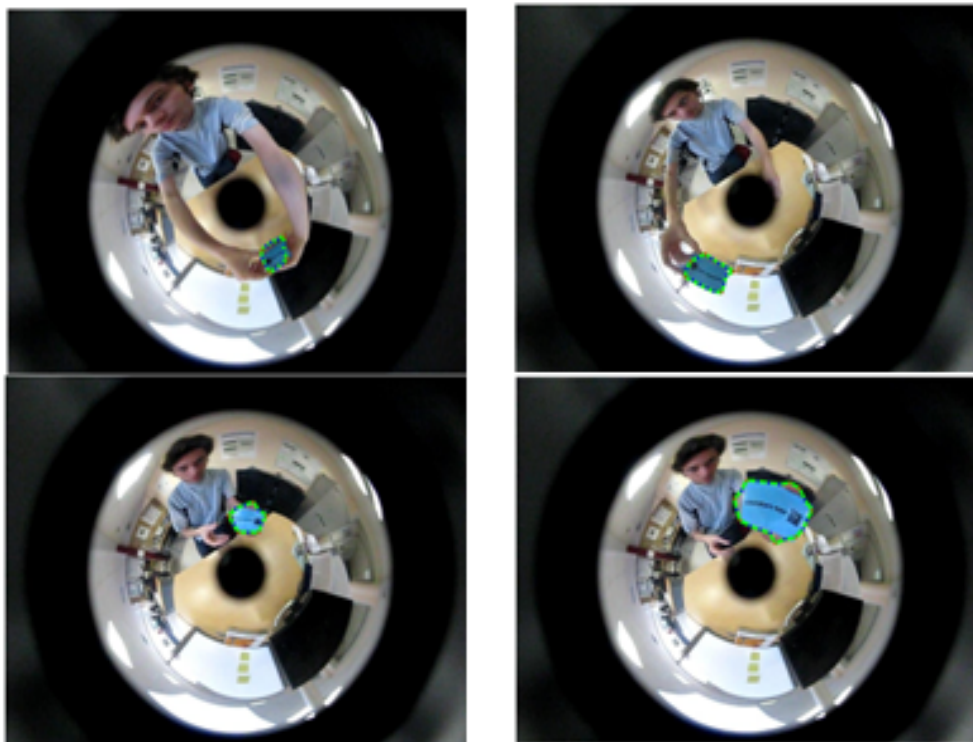


FIGURE 2.7 – Résultat de suivi de la méthode Snake

LE SUIVI SPHÉRIQUE PAR LES CONTOURS ACTIFS

Khald, A. Radgui, A. Rziza, Mohammed. (2014). Objects tracking in catadioptric images using spherical snake. VISAPP 2014 - Proceedings of the 9th International Conference on Computer Vision Theory and Applications. 3. 435-440.

3.1 Introduction

Dans un contexte de suivi, plusieurs chercheurs ont essayé de trouver une façon pour adapter le traitement de manière efficace dans le but de résoudre le problème de la géométrie déformée. En revanche, l'application des méthodes classiques directement sur telles images est délicate. Ce résultat est dû à l'information perdue dans l'image sur le centre et sur la périphérie. Pour remédier à ce problème, nous avons utilisé une méthode basée sur le suivi de points qui s'applique sur les points de contour et minimise la distance entre eux.

En se basant sur le calcul du gradient de l'image, les points se déplacent vers le contour de l'objet. Cette opération est répétée à chaque frame de la séquence. Le contour des points final dans le frame courant sera l'initial du frame suivant. Cette méthode appelée "les contours actifs". Et elle sera adaptée en faisant une projection des différents opérateurs mathématiques sur une demi sphère dans le but de réaliser un suivi sphérique.

3.2 Contours actifs classiques

Dans le contexte de suivi dans une séquence d'image perspectives. Et par le biais des contours actifs, l'utilisation de l'algorithme glouton pour minimiser l'énergie d'un contour actif, comme proposée par Williams et Shah dans [116], est devenue une alternative assez fréquente à l'approche variationnelle. Williams et Shah, de manière classique comme Kass et al discrétisent l'expression de l'énergie totale par "les différences finies". Ce qui rend l'algorithme de glouton plus rapide puisqu'il est à chaque itération en $O(nm)$ au lieu de $O(nm^3)$ pour un contour constituant n points et un voisinage de points de taille m .

Dans notre travail, nous avons choisi de travailler avec l'algorithme glouton pour les raisons suivantes :

- Un algorithme facile à appliquer.
- Un algorithme rapide
- Un algorithme qui peut être utilisé à des problèmes d'optimisation combinatoires et NP-complet.

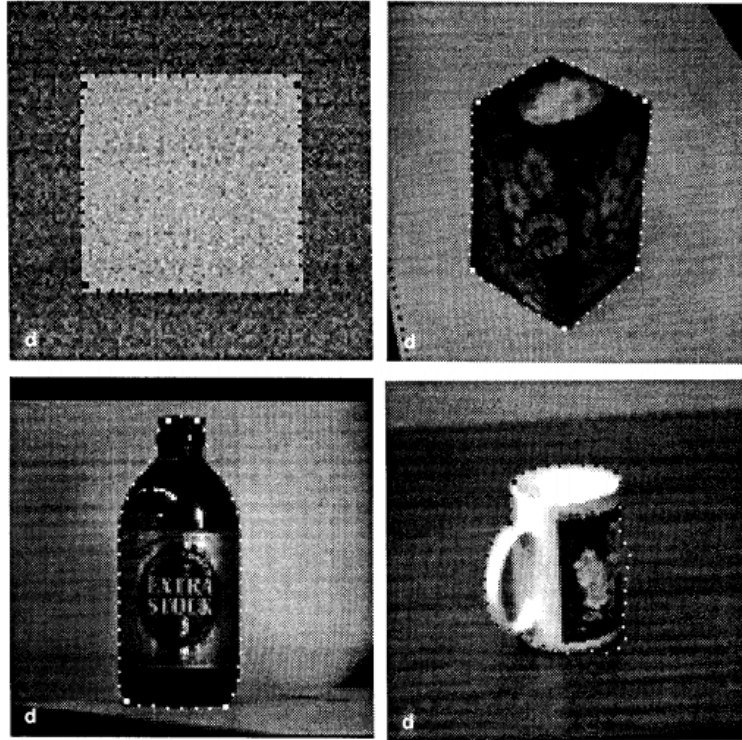


FIGURE 3.1 – Détection de Quatres types d’objets par l’algorithme Glouton

			Algorithme G			Approche variationnelle	
Image	Size	Th.	Secs.	Iter.	Cor	Secs.	Iter.
Square	58	3	0.250	2	3	0.350	4
Box	56	1	1.867	15	5	0.466	4
Bottle	50	1	1.217	11	4	1.499	12
Cup	46	1	0.700	7	3	0.300	4

TABLE 3.1 – Comparaison entre glouton et l’approche variationnelle.

Les résultats du tableau 3.1 montrent la grande concurrence entre les deux algorithmes. Dans les quatre types d’images (3.1), chaque algorithme montre sa robustesse, mais l’approche glouton est très efficace dans le cas de la détection du coins de l’objet.

3.2.1 Calcul des énergies

La minimisation de l’énergie revient à minimiser la distance entre les points du contour. Pour éviter la grande rétraction entre eux, Williams et Shah dans [116] ont proposé d’utiliser la différence de distance entre points par rapport à la distance moyenne.

Les fonctions d’énergies sont données par la formule suivante :

$$\mathbf{E}_{\text{Totale}} = \sum_{i=1}^N (\mathbf{a} * \mathbf{E}_{\text{continuit}}(\mathbf{P}_i) + \mathbf{b} * \mathbf{E}_{\text{courbure}}(\mathbf{P}_i) + \mathbf{c} * \mathbf{E}_{\text{gradient}}(\mathbf{P}_i) + \mathbf{d} * \mathbf{E}_{\text{ballon}}(\mathbf{P}_i)) \quad (3.1)$$

$\mathbf{P}_i = t(x_i, y_i)$ $i = 1..N$ sont les points du snake. a , b , c et d sont les coefficients attribués à

chaque énergie. On distingue deux types d'énergies, une énergie de continuité et une énergie de courbure. L'énergie de continuité est définie par la formule suivante :

$$E_{\text{continuité}} = \left| \text{distance}_{\text{moyenne}} - \sqrt{(\mathbf{x}_i - \mathbf{x}_{i-1})^2 - (\mathbf{y}_i - \mathbf{y}_{i-1})^2} \right| \quad (3.2)$$

Où $\text{distance}_{\text{moyenne}}$ est la moyenne des distances entre deux points successives du contour actif. Il est clair que si on minimise cette énergie, le point P_i doit se positionner à une distance égale à $\text{distance}_{\text{moyenne}}$ du point P_{i-1} .

L'énergie de courbure est donnée par la formule suivante :

$$E_{\text{courbure}} = \sqrt{(\mathbf{x}_{i-1} + 2\mathbf{x}_i + \mathbf{x}_{i+1})^2 - (\mathbf{y}_{i-1} + 2\mathbf{y}_i + \mathbf{y}_{i+1})^2} \quad (3.3)$$

3.2.2 Algorithme général

L'algorithme général de glouton est le suivant :

- Initialiser $Seuil, \alpha, \beta, \gamma$.
- Pour $i = 0$ à n [n : nombre de points du contour]
- $E_{min} < BIG$
 - pour $j = 0$ à $m-1$ [m : Taille du voisinage]
 - $E_j = \alpha_i E_{\text{cont},j} + \beta_i E_{\text{curv},j} + \gamma_i E_{\text{image},j}$
 - Si $E_j \leq E_{min}$
 - $E_{min} = E_j$
 - $j_{min} = j$
 - Se déplacer vers la position j_{min}
 - $c_i = \left\| \frac{u_i}{|u_i|} - \frac{u_{i+1}}{|u_{i+1}|} \right\|^2$
 - pour $i = 1$ à $n-1$
 - Si ($c_i \geq c_{i+1}$ et $c_i \geq c_{i-1}$ et $c_i \geq Seuil$)
 - alors $\beta_i = 0$ [test pour faire apparaître les coins de l'objet et critère d'arrêt]

3.2.3 Algorithme de suivi

Dans la première étape, nous avons réalisé un pré traitement. Pour cela nous avons appliqué un filtre gaussien pour réduire le bruit. Ce dernier a une grande influence sur notre contour. C'est un effet de lissage.

L'initialisation est manuelle dans la première image afin de détecter notre objet. Par contre, il existe des différentes méthodes basées sur la détection d'objet comme l'estimation de mouvement. Dans notre projet nous avons adopté une soustraction de fond.

Par la suite, nous avons appliqué notre algorithme sur les points du contour, et à chaque itération nous conservons les positions des points finaux pour les utiliser dans l'itération suivante et ainsi de suite jusqu'à la dernière image de la séquence. Les étapes de l'algorithme de suivi sont présentées dans l'organigramme.

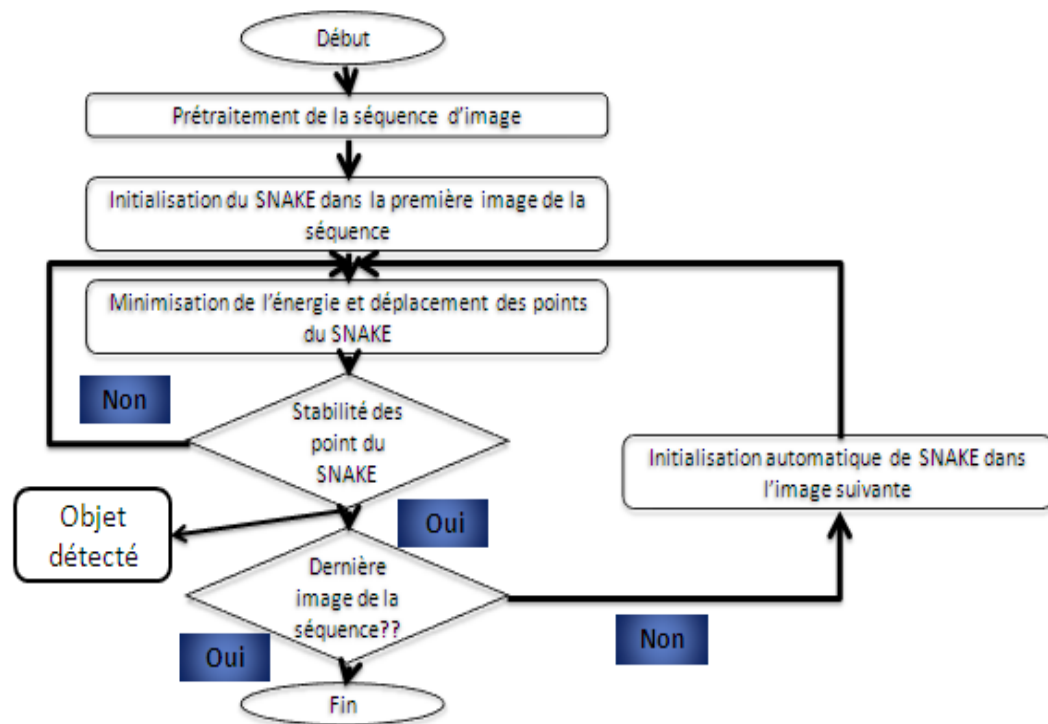


FIGURE 3.2 – Organigramme des contours actifs

3.3 Résultats dans le cas perspectif

3.3.1 Détection d'objet

Dans la phase d'initialisation des points de contour, nous avons fait le tour sur plusieurs méthodes qui permettent la détection automatique d'objet comme la méthode de l'estimation de mouvement ou celles basées sur la phase ou sur les énergies ou autres. Dans notre contexte nous avons choisis de travailler avec la soustraction de fond au raison de simplicité et temps de calcul très faible. Cette méthode consiste à faire une soustraction entre la première image de la séquence et l'image modèle qui ne contient pas d'objet.



FIGURE 3.3 – Images template et image avec objet

La figure 3.3 représente la séquence "Walk" qui contient 90 images caractérisée par une personne qui se déplace dans une scène. Comme résultat de la détection considéré comme l'initialisation des point du contour, nous avons procédé à la détection du contour de l'objet par le filtre de

Sobel. La figure 3.4 montre les résultats de la détection d'objet et de la sélection du contour.

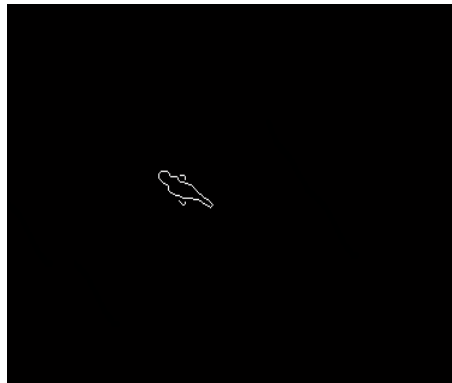


FIGURE 3.4 – Résultat de la détection d'objet par soustraction du fond (droite) et la détection du contour (gauche)

Les résultats sont satisfaisantes et correspondent parfaitement à notre besoin, une fois la détection d'objet dans la première image est faite, nous avons procédé à une initialisation des points du contours aux alentours du contour de l'objet détecté.

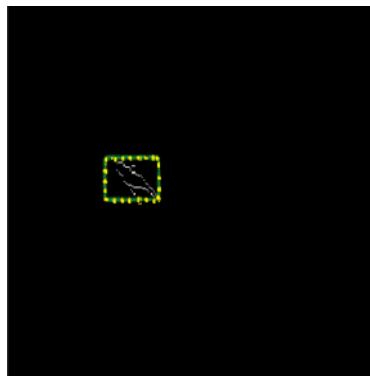


FIGURE 3.5 – Initialisation des points du contour

Après la détection et l'initialisation comme montre la figure 3.5, nous avons appliqué notre algorithme du suivi. C'est l'algorithme glouton pour la minimisation des énergies des points du contour.

3.3.2 Suivi d'objet

Dans le suivi, nous appliquons la méthode glouton pour la minimisation de la fonction de l'énergie totale selon l'algorithme décrit ci-dessous.

- Pour $i=2$:nbrimage
 - Pour chaque point du contour :
 - Initialise $\alpha, \beta, \gamma, \sigma$.
 - Calcule l'énergie image.
 - Calcule la distance moyenne entre les points du contour.
 - Calcule énergie continuité, courbure.

- normalisation des énergies.
- Chercher dans le voisinage.
- sélectionner le point du voisinage qui minimise l'énergie.
- déplacement vers ce point.
- Calcule la distance moyenne entre les points du contour.
- fin pour.
- fin pour.

Le choix des paramètres α, β, γ dépend de la séquence d'images utilisée. Il n'y a pas d'autres façons d'y remédier que de faire plusieurs tests sur la séquence pour avoir la bonne convergence de l'algorithme. La figure 3.6 montre l'évolution des points du contour pour arriver au contour de notre objet.

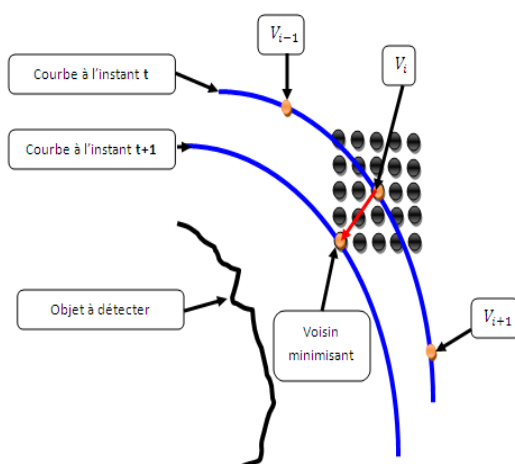


FIGURE 3.6 – Évolution de la courbe du contour

La raison pour laquelle nous avons normalisé les énergies est de mettre toutes les valeurs des énergies dans la même échelle avant de faire la sommation pondérée entre elles. Les valeurs que nous avons obtenue pour l'énergie d'image (Résultat du filtre de Sobel) sont de l'ordre de 100. Pour les énergies de continuité et de la courbure, les valeurs sont entre -1 et 1. La normalisation adoptée consiste à mettre toutes les valeurs de l'énergie entre 0 et 1. Pour cela nous avons divisé l'énergie de continuité et de la courbure par la valeur maximale. Pour une valeur X de l'énergie d'image, nous faisons le changement de variable comme montré dans l'équation (3.4).

$$\text{norm}(X) = \frac{X - \min(X)}{\max(X) - \min(X)} \quad (3.4)$$

Dans l'algorithme glouton, le voisinage du point du snake est défini par une fenêtre de taille 5. Dans notre base de données, nous utilisons deux sources de données : La séquence "WALK" de 90 images et la séquence "CUP" de 60 images. Les résultats du suivi, avec $\alpha = 1.2, \beta = 1, \gamma = 1.2$ sont illustrés dans les figures 5.5 et 3.8.



FIGURE 3.7 – Résultat du suivi pour la séquence WALK, avec $\alpha = 1.2, \beta = 1, \gamma = 1.2$ et $\sigma = 1$



FIGURE 3.8 – Résultat du suivi pour la séquence CUP, avec $\alpha = 1.2, \beta = 1, \gamma = 1.2$ et $\sigma = 1$

Les résultats obtenus correspondent parfaitement à notre besoin et ils sont satisfaisants. Pour la séquence "WALK" nous avons obtenu un temps de détection pour la première image de 1.05 seconde et un temps de suivi moyen pour les autres images de 0.43s. Pour la séquence "CUP" nous avons obtenu un temps de détection de 0.51s et un temps de suivi de 0.41s sachant que le traitement a été fait dans un environnement de Pc Core2duo 2.4G, 2G de mémoire et 1G de carte graphique qui sont des caractéristique moyennes.

3.3.3 Synthèse

Le suivi dans les images perspectives a donné de bons résultats en terme de poursuite de contour de l'objet à chaque instant et pour chaque image de la séquence par le biais des contours actifs. Cette méthode rencontre plusieurs limitations, Parmi elles, on trouve d'une part le choix des paramètres $\alpha, \beta, \gamma, \sigma$ que nous pouvons résoudre par plusieurs expérimentations ce qui demandent du temps. D'autre part, nous pouvons conserver l'erreur si elle a eu lieu dans le suivi puisque notre méthode utilise les points du contour de l'étape précédente.

Dans un autre contexte, nous pouvons améliorer l'algorithme pour qu'il soit capable de détecter plusieurs objets en même temps en utilisant les Topologie–Snake et qu'il soit robuste dans les grandes déplacements par le biais du MultiScale–Snake.

Les avantages de l'algorithme utilisé réside dans le fait que c'est une méthode extensible et permet de savoir la forme de l'objet au cours de son déplacement, ainsi, les contours actifs sont indépendants de changement de la taille et de la forme de l'objet.

3.4 Voisinage Sphérique

Dans un système catadioptrique, chaque image formée par le principe de point de vue unique ([47],[94],[95]) est modélisée par une image sphérique illustrée dans la figure 3.9. Cette présentation est nommée modèle de projection unifié et introduit par Dannilidis dans [47].

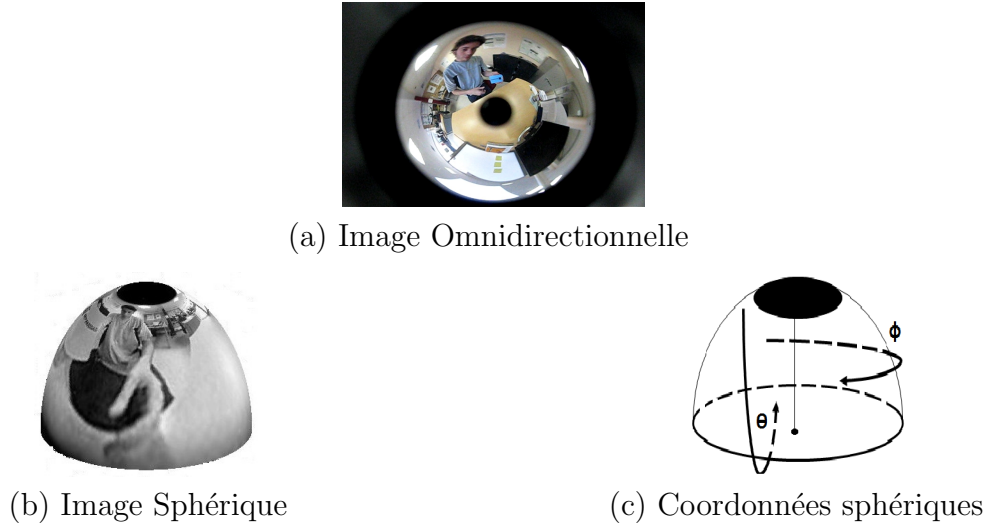


FIGURE 3.9 – La géométrie des images omnidirectionnelle

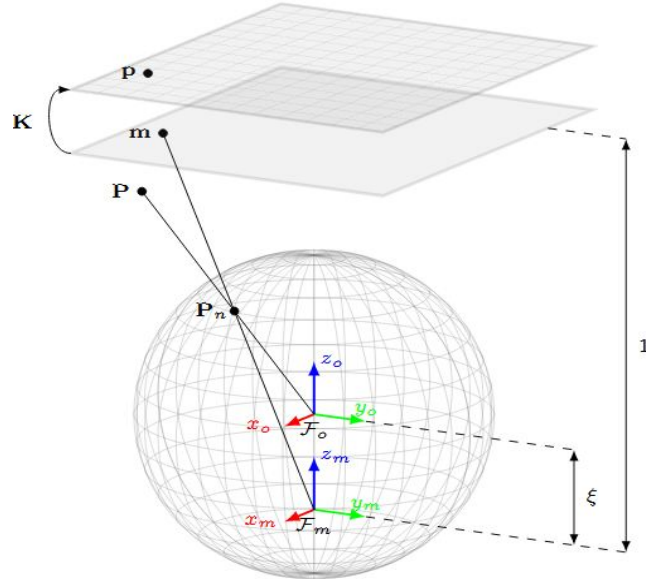


FIGURE 3.10 – Le modèle de projection unifié [47].

Le point P est le point 3D dans l'espace autour de la caméra. Le point P_n est l'intersection entre la droite passant par P et le centre \mathcal{F}_0 et la surface de la sphère d'équivalence. Ensuite, En utilisant le paramètre ϵ qui représente la focale de la caméra et qui varie entre 0 et 1, on détermine le point $m(x, y, 1)$ sur le plan de projection avec la droite passant par le point P_n et le point $\mathcal{F}_m : (x_m, y_m, z_m - \epsilon)$.

Après ceci, le point p est obtenu en utilisant la relation $p = K * m$. Avec K est la matrice 3×3 qui contient les paramètres intrinsèques de la caméra. Les valeurs de K et ϵ sont obtenus par

l'opération de calibrage de la caméra. Pour notre cas, $K = f(\alpha_u, \alpha_v, u_0, v_0)$ comme présenté dans l'équation 3.5.

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} K_u & 0 \\ 0 & K_v \end{bmatrix} * \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \quad (3.5)$$

Avec K_u et K_v sont respectivement la taille horizontale et verticale d'un pixel. u_0 et v_0 sont les coordonnées de la projection de l'axe optique de la caméra (celui-ci est confondue avec l'axe de symétrie du miroir). La procédure d'étalonnage permet donc d'estimer $\alpha_u = \epsilon.K_u$ et $\alpha_v = \epsilon.K_v$ qui représentent les inclinaisons de la caméra.

Dans ce travail, on considère que notre caméra est déjà calibrée, et le miroir utilisé est de forme parabolique. C'est à dire que $\epsilon = 1$. Le point sphérique P_n est caractérisé par les coordonnées sphériques comme montré dans l'équation (3.6) :

$$\begin{cases} X_s = \cos(\varphi) \sin(\theta) \\ Y_s = \sin(\varphi) \sin(\theta) \\ Z_s = \cos(\theta) \end{cases} \quad (3.6)$$

La projection stéréographique de P_n de la sphère vers le plan catadioptrique est exprimée dans l'équation (3.7) :

$$\begin{cases} x = \frac{X_s}{1 - Z_s} \\ y = \frac{Y_s}{1 - Z_s} \end{cases} \quad (3.7)$$

En utilisant les équations (3.6) et (3.7), on obtient le point $p(x, y)$ exprimé par les coordonnées sphérique comme montré dans l'équation (3.8).

$$\begin{cases} x = \cot \frac{\theta}{2} \cos(\varphi) \\ y = \cot \frac{\theta}{2} \sin(\varphi) \end{cases} \quad (3.8)$$

Avec θ est l'attitude de valeur comprise entre 0 et π , et φ est la longitude dont la valeur est comprise entre 0 et 2π . Chaque localisation d'un point sur la sphère est défini par les coordonnées (θ, φ) .

Le voisinage sphérique exprimé en fonction des angles θ et ϕ comme le montre l'équation 3.9

$$N_s = \left(|\delta\theta| \leq \frac{1}{N}, 2\pi - \frac{1}{M} \leq \delta\varphi \leq \frac{1}{M} \right) \quad (3.9)$$

La taille du voisinage sphérique est variable, c'est à dire que les intervalles de $\delta\theta$ et $\delta\phi$ change de nombre de points en fonction de la position du pixel dans l'image. Cette caractéristique est très importante dans le traitement et aussi dans la représentation des bloc de voisinage. La figure 3.11 représente différents blocs avec les mêmes valeurs de $\delta\theta$ et $\delta\phi$.

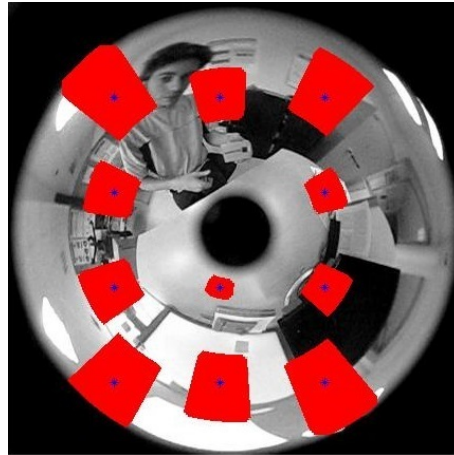


FIGURE 3.11 – Voisinage Sphérique

3.5 Contours actifs sphériques

Dans notre travail, nous définissons le voisinage sphérique exprimé dans la section 3.4. Dans l'algorithme des contours actifs, le voisinage des points du contour est présenté par des blocs de taille 5 comme illustré dans les figures 3.13 et 3.12 en mettant en œuvre le voisinage au périphéries et au centre de l'image.

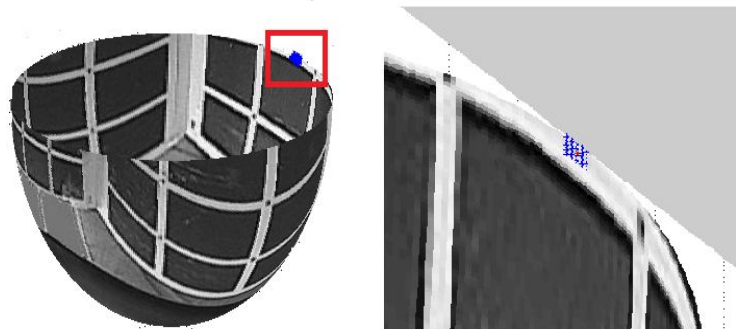


FIGURE 3.12 – Voisinage Sphérique au périphéries

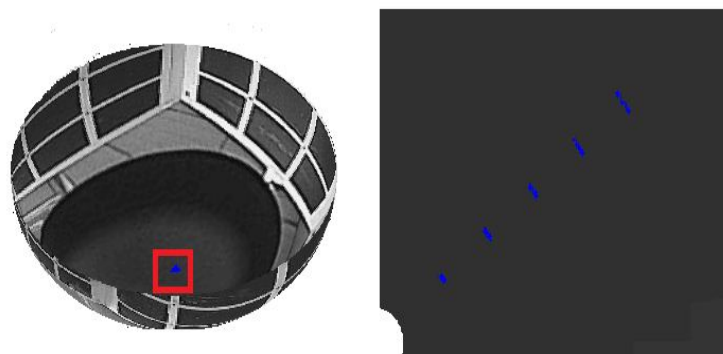


FIGURE 3.13 – Voisinage Sphérique au centre

L'équivalence image-sphère est très intéressante dans la mesure où elle permet de changer d'espace pour effectuer certains traitements en tenant compte des déformations. De nombreux auteurs ont ainsi utilisé ce changement d'espace afin de réaliser la dérivée d'une image omnidirectionnelle, le calcul de pose du capteur ou encore la détection de droites catadioptriques. C'est dans cette même optique que Démonceaux et Vasseur [17] ont proposé d'utiliser la sphère d'équivalence afin de déterminer le voisinage le plus approprié dans le cadre d'images catadioptriques centrales. Dans notre contribution, nous avons fait l'équivalence image-sphère dans une demi-sphère nord. La figure 3.14 montre notre choix de projection.

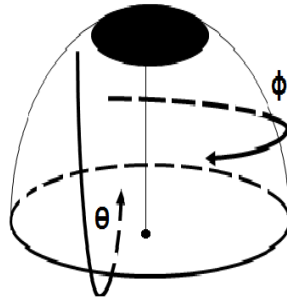


FIGURE 3.14 – Les angles sphérique θ et φ dans une demi-sphère

Par la suite, nous avons appliqué la méthode des contours actifs sur une séquence d'images omnidirectionnelles représentant un objet qui se déplace dans une scène 3D projetée sur une demi-sphère. l'équivalence entre le déplacement de l'objet dans le plan de l'image omnidirectionnelle et sa projection sphérique est présentée par la figure 3.15.

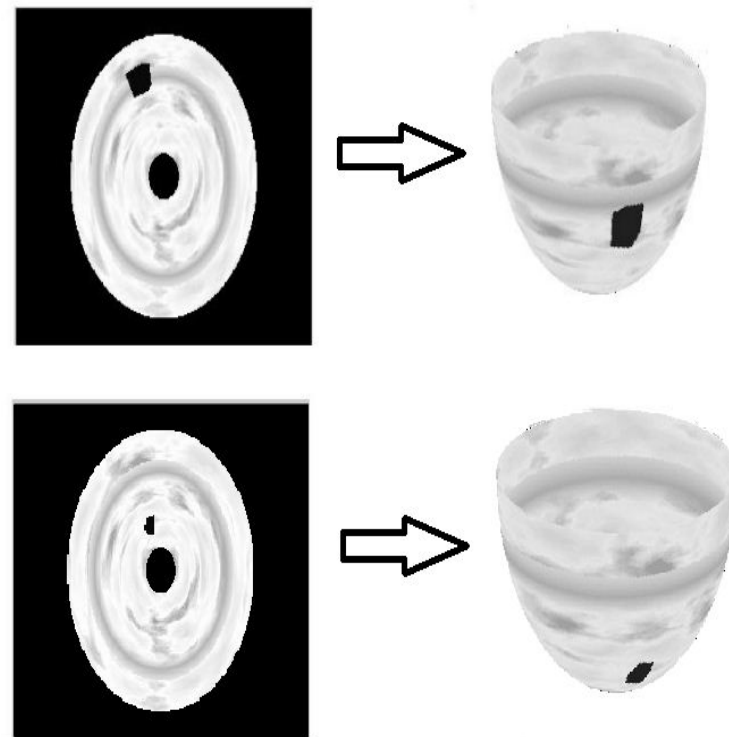


FIGURE 3.15 – Équivalence entre image omni et sa projection sphérique avec le déplacement de l'objet

Dans ce cas de figure, nous remarquons que lorsque l'objet se rapproche de la caméra, sa taille augmente et sa forme change.

3.5.1 Adaptation sphérique

L'adaptation du suivi dans les images omnidirectionnelles est nécessaire vu les contraintes de changement de taille et de forme de l'objet. Elle revient à adapter les traitements réalisés dans le cas perspective à la géométrie des images omnidirectionnelles. Pour cela, nous citons les différents opérateurs utilisés dans cet algorithme. Rappelons les différents étapes de suivi :

- Pré traitement de la séquence d'images : un filtre gaussien est appliqué pour réduire le bruit et pour le lissage.
- La détection d'objet : cette phase représente l'étape de détection d'objet par soustraction du fond et l'initialisation des points su contour actif.
- Le suivi d'objet : Nous appliquons l'algorithme de minimisation et le déplacement des points pour chaque image de la séquence.

3.5.2 Pré traitement de la séquence d'images sphériques

Dans cette phase, nous avons utilisé un filtre gaussien basé sur la dérivation sphérique défini par Danilidis [47] et présenté dans le chapitre précédent. Dans la sphère, nous avons appliqué un produit de convolution entre une gaussienne sphérique et notre image sphérique. Commençons par définir le produit de convolution classique :

$$f * g(x) = \int f(x - y) g(y) dy \quad (3.10)$$

par équivalence, un décalage dans le plan correspond à une rotation dans l'espace de la sphère, ce qui nous permet d'écrire :

$$(I * G)(P) = \int f(Rn_0) g(R^{-1}P) dR \quad (3.11)$$

avec :

$$\mathbf{G}_s(\theta, \varphi, \mathbf{t}) = e^{-\left(\frac{\cot^2(\frac{\theta}{2})}{\mathbf{t}}\right)} \quad (3.12)$$

une gaussienne centrée d'écart type σ .

et $n_0 = (0, 0, -1)$: le pôle sud du demi-sphère.

et R la matrice de rotation Défini par le produit des matrices suivantes et qui représente des différentes rotations des points de la sphère suivant les axes X, Y, et Z :

$$R_x(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{pmatrix} \quad (3.13)$$

$$R_y(\theta) = \begin{pmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{pmatrix} \quad (3.14)$$

$$R_{\mathbf{z}}(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (3.15)$$

Ce vecteur est la solution de l'équation :

$$R\mathbf{u} = \mathbf{u} \quad (3.16)$$

$\vec{u} \begin{pmatrix} u_x \\ u_y \\ u_z \end{pmatrix}$ (donc avec $u_x^2 + u_y^2 + u_z^2 = 1$) et d'un angle θ . La formule est :

$$R = \begin{pmatrix} u_x^2 + (1 - u_x^2)c & u_x u_y(1 - c) - u_z s & u_x u_z(1 - c) + u_y s \\ u_x u_y(1 - c) + u_z s & u_y^2 + (1 - u_y^2)c & u_y u_z(1 - c) - u_x s \\ u_x u_z(1 - c) - u_y s & u_y u_z(1 - c) + u_x s & u_z^2 + (1 - u_z^2)c \end{pmatrix} \quad (3.17)$$

où

$$c = \cos \theta, \quad s = \sin \theta \quad (3.18)$$

Si l'espace en 3D est orienté de façon conventionnelle, cette rotation se fera dans le sens inverse aux aiguilles d'une montre pour un observateur placé de telle sorte que le vecteur directeur \vec{u} pointe dans sa direction (règle de la main droite). Cette formule peut être simplifiée comme montre l'équation (3.19).

$$R = P + (I - P) \cos \theta + Q \sin \theta \quad (3.19)$$

Avec

$$P = \begin{pmatrix} u_x^2 & u_x u_y & u_x u_z \\ u_x u_y & u_y^2 & u_y u_z \\ u_x u_z & u_y u_z & u_z^2 \end{pmatrix} = \mathbf{u} \mathbf{u}^t \quad (3.20)$$

$$I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (3.21)$$

$$Q = \begin{pmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{pmatrix} \quad (3.22)$$

3.5.3 Détection sphérique de l'objet

Dans la détection d'objet, nous avons utilisé la méthode de soustraction du fond sphérique permettant la détection automatique d'objet sur la sphère. Comme résultat de détection de l'objet, nous avons obtenu comme illustré dans la figure 3.16 :

Nous utilisons les points du contour de l'objet détecté comme une initialisation du contour actif. Ensuite, nous appliquons l'algorithme de minimisation.



FIGURE 3.16 – Résultat de la détection

3.5.4 Suivi sphérique de l'objet

✓ Energies

Dans notre algorithme, nous avons calculé les énergies d'image, de continuité, et de courbure. Pour l'énergie d'image sphérique, nous appliquons un filtre de sobel sphérique pour la détection du contour. Avec un cout très faible et une implémentation facile, nous pouvons appliquer le filtre de Sobel sur une image perspective pour avoir un contour bien localisé. **Sobel** dans une conférence en 1968 et **prewit** en 1970 ont proposé des filtres basés sur des dérivés suivants θ et φ . L'approximation de $\frac{\partial I}{\partial \theta}$ donnée par le filtre :

$$\frac{1}{8} \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} \quad (3.23)$$

L'approximation de $\frac{\partial I}{\partial \varphi}$ donnée par le filtre :

$$\frac{1}{\sin^2(\theta_i)} * \frac{1}{8} \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \quad (3.24)$$

appliquée à la matrice $I(\theta_i, \varphi_i)$. En pratique, **Danilidis**, **Makadia** et **Bullow** ont effectué un changement de variables défini par :

$$I(\theta, \varphi) = I(u(\theta, \varphi), v(\theta, \varphi)) \quad (3.25)$$

$$\frac{\partial I}{\partial \theta} = \frac{\partial I}{\partial u} * \frac{\partial u}{\partial \theta} + \frac{\partial I}{\partial v} * \frac{\partial v}{\partial \theta} \quad (3.26)$$

$$\frac{\partial I}{\partial \varphi} = \frac{\partial I}{\partial u} * \frac{\partial u}{\partial \varphi} + \frac{\partial I}{\partial v} * \frac{\partial v}{\partial \varphi} \quad (3.27)$$

avec :

$$u(\theta, \varphi) = \cotg\left(\frac{\theta}{2}\right) * \cos(\varphi) \quad (3.28)$$

$$v(\theta, \varphi) = \cotg\left(\frac{\theta}{2}\right) * \sin(\varphi) \quad (3.29)$$

comme nous avons déjà cité dans la section des traitements des images omnidirectionnelles, l'adaptation dans l'espace de la sphère est représentée par les résultats suivants :

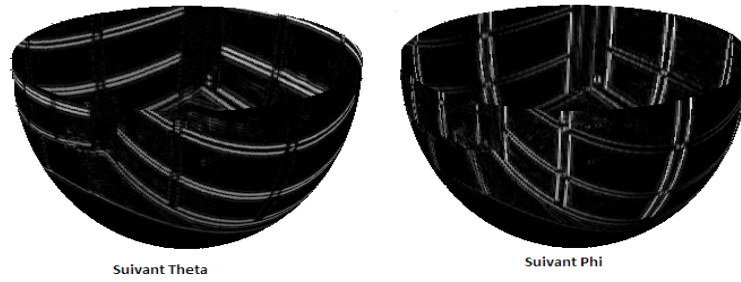


FIGURE 3.17 – Filtre de Sobel sphérique : suivant θ (gauche),suivant φ (droite)

C'est l'énergie d'image présentée par le gradient sphérique par le biais d'un filtre de sobel en fonction des angles θ et φ . Et comme résultat du sobel sphérique, nous obtenons :

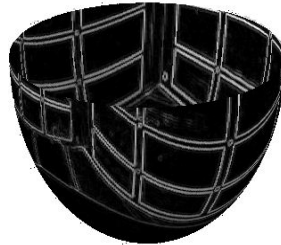


FIGURE 3.18 – Sobel sphérique.

Voici quelques exemples du sobel sphérique appliqué sur des images omnidirectionnelles projetés sur l'espace du demi-sphère.

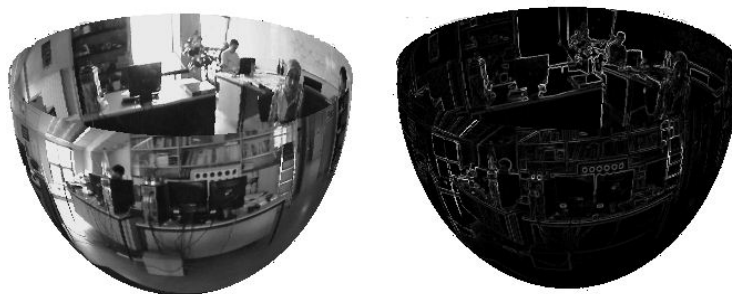


FIGURE 3.19 – Sobel sphérique de l'image "Laboratoire".

Pour l'énergie de continuité sphérique, elle dépend de la distance moyenne entre les points sphériques $P(X, Y, Z)$ et les coordonnées des points du contour actif.

$$E_{contS} = \left| \text{distance}_{moyenneS} - \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2} \right| \quad (3.30)$$

L'énergie de la courbure sphérique est la suivante :

$$E_{courbS} = \sqrt{(x_{i-1} + 2x_i + x_{i-1})^2 + (y_{i-1} + 2y_i + y_{i-1})^2 + (z_{i-1} + 2z_i + z_{i-1})^2} \quad (3.31)$$

Pour les calculs de distances dans la sphère (distance moyenne entre points du contour), nous faisons appel aux coordonnées X , Y et Z puisque dans la sphère, les distances entre les points du contour sont petites, donc nous pouvons les considérer comme linéaire au lieu de travailler sur les angles sphériques pour raison de faciliter les calculs. Donc, notre algorithme se base sur une minimisation d'énergies (image, continuité et courbure). L'énergie Totale des contours actifs sphérique est de la forme :

$$\mathbf{E}_{\text{TotaleS}} = \sum_{i=1}^N (\mathbf{a} * \mathbf{E}_{\text{contS}}(\mathbf{P}_i(\theta, \varphi)) + \mathbf{b} * \mathbf{E}_{\text{courbS}}(\mathbf{P}_i(\theta, \varphi)) + \mathbf{c} * \mathbf{E}_{\text{gradS}}(\mathbf{P}_i(\theta, \varphi)) \quad (3.32)$$

Pour rappel, la minimisation de l'énergie totale des contours actifs est réalisée dans le voisinage de chaque point du contour, ainsi le déplacement vers le contour de l'objet.

$$E_{i_{\min}} = \text{argmin}(E_{i_{\text{TotaleS}}}(\mathbf{P}_i(\theta, \varphi)))_V \quad (3.33)$$

3.6 Résultats & synthèse

3.6.1 Images de synthèse

L'outil utilisé pour construire les images de synthèse est POV-Ray. POV-Ray est un logiciel gratuit, qui est dédié pour des applications de lancé de rayon, mais il contient les outils nécessaires pour construire des images de synthèse. POV-Ray ne dispose pas d'une interface graphique (de modeleur 3D) intégré comme la plupart des logiciels de synthèse actuels, mais il utilise des scripts de description de scène, dans lesquels tous les objets et les lumières doivent être décrits.

```

global_settings { max_trace_level 10 assumed_gamma 1.0 }

#include "colors.inc"
#include "shapes.inc"
#include "textures.inc"
#include "stones.inc"
#include "skies.inc"
#include "woods.inc"
#include "metals.inc"

// Essais avec la caméra
#declare k = 3.92; // paramètre du miroir
#declare u0 = 117.93; // Position de la projection
#declare v0 = 159.84; // de l'axe optique sur l'image
#declare Alpha = 195.72; // relatif à la taille des pixels

#declare Diametre = 0.045; // diametre du miroir
//#declare Diametre = 0.3;
#declare Hauteur = 0.0138; // hauteur du miroir
//#declare Hauteur = 0.002;
#declare c = -0.25*(k/Hauteur)*sqrt((k-2)/k)*(4*Hauteur*Hauteur/(k-2))
#declare a = c*0.5*sqrt((k-2)/k);
#declare b = c*0.5*sqrt(2/k);
#declare f = 0.0012;
#declare K_u = Alpha/f;
#declare DAN = 0.00393; // diametre de l'aiguille noire
#declare PosAN = a*sqrt(1+(DAN*DAN)/(4*b*b)) - c/2;

```

FIGURE 3.20 – POV-RAY

Avec ce logiciel, nous avons construit une séquence d'images omnidirectionnelles contenant la forme d'un cube qui se déplace.

La séquence d'image construite par POV-RAY en appliquant le code est illustrée dans la figure 3.20. La caméra utilisée est caractérisée par les paramètres k , Alpha , u_0 , et v_0 . La séquence est présentée dans la figure 3.21.

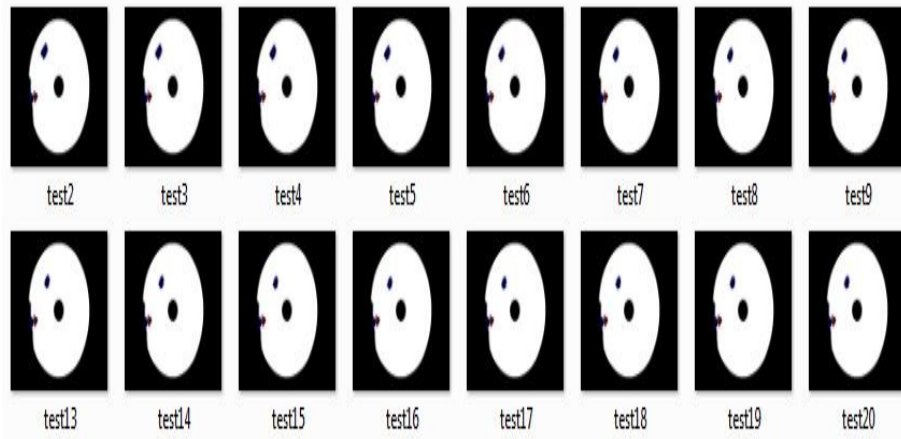


FIGURE 3.21 – Séquence d’images

3.6.2 La détection d’objet sphérique

Dans cette étape comme dans le cas perspectif, nous avons réalisé une initialisation automatique par le biais de la méthode de soustraction du fond suivi d’une détection du contour de l’objet pour pouvoir l’initialiser, comme illustré dans la figure 3.22.

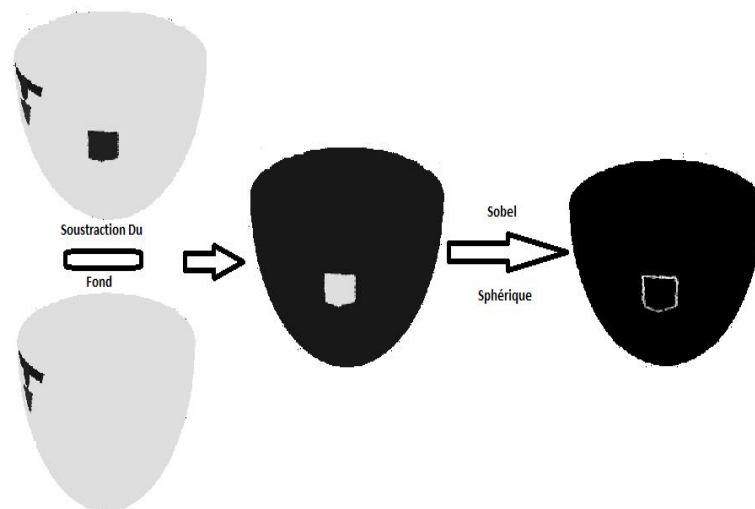


FIGURE 3.22 – Initialisation automatique par soustraction du fond et Filtre de Sobel sphérique

La figure 3.23 montre le contour initial, et le contour après 5 itérations par application de l’algorithme glouton pour suivre le contour de l’objet dans la première image de la séquence.

Dans cette étape, nous avons travaillé sur la première image de la séquence. Ensuite, le contour atteint pour une image par l’algorithme glouton est utilisé comme un contour initial de l’image suivante.

3.6.3 Le suivi sphérique

Nous avons appliqué notre algorithme sur deux séquences, une avec un cube et l’autre avec une sphère. Nous avons obtenue les résultats montrés dans les figures 3.24 et 3.25

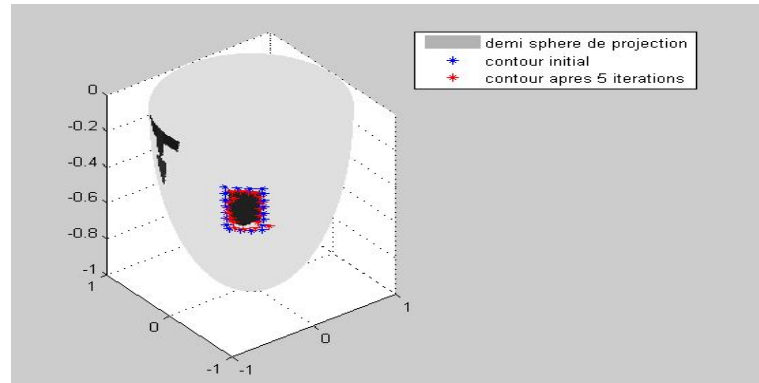


FIGURE 3.23 – Application de la minimisation sur la première image de séquence.

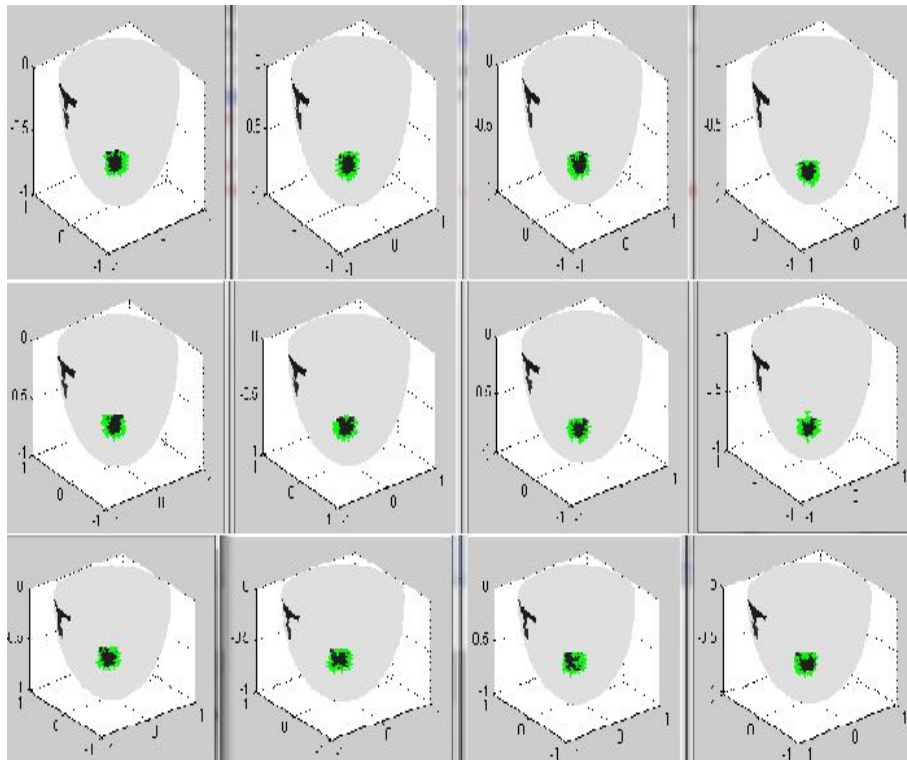


FIGURE 3.24 – Résultat de suivi Sphérique d'un cube

Et pour la séquence de la sphère, nous avons obtenue les résultats suivants :

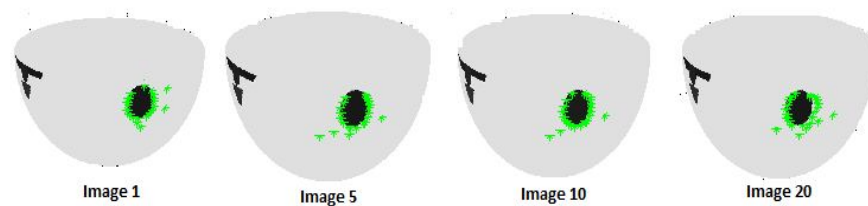


FIGURE 3.25 – application de la minimisation sur la première image de séquence.

Pour le suivi dans la séquence d'images omnidirectionnelle réelles, nous avons obtenue les résultats suivants :

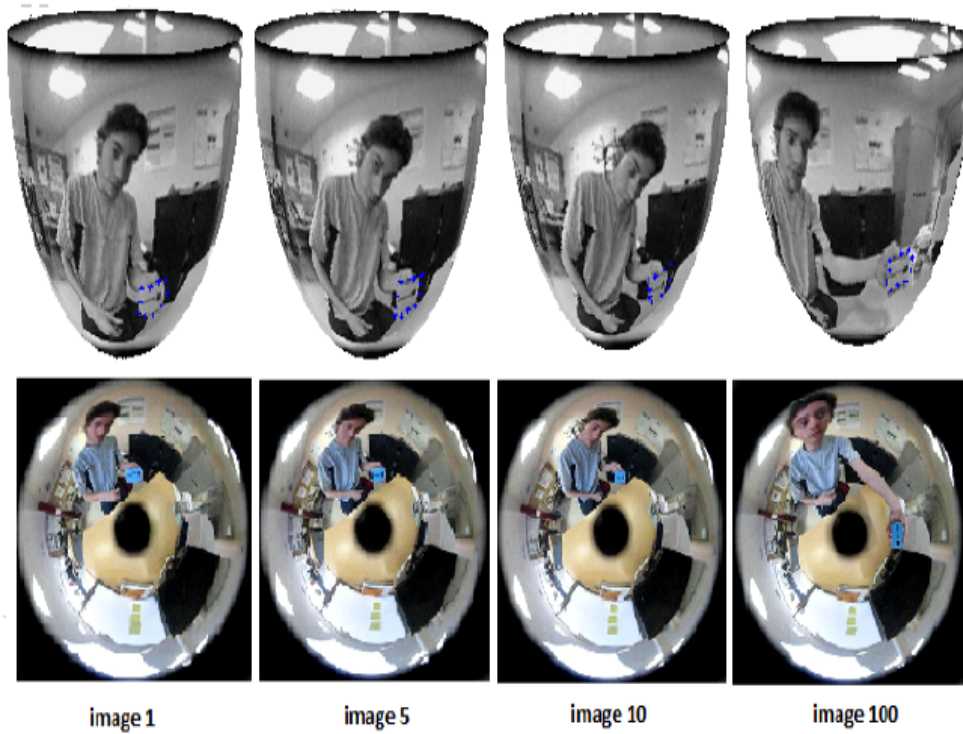


FIGURE 3.26 – Résultat du suivi sur des images réelles(1)

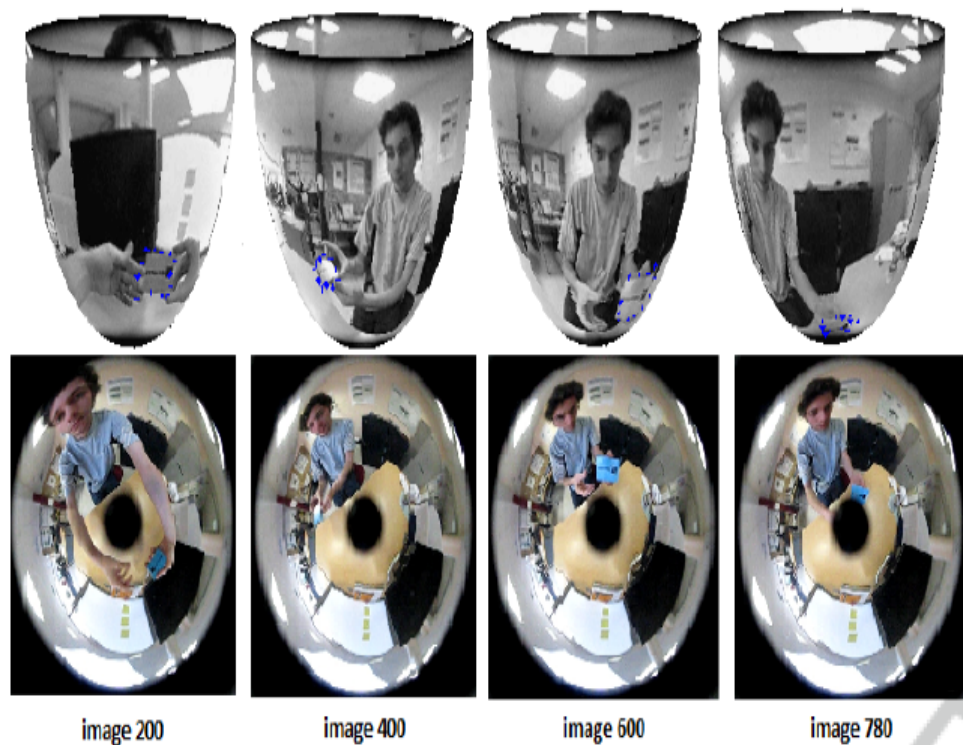


FIGURE 3.27 – Résultat du suivi sur des images réelles(2)

Dans le suivi sphérique, nous avons appliqué une détection d'objet par une soustraction du fond sur la première image. Ensuite, nous avons appliqué l'algorithme de minimisation sur le reste des images. Nous avons arrivé à bien suivre le contour de l'objet dans la demi-sphère faisant adapté le traitement et l'approche de projection choisi. Rappelons que nous avons travaillé sur le même

environnement du Pc. Le temps d'exécution est 0.41 s pour la détection et un temps de suivi moyenne de 0.34s. Nous pouvons dire que l'utilisation des contours actifs dans le suivi sphérique donne des résultats satisfaisantes en terme de détection de contour suivi, et de la convergence de l'algorithme utilisé pour la minimisation des énergies.

Dans l'algorithme greedy ou glouton, nous voulons bien préciser le critère d'arrêt. Dans la séquence du cube nous avons opté pour une détection de coin basée sur un seuillage de l'énergie image et de la courbure de contour en mettant le coefficient de rigidité à zéro.

3.7 Conclusion

Le suivi effectué par les contours actifs est l'une des méthodes les plus utilisées dans le domaine de la vidéo surveillance car c'est une méthode extensible et permet de connaître la forme et l'orientation de l'objet au cours de son déplacement. Ainsi, elle est adaptée au changement de la taille de l'objet dans les images omnidirectionnelles. Sauf que cette méthode a des points faibles à savoir le choix des paramètres de l'énergie et la conservation d'erreur commise dans la minimisation. Nous pouvons résoudre le problème des grandes déplacements de l'objet par les méthodes multi-échelle et le problème de suivi des objets multiples en utilisant l'algorithme de multi-topologie.

LE SUIVI D'OBJET PAR LES MÉTHODES STATISTIQUES

A. Khald, A. Radgui and M. Rziza, "Spherical approach : Objects Tracking in Catadioptric System Using The GGD Distribution," 2019 International Conference on Intelligent Systems and Advanced Computing Sciences (ISACS), 2019, pp. 1-7, doi : 10.1109/ISACS48493.2019.9068921.

4.1 Le suivi d'objet en utilisant la distribution de la gaussienne généralisée

4.1.1 Introduction

Dans ce chapitre, Nous présentons une approche statistique pour un suivi d'objet basée sur une modélisation des blocs des images en utilisant les paramètres de la distribution de probabilité. Notre méthode est omniprésente et largement utilisée pour résoudre le problème du suivi de noyau. Cependant, le défi rencontré est la déformation existante dans les images qui présentent des distorsions radiales importantes dont on doit tenir compte durant le traitement.

Dans ce travail, nous proposons un suivi de noyau basé sur un système de pyramides pour remédier au problème de changement d'échelle. Ensuite, nous adaptons cet algorithme en utilisant les coordonnées sphériques sur l'image plane.

Pour le fonctionnement de notre algorithme, nous divisons l'image en plusieurs blocs en $(d\theta, d\phi)$ qui sont les coordonnées sphériques utilisés. Chaque bloc de l'image est modélisé par la distribution gaussienne généralisée (GGD). C'est à dire que chaque bloc où se trouve notre objet est modélisé par α et β qui représentent les paramètres estimés de la probabilité de distribution. Ensuite, durant le traitement et entre le bloc de l'image courante et les blocs candidats, un calcul sera fait pour la similarité dans une zone de recherche. Cette opération est faite pour toute les images de la séquence.

4.1.2 Le modèle matching

En utilisant l'algorithme du modèle matching, nous pouvons réaliser la détection d'objet dans la première image de la séquence. Dans ce cas, nous devons appliquer un algorithme robuste. C'est pour cette raison que l'algorithme de Lucas and Kanade [71] est le plus convenable dans notre travail malgré qu'il y a d'autres méthodes qui ont montré leurs robustesse en terme de rapidité et flexibilité. Dans [65] et [76], les auteurs proposent une nouvelle approche d'estimation du flot optique basée sur la décomposition en ondelettes sphériques. Cette méthode est plus robuste que la méthode de Lucas et Kanade. Mais dans le contexte de suivi d'objet, il est toujours nécessaire de respecter la contrainte du temps réel. Nous utilisons la méthode KL comme montré sur la figure. 4.1.

La méthode de Lucas Kanade indique que le pas de déplacement de l'objet est approximati-

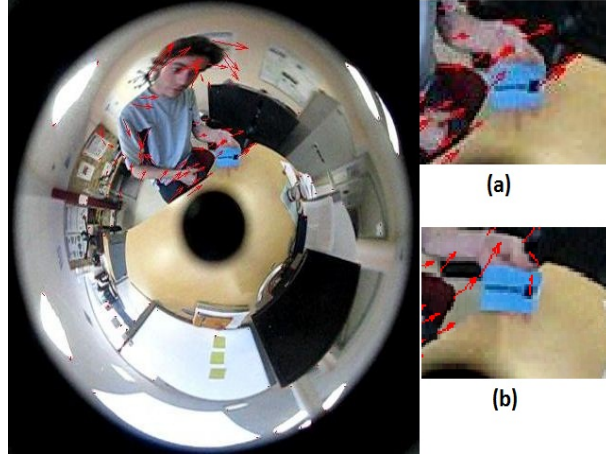


FIGURE 4.1 – Vecteurs du flot optique dans les images omnidirectionnelles : (a) : petit pas ; (b) : grand pas

vement petit et presque constant dans le voisinage d'un bloc d'images.

Ainsi, les équations du flot optique pour tous les points de la fenêtre nommée le vecteur de flot local d'image (V_x, V_y) sont présentée dans l'équation 4.1.

$$\begin{aligned}
 I_x(q_1)V_x + I_y(q_1)V_y &= -I_t(q_1) \\
 I_x(q_2)V_x + I_y(q_2)V_y &= -I_t(q_2) \\
 &\vdots \\
 I_x(q_n)V_x + I_y(q_n)V_y &= -I_t(q_n)
 \end{aligned} \tag{4.1}$$

Avec q_1, q_2, \dots, q_n sont les pixels internes, et $I_x(q_i), I_y(q_i), I_t(q_i)$ sont les dérivées partielles de l'image I avec la position x, y et le temps t évalués au point q_i à l'instant courant.

modèle matching classique Le modèle matching ou la mise en correspondance est un traitement haut niveau pour identifier une partie de l'image si elle correspond à une partie candidate appartenant à une autre image.

le modèle matching est une technique flexible et relativement robuste à utiliser. Son utilisation dans les applications de localisation et de suivi l'a rendu populaire. En revanche, son emploi est limité par la puissance de calcul et le temps d'exécution.

Dans ce travail, l'algorithme de correspondance de modèle est basé sur le calcul de la distance KLD entre deux distributions. Le but est de minimiser cette distance pour trouver le bloc candidat où se trouve notre objet.

Un modèle du bloc pour le suivi d'objet est généralement un algorithme de correspondance. Puisque notre objet s'approche et s'éloigne de la caméra, ce qui entraîne un changement de taille qui fait que l'utilisation des algorithmes classiques utilisant un bloc de taille fixe est non conforme à notre traitement. Notons que le modèle gaussien est une somme pondérée de M densités de composants. Il est présenté par l'équation 4.2.

$$P(x, \alpha, \beta) = \frac{\beta}{2\alpha\Gamma(\frac{1}{\beta})} e^{-(\frac{|x|}{\alpha})^\beta} \tag{4.2}$$

$\Gamma(\cdot)$ est la fonction définie par : $\Gamma(z) = \int_0^\infty e^{-t} t^{z-1} dt$. β est le paramètre de forme qui est

inversement proportionnel au taux décroissant du pic, et α est le paramètre d'échelle qui représente la largeur du pic PDF. Le GGD est une généralisation de la distribution gaussienne (Pour $\beta = 2$) et de la distribution de Laplace (pour $\beta = 1$). La figure 4.2 présente les PDF en fonction de plusieurs valeurs de α et β .

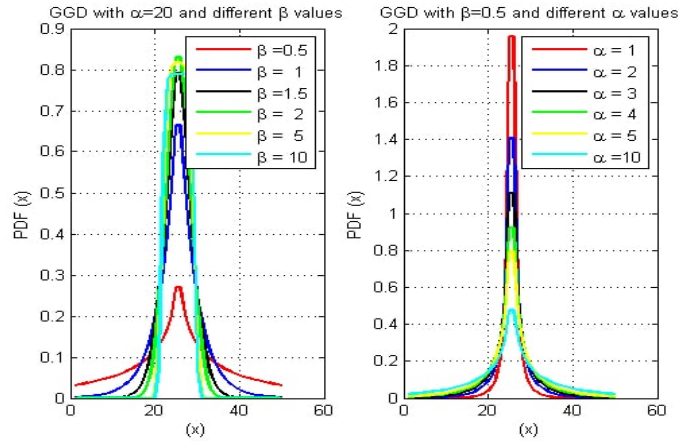


FIGURE 4.2 – La représentation de GGD avec plusieurs valeurs de α et β

Pour réaliser un modèle matching et après la modélisation des blocs des images par la GGD, un calcul de la mesure de la similarité doit être fait. A partir de ce calcul, nous arrivons à faire la correspondance entre les blocs de l'image courante et les blocs candidats dans l'image suivante dans la séquence. La distance de Kullback-leibler (KL) est une mesure fournie par la théorie de l'information. Elle est basée sur l'entropie mutuelle entre deux distributions de probabilités P et Q . Cette mesure présente un outil très solide et largement utilisé en reconnaissance de formes et la mise en correspondance. Sa forme générale est donnée par l'équation 4.3.

$$d_{kl}(P||Q) = \int_{x \in X} P(x) \log \frac{P(x)}{Q(x)} \quad (4.3)$$

Ensuite un principe de vraisemblance présenté dans l'équation 4.4 est utilisé pour maximiser sa fonction afin d'estimer les paramètres α et β .

$$L(X, \alpha, \beta) = \log \prod_{i=1}^N p(x_i, \alpha, \beta) \quad (4.4)$$

Après simplification de la formule 4.4, nous obtenons la formule 4.5.

$$D(P(., \alpha_i, \beta_i) || P(., \alpha_j, \beta_j)) = \log \left(\frac{\beta_i \alpha_j \Gamma(\frac{1}{\beta_j})}{\beta_j \alpha_i \Gamma(\frac{1}{\beta_i})} \right) + \left(\frac{\alpha_i}{\alpha_j} \right)^{\beta_i} \frac{\Gamma(\frac{\beta_j+1}{\beta_i})}{\Gamma(\frac{1}{\beta_i})} - \frac{1}{\beta_i} \quad (4.5)$$

Après le calcul de la distance KLD entre le bloc de l'image contenant notre objet et les blocs de l'image suivante dans le voisinage, nous obtenons après minimisation la nouvelle position de l'objet. Ensuite nous répétons la même procédure de modélisation et correspondance jusqu'à la dernière image de la séquence. Dans l'algorithme 4.1, nous présentons toute la démarche de calcul pour la réalisation d'un suivi dans la séquence d'images.

Dans cet algorithme, nous effectuons une lecture des images. Ensuite, nous réalisons une initialisation du bloc B_G de centre (x_0, y_0) qui contient l'objet via une délimitation par une taille qui correspond à une hauteur H et une longueur W . Dans la troisième étape, nous définissons

Algorithme 4.1 BMM Classique**step1 : Read images.**

- for $i=1 :N$
- Read Image i ,image $i+1$

step2 : Initialization .

- If $i=1$
- Block initialization (B_G, H, W, x_0, y_0)
- Compute the Block Distribution parameters H_G .

step3 : Research zone Defined.

- $HB=3H ;WB=3W$.
- $xb=x_0 - W ;yb=y_0 - H$.
- Region=image($yb : yb + HB, xb : xb + WB$).
- $[nbl, nbc]=size(Region)$.
- $nbp=0$.
- for $j=1 :5 :nbc$
- for $k=1 :5 :nbl$
- $B_D=Region(j :j+H-1, k :k+W-1)$
- Compute block distribution B_D
- Compute KLD.
- Storage KLDs.

step4 : Bloc move.

- Vals :Minimum of KLDs.
- Ind : Minimum KLD Index Block.
- Update : window search size.
- Block move.

une zone de recherche qui correspond à une taille de (HB, WB) . Ensuite, nous avons calculé les paramètres de la distribution relatifs au blocs qui se trouvent dans la zone de recherche. A la fin, nous calculons le KLD et nous minimisons cette mesure pour trouver le nouveau bloc.

modèle matching adapté(AMM)

Le voisinage adapté Nous savons que les fenêtres rectangulaires utilisé dans l'algorithme pour délimiter la zone de recherche dans le cas perspectif ne conviennent pas à une mise en correspondance dans le cas omnidirectionnelle si nous voulons tenir compte de la géométrie de ces images. Dans cette même optique, plusieurs travaux ont été effectués pour trouver une solution. Citons le travail de Radgui et al dans [65] qui ont proposé un voisinage adapté en utilisant la forme du miroir parabolique dans une application de l'estimation du mouvement. Ce voisinage sphérique adopté est basé sur la projection stéréographique inverse définie dans [47] et utilisé aussi dans de nombreuses applications telles que l'estimation du mouvement [76], le traitement classique comme déjà mentionné dans le chapitre 3. Également il est appliqué dans [26] en utilisant un modèle objet composé de lignes et de quatre miroirs pour le suivi des objets.

la formulation mathématique de ce voisinage est présentée dans la formule 4.7.

$$\forall s \in S, \Pi(s) = (R, \theta_s, \varphi_s) \quad (4.6)$$

$$V_s = |\delta\theta| \leq \frac{1}{N}, 2\pi - \frac{1}{M} \leq \delta\varphi \leq \frac{1}{M} \quad (4.7)$$

Avec N et M sont l'ordre du voisinage.

Dans notre algorithme, nous avons défini le voisinage donné par un bloc de taille égal à $\delta\theta$ et $\delta\phi$ donné.

lissage Sphérique Dans la première étape du pré-traitement de la séquence d'images, nous appliquons un filtre gaussien basé sur la rotation des points définie dans [128] utilisée dans [70]. Sur la sphère, nous avons appliqué un produit de convolution (Eq. (4.8)) entre une gaussienne sphérique (Eq. (4.9)) et notre image. la fonction f représente notre image après rotation et g la fonction de la gaussienne sphérique. $n_0 = (0, 0, -1)$ est le pôle sud de la sphère. R est la matrice de rotation des points sphériques.

$$(I * G_s)(P) = \int f(Rn_0) g(R^{-1}P) dR \quad (4.8)$$

$$G_s(\theta, \varphi, t) = e^{-\left(\frac{\cot^2 g^2(\frac{\theta}{2})}{t}\right)} \quad (4.9)$$

L'algorithme adapté Pour la démarche du suivi dans la séquence omnidirectionnelle, nous avons adopté un voisinage sphérique et un lissage pour la phase de pré-traitement. La figure 4.3 représente un schéma Ensuite, nous avons suivi la même procédure de l'algorithme 4.1. Les blocs sont caractérisés par une géométrie sphérique en fonction des coordonnées (θ, ϕ) . Chaque voisinage d'un point sphérique est représenté par les valeurs $(\delta\theta, \delta\phi)$ données. Nous modélisons les blocs par la GGD et ensuite nous calculons les valeurs de KLD. A la fin, nous minimisons cette mesure pour trouver la nouvelle position de l'objet. La localisation des blocs dans le voisinage est toujours en fonction des coordonnées sphériques. L'algorithme 4.2 représente toute la démarche de suivi d'objet.

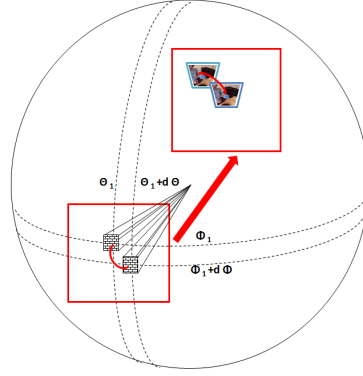


FIGURE 4.3 – Bloc matching en coordonnées sphériques

Algorithme 4.2 Spherical BMM

N : Frame number.

Initialization :

B =full image ;

$l=L-1 ; M_i=[0,0]$;

$\delta \theta = \frac{2\pi}{2B}$, $\delta \phi = \frac{\pi}{2B}$

divide g into uniform block of size $M \delta \theta \times \delta \phi$;

$i=0$;

repet

research zone :

(p_i, q_i) =position of g_i ;

$p \in [p_i + M_i(1) - \frac{W\delta\theta}{2}, p_i + M_i(1) + \frac{W\delta\theta}{2}]$ and

$q \in [q_i + M_i(2) - \frac{W\delta\phi}{2}, p_i + M_i(2) + \frac{W\delta\phi}{2}]$

$f_i = \operatorname{argmin}_{\omega} \operatorname{KLD}(g_i, f_i)$

Block move :

(w_i, t_i) =position of f_i ;

$M_i = [p_i + w_i, q_i + t_i]$;

$i=i+1$;

Dans l'objectif de valider notre modèle et le rendre le maximum possible adapté dans l'espace de sphère nous avons choisis une nouvelle méthode de recherche dans le voisinage qui sera détaillée dans la section suivante.

Algorithme de recherche sphérique Dans le cas des images perspectives, nous trouvons dans la littérature plusieurs applications utilisant des algorithmes de recherche pour la mise en correspondance entre blocs d'images. Premièrement, l'algorithme le plus rapide dans le traitement est ARPS (Adaptive rod pattern search) [92]. Le mouvement général dans une image est généralement cohérent, c'est-à-dire que si les blocs dans le voisinage du bloc courant se sont déplacés dans une direction particulière, il y a une forte probabilité que le bloc courant ait également un vecteur de mouvement similaire. Cet algorithme utilise le vecteur de mouvement du bloc précédent pour définir son propre vecteur de mouvement.

Le point fort de cet algorithme est qu'il n'utilise pas beaucoup de points dans la recherche. L'algorithme le plus populaire est TSS (Three Step Search) [39] qui consiste à générer un voisinage de 8 blocs à une distance égale à un pas (Step) du centre du bloc courant. ensuite on divise le pas

en deux réalisant une autre zone de recherche plus petite. le bloc choisi c'est le bloc qui correspond à la valeur minimale de SAD (Sum of absolute Differences) exprimée dans l'équation en fonction du bloc courant A et le bloc candidat B 4.10 .

$$SAD(A, B) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |A(i, j) - B(i, j)| \quad (4.10)$$

L'algorithme NTSS (New Three Step Search) [88] est une amélioration de TSS qui utilise un système de recherche biaisé et des dispositions pour la moitié des arrêts pour réduire le coûts de calcul. C'est l'un des premiers algorithmes rapides largement utilisés. SES (Simple and Efficient Search)[127] est une autre extension de TSS et exploite l'hypothèse de surface d'erreur uni modale. L'idée principale derrière cet algorithme est que pour une telle surface il ne peut y avoir deux minimums dans des directions opposées et donc les 8 points dans la recherche de modèle fixe de TSS peut être modifiée pour incorporer et économisez sur les calculs. Semblable à NTSS, 4SS (Four Step Search) [86] utilise également un biais central dans recherche et a une disposition d'arrêt à mi-chemin. 4SS définit une taille de motif de $S = 2$ pour la première étape. L'algorithme DS (Diamond Search) [91] est exactement le même que 4SS, mais la recherche du motif de points passe d'un carré à un losange, et là Il n'y a pas de limite au nombre d'étapes que l'algorithme peut prendre lorsque $2 + 7\log_2 w$ est le nombre de points de recherche. Dans le cas général $w = 16$.

Dans le cas des images omnidirectionnelles, et en utilisant un voisinage adapté au déplacement de notre objet. Nous proposons dans ce travail un algorithme de recherche basé sur une sélection des blocs dans un voisinage délimité par des anneaux de taille différentes. Les blocs qui appartiennent au même anneau, ils ont le même nombre de points dans un voisinage sphérique $(d\theta, d\phi)$. Ce qui correspond parfaitement à notre besoin. Cette méthode est nommée Ring-Search (RS) comme illustrée dans la figure 4.4.

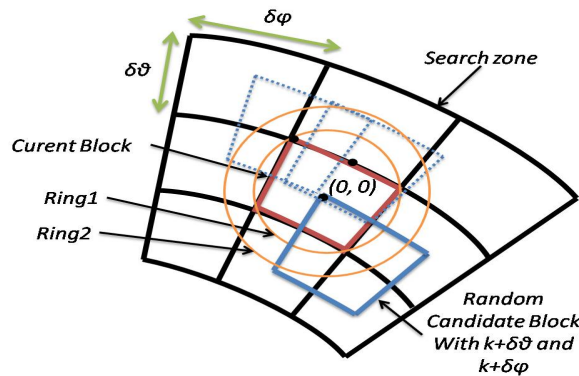


FIGURE 4.4 – Bloc matching avec la méthode Ring Search

Premièrement, RS trace un nombre n d'anneaux de centre $O(0,0)$ et de rayon β_i . Le point O est également le centre du bloc courant. Pour chaque anneau, RS sélectionne au hasard un nombre de points α_i qui appartiennent au périmètre. Ces points seront les centres des blocs candidats. Ensuite, l'algorithme (GGD + KLD) est appliqué pour sélectionner l'anneau approprié pour le bloc candidat. Pour le reste du traitement, cette procédure de recherche sera répétée mais avec les paramètres de l'anneau choisis (α et β). Cette technique nous a permis de déterminer le pas de déplacement de l'objet, et de réduire le nombre de points de recherche et le temps d'exécution.

Le tableau. 4.1 montre que l'algorithme RS est la technique la mieux adaptée à ce problème en terme de PSNR et le temps de d'exécution.

TABLE 4.1 – Bloc matching en utilisant la méthode Ring Search

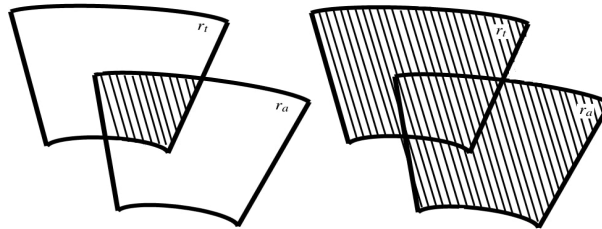
Algorithme	Nombre de points dans la recherche	PSNR	Temps
FS	$(2w + 1)^2$	35.59	213.84
TSS	$1 + 8\log_2 W$	34.82	16.94
ARPS	$3 + 2w$	36.54	12.29
DS	$2 + 7\log_2 w$	35.36	21.54
RS	$10\log_2 W$	35.24	24.08

4.1.3 Méthodes d'évaluation

Dans ce travail, nous utilisons le SOE (Spatial Overlapping Estimation) et la courbe ROC (Receiver Operating Characteristic) comme critères d'évaluation.

Spatial Overlapping Estimation

Étant donné r_t , le bloc trouvé et r_a , le bloc de vérité de terrain ou le résultat souhaité. Le score de "Overlapping" est défini par la relation $S = \frac{r_t \cap r_a}{r_t \cup r_a}$ où \cap et \cup sont respectivement, l'intersection et l'union entre deux régions. Pour mesurer la performance sur une séquence d'images, on compte le nombre de cas réussis dont le "Overlapping" S est plus grand qu'un seuil de 20% (trouvée dans l'expérience et qui dépend de la séquence).

FIGURE 4.5 – Estimation de chevauchement spatial. (gauche.) $r_t \cap r_a$; (droite) $r_t \cup r_a$

Operating Characteristic ROC La courbe ROC illustre le taux des faux positifs en fonction de celui des faux négatifs. Les faux positifs (FP) sont les cas déclarés positifs dans un bloc mais qui ne contient pas l'objet. Les faux négatifs (FN) sont des cas déclarés négatifs pour un bloc qui contient réellement l'objet.

Dans la courbe ROC, nous voulons être au plus près du point (0,1) que possible. Chaque point de la courbe de ROC est évalué en fonction du seuil c . Le suivi garde la détection depuis la dernière détection jusqu'à la perte de trace. Avec une confiance croissante, nous obtenons moins de faux positifs, mais aussi plus de faux négatifs. Pour un c particulier, nous calculons le coordonnées du diagramme comme suit :

$$FP(c) = \frac{\sum_{j=1}^n (FP, \text{wherer}_j > c)}{n} \quad (4.11)$$

$$FN(c) = \frac{\sum_{j=1}^n (FN, \text{wherer}_j > c)}{n} \quad (4.12)$$

où n est un nombre d'images en séquence. Le c est le seuil, et r est la confiance. Pour tracer la courbe ROC, nous calculons les débit des False Positif et le débit des True Positif comme montré

dans les équations :

$$TPR = \frac{TP}{TP + FN} \quad (4.13)$$

$$FPR = \frac{FP}{FP + TN} \quad (4.14)$$

le cas idéal pour le calcul est lorsque $TPR = 1$ et $FPR = 0$, et ceci est impossible dans la réalité. Notre objectif est obtenir une large marge entre les deux valeurs. Dans notre algorithme, pour chaque cas de suivi p on calcule le doublet (FPR, TPR) . Nous regroupons tous les cas de suivi pour toute la séquence, et nous traçons tous les point p correspondants. dans notre cas, nous avons obtenu une surface de 0,8614, c'est à dire un taux de suivi correct de 86,14%.

4.1.4 Résultats et synthèse

Une comparaison entre l'algorithme classique et notre approche basée sur une modélisation adaptée est faite dans des images omnidirectionnelles en terme de la courbe ROC présenté dans la figure 4.6.

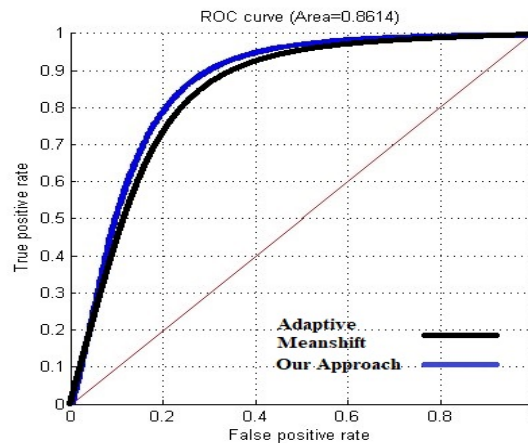


FIGURE 4.6 – La courbe ROC

Dans le tableau 4.2, nous avons fait une comparaison entre la méthode Mean-Shift adapté dans le cas sphérique présentée dans le travail de Caron et al.[94]. Cette méthode se base sur une adaptation sphérique du voisinage des blocs. Notre méthode a donné de bons résultats en terme de SOE avec une différence de 5% dans la séquence Indoor et Outdoor. Ce qui montre que notre méthode est performante en terme de overlapping et le bloc dans le voisinage sphérique est bien suivi en utilisant les caractéristiques statistiques. Pour une comparaison en terme de SOE, il faut la réaliser avec une méthode de même critère d'évaluation.

TABLE 4.2 – Mesure de Overlapping

Séquence	Moyenne	SMM	Mean-Shift adapté [94]
Séquence 1	Chevauchement	70,4%	65%
Séquence 2	Spatial	65,3%	60%

Dans la figure 5.5, nous présentons les résultats de suivi dans deux séquences omnidirectionnelles. La première séquence représente une scène Indoor avec une carte qui bouge autour de la

caméra. La deuxième séquence Outdoor illustre un déplacement des personnes autour de la caméra. Dans ces cas de figure, La région suivi est caractérisée par trois couleurs. La couleur bleu qui représente l'application de l'algorithme de bloc matching classique. La couleur verte représente la vérité de terrain, et la couleur rouge représente notre méthode.

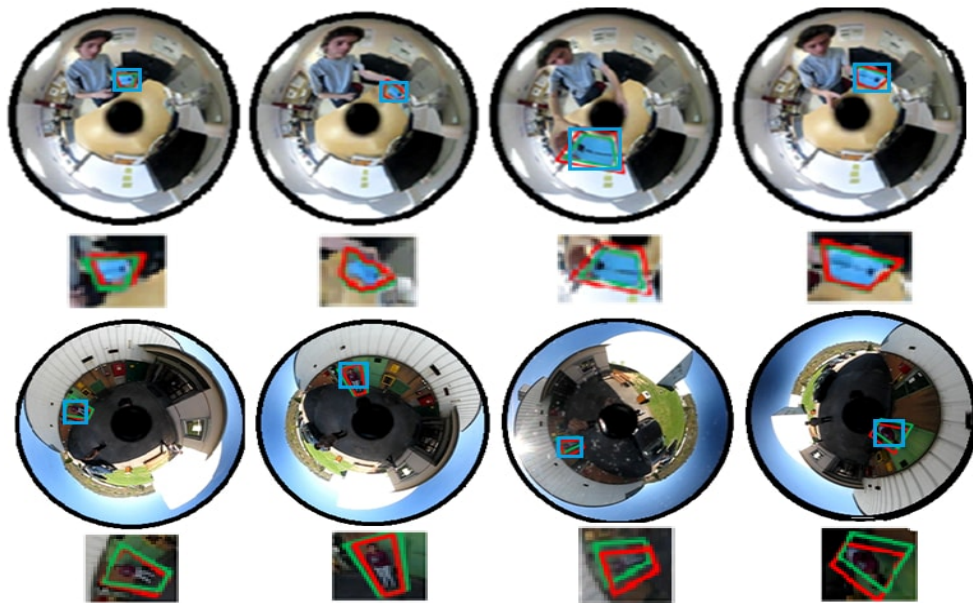


FIGURE 4.7 – Les résultats du suivi avec la correspondance de modèle classique sont représentés en bleu. Les résultats avec la méthode proposée sont affichés en rouge et la vérité terrain est en vert.

Dans les résultats, nous avons testé le suivi dans la séquence d'images omnidirectionnelles en utilisant le SOE. Ces résultats sont justifiés car ils ont été appliqué sur des images avec une géométrie spéciale déformée et en comparant avec une méthode de même envergure tel que le meanshift adapté. Par contre, l'application des méthodes classiques adaptées au changement de taille comme le CAM-shift donnent de bon résultats mais théoriquement il ne respectent pas la particularité du système.

4.1.5 Conclusion

Dans ce chapitre, une méthode de suivi dans une séquence d'images omnidirectionnelles est présentée. La méthode est basée sur l'adaptation de la formulation mathématique de la modélisation utilisée. le Modèle est basé sur l'estimation des paramètres de la distribution gaussienne. Ce paramétrage est appliqué aux blocs des images de la séquence pour estimer la position de l'objet. Cet Algorithme se base sur une nouvelle méthode de recherche nommée Ring Search et un voisinage sphérique adapté à la géométrie des images omnidirectionnelle ainsi que la manière avec laquelle l'objet se déplace. Les résultats expérimentales avec différentes séquences dans différentes conditions montrent une bonne performance.

4.2 Le suivi d'objet en utilisant la distribution de Von Mises Fisher

4.2.1 Introduction

Dans ce chapitre, notre objectif est de réaliser un suivi d'objet dans une séquence d'images omnidirectionnelles en se basant sur une approche statistique. Le choix de la méthode a pour raison de résoudre le problème de la géométrie des images utilisées. Dans un système catadioptrique, les objets ou les personnes qui changent d'emplacement dans la scène environnante subissent de manière objective un changement de taille et de forme. Ce qui nécessite une adaptation pour en tenir compte durant le traitement.

Notre méthode résout parfaitement les contraintes de changement de taille et de forme en considérant que l'approche statistique est liée seulement à l'information de la région interne de notre objet. Ce travail se concentre sur la réalisation d'un traitement adapté sur les images sphériques qui proviennent d'une projection stéréographique inverse de l'espace d'une sphère vers le plan catadioptrique plane.

Dans un contexte de suivi, les caméras omnidirectionnelles offrent un avantage pour réaliser un suivi d'objet non rigide et le trouver dans la scène environnante. Dans la plupart des travaux, nous trouvons des contraintes liées aux traitements classiques appliqués. Ces traitements marchent très bien dans le cas pratique pour un algorithme robuste. Mais il ne tiens pas compte de la géométrie déformée des images omnidirectionnelles. C'est pourquoi, nous devons adapter l'algorithme en fonction de la forme du miroir utilisé et les distorsions existantes dans les zones les plus proches et loin de la caméra pour ne pas perdre l'information.

Dans notre travail, nous effectuons le suivi comme un algorithme de correspondance des modèles. Ces derniers sont présentés par une distribution projetée dans l'espace de la sphère. La distribution utilisée est la distribution de von Mises-Fisher [27]. Elle a comme entrées des angles présentés par la phase de Gabor calculée pour chaque bloc de l'image. L'objectif dans ce travail est de minimiser la distance entre les modèles du bloc courant et les autres blocs candidats dans l'image suivante dans la séquence. Cette mesure de similarité est calculée par la distance de KLD (Kulback Liebler Distance).

les distributions de Von Mises-Fisher sphériques ont été utilisés pour le suivi de plusieurs intervenants dans des réunions en visio-conférence dans [108] et leur classification dans [44]. D'autres applications incluent l'estimation de l'orientation basée sur les quaternions [48], [63], la modélisation de la structure des protéines en biologie moléculaire [3], l'apprentissage automatique ([2], [67]), et la neuroscience [?]. Les premières utilisations des distributions sphériques ont été réalisées dans le domaine de la géologie [21]. Ces dernières années, ces approches ont trouvé leur place dans de nombreux domaines et ne se limitent pas aux statistiques descriptives, mais sont également utilisées pour l'estimation et le filtrage. Par exemple, la tâche de suivi d'objets à l'aide de caméras omnidirectionnelles [62] peut naturellement être mappée à un problème d'estimation sphérique. Ainsi, il devient même possible d'envisager un suivi simultané de plusieurs cibles sur la surface de la sphère unitaire [77]. Dans le même contexte, nous considérons les informations de phase de Gabor comme une entrée d'estimateur. Ceci est dû au fait qu'il peut prendre des valeurs très différentes même s'il est échantillonné à des emplacements d'image distants de seulement quelques pixels [103].

4.2.2 La phase de Gabor

Le filtre de gabor (nommé aussi les ondelettes de Gabor) est connu pour être un outil robuste pour l'extraction des caractéristiques. Il représente une information locale optimale dans les domaines spatial et fréquentiel. Dans le domaine spatial [103], la famille des filtres de Gabor 2D est définie comme suit :

$$\Psi_{(u,v)}(x, y) = \frac{f_u^2}{\pi\alpha\beta} e^{(-f_u^2[\alpha^{-2}x'^2 + \beta^{-2}y'^2])} e^{(j2\pi f_u x')} \quad (4.15)$$

Avec $x' = x\cos(\theta_v) + y\sin(\theta_v)$ et $y' = -\sin(\theta_v)x + \cos(\theta_v)y$ et $\theta_v = v * \frac{\pi}{8}$ est l'orientation de la sinusoïde. f_u est la fréquence défini par : $f_u = \frac{f_{max}}{2^{\frac{v}{2}}}$. Chaque filtre est représenté par un noyau gaussien défini par une fréquence centrale et une orientation respectivement définis par f_u et θ_v . α et β déterminent le rapport entre la fréquence centrale et la taille de l'enveloppe de la gaussienne.

Les paramètres θ_v et f_u définissent la bande passante du canal de Gabor et les caractéristiques spectrales du noyau de Gabor. Les noyaux de filtre de Gabor en quadrature dans une représentation en ondelettes séparables sont donnés par l'équation 5.2.

$$\begin{aligned} \Psi_+(x, y, f_u, \theta_v) &= g(x, y) \cos(2\pi f_u x'), \\ \Psi_-(x, y, f_u, \theta_v) &= g(x, y) \sin(2\pi f_u x') \\ g(x, y) &= e^{\left(\frac{-\lambda^2 f_u (x^2 + y^2)}{4\pi}\right)}, \end{aligned} \quad (4.16)$$

où λ est une constante. Dans l'équation (5.3), $G_{(u,v)}(x, y)$ représente la sortie de convolution complexe qui peut être décomposée en parties réelles $R_{(u,v)}(x, y)$ et imaginaires $Im_{(u,v)}(x, y)$ comme suit :

$$G_{(u,v)}(x, y) = I(x, y) * \Psi_{(u,v)}(x, y) \quad (4.17)$$

où $I(x, y)$ est l'image d'entrée. Le module M et la phase Φ , sont données par l'équation 5.4.

$$\begin{aligned} M(x, y, f_u, \theta_v) &= \sqrt{R_{(u,v)}(x, y)^2 + Im_{(u,v)}(x, y)^2}, \\ \Phi(x, y, f_u, \theta_v) &= \arctan\left(\frac{Im_{(u,v)}(x, y)}{R_{(u,v)}(x, y)}\right). \end{aligned} \quad (4.18)$$

La phase de gabor est représentée dans la figure 5.2.

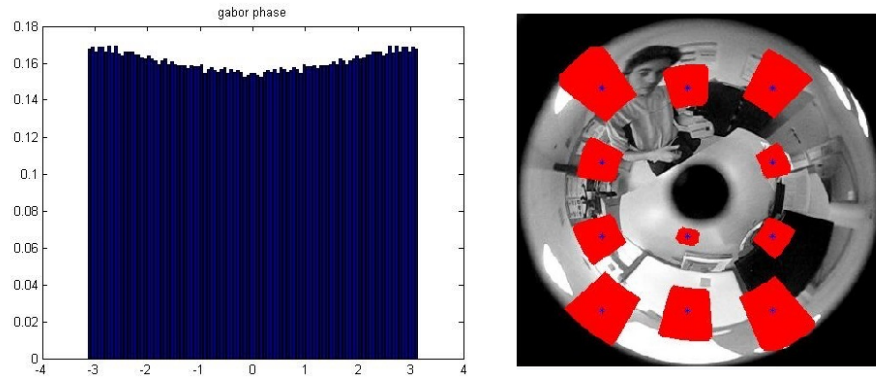


FIGURE 4.8 – Histogramme de phase de gabor pour un bloc d'une image omnidirectionnelle.

Nous appliquons le filtre de Gabor sur chaque bloc de l'image. La forme du bloc dans une image perspective est différente de celle dans une image omnidirectionnelle. Dans notre algorithme, nous avons défini le voisinage par bloc de taille défini par $\delta\theta$ et $\delta\phi$ comme montré dans la formule (5.5).

$$N_s = \left(|\delta\theta| \leq \frac{1}{N}, 2\pi - \frac{1}{M} \leq \delta\phi \leq \frac{1}{M} \right) \quad (4.19)$$

N et M sont les ordres de voisinage. La forme des blocs sphériques est représentée sur la figure 5.2. Le résultat de l'application de la phase de Gabor dans une image sphérique avant qu'elle soit projetée sur le plan 2D est présentée par la figure

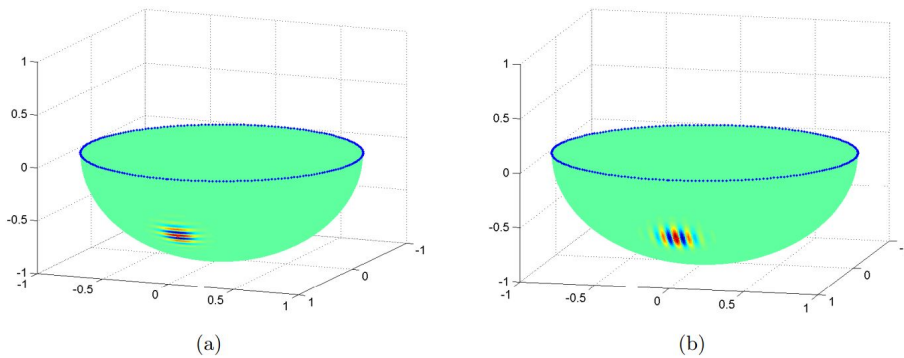


FIGURE 4.9 – La phase de Gabor sphérique pour un bloc d'une image omnidirectionnelle.

Nous extrayons la phase de l'historgramme de chaque bloc qui sera une entrée de l'estimateur de Von Mises-Fisher afin de créer notre modèle.

4.2.3 La distribution de von Mises-Fisher

La distribution de probabilité $f(\mu, \kappa)$ définit sur la sphère unitaire S^{d-1} , dont la fonction de densité de probabilité (pdf) est donnée par l'équation 5.6.

$$f(x; \mu; \kappa) = C_d(\kappa) e^{\kappa \mu x}; x \in S^{d-1} \quad (4.20)$$

Cette dernière est appelée la distribution de von Mises-Fisher (vMF) avec les paramètres $\kappa \geq 0$ et $\mu \in S^{d-1}$ désignent respectivement la concentration et la direction moyenne.

$$C_d(\kappa) = \frac{\kappa^{\frac{d}{2}-1}}{2\pi^{\frac{d}{2}} I_{\frac{d}{2}-1}(\kappa)} \quad (4.21)$$

$C_d(\kappa)$ est la constante de normalisation, où I_p désigne la fonction de Bessel modifiée [?] à l'ordre $p = \frac{d}{2} - 1$. La densité présentée dans l'équation (5.6) est invariante en rotation autour de la direction moyenne. C'est aussi l'équivalent de la distribution gaussienne multivariée. Cette distribution est caractérisée par le principe d'entropie maximale afin de maximiser l'entropie de Boltzmann-Shannon sous la moyenne directionnelle donnée. Les distributions de Von Mises-Fisher appartiennent à la famille exponentielle [73] défini par les paramètres naturels $\theta = \kappa\mu \in R^d$ et la fonction de normalisation logarithmique donnée par l'équation 5.8.

$$F_d(\theta) = -\log C_d(\|\theta\|) \quad (4.22)$$

La distribution vMF est entièrement déterminée par la moyenne directionnelle (angulaire) présentée dans l'équation 5.9.

$$E[x] = \int_{S^{d-1}} x f(x; \mu; \kappa) dx = \Delta F_d(\theta) =: A_d(\kappa)\mu, \quad (4.23)$$

Avec A_d est le gain de la fonction de Bessel présentée dans l'équation 5.10.

$$A_d = \frac{I_{d/2}(\kappa)}{I_{d/2-1}(\kappa)} \quad (4.24)$$

Après l'estimation des paramètres de la distribution de Von Mises Fisher, On obtient un vecteur aléatoire avec κ et μ . Et pour obtenir la direction, on a besoin seulement d'appliquer une rotation adéquate.

Dans un contexte de traitement des images omnidirectionnelle présentées dans un espace sphérique et projetés dans un plan catadioptrique, le suivi réalisé en utilisant la distribution de VMF est particulièrement appliqué dans le cas $d = 3$ dans un espace S^2 . Les expressions ci-dessus se simplifient en $C_3(\kappa) = \frac{\kappa}{4\pi \sinh(\kappa)}$ et $A_3(\kappa) = \frac{1}{\tanh(\kappa)} - \frac{1}{\kappa}$.

Un exemple de distribution de von Mises-Fisher sur la sphère unitaire S^2 avec différentes directions moyennes et paramètres de concentration est représenté sur la figure. 5.3.

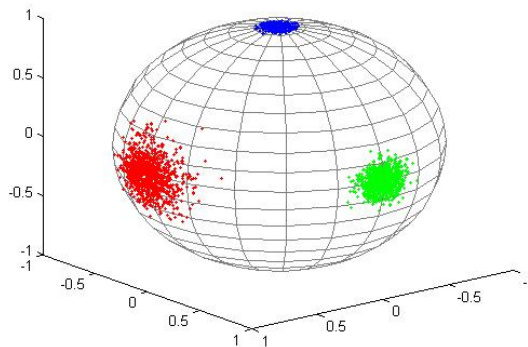


FIGURE 4.10 – La distribution de VMF sur la sphère unitaire avec différentes directions et concentrations 50 (rouge), 150 (vert), 500 (bleu).

Dans notre modèle, nous supposons que les objets en mouvement sont relativement lents. C'est-à-dire que les changements de position de l'objet entre deux images successives sont relativement petits. Mathématiquement, le mouvement est modélisé par un angle représenté par la phase de Gabor pour tous les blocs de l'image sous forme d'une séquence d'angles.

Soit une suite $T = \theta_1, \dots, \theta_n$ de valeurs angulaires.

- Chaque séquence d'angles est une entrée pour l'estimateur de VmF.
- Après estimation des paramètres (κ, μ) . Ces variables sont calculées pour chaque bloc d'image au voisinage du bloc courant dans l'image suivante.
- La sélection du bloc candidat effectuée en choisissant le KLD minimum calculé entre les modèles dans un voisinage adapté.

4.2.4 La mesure de similarité

Dans ce travail, nous utilisons une méthode qui calcule la distance entre le modèle courant et les modèles dans le voisinage. Nous proposons une mesure basée sur des distances statistiques [35]. La démonstration pour arriver la formule finale du KLD entre deux distribution de VonMises Fisher est présentée dans l'annexe 1. La formule du KLD est donnée par la relation (5.11).

$$D_{KL}(p, q) = \log\left(\frac{\kappa_p \sinh(\kappa_q)}{\kappa_q \sinh(\kappa_p)}\right) - \left(\frac{1}{\tanh(\kappa_p)} - \frac{1}{\kappa_p}\right) \mu_p^T (\kappa_q \mu_q - \kappa_p \mu_p) \quad (4.25)$$

Dans l'algorithme 5.1, nous avons présenté la démarche pour la réalisation de suivi d'objet dans une séquence d'images omnidirectionnelles en se basant sur le calcul de la similarité. Cette mesure présentée par le KLD. Pour chaque bloc d'image contenant notre objet dans la première image de la séquence, nous calculons la phase de Gabor relative. Ensuite, en utilisant l'estimation de Von Mises Fisher, nous avons déterminé les paramètres κ et μ . Nous calculons le KLD entre le bloc de l'image courante et tous les blocs dans le voisinage délimité par le centre du bloc et l'intervalle $\delta\theta$ et $\delta\phi$. A la fin, nous minimisons la distance pour trouver la nouvelle position du bloc dans l'image suivante de la séquence. Nous répétons cette procédure pour chaque image.

4.2.5 La Méthode d'évaluation

Étant donné r_t , le bloc trouvé et r_a , le bloc de vérité de terrain ou le résultat souhaité. Le score de "Ovelapping" est défini par la relation $S = \frac{r_t \cap r_a}{r_t \cup r_a}$ où \cap et \cup sont respectivement, l'intersection et l'union entre deux régions. Pour mesurer la performance sur une séquence d'images, on compte le nombre de cas réussis dont le "Ovelapping" S est plus grand qu'un seuil de 20% (trouvée dans l'expérience et qui dépend de la séquence).

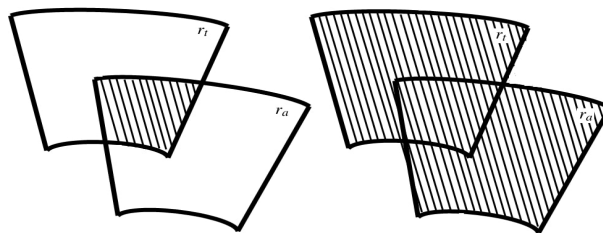


FIGURE 4.11 – Estimation de Overlapping spatial. (gauche.) $r_t \cap r_a$; (droite) $r_t \cup r_a$

Algorithme 4.3 Modèle matching sphérique

Entrées: N : Frame number ; I_1, I_2 : current and next images ; $\delta\theta = \frac{\pi}{40}$; $\delta\phi = \frac{\pi}{15}$

pour $n = 1 : N$ **faire**

Sorties: Initialization : first image block

si $i = 1$ **alors**

Select the object block $g_i(\delta\theta, \delta\phi)$;

Compute Gabor phase P_{g_i}

Estimate VMF parameters $[\kappa_{g_i}, \mu_{g_i}] = VMF(P_{g_i})$

sinon

Entrées: g_i, f_i : image blocks in I_1 and I_2

Sorties: research zone :

The Search zone SZ size is equal to $(M\delta\theta, W\delta\phi)$

$C_i = (p_i, q_i)$ = Center position of g_i

$C_s = (p, q)$ = Center position of Search zone

$p \in [p_i - \frac{M\delta\theta}{2}, p_i + \frac{M\delta\theta}{2}]$

$q \in [q_i - \frac{W\delta\phi}{2}, p_i + \frac{W\delta\phi}{2}]$

pour $i = 1 : size(SZ, 1)$ **faire**

pour $j = 1 : size(SZ, 2)$ **faire**

Compute Gabor phase P_{f_i}

$[\kappa_{f_i}, \mu_{f_i}] = VMF(P_{f_i})$

Compute $kld_{fg}(\kappa_{g_i}, \mu_{g_i}, \kappa_{f_i}, \mu_{f_i})$

Where $f_i(p_i, q_i)$ blocks in the vicinity of g_i .

Storage all kld_{fg}

fin pour

fin pour

$[i, j, kld] = argmin_{\omega} KLD$

Sorties: Block move :

$w_i = i; t_i = j$

(w_i, t_i) = position of f_i

$M_i = [p_i + w_i, q_i + t_i]$

fin si

fin pour

4.2.6 Résultats et Synthèse

Dans le tableau 5.1, nous avons comparé l'algorithme de MeanShift adapté [26] avec notre approche basée sur la correspondance de modèles adaptés dans des images omnidirectionnelles en terme de Overlapping spatial.

TABLE 4.3 – Adapted Model Block Matching

Seq.	Moyenne	SMM	MeanShift Adapté [94]
Seq.1	Overlapping	70,4%	65%
Seq.2	Spatiale	65,3%	60%

Dans la figure 5.5, nous présentons les résultat de suivi dans deux séquences omnidirectionnelles. La première séquence représente une scène Indoor avec une carte qui bouge autour de la caméra. La deuxième séquence Outdoor illustre un déplacement des personnes autour de la caméra. Dans ces cas de figure, La région suivi est caractérisée par trois couleurs. La couleur bleu représente l'application de l'algorithme de bloc matching classique. La couleur verte représente la vérité de terrain, et la couleur rouge représente notre méthode.

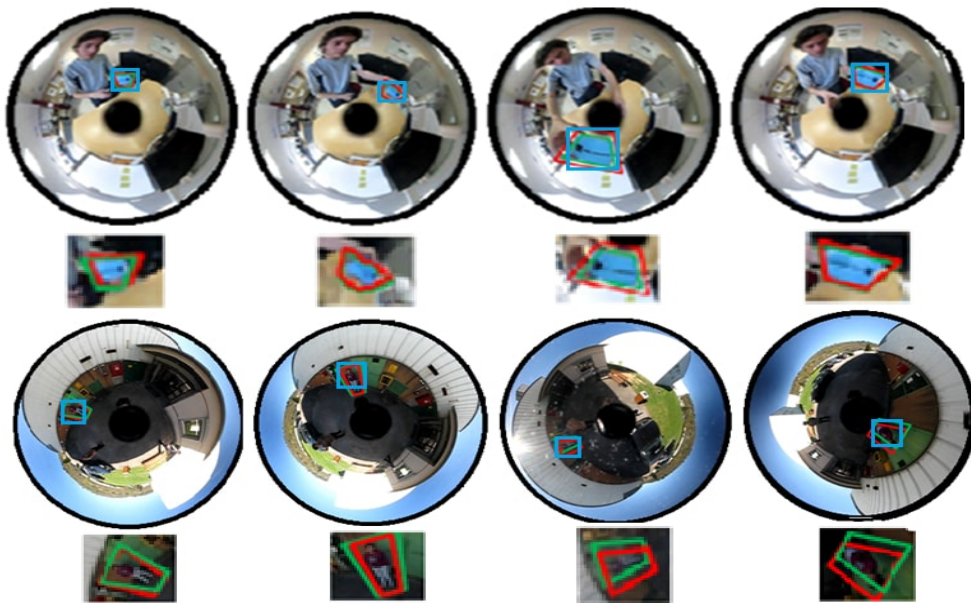


FIGURE 4.12 – Résultats du suivi. Les résultats du suivi avec la correspondance de modèle classique sont représentés en bleu. Les résultats avec la méthode proposée sont affichés en rouge et la vérité terrain en vert.

L'application de la méthode MeanShift pour le suivi est efficace surtout que son algorithme se base sur une décomposition de bloc selon les coordonnées sphériques. C'est une décomposition des blocs qui ne varient pas dans la taille si l'objet a subit des grandes déplacements. En revanche, notre algorithme est performant car il s'appuie sur la sélection des blocs qui changent tout le temps leur taille ainsi que le nombre de points dans le voisinage. Ceci entraîne une délimitation précise de bloc lorsque notre objet bouge soit autour de la caméra à la même distance, ou bien s'il se rapproche ou s'éloigne de la caméra.

Les résultats obtenues en utilisant les distributions VonMises-fisher sont plus performants que ceux basés sur la gaussienne. Ceci réside dans le fait que les tests effectués en utilisant la phase

de Gabor donnent plus d'invariance que l'intensité de pixel dans le cas gaussien. Ceci est dû à l'information de pixel qui peut subir des changements dans la séquence d'images dans le cas de suivi à savoir la luminosité et le bruit.

4.2.7 Conclusion

Dans ce travail, nous présentons une méthode de suivi dans un système catadioptrique basée sur une modélisation avec une approche statistique. Cette approche présentée par la distribution de VonMises Fisher, et adaptée à la géométrie des images omnidirectionnelles. L'adaptation est présentée par la projection d'image sur la sphère. Dans le contexte de suivi, nous avons minimisé la distance entre les modèles en utilisant la mesure KLD pour estimer la position de l'objet en mouvement. Les résultats expérimentaux prennent en compte les changements de taille et de forme du bloc et montrent une bonne performance. Dans les travaux futurs, nous allons opter pour une extension pour le suivi d'objets multiples avec un traitement en temps réel.

*Anisse, K., Amina, R. Mohamed, R. (2020). Spherical Object Tracking using Von Mises-Fisher Distribution in Catadioptric System. Journal of Computer Science, 16(9), 1229-1236.
<https://doi.org/10.3844/jcssp.2020.1229.1236>*

5.1 Introduction

Dans ce chapitre, notre objectif est de réaliser un suivi d'objet dans une séquence d'images omnidirectionnelles en se basant sur une approche statistique. Le choix de la méthode a pour raison de résoudre le problème de la géométrie des images utilisées. Dans un système catadioptrique, les objets ou les personnes qui changent d'emplacement dans la scène environnante subissent de manière objective un changement de taille et de forme. Ce qui nécessite une adaptation pour en tenir compte durant le traitement.

Notre méthode résout parfaitement les contraintes de changement de taille et de forme en considérant que l'approche statistique est liée seulement à l'information de la région interne de notre objet. Ce travail se concentre sur la réalisation d'un traitement adapté sur les images sphériques qui proviennent d'une projection stéréographique inverse de l'espace d'une sphère vers le plan catadioptrique plane.

Dans un contexte de suivi, les caméras omnidirectionnelles offrent un avantage pour réaliser un suivi d'objet non rigide et le trouver dans la scène environnante. Dans la plupart des travaux, nous trouvons des contraintes liées aux traitements classiques appliqués. Ces traitements marchent très bien dans le cas pratique pour un algorithme robuste. Mais il ne tiens pas compte de la géométrie déformée des images omnidirectionnelles. C'est pourquoi, nous devons adapter l'algorithme en fonction de la forme du miroir utilisé et les distorsions existantes dans les zones les plus proches et loin de la caméra pour ne pas perdre l'information.

Dans notre travail, nous effectuons le suivi comme un algorithme de correspondance des modèles. Ces derniers sont présentés par une distribution projetée dans l'espace de la sphère. La distribution utilisée est la distribution de von Mises-Fisher [27]. Elle a comme entrées des angles présentés par la phase de Gabor calculée pour chaque bloc de l'image. L'objectif dans ce travail est de minimiser la distance entre les modèles du bloc courant et les autres blocs candidats dans l'image suivante dans la séquence. Cette mesure de similarité est calculée par la distance de KLD (Kulback Liebler Distance).

les distributions de Von Mises-Fisher sphériques ont été utilisés pour le suivi de plusieurs intervenants dans des réunions en visio-conférence dans [108] et leur classification dans [44]. D'autres applications incluent l'estimation de l'orientation basée sur les quaternions [48], [63], la modélisation de la structure des protéines en biologie moléculaire [3], l'apprentissage automatique ([2],

[67]). Les premières utilisations des distributions sphériques ont été réalisées dans le domaine de la géologie [21]. Ces dernières années, ces approches ont trouvé leur place dans de nombreux domaines et ne se limitent pas aux statistiques descriptives, mais sont également utilisées pour l'estimation et le filtrage. Par exemple, la tâche de suivi d'objets à l'aide de caméras omnidirectionnelles [62] peut naturellement être mappée à un problème d'estimation sphérique. Ainsi, il devient même possible d'envisager un suivi simultané de plusieurs cibles sur la surface de la sphère unitaire [77]. Dans le même contexte, nous considérons les informations de phase de Gabor comme une entrée d'estimateur. Ceci est dû au fait qu'il peut prendre des valeurs très différentes même s'il est échantillonné à des emplacements d'image distants de seulement quelques pixels [103].

5.2 La phase de Gabor

Le filtre de gabor (nommé aussi les ondelettes de Gabor) est connu pour être un outil robuste pour l'extraction des caractéristiques. Il représente une information locale optimale dans les domaines spatial et fréquentiel. Dans le domaine spatial [103], la famille des filtres de Gabor 2D est définie comme suit :

$$\Psi_{(u,v)}(x, y) = \frac{f_u^2}{\pi\alpha\beta} e^{(-f_u^2[\alpha^{-2}x'^2 + \beta^{-2}y'^2])} e^{(j2\pi f_u x')} \quad (5.1)$$

Avec $x' = x\cos(\theta_v) + y\sin(\theta_v)$ et $y' = -\sin(\theta_v) + y\cos(\theta_v)$ et $\theta_v = v * \frac{\pi}{8}$ est l'orientation de la sinusoïde. f_u est la fréquence définie par : $f_u = \frac{f_{max}}{2^2}$. Chaque filtre est représenté par un noyau gaussien défini par une fréquence centrale et une orientation respectivement définies par f_u et θ_v . α et β déterminent le rapport entre la fréquence centrale et la taille de l'enveloppe de la gaussienne.

Les paramètres θ_v et f_u définissent la bande passante du canal de Gabor et les caractéristiques spectrales du noyau de Gabor. Les noyaux de filtre de Gabor en quadrature dans une représentation en ondelettes séparables sont donnés par l'équation 5.2.

$$\begin{aligned} \Psi_+(x, y, f_u, \theta_v) &= g(x, y) \cos(2\pi f_u x'), \\ \Psi_-(x, y, f_u, \theta_v) &= g(x, y) \sin(2\pi f_u x') \\ g(x, y) &= e^{\left(\frac{-\lambda^2 f_u (x^2 + y^2)}{4\pi}\right)}, \end{aligned} \quad (5.2)$$

où λ est une constante. Dans l'équation (5.3), $G_{(u,v)}(x, y)$ représente la sortie de convolution complexe qui peut être décomposée en parties réelles $R_{(u,v)}(x, y)$ et imaginaires $Im_{(u,v)}(x, y)$ comme suit :

$$G_{(u,v)}(x, y) = I(x, y) * \Psi_{(u,v)}(x, y) \quad (5.3)$$

où $I(x, y)$ est l'image d'entrée. Le module M et la phase Φ , sont données par l'équation 5.4.

$$\begin{aligned} M(x, y, f_u, \theta_v) &= \sqrt{R_{(u,v)}(x, y)^2 + Im_{(u,v)}(x, y)^2}, \\ \Phi(x, y, f_u, \theta_v) &= \arctan\left(\frac{Im_{(u,v)}(x, y)}{R_{(u,v)}(x, y)}\right). \end{aligned} \quad (5.4)$$

La phase de gabor est représentée dans la figure 5.2.

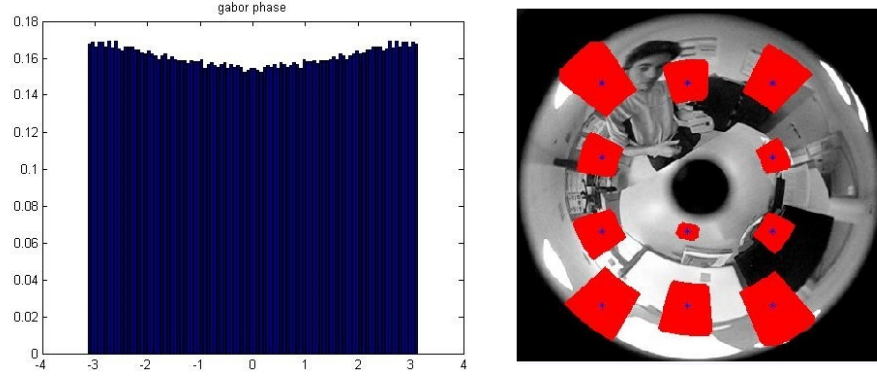


FIGURE 5.1 – Histogramme de phase de gabor pour un bloc d'une image omnidirectionnelle.

Nous appliquons le filtre de Gabor sur chaque bloc de l'image. La forme du bloc dans une image perspective est différente de celle dans une image omnidirectionnelle. Dans notre algorithme, nous avons défini le voisinage par bloc de taille définit par $\delta\theta$ et $\delta\phi$ comme montré dans la formule (5.5).

$$N_s = \left(|\delta\theta| \leq \frac{1}{N}, 2\pi - \frac{1}{M} \leq \delta\phi \leq \frac{1}{M} \right) \quad (5.5)$$

N et M sont les ordres de voisinage. La forme des blocs sphériques est représentée sur la figure 5.2. Le résultat de l'application de la phase de Gabor dans une image sphérique avant qu'elle soit projetée sur le plan 2D est présentée par la figure

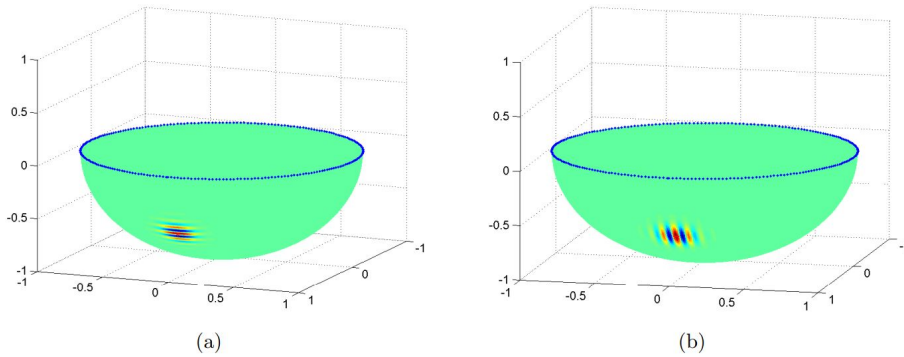


FIGURE 5.2 – La phase de Gabor sphérique pour un bloc d'une image omnidirectionnelle.

Nous extrayons la phase de l'histogramme de chaque bloc qui sera une entrée de l'estimateur de Von Mises-Fisher afin de créer notre modèle.

5.3 La distribution de von Mises-Fisher

La distribution de probabilité $f(\mu, \kappa)$ définit sur la sphère unitaire S^{d-1} , dont la fonction de densité de probabilité (pdf) est donnée par l'équation 5.6.

$$f(x; \mu; \kappa) = C_d(\kappa) e^{\kappa \mu x}; x \in S^{d-1} \quad (5.6)$$

Cette dernière est appelée la distribution de von Mises-Fisher (vMF) avec les paramètres $\kappa \geq 0$ et $\mu \in S^{d-1}$ désignent respectivement la concentration et la direction moyenne.

$$C_d(\kappa) = \frac{\kappa^{\frac{d}{2}-1}}{2\pi^{\frac{d}{2}} I_{\frac{d}{2}-1}(\kappa)} \quad (5.7)$$

$C_d(\kappa)$ est la constante de normalisation, où I_p désigne la fonction de Bessel modifiée à l'ordre $p = \frac{d}{2} - 1$. La densité présentée dans l'équation (5.6) est invariante en rotation autour de la direction moyenne. C'est aussi l'équivalent de la distribution gaussienne multivariée. Cette distribution est caractérisée par le principe d'entropie maximale afin de maximiser l'entropie de Boltzmann-Shannon sous la moyenne directionnelle donnée. Les distributions de Von Mises-Fisher appartiennent à la famille exponentielle [73] définie par les paramètres naturels $\theta = \kappa\mu \in R^d$ et la fonction de normalisation logarithmique donnée par l'équation.

$$F_d(\theta) = -\log C_d(\|\theta\|) \quad (5.8)$$

La distribution vMF est entièrement déterminée par la moyenne directionnelle (angulaire) présentée dans l'équation 5.9.

$$E[x] = \int_{S^{d-1}} xf(x; \mu; \kappa) dx = \Delta F_d(\theta) =: A_d(\kappa)\mu, \quad (5.9)$$

Avec A_d est le gain de la fonction de Bessel présentée dans l'équation 5.10.

$$A_d = \frac{I_{d/2}(\kappa)}{I_{d/2-1}(\kappa)} \quad (5.10)$$

Après l'estimation des paramètres de la distribution de Von Mises Fisher, On obtient un vecteur aléatoire avec κ et μ . Et pour obtenir la direction, on a besoin seulement d'appliquer une rotation adéquate.

Dans un contexte de traitement des images omnidirectionnelle présentées dans un espace sphérique et projetés dans un plan catadioptrique, le suivi réalisé en utilisant la distribution de VMF est particulièrement appliqué dans le cas $d = 3$ dans un espace S^2 . Les expressions ci-dessus se simplifient en $C_3(\kappa) = \frac{\kappa}{4\pi \sinh(\kappa)}$ et $A_3(\kappa) = \frac{1}{\tanh(\kappa)} - \frac{1}{\kappa}$.

Un exemple de distribution de von Mises-Fisher sur la sphère unitaire S^2 avec différentes directions moyennes et paramètres de concentration est représenté sur la figure. 5.3.

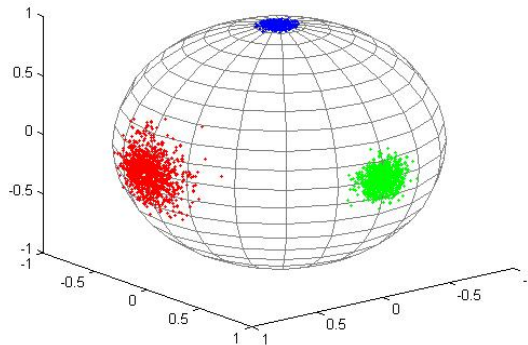


FIGURE 5.3 – La distribution de VMF sur la sphère unitaire avec différentes directions et concentrations 50 (rouge), 150 (vert), 500 (bleu).

Dans notre modèle, nous supposons que les objets en mouvement sont relativement lents. C'est-à-dire que les changements de position de l'objet entre deux images successives sont relativement petits. Mathématiquement, le mouvement est modélisé par un angle représenté par la phase de Gabor pour tous les blocs de l'image sous forme d'une séquence d'angles.

Soit une suite $T = \theta_1, \dots, \theta_n$ de valeurs angulaires.

- Chaque séquence d'angles est une entrée pour l'estimateur de VmF.
- Après estimation des paramètres (κ, μ) . Ces variables sont calculées pour chaque bloc d'image au voisinage du bloc courant dans l'image suivante.
- La sélection du bloc candidat effectuée en choisissant le KLD minimum calculé entre les modèles dans un voisinage adapté.

5.4 La mesure de similarité

Dans ce travail, nous utilisons une méthode qui calcule la distance entre le modèle courant et les modèles dans le voisinage. Nous proposons une mesure basée sur des distances statistiques [35]. La démonstration pour arriver la formule finale du KLD entre deux distribution de VonMises Fisher est présentée dans l'annexe 1. La formule du KLD est donnée par la relation (5.11).

$$D_{KL}(p, q) = \log\left(\frac{\kappa_p \sinh(\kappa_q)}{\kappa_q \sinh(\kappa_p)}\right) - \left(\frac{1}{\tanh(\kappa_p)} - \frac{1}{\kappa_p}\right) \mu_p^T (\kappa_q \mu_q - \kappa_p \mu_p) \quad (5.11)$$

Dans l'algorithme 5.1, nous avons présenté la démarche pour la réalisation de suivi d'objet dans une séquence d'images omnidirectionnelles en se basant sur le calcul de la similarité. Cette mesure présentée par le KLD. Pour chaque bloc d'image contenant notre objet dans la première image de la séquence, nous calculons la phase de Gabor relative. Ensuite, en utilisant l'estimation de Von Mises Fisher, nous avons déterminé les paramètres κ et μ . Nous calculons le KLD entre le bloc de l'image courante et tous les blocs dans le voisinage délimité par le centre du bloc et l'intervalle $\delta\theta$ et $\delta\phi$. A la fin, nous minimisons la distance pour trouver la nouvelle position du bloc dans l'image suivante de la séquence. Nous répétons cette procédure pour chaque image.

5.5 La Méthode d'évaluation

Étant donné r_t , le bloc trouvé et r_a , le bloc de vérité de terrain ou le résultat souhaité. Le score de "Ovelapping" est défini par la relation $S = \frac{r_t \cap r_a}{r_t \cup r_a}$ où \cap et \cup sont respectivement, l'intersection et l'union entre deux régions. Pour mesurer la performance sur une séquence d'images, on compte le nombre de cas réussis dont le "Ovelapping" S est plus grand qu'un seuil de 20% (trouvée dans l'expérience et qui dépend de la séquence).

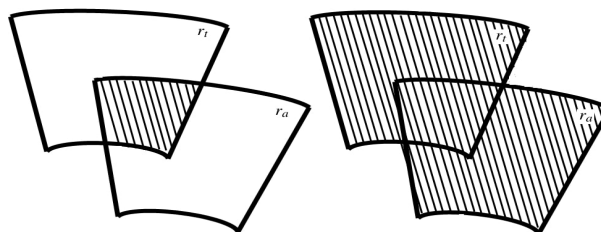


FIGURE 5.4 – Estimation de Overlapping spatial. (gauche.) $r_t \cap r_a$; (droite) $r_t \cup r_a$

Algorithme 5.1 Modèle matching sphérique

Entrées: N : Frame number ; I_1, I_2 : current and next images ; $\delta\theta = \frac{\pi}{40}$; $\delta\phi = \frac{\pi}{15}$

pour $n = 1 : N$ **faire**

Sorties: Initialization : first image block

si $i = 1$ **alors**

Select the object block $g_i(\delta\theta, \delta\phi)$;

Compute Gabor phase P_{g_i}

Estimate VMF parameters $[\kappa_{g_i}, \mu_{g_i}] = VMF(P_{g_i})$

sinon

Entrées: g_i, f_i : image blocks in I_1 and I_2

Sorties: research zone :

The Search zone SZ size is equal to $(M\delta\theta, W\delta\phi)$

$C_i = (p_i, q_i)$ = Center position of g_i

$C_s = (p, q)$ = Center position of Search zone

$p \in [p_i - \frac{M\delta\theta}{2}, p_i + \frac{M\delta\theta}{2}]$

$q \in [q_i - \frac{W\delta\phi}{2}, q_i + \frac{W\delta\phi}{2}]$

pour $i = 1 : size(SZ, 1)$ **faire**

pour $j = 1 : size(SZ, 2)$ **faire**

Compute Gabor phase P_{f_i}

$[\kappa_{f_i}, \mu_{f_i}] = VMF(P_{f_i})$

Compute $kld_{fg}(\kappa_{g_i}, \mu_{g_i}, \kappa_{f_i}, \mu_{f_i})$

Where $f_i(p_i, q_i)$ blocks in the vicinity of g_i .

Storage all kld_{fg}

fin pour

fin pour

$[i, j, kld] = argmin_{\omega} KLD$

Sorties: Block move :

$w_i = i; t_i = j$

(w_i, t_i) = position of f_i

$M_i = [p_i + w_i, q_i + t_i]$

fin si

fin pour

5.6 Résultats et Synthèse

Dans le tableau 5.1, nous avons comparé l'algorithme de MeanShift adapté [26] avec notre approche basée sur la correspondance de modèles adaptés dans des images omnidirectionnelles en terme de Overlapping spatial.

TABLE 5.1 – Adapted Model Block Matching

Seq.	Moyenne	SMM	MeanShift Adapté [94]
Seq.1	Overlapping	70,4%	65%
Seq.2	Spatiale	65,3%	60%

Dans la figure 5.5, nous présentons les résultats de suivi dans deux séquences omnidirectionnelles. La première séquence représente une scène Indoor avec une carte qui bouge autour de la caméra. La deuxième séquence Outdoor illustre un déplacement des personnes autour de la caméra. Dans ces cas de figure, la région suivie est caractérisée par trois couleurs. La couleur bleu représente l'application de l'algorithme de bloc matching classique. La couleur verte représente la vérité de terrain, et la couleur rouge représente notre méthode.

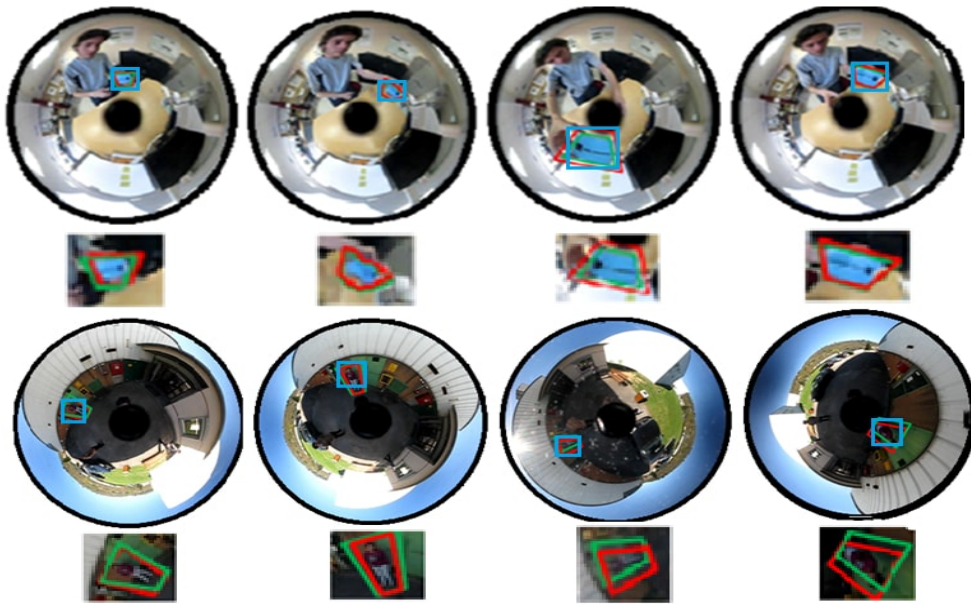


FIGURE 5.5 – Résultats du suivi. Les résultats du suivi avec la correspondance de modèle classique sont représentés en bleu. Les résultats avec la méthode proposée sont affichés en rouge et la vérité terrain en vert.

L'application de la méthode MeanShift pour le suivi est efficace surtout que son algorithme se base sur une décomposition de bloc selon les coordonnées sphériques. C'est une décomposition des blocs qui ne varient pas dans la taille si l'objet a subi de grands déplacements. En revanche, notre algorithme est performant car il s'appuie sur la sélection des blocs qui changent tout le temps leur taille ainsi que le nombre de points dans le voisinage. Ceci entraîne une délimitation précise de bloc lorsque notre objet bouge soit autour de la caméra à la même distance, ou bien s'il se rapproche ou s'éloigne de la caméra.

Les résultats obtenus en utilisant les distributions VonMises-fisher sont plus performants que ceux basés sur la gaussienne. Ceci réside dans le fait que les tests effectués en utilisant la phase

de Gabor donnent plus d'invariance que l'intensité de pixel dans le cas gaussien. Ceci est dû à l'information de pixel qui peut subir des changements dans la séquence d'images dans le cas de suivi à savoir la luminosité et le bruit.

5.7 Conclusion

Dans ce travail, nous présentons une méthode de suivi dans un système catadioptrique basée sur une modélisation avec une approche statistique. Cette approche présentée par la distribution de VonMises Fisher, et adaptée à la géométrie des images omnidirectionnelles. L'adaptation est présentée par la projection d'image sur la sphère. Dans le contexte de suivi, nous avons minimisé la distance entre les modèles en utilisant la mesure KLD pour estimer la position de l'objet en mouvement. Les résultats expérimentaux prennent en compte les changements de taille et de forme du bloc et montrent une bonne performance. Dans les travaux futurs, nous allons opter pour une extension pour le suivi d'objets multiples avec un traitement en temps réel.

LE SUIVI D'OBJET PAR LES MOMENTS INVARIANTS

*Khald, A., Radgui, A. Rziza, M. (2020). Omni-Zernike Algorithm for Template Matching in Catadioptric System. Journal of Computer Science, 16(12), 1789-1795.
<https://doi.org/10.3844/jcssp.2020.1789.1795>*

6.1 Introduction

Parmi les différentes classes de suivi d'objet nous citons l'utilisation des descripteurs d'images qui ont été identifiées et présentées dans ce chapitre. Les descripteurs d'images sont omniprésents dans les applications qui se basent sur l'extraction des caractéristiques des images. Pour cela, un état de l'art proposant un classement par famille et avec une aspect historique a été présenté en plus de revue de la littérature réalisée dans le chapitre 3. Il est possible de mesurer quels sont les descripteurs d'images les plus employés dans le cas de suivi d'objet dans le contexte du système catadioptrique. Le classement proposé dans ce travail est un classement par mode de fonctionnement. Ce choix est aussi fait dans d'autres états de l'art [[39],[38],[37]].

6.2 Classification des méthodes

En plus du classement de l'art de l'art des chercheurs et celui réalisé dans notre état de l'art dans le chapitre 2, un historique de l'apparition des descripteurs d'images les plus représentatifs est proposé dans la Figure 6.1 pour compléter cette présentation. Cet historique référence les descripteurs d'images les plus récents, et il se veut le plus exhaustif et le plus représentatif de l'état de l'art possible.

L'approche retenue pour cet état de l'art doit être originale et adaptable en fonction de la géométrie des images omnidirectionnelles. En effet, l'objectif n'est pas seulement de recenser les applications de suivi réalisées en utilisant les descripteurs d'images de manière indépendante mais il est bien d'identifier quels sont les algorithmes (couple suivi d'objet et descripteur d'images) réellement représentatifs de l'état de l'art. Seuls des exemples d'algorithmes récents, représentatifs et marquants seront présentés.

6.2.1 Les descripteurs colorimétriques

Les descripteurs d'images se basant sur l'information couleur comprennent : l'Gray, l'utilisation de patch (Patch), l'utilisation directe de l'espace couleur (RGB), ou encore des résultats de segmentation. L'utilisation de l'intensité niveaux de gris ou couleur (RGB) consiste à prendre en entrée directement les valeurs de sortie de la caméra. L'utilisation de patches d'intensité (Patch)

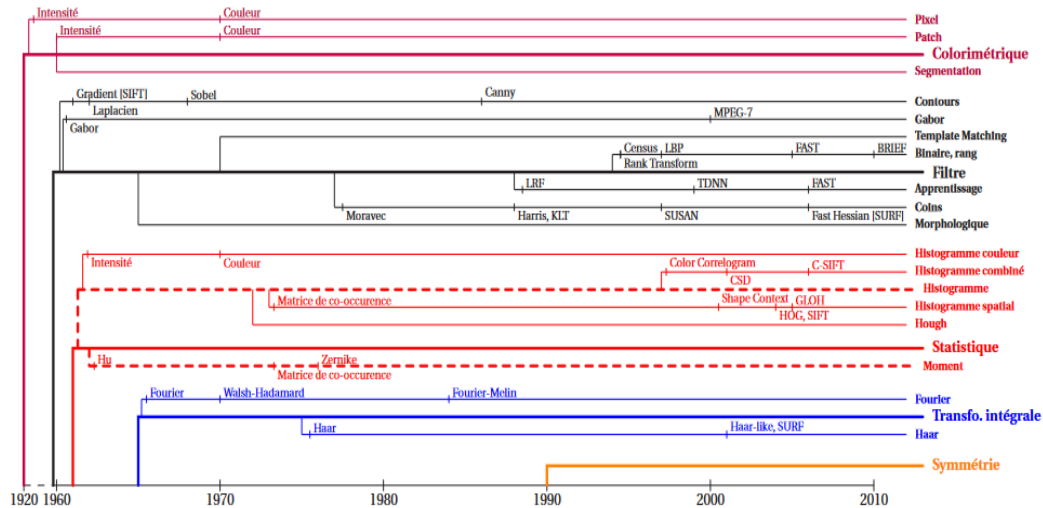


FIGURE 6.1 – Historique des descripteurs d’images les plus représentatifs de l’état de l’art. Cinq grandes familles sont identifiées. [SIFT, SURF] Certains descripteurs d’images sont employés dans des détecteurs de points d’intérêt, ils sont alors précisés entre crochets.

consiste à prendre des patches sur l’image en tant que vecteur de description. La comparaison se fait alors par la mesure de la corrélation entre ces patches. La mesure de corrélation la plus couramment employée est la mesure *Zero Mean Normalized Cross-Correlation* (ZNCC). Cette mesure a été appliquée dans [32] pour le calcul de la distance géodésique adapté à un voisinage sphérique dans une image omnidirectionnelle. Enfin, les techniques de segmentation consistent à rassembler et classifier des régions de l’image en se basant sur des caractéristiques colorimétriques similaires. Les descripteurs basés sur la couleur (RGB) ne doivent pas être confondus avec les techniques de segmentation. En effet, les descripteurs couleur (RGB) utilisent la couleur directement alors que la segmentation regroupe des valeurs en région. Les techniques de segmentation sont très nombreuses, [109] en fait une rapide classification. Comme le montre la Figure 6.1, la famille de descripteurs d’images colorimétriques comporte les premières méthodes employées en traitement et analyse d’images. Bien qu’elles soient simples, elles ont l’avantage d’avoir un temps de calcul quasiment nul. Elles s’avèrent en revanche moins robustes que d’autres descripteurs. Par exemple, l’utilisation de la couleur dépend des conditions d’éclairage [109] alors que l’utilisation de patch est limitée dans le cas d’images à texture répétitive [72].

6.2.2 Les Filtrés

Les filtres ont une image en sortie : chaque pixel ou région de pixel sont donc traités. Les descripteurs d’images de type filtre sont très variés. Les filtres de contours sont les plus employés. Parmi ces derniers, on retrouve des filtres basés sur la dérivée première (par exemple, GradX, GradY, Sobel (celui utilisé dans le suivi par les contour actifs dans le chapitre 3), Canny, et d’autres variantes [36]), des filtres basés sur la dérivé seconde (par exemple, Laplacien), des filtres basés sur des dérivées multiples ou d’autres méthodes moins courantes. Alors que le gradient classique (GradX, GradY) consiste simplement à déterminer la dérivé de l’image par différence de pixels de proche en proche, les deux autres filtres (Sobel, Canny) commencent par effectuer un lissage de l’image avant d’effectuer l’opération de dérivation. Ce filtrage a pour but d’éliminer les faux contours. Le filtre de Canny, datant de 1986 et encore plus développé que le filtre de Sobel de 1968, effectue ensuite une opération de sélection des maxima et de seuillage. Cette étape supplémentaire

permet une fois encore d'éliminer les faux contours dus au bruit de la caméra lors de la capture. Les filtres de coins sont aussi beaucoup utilisés. Ces descripteurs sont basés sur la dérivée première (par exemple, Harris, KLT), sur la dérivée seconde ou encore sur l'utilisation directe de l'intensité avec un filtre de coins spécifique. La plupart de ces descripteurs reprennent des idées du descripteur de Harris. Le filtre de Gabor est aussi fait partie de cette classe de filtre et ils sont déjà introduits dans le chapitre 4. Les filtres binaires et les filtre de rang ont pour point commun de ne pas se focaliser sur les valeurs des pixels mais plutôt sur les relations qui existe entre elles (relation logique), on parle de descripteur non paramétrique. Cette approche est donc originale et différente des autres descripteurs de type filtre. Les descripteurs binaires utilisent un test logique pour créer des valeurs ou des vecteurs de description. Certains comparent par exemple des couples de pixels sur un patch donné (par exemple, *BRIEF*, *Local Binary Pattern (LBP)*, *Census*, *FAST* (Cité dans le chapitre 3)).

6.2.3 Les descripteurs fréquentiels

L'utilisation des descripteurs dans le domaine fréquentiel nous ramène à l'application du transformé de fourrier de l'image et ensuite réaliser le traitement sur les caractéristiques fréquentielles de l'image comme le module et la phase et autres. ils sont nommés des descripteurs de fourrier et ils sont beaucoup employés dans l'extraction des caractéristiques dans la vision par ordinateur. Nous nous plaçons dans ce cadre pour cette partie. Il est clair que les coefficients d'une transformée de Fourier ne peuvent être utilisés tels quels car ils ne sont pas invariants par translation, rotation, ou changement d'échelle. Pour obtenir cette invariance deux familles de descripteurs dédiés aux images en niveaux de gris sont particulièrement cités dans la littérature : les descripteurs de Fourier génériques de D. Zhang et G. Lu [101] et les descripteurs de Fourier généralisés de F. Smach et al. [81].

descripteur de fourrier générique est exprimé par la relation (6.1).

$$\widehat{F}(u, v) = \sum_0^M \sum_0^N f(x, y) \exp^{-2i\pi(\frac{ux}{M} + \frac{vy}{N})} \quad (6.1)$$

où M et N sont respectivement la largeur et la hauteur de l'image f . Par changement de variable, la transformée suivante est obtenue (6.2).

$$\widehat{PF}_1(\rho_1, \psi) = \sum_p \sum_q f(r_p, \theta_q) \cdot r_p \cdot \exp^{-2i\pi r_p \rho_1 \sin(\theta_q + \psi)} \quad (6.2)$$

où (r, θ) (*resp.* (ρ, ψ)) sont les coordonnées polaires de l'image (*resp.* du spectre de Fourier).

Une seconde transformée de fourrier a été développée. PF_2 est tout simplement la transformée de Fourier 2D discrète de l'image passée en coordonnées polaire, ce qui donne (La formule (6.3)).

$$\widehat{PF}_2(\rho_1,) = \sum_k \sum_l f(k, \theta_l) \cdot \exp^{[2i\pi(\frac{k}{R}\rho) + \frac{2}{T}]} \quad (6.3)$$

où $0 \leq k < R$ et $\theta_l = \frac{2}{T}$ avec $0 \leq l < T$ et $0 \leq \rho < R$. R et T sont respectivement la résolution radiale et la résolution angulaire dans le domaine fréquentiel. L'invariance en translation est obtenue en prenant le centre de gravité de l'objet comme origine de l'espace polaire; puis, l'invariance en échelle et en rotation est vérifiée par l'ensemble des descripteurs réels présentés par

la formule (6.4).

$$FD(f) = \left[\frac{PF_2(0,0)}{aire}, \frac{PF_2(0,1)}{PF_2(0,0)}, \dots, \frac{PF_2(m,n)}{PF_2(0,0)} \right] \quad (6.4)$$

où *aire* est la surface du cercle englobant l'objet d'intérêt et *m* et *n* sont respectivement le nombre maximum de fréquences radiales et le nombre de fréquences angulaires sélectionnée.

descripteur de fourier généralisés Les descripteurs de Fourier généralisés introduits par [101] sont construits à partir des actions de groupe et plus particulièrement du groupe des déplacements *M2*. Le groupe *M2* est constitué des translations et des rotations dans le plan. Deux types de descripteurs ont alors été développés :

— Les invariants de type 'densité spectrale' :

$$I_1^r(f) = \int_0^{2\pi} |\hat{f}(r, \theta)|^2 d\theta \quad (6.5)$$

— Les invariants de type 'décalage de phase' :

$$I^{\epsilon_1, \epsilon_2}(f) = \int_0^{2/\pi} \hat{f}(R_\theta(\epsilon_1 + \epsilon_2)) \bar{f}(R_\theta(\epsilon_1)) \bar{f}(R_\theta(\epsilon_2)) d\theta \quad (6.6)$$

où *f* est l'image, $\hat{f}(r, \theta)$ est la transformée de Fourier exprimée en coordonnées polaire dans le plan fréquence ϵ_1, ϵ_2 sont des variables du plan fréquence et R_θ est une rotation d'angle θ . Ces deux descripteurs généralisés sont beaucoup mieux justifiés théoriquement que les descripteurs génériques.

Pour une représentation efficace de l'objet, seulement un petit nombre de caractéristiques GFD (Generalized Fourier Descriptor) sont sélectionnées. Dans l'implémentation de Zhang et Lu [125], 36 caractéristiques GFD reflétant 4 fréquences radiales et 9 fréquences angulaires sont sélectionnées pour indexer l'objet. Les caractéristiques GFD sélectionnées forment un vecteur descripteur qui est utilisé pour indexer un objet. Pour deux formes représentées par leurs descripteurs GFD, la similarité entre ces deux formes est mesurée par la distance City Block entre leurs GFD. Par conséquent, la mise en correspondance en ligne est simple et efficace.

Zhang et Lu ont comparé leur descripteur GFD avec celui basé sur les moments de Zernike (détaillés dans la section 6.2.4), ils ont pu conclure que : le GFD est plus simple à calculer, les attributs sont purement spectraux et montrent de meilleures performances de recouvrement car l'analyse multi-résolution dans les deux directions radiale et spectrale, est possible. Enfin, les auteurs ont montré aussi que le GFD donnait de meilleurs résultats que les moments de Zernike.

6.2.4 Les descripteurs géométriques

Les invariants projectifs

Ce sont les mathématiciens de la Grèce antique qui les premiers ont introduit la géométrie projective. Ils ont énoncé ses propriétés géométriques et découvert son invariant fondamental : le birapport. La géométrie projective a ensuite été utilisée par les peintres de la Renaissance qui cherchaient à rendre l'effet de profondeur dans leurs toiles, c'est à dire à simuler l'oeil et de la vision humaine. C'est au XIX^{ème} siècle que la géométrie projective a été étudiée de manière approfondie et qu'ont été énoncés tous les théorèmes que nous connaissons aujourd'hui. On peut citer les

travaux de Poncelet, qui dans son *Traité des propriétés projectives des figures* (1822) a étudié les propriétés invariantes par une série de projections perspectives d'un plan dans un autre. Klein, dans son discours Erlangen Programme (1872) a généralisé la notion d'invariant associé à un groupe de transformations. Les photogrammétriques ont depuis plusieurs décennies utilisé ces propriétés, mais c'est récemment que la géométrie projective et ses invariants ont fait l'objet d'un intérêt croissant dans la communauté de la vision artificielle.

Les invariants projectifs sont une solution pour extraire des caractéristiques des images avec multiple vues des images. Comment extraire cette information ? C'est une difficulté à surmonter dans un système de vision dans les images 2D. C'est à dire, que les moments sont utilisés pour garder les caractéristiques de l'image avec les changement de luminosité, échelle, rotation, translation et occultation partielle et totale des objets en mouvement.

Parmi les invariants projectifs les plus connus, il y a LE BIRAPPORT. la plupart des invariants projectifs plus complexes peuvent s'exprimer en termes de birapports. Par exemple, la fonction algébrique utilisée par Zisserman et al. [125] pour caractériser deux coniques coplanaires est basée sur les matrices symétriques associées à chacune des coniques. On peut montrer qu'une paire de coniques est caractérisée par des invariants s'exprimant en fonction de birapports. Leur calcul ne nécessite alors que des constructions géométriques simples sur les coniques.

Birapport de quatre points alignés (La figure 6.2) Soient A, B, C, et D quatre points colinéaires, nous définissons leur birapport, noté $[A, B, C, D]$ comme :

$$[A, B, C, D] = \frac{\overline{AC} \times \overline{BD}}{\overline{AD} \times \overline{BC}} \quad (6.7)$$

où \overline{AB} est la mesure algébrique de AB. La conservation du birapport est illustré par la figure 6.2. Ce théorème peut se démontrer en utilisant le théorème de Thalès. Semple et Kneebone donnent une autre démonstration utilisant les secteurs angulaires. Et peut s'exprimer par $[A, B, C, D] = [A', B', C', D']$.

Le birapport d'un faisceau de quatre droites l_1, l_2, l_3, l_4 concourantes en O est défini par le birapport $[A; B; C; D]$ des points d'intersection du faisceau avec une droite quelconque l , qui ne contient pas O. On le note aussi $[O; A; B; C; D]$ (birapport des droites OA;OB;OC;OD).

Le birapport de quatre droites peut s'exprimer sous différentes formes. On peut le calculer en fonction des coordonnées homogènes des points O, A, B, C, D sous la forme proposée par Mobius [81], lorsque les points A, B, C, D ne sont pas alignés (Equation (6.8)).

$$k = \frac{|OAC||OBD|}{|OAD||OBC|} \quad (6.8)$$

De cette expression découle immédiatement l'expression du birapport en fonction des angles entre les droites du faisceau comme montré dans l'équation (6.9).

$$k = \frac{\sin(\overrightarrow{OA}, \overrightarrow{OC}) \sin(\overrightarrow{OB}, \overrightarrow{OD})}{\sin(\overrightarrow{OA}, \overrightarrow{OD}) \sin(\overrightarrow{OB}, \overrightarrow{OC})} \quad (6.9)$$

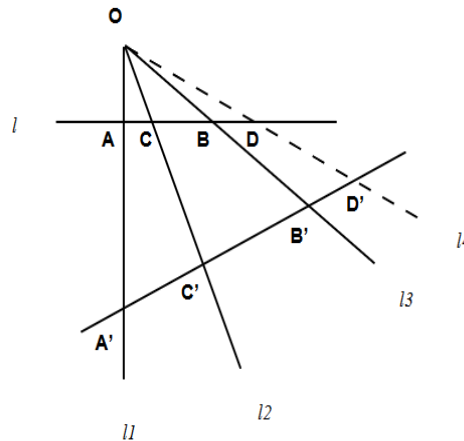


FIGURE 6.2 – Birapport de quatre droites

Un invariant, s'il existe, doit être constant sur chacune des orbites; il prend donc la même valeur.

Les moments invariants

Dans la littérature, nous avons trouvé plusieurs exemples des descripteurs basés sur les moments invariants comme les moments géométriques, Orthogonales, et complexes. Les moments orthogonales compte à eux, ils sont caractérisés par un minimum d'informations de redondance. L'application des moments dans le domaine de la reconstruction et la récupération de caractéristiques dans les images a été introduite par Teague en se basant sur les moments orthogonales [47].

Ces Moments ont été introduits premièrement dans l'extraction de caractéristiques en 1962 par Hu [94] qui les a employé dans le domaine de la théorie de l'algèbre et il a calculé les sept dérivées des moments invariants pour la rotation des objets 2D. Depuis ce temps, de nombreux travaux ont été consacrés à plusieurs améliorations et généralisations des invariants de Hu ainsi qu'à leurs utilisation dans de nombreux domaines d'applications.

Flusser et Suk [114] ont utilisé ces moments dans la correspondance de modèles et la sauvegarde des images satellites. Mukundan [30] les a appliqué pour estimer la position et l'attitude de l'objet dans l'espace 3D. Récemment, un ensemble de fonctions de moments orthogonales discrètes basées sur des polynômes orthogonaux discrets. Les polynômes de Chebyshev [11] et les polynômes de Krawtchouk [57] ont été introduits avec succès comme alternatives aux moments orthogonaux continus.

6.3 Les moments de Zernike

Dans un contexte d'extraction des caractéristiques pour le suivi d'objet, et plus précisément les invariances à l'échelle et à la rotation, les moments de Zernike sont les plus utilisés. Ils sont largement utilisés dans plusieurs applications à savoir la vidéo surveillance, la reconstruction de formes géométriques, transfert de données. Avec leur formulation mathématique, nous pouvons appliquer

ces moments dans un contexte Omnidirectionnel suivant les coordonnées polaires employées dans ces moments dans la phase d'extraction de caractéristiques.

Teague [47] a proposé des moments de Zernike basés sur les polynômes de Zernike orthogonaux. Il est bien connu qu'une fonction d'image discrète peut être reconstruite par les moments de Zernike [128]. Khotanzad et coll [70] ont utilisé les invariants de moments de Zernike pour la reconnaissance et l'estimation de la pose d'objets tridimensionnels. Belkasim et coll. [127] et [88] ont fait une étude comparative sur les invariants des moments de Zernike et les ont utilisés dans la reconnaissance de forme. Ghosal et coll. [86] utilisent les moments de Zernike dans la détection de bord d'objets tridimensionnels. En particulier, les moments de Zernike se sont avérés être invariants en rotation et robustes au bruit. Les moments d'ordre inférieur représentent la forme globale d'un motif et d'ordre supérieur pour les détails.

6.3.1 Les moments de Zernike classiques

Les moments de Zernike ont été introduits sur la base d'une fonction orthogonale continue appelée polynômes de Zernike. Les moments de zernike d'une image numérique peut être calculé en utilisant l'équation (6.10) (

$$Z_{mn} = \frac{n+1}{\pi} \sum_{k=m}^n B_{nmk} \sum_{x=1}^N \sum_{y=1}^M (x-iy)^m (x^2+y^2)^{\frac{k-m}{2}} f(x,y) \quad (6.10)$$

Avec :

$$B_{nmk} = \frac{(-1)^{(n-k)2} \left(\frac{n+k}{2}\right)!}{\left(\frac{n-k}{2}\right)! \left(\frac{k+m}{2}\right)! \left(\frac{k-m}{2}\right)!} \quad (6.11)$$

m représente la dépendance angulaire (il peut être négative ou positive), et n l'ordre du moment. f(x,y) est la fonction décrite.

Les equations de zernike des moments géométriques pour les facteurs de rotation et d'échelle peuvent être dérivées de (6.10) en utilisant (6.11). Ainsi, la grandeur $|Z_{mn}|$ du moment peut être considérée comme une caractéristique invariante de rotation de la fonction d'image sous-jacente comme exprimé dans l'équation (6.12) [61].

$$\begin{aligned} |Z_{20}| &= \left(\frac{3}{\pi}\right)(2(\eta_{20} + \eta_{02}) - \eta_{00}) \\ |Z_{22}|^2 &= \left(\frac{3}{\pi}\right)^2 [(\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2] \\ |Z_{31}|^2 &= \left(\frac{12}{\pi}\right)^2 [(\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2] \\ |Z_{33}|^2 &= \left(\frac{4}{\pi}\right)^2 [(\eta_{30} - \eta_{12})^2 + (\eta_{21} - \eta_{03})^2] \\ |Z_{42}|^2 &= \left(\frac{5}{\pi}\right)^2 ([4(\eta_{40} - \eta_{04}) - 3(\eta_{20} - \eta_{02})]^2 + [6\eta_{11} - 8(\eta_{31} - \eta_{13})]^2) \\ |Z_{40}| &= \left(\frac{5}{\pi}\right)(\eta_{00} - 6(\eta_{20} + \eta_{02}) + 6(\eta_{40} + \eta_{04} + 2\eta_{22})) \end{aligned} \quad (6.12)$$

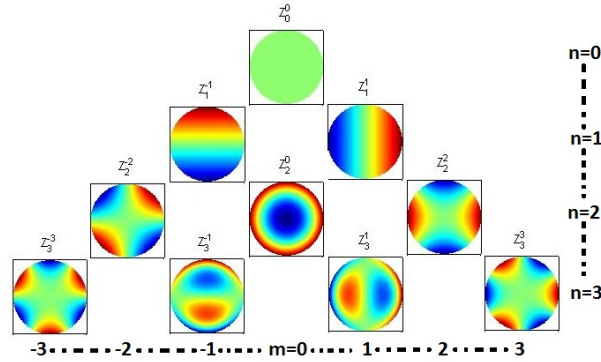


FIGURE 6.3 – Les moments de Zernike avec des valeur de m et n.

6.3.2 Les moments de Zernike Omnidirectionnelles

Le calcul du moment de Zernike via l'équation (6.10) repose sur l'utilisation des coordonnées cartésiennes vu que les images numériques sont représentées par des pixels carrés. Cependant, ce calcul ne prend pas en charge la nature circulaire des polynômes de Zernike et aussi la géométrie des images omnidirectionnelles.

Dans cette section, nous présentons un algorithme pour le calcul des moments de Zernike en coordonnées polaires, dans lequel aucune erreur géométrique ni erreur numérique n'est présente. En conséquence, les moments radiaux d'ordre p et de répétition q sont définis dans l'équation 6.13 et le schéma d'adaptation en espace polaire est présenté dans la figure 6.4

$$D_{pq} = \int_{\theta=0}^{2\pi} \int_{r=0}^{\infty} r^p e^{-jq\theta} dr d\theta, j^2 = -1 \tag{6.13}$$

- Les pixels polaires doivent être aussi "carrés" que possible, c'est-à-dire que les longueurs des limites d'un secteur doivent être suffisamment proches.
- Les pixels polaires doivent être organisés aussi régulièrement que possible pour faciliter le stockage et le calcul.
- Le cercle unitaire est uniformément divisé le long de la direction radiale en sections.

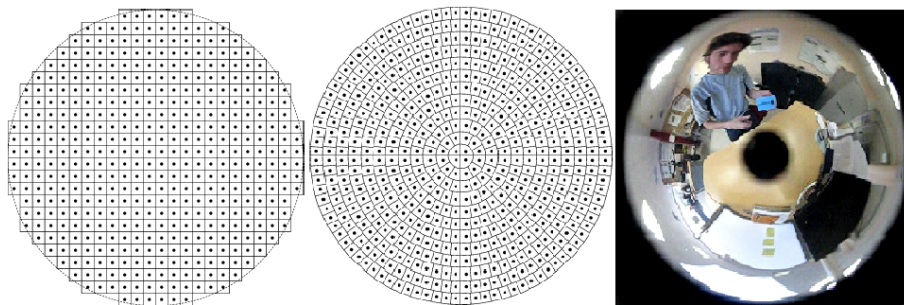


FIGURE 6.4 – Schéma d'adaptation des pixels de l'image perspective en cordonnées polaire

où $p = 0, 1, 2, \dots; \infty$ et q prennent toutes les valeurs entières positives ou négatives. Le noyau des moments de Zernike est un ensemble de polynômes de Zernike orthogonaux définis sur l'espace polaire à l'intérieur d'un cercle unitaire comme l'image omnidirectionnelle normalisée. Les moments de Zernike en deux dimensions d'ordre p avec répétition q d'une fonction d'intensité d'image $f(r, \theta)$ [70] sont définis dans l'équation 6.14.

$$Z_{pq} = \frac{p+1}{\pi} \int_{\theta=0}^{2\pi} \int_{r=0}^{\infty} V^*(r, \theta) f(r, \theta) r dr d\theta, |r| \leq 1 \quad (6.14)$$

avec $V_{pq}(r, \theta)$ sont donné par la relation $V_{pq}(r, \theta) = R_{pq} e^{jq\theta}$. Le polynôme radial à valeur réelle, $R_{pq}(r)$ est exprimé par la relation 6.15.

$$R_{pq}(r) = \sum_{k=0}^{p-\frac{|q|}{2}} (-1)^k \frac{(pk)!}{k! \left(\left(\frac{p+|q|}{2} - k\right)! \left(\frac{p-|q|}{2} - k\right)!\right)} \quad (6.15)$$

Dans un contexte omnidirectionnel, notre image a la même géométrie que celle des moments de Zernike.

La mesure de similarité pour comparer deux descripteurs de Zernike est une simple distance euclidienne entre les moments de Zernike présentée par la formule 6.16.

$$d^2 = \sigma \sigma_{(p,q) \in D} (|Z_{pq}| - |Z_{pq}'|)^2 \quad (6.16)$$

Nous désignons cette distance comme la distance classique. elle est basée sur des modules de moment (deux blocs image sont considérés comme identiques puisque leurs modules sont identiques). Cette conclusion est déduit car l'angle de rotation du bloc ou l'objet est invariante. Ce qui explique que nous pouvons seulement faire une correspondance en utilisant les modules. En revanche, cette approche permet une perte de donnée surtout celle de la phase de notre résultat pour récupérer l'angle de rotation entre les images car ces informations sont codées par la phase de moment.

La mesure donnée tient compte de cette contrainte. Ce score de similarité est plus robuste que cette méthode récupère la phase de deux blocs d'images. L'angle est optimal lorsque la distance euclidienne entre le premier bloc et le second est minimisée.

$$d_{I,J}^2(\theta) = \sum_{(r,\theta)} \sum_{(p,q) \in D} \left[\sum_{(p,q) \in D} Z_{pq}^I \cdot V_{pq}(r, \theta) - \sum_{(p,q) \in D} Z_{pq}^J e^{iq\theta} \cdot V_{pq}(r, \theta) \right]^2 \quad (6.17)$$

Cette distance représente la similarité entre les moments de zernike de chaque image ou bloc d'image.

Cette mesure va être utilisée pour la mise en correspondance entre deux moments du bloc. Chaque bloc est modélisé par un vecteur de moments $Z(p, q)$. pour réaliser un suivi, il suffit de calculer une distance euclidienne entre les vecteurs des moments calculés.

6.4 Résultats et discussions

La figure 6.5 montre que les valeurs de chaque moment ne changent pas beaucoup si nous effectuons des rotation des blocs. Dans ce cas de figure, nous avons testé des rotations de $9^\circ, 45^\circ, 90^\circ$, et 180° . Ces descripteurs sont invariants par rotation de bloc, ce qui permet de conclure que l'utilisation des moments de Zernike dans le suivi va donner des bons résultats.

En plus de l'invariance par totation, nous avons effectué aussi des testes de changement d'échelle pour le blocs de notre image omnidirectionnelle. Nous avons appliqué les échelles de 0, 2, 0, 5, 1, 5,

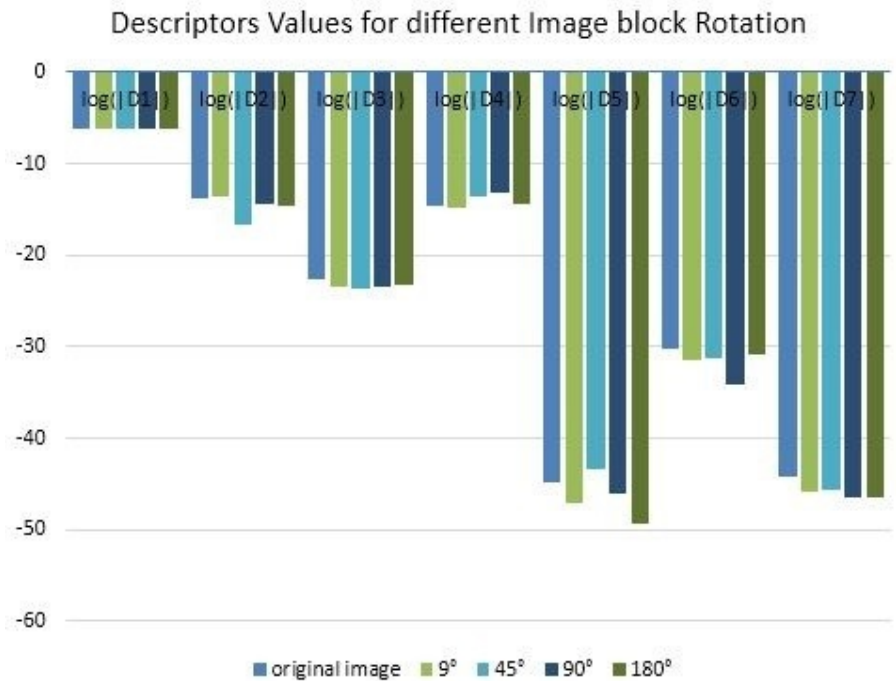


FIGURE 6.5 – Invariance de rotation

et 2. Dans ce cas et comme le montre la figure 6.6, toutes les valeurs de chaque moment subissent des légers changements de valeurs pour chaque échelle.

Ces moments possèdent certains avantages dont :

- Les modules des moments de Zernike sont invariants par rotation [[6], [115]],
- Ils sont robustes aux altérations telles que le bruit, changement de luminosité, etc.,

Une des limitations de ce descripteur, c'est que la complexité de calcul des polynômes de Zernike augmente fortement dans les hautes fréquences de l'image qui sont principalement codées dans les moments d'ordres supérieurs.

Dans une application de suivi, nous allons réaliser la mise en correspondance pour une séquence d'images omnidirectionnelles en se basant sur les invariances de rotation et échelles des régions. Ceci va permettre de suivre efficacement un objet malgré le changement de taille ou de forme.

6.5 Conclusion

Dans ce travail, nous avons réalisé un état de l'art de l'ensemble des descripteurs dans les différents types d'applications à savoir la mise en correspondance, la reconstruction des images, la détection des objets, le calibrage des caméras. Ensuite, nous avons vérifié l'invariance par échelle et rotation en utilisant les moments orthogonaux comme les moments de Zernike. Ces moments sont appliqués directement aux images omnidirectionnelles. Le calcul des moments a été fait sur les blocs de l'image dans un voisinage adapté en fonction des coordonnées sphériques. Dans la perspective de ce travail, nous allons utiliser cette méthode dans le suivi d'objet. Le suivi est justifié en calculant une mesure de similarité entre les moments. Nous minimisons cette distance entre les modèles pour trouver le bloc contenant notre objet après déplacement.

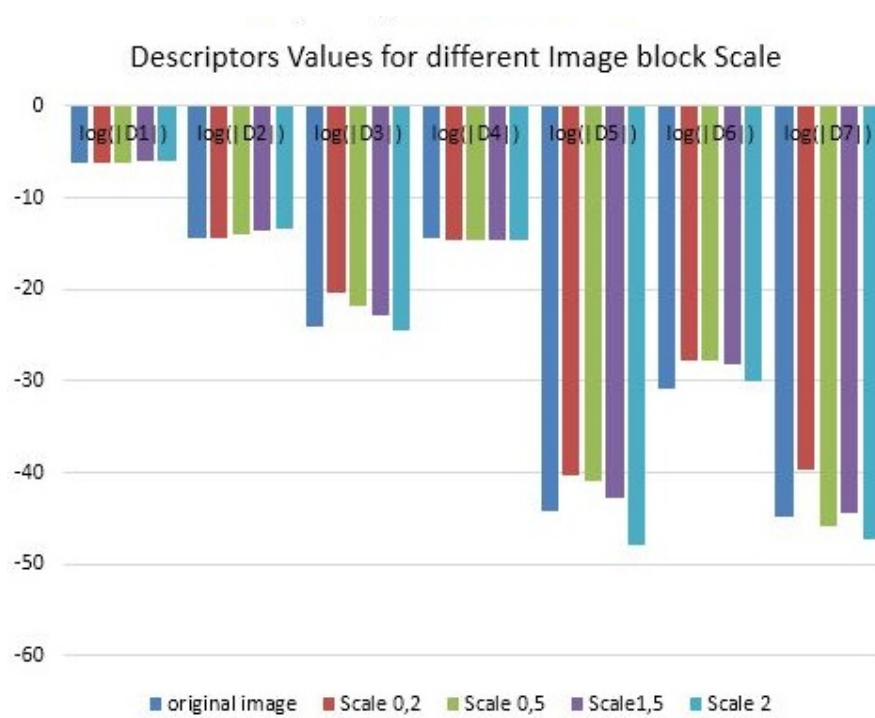


FIGURE 6.6 – Invariance d'échelle



CONCLUSION ET PERSPECTIVES

Dans ce manuscrit, nous nous sommes intéressé au domaine de la vision par ordinateur, et plus précisément le domaine des systèmes catadioptriques qui offre un champ de vision très large. C'est dans cette optique que nous avons mené à bien notre travail, dans le but de réaliser un suivi d'objet dans une séquence d'images omnidirectionnelles. L'étude faite a montrée que les caméras catadioptriques offrent un compromis optimal entre le cout et les performances du système.

La thèse commence par une introduction générale à la vision omnidirectionnelle, y compris une description technique des systèmes de formation des images à champ large. Nous avons présenté dans cette partie les différents capteurs 3D d'acquisition a savoir les systèmes rotatifs, les systèmes à lentille spéciale, . . . Dans Ce travail, nous avons utilisé une caméra catadioptrique avec un système de calibrage qui se base sur un modèle sphérique et en utilisant un miroir de forme parabolique. Ces outils nous ont aidé à réaliser une projection stéréographique inverse des points 3D de la scène environnante sur l'espace de la sphère équivalente de manière efficace.

Pour réaliser le suivi d'objet, nous avons fait le tour dans la littérature pour chercher les méthodes les plus performantes pour notre contexte. L'une des principales contributions de cette thèse est l'adaptation de toutes les méthodes utilisées à la géométrie des images omnidirectionnelles. La nécessité d'adapter le traitement revient à ce que chaque pixel de l'image doit etre traité différemment que se soit au centre ou à la périphérie de l'image.

Nous avons présenté dans ce travail plusieurs résultats pour des méthodes performantes dans le cas classiques et nous les avons adapté dans notre contexte. Après une classification soigneuse des différentes méthodes, nous avons choisis de travailler sur différentes classes :

- La classe des méthodes basées sur le suivi de points : Dans cette classe, nous avons choisis les contours actifs (snake) nous avons adapté toute la formulation mathématique de son algorithme à la géométrie de l'image omnidirectionnelle et en utilisant la projection sphérique.
- La classe des méthodes basées sur le suivi de noyau : Le choix été la modélisation de l'images omnidirectionnelle par une approche statistique et autre spatial. D'une part, en utilisant les distributions de gaussienne généralisée et VonMiseFisher et d'autre part, la méthode qui se base sur les moments invariants (les moments de Zernike).

Apports de ce travail

Les apports de l'étude réalisée pour les systèmes d'acquisition est l'évaluation comparative des caméras catadioptriques en fonction des paramètres et des caractéristiques des miroirs utilisés. Ce système permet d'avoir l'intégralité de l'information qui entoure la caméra.

Le choix de la méthode de traitement repose sur les éléments suivants :

- Il faut que le traitement tiens compte de la géométrie déformée des images omnidirectionnelles.
- Il faut que la méthode choisie soit performante dans le cas des images perspectives.
- Il faut que la méthode soit adaptable dans la formulation et la mise en oeuvre.

Les méthodes utilisées ont montré leurs performances pour l'extraction des caractéristiques puisqu'elles se basent sur l'information de la région interne même si l'objet change de taille ou de forme durant son déplacement.

Perspectives

Nos contributions ont montré une grande performance en terme de mesures de similarité et d'identification d'objet, chose qui permet un développement de plus de techniques de détection d'objet et extraction de caractéristiques et faire plusieurs comparaisons en terme de plusieurs critères.

Pour la validation de notre traitement, nous avons utilisé le mécanisme de seuillage au niveau des blocs suivis. Nous voudrions par la suite passer à des méthodes de décision plus complexe afin de pouvoir évaluer notre système sur différentes échelles à savoir le suivi multi-objet et traitant plusieurs contraintes comme les occultations totales et changement de luminosité, et réaliser un suivi en temps réel.

On compte déployer notre algorithme sur un cas pratique par exemple dans la robotique afin de réaliser un suivi avec aussi une caméra mobile.

6.6 Le calcul du KLD pour la distribution de VonMises Fisher

La démonstration pour arriver la formule finale du KLD entre deux distribution de VonMises Fisher est présentée une dérivation de la divergence de Renyi α [95] pour la distribution vMF et nous la considérons comme une distance utilisée pour déterminer la mesure de similarité.

Nous avons choisi cette divergence car il s'agit d'une mesure statistique de distances généralisées qui comprend les cas des distances de Kullback-Leibler (KL) et Bhattacharyya.

La divergence de Renyi- α est donnée par l'expression 6.18.

$$D_R^{(\alpha)}(p, q) = \frac{1}{\alpha - 1} \log \int_{\psi} p(x)^{\alpha} q(x)^{1-\alpha} dx. \quad (6.18)$$

Elle est caractérisée par le paramètre réel α , qui est dans une limite $\alpha \rightarrow 1$ devient la distance KL définie dans l'équation 6.19.

$$D_{KL}(p, q) = \int_{\psi} \log\left(\frac{p(x)}{q(x)}\right) dx. \quad (6.19)$$

Pour calculer la distance D_F entre deux densités vMF, $p(x; \mu_p; \kappa_p)$ et $q(x; \mu_q; \kappa_q)$ avec leurs paramètres respectifs θ_p et θ_q , nous utilisons les expressions de la forme suivante pour la famille exponentielle des distributions

$$D_F^{(\alpha)}(\theta_p, \theta_q) = \begin{cases} \frac{1}{1-\alpha} J_F^{(\alpha)}(\theta_p, \theta_q), & \alpha \in (0, 1) \\ B_F(\theta_p, \theta_q), & \alpha = 1 \end{cases}$$

Avec B_F définit la divergence de Bregman générée par la fonction convexe F .

$$\begin{cases} B_F(\theta_p, \theta_q) = F(\theta_q) - F(\theta_p) - \Delta F(\theta_p) \cdot (\theta_q - \theta_p), \\ \theta_p, \theta_q \in \eta \end{cases} \quad (6.20)$$

Avec J_F nommée la α -divergence de Jensen générée par la fonction F ,

$$\begin{cases} J_F^{(\alpha)}(\theta_p, \theta_q) = \alpha F(\theta_p) + (1 - \alpha) F(\theta_q) - F(\alpha \theta_p + (1 - \alpha) \theta_q), \\ \theta_p, \theta_q \in \eta. \end{cases} \quad (6.21)$$

La divergence de Bregman est utilisée dans le cas où $\alpha = 1$ puisque $B_F(\theta_p, \theta_q) = D_{KL}(p, q)$, on note que les paramètres sont dans l'ordre inverse. Dans le cas d'une distribution vMF, les calculs montrent que $\alpha \in (0, 1)$

$$D_R^{(\alpha)}(p, q) = \frac{\alpha}{1-\alpha} \log\left(\frac{4\pi \sinh \kappa_p}{\kappa_q}\right) + \log\left(\frac{4\pi \sinh \kappa_q}{\kappa_p}\right) - \left(\frac{1}{1-\alpha}\right) \log\left(\frac{4\pi \sinh(\kappa_{pq})}{\kappa_{pq}}\right) \quad (6.22)$$

Avec $\kappa_{pq} = \|\alpha\mu_p\kappa_p + (1-\alpha)\mu_q\kappa_q\|$. Pour $\alpha = 1$, La distance KL distance, devient :

$$D_{KL}(p, q) = \log\left(\frac{\kappa_p \sinh(\kappa_q)}{\kappa_q \sinh(\kappa_p)}\right) - \left(\frac{1}{\tanh(\kappa_p)} - \frac{1}{\kappa_p}\right) \mu_p^T (\kappa_q \mu_q - \kappa_p \mu_p) \quad (6.23)$$



BIBLIOGRAPHIE

- [1] Optical flow estimation from multichannel spherical image decomposition. *Computer Vision and Image Understanding*, 115(9) :1263 – 1272, 2011.
- [2] J. Ghosh A. Banerjee, I. S. Dhillon and S. Sra. Clustering on the unit hypersphere using von misesâfisher distributions. *Journal of Machine Learning Research*, 6, 2005.
- [3] M. Breitenbach A. S. Bijral and G. Z. Grudic. Mixture of watson distributions : A generative model for hyperspherical embeddings. 2007.
- [4] Daniel Aliaga. Accurate catadioptric calibration for real time pose estimation of room size environments. volume 1, pages 127–134, 01 2001.
- [5] Hadi Aliakbarpour, Omar Tahri, and Helder Araujo. Visual servoing of mobile robots using non-central catadioptric cameras. *Robotics and Autonomous Systems*, 62(11) :1613 – 1622, 2014. Special Issue on Visual Control of Mobile Robots.
- [6] Helene Laurent Anant Choksuriwong, Bruno Emile and Christophe Rosenberger. Comparative study of global invariant descriptors for object recognition. *Journal of Electronic imaging*, 17(2), 2008.
- [7] Fatima Aziz, Ouiddad Labbani-Igbida, Amina Radgui, and Ahmed Tamtaoui. Generic spatial-color metric for scale-space processing of catadioptric images. *Computer Vision and Image Understanding*, 176-177 :54 – 69, 2018.
- [8] P. Baker, C. Fermuller, Y. Aloimonos, and R. Pless. A spherical eye from multiple cameras (makes better models of the world). volume 1, pages I–I, Dec 2001.
- [9] Simon Baker and Shree K. Nayar. Single viewpoint catadioptric cameras. 04 2000.
- [10] Simon Baker and Shree K. Nayar. A theory of single-viewpoint catadioptric image formation. *Int. J. Comput. Vision*, 35(2) :175–196, November 1999.
- [11] H. Bakstein and A. Leonardis. Catadioptric image-based rendering for mobile robot localization. pages 1–6, Oct 2007.
- [12] Y. Bar-Shalom. Tracking and data association. *Academic Press Professional*.
- [13] J. P. Barreto and H. Araujo. Geometric properties of central catadioptric line images and their application in calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8) :1327–1333, Aug 2005.

- [14] P. J. Barreto and H. Araujo. Issues on the geometry of central catadioptric image formation. pages 422–427, 2001.
- [15] Yalin Bastanlar, Luis Puig, Peter Sturm, Josechu Guerrero, and Joao Barreto. Dlt-like calibration of central catadioptric cameras. 08 2008.
- [16] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3) :346 – 359, 2008. Similarity Matching in Computer Vision and Multimedia.
- [17] Jean Bazin, Kuk-Jin Yoon, In-So Kweon, CÃ©dric D'haeseleer, and Pascal Vasseur. Particle filter approach adapted to catadioptric images for target tracking application. 01 2009.
- [18] F.a d Benamar, S.a b El Fkihi, C.c D'haeseleer, E.d Mouaddib, and D.a Aboutajdine. Gradient-based time to contact on paracatadioptric camera. pages 5–9, 2013.
- [19] Stephanie Bigot-Marchand, Djemaa Kachi, and Sylvain Durand. Spherical edge detector : application to omnidirectional imaging. Number 5259, pages 554–565, France, 2008.
- [20] Stephanie Bigot-Marchand, Djemaa Kachi, Sylvain Durand, and El Mustapha Mouaddib. Spherical image denoising and its application to omnidirectional imaging. Spain, 2007.
- [21] C. Bingham. An antipodally symmetric distribution on the sphere. *The Annals of Statistics*, 2, Nov. 1974.
- [22] S.S. Blackman and R. Popoli. Design and analysis of modern tracking systems. *Artech House*.
- [23] R. Boutteau, X. Savatier, and J. Y. Ertaud. A dynamic programming algorithm applied to omnidirectional vision for dense 3d reconstruction. pages 927–931, Nov 2013.
- [24] Jack Bresenham. A linear algorithm for incremental digital display of circular arcs. *Commun. ACM*, 20(2) :100–106, February 1977.
- [25] Vincenzo Caglioti, Pierluigi Taddei, Giacomo Boracchi, Simone Gasparini, and Alessandro Giusti. Single-image calibration of off-axis catadioptric cameras using lines. pages 1–6, 10 2007.
- [26] Guillaume Caron, El Mustapha Mouaddib, and Eric Marchand. 3d model based tracking for omnidirectional vision : A new spherical approach. *Robotics and Autonomous Systems*, 60(8) :1056 – 1068, 2012.
- [27] A. Chiuso and G. Picci. Visual tracking of points as estimation on the unit sphere. volume 237, page 90â105, 2001.
- [28] Dorin Comaniciu, Visvanathan Ramesh, and Peter Meer. Real-time tracking of non-rigid objects using mean shift. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2, 05 2000.
- [29] Diana Margarita Cordova, Esparza, Jose Joel Gonzalez, Barbosa, Juan Bautista Hurtado, Ramos, and Francisco Javier Ornelas, Rodriguez. A panoramic 3d reconstruction system based on the projection of patterns. *International Journal of Advanced Robotic Systems*, 11 :1, 04 2014.

- [30] K. Daniilidis, A. Makadia, and T. Bulow. Image processing in catadioptric planes : spatio-temporal derivatives and optical flow computation. pages 3–10, 2002.
- [31] Tomas Svoboda David Hurych, Karel Zimmermann. Fast learnable object tracking and detection in high-resolution omnidirectional images. 1999.
- [32] C. Demonceaux and P. Vasseur. Omnidirectional image processing using geodesic metric. In *2009 16th IEEE International Conference on Image Processing (ICIP)*, pages 221–224, 2009.
- [33] CÃ©dric Demonceaux and Pascal Vasseur. Omnidirectional image processing using geodesic metric. pages 221 – 224, 12 2009.
- [34] Xiao-Ming DENG, Fuchao Wu, and Yihong Wu. An easy calibration method for central catadioptric cameras. *Acta Automatica Sinica*, 33 :801–808, 08 2007.
- [35] T.R. Diethel. A note on the kullback-leibler divergence for the von mises-fisher distribution. *Department of Electrical and Electronic Engineering, University of Bristol*, 1, 2015.
- [36] DALAL et B TRIGGS. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005.
- [37] Krystian MIKOLAJCZYK et Cordelia SCHMID. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27, 2005.
- [38] WY MA et HJ ZHANG. Benchmarking of image features for content-based retrieval. *32nd Asilomar Conference on Signals, Systems Computers*, 1998.
- [39] Jing LI et Nigel M. ALLINSON. A comprehensive review of current local features for computer vision. *Neurocomputing*, 71, 2008.
- [40] Espuny. Ferran and Burgos Gil jose. Generic self calibration of central cameras from two real rotational flows. 10 2014.
- [41] R. T. Fomena, H. U. Yoon, A. Cherubini, F. Chaumette, and S. Hutchinson. Coarsely calibrated visual servoing of a mobile robot using a catadioptric vision system. pages 5432–5437, Oct 2009.
- [42] Oliver Frank, Roman Katz, Christel-Loïc Tisse, and Hugh F. Durrant-Whyte. Camera calibration for miniature, low-cost, wide-angle imaging systems. 2007.
- [43] R. T. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4) :439–451, July 1988.
- [44] S. Julier G. Kurz, I. Gilitschenski and U. D. Hanebeck. Recursive bingham filter for directional estimation involving 180 degree symmetry. *Journal of Advances in Information Fusion*, 9, Dec. 2014.
- [45] J. Gaspar, N. Winters, and J. Santos-Victor. Vision-based navigation and environmental representations with an omnidirectional camera. *IEEE Transactions on Robotics and Automation*, 16(6) :890–898, Dec 2000.

- [46] C. Geyer and K. Daniilidis. Paracatadioptric camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5) :687–695, May 2002.
- [47] Christopher Geyer and Kostas Daniilidis. Catadioptric projective geometry. *International Journal of Computer Vision*, 45(3) :223–243, Dec 2001.
- [48] J. Glover and L. P. Kaelbling. Tracking the spin on a ping pong ball with the quaternion bingham filter. 2014.
- [49] Jose.Joel Gonzalez.Barbosa. A panoramic 3d reconstruction system based on the projection of patterns. *International Journal of Advanced Robotic Systems*, 11, 02 2014.
- [50] Helmut Grabner, Jiri Matas, Luc Van Gool, and Philippe C. Cattin. Tracking the invisible : Learning where the object might be. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1285–1292, 2010.
- [51] Valdir Grassi Jr and Jun Okamoto Jr. Development of an omnidirectional vision system. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, XXVIII, 03 2006.
- [52] Yanwen Guo, Ye Chen, Feng Tang, Ang Li, Weitao Luo, and Mingming Liu. Object tracking using learned feature manifolds. *Computer Vision and Image Understanding*, 118 :128 – 139, 2014.
- [53] Hao Tang, S. M. Chu, and T. S. Huang. Generative model-based speaker clustering via mixture of von mises-fisher distributions. pages 4101–4104, April 2009.
- [54] D Hurych, K Zimmermann, and T Svoboda. Fast learnable object tracking and detection in high-resolution omnidirectional images. 2011.
- [55] F. Chaumette I. Markovic and I. Petrovic. Moving object detection, tracking and following using an omnidirectional camera on a mobile robot. Jun. 2014.
- [56] J. Cesic I. Markovic, M. Bukal and I. Petrovic. Direction-only tracking of moving objects on the unit sphere via probabilistic data association. Jun. 2014.
- [57] Kohei Ikeda, Ryuichi Yamaguchi, Norikazu Ikoma, Hideaki Kawano, and Hiroshi Maeda. Tracking of multiple moving objects in dynamic image of omni-directional camera for robot vision using phd filter. *Proceedings of the Fuzzy System Symposium*, 22 :14–14, 2006.
- [58] Hiroshi Ishiguro. Development of low-cost compact omnidirectional vision sensors and their applications. 01 1998.
- [59] Florence Jacquy, Frederic Comby, and Olivier Strauss. Fuzzy edge detection for omnidirectional images. *Fuzzy Sets and Systems*, 159(15) :1991 – 2010, 2008.
- [60] Jae-Hean Kim and Myung Jin Chung. Slam with omni-directional stereo vision sensor. volume 1, pages 442–447 vol.1, Oct 2003.
- [61] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi. Robust online appearance models for visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10) :1296–1311, Oct 2003.

- [62] J. Kannala and S. S. Brandt. A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(8) :1335–1340, Aug 2006.
- [63] J. T. Kent and T. Hamelryck. Using the fisher-bingham distribution in stochastic models for protein structure. *Quantitative Biology, Shape Analysis, and Wavelets*, 24, 2005.
- [64] A. Khald, A. Radgui, and M. Rziza. Objects tracking in catadioptric images using spherical snake. volume 3, pages 435–440, Jan 2014.
- [65] A. Khald, A. Radgui, and M. Rziza. Adaptive template matching model for object tracking in catadioptric system. volume 18, pages 37–44, Jan 2018.
- [66] Morales L-P. Omnidirectional multiview systems : calibration , features and 3d information. *These soutenue en decembre*, 2011.
- [67] P. Leong and S. Carlile. Methods for spherical data analysis and visualization. *Journal of Neuroscience Methods*, 80, 1998.
- [68] Shih-Schon Lin and Ruzena Bajcsy. Single-viewpoint, catadioptric cone mirror omnidirectional imaging theory and analysis. *J. Opt. Soc. Am. A*, 23(12) :2997–3015, Dec 2006.
- [69] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2) :91–110, November 2004.
- [70] Jianhua Lu and M. L. Liou. A simple and efficient search algorithm for block-matching motion estimation. *IEEE Transactions on Circuits and Systems for Video Technology*, 7(2) :429–433, Apr 1997.
- [71] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. IJCAI’81, pages 674–679, San Francisco, CA, USA, 1981. Morgan Kaufmann Publishers Inc.
- [72] S. LACROIX et L. GALLO MALLET. Position estimation in outdoor environments using pixel tracking and stereovision. *IEEE ICRA 2000*, 4, 2000.
- [73] K. V. Mardia and P. E. Jupp. 1998.
- [74] I. Markovic, M. Bukal, J. Cestic, and I. Petrovic. Direction-only tracking of moving objects on the unit sphere via probabilistic data association. In *17th International Conference on Information Fusion (FUSION)*, pages 1–7, July 2014.
- [75] I. Markovic, J. cestic, and I. Petrovic. Von mises mixture phd filter. *IEEE Signal Processing Letters*, 22(12) :2229–2233, Dec 2015.
- [76] C. Mei and E. Malis. Fast central catadioptric line extraction, estimation, tracking and structure from motion. pages 4774–4779, Oct 2006.
- [77] C. Mei and P. Rives. Single view point omnidirectional camera calibration from planar grids. pages 3945–3950, April 2007.
- [78] Christopher Mei, Selim Benhimane, Ezio Malis, and Patrick Rives. Constrained multiple planar template tracking for central catadioptric cameras. pages 619–628, 01 2006.

- [79] Christopher Mei, Selim Benhimane, Ezio Malis, and Patrick Rives. Homography-based tracking for central catadioptric cameras. *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 669–674, 2006.
- [80] B. Micusik and T. Pajdla. Structure from motion with wide circular field of view cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(7) :1135–1149, July 2006.
- [81] A.F. Mobius. 1885.
- [82] Nouredine Mohtaram, Amina Radgui, Guillaume Caron, and El Mustapha Mouaddib. AMIFT : AFFINE-MIRROR INVARIANT FEATURE TRANSFORM. Athenes, Greece, October 2018.
- [83] O. Morel and D. Fofi. Calibration of catadioptric sensors by polarization imaging. pages 3939–3944, April 2007.
- [84] Harsh Nanda and Ross Cutler. Practical calibrations for a real-time digital omnidirectional camera. Technical report, In *Technical Sketches, Computer Vision and Pattern Recognition*, 2001.
- [85] Tatsuya NARUSE, Atsushi Yamashita, Toru KANEKO, and Yuichi Kobayashi. 3d measurement of objects in water using fish-eye stereo camera. *Journal of the Japan Society for Precision Engineering*, 79 :344–348, 01 2013.
- [86] Yao Nie and Kai-Kuang Ma. Adaptive rood pattern search for fast block-matching motion estimation. *IEEE Transactions on Image Processing*, 11(12) :1442–1449, Dec 2002.
- [87] Luis Paya, Oscar Reinoso, Luis M. Jimenez, and Miguel Julia. Estimating the position and orientation of a mobile robot with respect to a trajectory using omnidirectional imaging and global appearance. *PLOS ONE*, 12(5) :1–25, 05 2017.
- [88] Lai-Man Po and Wing-Chung Ma. A novel four-step search algorithm for fast block motion estimation. *IEEE Transactions on Circuits and Systems for Video Technology*, 6(3) :313–317, Jun 1996.
- [89] L. Puig and J. J. Guerrero. Scale space for central catadioptric systems : Towards a generic camera feature extractor. pages 1599–1606, Nov 2011.
- [90] Luis Puig, J. Bermudez, Peter Sturm, and J.J. Guerrero. Calibration of omnidirectional cameras in practice : A comparison of methods. *Computer Vision and Image Understanding*, 116(1) :120 – 137, 2012.
- [91] A. Radgui, C. Demonceaux, E. Mouaddib, M. Rziza, and D. Aboutajdine. Optical flow estimation from multichannel spherical image decomposition. *Computer Vision and Image Understanding*, 115(9) :1263 – 1272, 2011.
- [92] A. Radgui, Cedric Demonceaux, El Mustapha Mouaddib, D. Aboutajdine, and M. Rziza. An adapted lucas-kanade’s method for optical flow estimation in catadioptric images. Marseille, France, Oct 2008. Rahul Swaminathan and Vincenzo Caglioti and Antonis Argyros.

- [93] Srikumar Ramalingam, Peter Sturm, and Suresh Lodha. Towards generic self-calibration of central cameras, omnivis. *6th Workshop on Omnidirectional Vision, Camera Networks and Non-Classical Cameras*, 10 2005.
- [94] Francois Rameau, Desire Sidibe, Cedric Demonceaux, and David Fofi. Visual tracking with omnidirectional cameras : an efficient approach. *Electronics Letters*, 47(21) :1183 – 1184, October 2011.
- [95] A. Renyi. Measures of entropy and information. *in Proc. Fourth Berkeley Symp. Math. Stat. and Probability, University of California Press*, 1, 2010.
- [96] R. Rossi, X. Savatier, J. Ertaud, and B. Mazari. Real-time 3d reconstruction for mobile robot using catadioptric cameras. pages 104–109, Nov 2009.
- [97] E. Rosten, R. Porter, and T. Drummond. Faster and better : A machine learning approach to corner detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1) :105–119, Jan 2010.
- [98] Kenshi Saho. Kalman filter for moving object tracking : Performance analysis and filter design. 2018.
- [99] D. Scaramuzza, A. Martinelli, and R. Siegwart. A flexible technique for accurate omnidirectional camera calibration and structure from motion. pages 45–45, Jan 2006.
- [100] Sing Bing Kang. Catadioptric self-calibration. volume 1, pages 201–207 vol.1, June 2000.
- [101] Lemaître C. Gauthier J.P. Miteran J. Atri M. Smach, F. Generalized fourier descriptors with applications to objects recognition in svm context. *Journal of Mathematical Imaging and Vision*, 30, 2008.
- [102] S. Spors and R. Rabenstein. A real-time face tracker for color video. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, volume 3, pages 1493–1496 vol.3, May 2001.
- [103] Vitomir Struc and Nikola Pavesic. From gabor magnitude to gabor phase features : Tackling the problem of face recognition under severe illumination changes. *Face Recognition, In-Tech, Vienna*, 1, 2010.
- [104] T. Svoboda and Tomas Pajdla. Epipolar geometry for central catadioptric cameras. *International Journal of Computer Vision*, 49 :23–37, 08 2002.
- [105] R. Swaminathan, M. D. Grossberg, and S. K. Nayar. Caustics of catadioptric cameras. volume 2, pages 2–9 vol.2, July 2001.
- [106] C. Toepfer and T. Ehlgen. A unifying omnidirectional camera model and its applications. pages 1–5, Oct 2007.
- [107] Carlo Tomasi and Takeo Kanade. Detection and tracking of point features. 1991.
- [108] J. Traa and P. Smaragdis. Multiple speaker tracking with the factorial von misesâfisher filter. *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, Sep. 2014.

- [109] M ZERVAKIS et K KALAITZAKIS V KASTRINAKI. A survey of video processing techniques for traffic applications. *Image and Vision Computing*, 21, 2003.
- [110] B. Vandeportaele, M. Cattoen, P. Marthon, and P. Gurdjos. A new linear calibration method for paracatadioptric cameras. volume 4, pages 647–651, Aug 2006.
- [111] Pascal Vasseur and Mustapha Mouaddib. Central catadioptric line detection. 01 2004.
- [112] N. Vlassis, Y. Motomura, I. Hara, H. Asoh, and T. Matsui. Edge-based features from omnidirectional images for robot localization. volume 2, pages 1579–1584 vol.2, May 2001.
- [113] Tomas Vojir, Jana Noskova, and Jiri Matas. Robust scale-adaptive mean-shift for tracking. *Pattern Recognition Letters*, 49 :250 – 258, 2014.
- [114] F. Wallhoff, M. Zobl, G. Rigoll, and I. Potucek. Face tracking in meeting room scenarios using omnidirectional views. volume 4, pages 933–936, Aug 2004.
- [115] Ake Wallin and Olaf KÅ¼bler. Complete sets of complex zernike moment invariants and the role of the pseudoinvariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11, 1995.
- [116] Donna J. Williams and Mubarak Shah. A fast algorithm for active contours and curvature estimation. *CVGIP : Image Understanding*, 55(1) :14 – 26, 1992.
- [117] A. Wood. Simulation of the von mises-fisher distribution. *Communications in statistics â simulation and computation*, 23, 1994.
- [118] Fuchao Wu, Fuqing Duan, Zhanyi Hu, and Yihong Wu. A new linear algorithm for calibrating central catadioptric cameras. *Pattern Recognition*, 41(10) :3166 – 3172, 2008.
- [119] Y. Wu, Y. Li, and Z. Hu. Easy calibration for para-catadioptric-like camera. pages 5719–5724, Oct 2006.
- [120] Xianghua Ying and Hongbin Zha. Simultaneously calibrating catadioptric camera and detecting line features using hough transform. pages 412–417, Aug 2005.
- [121] Fan Yang, Huchuan Lu, and Ming-Hsuan Yang. Learning structured visual dictionary for object tracking. *Image Vision Comput.*, 31(12) :992–999,.2013.09.008, dec 2013.
- [122] M. Yang, Y. Wu, and G. Hua. Context-aware visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(7) :1195–1209, July 2009.
- [123] Xianghua Ying and Zhanyi Hu. Catadioptric camera calibration using geometric invariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10) :1260–1271, Oct 2004.
- [124] Lin Zeng, Weigang Zhang, Shengping Zhang, and Dingwen Wang. Video image mosaic implementation based on planar-mirror-based catadioptric system. *Signal, Image and Video Processing*, 8, 09 2012.
- [125] Dengsheng Zhang and Guojun Lu. Generic fourier descriptor for shape-based image retrieval. *Multimedia and Expo, 2002. ICMEâ02*, 1, 2002.

- [126] Lu G. Zhang, D. Shape based image retrieval using generic fourier descriptors. *Signal Processing : Image Communication*, 17, 2002.
- [127] Shan Zhu and Kai-Kuang Ma. A new diamond search algorithm for fast block-matching motion estimation. *IEEE Transactions on Image Processing*, 9(2) :287–290, Feb 2000.
- [128] Shiyu Liu Zuoliang Cao. Dynamic omni-directional vision localization using a beacon tracker based on particle filter. volume 6764, pages 6764 – 6764 – 10, 2007.