



HAL
open science

Reconstruction 3D dense d'objets sans recul par vision catadioptrique

Noureddine Mohtaram

► **To cite this version:**

Noureddine Mohtaram. Reconstruction 3D dense d'objets sans recul par vision catadioptrique. Autre. Université de Picardie Jules Verne; Université Mohammed V-Agdal (Rabat, Maroc; 1993-2014), 2019. Français. NNT : 2019AMIE0023 . tel-03638162

HAL Id: tel-03638162

<https://theses.hal.science/tel-03638162v1>

Submitted on 12 Apr 2022

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

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



Thèse de Doctorat

Mention : Science pour l'ingénieur

Spécialité : Vision par ordinateur

présentée à *l'Ecole Doctorale en Sciences Technologie et Santé (ED 585)*

de l'Université de Picardie Jules Verne

Noureddine MOHTARAM

pour obtenir le grade de Docteur de l'Université de Picardie Jules Verne

***Reconstruction 3D dense d'objets sans recul par vision
catadioptrique***

Soutenue le 19 septembre après avis des rapporteurs, devant le jury d'examen :

M^{me} Véronique Cherfaoui, Professeur UTC

Présidente

M^{me} Samia Ainouz, MCF HDR INSA de Rouen

Rapporteur

M. Marc Pierrot-Desseilligny, Directeur de Recherche ENSG-IGN

Rapporteur

M. Claude Pégard, Professeur UPJV

Examineur

M. Guillaume Caron, MCF UPJV

Examineur

M^{me} Amina Radgui, MCF HDR, INPT Rabat (Maroc)

Encadrante

M. El Mustapha Mouaddib, Professeur UPJV

Directeur de thèse





TABLE DES MATIÈRES

| | |
|---|-----------|
| Liste des figures | ix |
| Liste des tableaux | x |
| Liste des algorithmes | xi |
| Chapitre 1 : Introduction générale | 3 |
| 1.1 Contexte | 3 |
| 1.2 Problématique de la thèse | 4 |
| 1.3 Contributions | 6 |
| 1.4 Plan du mémoire | 7 |
| Chapitre 2 : Systèmes catadioptriques planaires : Etat de l'art | 9 |
| 2.1 Introduction | 9 |
| 2.2 Modélisation des systèmes catadioptriques planaires | 10 |
| 2.2.1 Modélisation d'un système catadioptrique planaire à un miroir | 10 |
| 2.2.2 Modélisation d'un système catadioptrique planaire à deux miroirs | 12 |
| 2.3 Techniques de Reconstruction 3D à partir des SCPs | 13 |
| 2.3.1 Systèmes actifs | 14 |
| 2.3.2 Systèmes passifs | 16 |
| 2.3.2.1 Shape-From-silhouette | 16 |
| 2.3.2.2 Shape-From-shading | 18 |
| 2.4 Conclusion | 18 |
| Chapitre 3 : Étude et calibration du système catadioptrique planaire | 21 |
| 3.1 Introduction | 21 |
| 3.2 Étude du système catadioptrique planaire | 22 |
| 3.2.1 Généralités et définitions autour du SCP | 22 |
| 3.2.2 Vue d'ensemble et notations | 24 |
| 3.2.3 Modélisation de caméra perspective | 26 |
| 3.2.4 Paramètres internes | 27 |
| 3.2.5 Paramètres externes | 27 |
| 3.3 Étalonnage du système catadioptrique planaire | 29 |
| 3.3.1 État de l'art de l'étalonnage de SCP | 30 |
| 3.3.2 Etalonnage de caméra perspective | 30 |
| 3.3.3 Calcul de pose des miroirs et extraction des normales et distances | 31 |
| 3.3.4 Expérimentations et évaluation de l'étalonnage | 31 |

| | | |
|---------|--|----|
| 3.3.4.1 | Résultats sur des données de synthèse | 32 |
| 3.3.4.2 | Résultats sur des données réelles | 34 |
| 3.4 | Étude de la géométrie épipolaire réflexive du SCP | 34 |
| 3.4.1 | Matrice fondamentale réflexive | 36 |
| 3.4.2 | Matrice essentielle réflexive | 38 |
| 3.4.2.1 | Détermination de la matrice essentielle réflexive | 39 |
| 3.4.2.2 | Méthode basée sur la pose relative : | 39 |
| 3.4.2.3 | Méthode basée sur l'équation du plan : | 40 |
| 3.4.3 | Validation de la robustesse de la matrice fondamentale réflexive F_r | 40 |
| 3.5 | Conclusion | 43 |

Chapitre 4 : Mise en correspondance de points dans des images acquises par un système catadioptrique planaire 45

| | | |
|---------|---|----|
| 4.1 | Introduction | 45 |
| 4.2 | Synthèse | 46 |
| 4.3 | Mise en correspondance éparse basée AMIFT | 48 |
| 4.3.1 | Prise en compte de la transformation de réflexion | 48 |
| 4.3.1.1 | Ré-organisation d'un descripteur | 49 |
| 4.3.1.2 | Ordre des 16 cellules | 49 |
| 4.3.2 | Prise en compte de la transformation affine | 51 |
| 4.3.3 | Réflexive-RANSAC | 55 |
| 4.3.3.1 | Estimation de la matrice fondamentale | 55 |
| 4.3.3.2 | Méthode non linéaire : Symmetric-RANSAC | 55 |
| 4.4 | Validation expérimentale | 56 |
| 4.4.1 | Données d'évaluation | 56 |
| 4.4.2 | Évaluation quantitative | 56 |
| 4.4.2.1 | Métrique | 56 |
| 4.4.2.2 | Comparaison d'AMIFT avec les descripteurs existants : | 57 |
| 4.5 | Conclusion | 60 |

Chapitre 5 : Mise en correspondance de points dans des images acquises par un SCP basée homographies locales 61

| | | |
|---------|--|----|
| 5.1 | Introduction | 61 |
| 5.2 | Modélisation | 62 |
| 5.3 | Méthode proposée pour la recherche des pixels correspondants | 67 |
| 5.3.1 | Restriction aux pixels de fortes variations de contraste de la vue directe | 67 |
| 5.3.2 | Paramètres à estimer | 67 |
| 5.3.3 | Estimation robuste de l'homographie | 68 |
| 5.3.3.1 | Définition du critère de correspondance | 69 |
| 5.3.3.2 | Méthode d'optimisation Branch-and-Bound | 70 |
| 5.3.3.3 | Synthèse de l'algorithme proposé | 71 |
| 5.4 | Validation et évaluation de la méthode proposée | 73 |
| 5.4.1 | Stabilité de l'algorithme | 73 |
| 5.4.1.1 | Influence du paramètre δ sur la précision | 73 |
| 5.4.1.2 | Influence de la taille de la fenêtre | 74 |
| 5.4.2 | Critère d'évaluation existants | 74 |
| 5.4.3 | Résultats et apports de notre algorithme | 75 |

| | | |
|---|--|------------|
| 5.5 | Conclusion | 84 |
| Chapitre 6 : Application à la reconstruction 3D d'objets | | 85 |
| 6.1 | Reconstruction 3D | 86 |
| 6.2 | Résultats expérimentaux | 88 |
| 6.2.1 | Données synthétiques | 88 |
| 6.2.1.1 | Reconstruction d'un plan | 88 |
| 6.2.1.2 | Reconstruction d'un objet multi-plan | 89 |
| 6.2.1.3 | Objets de formes géométriques variées | 90 |
| 6.2.2 | Evaluation de la précision de la reconstruction 3D | 92 |
| 6.2.3 | Comparaison avec la méthode des silhouettes | 93 |
| 6.3 | Conclusion | 96 |
| Chapitre 7 : Conclusion générale et perspectives | | 97 |
| 7.1 | Bilan des travaux réalisés | 97 |
| 7.2 | Quelques perspectives | 98 |
| Bibliographie | | 99 |
| Résumé | | 107 |
| Abstract | | 107 |



LISTE DES FIGURES

| | | |
|-----|--|----|
| 1.1 | Quelques résultats du projet E-cathédrale du nuage de points de l'intérieur et de l'extérieur de la cathédrale d'Amiens. | 4 |
| 1.2 | Statues avec manque de recul | 4 |
| 1.3 | Système Stéréo Catadioptrique Planaire (SCP) composé d'une caméra et deux miroirs plans. | 6 |
| 2.1 | Schéma d'une seule réflexion [Panahandeh et Jansson (2011)] | 11 |
| 2.2 | Système catadioptrique avec une seule caméra et deux miroirs plans [Gluckman et Nayar (2001)] | 12 |
| 2.3 | Exemple de système SCP. | 14 |
| 2.4 | Exemples de système actif de vision catadioptrique planaire avec un projecteur de lumière. | 15 |
| 2.5 | Un système catadioptrique planaire de lumière structurée. | 15 |
| 2.6 | a) Image obtenue par le SCP. b) Cinq silhouettes de l'objet extraites de l'image. c) Coordonnées de cinq silhouettes dans l'image a. d) les droites bitangentes de cinq vues. | 16 |
| 2.7 | Une image prise par le système catadioptrique planaire. Il y a 5 vues de l'objet dans une seule image. Un point 3D et ses réflexions se situent sur un cercle. | 17 |
| 2.8 | Quelques résultats de reconstruction 3D utilisant les résultats d'étalonnage des points circulaires. | 18 |
| 3.1 | Point de vue unique pour un miroir planaire. | 22 |
| 3.2 | Point et caméra virtuelle. | 23 |
| 3.3 | Matrice de réflexions. | 23 |
| 3.4 | Système Stéréo Catadioptrique planaire composé d'une caméra et deux miroirs plans. | 24 |
| 3.5 | Image de réflexion multiples catadioptrique | 25 |
| 3.6 | Différentes transformations entre les repères | 25 |
| 3.7 | Modèle de la caméra pinhole, d'après Hartley & Zisserman Hartley et Zisserman (2003) | 26 |
| 3.8 | (a) Exemple de détection des coins sur la mire associée aux miroirs dans l'image. (b) Système équivalent, décrivant les emplacements des cinq caméras. La caméra réelle est en face des deux miroirs plans. Les quatre caméras virtuelles sont derrière les miroirs et sont situées à des emplacements symétriques à la caméra réelle par simple et double réflexion par rapport aux plans des deux miroirs. | 28 |

| | | |
|------|---|----|
| 3.9 | Images de sous-bandes d'une mire disposée sur le contour de la surface de chaque miroir planaire, projetées avec les points saillants détectés. | 31 |
| 3.10 | Projection de la grille détectée par rapport aux miroirs plans. Les coins des bandes de la mire sont colorés par des cercles en rouges. | 32 |
| 3.11 | Erreur de calibrage pour chaque image. | 32 |
| 3.12 | Ensemble des images choisies pour la procédure de calibrage. Utilisation de trois plans faisant une rotation suivant un seul axe. | 33 |
| 3.13 | Projection des points 3D de la mire gauche et droite au plan image pour une image de résolution 1280×1024 sans antialiasing. | 33 |
| 3.14 | Comparaison de la robustesse au bruit sur les positions des plans miroirs, positions des points 2D sélectionnées sur la mire. | 35 |
| 3.15 | Cette figure montre les points rouges extraits par le détecteur de Harris et les points verts sont les points calculés après l'estimation des paramètres de calibrage. Erreur moyenne quadratique de reprojection est : 1.09 pixels . . . | 35 |
| 3.16 | L'ensemble des plans | 36 |
| 3.17 | La géométrie de deux vue par un miroir Plan | 37 |
| 3.18 | La matrice fondamentale et l'auto-epipole entre chaque vue réelle et deux vues virtuelles. | 38 |
| 3.19 | Les droites epipolaires et les epipoles entre chaque vue et sa symétrie dans l'image synthétique. | 42 |
| 4.1 | Image capturée par un système SCP. | 47 |
| 4.2 | AMIFT pipeline | 48 |
| 4.3 | Mise en correspondance dans le cas de la réflexion par le miroir. | 49 |
| 4.4 | Différentes réflexions de miroir et le concept de MIFT. Orientation dominante dans l'image originale (a). (b) réflexion horizontale. (c) réflexion verticale. (d) réflexion combinée de (b) et (c). | 49 |
| 4.5 | Illustration de MIFT dans les situations avec et sans réflexion. (a) un point d'intérêt et sa région d'intérêt dans la vue directe. (b) La version inversée de (a).(c)-(d) Répartition des huit orientations dans la 6ème cellule de (a) et (b) | 50 |
| 4.6 | Différentes réflexions de miroir et le concept de MIFT. Un point d'intérêt (PI) est identifié dans l'image originale (a). (b) la réflexion combinée de (a). (c) L'orientation du PI dans l'image originale. (d) l'orientation du PI dans l'image reflétée. (e) l'orientation du PI en utilisant SIFT. (f) L'orientation du PI dans l'image miroir en utilisant SIFT. (g) L'orientation du PI dans l'image miroir en utilisant MIFT. | 50 |
| 4.7 | Exemple d'illustration d'une orientation dominante, en pointant à gauche, où $N_{bin} = 36$ et $n_d = 0$, $j=3$. Les flèches pointants vers la gauche sont les orientations dirigées vers la gauche(flèches bleues) et celles dirigées vers la droite(flèches rouges) par rapport à l'orientation dominante(flèche verte n_d). Les flèches pointées à gauche et pointées à droite signifient m_l et m_r respectivement. | 51 |
| 4.8 | L'interprétation géométrique de l'eq.4.4 | 53 |
| 4.9 | L'échantillonnage des paramètres $\theta = \arccos 1/t$ et ϕ . Les échantillons sont présentés par les points noirs. À gauche : illustration perspective de l'observation d'une hémisphère. À droite, l'observation d'une vue du zénith de l'hémisphère. Les valeurs de θ sont indiquées sur la figure | 54 |
| 4.10 | Vue globale sur la manipulation de l'algorithme AMIFT. | 54 |

| | | |
|------|--|----|
| 4.11 | Exemples d'images de la base de données réelles. | 56 |
| 4.12 | Exemples d'images de la base de données réelles. | 57 |
| 4.13 | Exemples d'images de la base de données réelles des géométries différentes. | 58 |
| 4.14 | Courbes rappel-précision pour tous les descripteurs avec et sans l'utilisation de Réflective-RANSAC | 58 |
| 4.15 | Les correspondances obtenues sur les bases de données synthétique et réelle en utilisant les différentes méthodes. | 59 |
| 4.16 | Quatre résultats d'appariement différents d'une image RMC avec l'utilisation des méthodes AMIFT, ASIFT, MIFT et SIFT | 59 |
| | | |
| 5.1 | Illustration de la mise en correspondance entre les clusters issus de réflexions simple et double. | 62 |
| 5.2 | Homographie induite par la surface plane du cube réel. | 63 |
| 5.3 | Une homographie entre le patch réel et virtuel par un miroir plan. | 64 |
| 5.4 | Extraction des points d'intérêts | 67 |
| 5.5 | Représentation graphique de l'ensemble de patches à partir des homographies potentielles. | 68 |
| 5.6 | Après la transformation, un pixel ne se trouve pas exactement sur une position entière | 69 |
| 5.7 | Espace de recherche des homographies | 71 |
| 5.8 | Résultat de la méthode Branch-and-Bound présenté sur une paire de vues. Les deux patches sont similaires. | 72 |
| 5.9 | Variation de la distance en fonction de l'erreur SAD en utilisant BnB. | 75 |
| 5.10 | L'affichage du patch réel et le patch virtuel transformé. | 76 |
| 5.11 | Variation de l'élévation en fonction de l'erreur SAD en utilisant BnB. | 77 |
| 5.12 | L'affichage du patch réel et le patch virtuel transformé. | 77 |
| 5.13 | Variation de l'azimuth en fonction de l'erreur SAD en utilisant BnB. | 78 |
| 5.14 | L'affichage du patch réel et le patch virtuel transformé. | 79 |
| 5.15 | Variation de l'azimuth et l'élévation en fonction de l'erreur SAD en utilisant BnB. | 80 |
| 5.16 | L'affichage du patch réel et le patch virtuel transformé. | 80 |
| 5.17 | Une représentation graphique montre la fréquence de chaque pixel utilisé pour le calcul d'erreur SAD en fonction des paramètres à estimer. | 81 |
| 5.18 | L'affichage du patch réel et le patch virtuel transformé. | 81 |
| 5.19 | Exemple d'un ensemble de patches mis en correspondance par les homographies locales estimées. | 82 |
| 5.20 | Calcul de l'erreur de recouvrement de différentes images pour 2 tailles de patches. | 83 |
| 5.21 | Résultat de mise en correspondance des patches dans une paire de vue (V2 et V21). | 84 |
| | | |
| 6.1 | Pipeline de la reconstruction 3D. | 85 |
| 6.2 | Géométrie euclidienne, affine et projective | 86 |
| 6.3 | Triangulation : les droites épipolaires ne se coupent pas forcément | 87 |
| 6.4 | Différentes étapes de la reconstruction 3D dense d'un plan | 88 |
| 6.5 | Représentation de nuage de points 3D de la vérité de terrain de deux façons différentes. | 89 |

| | | |
|------|--|----|
| 6.6 | Représentation de nuage de points 3D de notre méthode de deux façons différentes. | 89 |
| 6.7 | Reconstruction 3D du cube. | 89 |
| 6.8 | Analyse des erreurs de reconstruction : densité de probabilité des erreurs de reconstruction des points 3D par notre méthode par rapport à la vérité terrain. | 90 |
| 6.9 | Mise en correspondance entre les points de deux vues réelle/virtuelle par l'homographie induite par le plan | 90 |
| 6.10 | Reconstruction de la sphere | 91 |
| 6.11 | Reconstruction éparse de la théière pour des images 512×512 | 91 |
| 6.12 | Nuage de points 3D du plan reconstruit par notre approche et celui obtenu par le simulateur : Le premier plan à gauche est reconstruit par la vérité terrain, le second est reconstruit par l'approche proposée. | 92 |
| 6.13 | Représentation des deux plans (reconstruits et vérité) dans le même repère. | 92 |
| 6.14 | Exemple d'une image catadioptrique d'un seul objet plan | 93 |
| 6.15 | Exemple d'une image binaire | 94 |
| 6.16 | L'ensemble de contour de la silhouette obtenus par les points de tangence épipolaire | 94 |
| 6.17 | Un exemple de maillage reconstruit à partir de deux images catadioptriques prises de différents points de vue | 95 |
| 6.18 | L'affichage des différents nuage de points 3D sur le même repère. | 95 |



LISTE DES TABLEAUX

| | | |
|-----|--|----|
| 3.1 | Résultats de calibration de la caméra perspective | 32 |
| 3.2 | Les résultats quantitatifs des paramètres de calibrage. | 34 |
| 3.3 | Le calcul des quatre matrices fondamentales et quatre épipoles entre les quatre paires de vue obtenues par les 3 méthodes. | 41 |
| 3.4 | Comparaison de des auto-épipoles obtenus par les données réelles avec les auto-épipoles calculés par les 3 méthodes. | 42 |
| 4.1 | Synthèse des détecteurs classiques | 47 |
| 5.1 | Quatre mesures usuelles de corrélation. | 69 |
| 5.2 | Influence du paramètre δ | 73 |
| 5.3 | Influence de la taille du Patch | 74 |
| 5.4 | L'exécution de la méthode Branch-and-Bound avec un pas de 0.25 et 8 itérations pour converger vers la solution optimale $\text{dist}=0.8010$ | 76 |
| 5.5 | L'exécution de la méthode BnB par un pas de 0.25 avec 6 itérations pour converger vers la solution optimale $\phi = 1.57$ | 78 |
| 5.6 | L'exécution de la méthode Branch-and-Bound par un pas de 0.25 avec 6 itérations pour converger vers la solution optimale $\theta = -0.67$ | 79 |
| 5.7 | L'exécution de la méthode BnB par un pas de 0.25 avec 6 itérations pour converger vers la solution optimale $\text{nr}=[0,0.7809,-0.6247]$ | 79 |
| 5.8 | L'exécution de la méthode Branch-and-Bound par un pas de 0.25 avec 8 itérations pour converger vers la solution optimale $\text{nr}=[0,0.7809,-0.6247]$ et $\text{dist} = 0.8$ | 82 |
| 6.1 | Evolution de la précision en fonction de la résolution des images. | 93 |



LISTE DES ALGORITHMES

| | | |
|-----|--|----|
| 4.1 | ASIFT | 52 |
| 4.2 | AMIFT | 55 |
| 5.1 | Trouver la meilleure homographie par la méthode d'optimisation BnB | 72 |



AVANT-PROPOS

Les travaux présentés dans ce mémoire ont été effectués au Laboratoire de Recherche en Informatique et Télécommunications (LRIT), à la Faculté des Sciences de Rabat (FSR) sous la direction (jusqu'au Mars 2017) du Professeur Driss ABOUTAJDINE et le co-encadrement d'Amina Radgui et au laboratoire Modélisation, Information et Système (MIS) de l'université de Picardie Jules Verne (UPJV) sous la direction du Professeur El Mustapha MOUADDIB et le co-encadrement de Guillaume Caron dans le cadre d'une thèse en cotutelle pour l'obtention du grade de docteur de l'université de Picardie Jules Verne.

Au cours de cette thèse, j'ai bénéficié d'une bourse d'excellence octroyée par le centre nationale pour la recherche scientifique et technique (CNRST) dans le cadre du programme des bourses de recherche initié par le ministère de l'éducation national de l'enseignement supérieur, de la recherche scientifique et de la formation des cadres. Pour la cotutelle, j'ai bénéficié d'une bourse dans le cadre du programme de coopération franco-marocain PHC toubkal, projet ATHAR-3D

Je tiens à remercier Mme Véronique Cherfaoui, Professeur à l'Université de Technologie de Compiègne, d'avoir accepté de présider le jury de ma thèse. Mes remerciements vont également à M. Marc Pierrot-Desseilligny, directeur de recherche ENSG-IGN d'avoir accepté de rapporter sur mon travail. Je tiens également à remercier Mme Samia Ainouz, professeur à l'INSA de Rouen, d'avoir accepté de rapporter sur ce travail. Je tiens aussi à remercier M. Claude PEGARD Professeur d'enseignement supérieur à l'université de Picardie Jules Verne d'avoir accepté d'examiner ce travail.

Je veux également exprimer toute ma reconnaissance à El Mustapha MOUADDIB, professeur à l'université de Picardie Jules Verne, mon directeur de thèse. Je le remercie de son encadrement scientifique, sa patience et ses conseils multiples. Je lui exprime aussi ma gratitude pour ses relectures lors de la rédaction du manuscrit.

Mes remerciements vont aussi à Guillaume Caron, maître de conférences à l'université de Picardie Jules Verne de m'avoir fait profiter de ses compétences, sa qualité scientifique et surtout son enthousiasme permanent qui va rester un exemple à suivre pour moi. Enfin, je remercie Amina Radgui pour sa disponibilité. Je lui suis également reconnaissant du soutien et toute l'aide qu'elle m'a apportée en aspects scientifiques. Sa relecture très méticuleuse de chacun des chapitres qui m'a permis d'améliorer la qualité de mon manuscrit. Je remercie mes amis doctorants et postdocs du laboratoire LRIT et MIS, et plus particulièrement Oussama ZEGLAZI, Hassan RAMI, Mohammed HAMIDI, Farouk ACHAKIR, Richardson CIGUENE et Tawsif GOKHOOOL et les autres aussi pour tous les moments qu'on a passés ensemble et pour la bonne ambiance et surtout le partage scientifique durant cette période de la préparation de la thèse.

Je termine par les personnes que je ne saurais jamais remercier assez comme elles méritent. Merci à mes parents, mes frères ainsi que toute ma famille pour leur soutien moral dans les périodes difficiles que j'ai pu avoir durant la préparation de la thèse.

INTRODUCTION GÉNÉRALE

Bien que depuis plusieurs années, de nombreuses techniques ont été développées et appliquées avec succès dans l'industrie, la reconstruction 3D reste encore un problème en vision par ordinateur ouvert et présente plusieurs défis intéressants à résoudre. Ce travail de recherche est une contribution qui s'inscrit dans ce contexte. Il est le fruit d'une collaboration entre le laboratoire de recherche en Informatique et Télécommunications (LRIT) de l'université Mohammed V-Rabat et le laboratoire Modélisation, Information et Systèmes (MIS) de l'université de Picardie Jules Verne à Amiens. Cette thèse s'est déroulée dans le cadre du projet ATHAR-3D (PHC Toubkal 2015/2017). L'objectif est d'obtenir, à l'aide d'un système catadioptrique (caméra + miroirs), une représentation 3D d'objet de forme quelconque fidèle à la réalité. Plusieurs applications de la reconstruction 3D trouveront un intérêt dans l'utilisation de ce type de système, notamment le cas de la numérisation de la cathédrale d'Amiens qui abrite une variété importante de formes architecturales sur une très grande surface. Cette numérisation a pour objectif d'ouvrir une fenêtre dynamique sur le passé via la création d'une maquette 3D qui devrait être attractive pour le grand public, mais aussi très utile pour les spécialistes qui souhaitent valider les hypothèses de construction, étudier l'édifice et planifier des restaurations.

1.1 Contexte

Au cours des neuf dernières années, le laboratoire MIS s'est investi dans le programme E-Cathédrale, porté par le Professeur El Mustapha MOUADDIB. Ce programme vise la conservation, l'amélioration de l'accessibilité, la réalisation et l'exploitation d'une maquette numérique d'un patrimoine monumental. Cette dernière se veut être très précise, réaliste et complète, aussi bien de l'intérieur que de l'extérieur, pour laisser une trace complète du monument par un travail rigoureux et scientifique. Cette maquette sera par la suite destinée à des visites virtuelles et des études scientifiques (histoire de l'art, architecture, ingénierie en calcul de structures, etc.). La maquette numérique 3D peut s'avérer être un support efficace pour documenter l'état actuel d'un édifice historique mais aussi pour créer des ressources pour les chercheurs qui analysent son évolution historique. La numérisation s'effectue par plusieurs moyens techniques : par lasergrammétrie grâce à un scanner laser qui prélève des millions de points 3D, et par photogrammétrie qui consiste à acquérir plusieurs photos pour construire le modèle 3D. Jusqu'à présent, une grande partie de la numérisation de la cathédrale a été faite (Fig.1.1). Ces vues peuvent être très détaillées, au niveau d'une arche par exemple, afin de présenter les forces qui s'exercent sur les piliers et de reconstituer les plans anciens, erronés ou incomplets. Pour contribuer à ce programme E-cathédrale, cette thèse donne la possibilité de voir des endroits inaccessibles qui ne seraient jamais visibles au cours d'une visite réelle. Afin de répondre au problème de manque de recul des statues présentés sur les murs de la cathédrale, un système à

une seule caméra et plusieurs miroirs plans permettra de rendre visible les faces cachées inaccessibles directement par une caméra.

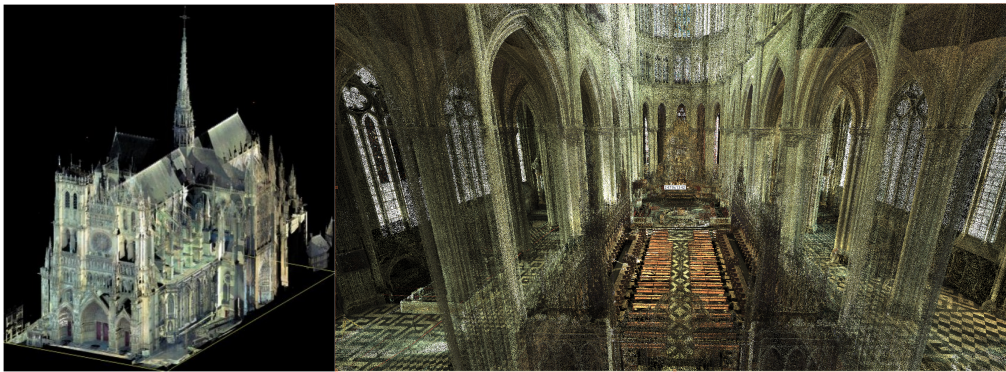


Figure 1.1 – Quelques résultats du projet E-cathédrale du nuage de points de l'intérieur et de l'extérieur de la cathédrale d'Amiens.

1.2 Problématique de la thèse

Cette thèse présente une nouvelle méthodologie complète de la reconstruction 3D d'objets sans recul (Fig.1.2) en faisant apparaître les zones occultées afin de reconstruire complètement l'objet. D'autre part, elle aborde le problème sous l'angle de la vision catadiop-



Figure 1.2 – Statues avec manque de recul

trique en exploitant les miroirs plans. L'objectif est de permettre la reconstruction d'un

objet 3D à partir d'une seule image qui contient toute l'information possible grâce à l'association des miroirs à la caméra, et d'utiliser un nombre minimal d'acquisitions. L'utilisation de miroirs offre donc une alternative potentielle par la présence de réflexions multiples dans cet environnement où l'objet à reconstruire ne peut pas être déplacé et la caméra doit rester dans une zone limitée de l'espace. Ces miroirs permettent de calibrer la caméra et, comme nous le verrons par la suite, il est possible d'estimer avec précision l'équation d'un miroir plan ainsi que la transformation rigide associée (matrice de réflexion). L'obtention de cette transformation est nécessaire pour déterminer la transformation rigide entre la caméra réelle et les caméras virtuelles : celles des images de réflexions sur les miroirs.

Le système stéréo catadioptrique planaire (SCP) que nous allons étudier est constitué d'une seule caméra perspective et deux miroirs plans. Il fournit une image contenant un objet réel et quatre objets "virtuels" obtenus par réflexion. L'image acquise est appelée l'image de Réflexion Multiple Catadioptrique (RMC) comme illustré dans la figure 1.3. L'image RMC comporte cinq vues du même objet.

Une attention particulière a été récemment accordée à ce type de système. En effet, à l'opposé des systèmes stéréos traditionnels à deux ou plusieurs caméras, les systèmes stéréo catadioptriques planaires permettent de capturer plusieurs vues de la même scène dans une seule image, sans besoin de synchroniser des caméras. Le fait d'employer ces systèmes permettra de compléter le système de vision en associant à la caméra réelle quatre caméras virtuelles, qu'il faut par la suite modéliser et étalonner. Notre intérêt a été focalisé sur la position 3D exacte de chaque point de l'objet et non sur l'enveloppe visuel l'englobant. En effet, l'obtention d'une forme précise des objets à reconstruire en 3D à partir de la silhouette reste toujours un défi et dépend de la scène étudiée et de l'exactitude de l'extraction des contours. La précision de l'estimation des paramètres des caméras dépend aussi de l'extraction des silhouettes. Néanmoins, cette technique de reconstruction 3D à partir des silhouettes appliquée sur un objet et ses inter-réflexions multiples devient difficile en présence de régions texturées. Cet aspect reste encore difficile car il n'existe pas de méthodes capables d'extraire la silhouette avec précision de tout type d'objets. Nous nous orientons vers la reconstruction 3D dense basée points homologues. Le calcul des points homologues est une première étape primordiale dans toutes les approches de reconstruction d'objets 3D. L'algorithme SIFT est le plus utilisé en pratique, mais il est limité quand la réflexion d'un miroir intervient et la géométrie entre deux vues varie trop fortement. Cette thèse propose une extension de SIFT qui combine des invariants à l'échelle, à la rotation, aux transformations affines et à la réflexion. Dans un deuxième temps, afin d'obtenir une reconstruction dense, une nouvelle approche de mise en correspondance dense par calcul d'homographies induites par les plans locaux observés entre l'image de l'objet et une de ses réflexions par les miroirs est proposée. L'estimation des homographies se fait par la minimisation de la somme des différences absolues (SAD) entre le voisinage de chaque pixel et son transformé.

La reconstruction 3D que nous proposons se base sur les algorithmes de la vision passive dense issus des différentes techniques d'appariements. L'originalité de cette thèse est de proposer une approche de reconstruction 3D dense d'objets naturels, sans modèle de forme et sans texture bien définie, dont des parties sont invisibles via la caméra directe. L'une des grandes difficultés dans la construction du modèle 3D d'un objet, reste la mise en correspondance des primitives extraites de l'objet observé de différents points de vues. Ces primitives doivent être particulièrement significatives, structurelles et robustes aux nombreuses transformations de l'image, comme le changement d'échelle, la réflexion et la transformation affine due au miroir. La qualité de la mise en correspondance va dépendre directement de la précision d'extraction des informations caractérisant les primitives. En règle générale, plus les critères de sélection sont sévères, plus le nombre de primitives à ap-

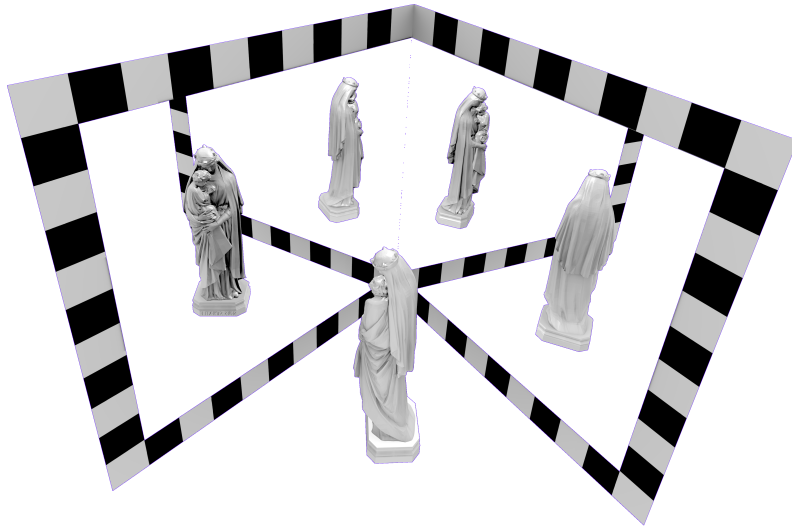


Figure 1.3 – Système Stéréo Catadioptrique Planaire (SCP) composé d’une caméra et deux miroirs plans.

parier sera restreint. Or, une reconstruction métrique dense est nécessaire pour permettre des mesures précises de dimensionnement, ce qui nécessite un très grand nombre d’appariements entre les images. On parle alors d’appariement dense (dense matching). L’idée d’exploiter la photogrammétrie dans un système SCP est de combiner la vue directe et ses quatre inter-réflexions d’une scène ou d’un même objet pour obtenir une information 3D. Pour cela, nous cherchons une méthode précise de mise en correspondance entre chaque paire de vues obtenue par le miroir plan. Dans la pratique, SIFT reste une méthode de référence et la plus connue, mais elle est limitée quand la réflexion de miroir intervient et la géométrie entre deux vues varie trop fortement à cause des déformations projectives importantes. Les méthodes classiques de photogrammétrie restent insuffisantes pour traiter les images de réflexions multiples catadioptrique et c’est sur ce point que nous nous concentrons dans cette thèse. Nous allons procéder en plusieurs étapes que nous détaillerons dans la suite de ce manuscrit. La première est l’appariement épars des points caractéristiques utilisant une nouvelle extension de SIFT invariante à certaines transformations géométriques. La seconde partie est consacrée à l’appariement dense qui est l’étape la plus lourde en terme de temps de calcul. Enfin la dernière partie présente le rendu final, avec la simplification du modèle par un maillage, dont l’aspect réaliste sera renforcé par un plaquage de texture.

1.3 Contributions

Les principales contributions de cette thèse sont :

- 1- La modélisation et étalonnage d'un SCP à deux miroirs sous la forme d'un réseau d'une seule caméra réelle et plusieurs caméras virtuelles.
- 2- La proposition d'une nouvelle variante de SIFT pour la mise en correspondance éparsée dans le cas des transformations de symétries et affines, et qui permet de détecter et d'apparier des points caractéristiques entre un objet et son reflet par un miroir plan. Elle sera rendue plus robuste par l'élimination de faux appariements en exploitant la géométrie épipolaire.
- 3- Une nouvelle méthode de mise en correspondance dense basée homographie : c'est une méthode qui se base sur les homographies induites par un plan local de l'objet observé par une caméra et deux miroirs plans, et qui permet d'estimer le correspondant d'un voisinage dans les vues réfléchies (simple et double) d'une vue directe.
- 4- Un ensemble de résultats de la reconstruction 3D dense d'objet à partir d'un SCP.

1.4 Plan du mémoire

Les travaux sont présentés de la manière suivante : Après ce premier chapitre, le chapitre 2 présente différentes méthodes de reconstruction 3D à base de systèmes de vision utilisant des miroirs et les dernières recherches réalisées dans ce domaine.

Chapitre 3 : Le troisième chapitre décrit la modélisation et l'étalonnage du système utilisé. Nous proposons une méthode d'étalonnage basée sur la méthode développée par Jean-Yves Bouguet. En premier lieu, le modèle géométrique du système caméra-miroirs est présenté. Ensuite, les paramètres intrinsèques sont estimés pour la caméra réelle, puis les paramètres extrinsèques sont identifiés afin de déterminer les positions des prises de vues virtuelles autour de l'objet.

Chapitre 4 : Le quatrième chapitre est consacré à la mise en correspondance des points entre chaque vue et son reflet en tenant compte de toutes les transformations des miroirs. Nous présentons une nouvelle variante de SIFT qui sera appelée AMIFT (Affine-Mirror SIFT).

Chapitre 5 : Le cinquième chapitre a pour objectif de proposer une nouvelle méthode de mise en correspondance dense entre la vue directe et les vues réfléchies. Cette approche se base sur l'estimation d'homographies en minimisant la somme des différences absolues (SAD) des pixels entre le voisinage d'un pixel dans l'image et son transformé.

Chapitre 6 : Le sixième chapitre sera dédié à la reconstruction 3D d'objets. Nous allons valider les différentes approches proposées par divers protocoles expérimentaux sur différents types d'objets afin d'aboutir à une reconstruction 3D dense d'objet par un SCP.

Enfin, le dernier chapitre de cette thèse a pour objectif de résumer les différents points clés abordés, et de proposer quelques perspectives de recherche.

Sommaire

| | | |
|-------|--|----|
| 2.1 | Introduction | 9 |
| 2.2 | Modélisation des systèmes catadioptriques planaires | 10 |
| 2.2.1 | Modélisation d'un système catadioptrique planaire à un miroir | 10 |
| 2.2.2 | Modélisation d'un système catadioptrique planaire à deux miroirs | 12 |
| 2.3 | Techniques de Reconstruction 3D à partir des SCPs | 13 |
| 2.3.1 | Systèmes actifs | 14 |
| 2.3.2 | Systèmes passifs | 16 |
| 2.4 | Conclusion | 18 |

2.1 Introduction

La combinaison de miroirs plans avec une unique caméra permet de compléter une caméra par plusieurs caméras virtuelles. Ce type de système de vision catadioptrique a été initialement définie par Hecht et Zajac [Hecht et Zajac (1974)]. Étymologiquement, on retrouve le terme “dioptré” qui s’intéresse à l’étude de lentilles réfractantes tandis que “catadioptré” s’intéresse à l’étude des miroirs réfléchissantes. L’utilisation des miroirs plans, permet de changer la direction de la vue, d’élargir le champ de vue et d’acquérir plusieurs vues dans le même instant. L’utilisation de miroirs offre une alternative potentielle aux situations où l’objet à reconstruire ne peut pas être déplacé (grande statue ou environnement architectural), et/ou lorsque la caméra doit rester dans une zone limitée de l’espace. Ce chapitre présente un état de l’art sur la géométrie et les méthodes de reconstruction 3D des systèmes de vision qui combinent l’utilisation des caméras avec plusieurs miroirs plans. Ces systèmes, appelés systèmes catadioptriques, peuvent être classifiés selon la combinaison d’un ou de plusieurs miroirs face à une ou plusieurs caméras conventionnelles. Il existe un grand nombre de systèmes catadioptriques connus qui utilisent divers types de combinaisons de miroirs plans, convexes ou concaves [Radu *et al.* (2006)].

Dans cette thèse, seule la configuration basée sur une caméra avec deux miroirs plans est étudiée pour l’obtention d’un système de reconstruction 3D d’objets. Ce système est destiné aux environnements confinés difficilement accessibles aux caméras ; ces miroirs rendant visibles les parties cachées. Les miroirs plans sont très simples et faciles à fabriquer, et c’est la raison pour laquelle les systèmes catadioptriques planaires jouent un rôle dans de nombreuses applications de vision par ordinateur : la robotique et l’assemblage [Mariottini *et al.* (2009)], la surveillance visuelle [Kumar *et al.* (2008)], [Tan *et al.* (2004)] et la modélisation 3D [Gluckman et Nayar (2002)], [Mitsumoto *et al.* (1992)]. Ces systèmes catadioptriques présentent plusieurs avantages puisque cette configuration permet d’obtenir

plusieurs points de vue d'un même objet avec une unique caméra, ce qui permet de s'affranchir des problèmes de synchronisation et des différences entre les réponses spectrales des caméras [Gluckman et Nayar (2001)]. Mais, ces systèmes ont aussi des inconvénients parmi lesquels on retrouve le problème de la réduction de la résolution et sa variation. D'autres contraintes comme le positionnement des miroirs sont aussi importantes parce qu'elles ont des conséquences sur le nombre d'images pouvant être obtenues. En effet l'agencement des miroirs influe sur la convergence des caméras virtuelles et la distance entre ces dernières et l'objet analysé. Cependant, l'obtention d'une forme précise des objets à reconstruire en 3D reste un défi et dépend de la scène étudiée.

Les méthodes de reconstruction 3D existantes reposent soit sur l'extraction des silhouettes des objets [Forbes *et al.* (2006)] soit sur l'exploitation de la présence d'une surface réfléchissante via le miroir permettant de faire de la vision stéréoscopique [Garbacz et Giesko (2010)]. Cela revient à traiter un problème de stéréovision binoculaire classique. Nous allons nous concentrer dans ce chapitre sur la description et la modélisation du modèle géométrique du système de vision catadioptrique planaire et sur les méthodes de reconstruction 3D proposées.

2.2 Modélisation des systèmes catadioptriques planaires

Le capteur catadioptrique planaire est obtenu en plaçant une caméra perspective face à un miroir plan de façon à refléter la scène et à faire apparaître des informations manquantes. Cette configuration peut-être employée même quand l'axe optique de la caméra est perpendiculaire au miroir. Ce choix permet de rendre visibles des parties de l'objet, côté miroir.

Un champ visuel plus large peut être obtenu en plaçant soigneusement plusieurs miroirs plans face à une caméra perspective. Selon le type de caméra et le nombre de miroirs plans, on classe les systèmes catadioptriques en systèmes catadioptriques à point de vue unique qui permettent d'élargir le champ de vision [Fujiyama *et al.* (2010)], [Wu (2007)], [Zhong *et al.* (2006)] et systèmes catadioptriques à points de vue multiples qui permettent d'ajouter de nouveaux points de vue virtuels et augmenter le nombre de redondants [Forbes *et al.* (2006)], [Huang et Lai (2006)], [Lanman *et al.* (2009)].

2.2.1 Modélisation d'un système catadioptrique planaire à un miroir

Ce type de système est particulièrement utilisé pour réaliser une tête stéréo à l'aide d'une seule caméra et un miroir plan. Dans Rodrigues *et al.* (2010), les auteurs placent un objet entre la caméra et le miroir plan. L'image d'un objet obtenue ainsi par réflexion sur un miroir plan est géométriquement équivalente à l'image de cet objet obtenue par une caméra virtuelle située derrière le miroir (Fig.2.1). D'autres travaux [Hu (2009)], [Hayashi *et al.* (2013)], [Long *et al.* (2015)], [Marchand et Chaumette (2017)] ont focalisé leur intérêt autour de la même idée afin de déterminer la pose du plan miroir par rapport à la caméra réelle.

Nous décrivons explicitement dans cette section la modélisation d'un système catadioptrique planaire à un miroir. La projection s'effectue en deux étapes. La première étape consiste à projeter un point de l'espace sur le plan image de la caméra virtuelle. La seconde étape consiste à transformer l'image virtuelle en une image réelle en utilisant la transformation de réflexion.

Rappelons que \mathbf{X}_R est un point 3D réel représenté dans le repère de la caméra réelle.

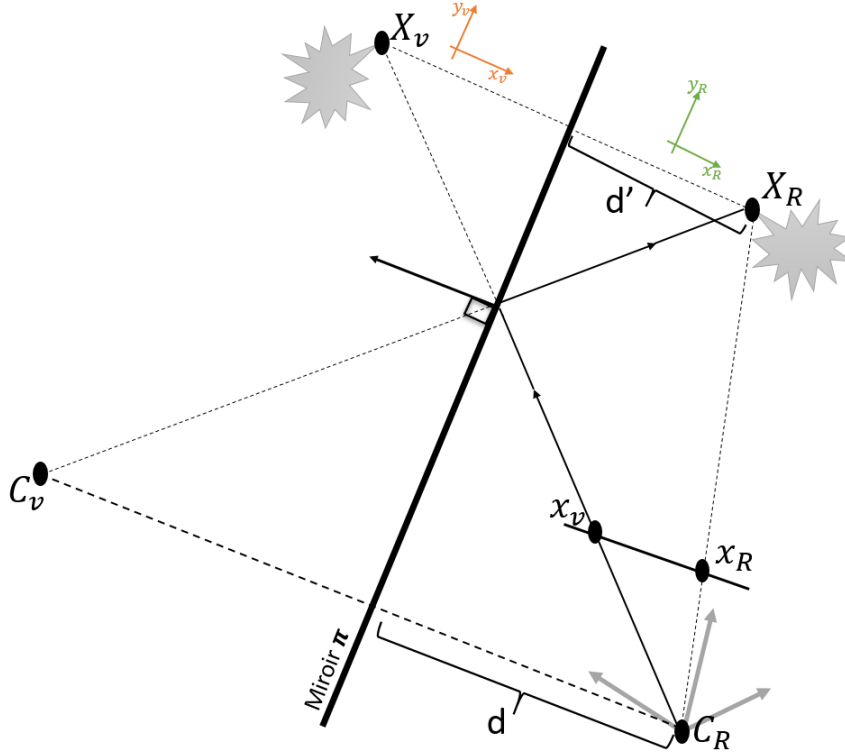


Figure 2.1 – Schéma d'une seule réflexion [Panahandeh et Jansson (2011)]

\mathbf{X}_V est un point 3D virtuel obtenu par réflexion par rapport à un miroir à partir de \mathbf{X}_R . l'image d'une paire de points symétriques ($\mathbf{X}_R, \mathbf{X}_V$) vue par une caméra réelle est identique à l'image du même paire de vue par la caméra réelle.

Soit Π le plan du miroir. Son équation donnée dans le repère direct de la caméra est :

$$\mathbf{X}_R \cdot \mathbf{n} + d = 0 \quad (2.1)$$

\mathbf{n} et d correspondent respectivement à la normale ($n \in \mathbb{R}^3$ et $\|n\| = 1$) et à la distance du miroir exprimées dans \mathcal{R} . Nous définissons une caméra virtuelle C_V comme la réflexion de la caméra réelle C_R par rapport au miroir (Fig.2.1). Elle est modélisée par une transformation de réflexion composée d'une rotation et d'une translation, décrite de la manière suivante :

$$C_V = \mathbf{R}C_R + \mathbf{t} = \mathbf{R}C_R + 2d\mathbf{n} \quad (2.2)$$

où $\mathbf{t} = 2d\mathbf{n}$ est le vecteur de translation entre la caméra réelle et la caméra virtuelle.

\mathbf{R} est la rotation entre la caméra réelle et la caméra virtuelle. Par la suite, nous allons définir les expressions de \mathbf{R} et \mathbf{t} en fonction des paramètres de la normale. Pour cela, nous considérons un point 3D \mathbf{X}_R projeté par la caméra directe, sa relation avec son reflet dans le miroir est donnée par l'équation :

$$\mathbf{X}_R = \mathbf{X}_V - 2d'\mathbf{n} \quad (2.3)$$

où $\mathbf{t}' = 2d'\mathbf{n}$ est le vecteur de translation du point réel vers le miroir plan. Aussi la projection de \mathbf{X}_V dans la caméra directe est donnée par :

$$d' + d = \mathbf{n}^T \mathbf{X}_V \quad (2.4)$$

à partir de 2.3 et 2.4 on obtient :

$$\mathbf{X}_R = \mathbf{X}_V - 2(-d + \mathbf{n}^T \mathbf{X}_V) \mathbf{n} \quad (2.5)$$

$$\mathbf{X}_R = \mathbf{X}_V + 2d\mathbf{n} - 2\mathbf{n}\mathbf{n}^T \mathbf{X}_V \quad (2.6)$$

$$\mathbf{X}_R = (\mathbf{I} - 2\mathbf{n}\mathbf{n}^T) \mathbf{X}_V + 2d\mathbf{n} \quad (2.7)$$

ou encore sous forme matricielle :

$$\mathbf{X}_R = \begin{bmatrix} \mathbf{H} & 2d\mathbf{n} \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \mathbf{X}_V = \mathbf{D} \mathbf{X}_V \quad (2.8)$$

où $\mathbf{H} = \mathbf{I}_{3 \times 3} - 2\mathbf{n}\mathbf{n}^T$ est une matrice de réflexion de Householder 3×3 . La matrice \mathbf{H} est orthogonale symétrique et \mathbf{D} exprime la transformation de réflexion. Donc la pose relative entre la caméra réelle et la caméra virtuelle, est exprimée comme suit :

$$\mathbf{D} = \begin{bmatrix} \mathbf{I} - 2\mathbf{n}\mathbf{n}^T & 2d\mathbf{n} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \quad (2.9)$$

La projection de \mathbf{X}_R dans l'image 2D en coordonnées homogènes est un point \mathbf{x} donné par :

$$\lambda \mathbf{x} = \mathbf{K} [\mathbf{I} | 0] \mathbf{X}_R \quad (2.10)$$

La matrice de réflexion est identique à son inverse, c'est-à-dire que $\mathbf{D}\mathbf{D} = \mathbf{I}$ ou $\mathbf{D} = \mathbf{D}^{-1}$ [Mariottini *et al.* (2012)].

Finalement, un point 3D virtuel \mathbf{X}_V peut être défini dans le plan image de la caméra réelle par la relation suivante :

$$\lambda \mathbf{x} = \mathbf{K} [\mathbf{I} | 0] \mathbf{D} \mathbf{X}_V \quad (2.11)$$

2.2.2 Modélisation d'un système catadioptrique planaire à deux miroirs

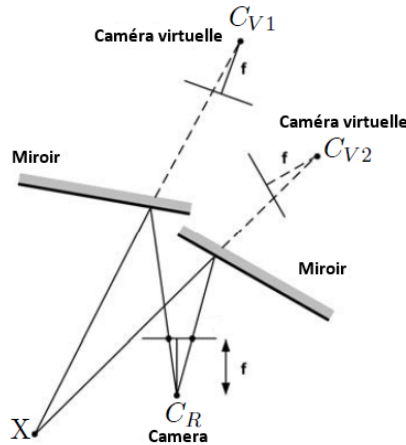


Figure 2.2 – Système catadioptrique avec une seule caméra et deux miroirs planaires [Gluckman et Nayar (2001)]

Un objet situé face à deux miroirs plans faisant entre eux un angle α donne naissance

à deux images virtuelles de l'objet. La première est reflétée par le premier miroir et la deuxième est reflétée par le second miroir. Du point de vue mathématique, la composition des symétries par rapport à deux plans sécants est une rotation autour de la droite d'intersection des plans dont l'angle est le double de l'angle α . Pour deux plans qui se coupent à angle droit, la rotation est d'un demi-tour autour de leur intersection. On observe donc quatre images virtuelles de l'objet : deux obtenues par une simple réflexion et les autres sont obtenues par une réflexion de réflexion.

Le système à deux miroirs plans (figure 2.2) revient à simuler à partir d'une seule caméra réelle la présence de deux caméras virtuelles synchronisées qui ont les mêmes paramètres intrinsèques que la caméra réelle, permettant ainsi de faire de la vision stéréoscopique.

Soit :

$$\mathbf{X}_{V1} = \mathbf{D}_1 \mathbf{X}_R \quad \text{et} \quad \mathbf{X}_{V2} = \mathbf{D}_2 \mathbf{X}_R \quad (2.12)$$

où \mathbf{D}_1 et \mathbf{D}_2 sont respectivement les transformations de réflexion par rapport aux miroirs M_1 et M_2 .

On suppose que les surfaces des miroirs sont représentées par les vecteurs normaux \mathbf{n}_1 et \mathbf{n}_2 , et que les distances entre le centre de la caméra réelle et les miroirs sont d_1 et d_2 . D'après l'équation 2.9, les deux transformations de réflexion sont exprimées de la façon suivante :

$$\mathbf{D}_1 = \begin{bmatrix} \mathbf{I} - 2\mathbf{n}_1\mathbf{n}_1^T & 2d_1\mathbf{n}_1 \\ 0 & 1 \end{bmatrix} \quad (2.13)$$

et

$$\mathbf{D}_2 = \begin{bmatrix} \mathbf{I} - 2\mathbf{n}_2\mathbf{n}_2^T & 2d_2\mathbf{n}_2 \\ 0 & 1 \end{bmatrix} \quad (2.14)$$

La combinaison de \mathbf{D}_1 et \mathbf{D}_2 permet d'obtenir la transformation de réflexion \mathbf{D} entre ces deux points de vues virtuels :

$$\mathbf{X}_{V2} = \mathbf{D}_2 \mathbf{X}_R = \mathbf{D}_2 \mathbf{D}_1^{-1} \mathbf{X}_R = \mathbf{D} \mathbf{X}_{V1} \quad (2.15)$$

où

$$\mathbf{D} = \mathbf{D}_2 \mathbf{D}_1^{-1} = \mathbf{D}_2 \mathbf{D}_1 = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \quad (2.16)$$

avec :

$$\mathbf{R} = \mathbf{I} + 4(\mathbf{n}_1 \cdot \mathbf{n}_2)\mathbf{n}_1\mathbf{n}_2^T - 2\mathbf{n}_1\mathbf{n}_1^T - 2\mathbf{n}_2\mathbf{n}_2^T \quad (2.17)$$

$$\mathbf{t} = 2d_1\mathbf{n}_1 - (2d_1(\mathbf{n}_1 \cdot \mathbf{n}_2) + 2d_2)\mathbf{n}_2 \quad (2.18)$$

Nous venons de montrer qu'on peut trouver la relation entre la caméra réelle et les caméras virtuelles, et estimer les poses des caméras virtuelles par rapport aux miroirs plans. Les publications liées à ce problème concernent le cas d'une simple réflexion. Nous allons étendre ces travaux pour qu'ils soient applicables dans notre contexte lorsqu'on rajoute une double réflexion.

2.3 Techniques de Reconstruction 3D à partir des SCPS

Plusieurs méthodes basées sur le système catadioptrique planaire ont été utilisées pour la reconstruction 3D par stéréoscopie [Gluckman et Nayar (2001)], [Goshtasby et Gruver

(1993)], [Nene et Nayar (1998)], [Mariottini *et al.* (2012)], [Fujiyama *et al.* (2010)], [Xu *et al.* (2015)], à partir de la silhouette [Forbes *et al.* (2006)], [Huang et Lai (2006)], [Reshetouski *et al.* (2011)], la lumière structurée [Lanman *et al.* (2009), Tahara *et al.* (2015)], la structure à partir du mouvement circulaire [Zhang *et al.* (2012)] et aussi pour l'analyse de réflectance [Ihrke *et al.* (2012), Inoshita *et al.* (2013), Tagawa *et al.* (2012)] et l'imagerie plénoptique [Fuchs *et al.* (2013), Levoy *et al.* (2004), Sen *et al.* (2005)]. L'objectif de l'étape de reconstruction est de construire un modèle 3D de l'objet à partir de cinq vues issues d'une caméra réelle et quatre caméras virtuelles. Nous considérons pour commencer deux types de systèmes : les systèmes passifs et les systèmes actifs.

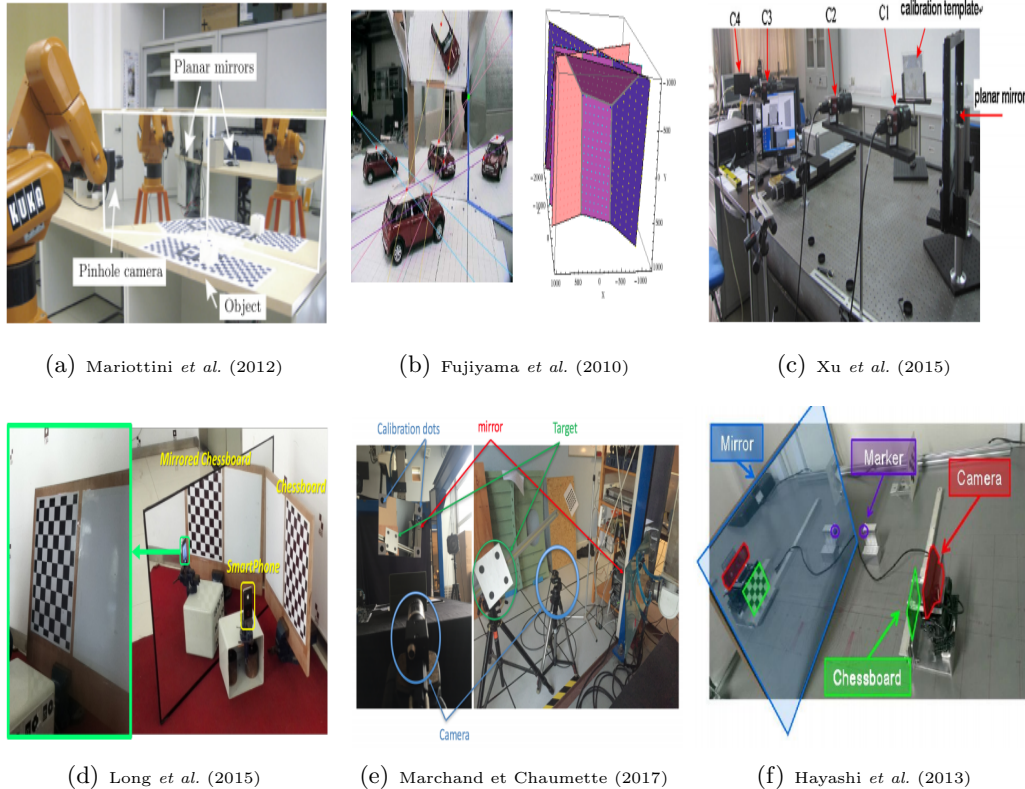


Figure 2.3 – Exemple de système SCP.

2.3.1 Systèmes actifs

La reconstruction par lumière structurée est une technique basée à la fois sur un projecteur et une caméra. Les systèmes actifs de vision catadioptrique planaire exploitent un projecteur de lumière structurée [Tahara *et al.* (2015)], [Nobuhara *et al.* (2016)], [Lanman *et al.* (2007)], [Nguyen *et al.* (2018)], [Kim *et al.* (2014)]. Pour obtenir une reconstruction 3D dense de la scène, ces techniques nécessitent un système de balayage. La distance entre le capteur et le point de la scène éclairé peut être évaluée par triangulation. Ceci se fait par projection d'une lumière sur la surface de l'objet et en observant cette surface avec une caméra.

L'utilisation d'un projecteur dans un système de reconstruction peut modifier l'illumination de la scène. Il faut donc, au cours de la reconstruction, pouvoir différencier une surface illuminée par le projecteur d'une surface qui ne l'est pas. [Lanman *et al.* (2007)] a utilisé

la séquence de code binaire comme illustré dans la figure 2.4, dans laquelle chaque motif est composé d'une séquence de bandes noires et blanches orientées le long des lignes de balayage du projecteur. En éclairant l'objet avec une séquence temporelle multiplexée de motifs de code gris, la ligne de balayage correspondante du projecteur peut être identifiée pour chaque pixel de la caméra. Ensuite, l'intersection des lignes de vue peut être utilisée pour reconstruire un point 3D sur la surface.

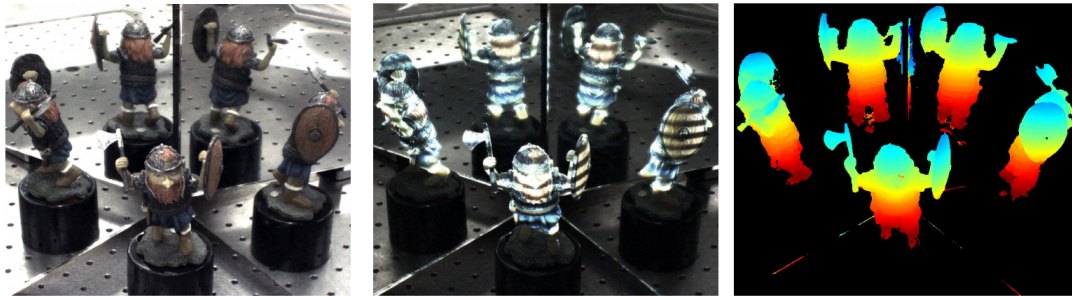


Figure 2.4 – Exemples de système actif de vision catadioptrique planaire avec un projecteur de lumière.

En vision active, l'intégration de miroirs plans à l'intérieur d'un système de lumière struc-

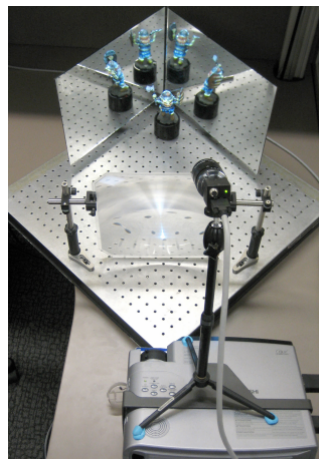


Figure 2.5 – Un système catadioptrique planaire de lumière structurée.

turée permet d'ajouter de nouveaux points de vue virtuels à la fois pour la caméra et pour le projecteur et d'offrir une alternative pour éviter les conflits entre les motifs de lumière projetés et observés sur les miroirs incluant les inter-réflexions multiples. Ces motifs de lumière structurée sont robustes aux réflexions miroirs.

Reconstruire des objets dans un système de lumière structurée exige d'exprimer dans un même repère les différentes composantes du système. Il faut donc procéder à une étape de calibration de la caméra et du projecteur. Le projecteur émet les motifs de lumière structurée sur l'objet à reconstruire et les déformations de ce motif sont capturées par la caméra réelle. Classiquement les motifs ont la forme d'une grille, d'une bande ou d'un ensemble des lignes. La lumière structurée peut comporter aussi un motif laser binaire encodé spatialement ou encore un multiplexage temporel [Salvi *et al.* (2004)]. Lorsque l'encodage des frontières est utilisé comme code de lumière structurée, un problème se pose aux silhouettes de l'objet et certains points erronés (outliers) sont reconstruits. La discontinuité entre un code binaire identifié sur la surface de l'objet et un code binaire

identifié sur l'arrière-plan est considérée comme une frontière entre deux bandes de lumière structurée. Pour prévenir ces erreurs, on peut procéder en détectant les silhouettes de l'objet et en ignorant les codes identifiés à proximité. Malheureusement ce problème peut aussi apparaître sur les silhouettes internes de l'objet.

2.3.2 Systèmes passifs

Les systèmes de reconstruction 3D passifs cherchent à retrouver la structure 3D de la scène sans émettre d'onde ou de lumière spécifique. Les systèmes passifs utilisent des capteurs de vision pour retrouver la forme 3D des objets à partir d'une ou plusieurs images. En vision par ordinateur, ces techniques de reconstruction 3D à partir d'images sont appelées techniques de "Shape-From-X", où X peut être "Stereo", "Motion", "Shading", etc. Dans cette section, nous présentons un rapide survol de l'état de l'art sur les techniques de reconstruction 3D et sur l'utilisation des miroirs en vision catadioptrique planaire pour ces systèmes passifs. Le but de cette partie est de présenter certaines techniques de reconstruction existantes et les principales recherches en rapport avec ce manuscrit.

2.3.2.1 Shape-From-silhouette

Les silhouettes ont été utilisées pour la première fois par [Forbes *et al.* (2006)] pour trouver des approximations de formes de l'objet par l'estimation des droites bi-tangentes en intersectant les cônes engendrés par les silhouettes. Cette technique permet de créer un modèle 3D en se basant sur l'enveloppe visuelle englobant l'ensemble des silhouettes. L'enveloppe visuelle est l'intersection des cônes des silhouettes d'entrée. Le principe est de placer un objet devant un fond connu, d'acquérir une image contenant les vues entourant cet objet et ensuite de traiter l'image afin d'extraire le fond de la scène. Grâce aux différents points de vue, on englobe l'objet dans un volume appelé enveloppe visuelle. Plus on utilise

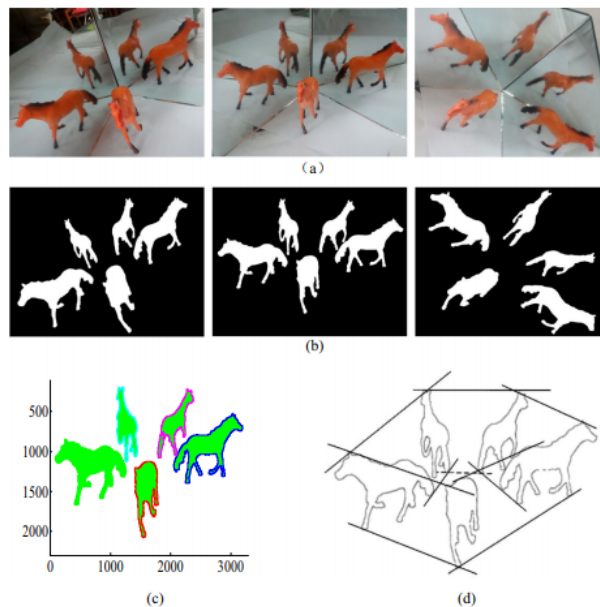


Figure 2.6 – a) Image obtenue par le SCP. b) Cinq silhouettes de l'objet extraites de l'image. c) Coordonnées de cinq silhouettes dans l'image a. d) les droites bitangentes de cinq vues.

de silhouettes, plus l'enveloppe visuelle convergera vers un volume plus serré et sera plus proche de la forme de l'objet réel. De nombreuses études ont proposé diverses méthodes d'obtention de silhouettes d'objets [Ying *et al.* (2013), [Zhao et Li (2017)], [Zhao *et al.* (2016)], [Hu (2009)]. Ces méthodes se focalisent sur la silhouette d'objet pour récupérer les positions et les paramètres intrinsèques de la caméra. De surcroît, dans [Forbes *et al.* (2006)] et [Huang et Lai (2006)] les lignes bi-tangentes aux silhouettes sont extraites d'une seule image contenant une inter-réflexion et ont été utilisées pour estimer les paramètres intrinsèques de la caméra, les paramètres du plan miroir et la pose de la caméra. [Ying *et al.* (2010)] estime l'image de la conique absolue [Hartley et Zisserman (2003)] directement à partir des points correspondants dans une image unique contenant une inter-réflexion. L'idée se base sur le fait qu'un point 3D et ses réflexions se trouvent sur un cercle (Figure 2.7), et que tous ces cercles sont parallèles les uns aux autres. Plusieurs méthodes ont été proposées pour l'étalonnage en utilisant des cercles parallèles (ou coplanaires) [Gurdjos *et al.* (2006)], [Kim *et al.* (2005)] en trouvant les images de points circulaires [Hartley et Zisserman (2003)] [Semple et Kneebone (1998)].

La déformation des contours d'un objet dans l'image fournit des informations sur la forme

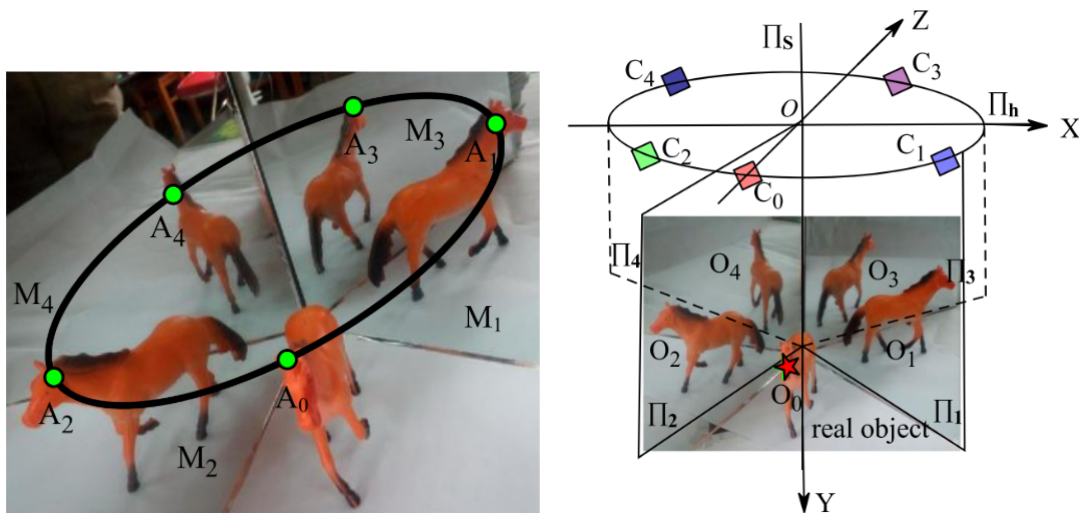


Figure 2.7 – Une image prise par le système catadioptrique planaire. Il y a 5 vues de l'objet dans une seule image. Un point 3D et ses réflexions se situent sur un cercle.

de sa surface. Le travail de [Huang et Lai (2006)] est basé sur une estimation de la forme 3D de tous les contours de l'objet (Fig 2.8). Le modèle 3D de l'objet peut être calculé à partir des contours de l'objet par une seule image de réflexion multiple catadioptrique. Dans [Huang et Lai (2006)], [Hu *et al.* (2005)], [Ying *et al.* (2013)] les auteurs ont utilisé un système catadioptrique avec deux miroirs plans pour capturer cinq vues de contour. L'idée de ces travaux est d'intégrer les informations du contour de l'objet à partir de différents points de vue pour la reconstruction d'objets 3D afin d'éviter de passer par l'étape de mise en correspondance des points. Ainsi, [Zhao *et al.* (2016)] ont proposé deux méthodes d'étalonnage des paramètres intrinsèques d'une caméra en fonction des points images circulaires. La première méthode proposée utilise l'inférence du théorème de Laguerre en géométrie projective pour résoudre l'équation des points circulaires. L'autre méthode utilise l'intersection entre l'image des points circulaires et la ligne épipolaire pour trouver les points circulaires. Néanmoins, ces méthodes sont incapables de récupérer certaines régions concaves et la précision dépend d'une bonne estimation des silhouettes. Ces techniques sont rapides mais ne permettent pas de trouver un modèle 3D de bonne qualité même pour le cas des objets de forme simple. Pour pouvoir déterminer correctement

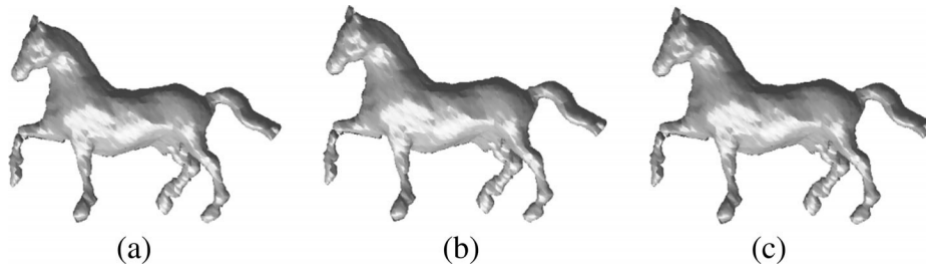


Figure 2.8 – Quelques résultats de reconstruction 3D utilisant les résultats d'étalonnage des points circulaires.

un nuage de points 3D dense, qui fournira par intégration la reconstruction 3D dense, il est nécessaire d'avoir des méthodes de mise en correspondances qui permettent d'obtenir de manière robuste les bons appariements entre les différentes paires de vue dans un système SCP.

2.3.2.2 Shape-From-shading

Le Shape-From-Shading, ou relief à partir des ombrages, consiste à calculer la forme tridimensionnelle d'une surface à partir des variations graduelles de son ombre dans une seule image 2D. La modélisation du processus de formation des images permet en effet d'exprimer le niveau de gris d'un pixel de l'image en fonction de la direction de la source de lumière et de la normale à la surface de l'objet.

Plusieurs études en vision passive, utilisant un système de vision catadioptrique planaire, ont proposé diverses méthodes pour obtenir des silhouettes d'objet [Ying *et al.* (2013), Zhang *et al.* (2012), Zhao et Li (2017)]. Certaines méthodes se basent sur les contours pour estimer les paramètres de la caméra [Huang et Lai (2006)]. Ces méthodes utilisent des contraintes spéciales telles que le mouvement circulaire, tandis que la caméra réelle et ses réflexions simples et doubles se situent sur le même cercle [Ying *et al.* (2010)]. Ainsi, les points de réflexion correspondants se trouvent sur un cercle et les points d'intersection du cercle et de la ligne à l'infini sont des points circulaires. Dans [Bastos et Stürzlinger (1998)], l'auteur a exploité une technique d'ombrage basée sur la projection par réflexion pour simuler des réflexions spéculaires pour un rendu interactif. [Shen et Cheng (2011)] ont utilisé une source de lumière pour étalonner géométriquement la caméra à l'aide d'un miroir plan avec un motif en damier pour pouvoir étalonner photométriquement la source, de façon à expliciter l'orientation commune des sources élémentaires (les pixels) et leurs coordonnées 3D. La direction de la source lumineuse peut être estimée avec précision à partir de l'image avec une orientation du miroir en récupérant la normale du plan du miroir.

2.4 Conclusion

Les systèmes catadioptrique planaires permettent de capturer plusieurs vues de la même scène dans une seule image. Le fait d'employer ces systèmes revient à utiliser une caméra réelle et plusieurs caméras virtuelles disposées autour de l'objet, ce qui nous permettra de réaliser une reconstruction 3D complète. Cependant, l'obtention d'une forme précise des objets reste un défi car les méthodes de reconstruction 3D existantes dépendent

de l'extraction des silhouettes des objets qui est particulièrement difficile quand les objets sont texturés, même avec plusieurs vues. Pour tenter de remédier à ces problèmes, nous présentons un nouveau pipeline de reconstruction 3D basé sur les points d'intérêt détectés sur l'objet observé par un SCP.

Sommaire

| | | |
|-------|--|----|
| 3.1 | Introduction | 21 |
| 3.2 | Étude du système catadioptrique planaire | 22 |
| 3.2.1 | Généralités et définitions autour du SCP | 22 |
| 3.2.2 | Vue d'ensemble et notations | 24 |
| 3.2.3 | Modélisation de caméra perspective | 26 |
| 3.2.4 | Paramètres internes | 27 |
| 3.2.5 | Paramètres externes | 27 |
| 3.3 | Étalonnage du système catadioptrique planaire | 29 |
| 3.3.1 | État de l'art de l'étalonnage de SCP | 30 |
| 3.3.2 | Étalonnage de caméra perspective | 30 |
| 3.3.3 | Calcul de pose des miroirs et extraction des normales et distances | 31 |
| 3.3.4 | Expérimentations et évaluation de l'étalonnage | 31 |
| 3.4 | Étude de la géométrie épipolaire réfléchive du SCP | 34 |
| 3.4.1 | Matrice fondamentale réfléchive | 36 |
| 3.4.2 | Matrice essentielle réfléchive | 38 |
| 3.4.3 | Validation de la robustesse de la matrice fondamentale réfléchive F_r | 40 |
| 3.5 | Conclusion | 43 |

3.1 Introduction

Comme évoqué au chapitre 2, la vision catadioptrique planaire est un procédé de vision qui fournit plusieurs vues d'un objet simultanément. Les caméras virtuelles créées par les réflexions sur les miroirs plans d'un tel système de vision ont plusieurs avantages par rapport aux systèmes de caméras multiples : synchronisation automatique utile pour les scènes dynamiques, paramètres de calibrage intrinsèques identiques ce qui simplifie la modélisation de la projection des points de la scène dans le plan image d'une caméra virtuelle. La reconstruction 3D d'un objet à partir de système de vision catadioptrique planaire exige d'exprimer dans une même référence les différentes composantes du système. Il faut procéder donc à une étape de calibration de la caméra et l'estimation de la pose de chaque miroir.

Notre système de vision Stéréo Catadioptrique Planaire (**SCP**) est composé d'une seule caméra et de deux miroirs plans judicieusement positionnés. Ce chapitre présente une méthode de calibration intrinsèque et extrinsèque du système par une modélisation sous la forme d'un réseau de caméras virtuelles. L'objectif principal dans ce chapitre est de

trouver la relation entre les différents capteurs réels et virtuels, et d'étudier l'influence de la réflexion induite par le miroir.

3.2 Étude du système catadioptrique planaire

3.2.1 Généralités et définitions autour du SCP

Dans cette section, nous allons définir l'ensemble du système catadioptrique planaire. Ensuite, nous présentons les caractéristiques apportées par des miroirs. Nous finissons par démontrer quelques propriétés des surfaces planes utilisées.

Un système catadioptrique est défini comme un système optique qui utilise à la fois des lentilles et des miroirs.

Les systèmes catadioptriques planaires sont des systèmes qui utilisent des miroirs plans. Ils ont la propriété du point de vue unique. Le centre de projection est situé symétriquement par rapport au plan du miroir et la caméra (Fig.3.1)

Pour un système composé d'un miroir plan et une caméra perspective, le point de vue unique est situé sur la bissectrice perpendiculaire à la droite reliant le point focal et son image virtuelle.

Une image virtuelle d'un objet est l'image formée par la réflexion d'un objet sur une

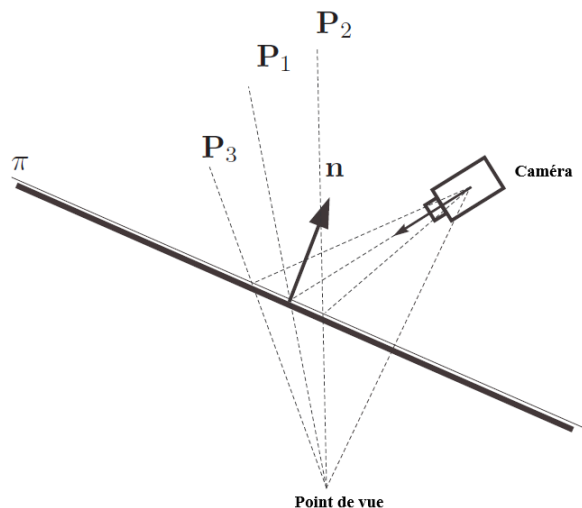


Figure 3.1 – Point de vue unique pour un miroir planaire.

surface. La nature de la réflexion est liée aux caractéristiques de la surface du miroir.

Une caméra virtuelle est une caméra obtenue par symétrie par rapport au plan du miroir à partir d'une caméra réelle (Fig.3.2).

Un point virtuel est défini comme un point obtenu par réflexion par rapport à un miroir à partir d'un point réel (Fig.3.2).

Dans le paragraphe suivant, nous allons présenter la matrice de réflexion d'un point par un miroir plan.

Théorème 1 : Soit Π_1 un plan défini dans le repère du monde \mathfrak{S}_W ayant la normale $\mathbf{n} = [n_x \ n_y \ n_z]$. La matrice de réflexion relative au plan Π_1 pour les points exprimés dans

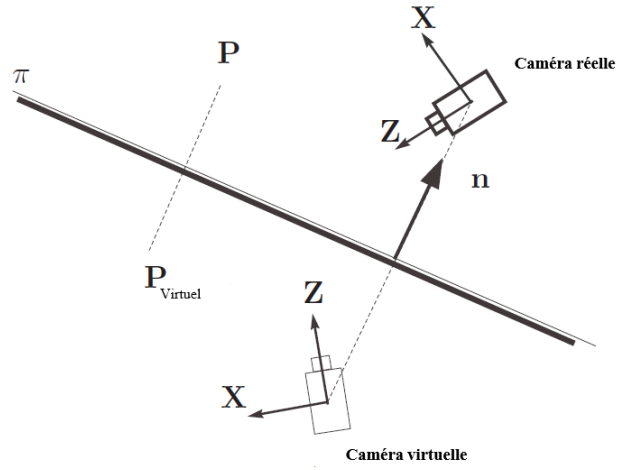


Figure 3.2 – Point et caméra virtuelle.

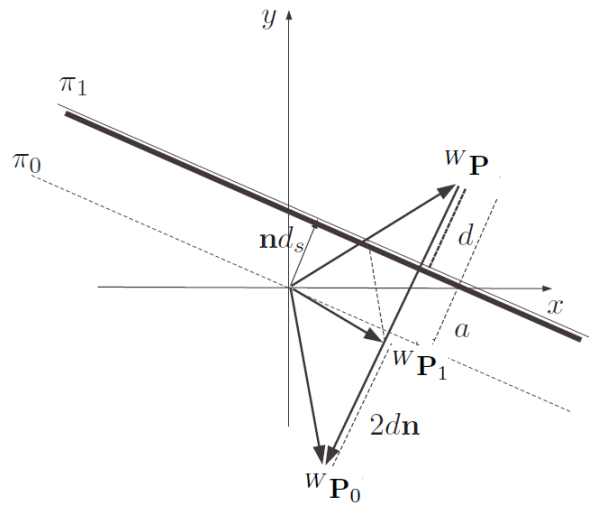


Figure 3.3 – Matrice de réflexions.

le repère monde \mathfrak{S}_W est définie comme :

$${}^W \mathbf{P} = (\mathbf{I} - 2\mathbf{nn}^T) {}^W \mathbf{P}_1 \quad (3.1)$$

avec $\mathbf{I} \in R^{3 \times 3}$ est une matrice d'identité.

Démonstration : Supposons que nous ayons un plan Π_1 passant par l'origine du repère du monde. Soit ${}^W \mathbf{P}_1$ le point réfléchi de ${}^W \mathbf{P}$ par rapport au plan Π_1 exprimé dans le repère monde \mathfrak{S}_W comme illustré dans la Fig 3.3

Le vecteur ${}^W \mathbf{P}_1$ peut être exprimé comme :

$${}^W \mathbf{P}_1 = {}^W \mathbf{P} + 2d\mathbf{n}. \quad (3.2)$$

où d'après la figure 3.3 nous avons $d = (-{}^W \mathbf{P}\mathbf{n})$ alors :

$${}^W \mathbf{P}_1 = (\mathbf{I} - 2\mathbf{nn}^T) {}^W \mathbf{P} \quad (3.3)$$

Nous allons procéder de la même façon mais cette fois nous considérons ${}^W\mathbf{P}_1$ est le point de réflexion et nous essayons de calculer ${}^W\mathbf{P}$ à partir de ${}^W\mathbf{P}_1$:

Le vecteur ${}^W\mathbf{P}$ est exprimé comme :

$${}^W\mathbf{P} = {}^W\mathbf{P}_1 - 2d\mathbf{n} \quad (3.4)$$

avec $d = ({}^W\mathbf{P}_1\mathbf{n})$ donc :

$${}^W\mathbf{P} = (\mathbf{I} - 2\mathbf{n}\mathbf{n}^T){}^W\mathbf{P}_1 \quad (3.5)$$

Théorème 2 : Orthogonalité de la matrice de réflexion \mathbf{D} :

$$\mathbf{D} = \mathbf{D}^{-1} \quad (3.6)$$

Démonstration : En référence aux équations précédentes, en substituant (3.6) dans (3.4), il vient : ${}^W\mathbf{P}_1 = \mathbf{D}\mathbf{D}{}^W\mathbf{P}_1$ D'où : $\mathbf{D}\mathbf{D} = \mathbf{I}$

donc :

$$\mathbf{D} = \mathbf{D}^{-1} \quad (3.7)$$

3.2.2 Vue d'ensemble et notations

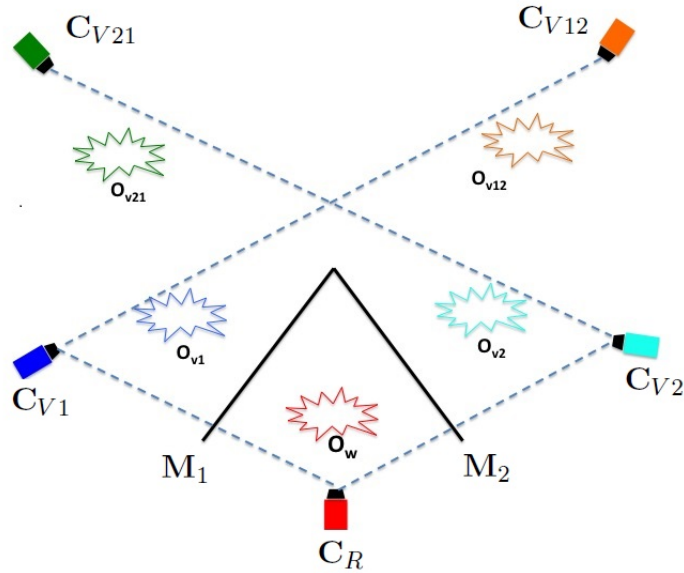


Figure 3.4 – Système Stéréo Catadioptrique planaire composé d'une caméra et deux miroirs plans.

La figure 3.4 montre une configuration à deux miroirs de notre SCP qui est utilisé pour capturer cinq vues d'un objet dans une seule image 3.5. La vue directe I_{C_R} est une vue réelle de l'objet observé par la caméra réelle. La vue symétrique I_{C_V} est le reflet de l'objet réel sur un miroir. La caméra est centrée sur C_R et observe un objet réel O_w . La caméra capture également l'image de chacun des quatre objets virtuels $O_{v1}, O_{v2}, O_{v12}, O_{v21}$. L'objet O_{v1} est le reflet de O_w par le miroir M_1 . L'objet O_{v2} est obtenu par une réflexion simple de O_w par le miroir M_2 . O_{v12} est obtenu par une réflexion double de O_w par le miroir M_1 puis le miroir M_2 . O_{v21} est obtenu par une réflexion double de O_w en miroir M_2 puis le miroir M_1 .

On note C_R la caméra réelle qui observe les différentes vues de la scène via les deux miroirs plans. Ainsi, C_{V1} est la caméra virtuelle symétrique à C_R par rapport au miroir M_1 .

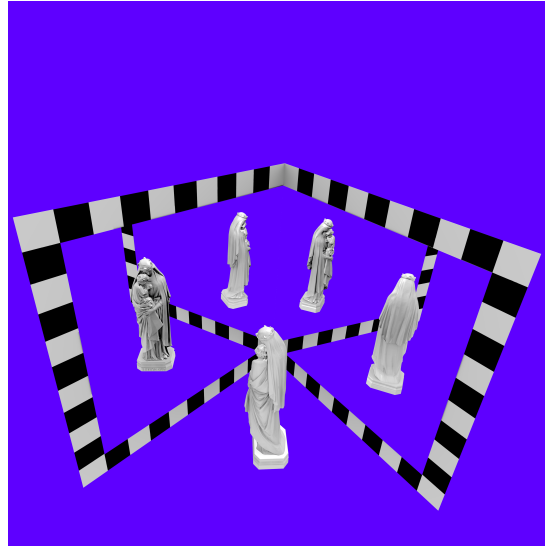


Figure 3.5 – Image de réflexion multiples catadioptrique

C_{V2} est la caméra virtuelle de C_R par rapport au miroir M_2 . C_{V12} est la caméra virtuelle symétrique à C_R obtenue par une double réflexion via M_1 puis M_2 , et C_{V21} est la caméra virtuelle obtenue par une double réflexion via M_2 puis M_1 .

Dans tout ce qui suit, nous utilisons le formalisme des coordonnées homogènes pour spécifier un point dans l'espace (donc 4 coordonnées) et les matrices de transformation sont de dimensions 4×4 .

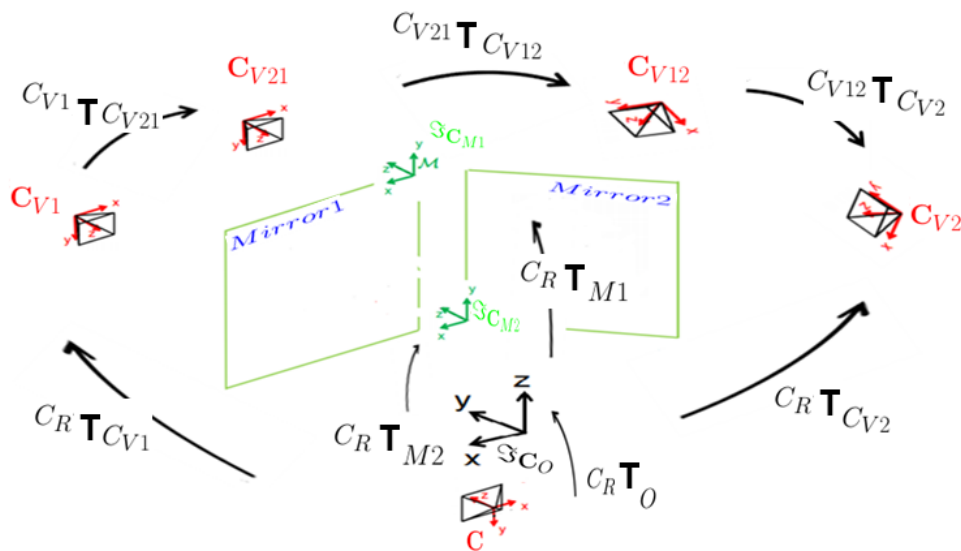


Figure 3.6 – Différentes transformations entre les repères

La figure 3.6 modélise l'ensemble du système, où les repères sont définis comme suit :

- $\mathfrak{S}_{C_R}(C, {}^c X, {}^c Y, {}^c Z)$ est le repère lié à la caméra perspective (ou conventionnelle) où :
 - C est le centre de projection de la caméra,
 - ${}^c Z$ est confondu avec l'axe optique de la caméra,

- cX et cY sont, respectivement, parallèles aux lignes et aux colonnes de l'image.
- $\mathfrak{S}_M(M, {}^mX, {}^mY, {}^mZ)$ est le repère lié au miroir de révolution.
- $\mathfrak{S}_O(O_o, {}^oX, {}^oY, {}^oZ)$ est le repère dans lequel est défini la structure 3D (ou modèle géométrique 3D) de l'objet observé, lorsque ce modèle est disponible.
- $\mathfrak{S}_W(W, {}^wX, {}^wY, {}^wZ)$ est le repère lié à la scène.
- $\mathfrak{S}_{C_{V_{ij}}}(C_{V_{ij}}, {}^{C_{V_{ij}}}X, {}^{C_{V_{ij}}}Y, {}^{C_{V_{ij}}}Z)$ est le repère lié à la caméra virtuelle ij

Dans notre cas, nous allons définir tout dans le repère de la caméra réelle. L'angle relatif entre les miroirs est fixe.

3.2.3 Modélisation de caméra perspective

La caméra est un capteur qui réalise la projection d'une scène tridimensionnelle dans un espace à deux dimensions appelé *plan image*. Dans le cas du modèle sténopé, la caméra est modélisée par un modèle de projection perspective. C'est le modèle le plus couramment utilisé dans la littérature. L'objectif de la caméra est modélisé alors comme un point C , le centre optique, par lequel passent les rayons lumineux réfléchis par la scène. Chaque point \mathcal{X} de la scène se situant devant la caméra est alors représenté par un rayon qui passe par \mathcal{X} et C . L'intersection de ce rayon avec le plan image est la projection de ce point dans l'image $\mathbf{x}_i = (u, v)$. La figure 3.7 illustre ce modèle. \mathbf{x}_0 est le point principal. C'est la

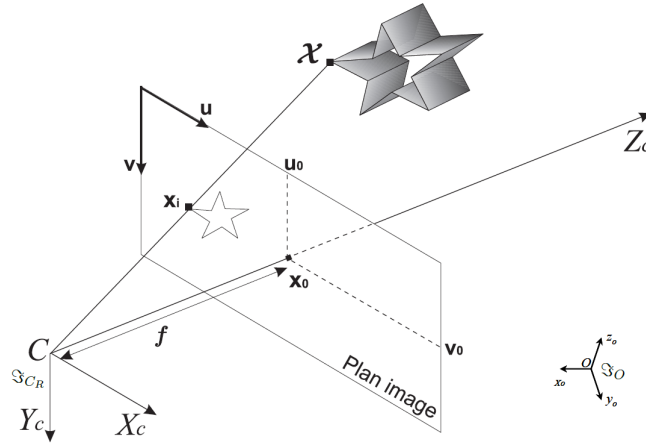


Figure 3.7 – Modèle de la caméra pinhole, d'après Hartley & Zisserman Hartley et Zisserman (2003)

projection du centre optique dans l'image. La distance entre le point C et le point \mathbf{x}_0 est appelée distance focale notée f .

Considérons le repère cartésien \mathfrak{S}_{C_R} associé à la caméra réelle d'origine C et d'axes (X_c, Y_c, Z_c) . Le point ${}^c\mathcal{X}$ est un vecteur de coordonnées exprimé dans le repère caméra réelle \mathfrak{S}_{C_R} .

$${}^c\mathcal{X} = [{}^cX \quad {}^cY \quad {}^cZ]^T \quad (3.8)$$

La projection d'un point ${}^c\mathcal{X}$ dans le plan image s'écrit en coordonnées homogènes :

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \mathbf{K}[I_3|0_{3 \times 1}] \begin{bmatrix} {}^cX \\ {}^cY \\ {}^cZ \\ 1 \end{bmatrix} \quad (3.9)$$

Où \mathbf{K} est la matrice des paramètres intrinsèques définie par l'équation :

$$\mathbf{K} = \begin{bmatrix} p_u & 0 & u_0 \\ 0 & p_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.10)$$

avec $p_u = k_u f$ et $p_v = k_v f$. k_u et k_v sont respectivement les coefficients de conversion en pixels suivant \vec{u} et \vec{v} .

3.2.4 Paramètres internes

Il s'agit des paramètres intrinsèques de quatre caméras virtuelles qui sont les reflets de la caméra réelle C_R . Ces quatre caméras virtuelles partagent le même jeu de paramètres intrinsèques que la caméra réelle. L'étude se restreint aux paramètres intrinsèques qui décrivent les propriétés internes de la caméra réelle : p_u, p_v, u_0 et v_0 . Ces paramètres sont présentés dans la section 3.2.5

3.2.5 Paramètres externes

Rappelons que nous avons un ensemble constitué de cinq caméras. On appelle paramètres extrinsèques, les paramètres de la transformation rigide entre deux caméras (fig 3.8b). En instanciant cette transformation en rotation $\{{}^{C_R}\mathbf{R}_{M1}, {}^{C_R}\mathbf{R}_{M2}\}$ et translation $\{{}^{C_R}\mathbf{t}_{M1}, {}^{C_R}\mathbf{t}_{M2}\}$. Soit la matrice homogène ${}^{C_R}\mathcal{T}_M$ de changement de repère du repère de miroir vers le repère caméra :

$${}^{C_R}\mathcal{T}_M = \begin{bmatrix} {}^{C_R}\mathbf{R}_M & {}^{C_R}\mathbf{t}_M \\ 0 & 1 \end{bmatrix} \quad (3.11)$$

Soit un point ${}^{C_R}\mathcal{X}$ représenté dans le repère de la caméra réelle et relié avec les points d'une mire collée sur les miroirs gauche et droite $\{{}^{M1}\mathcal{X}, {}^{M2}\mathcal{X}\}$ par la relation suivant :

$$\begin{aligned} {}^{C_R}\mathcal{X} &= {}^{C_R}\mathbf{R}_{M1} {}^{M1}\mathcal{X} + {}^{C_R}\mathbf{t}_{M1} \\ {}^{C_R}\mathcal{X} &= {}^{C_R}\mathbf{R}_{M2} {}^{M2}\mathcal{X} + {}^{C_R}\mathbf{t}_{M2} \end{aligned}$$

Nous nous intéressons maintenant au calcul de la pose entre la caméra réelle et les quatre caméras virtuelles. La matrice de projection entre la caméra réelle et la caméra virtuelle s'écrit sous la forme suivante :

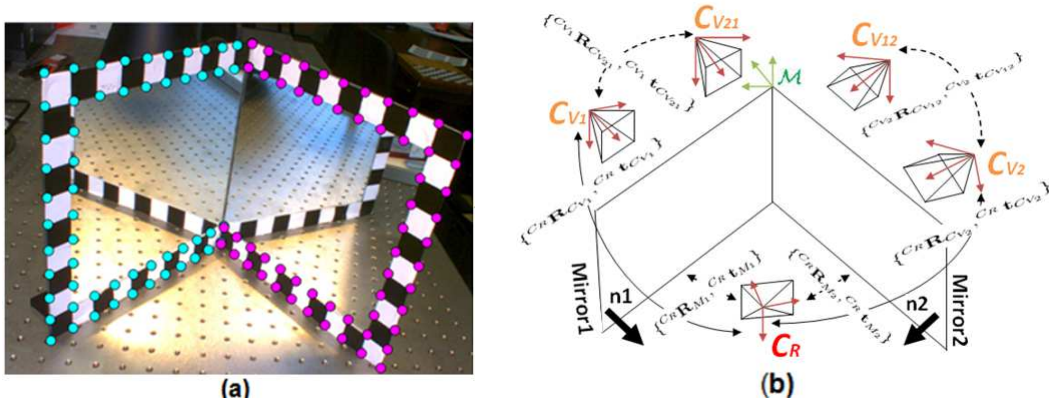


Figure 3.8 – (a) Exemple de détection des coins sur la mire associée aux miroirs dans l'image. (b) Système équivalent, décrivant les emplacements des cinq caméras. La caméra réelle est en face des deux miroirs planaires. Les quatre caméras virtuelles sont derrière les miroirs et sont situées à des emplacements symétriques à la caméra réelle par simple et double réflexion par rapport aux plans des deux miroirs.

$$\begin{aligned}
 P_v &= \mathbf{K} {}^{C_R} \mathcal{T}_M \mathcal{D}^{\Pi_M} \\
 &= \mathbf{K} \mathbf{R}_{C_R} [I \quad | \quad -\mathbf{t}_{C_R}] \begin{bmatrix} I - 2\mathbf{n}_m \mathbf{n}_m^T & -2d_m \mathbf{n}_m \\ \mathbf{0}_{1 \times 3}^T & 1 \end{bmatrix} \\
 &= \mathbf{K} \mathbf{R}_{C_R} (I - 2\mathbf{n}_m \mathbf{n}_m^T) (2d_m \mathbf{n}_m - (I - 2\mathbf{n}_m \mathbf{n}_m^T) \mathbf{t}_{C_R})
 \end{aligned} \tag{3.12}$$

où P_v est la matrice de projection de la caméra virtuelle. \mathbf{K} est la matrice de calibration et \mathcal{D}^{Π_M} est la matrice de réflexion. On peut réécrire l'équation (3.12) de la manière suivante :

$$P_v = K [\mathbf{R}' | \mathbf{t}'] \text{ avec } \begin{cases} \mathbf{R}' = \mathbf{R}_{C_R} (2\mathbf{n}_m \mathbf{n}_m^T - \mathbf{I}) \\ -\mathbf{t}' = 2d_m \mathbf{n}_m - (\mathbf{I} - 2\mathbf{n}_m \mathbf{n}_m^T) \mathbf{t}_{C_R} \end{cases} \tag{3.13}$$

où, R' représente la matrice de rotation et t' la translation entre les deux caméras.

a) Transformation entre plusieurs vues :

Étant donné ${}^{C_R} \mathbf{D}_{C_{V_1}}$ la transformation de réflexion entre la caméra réelle et la caméra virtuelle C_{V_1} , notons cette transformation comme suit :

$${}^{C_R} \mathbf{D}_{C_{V_1}} = \begin{bmatrix} \mathbf{I} - 2\mathbf{n}_1 \mathbf{n}_1^T & 2d_1 \mathbf{n}_1 \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}$$

La transformation ${}^{C_{V_1}} \mathbf{D}_{C_{V_{12}}}$ de réflexion entre la caméra virtuelle C_{V_1} et la caméra virtuelle $C_{V_{12}}$ est :

$${}^{C_{V_1}} \mathbf{D}_{C_{V_{12}}} = \begin{bmatrix} \mathbf{I} - 2\mathbf{n}_1 \mathbf{n}_1^T & 2d_1 \mathbf{n}_2 \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}$$

La transformation entre la caméra réelle et la caméra virtuelle C_{V12} est obtenue par une double réflexion :

$${}^{C_R}\mathbf{D}_{C_{V12}} = {}^{C_R}\mathbf{D}_{C_{V1}} {}^{C_{V1}}\mathbf{D}_{C_{V12}} = \begin{bmatrix} \mathbf{I} - 2\mathbf{n}_1\mathbf{n}_1^T & 2d_1\mathbf{n}_1 \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I} - 2\mathbf{n}_1\mathbf{n}_1^T & 2d_1\mathbf{n}_2 \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}$$

On procède de la même manière pour le deuxième miroir et on en déduit la transformation entre la caméra réelle et la caméra virtuelle C_{V21} :

$${}^{C_R}\mathbf{D}_{C_{V21}} = {}^{C_R}\mathbf{D}_{C_{V2}} {}^{C_{V2}}\mathbf{D}_{C_{V21}} = \begin{bmatrix} \mathbf{I} - 2\mathbf{n}_2\mathbf{n}_2^T & 2d_2\mathbf{n}_2 \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I} - 2\mathbf{n}_2\mathbf{n}_2^T & 2d_2\mathbf{n}_1 \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}$$

Soit un repère de l'objet \mathfrak{S}_O et le repère de la caméra virtuelle C_{V1} $\mathfrak{S}_{C_{V1}}$. Un point ${}^O\mathbf{X}$ défini dans \mathfrak{S}_O peut être exprimé dans $\mathfrak{S}_{C_{V1}}$ à travers la fonction suivante :

$${}^{C_{V1}}\mathbf{X} = {}^{C_{V1}}\mathbf{D}_{C_R} {}^{C_R}\mathbf{T}_O {}^O\mathbf{X} \quad (3.14)$$

avec ${}^{C_R}\mathbf{D}_{C_{V1}} = {}^{C_{V1}}\mathbf{D}_{C_R}$.

b) Détermination de l'orientation et de la position relatives :

Ayant calculé les normales des miroirs plans, nous pouvons calculer les orientations relatives des caméras virtuelles en utilisant la propriété de réflexion de miroir comme suit :

$$\begin{aligned} {}^{C_R}\mathbf{R}_{C_{V1}} &= \mathbf{R}_{C_R} - 2\mathbf{R}_{C_R}\mathbf{n}_1\mathbf{n}_1^T & {}^{C_R}\mathbf{R}_{C_{V2}} &= \mathbf{R}_{C_R} - 2\mathbf{R}_{C_R}\mathbf{n}_2\mathbf{n}_2^T \\ {}^{C_{V1}}\mathbf{R}_{C_{V21}} &= {}^{C_R}\mathbf{R}_{C_{V1}} - 2{}^{C_R}\mathbf{R}_{C_{V1}}\mathbf{n}_2\mathbf{n}_2^T & {}^{C_{V2}}\mathbf{R}_{C_{V12}} &= {}^{C_R}\mathbf{R}_{C_{V2}} - 2{}^{C_R}\mathbf{R}_{C_{V2}}\mathbf{n}_1\mathbf{n}_1^T \end{aligned}$$

La translation \mathbf{t} peut être calculée de la façon suivante :

$$\begin{aligned} {}^{C_R}\mathbf{t}_{C_{V1}} &= 2d_1\mathbf{n}_1 & {}^{C_R}\mathbf{t}_{C_{V2}} &= 2d_2\mathbf{n}_2 \\ {}^{C_{V1}}\mathbf{t}_{C_{V21}} &= 2d_2\mathbf{n}_1 & {}^{C_{V2}}\mathbf{t}_{C_{V12}} &= 2d_1\mathbf{n}_2 \end{aligned}$$

3.3 Étalonage du système catadioptrique planaire

Dans cette section, nous allons proposer une solution au problème de la calibration de notre système qui fournit plusieurs caméras virtuelles à travers les miroirs plans. Pour cela, nous allons estimer les paramètres intrinsèques et extrinsèques de chaque caméra. La phase de calibration permet de déterminer les paramètres du modèle géométrique liant les coordonnées des points 3D de la scène observée aux coordonnées 2D de leur projection dans l'image. L'ensemble des modèles décrivant la géométrie d'un SCP est composé de deux éléments : les transformations de réflexion des deux miroirs plans et la transformation rigide de la caméra réelle donnant la position des quatre caméras virtuelles. Ainsi, nous séparons les paramètres du modèle d'un SCP en deux catégories. La première correspond aux paramètres intrinsèques de la caméra réelle. La seconde consiste à déterminer les paramètres extrinsèques de chaque point de vue. Ce qui revient à déterminer la pose de chaque caméra par rapport à l'autre ou par rapport à un point de vue pris comme référence.

3.3.1 État de l'art de l'étalonnage de SCP

Pour calibrer un système SCP, nous cherchons à estimer les paramètres des plans miroirs et la pose de la caméra réelle par résolution d'un système linéaire. On en déduira les poses des caméras virtuelles. Des travaux précédents ont été effectués sur le calibrage d'un système multi-caméra, via un miroir et une mire plane connue [Kumar *et al.* (2008)]. Une autre utilisation des miroirs plans est faite dans [Gluckman et Nayar (2001)] pour réaliser une stéréovision à l'aide d'une seule caméra. [Martins et Dias (2004)] place un objet entre la caméra et des miroirs plans pour calibrer le système par la présence de réflexions dans une image. Récemment, des miroirs plans ont été utilisés pour déterminer la pose d'une mire de calibration, par rapport à une caméra [Sturm et Bonfort (2006)]. Des travaux similaires ont été menés dans [Hesch *et al.* (2008)], afin de résoudre analytiquement le problème. Enfin, des travaux ont été effectués sur la calibration extrinsèque d'un système catadioptrique planaire, via des mires positionnées aux emplacements des miroirs plans connus [Lanman *et al.* (2009)].

Plusieurs autres méthodes se basent sur l'auto-calibrage sans utiliser les mires de calibrage [Forbes *et al.* (2006)], [Gluckman et Nayar (1999)], [Hu *et al.* (2005)], [Lin *et al.* (2002)]. En bref, ces méthodes comprennent : l'obtention d'une calibration miroir à partir de silhouettes d'objets extraites des images réelles et réfléchies [Forbes *et al.* (2006)], l'exploitation de la géométrie épipolaire d'un couple d'images réelles et réfléchies [Gluckman et Nayar (1999)], et l'utilisation d'une procédure d'ajustement de faisceaux pour aligner les images réelles et prédites [Lébraly13 *et al.* (2010)].

Dans cette étude nous proposons un système qui permet un calibrage direct. Pour cela, nous rajoutons au système SCP une mire de calibrage collée sur les bords des deux miroirs plans. Nous supposons que chaque plan miroir représente une mire contenant un nombre de points suffisant pour faire une estimation de la matrice de calibration. Par l'utilisation du même principe de la théorie de Zhang [Zhang (1999)], nous pourrions déterminer directement les paramètres internes à l'aide de la toolbox de [Bouguet (2004)]. De plus, une étude géométrique du système permettrait de déterminer le positionnement relatif entre chaque paire de caméras (souvent appelée la pose relative), ainsi que les paramètres de chaque plan miroir. La résolution de ce problème va rendre le calibrage beaucoup plus simple et efficace par rapport aux autres méthodes.

3.3.2 Etalonnage de caméra perspective

Nous allons commencer par l'estimation des paramètres intrinsèques de la caméra réelle à partir des bandes du damier collées autour des miroirs comme illustrés dans la figure (3.8-a). Nous obtenons avec cette configuration deux poses différentes pour chaque image avec un nombre de points suffisants.

On transforme le problème en un problème classique de calibration de caméra perspective. Pour cela on enregistre une série de paire d'images d'une mire disposée sur le contour de chaque surface de miroir plan comme illustré dans la figure 3.9. Chaque image est découpée en deux imageries contenant chacune un plan miroir comprenant un motif en damier (Fig.3.10) afin d'appliquer la toolbox bouguet. Dans notre implémentation, nous utilisons la méthode de Bouguet [Bouguet (2004)].

3.3.3 Calcul de pose des miroirs et extraction des normales et distances

Pour déterminer l'emplacement d'un miroir planaire, nous commençons par estimer son plan de support. Nous avons disposé sur le contour de la surface du miroir des sous-bandes de la mire. Les points 2D projetés sur la zone de contour du miroir sont déterminés par la toolbox de Bouguet. Une fois ces points sont déterminés, on peut retrouver la normale du plan par un simple produit vectoriel. La distance du plan est finalement trouvée en résolvant l'équation du plan pour un point donné ou pour le centre du groupe de points. De plus, pour définir l'équation du plan on crée le vecteur $\Pi_m = (\mathbf{n}_m^T, -d_m)^T = (n_x, n_y, n_z, -d_m)^T$. Tout point de coordonnées $(X_m, Y_m, Z_m, 1)^T$ dans le repère monde appartenant au plan Π_m vérifie l'équation de la forme :

$$\Pi_m^T \begin{bmatrix} X_m \\ Y_m \\ Z_m \\ 1 \end{bmatrix} = 0 \quad (3.15)$$

où $\mathbf{n}_m = [n_x \ n_y \ n_z]^T$ est le vecteur normal au plan miroir et d_m est la distance orthogonal du plan au centre de la caméra réelle. En particulier, si le point 3D X_m (défini dans le même repère) appartient au plan miroir, par conséquent, on peut calculer la valeur d_m en posant :

$$0 = \Pi_m^T \mathbf{X}_m \quad (3.16)$$

$$d_m = n_x X_m + n_y Y_m + n_z Z_m = \mathbf{n}_m^T \mathbf{X}_m \quad (3.17)$$

Alors, la réflexion par le plan miroir Π_M sera représenté par la matrice de réflexion sous la forme suivante :

$$\mathbf{D}^{\Pi_M} = \begin{bmatrix} \mathbf{I} - 2\mathbf{n}_m \mathbf{n}_m^T & -2d_m \mathbf{n}_m \\ 0^T & 1 \end{bmatrix} \quad (3.18)$$

3.3.4 Expérimentations et évaluation de l'étalonnage

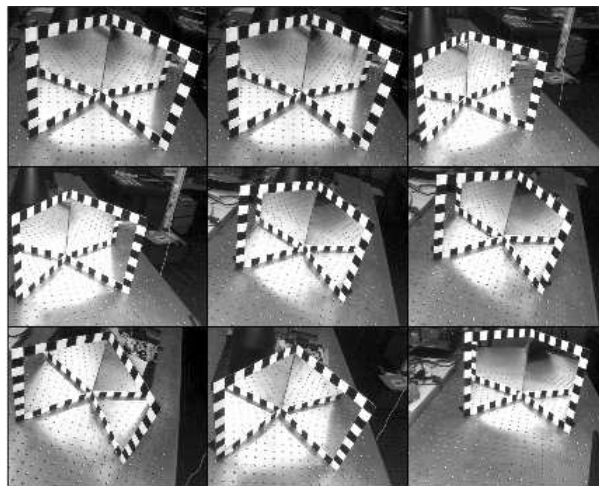


Figure 3.9 – Images de sous-bandes d'une mire disposée sur le contour de la surface de chaque miroir planaire, projetées avec les points saillants détectés.

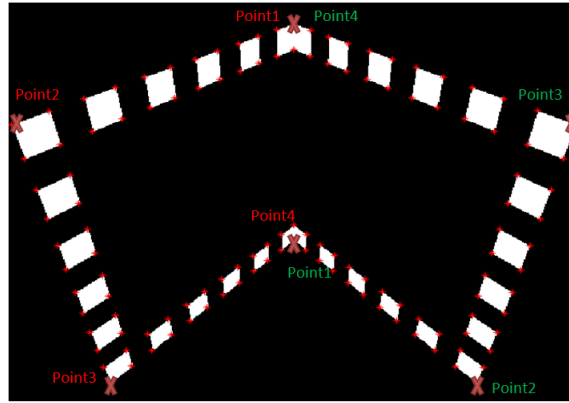


Figure 3.10 – Projection de la grille détectée par rapport aux miroirs planaires. Les coins des bandes de la mire sont colorés par des cercles en rouges.

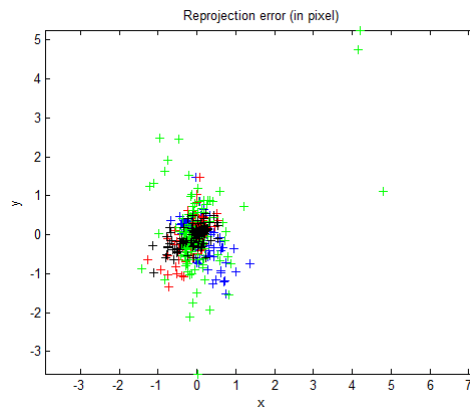


Figure 3.11 – Erreur de calibrage pour chaque image.

3.3.4.1 Résultats sur des données de synthèse

Résultat obtenus :

La distance focale : $fc = [1016.61 \quad 1021.99] \pm [4.30 \quad 2.32]$

Principal point : $cc = [639.47 \quad 513.07] \pm [1.98 \quad 2.32]$

Facteur d'obliquité : $alphac = [0.00] \pm [0.00] \Rightarrow$ angle of pixel axes = 90.00 ± 0.00 degrees

Distorsion : $kc = [0.00 \quad 0.00 \quad 0.00 \quad 0.00 \quad 0.00] \pm [0.00 \quad 0.00 \quad 0.00 \quad 0.00 \quad 0.00]$

Erreur pixellique = $[0.35 \quad 0.37]$

Le tableau 3.1 montre un écart entre les paramètres de la vérité de terrain et les para-

| Paramètres | Params réels | Params Estimés |
|------------|--------------|----------------|
| α_u | 1024 | 1016.61 |
| α_v | 1024 | 1021.99 |
| u_0 | 640 | 639.47 |
| v_0 | 512 | 513.07 |

Tableau 3.1 – Résultats de calibration de la caméra perspective

mètres estimé dû à l'utilisation d'un nombre minimum de prises de vue même si l'erreur pixellique est faible. Lorsqu'on augmente le nombre d'images nous améliorons les paramètres de calibrage et l'écart devient très faible.

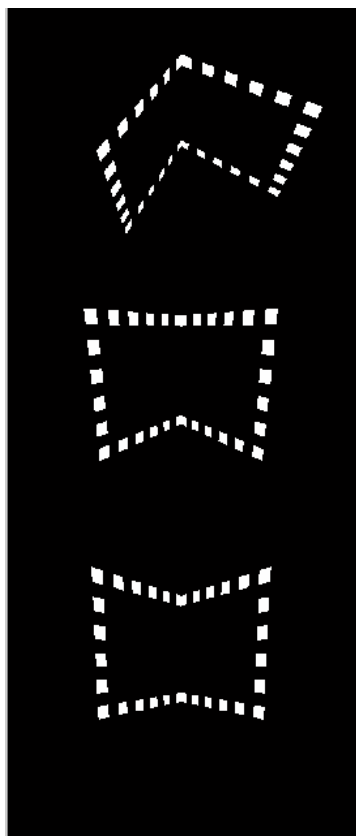


Figure 3.12 – Ensemble des images choisies pour la procédure de calibrage. Utilisation de trois plans faisant une rotation suivant un seul axe.

Le tableau 3.2 montre le résultat obtenu sur la mire après calibration de la caméra. On

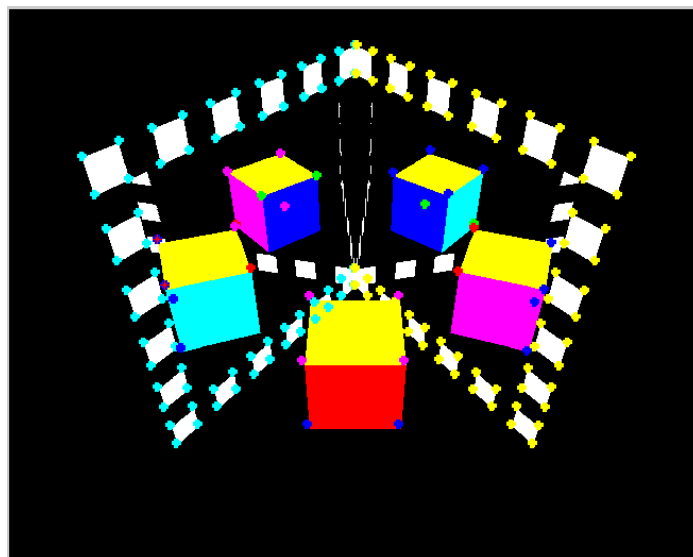


Figure 3.13 – Projection des points 3D de la mire gauche et droite au plan image pour une image de résolution 1280×1024 sans antialiasing.

obtient une erreur quadratique moyenne de reprojection des points de 0.33 pixels pour le cas de l'image synthétique 512×512 et 0.36 pixels pour le cas de l'image synthétique 1280×1024 . Ces résultats montrent la précision des paramètres de calibrage.

| | Erreur moyenne | Erreur minimale | Erreur maximale | Ecart type |
|--------------------------------|----------------|-----------------|-----------------|------------|
| Image Synthétique 512x512 | 0.33 | -1.25 | 1.20 | 0.28 |
| Image Synthétique 1024x768 | 0.31 | -1.41 | 1.34 | 0.30 |
| Image Synthétique 1280x1024 | 0.36 | -1.71 | 1.70 | 0.35 |

Tableau 3.2 – Les résultats quantitatifs des paramètres de calibrage.

Nous allons ajouter un bruit entre 0 et 1 aux points 2D après l'extraction à partir du détecteur de Harris, nous remarquons qu'il y a pas un grand changement au niveau des résultats obtenus comme illustré dans la figure 3.14. Pour une image de résolution 1024 ×

768. La matrice de calibration estimée est : $K = \begin{bmatrix} 763 & 0 & 512 \\ 0 & 766 & 386 \\ 0 & 0 & 1 \end{bmatrix}$;

$T = [-0.0175 \quad 0.0085 \quad 1.5997]$ et $R = \begin{bmatrix} 0.5788 & 0.0055 & -0.8155 \\ 0.5091 & -0.7836 & 0.3560 \\ -0.6371 & -0.6212 & -0.4564 \end{bmatrix}$.

pour une seule acquisition (Image de la mire).

3.3.4.2 Résultats sur des données réelles

Dans cette partie, nous allons évaluer la robustesse de la matrice de projection. On commence par reprojeter les points utilisés dans le calcul de la projection pour déterminer l'exactitude de la matrice de projection obtenue. L'erreur quadratique moyenne est calculée et les points reprojétés sont affichés pour la comparaison graphique du résultat (fig 3.15). De manière à estimer l'ensemble des onze paramètres, nous calculons pour chacune des trois images (6 plans) la moyenne de l'ensemble des valeurs. Une fois le calcul de la projection effectué, on reprojette les points qui ont servi à déterminer la dite projection et on calcule l'erreur quadratique moyenne associée pour que l'on puisse vérifier la pertinence du résultat obtenu. La figure 3.15 montre la précision des points projetés avec les paramètres de calibration estimés. On obtient une erreur quadratique moyenne de reprojection des points de 1.09 pixels pour image de 2048x2048 pixels .

La figure 3.16 regroupe l'ensemble des plans utilisés en changeant l'angle entre les miroirs.

3.4 Étude de la géométrie épipolaire réflexive du SCP

La géométrie épipolaire s'applique aux systèmes composés d'une seule caméra et un miroir plan parce qu'ils fournissent deux vues d'une même scène prise de points de vue différents. L'image de l'objet réel et l'image de l'objet virtuel sont liées par des contraintes qu'il est possible de traduire à l'aide d'une matrice singulière 3×3 appelée matrice fondamentale. Dans cette partie, nous analysons les propriétés de la géométrie d'un système caméra-miroirs, et nous montrons qu'il existe des contraintes fortes dans cette géométrie

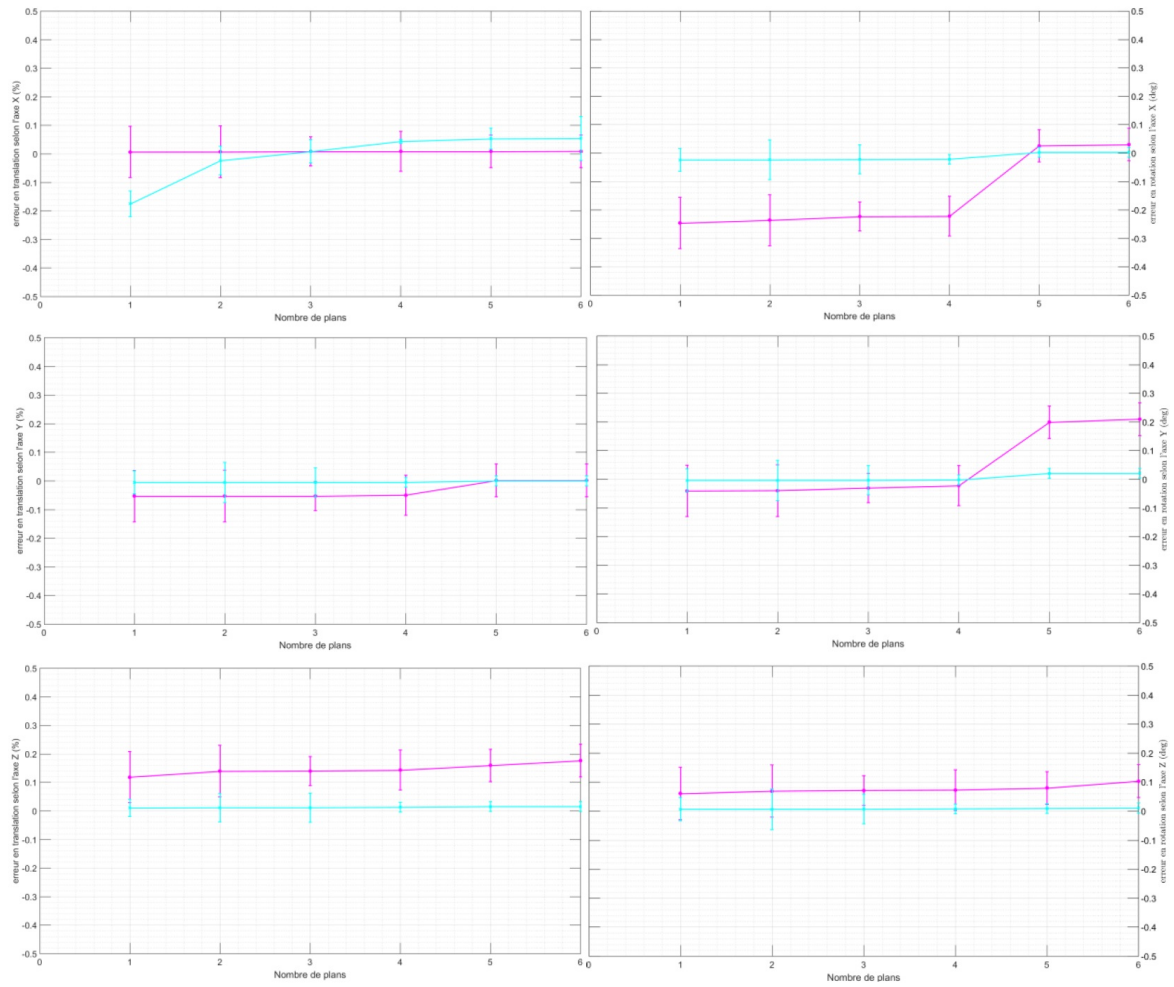


Figure 3.14 – Comparaison de la robustesse au bruit sur les positions des plans miroirs, positions des points 2D sélectionnés sur la mire.

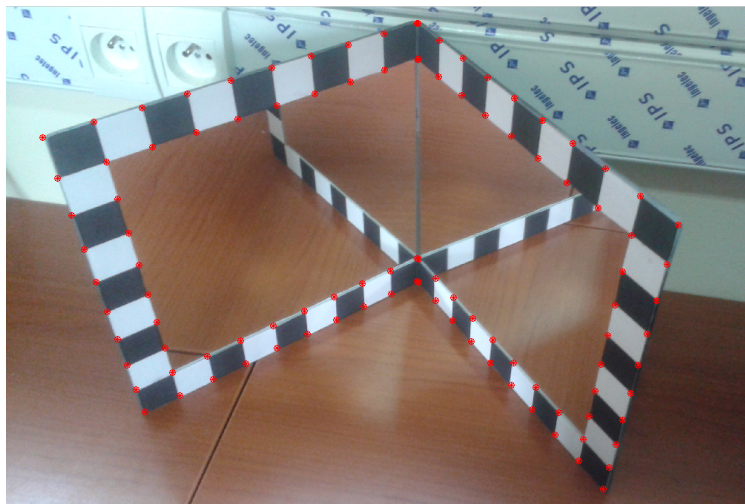


Figure 3.15 – Cette figure montre les points rouges extraits par le détecteur de Harris et les points verts sont les points calculés après l'estimation des paramètres de calibration.

Erreur moyenne quadratique de reprojection est : 1.09 pixels

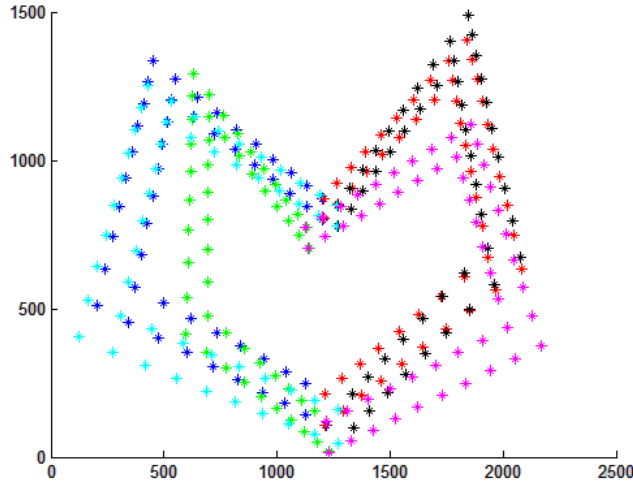


Figure 3.16 – L'ensemble des plans

pour rendre la matrice fondamentale plus simple. Nous verrons ensuite que le nombre de degrés de liberté (ddl) pourra diminuer de sept à deux, réduisant ainsi le nombre de points nécessaires pour récupérer la géométrie épipolaire entre la caméra réelle et la caméra virtuelle. Nous introduisons la géométrie épipolaire réflexive qui est plus appropriée pour les systèmes stéréo catadioptriques planaires **SCP**. Les résultats expérimentaux montreront que cette simplification permet d'augmenter la précision et la robustesse de l'estimation de la matrice fondamentale \mathbf{F} .

3.4.1 Matrice fondamentale réflexive

La géométrie épipolaire réflexive est la configuration géométrique formée par un système composé d'une seule caméra et au moins un miroir plan qui fournissent deux vues symétriques. Elle dépend des paramètres intrinsèques des caméras (réelle/virtuelle), ainsi que leurs positions relatives en plus des paramètres des plans miroirs. Les lignes épipolaires et les épipoles sont identiques dans les images réelles et virtuelles. C'est la **contrainte épipolaire réflexive**. Dans ce cas, la matrice fondamentale \mathbf{F}_r peut être décrite comme suit :

$$\mathbf{F}_r = \begin{bmatrix} 0 & -e_3 & e_2 \\ e_3 & 0 & -e_1 \\ -e_2 & e_1 & 0 \end{bmatrix} \quad \text{où } [e_1, e_2, e_3] = \mathbf{e} \text{ est l'épipole entre les caméras réelle et virtuelle (Fig.3.17).}$$

Considérons un point u_R dans le plan image de la caméra réelle et son correspondant u_V dans le plan image de la caméra virtuelle. En utilisant la matrice fondamentale réflexive \mathbf{F}_r , la relation entre les deux points est décrite dans [Hartley et Zisserman (2003)] comme suit :

$$u_V^T \mathbf{F}_r u_R = 0 \quad (3.19)$$

La matrice fondamentale réflexive est une matrice antisymétrique $[\mathbf{F}_r]_{3 \times 3}$ et singulière puisque $\det(\mathbf{F}_r) = 0$ et de rang 2. En effet, \mathbf{F}_r est une matrice de passage qui relie 2 matrices 3×4 de projection, \mathbf{P}_R et \mathbf{P}_V de la caméra réelle et virtuelle respectivement, ayant 10 degrés de liberté (3 pour la rotation, 3 pour la translation et 3 pour le plan du miroir). Le fait d'utiliser une seule caméra permet de réduire le nombre de paramètres

intrinsèques de 9 à 5 [Gluckman et Nayar (2001)]. En outre, la contrainte du mouvement planaire permet de réduire les ddl de la pose relative de 6 à 5 (3 pour la rotation et 2 pour la translation dans un plan). Par ailleurs, les deux centres optiques C_R et C_V sont liés par une transformation rigide dont la matrice a 12 degrés de liberté, et on sait que les

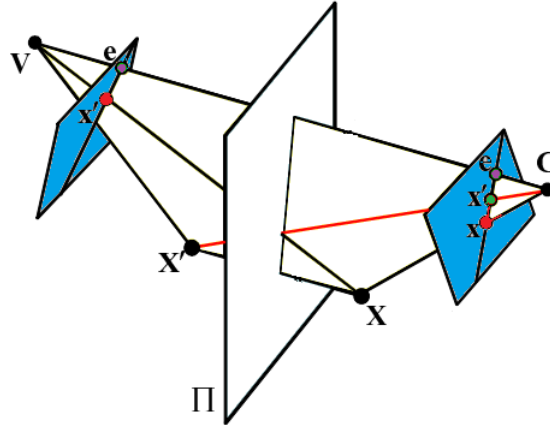


Figure 3.17 – La géométrie de deux vue par un miroir Plan

projections en question sont définies par rapport à ces centres optiques. Les 12 paramètres indépendants sont liés à la rotation et à la translation qui permettent de passer de la caméra réelle à une caméra virtuelle. En conséquence, parmi 12 degrés de liberté, 10 peuvent être levés et on dispose $12-10 = 2$ ddl pour la matrice \mathbf{F}_r qui relie ces projections. c-à-d que ces deux projections sont liés juste par la matrice de réflexion.

L'explication décrite ci-dessus fournit une matrice fondamentale réflexive qui satisfait la contrainte de rang 2. Avec cette contrainte, toutes les droites épipolaires dans une imagette et leur reflet passent par trois points colinéaires : le point réel u_R , son correspondant dans l'image miroir u_M et l'auto-épipole e_{RM} . Ceci montre que les droites épipolaires de l'image gauche et droite sont confondues.

La matrice fondamentale entre une vue réelle et vue virtuelle dite réflexive comme montrée dans la Fig.3.18 n'est autre qu'une représentation algébrique de la contrainte de coplanarité des points. Elle permet de simplifier la phase de recherche des points de correspondance des 2 images.

L'épipole e_R correspond au noyau de la matrice fondamentale \mathbf{F}_r . De même l'épipole e_V correspond au noyau gauche de \mathbf{F}_r . A partir de \mathbf{F}_r et des épipoles, il est possible d'estimer un couple de matrices projectives en adéquation avec la géométrie épipolaire. Il est toujours possible de fixer l'espace projectif arbitrairement de telle manière que la matrice de la caméra réelle soit $\mathbf{P}_1 = [\mathbf{I}|0]$.

Proposition 1 : La matrice \mathbf{F}_r est une matrice fondamentale réflexive associée aux caméras réelle et virtuelle si et seulement si \mathbf{F}_r est antisymétrique.

Preuve :

\mathbf{F}_r est antisymétrique $\Leftrightarrow \mathbf{E} = [\mathbf{t}]_{\times}$ avec $\mathbf{R} = \mathbf{I}_3$ et $\mathbf{K} = \mathbf{K}'$ alors $\mathbf{F}_r = \mathbf{K}^{-T}[\mathbf{t}]_{\times}\mathbf{K}$.

si \mathbf{F}_r est antisymétrique $e_R = e_V$ c-à-d le noyau gauche de \mathbf{F}_r égal au noyau droite de \mathbf{F}_r .

Il est alors facile de vérifier que \mathbf{F}_r est antisymétrique en posant $\mathbf{P}_1 = [\mathbf{I}|0]$ et $\mathbf{P}_2 = [[\mathbf{e}]_{\times}\mathbf{F}_r|\mathbf{e}]$. Ainsi ces caméras projectives satisfaisant la géométrie épipolaire décrite par \mathbf{F}_r .

Remarque :

Nous pouvons interpréter ce cas autrement. Une caméra virtuelle est une caméra qui a effectuée un déplacement le long du plan miroir (sans changement d'orientation), avec sa

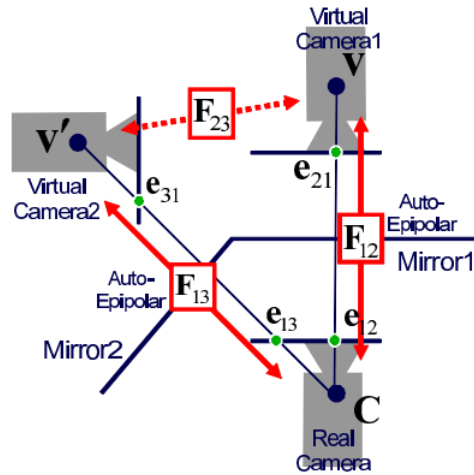


Figure 3.18 – La matrice fondamentale et l’auto-épipole entre chaque vue réelle et deux vues virtuelles.

matrice de calibration, qui reste inchangée au cours du déplacement. Ceci montre que la translation entre la vue directe et la vue virtuelle est une *translation pure*, et qui prouve que F est *anti-symétrique*.

Le calcul de la matrice fondamentale réflexive repose sur l’idée de choisir un point réel u_R sur l’image réelle, puis de déterminer un point objet \mathbf{X}_R se projetant sur ce point image, et finalement de calculer l’image \mathbf{X}_V de \mathbf{X}_R sur l’image virtuelle. Ce point \mathbf{X}_V et l’épipole e_V de la deuxième image miroir se trouvent sur la droite épipolaire et la déterminent donc de façon unique.

On a pu montrer que F_r est anti-symétrique et ne dépend que de l’auto-épipole. De plus, cet auto-épipole et le couple correspondant appartiennent à la même droite. On peut conclure que la translation pure de la caméra virtuelle suivant l’axe perpendiculaire à l’axe optique du miroir plan, et qui est symétrique à la caméra réelle, rend la géométrie épipolaire un peu particulière : les lignes épipolaires et les épipoles sont identiques dans l’image réelle et virtuelle. C’est ce qu’on peut appeler la CONTRAINTE AUTO-ÉPIPOLAIRE.

3.4.2 Matrice essentielle réflexive

La matrice essentielle représente la géométrie épipolaire, mais cette fois-ci exprimée par rapport aux repères normalisés. On dit que la matrice essentielle représente la *géométrie épipolaire calibrée*.

Tout point \mathbf{X} situé sur la droite reliant le centre de projection et \mathbf{x} peut-être considéré comme le point mesuré. Dans un système SCP, il est possible de considérer la mesure \mathbf{x} comme une droite reliant le centre de projection C_R à \mathbf{X} . Cette droite peut-être reprojétée dans le repère \mathfrak{S}_{C_V} puisque ses paramètres sont connus dans le repère \mathfrak{S}_{C_R} .

En conséquence, si le point est visible dans l’image, sa position ${}^{C_V}\mathbf{x}$ est nécessairement le long de la projection de cette droite dans l’image. Le fait que le point soit forcément devant l’objectif de la caméra permet d’éliminer tous les points situés avant la projection \mathbf{e} de l’origine de \mathfrak{S}_{C_R} dans l’image de la caméra C_{V1} . Cette contrainte, communément appelée contrainte épipolaire, permet de limiter l’espace de recherche lors de la mise en correspondance d’un point dans deux images. Elle s’exprime analytiquement par l’équation :

$${}^{C_{V1}\mathbf{x}^T} {}^{C_V} \mathbf{E}_{C_R} {}^{C_R}\mathbf{x} = 0 \quad (3.20)$$

${}^{C_V} \mathbf{E}_{C_R}$ est appelée la matrice essentielle et est construite en fonction des paramètres de transformation entre ${}^{C_R}\mathbf{x}$ et ${}^{C_{V1}\mathbf{x}}$

$${}^{C_V} \mathbf{E}_{C_R} = [{}^{C_V} \mathbf{t}_{C_R}]_{\times} {}^{C_V} \mathbf{R}_{C_R} \quad (3.21)$$

Notons que dans le cas d'une rotation pure, ${}^{C_V} \mathbf{t}_{C_R}$ est nulle. Dans ce cas, la matrice essentielle n'est pas définie. La relation avec la matrice fondamentale ${}^{C_V} \mathbf{F}_{C_R}$ se fait avec :

$${}^{C_V} \mathbf{F}_{C_R} = \mathbf{K}^T {}^{C_{V1}} \mathbf{E}_{C_R} \mathbf{K} \quad (3.22)$$

Le calcul de la matrice essentielle entre la caméra réelle et la caméra virtuelle nécessite ${}^{C_V} \mathbf{R}_{C_R}$ et de ${}^{C_V} \mathbf{t}_{C_R}$. Il est possible d'obtenir ces deux informations à partir d'informations exprimées dans un repère global.

$${}^{C_V} \mathbf{T}_{C_R} = {}^{C_V} \mathbf{T}_O {}^O \mathbf{T}_{C_R} \quad (3.23)$$

et donc

$${}^{C_V} \mathbf{R}_{C_R} = {}^{C_V} \mathbf{R}_O {}^O \mathbf{R}_{C_R} \quad (3.24)$$

$$= {}^{C_V} \mathbf{R}_O {}^{C_R} \mathbf{R}_O^T \quad (3.25)$$

On constate que la matrice essentielle ne dépend que de la position et l'orientation des caméras. D'où son utilité pour l'estimation du placement relatif des deux caméras.

Considérons l'équation $\mathbf{E} = \mathbf{K}^T \mathbf{F} \mathbf{K} = \mathbf{R}_V [\mathbf{t}_R - \mathbf{t}_V]_{\times} \mathbf{R}_R$ définissant la matrice essentielle. Les paramètres extrinsèques \mathbf{R}_R , \mathbf{R}_V , \mathbf{t}_R et \mathbf{t}_V sont définis par rapport au repère monde. Ils expriment le positionnement des deux prises d'images. Lors de l'étalonnage, on positionne la caméra réelle en position "canonique", i.e. : $\mathbf{R}_R = \mathbf{I}$ et $\mathbf{t}_R = \mathbf{0}$

L'équation devient : $\mathbf{E} = [\mathbf{t}_V]_{\times} \mathbf{R}_V$ où \mathbf{R}_V et \mathbf{t}_V expriment l'orientation et la position de la caméra virtuelle, relativement à la caméra réelle.

3.4.2.1 Détermination de la matrice essentielle réfective

3.4.2.2 Méthode basée sur la pose relative :

Soit la transformation réfective $\mathbf{T}_v = \mathbf{T} \mathbf{D}$. Cette transformation est composée de la transformation de réflexion et la transformation rigide

$$\mathbf{T}_v = \begin{bmatrix} \mathbf{R}_v & \mathbf{t}_v \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{I} - 2\mathbf{nn}^T & 2d\mathbf{n} \\ 0 & 1 \end{bmatrix} \quad (3.26)$$

Avec

$$\mathbf{R}_v = \mathbf{R}(\mathbf{I} - 2\mathbf{nn}^T) \quad (3.27)$$

$$\mathbf{t}_v = 2d\mathbf{R}\mathbf{n} + \mathbf{t} \quad (3.28)$$

La matrice essentielle réfective est définie par :

$$\mathbf{E}_r = [\mathbf{t}_v]_{\times} \mathbf{R}_v \quad (3.29)$$

où $(\mathbf{R}_v, \mathbf{t}_v)$ la transformation rigide réfective entre la caméra réelle et la caméra virtuelle.

3.4.2.3 Méthode basée sur l'équation du plan :

Soit $\mathbf{P}_R = [\mathbf{I}|0]$ et $\mathbf{P}_V = [M|m]$.

En posant

$$u_V = \mathbf{P}_V \mathbf{X} \quad (3.30)$$

et

$$u_R = \mathbf{P}_R \mathbf{X} \quad (3.31)$$

On vérifie la relation de la matrice fondamentale :

$$u_V^T \mathbf{F}_r u_R = 0 \quad (3.32)$$

Après manipulations, on obtient $\mathbf{E}_r = 2d[\mathbf{n}]_{\times}$

3.4.3 Validation de la robustesse de la matrice fondamentale réfective F_r

Il existe de nombreuses méthodes disponibles pour l'estimation de la matrice fondamentale [Hartley et Zisserman (2003)]. Nous avons adapté ces méthodes pour le cas de la géométrie épipolaire réfective. Nous avons ensuite comparé les résultats trouvés avec la méthode linéaire basée sur au moins deux paires de correspondances pour estimer la matrice fondamentale réfective \mathbf{F}_r . En outre, le fait que le système **SCP** soit bien calibré et les paramètres des plans bien calculés, rend possible le calcul de la matrice essentielle et fondamentale réfectives par deux façons différentes. La comparaison de ces méthodes avec notre algorithme de deux points montre l'importance de l'identification d'un bon modèle de matrice fondamentale réfective avant l'estimation et la vérification du système qui satisfait aussi la géométrie épipolaire réfective.

Après l'étalonnage, nous recalculons la matrice fondamentale réfective par trois méthodes en utilisant les données réelles (algorithme de 4 points, matrice essentielle réfective \mathbf{E}_r en fonction de \mathbf{R}_v et \mathbf{t}_v et \mathbf{E}_r en fonction des paramètres du plan miroir). Les résultats du tableau 3.3 sont obtenus pour quatre groupes de paires de vue. Chaque groupe contenant une vue et sa réflexion, par exemple le groupe RV1 contient la vue réelle et sa réflexion virtuelle 1, le groupe RV2 contient la vue réelle et sa réflexion virtuelle 2, le groupe V1V21 contient la vue virtuelle 1 simple et sa réflexion virtuelle 21 double et le dernier groupe V2V12 contient la vue virtuelle 2 simple et sa réflexion virtuelle 12 double. Une fois les matrices fondamentales calculées entre chaque paire de vues symétrique, nous déduisons l'auto-épipole associé à chaque matrice fondamentale. Après, nous traçons les droites épipolaires qui passent par les points d'une vue et son reflet. Nous remarquons que les épipoles sont l'intersection des droites épipolaires. Après nous comparons les résultats obtenus avec ceux de la vérité de terrain. Nous remarquons que les résultats obtenus par l'algorithme de 4 points sont les plus précis, ce qui prouve la robustesse de cette méthode proposée.

Nous utilisons les quatre groupes de la figure 3.19. L'idée est de former des paires d'images permettent de calculer tous les épipoles et valider l'hypothèse que l'épipole obtenus par la vue directe et l'épipole obtenu par la vue symétrique sont confondus dans un seul point appelé auto-épipole. Les résultats du tableau 3.4 présentent les valeurs des auto-épipoles obtenus par les données de la vérité de terrain et les auto-épipoles obtenus par les trois méthodes.

| | | M1 : Algorithme de 4 points | M2 : $\mathbf{E} = 2 \mathbf{d}[\mathbf{n}]_{\times}$ | M3 : $\mathbf{E} = \mathbf{R}[\mathbf{t}]_{\times}$ |
|-------------------|-------------|--|--|--|
| F_r entre RV1 | F_{RV1} | $\begin{bmatrix} 0 & 1.29e-03 & 2.39e-01 \\ -1.29e-03 & 0 & -9.70e-01 \\ -2.39e-01 & 9.70e-01 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 1.20e-06 & 1.84e-04 \\ -1.20e-06 & 0 & -8.16e-04 \\ -1.84e-04 & 8.16e-04 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 1.20e-06 & 1.84e-04 \\ -1.20e-06 & 0 & -8.16e-04 \\ -1.84e-04 & 8.16e-04 & 0 \end{bmatrix}$ |
| | e_{RV1} | $\begin{bmatrix} -6.80e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} -6.80e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} -6.80e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ |
| F_r entre RV2 | F_{RV2} | $\begin{bmatrix} 0 & -8.31e-04 & -1.27e-01 \\ 8.31e-04 & 0 & -9.91e-01 \\ 1.27e-01 & 9.91e-01 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & -1.20e-06 & -1.84e-04 \\ 1.20e-06 & 0 & -1.43e-03 \\ 1.84e-04 & 1.43e-03 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & -1.20e-06 & -1.84e-04 \\ 1.20e-06 & 0 & -1.43e-03 \\ 1.84e-04 & 1.43e-03 & 0 \end{bmatrix}$ |
| | e_{RV2} | $\begin{bmatrix} 1.18e+03 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 1.12e+03 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 1.12e+03 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ |
| F_r entre V1V21 | F_{V1V21} | $\begin{bmatrix} 0 & -2.18e-03 & -3.35e-01 \\ 2.18e-03 & 0 & -9.42e-01 \\ 3.35e-01 & 9.42e-01 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 2.02e-06 & 3.10e-04 \\ -2.02e-06 & 0 & 1.21e-03 \\ -3.10e-04 & -1.21e-03 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 2.02e-06 & 3.10e-04 \\ -2.02e-06 & 0 & 1.21e-03 \\ -3.10e-04 & -1.21e-03 & 0 \end{bmatrix}$ |
| | e_{V1V21} | $\begin{bmatrix} 4.27e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 4.21e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 4.21e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ |
| F_r entre V2V12 | F_{V2V12} | $\begin{bmatrix} 0 & 5.77e-03 & 8.87e-01 \\ -5.77e-03 & 0 & 4.60e-01 \\ -8.87e-01 & -4.60e-01 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 4.07e-06 & 6.29e-04 \\ -4.07e-06 & 0 & 4.12e-04 \\ -6.29e-04 & -4.12e-04 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 4.07e-06 & 6.29e-04 \\ -4.07e-06 & 0 & 4.12e-04 \\ -6.29e-04 & -4.12e-04 & 0 \end{bmatrix}$ |
| | e_{V2V12} | $\begin{bmatrix} 7.97e+01 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 8.03e+01 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 8.03e+01 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ |

Tableau 3.3 – Le calcul des quatre matrices fondamentales et quatre épipoles entre les quatre paires de vue obtenues par les 3 méthodes.

Alignement des quatre auto-épipoles :

Si e est l'auto-épipole correspondant au vecteur normal du miroir alors le couple (u_R, u_V) et e_{RV} sont colinéaires.

En effet,

$$(u_V \times e_{RV}) \cdot u_R = 0 \quad (3.33)$$

le produit vectoriel de deux vecteurs peut être exprimé en produit de la matrice anti-symétrique et un vecteur comme suit :

$$u_V^T [e_{RV}]_{\times} u_R = 0 \quad (3.34)$$

donc $F_r = [e_{RV}]_{\times}$.

Par exemple, pour l'objet réel et son reflet par le miroir1, $F_{12} \sim F_{21} \sim [e_{12}]_{\times} \sim [e_{21}]_{\times} \sim [e]_{\times}$, et l'épipole réel et virtuel coïncide en e , où e est l'auto-épipole correspondant au

| Auto-épipoles | M0 :Données réelles (GT) | M1 : Algorithme de 4 points | M2 : $\mathbf{E} = \mathbf{2 d[n]}_{\times}$ | M3 : $\mathbf{E} = \mathbf{R[t]}_{\times}$ |
|---------------|--|--|--|--|
| e_{RV1} | $\begin{bmatrix} -6.88e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} -6.80e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} -6.80e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} -6.80e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ |
| e_{RV2} | $\begin{bmatrix} 1.19e+03 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 1.18e+03 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 1.12e+03 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 1.12e+03 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ |
| e_{V1V21} | $\begin{bmatrix} 4.31e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 4.27e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 4.21e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 4.21e+02 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ |
| e_{V2V12} | $\begin{bmatrix} 7.91e+01 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 7.97e+01 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 8.03e+01 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ | $\begin{bmatrix} 8.03e+01 \\ -1.53e+02 \\ 1.00e+00 \end{bmatrix}$ |

Tableau 3.4 – Comparaison de des auto-épipoles obtenus par les données réelles avec les auto-épipoles calculés par les 3 méthodes.

vecteur normal au plan miroir. Les matrices fondamentales entre chaque paire de vues sont représentées de la façon suivante :

$$\begin{aligned} F_{12} &\sim F_{21} \sim [e1]_{\times} \\ F_{13} &\sim F_{31} \sim [e2]_{\times} \\ F_{24} &\sim F_{42} \sim [e3]_{\times} \\ F_{35} &\sim F_{53} \sim [e4]_{\times} \end{aligned}$$

et les auto-épipoles sont donnés par :

$$\begin{aligned} e_{12} &\sim e_{21} \sim e1 \\ e_{13} &\sim e_{31} \sim e2 \\ e_{24} &\sim e_{42} \sim e3 \\ e_{35} &\sim e_{53} \sim e4 \end{aligned}$$

De plus, les quatre auto-épipoles obtenus sont **alignés** (Fig.3.19).

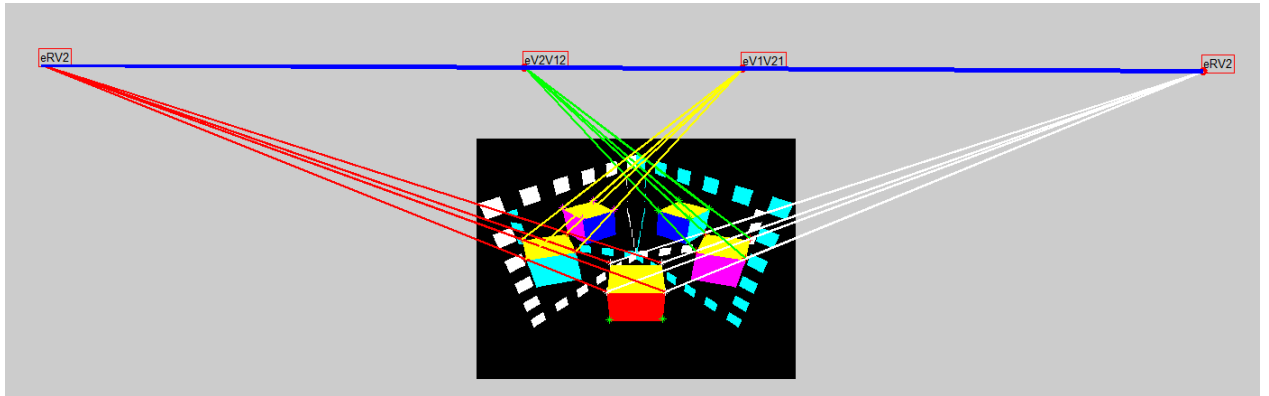


Figure 3.19 – Les droites épipolaires et les epipoles entre chaque vue et sa symétrie dans l'image synthétique.

3.5 Conclusion

L'utilisation d'un système à base de deux miroirs faisant un certain angle précis avec l'axe optique permet l'acquisition de différents points de vue à l'aide d'une seule caméra. Les deux miroirs permettent d'obtenir 4 images supplémentaires du même objet. Le système permet de s'adapter à des situations dans lesquelles on ne peut pas intégrer un système multi-caméra. Parmi ses avantages, on pourra exploiter le fait que les quatre caméras virtuelles partagent le même jeu de paramètres intrinsèques. Dans cette étude, nous avons proposé un nouveau système qui rend le calibrage plus simple et fiable. Ensuite, nous avons proposé une matrice fondamentale réflexive simple et précise. Ainsi, nous avons montré que cette matrice permet un gain en précision et robustesse par rapport aux autres méthodes. Le calcul de cette matrice permettra de valider la mise en correspondance des points, objet de notre chapitre suivant.

MISE EN CORRESPONDANCE DE POINTS DANS DES IMAGES ACQUISES PAR UN SYSTÈME CATADIOPTRIQUE PLANAIRE

Sommaire

| | | |
|-------|---|----|
| 4.1 | Introduction | 45 |
| 4.2 | Synthèse | 46 |
| 4.3 | Mise en correspondance éparsée basée AMIFT | 48 |
| 4.3.1 | Prise en compte de la transformation de réflexion | 48 |
| 4.3.2 | Prise en compte de la transformation affine | 51 |
| 4.3.3 | Réflexive-RANSAC | 55 |
| 4.4 | Validation expérimentale | 56 |
| 4.4.1 | Données d'évaluation | 56 |
| 4.4.2 | Évaluation quantitative | 56 |
| 4.5 | Conclusion | 60 |

4.1 Introduction

Le problème abordé dans ce chapitre est celui de la mise en correspondance d'images acquises par un SCP. Nous proposons dans ce chapitre de décrire ce problème, de présenter les approches les plus récentes et par la suite de présenter notre contribution pour la mise en correspondance de points pour un SCP.

La méthode que nous adoptons pour la détection et l'appariement des points caractéristiques est une nouvelle approche qui combine des invariants à l'échelle, à la rotation, aux transformations affines et à la réflexion miroir. Cette approche améliore l'appariement pour notre système de vision parce qu'elle tient compte des différents points caractéristiques entre les cinq vues de l'objet. Cette approche sera notée AMIFT (Affine-Mirror SIFT) et a été rendue plus robuste par l'élimination de faux appariements en exploitant la géométrie épipolaire étudiée dans le chapitre précédent.

Ce chapitre s'organise en quatre sections. La première propose une synthèse des variantes de SIFT pour la détection et la mise en correspondance des points d'intérêt sous plusieurs contraintes. La seconde présente une nouvelle méthode de la mise en correspondance de points d'intérêt, ainsi que leurs descripteurs. Ces descripteurs permettent une mise en correspondance tenant compte de l'effet des miroirs. Notre approche intègre également une étape supplémentaire d'élimination des faux appariements. Enfin, la dernière section décrit les expérimentations réalisées afin de valider l'approche proposée en utilisant des métriques appropriées.

4.2 Synthèse

La description des caractéristiques est une étape qui vient juste après leur détection. Pour les décrire, on considère un patch local autour des caractéristiques détectées pour construire un descripteur. Ce dernier est un vecteur caractérisant le voisinage d'un point d'intérêt. Les descripteurs caractéristiques peuvent être classés en quatre types :

- (i) Descripteurs basés sur la distribution. Ces descripteurs utilisent la représentation par histogramme pour caractériser les points d'intérêt. Dans les applications de traitement d'images, la performance de ces descripteurs est meilleure que celle des autres descripteurs.
- (ii) Descripteurs basés sur la fréquence spatiale. Ici, les points d'intérêt détectés sont décrits en considérant le contenu en fréquence de l'image.
- (iii) Descripteurs différentiels. Dans ces descripteurs, le voisinage local d'un point est approximé par le calcul de sa dérivée spatiale partielle.
- iv) Autres. Cette classification comprend des descripteurs autres que les trois types mentionnés. Cette catégorie comprend la description des caractéristiques basées sur le moment, la phase et la couleur.

L'utilisation d'un pipeline de reconstruction 3D nécessite la mise en correspondance entre toutes les paires de vues afin de pouvoir estimer les points 3D en exploitant les contraintes de la géométrie [Hartley et Zisserman (2003)]. Afin d'appliquer un tel pipeline sur une image acquise par un système SCP comme illustré dans la Fig.4.1, nous allons présenter une méthode de mise en correspondance de points adéquate à ce cas. La phase de détection et d'appariement de points caractéristiques dans l'image multi-vues représente une phase délicate. En effet, les points extraits doivent être précisément appariés pour réussir l'étape de la reconstruction 3D dense. Parmi les nombreuses méthodes qui existent dans la littérature, le couple détecteur et descripteur SIFT (Scale Invariant Feature Transform), développé par [Lowe (2004)] présente l'avantage majeur d'être à la fois invariant aux rotations et aux changements d'échelle. Sa particularité réside dans le calcul combiné des points d'intérêt et de leurs descripteurs. Les descripteurs de deux vues sont ensuite comparés pour appairer les points. Cette méthode n'est pas efficace en présence d'autres transformations. D'autres méthodes ont été développées pour rendre SIFT robuste à d'autres transformations géométriques. Morel et Al, [Morel et Yu (2009)] s'intéressent aux déformations de l'image, modélisables localement par une fonction affine en considérant la scène localement plane. La méthode ASIFT (Affine-SIFT) se base sur la modélisation de la transformation affine du plan image comme étant une transformation due à un changement de position de la caméra. Dans une optique différente, Guo et al [Guo *et al.* (2009)] ont introduit MIFT (Mirror-SIFT), qui est également une variante de SIFT, robuste aux réflexions des miroirs. La méthode MIFT repose sur la réorganisation du vecteur descripteur dans le cas d'une symétrie des vues due à l'utilisation d'un miroir. D'autres recherches se sont concentrées sur les descripteurs invariants au miroir, ISIFT [Song *et al.* (2016)] et MI-SIFT [Ma *et al.* (2010)] ont également été proposés pour résoudre ce problème. Max-SIFT [Xie *et al.* (2014)] et FIND [Guo et Cao (2010)] ont fourni un algorithme de calcul des descripteurs locaux invariants à la symétrie due à la réflexion. MBR-SIFT [Su *et al.* (2017)] fournit également des descripteurs binaires invariants à la réflexion. Les variantes comme MIFT ou ASIFT généralisent SIFT en ajoutant plus de degrés de liberté aux transformations auxquelles le descripteur est invariant à la transformation miroir pour MIFT et affine pour ASIFT. Une fois les correspondances des candidats calculées, un consensus de type RANSAC [Fischler et Bolles (1981)] exploitant la géométrie épipolaire est exécuté afin de ne conserver que les correspondances correctes (inliers), pour une reconstruction 3D robuste. Cependant, la

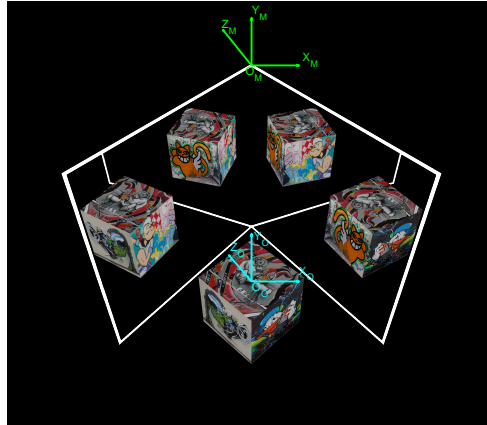


Figure 4.1 – Image capturée par un système SCP.

performance de ces correspondances basées sur les descripteurs diminue considérablement lorsqu'on est en présence à la fois des transformations affines et miroir, ce qui est le cas des systèmes SCP.

Le tableau 4.1 récapitule les performances des différents détecteurs de primitives locales. Le choix d'un détecteur de points d'intérêt repose essentiellement sur l'utilisation souhaitée. Une classification suivant le type d'invariance gérée peut également s'avérer judicieuse. Il sera en effet plus aisé de sélectionner le détecteur approprié en fonction des différentes transformations rencontrées dans l'image.

Pour conclure provisoirement sur la performance de divers descripteurs de caractéristiques

| Détecteur | Coins | Zones locales | Invariance à la rotation | Invariance à l'échelle | Invariance à la transformation affine | Invariance à la transformation de réflexion | Répétabilité | Précision de la localisation | Robustesse | Efficience |
|-----------------|-------|---------------|--------------------------|------------------------|---------------------------------------|---|--------------|------------------------------|------------|------------|
| Harris | x | | x | | | | +++ | +++ | +++ | ++ |
| Hessian | | x | x | | | | ++ | ++ | ++ | + |
| Harris-Laplace | x | x | x | x | | | +++ | +++ | ++ | + |
| Hessian-Laplace | x | x | x | x | | | +++ | +++ | +++ | + |
| DoG | x | x | x | x | | | ++ | ++ | ++ | ++ |
| GLOH | x | x | x | x | | | ++ | +++ | +++ | ++ |
| SIFT | x | x | x | x | | | +++ | +++ | +++ | ++ |
| PCA-SIFT | x | x | x | x | | | +++ | +++ | +++ | +++ |
| MIFT | x | x | x | x | | x | +++ | +++ | +++ | ++ |
| RIFT | x | x | x | | | x | +++ | +++ | +++ | ++ |
| F-SIFT | x | x | x | x | | x | +++ | +++ | ++ | + |
| MI-SIFT | x | x | x | x | | x | +++ | +++ | +++ | ++ |
| MI-HOG | x | x | x | | | x | +++ | +++ | +++ | ++ |
| Max-SIFT | x | x | x | x | | x | +++ | +++ | +++ | ++ |
| I-SIFT | x | x | x | x | | x | +++ | +++ | +++ | ++ |
| FIND | x | x | x | | | x | +++ | +++ | +++ | ++ |
| Harris-affine | x | x | x | x | x | | +++ | +++ | +++ | ++ |
| Hessian-affine | x | x | x | x | x | | +++ | +++ | +++ | ++ |
| ASIFT | x | x | x | x | x | | +++ | +++ | +++ | ++ |

Tableau 4.1 – Synthèse des détecteurs classiques

à partir de plusieurs études [Kazmi et Andersen (2015)], nous constatons que ASIFT surpasse nettement les méthodes existantes, y compris les SIFT, MSER, Harris-Affine et Hessian-Affine.

Nous remarquons également d'après l'étude que nous avons précédemment réalisée (voir tableau 4.1) que la méthode MIFT surpasse les diverses extensions de SIFT invariantes aux réflexions. Ainsi, elle engendre de correspondances par rapport aux autres. Toutes ces méthodes permettent d'étudier le cas de la symétrie par réflexion mais ne garantissent pas

la robustesse de l'invariance à la réflexion en présence de la contrainte de la transformation affine.

Pour surmonter les problèmes rencontrés par la méthode SIFT appliquée au SCP, nous proposons de :

- Combiner ASIFT et MIFT en un nouveau détecteur nommé AMIFT.
- Etudier la géométrie épipolaire pour un SCP et l'exploiter pour retenir des appariements robustes basés sur RANSAC, en tant que Reflective-RANSAC.

4.3 Mise en correspondance éparsée basée AMIFT

L'objectif est de concevoir un descripteur de caractéristiques locale qui décrit un point d'intérêt et qui doit être robuste :

- À la translation ;
- Au changement d'échelle ;
- À la rotation ;
- Aux transformations affines ;
- Aux transformations de réflexion.

Pour cela, nous allons modifier l'algorithme ASIFT en incorporant la méthode du MIFT, qui nous permet de définir la nouvelle méthode AMIFT comme illustre dans la (figure 4.2). AMIFT permet la mise en correspondance robuste aux transformations évoquées (Fig.4.3)

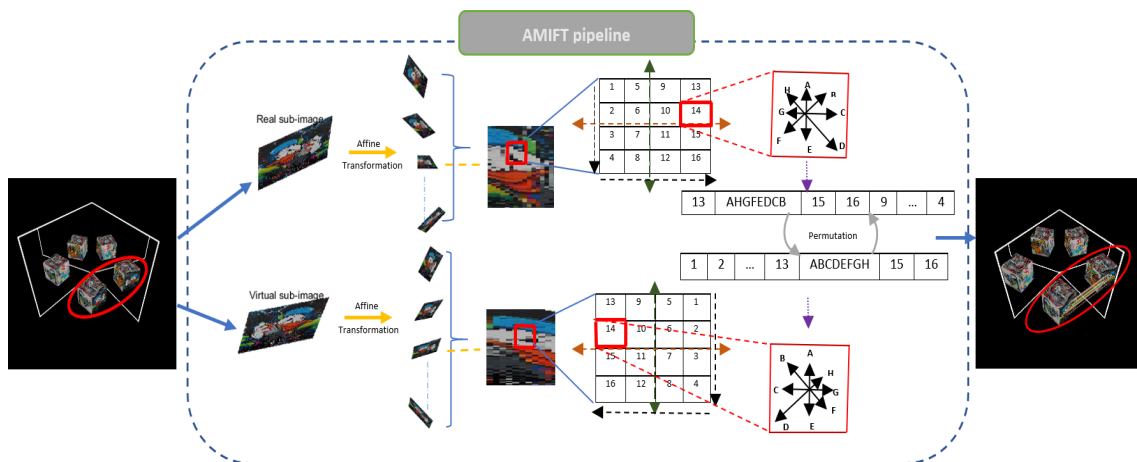


Figure 4.2 – AMIFT pipeline

4.3.1 Prise en compte de la transformation de réflexion

La réflexion est décrite par les composantes des orientations déterminées localement sur l'image à partir de la direction des gradients dans un voisinage autour d'un point d'intérêt. L'idée est d'inverser les positions des gradients par rapport à un pic correspondant à l'orientation dominante.

Une région d'intérêt est généralement divisée en petites cellules, comme illustré dans la Fig. 4.4. L'ordre de ces cellules n'est pas invariant aux réflexions. Par exemple, pour les

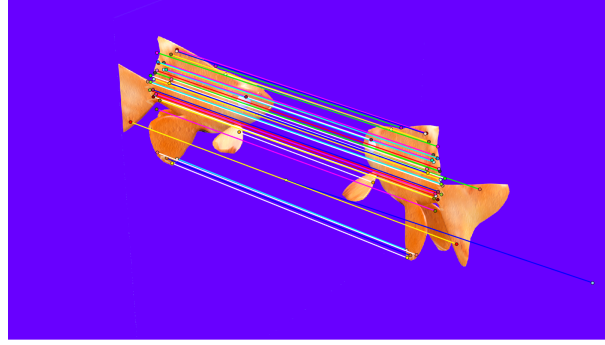


Figure 4.3 – Mise en correspondance dans le cas de la réflexion par le miroir.

régions d'intérêt rectangulaires, l'ordre des colonnes des cellules est de "droite à gauche" pour le cas de réflexion (Fig. 4.4 (b)) alors qu'il est inversé pour l'image originale (Fig. 4.4 (a)).

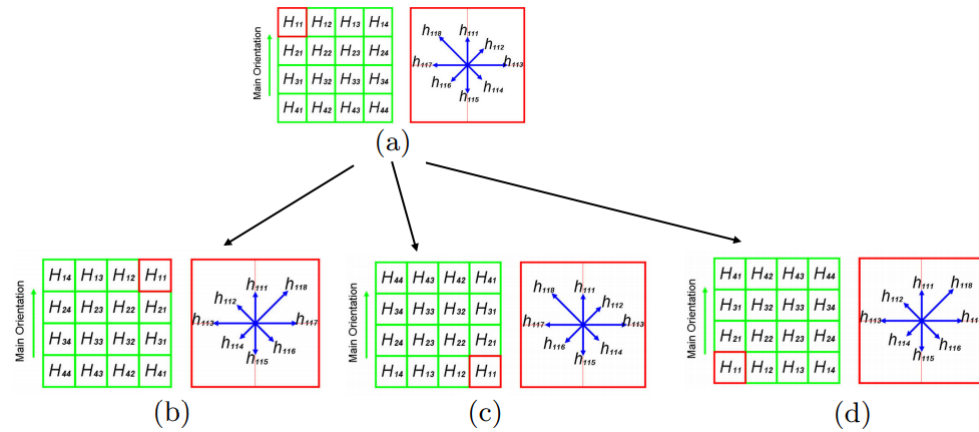


Figure 4.4 – Différentes réflexions de miroir et le concept de MIFT. Orientation dominante dans l'image originale (a). (b) réflexion horizontale. (c) réflexion verticale. (d) réflexion combinée de (b) et (c).

4.3.1.1 Ré-organisation d'un descripteur

Dans cette partie, nous appliquons le concept MIFT au descripteur SIFT. Un vecteur contenant les orientations des gradients autour du point d'intérêt permet de créer un descripteur final de taille $128 = 4 \times 4 \times 8$ dont chaque région est divisée en 4 sous-régions de taille 4×4 dans lesquelles on réorganise les 8 orientations du gradient. Le même principe est choisi pour représenter le descripteur MIFT. La figure 4.5 (a) représente un point d'intérêt avec sa région dans l'image originale. La figure 4.5 (b) représente son image réfléchi horizontalement. Les deux régions ont d'abord été réorientées suivant l'orientation dominante.

4.3.1.2 Ordre des 16 cellules

SIFT utilise un ordre fixe pour organiser les 16 cellules de la région d'intérêt après l'assignation de l'orientation dominante des 36 candidats. Notons qu'il peut y avoir plu-

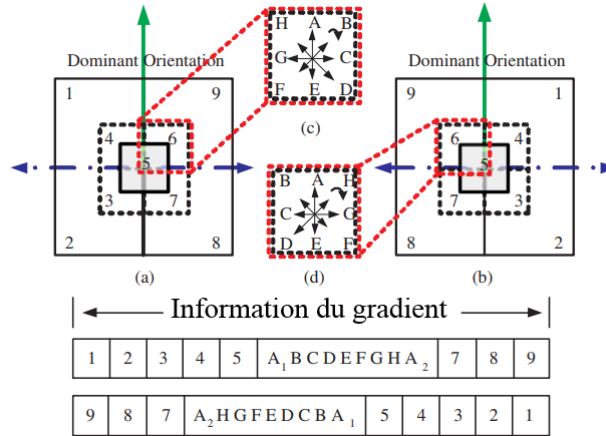


Figure 4.5 – Illustration de MIFT dans les situations avec et sans réflexion. (a) un point d'intérêt et sa région d'intérêt dans la vue directe. (b) La version inversée de (a). (c)-(d) Répartition des huit orientations dans la 6ème cellule de (a) et (b)

seurs descripteurs pour la même combinaison de la position et l'échelle, car il est possible que plusieurs pics de 36 orientations soient proches du pic le plus élevé. Comme le montre la figure 4.6 (a), SIFT pourrait adopter la stratégie d'encodage des colonnes par ordre croissant. En conséquence, les 16 cellules sont ordonnées comme dans la figure 4.6 (e). Cependant, l'ordre des colonnes est inversé après la réflexion par le miroir comme indiqué dans la Fig. 4.6 (b). La stratégie d'encodage original fixe utilisé dans SIFT arrange les 16 cellules comme Fig. 4.6 (f).

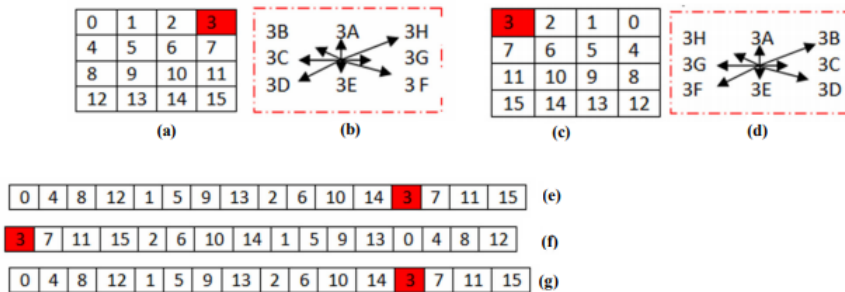


Figure 4.6 – Différentes réflexions de miroir et le concept de MIFT. Un point d'intérêt (PI) est identifié dans l'image originale (a). (b) la réflexion combinée de (a). (c) L'orientation du PI dans l'image originale. (d) l'orientation du PI dans l'image reflétée. (e) l'orientation du PI en utilisant SIFT. (f) L'orientation du PI dans l'image miroir en utilisant SIFT. (g) L'orientation du PI dans l'image miroir en utilisant MIFT.

Bien que cette stratégie de codage soit invariante à la rotation et à l'échelle et tolérante à la transformation affine, on n'obtient pas le même ordre dans la situation de la réflexion de miroir. [Guo et Cao (2012)] introduit une technique de codage adaptatif qui permet de garder l'ordre des 16 cellules dans le cas d'une réflexion par miroir dans deux directions. Intuitivement, deux amplitudes des orientations directement pointées de gauche et de droite peuvent être utilisées pour sélectionner la direction des gradients. La direction est plutôt décidée sur la base des sommations des amplitudes de toutes les orientations pointant vers la gauche et pointant vers la droite.

Pour déterminer quelle stratégie d'encodage adopter, on peut retenir le critère qui utilise

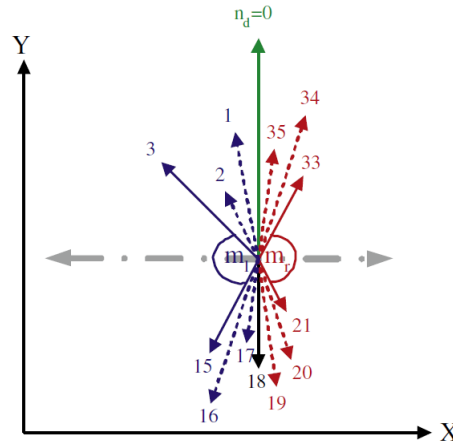


Figure 4.7 – Exemple d’illustration d’une orientation dominante, en pointant à gauche, où $N_{bin} = 36$ et $n_d = 0$, $j=3$. Les flèches pointants vers la gauche sont les orientations dirigées vers la gauche (flèches bleues) et celles dirigées vers la droite (flèches rouges) par rapport à l’orientation dominante (flèche verte n_d). Les flèches pointées à gauche et pointées à droite signifient m_l et m_r respectivement.

la relation entre les $(N_{bin}/2 + 1) - 2 \times j$ amplitudes de tous les gradients :

$$m_r = \sum_{k=j}^{N_{bin}/2-j} L(n_d - k + N_{bin}) \% N_{bin}, \quad (4.1)$$

$$m_l = \sum_{k=j}^{N_{bin}/2-j} L(n_d + k + N_{bin}) \% N_{bin}, \quad (4.2)$$

où N_{bin} est le nombre d’intervalles d’orientation (bins) qui est 36 pour MIFT, n_d est l’indice de l’orientation dominante et L_i est l’amplitude de gradient dans la direction de $\frac{i \times 2\pi}{N_{bin}}$. Selon cette mesure, nous changeons de manière adaptative la stratégie d’encodage de l’ordre fixe à celle indiquée par le gagnant entre m_l et m_r . Théoriquement, cette nouvelle approche d’encodage rend la séquence de cellules unique dans n’importe quel cas, comme indiqué dans la Fig.4.6(g). Néanmoins, m_l et m_r pourraient être proches en raison de bruits, variations d’éclairage et d’autres facteurs. Pour être plus robuste dans de telles situations, un descripteur supplémentaire est créé dans l’ordre inverse pour supporter la caractéristique, où $\min\{m_l, m_r\} > \tau \max\{m_l, m_r\}$. Dans nos expériences, τ est de 70% pour toutes les évaluations.

4.3.2 Prise en compte de la transformation affine

La mise en correspondance, dans un système SCP, permet de trouver les appariements entre une vue directe et son reflet dans le cas où l’axe de la caméra est perpendiculaire au miroir plan, mais ne permet pas de trouver les bonnes correspondances surtout en présence de déformations importantes. Pour remédier à ce problème, l’approche ASIFT combine la normalisation et la simulation de point de vue pour réaliser l’appariement en présence de forts changements de points de vue.

Simulation de points de vue [Lepetit et Fua (2006)] proposent une approche qui utilise un classifieur entraîné à reconnaître les voisinages des points d'intérêt sous différents points de vue. Les points sont détectés à l'aide d'extrema du Laplacien, via une approximation efficace en temps de calcul. Les voisinages des points utilisés pour l'apprentissage sont calculés à l'aide de transformations affines choisies. Elles s'écrivent :

$$A = R_\theta R_\theta^{-1} S R_\theta \quad (4.3)$$

La version de [Morel et Yu (2009)], profite de l'invariance de SIFT pour s'épargner une partie de la simulation de points de vue. C'est en fait un premier pas vers l'algorithme décrit ci-dessus.

Algorithme 4.1 ASIFT

Entrée: deux images I et I'.

Sortie: Ensemble d'appariements filtrés

Phase 1 : basse résolution.

- 1 Pour chaque image, générer la nouvelle collection d'images $I_{t,\phi}$ et $I'_{t',\phi'}$
- 2 Extraire les points d'intérêt SIFT dans $I_{t,\phi}$ et $I'_{t',\phi'}$
- 3 Mettre en correspondance les points d'intérêt SIFT :
pour chaque paire de l'ensemble réduit de l'étape 1, mettre en correspondance chaque paire $(I_{t,\phi}, I'_{t',\phi'})$ via une technique basée sur le rapport des distances euclidiennes entre plus proche et second plus proche voisin qui doit être inférieur à un seuil fixé (0.6 pour ASIFT).

Phase 2 : haute résolution.

- 1 Pour chaque image, générer la nouvelle collection d'images $I_{t,\phi}$ et $I'_{t',\phi'}$ uniquement pour un nombre limité de $(t, \phi), (t', \phi')$, venants des M paires de déformations donnant le plus de points d'intérêt en basse résolution.
 - 2 Extraire les points d'intérêt SIFT dans les images générées
 - 3 Mettre en correspondance les points d'intérêt SIFT :
pour chaque paire de l'ensemble réduit de l'étape 1, mettre en correspondance chaque point d'intérêt dans $I_{t,\phi}$, avec son plus proche voisin dans $I'_{t',\phi'}$, pourvu que le rapport de distance entre descripteur plus proche et second plus proche voisin soit inférieur à 0.6
 - 4 Re-projeter les points d'intérêt SIFT de $I_{t,\phi}$ et $I'_{t',\phi'}$ dans I et I'
re-projeter les points d'intérêt appariés uniquement si il n'y a pas déjà un point positionné à une distance de moins de $\sqrt{3}$ pixels.
 - 5 Retirer les appariements potentiellement erronés :
utiliser la méthode ORSA Moisan et Stival (2004) qui améliore la méthode classique RANSAC Fischler et Bolles (1981) pour vérifier la consistance avec la géométrie épipolaire uniquement
-

Adaptation d'ASIFT Elle s'appuie sur les différents outils détaillés dans la méthode SIFT. L'objectif est d'intégrer l'ensemble des transformations que peut subir l'image dans la méthode de description afin d'augmenter l'invariance. Pour ce faire, Morel et Yu (2009)

utilisent un modèle de déformation pour un point x par :

$$x' = Ax+B \text{ avec } A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \text{ et } B = \begin{bmatrix} e \\ f \end{bmatrix}$$

En représentant la matrice A comme une matrice de transformation géométrique définie par :

$$A = \lambda R(\phi) \begin{pmatrix} t & 0 \\ 0 & 1 \end{pmatrix} R(\psi) \quad (4.4)$$

avec $\lambda > 0$, $R(\psi)$ et $R(\phi)$ des matrices de rotation, $\phi \in [0, 180^\circ[$. Le paramètre $t = 1/\cos(\theta)$ est le tilt de la caméra.

Il est possible de schématiser le modèle de caméra affine par la figure. 4.8

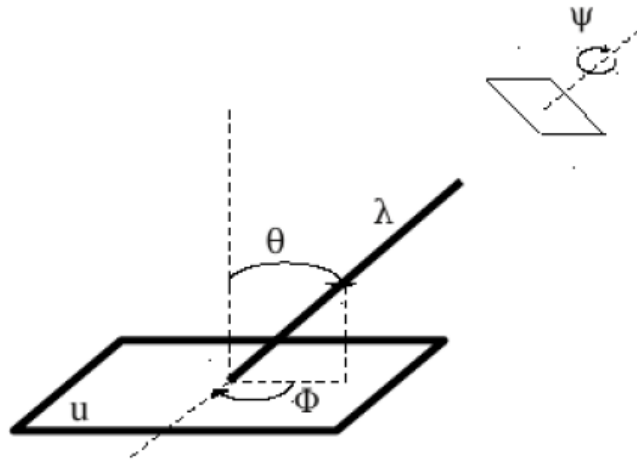


Figure 4.8 – L'interprétation géométrique de l'eq.4.4

Comme les descripteurs SIFT sont invariants aux changements d'échelle et aux rotations dans le plan image, une collection de descripteurs ASIFT invariants aux transformations affines d'une image I s'obtient en extrayant les descripteurs SIFT des images simulées par la relation :

$$I_{t,\phi} = \begin{pmatrix} t & 0 \\ 0 & 1 \end{pmatrix} R(\psi)I \quad (4.5)$$

En effet, le positionnement des points d'intérêt SIFT se veut covariant avec tout changement d'échelle et rotation $\lambda R(\psi)$ appliqué à $I_{t,\phi}$, et le descripteur associé ne change quasiment pas. La discussion présentée dans Morel et Yu (2009) prouve qu'il est suffisant de discrétiser t et ϕ tels que : $t \in \{1, \sqrt{2}, 2, 2\sqrt{2}, 4\}$ (Voir figure 4.9)

L'étape suivante consiste à mettre en correspondance les points d'intérêt ASIFT ainsi générés entre les deux images I et I' . Morel et Yu proposent une approche utilisant deux échelles. D'abord, les images simulées $I_{t,\phi}$ et $I'_{t',\phi'}$, sont générées à partir des images de taille réduite (d'un facteur 3). Ensuite les points d'intérêt SIFT extraits de chaque paire $(I_{t,\phi}, I'_{t',\phi'})$ sont mis en correspondance via l'algorithme habituel de SIFT. Le rapport des distances euclidiennes entre plus proche et second plus proche voisin doit être inférieur à un seuil fixé (0.6 pour ASIFT). Les déformations correspondant aux M paires (M typique-

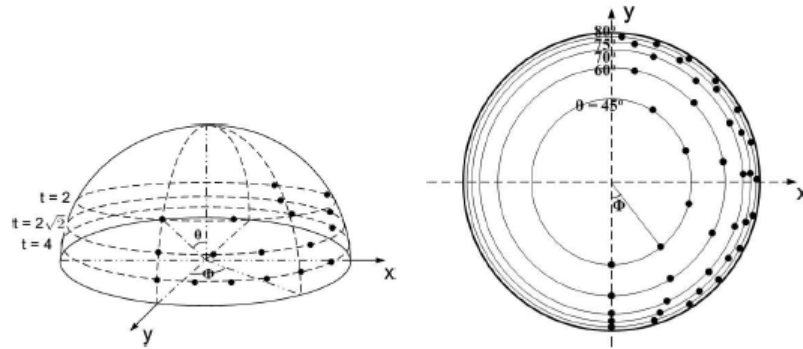


Figure 4.9 – L'échantillonnage des paramètres $\theta = \arccos 1/t$ et ϕ . Les échantillons sont présentés par les points noirs. À gauche : illustration perspective de l'observation d'une hémisphère. À droite, l'observation d'une vue du zénith de l'hémisphère. Les valeurs de θ sont indiquées sur la figure

ment entre 5 et 15) qui mènent au plus grand nombre d'appariements sont utilisées sur les images de pleine résolution I et I', qui donnent des points d'intérêt SIFT qu'on met en correspondance par le critère que l'on vient de mentionner. Remarquons que la plupart des appariements ont pour origine la même paire d'images. Les appariements obtenus sont alors re-projetés dans I et I', pourvu que les appariements déjà détectés soient éloignés d'une distance supérieure à $\sqrt{3}$. Cette stratégie est utilisée pour limiter le coût calculatoire et pour éviter les redondances entre points d'intérêt SIFT venant de différentes déformations. Une étape supplémentaire optionnelle consiste à éliminer les appariements aberrants avec RANSAC en imposant la compatibilité avec la contrainte épipolaire.

Le AMIFT s'appuie sur les différentes étapes détaillées dans la méthode ASIFT. Ce modèle permet de déterminer un certain nombre d'images clés représentant l'ensemble des transformations possibles. La figure 4.10 illustre la mise en correspondance d'une image E et son reflet ainsi que de leurs transformations.

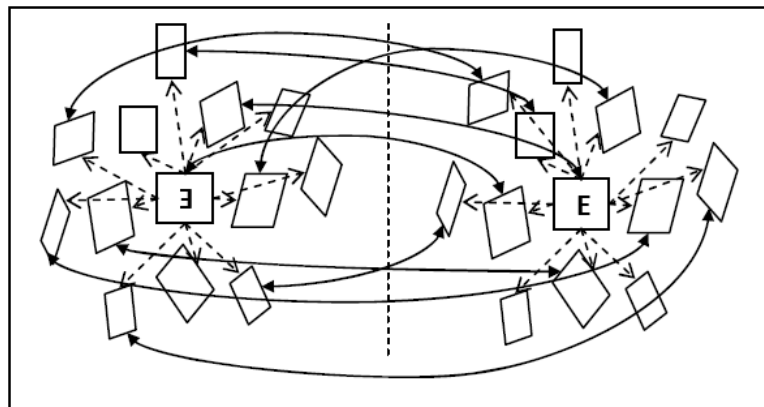


Figure 4.10 – Vue globale sur la manipulation de l'algorithme AMIFT.

Algorithme 4.2 AMIFT

Entrée: deux images I et I' et $S = \sqrt{3}$ **Sortie:** Ensemble des bonnes paires mises en correspondance

- 1 Pour chaque image, générer la nouvelle collection d'images $I_{t,\phi}$ et $I'_{t',\phi'}$ uniquement pour un nombre limité de $(t, \phi), (t', \phi')$, venant des M paires de déformations donnant le plus de points d'intérêt en basse résolution.
 - 2 Extraire les points d'intérêt des images générées
 - 3 Mettre en correspondance les points d'intérêt :
pour chaque paire de l'ensemble réduit de l'étape 1, mettre en correspondance chaque point d'intérêt dans $I_{t,\phi}$, avec son plus proche voisin dans $I'_{t',\phi'}$, pourvu que le rapport de distance entre descripteur plus proche et second plus proche voisin soit inférieur à 0.75
 - 4 Re-projeter les points d'intérêt de $I_{t,\phi}$ et $I'_{t',\phi'}$ dans I et I'
re-projeter les points d'intérêt appariés uniquement si il n'y a pas déjà un point positionné à une distance inférieure à S pixels.
 - 5 Retirer les appariements potentiellement erronés :
utiliser la méthode adaptée Réflective-RANSAC qui améliore la méthode classique RANSAC pour vérifier la consistance avec la géométrie épipolaire uniquement
-

4.3.3 Réflective-RANSAC

Afin d'éliminer d'éventuelles faux appariements de AMIFT nous avons ajouté des contraintes géométriques dans l'algorithme. Les entrées de l'algorithme Reflective-RANSAC sont l'ensemble des points AMIFT de l'image mis en correspondance avec les points de l'image reflétée. En sortie de l'algorithme, on obtient un ensemble d'inliers, c'est-à-dire un ensemble de mises en correspondance validant la matrice fondamentale F.

4.3.3.1 Estimation de la matrice fondamentale

À ce stade, nous disposons d'un jeu de plusieurs dizaines à plusieurs centaines d'appariements, que nous utilisons pour obtenir une estimation de la matrice fondamentale. L'algorithme choisi doit garder de bonnes performances malgré la présence d'appariements erronés ("outliers").

Afin d'obtenir la plus grande fiabilité, il est important d'avoir des points qui couvrent l'ensemble de l'image. La matrice fondamentale F est alors calculée en utilisant la méthode linéaire proposée de "4 points" (Chap.3), qui donne le résultat le plus fiable en l'absence d'appariement erronés.

4.3.3.2 Méthode non linéaire : Symmetric-RANSAC

L'algorithme de Symmetric-RANSAC (RANDOM SAMPLE CONSENSUS) est un algorithme robuste qui utilise au minimum 2 paires de points. Le principe de cet algorithme est de sélectionner de façon aléatoire l'ensemble des points utilisés pour le calcul de la matrice fondamentale. Plusieurs matrices fondamentales sont calculées de cette manière et on choisit celle qui maximisera le nombre de points remplissant la contrainte épipolaire.

4.4 Validation expérimentale

Pour chaque paire d'images de test (une image directe et son reflet par l'un des miroirs plan), nous avons défini manuellement des points de correspondance qui sont utilisés pour calculer une matrice fondamentale F_{ij} , qui sera comparée à celle estimée. Pour présenter les résultats et faire ressortir les paramètres importants, on utilise deux bases de données synthétiques et réelles sur lesquelles nous comparons les différents descripteurs. Dans notre cas, ce test se base sur deux métriques : la précision et le rappel. Il permet de quantifier les performances de l'estimation de correspondances et conclure si une méthode est significativement meilleure que les autres.

4.4.1 Données d'évaluation

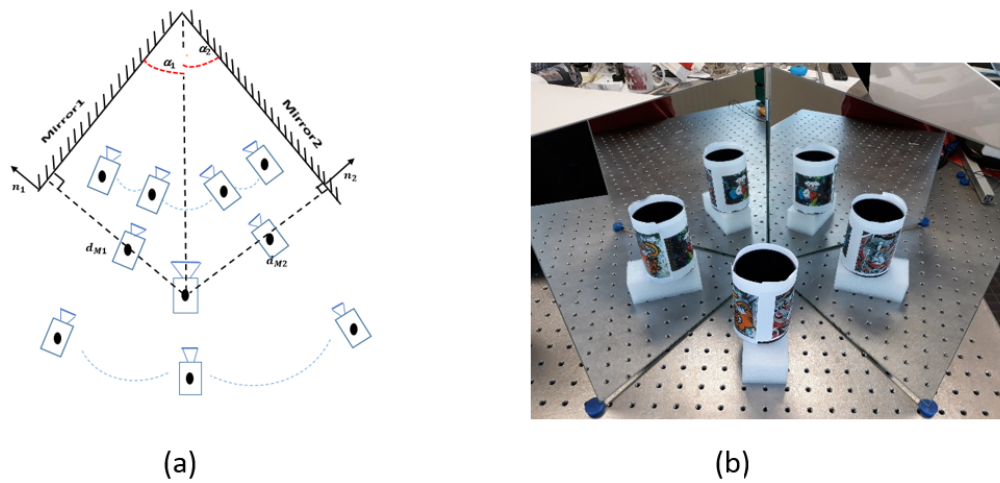


Figure 4.11 – Exemples d'images de la base de données réelles.

Nous présentons dans cette section les bases d'images des réflexions multiples catadioptriques utilisées pour effectuer nos expérimentations. Nous utilisons premièrement une base d'images synthétiques variées créées par PovRay. Nous présentons ensuite d'autres bases d'images créées par Blender annotées.

Cette base est construite à partir de 10 images. Nous divisons chaque image en quatre imagerie dont chacune contient une paire de vues réelles/virtuelles. Ces 10 images donnent 40 vues utilisées pour tester l'algorithme en gardant la même configuration et en changeant la position de la caméra comme illustré dans la figure 4.11(a). Pour mesurer le taux de rappel et la précision, nous effectuons des tests sur une base de données de 40 imagerie (voir Fig.4.12).

4.4.2 Evaluation quantitative

4.4.2.1 Métrique

Cette partie explique les choix qui ont été faits durant cette thèse pour analyser et mesurer les performances des détecteurs. Nous présentons d'abord les courbes précision-rappel, puis les données employées tout au long de ce travail.

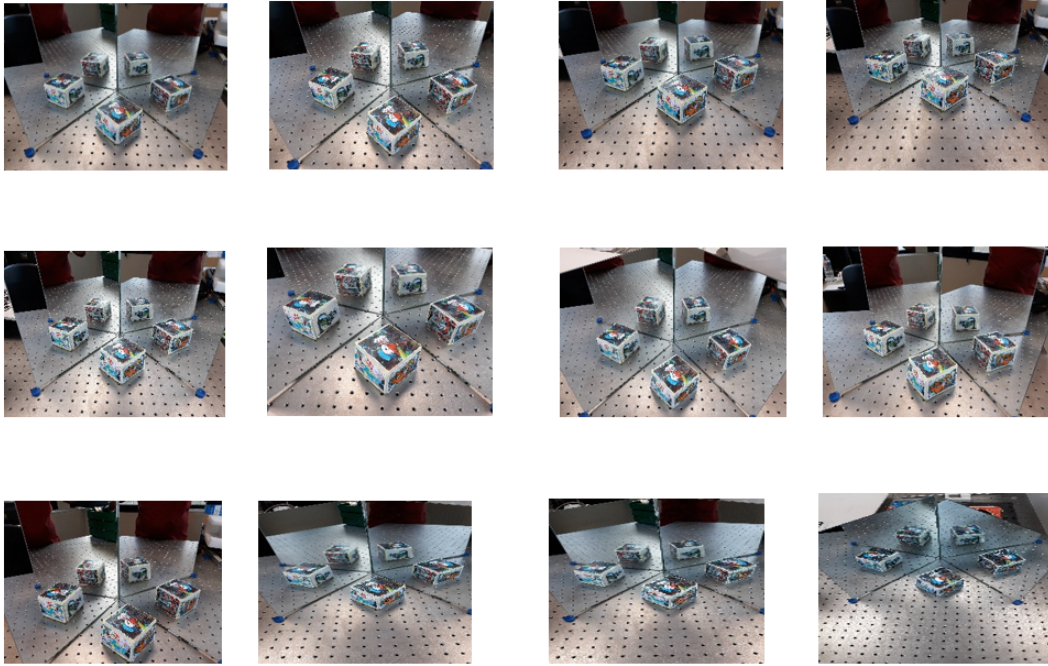


Figure 4.12 – Exemples d’images de la base de données réelles.

Idéalement, un bon système donne les meilleurs taux de précision et de rappel en même temps. En effet, un système qui aurait 100% pour la précision et pour le rappel signifie qu’il trouve toutes les correspondances (rappel) avec les points corrects (précision). Cela signifie que le descripteur discerne parfaitement les points valides des points erronés.

Les courbes précision-rappel La qualité des points de correspondances est généralement évaluée dans nos travaux par la courbe donnant la précision en fonction du rappel. La précision est définie comme étant la proportion de correspondances valides parmi toutes les correspondances établies, soit :

$$\text{précision} = \frac{\#\text{corr. valides établies}}{\#\text{corr. établies}}$$

Où # désigne le cardinal d’un ensemble.

Le rappel est défini comme étant le nombre de correspondances valides trouvées divisé par le nombre total de correspondances valides possibles, c’est-à-dire :

$$\text{Rappel} = \frac{\#\text{corr. valides établies}}{\#\text{corr. valides possibles}}$$

4.4.2.2 Comparaison d’AMIFT avec les descripteurs existants :

Nous nous sommes comparés en terme de courbes rappel-précision (RP) (Figure 1.20) avec les méthodes de détection des points d’intérêts à base de SIFT les plus utilisées dans l’état de l’art.

Pour SIFT, nous avons utilisé la méthode de RANSAC standard avec une homographie et exploité les points d'intérêts pour un appariement rapide basé sur le ratio des distances du premier au second plus proche voisin inférieur à 0.8. Cette homographie nécessite au moins 4 points pour calculer une pose, avec un choix du ratio d'appariements corrects dont la valeur est fixée à 0.25.

Pour ASIFT, nous avons utilisé la méthode ORSA basée sur RANSAC standard pour éliminer les faux appariements (outliers).

Pour MIFT et AMIFT, nous avons utilisé notre méthode de RANSAC symétrique basée sur la matrice fondamentale symétrique entre une vue et son reflet. Les points d'intérêt exploités pour un appariement rapide (basé sur le ratio des distances du premier au second plus proche voisin inférieur à un seuil égal à 0.8). Cette matrice fondamentale symétrique nécessite au moins 2 points pour calculer une pose, avec un choix précis du ratio (le vrai ratio étant généralement plus petit) afin de ne garder que les appariements corrects.

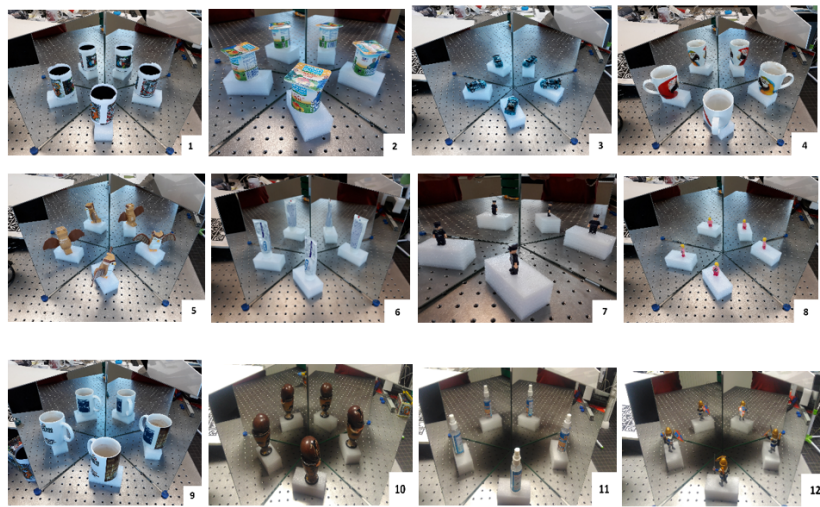


Figure 4.13 – Exemples d'images de la base de données réelles des géométries différentes.

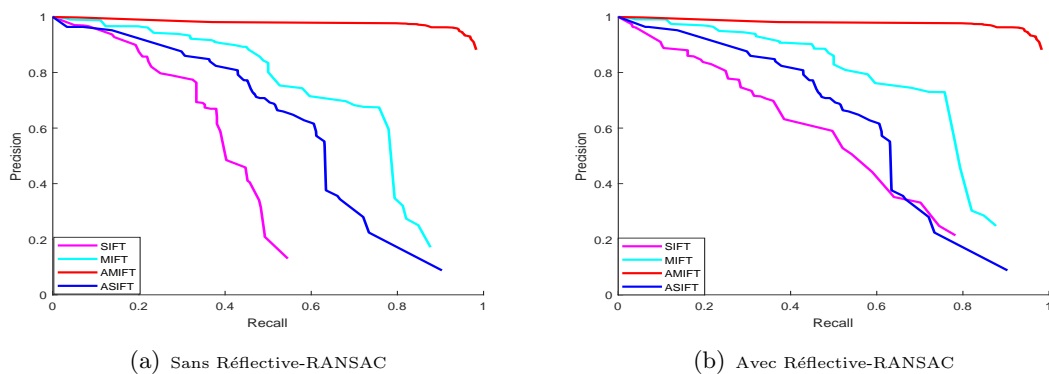


Figure 4.14 – Courbes rappel-précision pour tous les descripteurs avec et sans l'utilisation de Réflective-RANSAC

La figure 4.14 présente des exemples de courbes, pour quatre types de descripteurs calculés sur les vues symétriques. Ce graphique montre le rappel en fonction de la précision, Le descripteur le plus performant dans ce cas est donc celui correspondant à la courbe rouge. Il faut préciser que tracer ces courbes implique que les vraies correspondances entre les

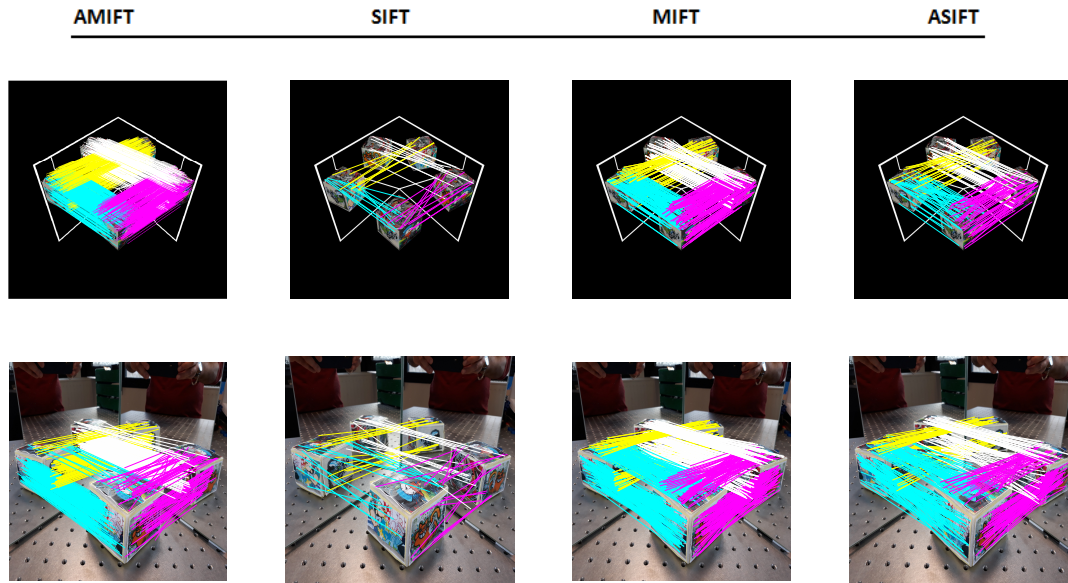


Figure 4.15 – Les correspondances obtenues sur les bases de données synthétique et réelle en utilisant les différentes méthodes.

vues symétriques soient connues.

La figure 4.15 présente les résultats obtenus sur deux images RMC choisies de la base de données synthétique et réelle. Nous avons appliqué les méthodes AMIFT, MIFT, ASIFT et SIFT sur les deux images RMC de la figure 4.15. Nous avons constaté que notre nouvelle approche de mise en correspondance trouve un nombre de correspondances plus important par rapport aux autres méthodes (figure 4.16). Cette approche est robuste et efficace, et ce même dans des cas où la transformation géométrique entre les différentes vues est importante.

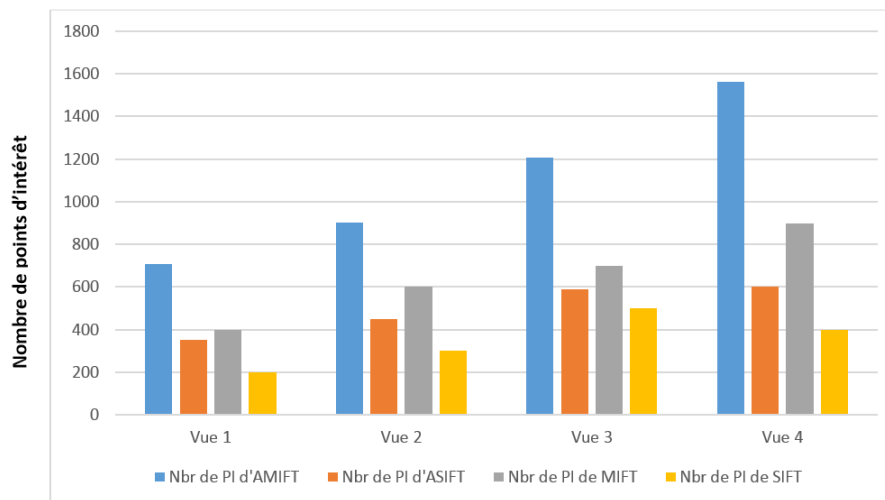


Figure 4.16 – Quatre résultats d'appariement différents d'une image RMC avec l'utilisation des méthodes AMIFT, ASIFT, MIFT et SIFT .

4.5 Conclusion

Nous avons présenté dans ce chapitre une synthèse des variantes de SIFT. Le choix du descripteur le plus efficace ou le plus adapté reste délicat. Le descripteur AMIFT offre de meilleures performances, malgré sa dimensionnalité, et le temps de calcul qui est coûteux par rapport à MIFT. En plus, nous avons présenté une nouvelle méthode de mise en correspondance, qui donne déjà de bons résultats dans de nombreux cas où la réflexion par miroir intervient sous une déformation importante dans le cas des images RMC.

MISE EN CORRESPONDANCE DE POINTS DANS DES IMAGES ACQUISES PAR UN SCP BASÉE HOMOGRAPHIES LOCALES

Sommaire

| | | |
|-------|--|-----------|
| 5.1 | Introduction | 61 |
| 5.2 | Modélisation | 62 |
| 5.3 | Méthode proposée pour la recherche des pixels correspondants | 67 |
| 5.3.1 | Restriction aux pixels de fortes variations de contraste de la vue directe | 67 |
| 5.3.2 | Paramètres à estimer | 67 |
| 5.3.3 | Estimation robuste de l'homographie | 68 |
| 5.4 | Validation et évaluation de la méthode proposée | 73 |
| 5.4.1 | Stabilité de l'algorithme | 73 |
| 5.4.2 | Critère d'évaluation existants | 74 |
| 5.4.3 | Résultats et apports de notre algorithme | 75 |
| 5.5 | Conclusion | 84 |

5.1 Introduction

La recherche des primitives qui vérifient les contraintes apportées par les miroirs plans entre une vue directe et les inter-réflexions décrite dans la section précédente a pour objectif de réaliser une mise en correspondance éparsée. Le descripteur AMIFT exposé dans le chapitre précédent, répond à cet objectif et fournit des primitives fiables avec une meilleure invariance à la réflexion du miroir. Mais pour tendre vers une reconstruction dense, il faut permettre l'appariement de la totalité des pixels. Pour cela, nous cherchons des transformations locales du voisinage (patch) de chaque pixel. La principale difficulté du problème est d'estimer la transformation projective qui permet de passer du patch vers son similaire réfléchi. Pour cela, nous utilisons l'homographie induite par le plan supposé porté par un patch. Pour le choix de la fonction de coût, nous nous appuyons sur l'utilisation de la SAD (Somme des différences absolues) des intensités des pixels des patches. La minimisation de cette fonction de coût se fait par la méthode d'optimisation Branch-and-Bound et permet d'estimer l'homographie.

L'image de réflexion multiple catadioptrique RMC est segmentée manuellement en quatre groupes issus des réflexions simples et doubles (figure 5.1) et la mise en correspondance suit les étapes suivantes :

1. Considérer tous les pixels de la vue directe.
2. Trouver l'ensemble des correspondances dans chaque groupe contenant une vue et sa réflexion.

3. Estimer l'homographie \mathbf{H} en minimisant la somme des différences absolues (SAD) des patches construits autour des points d'intérêt.
4. Projeter les points sur l'image RMC à l'aide des homographies estimées (figure 5.3).

À noter que pour assurer, dans un premier temps, une SAD suffisamment variable localement, un sous-ensemble des pixels pour lesquels la réponse de Harris est localement maximale sera considéré.

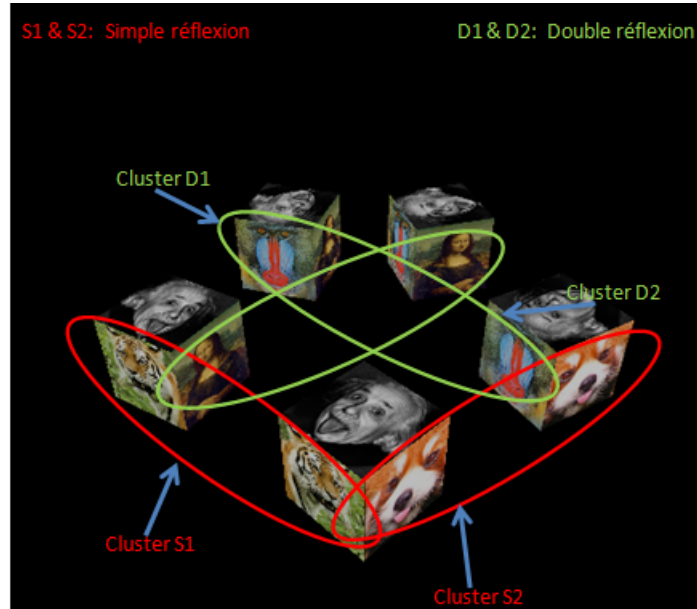


Figure 5.1 – Illustration de la mise en correspondance entre les clusters issus de réflexions simple et double.

5.2 Modélisation

Nous cherchons la fonction \mathbf{H} qui transforme le point $x_r(x_r, y_r, 1)$ de la vue directe de X sur I_R en $x_v(x_v, y_v, 1)$, sa projection sur l'image derrière le miroir I_V . L'homographie est une transformation projective entre deux patches dans l'image d'un plan [Hartley et Zisserman (2003), p41] :

$$\lambda \begin{bmatrix} x_v \\ y_v \\ 1 \end{bmatrix} = \mathbf{H}(\theta) \cdot \begin{bmatrix} x_r \\ y_r \\ 1 \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix} \begin{bmatrix} x_r \\ y_r \\ 1 \end{bmatrix}$$

avec $\theta = [h_1, \dots, h_9]^T$

L'homographie dépend de 9 paramètres, mais ils sont définis à un facteur près. Le nombre de degrés de liberté est donc 8. Nous cherchons à estimer ces 8 paramètres. L'homographie qui relie les coordonnées x_r de chaque point de l'image directe de l'objet

à ses coordonnées x_v dans l'image virtuelle est notée :

$$\lambda \begin{bmatrix} x_v \\ y_v \\ 1 \end{bmatrix} = {}^{C_V} \mathbf{H}_{C_R} \begin{bmatrix} x_r \\ y_r \\ 1 \end{bmatrix} \quad (5.1)$$

où :

$${}^{C_V} \mathbf{H}_{C_R} = \mathbf{K} ({}^{C_V} \mathbf{R}_{C_R} - \frac{{}^{C_V} \mathbf{T}_{C_R} \mathbf{n}_r}{d_r}) \mathbf{K}^{-1} \quad (5.2)$$

avec

- ▶ ${}^{C_V} \mathbf{R}_{C_R} = \mathbf{I} - 2\mathbf{n}_m \mathbf{n}_m^T$ la matrice de rotation et ${}^{C_V} \mathbf{T}_{C_R} = 2d_m \mathbf{n}_m$ le vecteur de translation, dans le cas du SCP
- ▶ d_r la distance algébrique du centre de la caméra réelle au plan réel ;
- ▶ \mathbf{n}_r la normale au plan réel exprimée dans le repère de la caméra réelle
- ▶ \mathbf{K} la matrice des paramètres intrinsèques des caméras (supposée identique pour les deux vues).

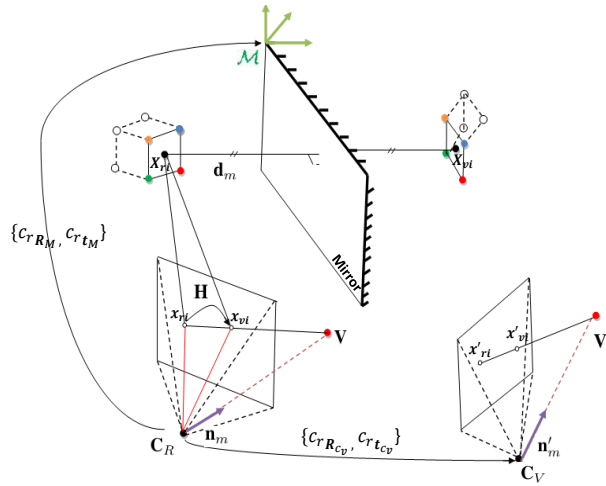


Figure 5.2 – Homographie induite par la surface plane du cube réel.

La décomposition est unique pour des valeurs singulières de la matrice d'homographie qui n'ont pas une multiplicité égale à 1 selon [Hartley et Zisserman (2003)], mais dépendante du plan observé dominant de l'homographie (figure 5.2).

En ce qui nous concerne, nous cherchons à estimer la distance et l'orientation de la normale au plan local concerné. On suppose que la caméra est calibrée et que par conséquent les paramètres intrinsèques sont connus.

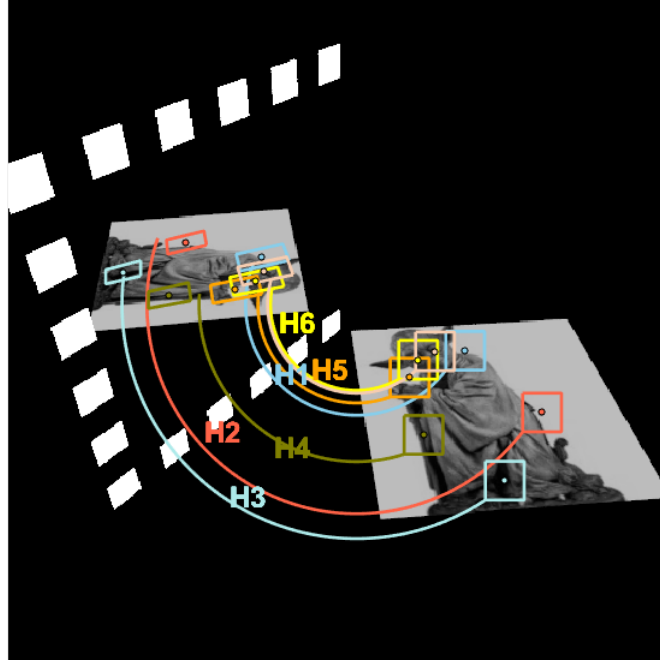


Figure 5.3 – Une homographie entre le patch réel et virtuel par un miroir plan.

La formule (5.2) s'applique lorsque le centre de la première caméra est considéré comme l'origine du repère. Lorsque le repère monde n'est pas confondu à celui de la première caméra, il faut tenir compte de la transformation rigide reliant les deux repères.

Soient deux caméras représentées par leur matrice de projection dans le repère monde (celui de la caméra réelle) :

$$P_{C_R} = K {}^{C_R}R_w [I_{3 \times 3} | -{}^{C_R}T_w] \quad (5.3)$$

$$P_{C_V} = K D {}^{C_R}R_w [I_{3 \times 3} | -{}^{C_R}T_w] \quad (5.4)$$

$$= K [R_{C_V} | T_{C_V}] \quad (5.5)$$

où $R_{C_V} = R_{C_R}(\mathbf{I} - 2\mathbf{n}_m\mathbf{n}_m^T)$ et $T_{C_V} = 2d_m\mathbf{n}_m - (\mathbf{I} - 2\mathbf{n}_m\mathbf{n}_m^T)T_{C_R}$.

Supposons que les points 3D X_w sont connus dans le repère monde. On peut calculer les coordonnées du point dans l'image réelle \mathbf{X}_r en utilisant les équations de projection (5.6) :

$$\begin{cases} x_r = X_c/Z_c \\ y_r = Y_c/Z_c \end{cases} \quad (5.6)$$

L'équation de reprojection est définie sous la forme suivante :

$$X_w(\lambda) = \lambda P_{C_R}^+ \mathbf{X}_r + \mathbf{C} \quad (5.7)$$

$P_{C_R}^+$ désigne la pseudo-inverse de la matrice de projection P_{C_R} . λ indique la position des points 3-D le long de leurs rayons de projection, et \mathbf{C} sont les coordonnées du centre de la caméra exprimé dans le repère monde.

Cela forme une infinité de solutions variant en fonction de λ . En effet, tous les points situés sur une même droite passant par le centre de la caméra se projettent au même endroit dans l'image. En l'absence d'autres informations, on ne pourrait donc pas connaître complètement les coordonnées du point X_w . Cependant, on sait que le point X_w appartient au plan Π_r . En utilisant cet a priori, la position du point devient unique. Nous pouvons

donc trouver la valeur de λ , en remplaçant X_w dans l'équation du plan par (eq 5.7) :

$$0 = \Pi_r^T X_w \quad (5.8)$$

$$0 = [\mathbf{n}_r^T \quad -d_r] \begin{bmatrix} R_{C_R} & T_{C_R} \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X_r \\ \lambda \end{bmatrix} \quad (5.9)$$

$$\lambda d_r = \mathbf{n}_r^T R_{C_R}^T x_r + \lambda \mathbf{n}_r^T T_{C_R} \quad (5.10)$$

$$\lambda = \frac{\mathbf{n}_r^T R_{C_R}^T x_r}{d - \mathbf{n}_r^T T_{C_R}} \quad (5.11)$$

$$\lambda = \frac{\mathbf{n}_r^T R_{C_R}^T x_r}{\mathbf{n}_r^T X_w - \mathbf{n}_r^T T_{C_R}} \quad (5.12)$$

Nous avons alors une solution unique pour les coordonnées X_w

$$\lambda X_w = \begin{bmatrix} R_{C_R} & T_{C_R} \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} x_r \\ \frac{\mathbf{n}_r^T R_{C_R}^T x_r}{\mathbf{n}_r^T X_w - \mathbf{n}_r^T T_{C_R}} \end{bmatrix} \quad (5.13)$$

Nous pouvons écrire l'équation de projection dans l'image virtuelle sous la forme :

$$x_v = P_{C_V} X_w \quad (5.14)$$

$$= [R_{C_V} \quad -R_{C_V} T_{C_V}] \begin{bmatrix} R_{C_R} & T_{C_R} \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X_r \\ \lambda \end{bmatrix} \quad (5.15)$$

$$= [R_{C_V} R_{C_R}^T + R_{C_V} (T_{C_R} - T_{C_V})] \begin{bmatrix} X_r \\ \frac{\mathbf{n}_r^T R_{C_R}^T X_r}{\mathbf{n}_r^T X_w - \mathbf{n}_r^T T_{C_R}} \end{bmatrix} \quad (5.16)$$

$$X_v = (R_{C_V} R_{C_R}^T + R_{C_V} (T_{C_R} - T_{C_V}) \frac{\mathbf{n}_r^T R_{C_R}^T X_r}{\mathbf{n}_r^T X_w - \mathbf{n}_r^T T_{C_R}}) X_r \quad (5.17)$$

Nous obtenons ainsi la relation entre les coordonnées 2D du point réel et le point virtuel dans le plan image de la caméra réelle. Pour avoir la relation directement depuis les coordonnées en pixel, on utilise la matrice de paramètres intrinsèques \mathbf{K} .

$$x_v = K (R_{C_V} R_{C_R}^T + R_{C_V} (T_{C_R} - T_{C_V}) \frac{\mathbf{n}_r^T R_{C_R}^T x_r}{\mathbf{n}_r^T X_w - \mathbf{n}_r^T T_{C_R}}) K^{-1} x_r \quad (5.18)$$

Nous déduisons du résultat 5.18 la valeur de l'homographie reliant x_r et x_v

$${}^{C_V} H_{C_R} = K (R_{C_V} R_{C_R}^T + R_{C_V} (T_{C_R} - T_{C_V}) \frac{\mathbf{n}_r^T R_{C_R}^T}{\mathbf{n}_r^T X_w - \mathbf{n}_r^T T_{C_R}}) K^{-1} \quad (5.19)$$

Nous pouvons remarquer que bien que X_w fait partie de l'équation, l'homographie reste la même pour tout point du plan Π_r . En effet tout point X'_w appartenant à Π_r mais différent de X_w pourrait s'écrire

$$X'_w = X_w + u_{\Pi_r} \quad (5.20)$$

avec u_{Π_r} un vecteur du plan Π_r . Cependant dans l'équation de l'homographie, X_w n'apparaît qu'à travers le calcul de la valeur $\mathbf{n}_r^T X_w$. Si on applique le calcul en X'_w , on obtient :

$$\begin{aligned} \mathbf{n}_r^T X'_w &= \mathbf{n}_r^T (X_w + u_{\Pi_r}) \\ \mathbf{n}_r^T X'_w &= \mathbf{n}_r^T X_w + \mathbf{n}_r^T u_{\Pi_r} \end{aligned}$$

$$\mathbf{n}_r^T X'_w = \mathbf{n}_r^T X_w \text{ car } u_{\Pi_r} \text{ appartient au plan donc } \mathbf{n}_r^T u_{\Pi_r} = 0$$

Donc l'équation de l'homographie est bien la même pour tous les points du plan Π_r .

La nouvelle formule de l'homographie reliant une paire de correspondance réel-virtuel X_R et X_V dans le cas où la caméra réelle est située dans une position quelconque est la suivante :

$${}^{C_V}H_{C_R} = K(R_{C_V}R_{C_R}^T + R_{C_V}(T_{C_R} - T_{C_V})\frac{n_r^T R_{C_R}^T}{N^T X_w - n_r^T T_{C_R}})K^{-1} \quad (5.21)$$

avec

- R_{C_R}, T_{C_R} (resp. R_{C_V}, T_{C_V}) la matrice de rotation et le vecteur de translation de la caméra réelle (resp. caméra virtuelle).
- n_r la normale au plan exprimée dans le repère monde.

Décomposition des homographies

Nous allons décomposer les éléments de l'homographie en fonction des paramètres de la caméra, du plan miroir et du plan observé afin d'identifier les paramètres du plan observé dans chaque élément de l'homographie. Cette dernière est définie par sept paramètres :

$$\mathbf{K} = \begin{bmatrix} \alpha & 0 & u_0 \\ 0 & -\alpha & v_0 \\ 0 & 0 & 1 \end{bmatrix} \Rightarrow \mathbf{K}^{-1} = \begin{bmatrix} \frac{1}{\alpha} & 0 & -\frac{u_0}{\alpha} \\ 0 & -\frac{1}{\alpha} & \frac{v_0}{\alpha} \\ 0 & 0 & 1 \end{bmatrix} = \frac{1}{\alpha} \begin{bmatrix} 1 & 0 & -u_0 \\ 0 & -1 & v_0 \\ 0 & 0 & \alpha \end{bmatrix}$$

$$\mathbf{R} = \mathbf{I} - 2\mathbf{n}_m \mathbf{n}_m^T = \begin{bmatrix} 1 - 2n_{mx}^2 & -2n_{mx}n_{my} & -2n_{mx}n_{mz} \\ -2n_{mx}n_{my} & 1 - 2n_{my}^2 & -2n_{my}n_{mz} \\ -2n_{mx}n_{mz} & -2n_{my}n_{mz} & 1 - 2n_{mz}^2 \end{bmatrix}$$

et

$$\mathbf{T} = 2d_m \mathbf{n}_m$$

$$\Rightarrow \frac{\mathbf{T} \mathbf{n}_r^T}{d_r} = 2 \frac{d_m}{d_r} \mathbf{n}_m \mathbf{n}_r^T = 2 \frac{d_m}{d_r} \begin{bmatrix} n_{mx}n_{rx} & n_{mx}n_{ry} & n_{mx}n_{rz} \\ n_{my}n_{rx} & n_{my}n_{ry} & n_{my}n_{rz} \\ n_{mz}n_{rx} & n_{mz}n_{ry} & n_{mz}n_{rz} \end{bmatrix}$$

En effet,

$$\mathbf{H} = \begin{bmatrix} \alpha & 0 & u_0 \\ 0 & -\alpha & v_0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 - 2n_{mx}^2 - 2n_{mx}n_{rx} \frac{d_m}{d_r} & -2n_{mx}n_{my} - 2 \frac{d_m}{d_r} n_{mx}n_{ry} & -2n_{mx}n_{mz} - 2 \frac{d_m}{d_r} n_{mx}n_{rz} \\ -2n_{mx}n_{my} - 2 \frac{d_m}{d_r} n_{my}n_{rx} & 1 - 2n_{my}^2 - 2 \frac{d_m}{d_r} n_{my}n_{ry} & -2n_{my}n_{mz} - 2 \frac{d_m}{d_r} n_{my}n_{rz} \\ -2n_{mx}n_{mz} - 2 \frac{d_m}{d_r} n_{mz}n_{rx} & -2n_{my}n_{mz} - 2 \frac{d_m}{d_r} n_{mz}n_{ry} & 1 - 2n_{mz}^2 - 2 \frac{d_m}{d_r} n_{mz}n_{rz} \end{bmatrix} \begin{bmatrix} \frac{1}{\alpha} & 0 & -\frac{u_0}{\alpha} \\ 0 & -\frac{1}{\alpha} & \frac{v_0}{\alpha} \\ 0 & 0 & 1 \end{bmatrix}$$

$$\text{On peut déterminer les composantes de la matrice d'homographie } \mathbf{H} = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix}$$

par une identification.

5.3 Méthode proposée pour la recherche des pixels correspondants

5.3.1 Restriction aux pixels de fortes variations de contraste de la vue directe

Pour identifier les pixels autour desquels le contraste varie fortement, le détecteur de Harris et Stephens (1988) reste une méthode efficace. Nous appliquons ce détecteur sur la vue directe de l'objet réel dans l'image RMC pour obtenir un ensemble de points d'intérêt dits réels. L'idée est de chercher à partir d'un patch spécifique, engendrant un point d'intérêt dans l'objet réel, son correspondant réfléchi par le miroir planaire. Par l'utilisation d'une méthode d'optimisation, nous cherchons à trouver la meilleure homographie via l'exploration de l'espace des homographies. La restriction à ces points d'intérêt permet de se placer dans les meilleures conditions pour la minimisation de la SAD de patches.

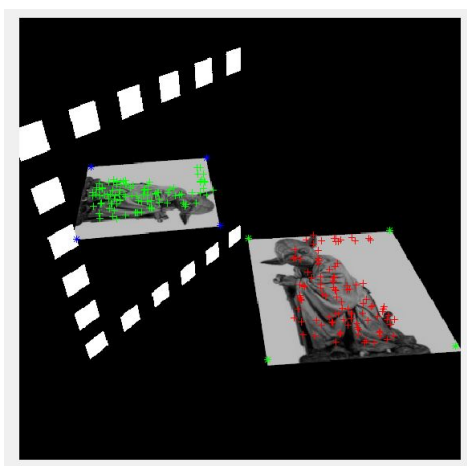


Figure 5.4 – Extraction des points d'intérêts

5.3.2 Paramètres à estimer

Paramétrisation de la normale :

Pour pouvoir définir les bornes des composantes de la normale, nous représentons la normale par deux angles : l'azimuth et l'élévation. Cela permet de déterminer d'une part les bornes de deux nouveaux paramètres et d'autre part de réduire le nombre de paramètres à estimer.

Contrainte sur la distance au plan local au pixel d'intérêt :

Nous obtenons les valeurs d_{min} et d_{max} correspondant respectivement à la distance la plus petite et la plus grande entre un point de l'objet et le centre de la caméra réelle. Ensuite, nous faisons varier le paramètre de la distance d_r entre d_{min} et d_{max} . Cela permet de diminuer le temps de calcul et de générer un espace de recherche raisonnable.

5.3.3 Estimation robuste de l'homographie

Le principe de l'approche proposée est de trouver la transformation de chaque patch de l'objet observé vers des patches de l'objet réfléchi par un miroir plan, en utilisant une méthode d'optimisation. L'observation clé est que le nombre d'homographies possibles qui devrait être évalué est limité par la variation totale du patch réel. Étant donné un paramètre de précision désiré δ , nous construisons un espace de recherche des homographies, appelé réseau, en vérifiant ce critère de précision. Pour chaque homographie dans le réseau, nous estimons l'erreur SAD en utilisant un échantillonnage aléatoire. Plus δ est fin, plus la taille du réseau est importante. Nous commençons par un réseau éparsé, puis nous éliminons toutes les homographies dans le réseau dont les erreurs dépassent un seuil donné. Pour les autres, nous augmentons le taux d'échantillonnage. La possibilité de choisir un paramètre d'échantillonnage avec une précision attendue est particulièrement intéressante pour avoir une bonne projection et dans un temps de calcul raisonnable. Cette approche nous permet non seulement de calculer un coût entre patches, mais également de déterminer la meilleure homographie dans un espace de recherche très vaste comme illustré dans la figure 5.5.

Après avoir construit le réseau des homographies, il ne reste plus qu'à chercher l'homogra-

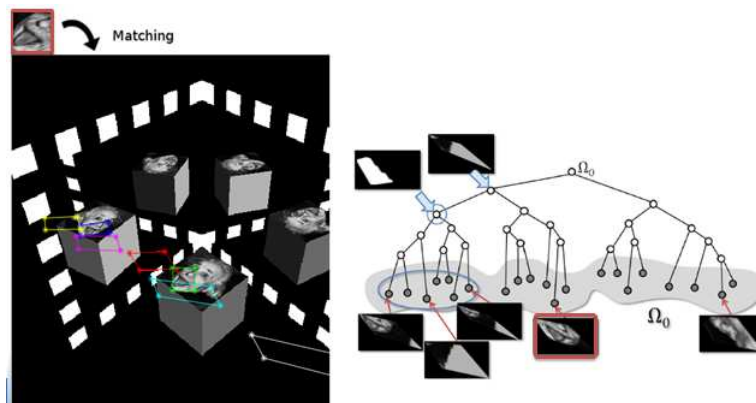


Figure 5.5 – Représentation graphique de l'ensemble de patches à partir des homographies potentielles.

phie optimale. Cette homographie tient compte de plusieurs transformations incluant la transformation de réflexion entre la vue réelle et la vue virtuelle. Il est possible d'estimer cette homographie si on connaît la pose des caméras et la position du plan miroir, et on considère l'objet étudié comme une surface planaire (au moins localement à chaque pixel). Nous prenons en considération plusieurs hypothèses :

- 1) Les deux images sont en niveaux de gris et les valeurs de chaque pixel sont comprises entre 0 et 1.
- 2) Nous utilisons la fonction de coût SAD qui dépend de la déformation du patch.
- 3) Nous appliquons un lissage sur le patch de référence pour éviter l'influence du bruit sur les résultats obtenus.
- 4) Nous adaptons la forme de la fenêtre.
- 5) Nous utilisons une interpolation lors du calcul de la SAD.

5.3.3.1 Définition du critère de correspondance

La bonne estimation des paramètres de l'homographie est nécessaire pour avoir une précision accrue lors de la phase de mise en correspondance afin de ne pas répercuter les erreurs d'appariements sur la reconstruction 3D. Quelques mesures usuelles sont rappelées dans le tableau 5.1 ; elles mesurent la ressemblance d'un point (u_1, v_1) de l'image 1 I_r à celle d'un point (u_2, v_2) de l'image 2 I_v sur un masque carré $(2n + 1) \times (2n + 1)$ (une fenêtre, ou un patch).

| Mesure | Expression |
|--------|---|
| SAD | $\sum_{du=-n}^{du=+n} \sum_{dv=-n}^{dv=+n} I_v(u_2 + du, v_2 + dv) - I_r(u_r + du, v_r + dv) $ |
| ZSAD | $\sum_{du=-n}^{du=+n} \sum_{dv=-n}^{dv=+n} I_v(u_2 + du, v_2 + dv) - \bar{I}_v - I_r(u_r + du, v_r + dv) + \bar{I}_r $ |
| SSD | $\sum_{du=-n}^{du=+n} \sum_{dv=-n}^{dv=+n} (I_v(u_2 + du, v_2 + dv) - I_r(u_r + du, v_r + dv))^2$ |
| ZSSD | $\sum_{du=-n}^{du=+n} \sum_{dv=-n}^{dv=+n} (I_v(u_2 + du, v_2 + dv) - \bar{I}_v - I_r(u_r + du, v_r + dv) + \bar{I}_r)^2$ |

Tableau 5.1 – Quatre mesures usuelles de corrélation.

Ces mesures sont assez intuitives : SAD correspond mathématiquement à la norme L1, évaluée entre les fonctions I_r et I_v en $(2n + 1) \times (2n + 1)$ points et SSD au carré de L2. ZSAD et ZSSD sont centrées, par soustraction de la moyenne locale des intensités I_r et I_v , sur les patches courants. Ceci permet d'annuler l'effet des changements d'intensité locaux, mais a pour inconvénient de donner de bons scores à tort à des patches très différents. Avec ZSAD ou ZSSD, une fenêtre uniformément blanche correspond parfaitement à une fenêtre uniformément noire. Pour compenser l'effet des changements de luminosité, il est plus correct de modifier globalement les intensités des images, puis d'appliquer des mesures non-centrées. Les mesures de ressemblance que nous avons évoquées intègrent les différences des intensités sur des voisinages rectangulaires (généralement carrés) des pixels considérés.

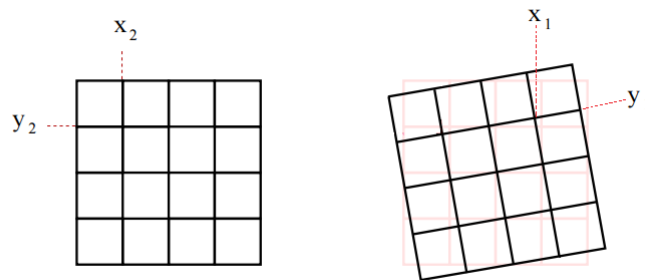


Figure 5.6 – Après la transformation, un pixel ne se trouve pas exactement sur une position entière

Après l'application de l'homographie sur les coordonnées des pixels du patch réel, nous obtenons les coordonnées des pixels du patch virtuel qui ne sont pas forcément entières (Figure 5.6). Une interpolation au plus proche voisin (l'interpolation d'ordre zéro) sera

utilisée pour arrondir les coordonnées transférées des pixels du patch virtuel.

5.3.3.2 Méthode d'optimisation Branch-and-Bound

Les algorithmes séparation-évaluation sont essentiellement des méthodes d'énumération intelligentes. Ils permettent de résoudre une large variété de problèmes d'optimisation non convexes sous contraintes. La méthode par séparation et évaluation, notée Branch-and-Bound, est une méthode qui permet la résolution exacte de problèmes d'optimisation, notamment les problèmes d'optimisation combinatoires dont on cherche à minimiser le coût de la recherche. Branch-and-Bound propose un mécanisme de recherche, qui permet une bonne exploitation de l'espace de recherche et l'aboutissement à la solution optimale plus rapidement que d'autres méthodes exactes en combinant deux principes primordiaux : la séparation et l'évaluation.

Le principe de base de la méthode Branch-and-Bound exploite la technique « Diviser pour régner ». Elle consiste à diviser le problème de manière à le représenter sous forme d'une arborescence, où chaque nœud correspond à une solution partielle. Ces solutions se forment de manière incrémentale en parcourant dans l'arbre. Chacune des solutions partielles potentielles possède une borne supérieure et une autre inférieure. Ces dernières sont utilisées pour couper quelques branches de l'arbre et ainsi éviter de l'explorer entièrement. En fait, si l'évaluation partielle d'un nœud x_i a montré que sa qualité est supérieure à la borne supérieure, le sous arbre en question sera élaguée ; sinon, le nœud sera divisé en sous nœuds. Ce processus se répète tant qu'il reste des branches non parcourues et la recherche continue jusqu'à trouver la solution optimale si elle existe. L'utilisation de la méthode Branch-and-Bound nécessite :

- Une solution initiale permettant d'entamer la recherche.
- Une stratégie permettant la division du problème P en sous problème P_i
- Une fonction permettant le calcul des différents bornes.
- Une stratégie de parcours de l'arbre : parcourir en profondeur, en largeur, etc...

Le point fort de cette méthode réside dans le fait qu'elle ne parcourt pas les sous branches dont on peut savoir a priori qu'elles ne permettent pas d'améliorer la solution rencontrée. Nous appliquons la méthode d'optimisation Branch-and-Bound pour ajuster explicitement les paramètres construisant l'homographie (plus précisément la normale et la distance du plan objet réel). L'idée est de trouver une homographie approximative \hat{H} à partir d'un ensemble de candidats qui peut être important. Dans le cas idéal, \hat{H} est presque égale à \bar{H} , où \bar{H} est l'homographie la plus proche à celle de la vérité terrain H parmi tous les candidats possibles (Fig.5.7). Nous pouvons exprimer l'estimation de \hat{H} par la minimisation de l'erreur SAD via la relation suivante :

$$\hat{H} = \underset{H \in \mathbb{F}_{2^{4n}}}{\operatorname{argmin}} SAD_H(P_R, P_V) \quad (5.22)$$

D'après l'équation (5.22), nous ne sommes pas sûrs que l'homographie \hat{H} est proche de \bar{H} , parce que la SAD ne dépend pas que de la déformation de la fenêtre. Elle est reliée à celle de la variation temporelle des intensités. Il est évident qu'une telle définition nous amène tout naturellement vers une évaluation de la variation de chaque pixel. Une variation est définie comme la somme des différences maximales de chaque pixel avec ses huit voisins. Nous avons :

$$\nu = \sum_{p \in I} \max_{q \in \mathcal{N}_8(p_1)} |I(p) - I(q)| \quad (5.23)$$

Une grande variation signifie que la fenêtre n'est pas lissée. Dans ce cas, les valeurs SAD entre deux homographies candidates peuvent varier même si ces deux homographies sont assez proches.

Pour quantifier l'erreur entre deux homographies. Nous définissons la formule de la façon suivante :

$$errSAD(H_1, H_2) = |SAD_{H_1}(P_R, P_V) - SAD_{H_2}(P_R, P_V)| \quad (5.24)$$

Il a été prouvé dans Korman *et al.* (2013) que la limite supérieure de $errSAD(H_1, H_2)$ est associée à trois facteurs dans un ensemble discret affine : le nombre de pas, la variation de la fenêtre et sa taille. Nous pouvons alors écrire :

$$errSAD(\bar{H}, H') \leq \Theta(\delta \cdot \frac{\nu}{n_1}) \quad (5.25)$$

Cependant, la mise en correspondance de points entre deux images nécessite un compromis

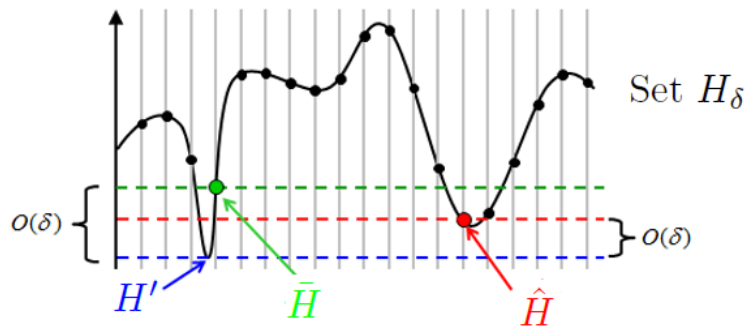


Figure 5.7 – Espace de recherche des homographies

entre une invariance à de nombreux points de vue, ou une discrimination plus forte. Nos travaux consistent pour l'instant à modéliser les objets réels sous la forme de patches-plan, puis à les utiliser pour chercher leur patchs miroir. Tous les paramètres de l'homographie sont supposés connus à l'exception de la normale et la distance qui ne dépend que du plan observé réel. Nous obtenons une évaluation de la différence entre deux images pour une valeur de l'homographie, c'est à dire la normale et la distance. En théorie cette fonction de coût est positive ou nulle lorsque la normale correspond à celle du plan.

Dans la figure 5.8 nous choisissons un groupe de paire de vues, et nous appliquons notre algorithme sur un patch localement plan construit à partir d'un point d'intérêt sélectionné avec le détecteur de Harris. L'homographie du plan est estimée par la méthode d'optimisation Branch-and-Bound. Une fois l'homographie du plan calculée, nous l'appliquons sur le patch observé pour obtenir le patch réfléchi candidat. Le résultat (Fig.5.8) montre le degré de ressemblance entre les deux patchs, ce qui montre que l'algorithme converge vers la meilleure homographie. Ainsi, la méthode présentée a pour avantage de fournir de bons résultats même sur une structure texturée de l'objet mais, en toute rigueur, elle nécessite que l'objet soit entièrement composé de zones planes. De plus, elle nécessite une pré-segmentation manuelle de l'image RMC.

5.3.3.3 Synthèse de l'algorithme proposé

Partant des points d'intérêt détectés dans une région de la vue directe délimitée manuellement, le principe général est de rechercher automatiquement les correspondants dans les vues réfléchies (simple ou double). Cette recherche se fait en utilisant le voisinage du

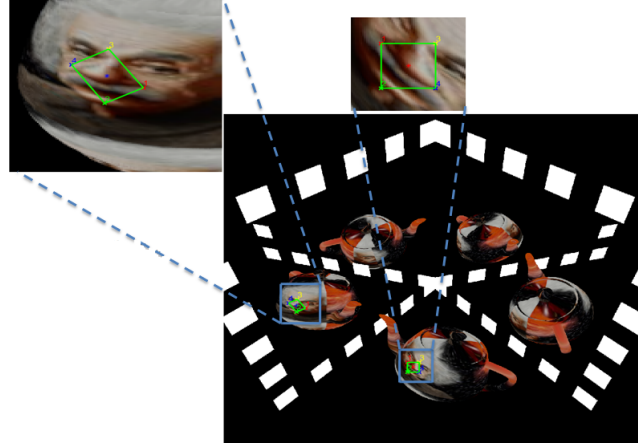


Figure 5.8 – Résultat de la méthode Branch-and-Bound présenté sur une paire de vues.
Les deux patches sont similaires.

point d'intérêt supposé plan et en calculant l'homographie optimale $\hat{\mathbf{H}}_{3 \times 3}$ minimisant la somme des différences absolues (SAD) des pixels entre le voisinage et son transformé, dans une autre vue par \mathbf{H} :

$$\hat{\mathbf{H}} = \underset{\mathbf{H}}{\operatorname{argmin}} \sum_{i=1}^k |I(\mathbf{p}_i) - I(\mathbf{H}\mathbf{p}_i)|/k, \quad (5.26)$$

pour les k pixels \mathbf{p}_i du voisinage du point d'intérêt dans l'image I . L'optimisation de \mathbf{H} se fait par l'algorithme de Branch-and-Bound [Korman *et al.* (2013)].

Algorithme 5.1 Trouver la meilleure homographie par la méthode d'optimisation BnB

Entrée: Patch à chercher et l'image désirée I_1, I_2 , et un paramètre de précision δ^*

Sortie: Meilleure homographie

$m = 0$

(1) Génération d'une population de solution $\mathbf{P}^m = \{H_0, \dots, H_{\delta_m}\}$ fixée par le paramètre δ_0 .

tantque $\delta_m > \delta^*$ **faire**

(a) Evaluer toutes les homographies par l'approximation de $SAD_H = \sum_{i=1}^k |I(p_i) - I(\mathbf{H}(p_i))|/k$

(b) Retourner la meilleure homographie H_m^{Best} avec sa valeur minimale d_H

(c) Soit $\mathbf{L}^m = \{H_m \in \mathbf{P}^m / SAD_{H_m}(I_1, I_2) - SAD_{H_m^{Best}}(I_1, I_2) < \delta \times \alpha + \beta\}$ avec α et β à régler.

(d) Améliorer le paramètre de précision $\delta_{m+1} = fact * \delta_m$ avec $0 < fact < 1$

(e) Re-générer une autre population de solution $\mathbf{P}^{m+1} = \{H_i | H_i \in \cup(\mathbf{P}^m, \mathbf{L}^m), \exists H_j \in \mathbf{L}^m \text{ i.e } |SAD_{H_i}(I_1, I_2) - SAD_{H_j}(I_1, I_2)| < \Theta(\delta \cdot \frac{\nu}{n_1})\}$

$m = m + 1$

fin tantque

(2) Retourner H_i^{Best}

5.4 Validation et évaluation de la méthode proposée

L'objectif de cette section est d'analyser les performances de l'algorithme proposé sur un ensemble d'images RMC. Nous commencerons par l'étude de l'influence des paramètres de l'algorithme sur sa stabilité. Après, nous présentons les résultats obtenus avec un seul miroir puis avec deux miroirs plans. Nous avons étudié la précision de l'algorithme dans plusieurs cas en variant la distance puis la normale, et les deux simultanément.

5.4.1 Stabilité de l'algorithme

Les résultats de l'estimation des homographies dépendent de différents paramètres qui influencent directement la convergence de notre algorithme. Nous citons en particulier le pas d'échantillonnage δ qui permet de contrôler l'espace de recherche, la taille du patch autour de chaque point, et le seuil à fixer pour passer à la prochaine itération. Pour mesurer l'influence de ces paramètres, nous prenons comme cas d'étude, l'exemple présenté dans la figure 5.8 qui permet d'estimer la normale et la distance simultanément. Pour chaque patch nous considérons tous les paramètres de calibration fixes et nous nous intéressons à l'orientation du patch transformé supposé plan. L'échantillonnage des paramètres du plan observé selon des intervalles connus nous permet de construire un espace de recherche de toutes les solutions possibles. La méthode de Branch-and-Bound est appliquée directement sur cet espace.

5.4.1.1 Influence du paramètre δ sur la précision

Nous prenons l'image dans la figure 5.8 pour laquelle nous avons la vérité de terrain : la normale est $n_r = [0, 0.7809, -0.6247]$ et la distance est 0.8. Les résultats du tableau 5.2 montrent l'influence du paramètre δ sur la convergence de la méthode vers les bons paramètres. La première erreur $bestDist = \min(\sum_{i=1}^k |I1(p_i) - I2(H(p_i))|) / k$ indique l'erreur SAD entre les deux patches et donne la meilleure erreur minimale parmi toutes les solutions existantes. La seconde $ErrOverlap = 1 - \frac{P_{vGT} \cap P_{vest}}{P_{vGT} \cup P_{vest}}$ indique l'erreur de recouvrement entre les deux patches, tant que cet indicateur est minimal la probabilité de mettre en correspondance deux patch est plus forte. La troisième $ErrPx = \sqrt{\sum_{i=1}^k (x_{ri} - x_{vi})^2 + (y_{ri} - y_{vi})^2}$ indique l'erreur pixellique entre les deux patches.

| δ | n_{rx} | n_{ry} | n_{rz} | distance | bestDist | ErrOverlap | ErrPx |
|----------|----------|----------|----------|----------|----------|------------|-------|
| 0.5 | 0.0214 | 0.7942 | -0.6035 | 0.7778 | 0.031 | 10.3% | 0.064 |
| 0.15 | 0.01 | 0.7887 | -0.6126 | 0.7878 | 0.030 | 9.4% | 0.062 |
| 0.25 | 0.03 | 0.7768 | -0.6226 | 0.7979 | 0.028 | 8.2% | 0.069 |

Tableau 5.2 – Influence du paramètre δ

Les résultats du tableau 5.2 montrent la précision des paramètres estimés ($n_{rx}, n_{ry}, n_{rz}, dist$) pour différentes valeurs de δ . La précision est évaluée par l'utilisation de trois indicateurs bestDist, ErrOverlap et ErrPx.

5.4.1.2 Influence de la taille de la fenêtre

Nous choisissons $\delta = 0.25$ pour réaliser les expérimentations de cette section. Le tableau 5.3 montre que le choix de la taille de la fenêtre est crucial pour garantir le succès de la méthode proposée. Le patch doit être large surtout dans les zones peu texturées afin de rendre l'appariement fiable. L'erreur de recouvrement est évaluée sur plusieurs variations de la taille de fenêtre.

| Taille de la fenêtre | δ | nr_x | nr_y | nr_z | distance | bestDist | ErrOverlap | ErrPx |
|----------------------|----------|--------|--------|---------|----------|----------|------------|--------|
| 11 | 0.25 | 0.0759 | 0.7953 | -0.6213 | 0.7968 | 0.022 | 16.8% | 0.0768 |
| 13 | 0.25 | 0.0338 | 0.7795 | -0.6226 | 0.7981 | 0.025 | 17.9% | 0.0711 |
| 15 | 0.25 | 0.0426 | 0.7978 | -0.6226 | 0.7993 | 0.027 | 12.5% | 0.0599 |
| 17 | 0.25 | 0.0196 | 0.7797 | -0.6277 | 0.8039 | 0.027 | 6.6% | 0.0549 |
| 19 | 0.25 | 0.0120 | 0.7873 | -0.6252 | 0.8021 | 0.033 | 5.5% | 0.0519 |

Tableau 5.3 – Influence de la taille du Patch

Les résultats obtenus en faisant varier les paramètres qui influencent la convergence de l'algorithme indiquent qu'il faut trouver un compromis entre ces paramètres pour aboutir à une bonne précision de la méthode. Nous avons utilisé pour nos expériences, les valeurs $\delta = 0.25$ et $T = 19$ qui donnent des résultats satisfaisants.

5.4.2 Critère d'évaluation existants

Pour comparer le résultat obtenu avec le jeu de données de la vérité terrain, une meilleure évaluation de la performance de l'algorithme utilisé est donnée par le critère de répétabilité, qui permet de quantifier le recouvrement entre la position exacte du patch de référence et la position estimée du patch transformé (voir les quadrilatères verts illustrés dans la figure 5.8). La répétabilité, indique le nombre moyen de points (ou régions) correspondants détectés dans deux images. Plus la répétabilité entre deux images est grande, plus la probabilité de trouver les mêmes correspondants entre les deux images est forte. Un moyen de calculer la répétabilité est présenté par Mikolajczyk *et al.* (2005). On commence par définir une mesure de recouvrement ϵ_s comme étant le rapport entre l'intersection et l'union de deux régions correspondantes. Soit Pv_{GT} et Pv_{est} deux patches transformés dans l'objet virtuel. L'erreur de recouvrement est définie par :

$$\epsilon_s = 1 - \frac{Pv_{GT} \cap Pv_{est}}{Pv_{GT} \cup Pv_{est}} \quad (5.27)$$

Mikolajczyk évalue la répétabilité à partir d'une mesure du recouvrement de deux patches. En effet, la probabilité de mettre en correspondance deux patches augmente lorsque l'erreur de recouvrement de ceux-ci diminue. Le score de répétabilité entre deux patches, est donc le rapport du nombre de patches se correspondant au sens du recouvrement, sur le plus petit des deux nombres de patches. Ainsi, Mikolajczyk *et al.* (2005) affirme qu'une erreur de recouvrement de $\epsilon_0 = 0.5$ entre deux patches permet encore de les mettre en correspondance avec succès. Cependant, lors de l'étude expérimentale proposée dans le travail de Korman *et al.* (2013), ce paramètre est fixé à 0.2. Nous avons retenu ce seuil 0.2 pour tester la robustesse de nos estimations.

Nous évaluons la performance de l'algorithme par le calcul de l'erreur SAD entre les deux

patches transformés par la formule suivante :

$$\epsilon_p = \sum_{i=1}^k |I(H_{GT}p_i) - I(H_{est}p_i)| \quad (5.28)$$

Une fois l'homographie calculée, il est facile de vérifier l'exactitude de l'ensemble des couples de correspondants (p_r, p_v) . En effet, si H désigne l'homographie entre les deux patches réel-virtuel, alors le patch réel est correctement apparié avec le patch virtuel si :

$$\sum_{i=1}^k \|p_{v_i} - Hp_{r_i}\| < \epsilon \quad (5.29)$$

où ϵ est l'erreur de localisation (généralement entre 1 et 5 pixels). On peut aussi utiliser l'erreur de recouvrement introduite par Mikolajczyk *et al.* (2005) (Equation 5.27), pour sélectionner les correspondants corrects connaissant l'homographie entre les régions. H étant estimée à partir de l'ensemble initial de correspondants, il est important que celui-ci comporte peu de correspondants incorrects.

5.4.3 Résultats et apports de notre algorithme

Nous avons commencé par l'estimation de l'homographie en faisant varier la distance au plan observé (celui sur lequel le patch est défini), et nous considérons que les autres paramètres sont connus. Afin de converger vers la meilleure solution de la distance par la méthode de Branch-and-Bound, nous devons générer un nombre aléatoire dans l'intervalle $[d_{min}, d_{max}]$ pour chaque itération selon la variation du paramètres d'échantillonnage comme le montre la figure 5.9. Ensuite, nous appliquons les différentes étapes de l'algo-

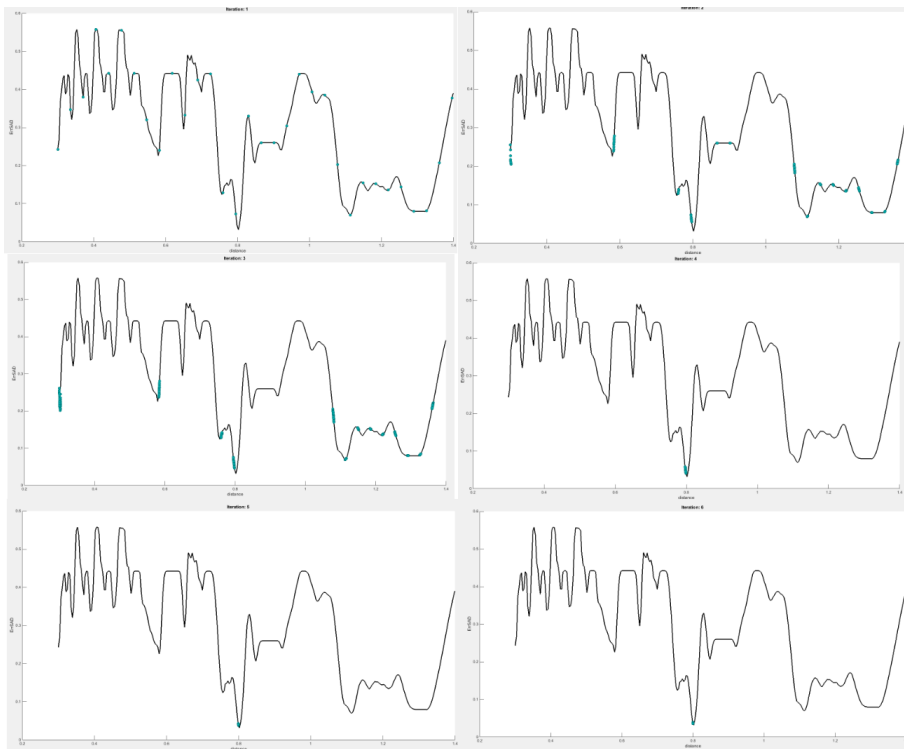


Figure 5.9 – Variation de la distance en fonction de l'erreur SAD en utilisant BnB.

rithme mentionné ci-dessus. Cette figure représente l'espace de recherche et le nombre de solutions générées par un pas d'échantillonnage variable.

La figure 5.10 montre le patch rouge qui représente le patch transformé obtenu par la valeur exacte de la distance et le patch vert qui représente le patch transformé par la distance estimée. L'erreur de recouvrement (Overlap error) entre les deux patch est 14.1%. L'erreur pixellique est 0.10.

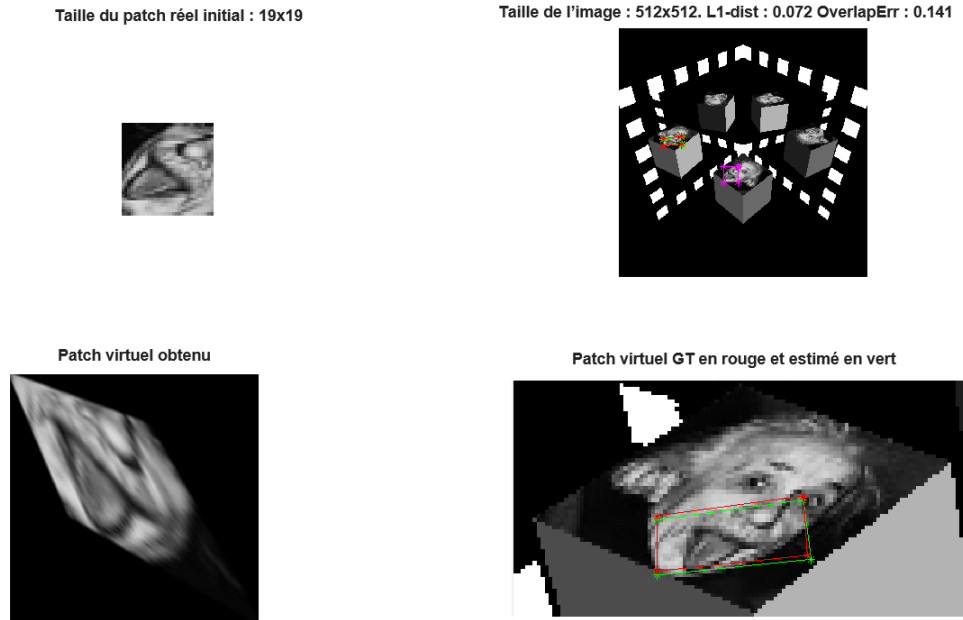


Figure 5.10 – L’affichage du patch réel et le patch virtuel transformé.

Le tableau 5.4 dresse les résultats concernant l'ensemble des itérations effectués durant l'exécution de l'algorithme pour converger vers le minimum optimal avec un pas de 0.25 pour un patch de taille 19×19 . Globalement, nous pouvons constater que les calculs mènent à une estimation précise de la distance après 5 itérations. À noter que les paramètres intrinsèques sont : $\alpha_u = 512, \alpha_v = 512$ et $u_0 = v_0 = 256$.

| NbrIter | nm_x | nm_y | nm_z | dm | nr_x | nr_y | nr_z | distance |
|---------|--------|---------|---------|---------|--------|--------|---------|----------|
| 1 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0 | 0.7809 | -0.6247 | 0.7950 |
| 2 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0 | 0.7809 | -0.6247 | 0.8091 |
| 3 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0 | 0.7809 | -0.6247 | 1.3023 |
| 4 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0 | 0.7809 | -0.6247 | 1.3207 |
| 5 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0 | 0.7809 | -0.6247 | 0.8010 |
| 6 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0 | 0.7809 | -0.6247 | 0.8015 |
| 7 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0 | 0.7809 | -0.6247 | 0.8014 |
| 8 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0 | 0.7809 | -0.6247 | 0.8016 |

Tableau 5.4 – L’exécution de la méthode Branch-and-Bound avec un pas de 0.25 et 8 itérations pour converger vers la solution optimale $\text{dist}=0.8010$.

Variation de l'angle de l'élévation :

Dans ce cas, on fait varier la forme et la position du patch virtuel en fonction de l'angle d'élévation et on garde la valeur de l'angle de l'azimuth $\theta = -0.67rad$. La vraie valeur de l'angle de l'élévation est égale à : $1.57rad$.

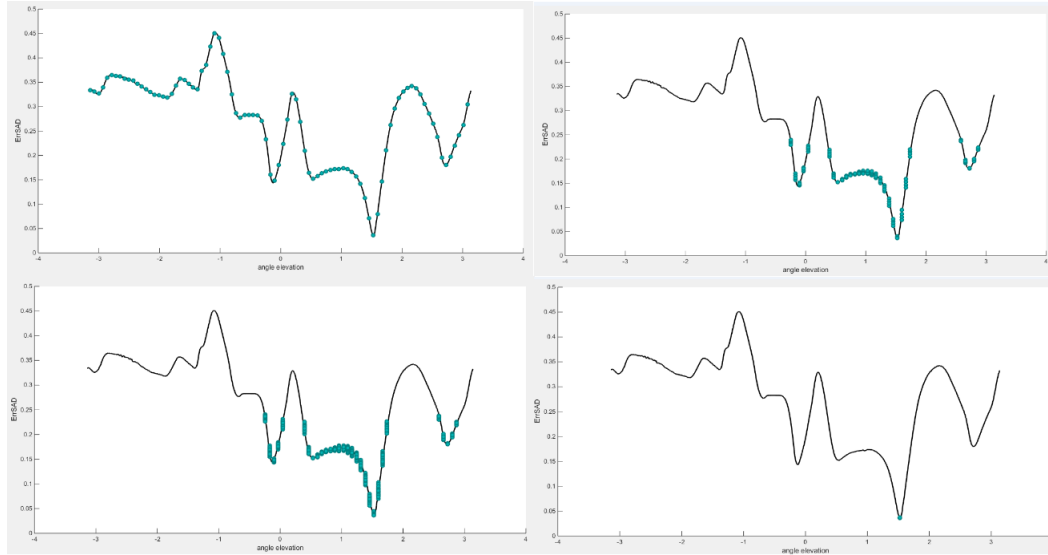


Figure 5.11 – Variation de l'élévation en fonction de l'erreur SAD en utilisant BnB.

La figure 5.12 montre que le patch rouge représente le patch transformé obtenu par la valeur exacte de l'angle élévation ($\phi = 1.57rad$) et le patch vert représente le patch transformé par l'angle élévation θ estimé. L'erreur de recouvrement (Overlap error) entre les deux patches est égale à 9.5%. L'erreur pixellique est 0.1233.

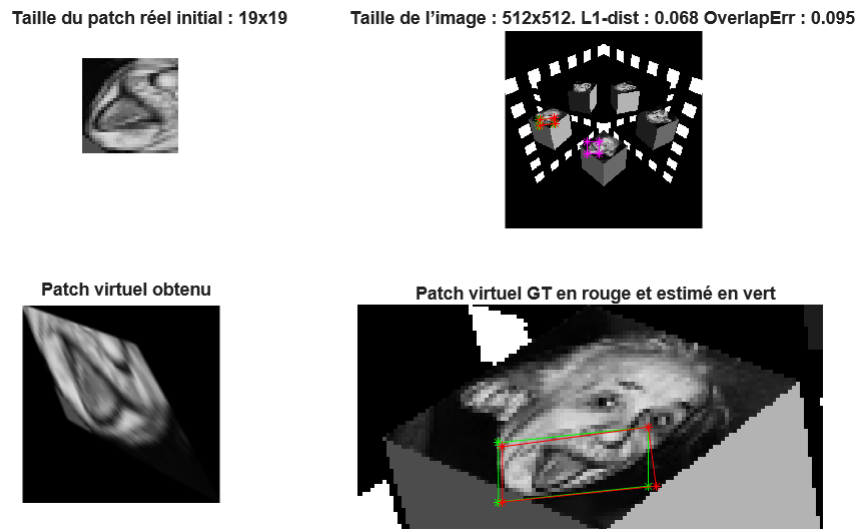


Figure 5.12 – L'affichage du patch réel et le patch virtuel transformé.

Le tableau 5.5 montre l'ensemble des itérations effectuées durant l'exécution de l'algorithme pour converger au minimum optimal avec un pas de 0.25 pour un patch de taille 19×19 .

| NbrIter | nm_x | nm_y | nm_z | dm | θ | ϕ | distance |
|---------|--------|---------|---------|---------|----------|--------|----------|
| 1 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.67 | 1.4142 | 0.8 |
| 2 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.67 | 1.5910 | 0.8 |
| 3 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.67 | 1.4319 | 0.8 |
| 4 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.67 | 1.5464 | 0.8 |
| 5 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.67 | 1.5464 | 0.8 |
| 6 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.67 | 1.5464 | 0.8 |

Tableau 5.5 – L’exécution de la méthode BnB par un pas de 0.25 avec 6 itérations pour converger vers la solution optimale $\phi = 1.57$.

Variation de l’angle de l’azimuth :

On fait varier la forme et la position du patch virtuel en fonction de l’angle azimuth et on garde la valeur de l’angle de l’élévation $\phi = 1.57rad$. La vraie valeur de l’angle azimuth est égale : $-0.67rad$.

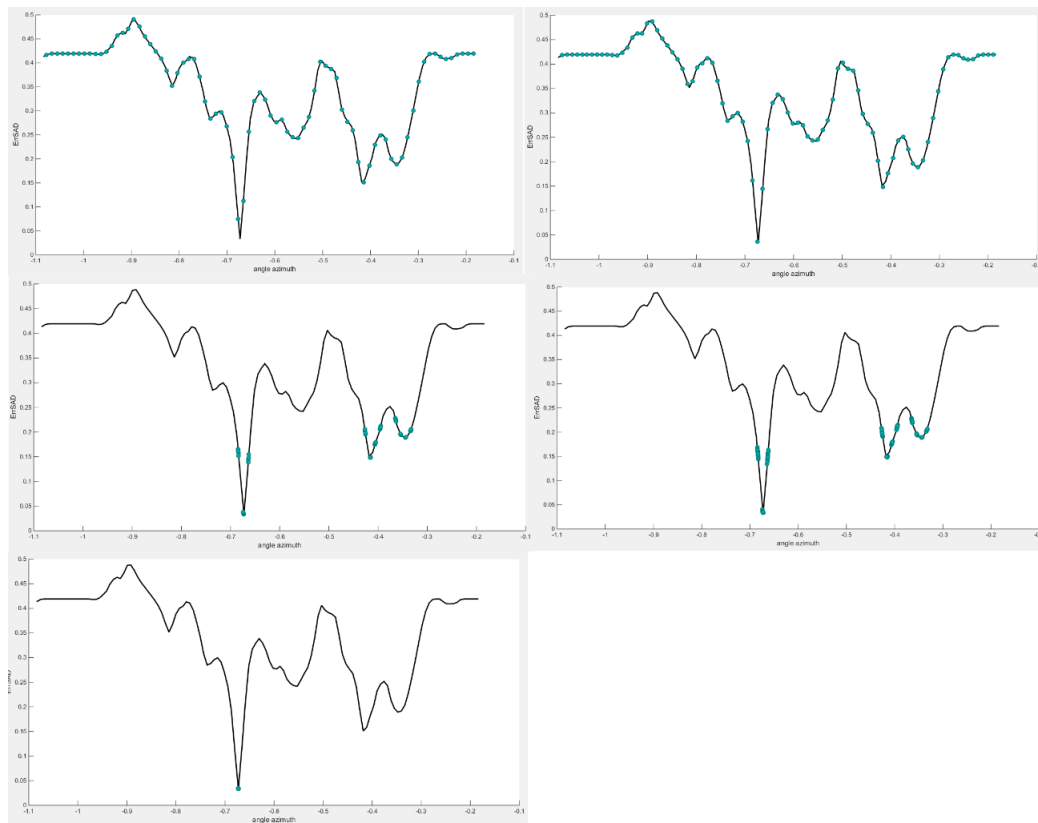


Figure 5.13 – Variation de l’azimuth en fonction de l’erreur SAD en utilisant BnB.

La figure 5.14 montre que le patch rouge représente le patch transformé obtenu par la valeur exacte de l’angle azimuth ($\theta = -0.67rad$) et le patch vert représente le patch transformé par l’angle azimuth θ estimé. L’erreur de recouvrement (Overlap error) entre les deux patch est 11.1%. L’erreur pixellique est 0.1345.

Le tableau 5.6 montre l’ensemble des itérations effectuées durant l’exécution de l’algorithme pour converger au minimum optimal avec un pas de 0.25 pour un patch de taille 19×19 .

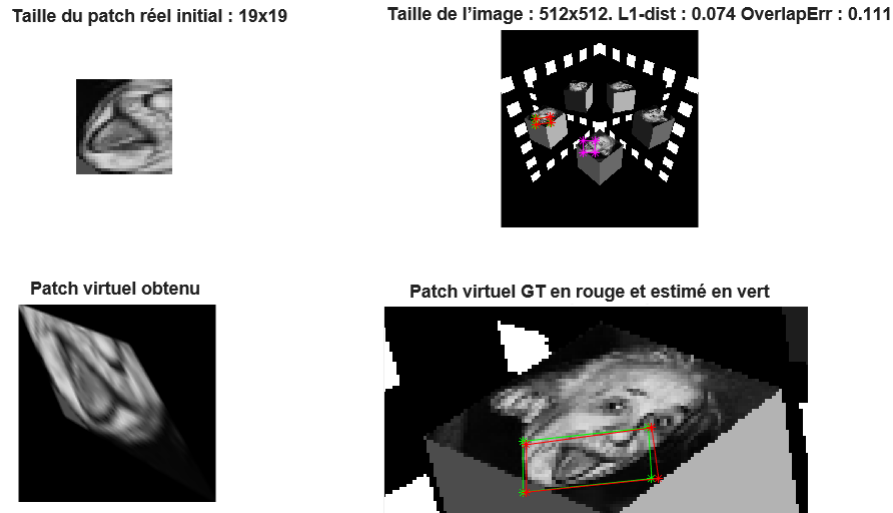


Figure 5.14 – L’affichage du patch réel et le patch virtuel transformé.

| NbrIter | nm_x | nm_y | nm_z | dm | θ | ϕ | distance |
|---------|--------|---------|---------|---------|----------|--------|----------|
| 1 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.6671 | 1.57 | 0.8 |
| 2 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.6773 | 1.57 | 0.8 |
| 3 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.6739 | 1.57 | 0.8 |
| 4 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.6729 | 1.57 | 0.8 |
| 5 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.6728 | 1.57 | 0.8 |
| 6 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | -0.6728 | 1.57 | 0.8 |

Tableau 5.6 – L’exécution de la méthode Branch-and-Bound par un pas de 0.25 avec 6 itérations pour converger vers la solution optimale $\theta = -0.67$.

Variation de l’élévation et l’azimuth :

Nous faisons varier l’angle de rotation azimuth entre 0 à 2π et l’angle de rotation élévation varie entre $-\pi/2$ à $\pi/2$ avec un pas de $\delta = 2.5$. Dans la figure 5.16, le patch rouge représente le patch transformé obtenu par la valeur exacte de la distance et le patch vert représente le patch transformé par la distance estimé. L’erreur de recouvrement (Overlap error) entre les deux patches est égale à 15.6%. L’erreur pixellique est 0.1223.

| NbrIter | nm_x | nm_y | nm_z | dm | nr_x | nr_y | nr_z | distance |
|---------|--------|---------|---------|---------|--------|--------|---------|----------|
| 1 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.4716 | 0.6953 | -0.5424 | 0.8 |
| 2 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.1992 | 0.7656 | -0.6116 | 0.8 |
| 3 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0448 | 0.7820 | -0.6216 | 0.8 |
| 4 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0448 | 0.7825 | -0.6207 | 0.8 |
| 5 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0448 | 0.7831 | -0.6207 | 0.8 |
| 6 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0015 | 0.7845 | -0.6226 | 0.8 |

Tableau 5.7 – L’exécution de la méthode BnB par un pas de 0.25 avec 6 itérations pour converger vers la solution optimale $nr=[0,0.7809,-0.6247]$.

Le tableau 5.7 montre l’ensemble des itérations effectuées durant l’exécution de l’algorithme pour converger au minimum optimal avec un pas de 0.25 pour un patch de taille 19×19 .

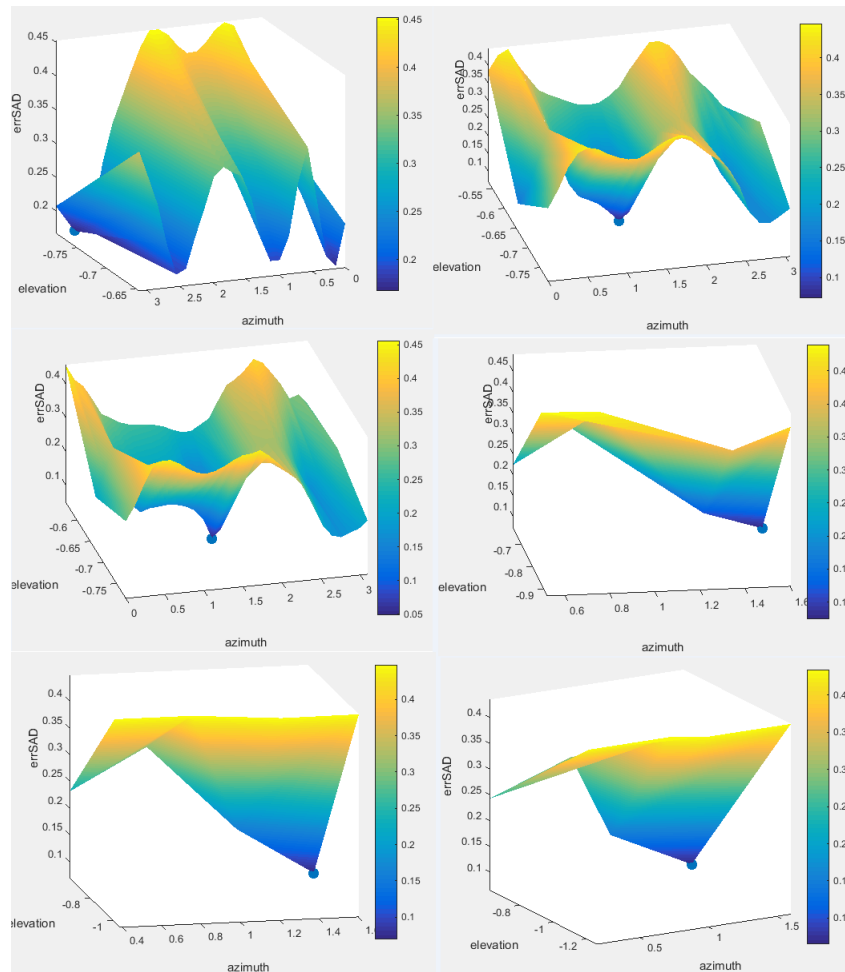


Figure 5.15 – Variation de l’azimuth et l’élévation en fonction de l’erreur SAD en utilisant BnB.

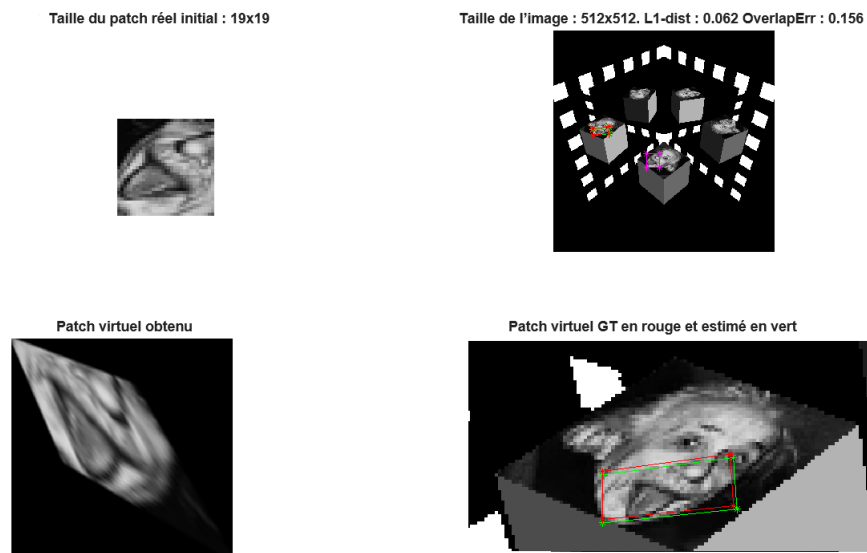


Figure 5.16 – L’affichage du patch réel et le patch virtuel transformé.

Variation de la normale et la distance :

Ici, nous évaluons l'algorithme en faisant varier la distance entre $[d_{min}, d_{max}]$, ϕ entre $[0, 2\pi]$ et θ entre $[-\pi/2, \pi/2]$. Cet échantillonnage permet de construire l'espace de recherche de toutes les solutions. L'algorithme converge vers la meilleure solution après plusieurs itérations comme illustré dans la figure 5.17. Cette figure montre la fréquence de chaque pixel utilisé pour le calcul d'erreur SAD en fonction des paramètres à estimer. Avec la diminution du nombre de la population, le nombre de patches correspondants total diminue tandis que le patch souhaité est plus précis.

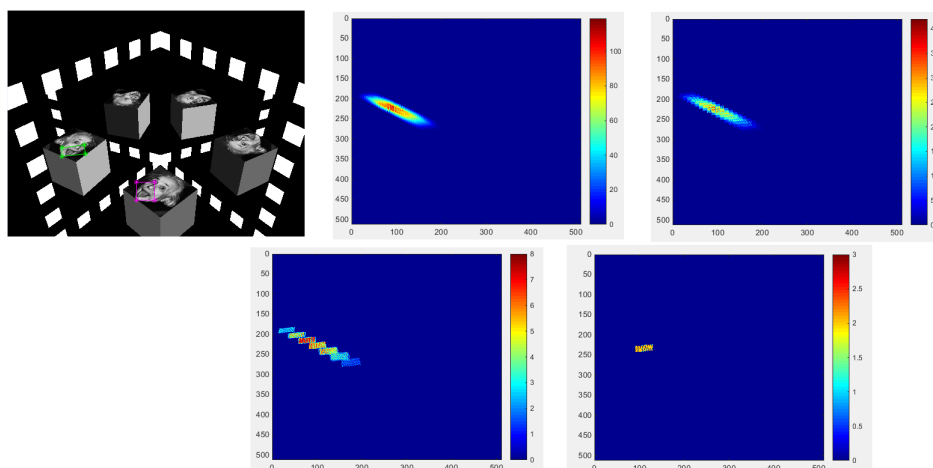


Figure 5.17 – Une représentation graphique montre la fréquence de chaque pixel utilisé pour le calcul d'erreur SAD en fonction des paramètres à estimer.

La figure 5.18 montre le patch rouge qui représente le patch transformé obtenu par la valeur exacte de la distance et le patch vert qui représente le patch transformé par la distance estimée. L'erreur de recouvrement (Overlap error) entre les deux patches est 3.7%. L'erreur pixellique est 0.0645.

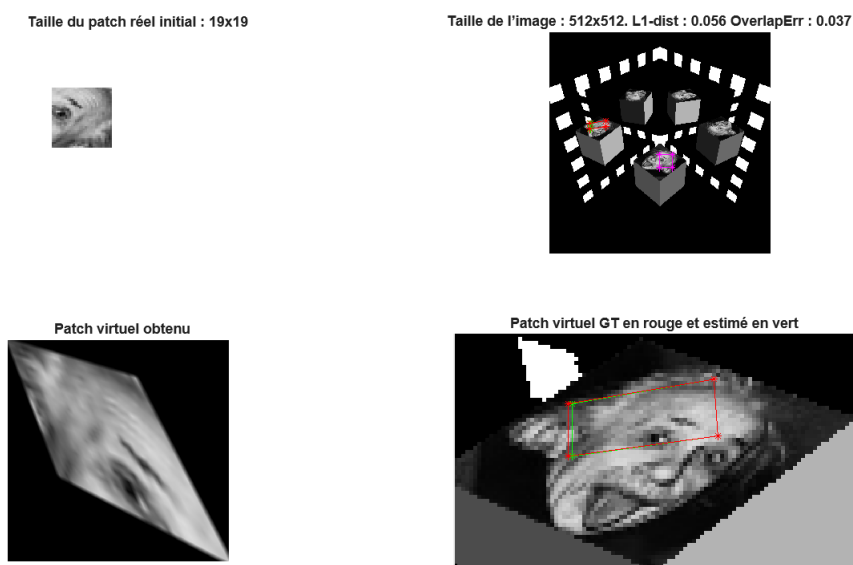


Figure 5.18 – L'affichage du patch réel et le patch virtuel transformé.

La table 5.8 montre l'ensemble des itérations effectuées durant l'exécution de l'algorithme avant de converger vers le minimum optimal avec un pas de 0.25 pour un patch de taille 19×19 .

| NbrIter | nm_x | nm_y | nm_z | dm | nr_x | nr_y | nr_z | distance |
|---------|--------|---------|---------|---------|--------|--------|---------|----------|
| 1 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0682 | 0.7922 | -0.6064 | 0.7778 |
| 2 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0149 | 0.7360 | -0.6769 | 0.8591 |
| 3 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0184 | 0.7571 | -0.6531 | 0.8305 |
| 4 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0195 | 0.8001 | -0.5995 | 0.7732 |
| 5 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0195 | 0.7975 | -0.5995 | 0.7732 |
| 6 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0273 | 0.7962 | -0.5995 | 0.7732 |
| 7 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0219 | 0.7905 | -0.6465 | 0.8269 |
| 8 | 0.7880 | -0.3846 | -0.4808 | -0.7696 | 0.0484 | 0.7921 | -0.6085 | 0.7819 |

Tableau 5.8 – L'exécution de la méthode Branch-and-Bound par un pas de 0.25 avec 8 itérations pour converger vers la solution optimale $nr=[0,0.7809,-0.6247]$ et $dist=0.8$.

Cas d'un seul miroir plan : Nous évaluons la performance de l'algorithme dans le cadre de la mise en correspondance par patch-plan avec un changement de point de vue important et une symétrie due à l'utilisation de miroir plan. Pour ce faire, nous choisissons une image de test présentant des transformations géométriques (changement de point de vue, changement d'échelle, rotation et réflexion). Pour évaluer la performance de l'algorithme, nous utilisons les critères d'évaluation introduits dans la section 5.4.2. (eq 5.27 et 5.28).

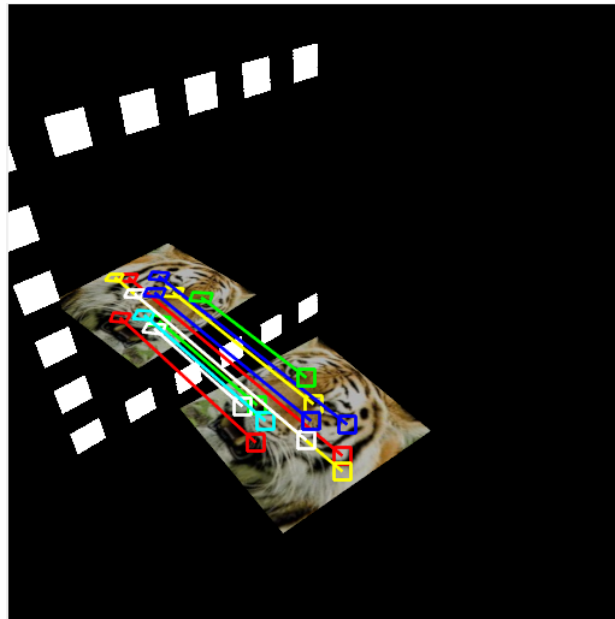


Figure 5.19 – Exemple d'un ensemble de patches mis en correspondance par les homographies locales estimées.

La figure 5.20 illustre l'évaluation de l'algorithme par le calcul de l'erreur de recouvrement sur le jeu de données synthétiques. Cette base de données présente différentes conditions d'objets en jouant sur la texture, le changement d'illumination et les zooms. Les résultats montrent que les patches transformés obtenus sont corrects.


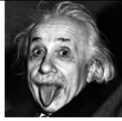
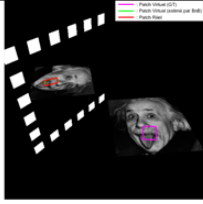
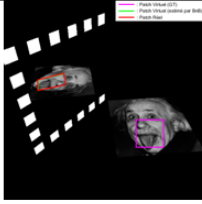

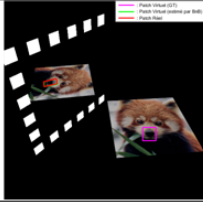
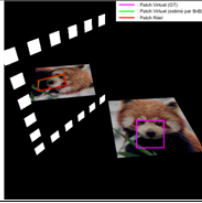

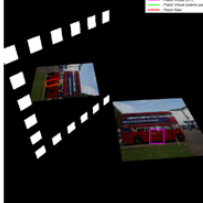
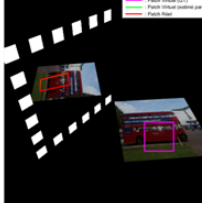


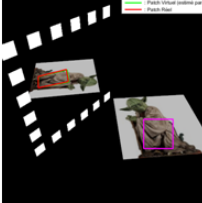

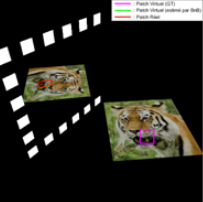
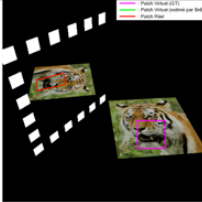



|  | | |
|---|---|--|
| Images | Template de Taille 17x17 | Template de taille 35x35 |
|  |  |  |
| Overlaps : | 10.4% | 3.4% |
|  |  |  |
| Overlaps : | 7.6% | 3.5% |
|  |  |  |
| Overlaps : | 9.8% | 2.3% |
|  |  |  |
| Overlaps : | 11.8% | 3.4% |
|  |  |  |
| Overlaps : | 9.9% | 3.4% |
|  |  |  |
| Overlaps : | 4.6% | 3.5% |

Figure 5.20 – Calcul de l'erreur de recouvrement de différentes images pour 2 tailles de patches.

Cas de deux miroirs plans : La figure 5.21 illustre l'ensemble des homographies calculées entre la vue directe et la vue virtuelle dans chaque groupe de paires de vue afin d'obtenir patches similaires, dans le cas d'objet non planaire pour évaluer la robustesse de la méthode.

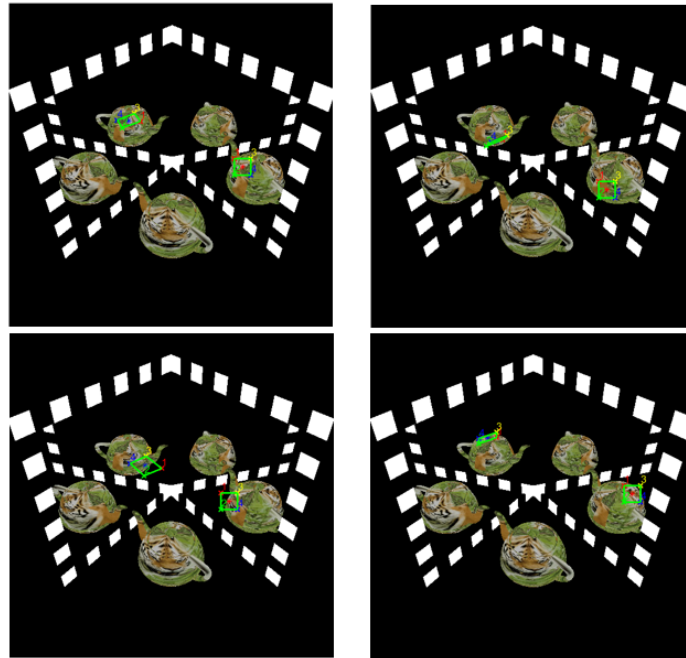


Figure 5.21 – Résultat de mise en correspondance des patches dans une paire de vue (V2 et V21).

5.5 Conclusion

Dans de ce chapitre, nous avons rappelé différents outils mathématiques de la géométrie projective qui servent à estimer les homographies. L'image issue du système de vision est segmentée manuellement en quatre paires de vues. La restriction de l'objet à un plan local simplifie le problème car dans ce cas, la transformation entre les deux repères caméras est une homographie. Le calcul d'homographie à partir d'un jeu de données nécessite une connaissance des paramètres de la caméra afin d'optimiser l'erreur entre les deux patches. Cette nouvelle méthode propose l'estimation de l'homographie dans un système composé de miroirs-caméra, nous avons proposé un algorithme basé sur la méthode d'optimisation Branch-and-Bound pour estimer la meilleure homographie entre chaque patch réel-virtuel. Cette méthode permet d'estimer les paramètres de plan observé pour trouver leur correspondant réfléchi par un miroir plan. Ce qui permet de détecter de manière optimale la bonne position des patches virtuels de l'objet réel dans une région candidate à des positions choisies (les points caractéristiques). Les résultats de la décomposition de l'homographie obtenus sont souvent corrects.

APPLICATION À LA RECONSTRUCTION 3D D'OBJETS

Sommaire

| | | |
|-------|--|-----------|
| 6.1 | Reconstruction 3D | 86 |
| 6.2 | Résultats expérimentaux | 88 |
| 6.2.1 | Données synthétiques | 88 |
| 6.2.2 | Evaluation de la précision de la reconstruction 3D | 92 |
| 6.2.3 | Comparaison avec la méthode des silhouettes | 93 |
| 6.3 | Conclusion | 96 |

Nous allons appliquer dans ce chapitre le processus de la reconstruction sur différents exemples. Pour ce faire, nous avons commencé par une technique de propagation d'appariements de points. Pour chaque point d'intérêt et son homographie associée, nous calculons tous ses voisins par la méthode du plus proche voisin en calculant la ressemblance entre les deux pixels et leur voisinage. Le résultat de l'algorithme est un nuage de points dense représentant la surface de l'objet. Enfin, quelques exemples de la reconstruction 3D sont présentés afin de permettre des mesures quantitatives sur l'objet reconstruit.

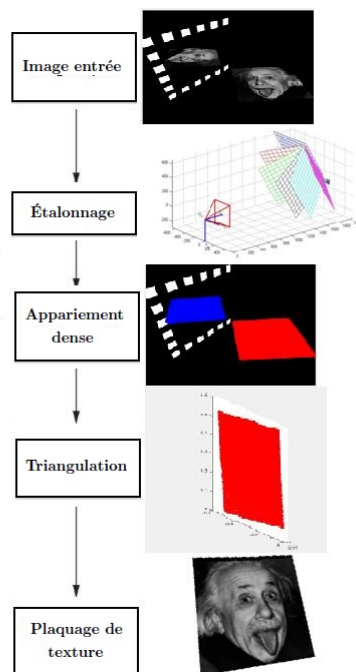


Figure 6.1 – Pipeline de la reconstruction 3D.

La reconstruction 3D est la dernière étape pour obtenir une représentation en trois dimensions d'un objet à partir d'un ensemble d'images prises sous différents points de vue. Le processus de reconstruction 3D se décompose en plusieurs étapes que nous détaillerons dans la première partie (cf. fig6.1).

6.1 Reconstruction 3D

La figure 6.2 illustre différentes représentations d'un plan dans chacune des différentes géométries. Le niveau d'une reconstruction définit le type de transformation géométrique liant l'objet et son modèle tridimensionnel.

La mise en correspondance éparses est appliquée pour obtenir un nombre limité de points d'intérêt. Si les modèles de projection des caméras sont connus, il est possible de calculer par triangulation les coordonnées 3D des points appariés.

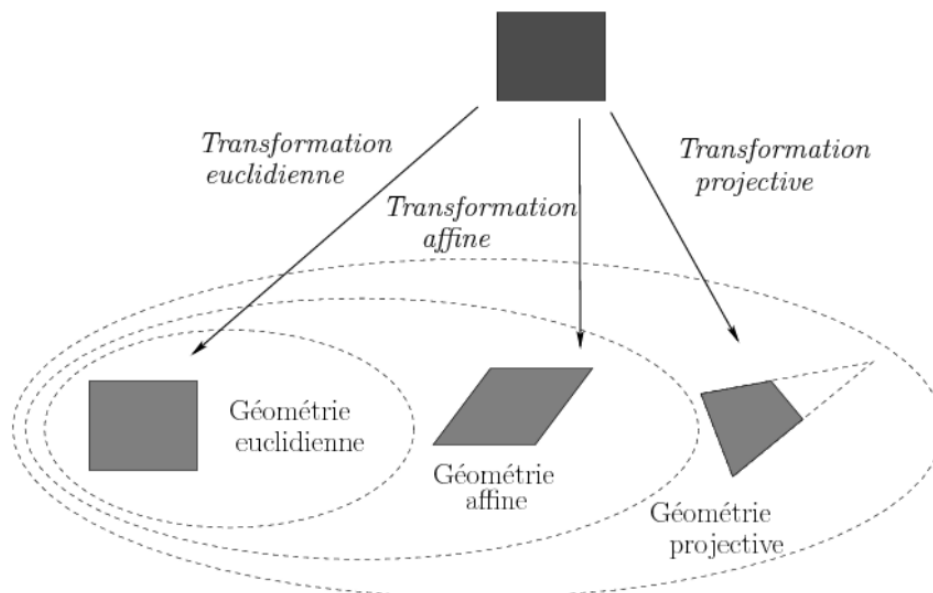


Figure 6.2 – Géométrie euclidienne, affine et projective

La mise en correspondance dense sera importante pour le passage à une reconstruction 3D réaliste. L'appariement dense est calculé à partir des homographies estimées autour d'une zone plane. Ensuite, une méthode simple basée sur la propagation du plus proche voisin est utilisée pour obtenir une reconstruction 3D dense. Dans cette méthode, on ne garde que les points de l'image qui ont été appariés et on propage les points d'intérêt autour de ces points dans une zone de recherche réduite, et ainsi de suite. Cette méthode permet d'accélérer les calculs car nous supposons que la zone de recherche est planaire et subit la même transformation affine que le point estimé initialement. La zone de recherche des correspondants est réduite aux voisinages d'appariements fiables, en faisant l'hypothèse que deux pixels voisins ont la même transformation affine. Cependant, le succès de ce type de méthode est fortement dépendant du choix des appariements initiaux. En plus, cette technique ne considère qu'une mise en correspondance à la fois et ne permet la propagation que dans un petit voisinage.

Pour chaque caméra i , sa matrice associée P_i est exprimée par :

$$P_i = K_i[R_i|t_i] \quad (6.1)$$

La relation entre un point 3D $\tilde{\mathcal{X}}(X, Y, Z, 1)$ et un point image $x(u, v)$ s'écrit :

$$x = \begin{pmatrix} \frac{P_i^1 \tilde{\mathcal{X}}}{P_i^3 \tilde{\mathcal{X}}} \\ \frac{P_i^2 \tilde{\mathcal{X}}}{P_i^3 \tilde{\mathcal{X}}} \end{pmatrix} \quad (6.2)$$

Chaque point 3D donne lieu à 2 équations qu'on peut déduire de (6.2) :

$$(uP_i^3 - P_i^1)\tilde{\mathcal{X}} = 0 \quad (6.3)$$

$$(vP_i^3 - P_i^2)\tilde{\mathcal{X}} = 0 \quad (6.4)$$

Chaque équation définit une droite (voir figure 6.3) qui est cependant, dépendante de

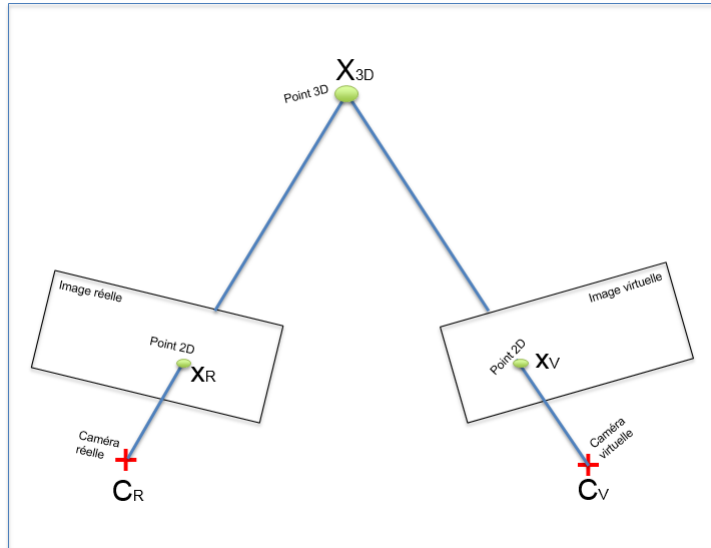


Figure 6.3 – Triangulation : les droites épipolaires ne se coupent pas forcément

la bonne estimation des paramètres internes et externes des caméras : un mauvais appariement provoque un écart entre ces droites. Pratiquement, la probabilité pour que ces droites s'intersectent en un point est très faible. Hartley et Sturm (1997) ont comparé les différentes méthodes existantes de triangulation. Nous avons choisi d'implémenter la méthode du point milieu pour sa simplicité et parce que nous travaillons en reconstruction euclidienne.

Dans le cas d'une correspondance dans n images (x_1, \dots, x_n) et m caméras (P_1, \dots, P_m) , on peut raffiner le point reconstruit \mathcal{X} en minimisant l'erreur suivante par rapport à \mathcal{X} .

$$\sum_i^m \left[\left(u - \frac{P_i^1 \mathcal{X}}{P_i^3 \mathcal{X}} \right)^2 + \left(v - \frac{P_i^2 \mathcal{X}}{P_i^3 \mathcal{X}} \right)^2 \right] \quad (6.5)$$

Il s'agit d'un problème non linéaire par rapport aux 3 inconnues \mathcal{X} qu'on peut par exemple résoudre en utilisant la méthode itérative de Levenberg-Marquardt.

On peut améliorer la précision de la reconstruction en appliquant l'ajustement de faisceaux Triggs *et al.* (1999) qui réalise la minimisation de la somme des erreurs de reprojection,

par rapport aux points 3D et aux matrices de projection.

6.2 Résultats expérimentaux

Dans cette section, nous allons présenter les résultats de reconstruction 3D obtenus avec la méthode détaillée précédemment. La reconstruction est représentée pour chaque objet de type différent afin de suivre pas-à-pas le processus conduisant à la structure 3D, et afin de mieux comprendre les difficultés rencontrées.

6.2.1 Données synthétiques

Toutes les images synthétiques ont été construites à l'aide du logiciel PovRay avec des objets de formes différentes, ce qui permet la comparaison de nos résultats avec la vérité terrain. L'algorithme de reconstruction dense proposé ici s'applique à un objet en vue directe et ses multiples réflexions obtenues par les deux miroirs planaires.

6.2.1.1 Reconstruction d'un plan

Afin de reconstruire l'objet plan, nous mettons en correspondance les points obtenus par les homographies estimées. De cette manière, nous obtenons une première structure 3D définie par un nuage de points. Afin de compenser les erreurs inévitables lors de la mise en correspondance, nous avons appliqué un ajustement de faisceaux. La figure 6.4 illustre une représentation 3D à partir de points caractéristiques mis en correspondance entre la vue directe et sa réflexion. L'objet a été reconstruit à partir de 11604 correspondances de points.

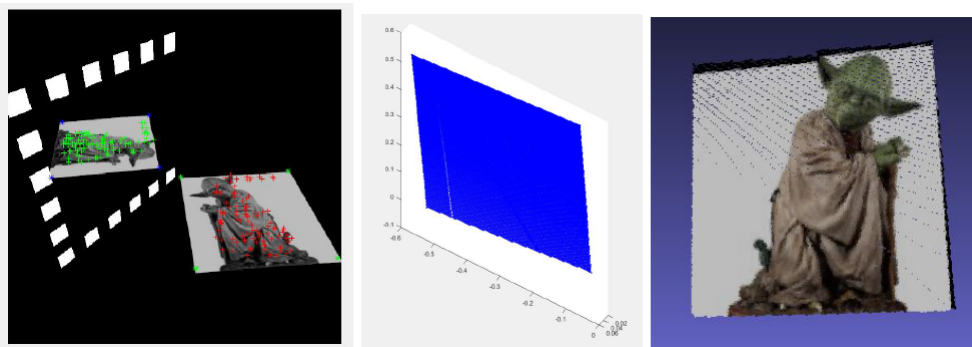


Figure 6.4 – Différentes étapes de la reconstruction 3D dense d'un plan

Les figures 6.5 et 6.6 représentent l'ensemble des points reconstruits à partir d'une paire de vues symétrique en utilisant un seul miroir. Les deux vues sont affichées dans un même repère. Pour évaluer la précision de la surface planaire reconstruite, nous comparons les résultats issus de notre méthode avec la vérité terrain. Sur cette image de résolution 512×512 , nous obtenons une erreur inférieure au millimètre.

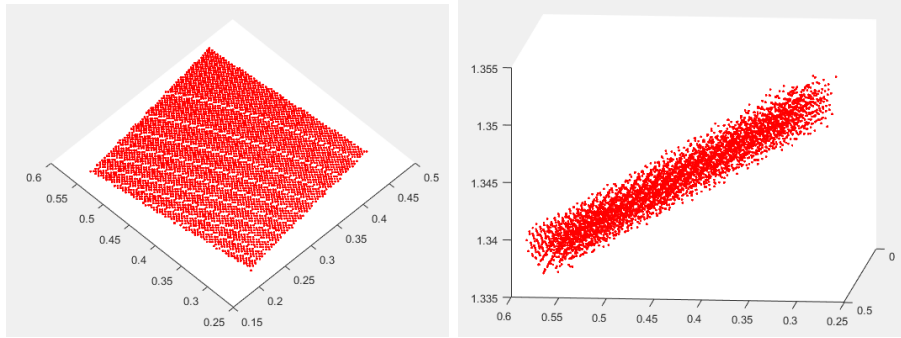


Figure 6.5 – Représentation de nuage de points 3D de la vérité de terrain de deux façons différentes.

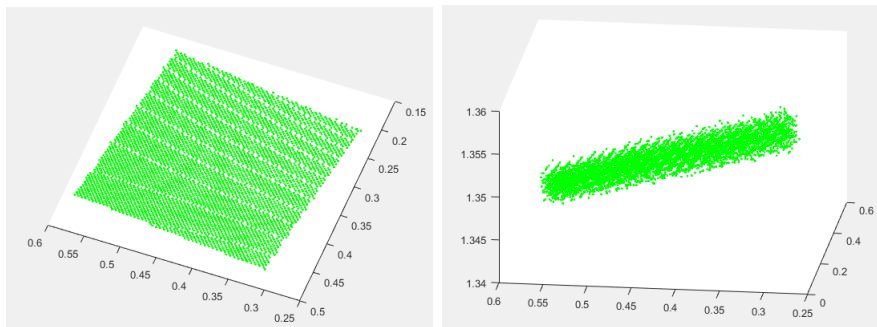


Figure 6.6 – Représentation de nuage de points 3D de notre méthode de deux façons différentes.

6.2.1.2 Reconstruction d'un objet multi-plan

Nous appliquons le processus décrit ci-dessus sur un objet multi-plan. Avec une seule image RMC, l'objet est reconstruit dans sa totalité de manière dense. La figure 6.7 montre quatre vues de ce modèle 3D, obtenues à partir de l'image. Chaque miroir fait 60 cm de côté et la caméra est placée à 1,25 m de l'objet (voir figure 6.7.a).

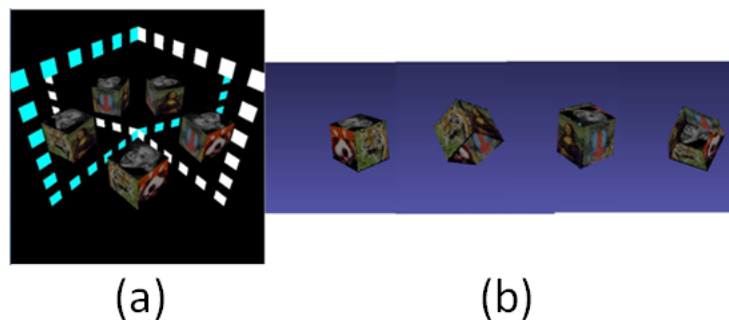


Figure 6.7 – Reconstruction 3D du cube.

L'histogramme (fig.6.8) des erreurs de reconstruction montre qu'on peut reconstruire, avec une probabilité de 78%, un point en 3D avec une erreur de 0,1 mm au plus .

En utilisant un objet composé de cinq surfaces planaires, de cotés égaux à 10 cm, nous

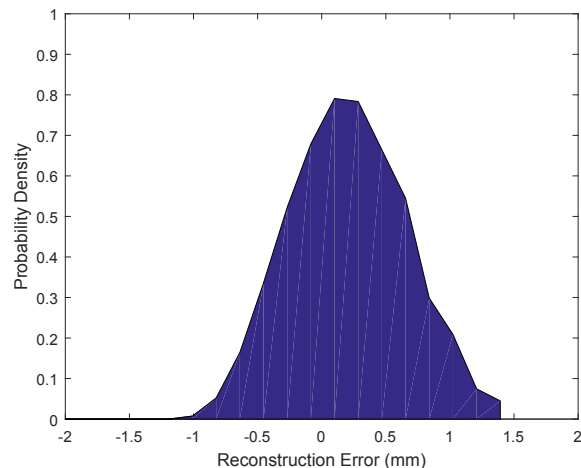


Figure 6.8 – Analyse des erreurs de reconstruction : densité de probabilité des erreurs de reconstruction des points 3D par notre méthode par rapport à la vérité terrain.

avons pu reconstruire 28728 points 3D. La moyenne de l'erreur de reprojection (distance euclidienne entre les points mesurés et les points reprojétés) en fonction du nombre de points utilisés est inférieure à 3 pixels. L'écart-type pour les erreurs de reconstruction entre la méthode proposée et la vérité terrain est 0.1743 cm. L'amélioration de la détection permettra une reconstruction des plans plus proche de la réalité.

6.2.1.3 Objets de formes géométriques variées

Nous proposons dans cette section des expérimentations sur d'autres objets. Dans un premier temps, nous allons travailler sur une sphère sur laquelle une image texturée a été plaquée pour tester la robustesse de la mesure de similarité. À chaque itération de reconstruction du modèle, nous obtenons généralement entre 500 et 1000 points du modèle à reconstruire.

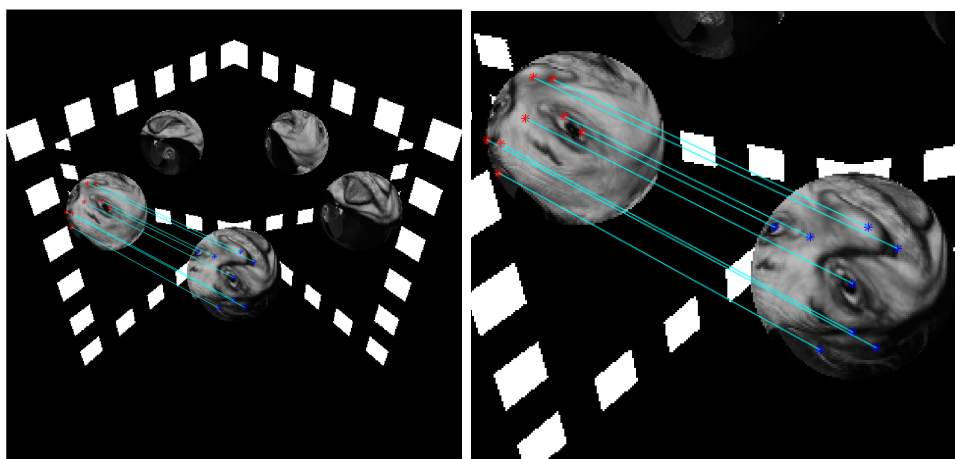


Figure 6.9 – Mise en correspondance entre les points de deux vues réelle/virtuelle par l'homographie induite par le plan

La figure 6.10 montre la reconstruction complète de ce même objet. Pour évaluer la mé-

thode de reconstruction de la sphère à partir d'une image RMC, nous fixons les quatre groupes contenant les différentes paires de vues. Plus les ensembles de points estimés sont grands, plus la reconstruction sphère est dense.

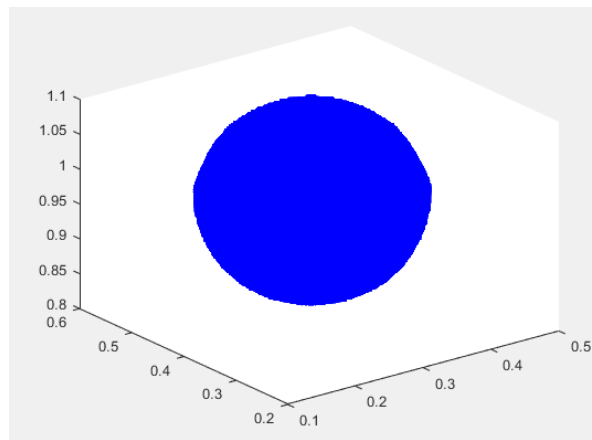


Figure 6.10 – Reconstruction de la sphere

Nous avons ensuite appliqué la méthode à un objet Théière. La figure 6.11 représente les nuages de points 3D obtenus superposés à la vérité terrain de l'objet. On peut observer que le système proposé a permis de reconstruire des points sur le bout de la partie de l'objet. Sur cette série, on obtient une précision de reconstruction de l'ordre de 0.42mm.

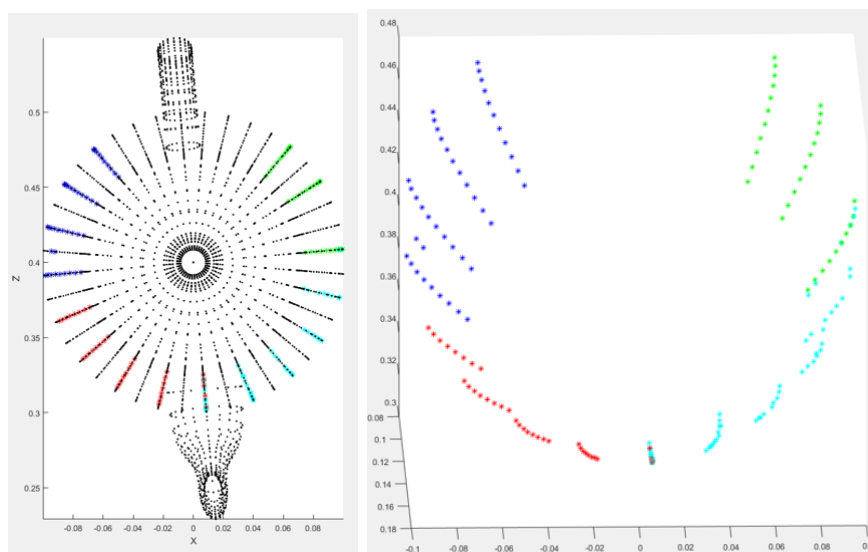


Figure 6.11 – Reconstruction éparsée de la théière pour des images 512×512 .

Ces résultats valident donc l'approche proposée et répondent à l'une des problématiques majeure de la thèse : il est possible d'obtenir le modèle 3D dense d'un objet contenant des zones inaccessibles grâce à des miroirs incluant les inter-réflexions.

6.2.2 Evaluation de la précision de la reconstruction 3D

Pour estimer la qualité de la reconstruction, nous proposons d'estimer la précision en utilisant les données du simulateur comme vérité terrain. La figure 6.12 présente les deux plans ainsi que le résultat de leur appariement. Sur l'image la plus à droite, le plan coloré en rouge correspond à la vérité terrain et le plan estimé par notre approche est coloré en bleu. Notons que le plan reconstruit est mat ce qui explique la bonne reconstruction que nous pouvons remarquer.

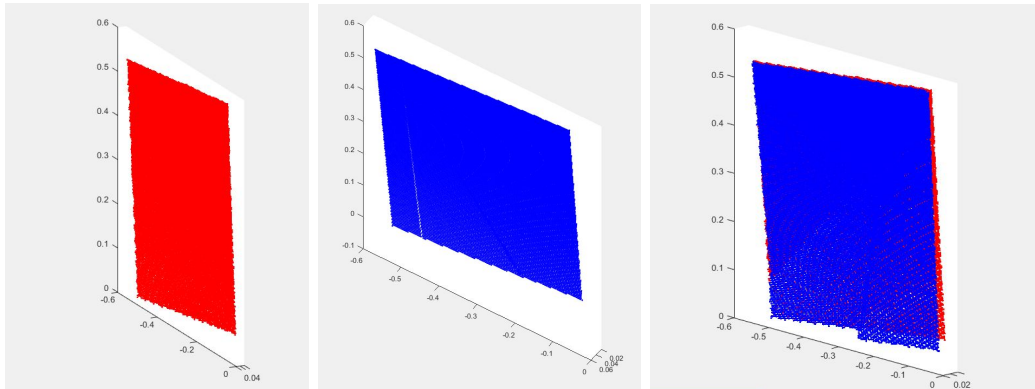


Figure 6.12 – Nuage de points 3D du plan reconstruit par notre approche et celui obtenu par le simulateur : Le premier plan à gauche est reconstruit par la vérité terrain, le second est reconstruit par l'approche proposée.

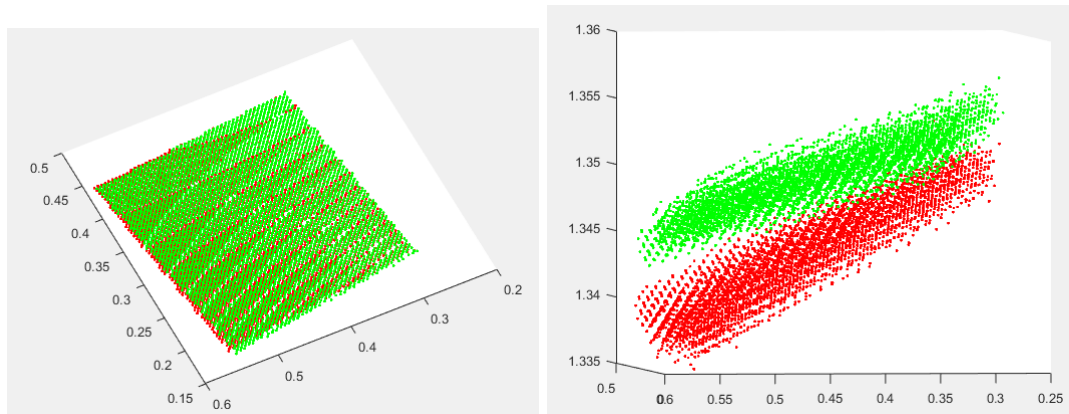


Figure 6.13 – Représentation des deux plans (reconstruits et vérité) dans le même repère.

La figure 6.13 montre les deux plans reconstruits. Le nuage de points 3D en vert est reconstruit par notre méthode et le nuage en rouge est la vérité terrain. Sur des images de résolution 512×512 , l'erreur est inférieure à un millimètre. Cependant, lorsque la résolution devient plus faible, l'erreur de reconstruction augmente rapidement.

La précision globale est estimée via la racine carrée de l'erreur quadratique moyenne :

$$REQM = \sqrt{\frac{1}{n} \sum_i (\hat{x}_i^{(k)} - x_i)^2} \quad (6.6)$$

avec n le nombre de paramètres estimés, $\hat{x}_i^{(k)}$ le i ème paramètre calculé à l'itération k et x_i le paramètre exact associé.

Dans le tableau 6.1, nous avons comparé notre méthode avec la vérité terrain pour des images ayant différentes résolutions. Ainsi, sur l'exemple de la figure 6.13, nous obtenons une erreur maximale de 0.42mm pour une résolution de 512×512 , et une erreur maximale de 0.28mm pour une résolution de 2048×2048 .

| résolution | 256×256 | 512×512 | 1024×1024 | 2048×2048 |
|-----------------|------------------|------------------|--------------------|--------------------|
| erreur maximale | 0.9mm | 0.42mm | 0.33mm | 0.28mm |

Tableau 6.1 – Evolution de la précision en fonction de la résolution des images.

6.2.3 Comparaison avec la méthode des silhouettes

Nous avons comparé notre méthode avec la méthode de Forbes *et al.* (2006). Pour cela, nous considérons que chaque image RMC dispose d'un ensemble de cinq silhouettes d'un même objet 3D extraits des images de différents points de vue comme illustré dans la figure 6.14. Nous voulons vérifier, à partir de l'exactitude de la segmentation des silhouettes et le degré de cohérence entre les silhouettes, la qualité de la reconstruction 3D par la méthode de Forbes par rapport à la nôtre.

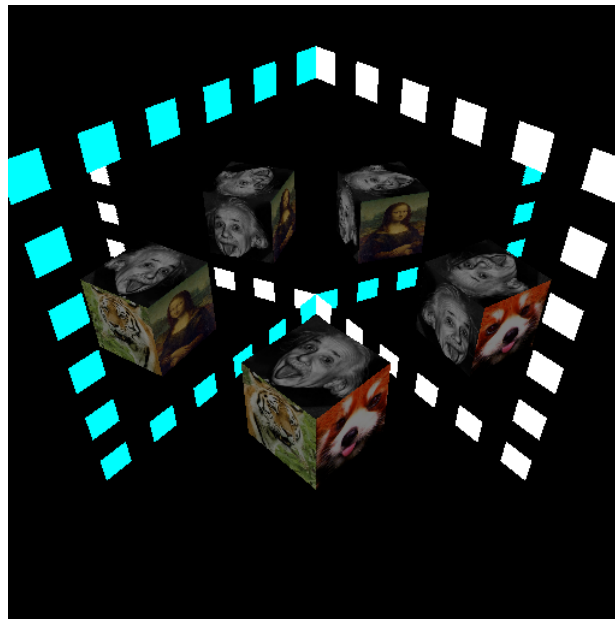


Figure 6.14 – Exemple d'une image catadioptrique d'un seul objet plan

Nous commençons par l'objet "Cube" en utilisant deux images catadioptriques. Pour exécuter l'algorithme de Forbes, nous utilisons un masque binaire pour avoir une bonne segmentation des silhouettes comme illustré dans la figure 6.15. Après, Nous exploitons les droites bitangentes comme montré dans la figure 6.16 pour estimer les auto-épipoles et les paramètres de la caméra afin d'obtenir une modélisation 3D d'objet. Finalement, nous affichons le maillage 3D obtenu par Forbes. La figure 6.17 montre la reconstruction partielle

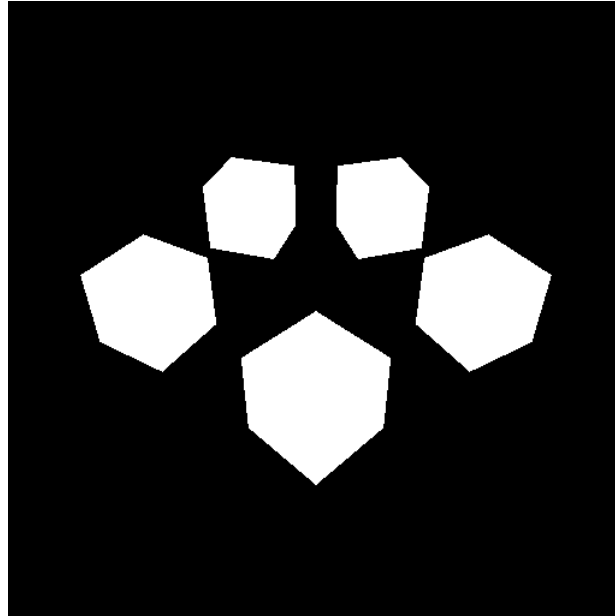


Figure 6.15 – Exemple d'une image binaire

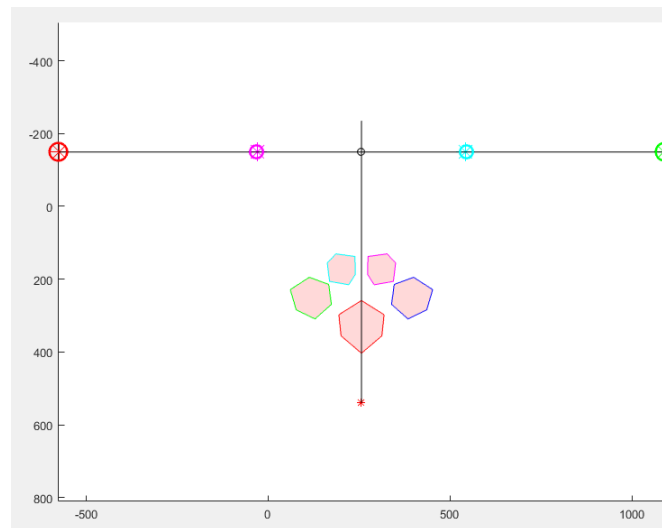


Figure 6.16 – L'ensemble de contour de la silhouette obtenus par les points de tangence épipolaire

du cube sous différents angles. Nous remarquons que les faces du cube 3D reconstruit ne sont pas planes. Même avec une quinzaine de vues différentes, l'algorithme n'arrive pas à aboutir une bonne reconstruction. Une comparaison visuelle entre la reconstruction par la méthode de Forbes et notre méthode illustrée dans la figure 6.17, montre que notre approche préserve la forme de l'objet et donne une bonne qualité de reconstruction.

La figure 6.18 illustre le cas d'un objet plan reconstruit avec les trois méthodes différentes. Le nuage en bleu est reconstruit par la méthode de Forbes. Le nuage de points 3D vert est reconstruit par notre méthode et le nuage en rouge représente la vérité terrain. Comme nous le voyons, l'erreur de reconstruction est élevée et supérieure au millimètre. Le nombre de points 2D du plan utilisé est 9632 points. Le nuage de points 3D reconstruit par notre méthode contient le même nombre de points 3D. Par contre, le nuage de points 3D reconstruit par la méthode de Forbes est 58428 points, ce qui ne permet pas de comparer les

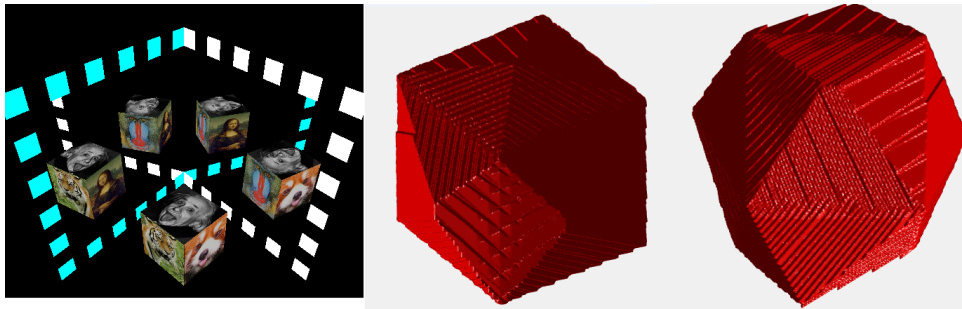


Figure 6.17 – Un exemple de maillage reconstruit à partir de deux images catadioptriques prises de différents points de vue

deux modèles 3D. Nous pensons qu'il serait particulièrement intéressant de combiner la méthode de Forbes avec une technique de reconstruction 3D par la mise en correspondance dense.

Nous avons utilisé un seul plan et plusieurs plans pour identifier et mesurer facilement la précision de notre approche. Les différents exemples de reconstruction présentés dans cette expérimentation, illustrent bien la robustesse et la précision de la méthode proposée.

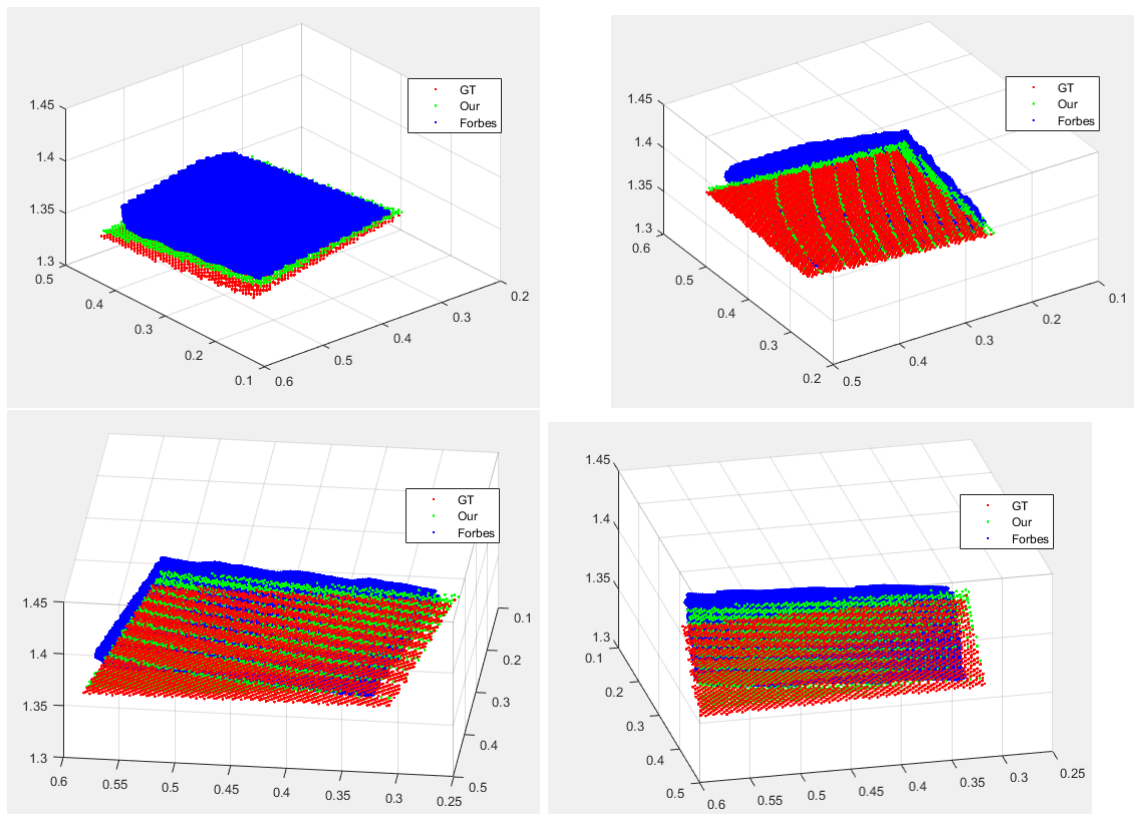


Figure 6.18 – L'affichage des différents nuage de points 3D sur le même repère.

6.3 Conclusion

Les différents exemples de reconstruction présentés ici, illustrent la robustesse et la précision de la méthode. Plusieurs extensions de cette approche sont possibles. Dans un premier temps, il serait intéressant, d'introduire une méthode de propagation invariante à toutes les transformations apportées par la réflexion de miroir. Nous envisageons d'étendre l'étude sur des objets de forme quelconque. Ensuite, il faudrait valider ce processus sur des images réelles en appliquant une étape de traitement comme nous l'avons effectué sur des images synthétiques pour la reconstruction 3D complète avec toutes les données possibles.

CONCLUSION GÉNÉRALE ET PERSPECTIVES

7.1 Bilan des travaux réalisés

Le présent mémoire a permis de traiter le problème de la reconstruction 3D complète des objets sans recul. Nous avons présenté l'ensemble des étapes pour réaliser la reconstruction 3D dense à partir d'un objet réel et ses quatre inter-réflexions. Le contexte applicatif est celui de la numérisation d'espaces exigus, comme les côtés et les dos des statues. Nous proposons d'aborder le problème sous l'angle de la vision catadioptrique. L'idée fondamentale est de réaliser l'acquisition d'image des côtés et du dos de formes 3D en disposant judicieusement des miroirs dont les reflets permettront de reconstruire la forme par photogrammétrie. L'extension de la photogrammétrie à ce système de vision passe par son étude géométrique et la conception d'une nouvelle méthode de mise en correspondance de points d'intérêt. En premier lieu, nous avons proposé un nouveau système qui rend le calibrage plus simple et fiable. La méthode de calibration que nous avons proposé permet d'estimer les paramètres intrinsèques et extrinsèques d'un réseau de caméras virtuelles. Après, nous avons proposé une matrice fondamentale réflexive simple et précise. Ainsi, nous avons montré que cette matrice permet un gain en précision et robustesse par rapport aux autres méthodes.

Ensuite, Dans l'optique d'une solution de reconstruction 3D basée sur la triangulation des correspondances denses, nous avons proposé un nouveau pipeline utilisant la mise en correspondance dense d'images. Nous avons pour cela mis en place une méthode de la détection et d'appariement de points caractéristiques dans l'image de multiples réflexions. Cette approche permet de construire un descripteur invariant à l'échelle, à la rotation, aux transformations affines et à la réflexion miroir. Ces correspondances éparses sont robustes en tenant compte des déformations fortes entre l'objet réel et ses quatre inter-réflexions. Ces descripteurs discriminants peuvent rendre les correspondances éparses très robustes tout en filtrant les correspondances erronées par une adaptation du filtre RANSAC. Nous nous sommes ensuite intéressés aux contraintes géométriques qui pourraient déterminer des correspondances dense. Nous avons alors, estimé des transformations locales des patches associés aux points d'intérêt pour permettre une meilleure mise en correspondance pour de nombreuses zones planaires distribuées dans l'objet entier, et fournir des correspondances quasi-densément réparties. Pour ce faire, nous avons proposé une méthode basée sur l'homographie induite par le plan observé entre une vue de l'objet et sa réflexion parmi les multiples réflexions. La transformation projective qui permet de passer d'un patch réel vers son similaire virtuel se fait par une minimisation de l'erreur photométrique en utilisant une méthode d'optimisation Branch-and-Bound entre les voisinages d'un point concerné. Ce qui nous a permis de détecter de manière optimale la bonne position des patches virtuels de l'objet réel dans une région candidate à des positions choisies (les points caractéristiques). Finalement, nous avons exploité les méthodes basées sur la triangulation classique pour

avoir un nuage de points dense représentant la surface 3D de l'objet.

Nous avons consacré le dernier chapitre de ce mémoire à la partie expérimentale. Pour cette validation, nous avons utilisé des objets de différentes formes et tailles. Nous avons effectué une reconstruction à partir de cet objet réel et ses inter-réflexions. Ces expérimentations ont montré que la reconstruction 3D dense des objets sans recul fonctionne de manière précise et donne des résultats très satisfaisants.

7.2 Quelques perspectives

Plusieurs perspectives et pistes de recherche peuvent être envisagées dans ce travail. Nous pensons que notre recherche sur l'utilisation de miroirs conjointement avec une seule caméra offre des résultats concluants et intéressants. Nous pensons que les miroirs pourraient potentiellement être bénéfiques à toutes les autres techniques de reconstruction 3D. La solution viendra probablement de l'assemblage de plusieurs techniques et, selon nous, l'utilisation de miroirs pourrait faire partie de cet assemblage.

Une autre piste intéressante à explorer et qui s'inscrit dans la continuité de ce travail, consiste à segmenter automatiquement l'objet réel et ses quatre inter-réflexions.

D'autres solutions sont envisageables pour améliorer la reconstruction 3D dense. Parmi les perspectives actuelles de ce travail, on pourra exploiter ces points 3D obtenus par photogrammétrie et qui seront combinés aux points obtenus par lasergrammétrie afin d'aboutir au développement d'une méthode de reconstruction 3D dense, complète et fidèle à l'architecture réelle de la scène.



BIBLIOGRAPHIE

- BASTOS, R. et STÜRZLINGER, W. (1998). Forward mapped planar mirror reflections. *University of North Carolina at Chapel Hill, Computer Science Technical Report TR98-206*.
- BOUGUET, J.-Y. (2004). Camera calibration toolbox for matlab. http://www.vision.caltech.edu/bouguetj/calib_doc/index.html.
- FISCHLER, M. A. et BOLLES, R. C. (1981). Random sample consensus : a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395.
- FORBES, K., NICOLLS, F., DE JAGER, G. et VOIGT, A. (2006). Shape-from-silhouette with two mirrors and an uncalibrated camera. *In European Conference on Computer Vision*, pages 165–178. Springer.
- FUCHS, M., KÄCHELE, M. et RUSINKIEWICZ, S. (2013). Design and fabrication of faceted mirror arrays for light field capture. *In Computer Graphics Forum*, volume 32, pages 246–257. Wiley Online Library.
- FUJIYAMA, S., SAKAUE, F. et SATO, J. (2010). Multiple view geometries for mirrors and cameras. *In Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 45–48. IEEE.
- GARBACZ, P. et GIESKO, T. (2010). Concept of binocular stereo vision system for inspection of surfaces. *Problemy Eksploatacji*, pages 193–205.
- GLUCKMAN, J. et NAYAR, S. K. (1999). Planar catadioptric stereo : Geometry and calibration. *In Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 1, pages 22–28. IEEE.
- GLUCKMAN, J. et NAYAR, S. K. (2001). Catadioptric stereo using planar mirrors. *International Journal of Computer Vision*, 44(1):65–79.
- GLUCKMAN, J. et NAYAR, S. K. (2002). Rectified catadioptric stereo sensors. *IEEE transactions on pattern analysis and machine intelligence*, 24(2):224–236.
- GOSHTASBY, A. et GRUVER, W. A. (1993). Design of a single-lens stereo camera system. *Pattern Recognition*, 26(6):923–937.
- GUO, X. et CAO, X. (2010). Find : A neat flip invariant descriptor. *In Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 515–518. IEEE.
- GUO, X. et CAO, X. (2012). Mift : A framework for feature descriptors to be mirror reflection invariant. *Image and Vision Computing*, 30(8):546–556.

- GUO, X., CAO, X., ZHANG, J. et LI, X. (2009). Mift : A mirror reflection invariant feature descriptor. *In Asian Conference on Computer Vision*, pages 536–545. Springer.
- GURDJOS, P., STURM, P. et WU, Y. (2006). Euclidean structure from $n \geq 2$ parallel circles : theory and algorithms. *In European Conference on Computer Vision*, pages 238–252. Springer.
- HARRIS, C. et STEPHENS, M. (1988). A combined corner and edge detector. *In Alvey vision conference*, volume 15, pages 10–5244. Citeseer.
- HARTLEY, R. et ZISSERMAN, A. (2003). *Multiple view geometry in computer vision*. Cambridge university press.
- HARTLEY, R. I. et STURM, P. (1997). Triangulation. *Computer vision and image understanding*, 68(2):146–157.
- HAYASHI, N., TOMIZAWA, T., SUEHIRO, T. et KUDOH, S. (2013). Stereo calibration method using mirrored images containing a camera. *In Robotics and Biomimetics (ROBIO), 2013 IEEE International Conference on*, pages 1420–1425. IEEE.
- HECHT, E. et ZAJAC, A. (1974). Optics addison-wesley. *Reading, Mass*, 19872:350–351.
- HESCH, J. A., MOURIKIS, A. I. et ROUMELIOTIS, S. I. (2008). Determining the camera to robot-body transformation from planar mirror reflections. *In Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 3865–3871. IEEE.
- HU, B. (2009). It’s all done with mirrors : Calibration-and-correspondence-free 3d reconstruction. *In Computer and Robot Vision, 2009. CRV’09. Canadian Conference on*, pages 148–154. IEEE.
- HU, B., BROWN, C. et NELSON, R. (2005). Multiple-view 3-d reconstruction using a mirror. *Technical Report TR863, Computer Science Dept*.
- HUANG, P.-H. et LAI, S.-H. (2006). Contour-based structure from reflection. *In Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 379–386. IEEE.
- IHRKE, I., RESHETOUSKI, I., MANAKOV, A., TEVS, A., WAND, M. et SEIDEL, H.-P. (2012). A kaleidoscopic approach to surround geometry and reflectance acquisition. *In Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 29–36. IEEE.
- INOSHITA, C., TAGAWA, S., MANNAN, M. A., MUKAIGAWA, Y. et YAGI, Y. (2013). Full-dimensional sampling and analysis of bssrdf. *Information and Media Technologies*, 8(4): 1105–1109.
- KAZMI, W. et ANDERSEN, H. J. (2015). A comparison of interest point and region detectors on structured, range and texture images. *Journal of Visual Communication and Image Representation*, 32:156–169.
- KIM, J.-S., GURDJOS, P. et KWEON, I.-S. (2005). Geometric and algebraic constraints of projected concentric circles and their applications to camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):637–642.

- KIM, M.-H., PARK, J. et KWEON, I. S. (2014). Rgb-d sensor and mirrors : A practical setup for 3d reconstruction of dynamic objects. *In Ubiquitous Robots and Ambient Intelligence (URAI), 2014 11th International Conference on*, pages 651–652. IEEE.
- KORMAN, S., REICHMAN, D., TSUR, G. et AVIDAN, S. (2013). Fast-match : Fast affine template matching. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2331–2338.
- KUMAR, R. K., ILIE, A., FRAHM, J.-M. et POLLEFEYS, M. (2008). Simple calibration of non-overlapping cameras with a mirror. *In Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–7. IEEE.
- LANMAN, D., CRISPELL, D. et TAUBIN, G. (2007). Surround structured lighting for full object scanning. *In 3-D Digital Imaging and Modeling, 2007. 3DIM'07. Sixth International Conference on*, pages 107–116. IEEE.
- LANMAN, D., CRISPELL, D. et TAUBIN, G. (2009). Surround structured lighting : 3-d scanning with orthographic illumination. *Computer Vision and Image Understanding*, 113(11):1107–1117.
- LÉBRALY¹³, P., AIT-AIDER¹³, O., ROYER²³, E. et DHOME¹³, M. (2010). Calibration of non-overlapping cameras-application to vision-based robotics. BMVC.
- LEPETIT, V. et FUA, P. (2006). Keypoint recognition using randomized trees. *IEEE transactions on pattern analysis and machine intelligence*, 28(9):1465–1479.
- LEVOY, M., CHEN, B., VAISH, V., HOROWITZ, M., MCDOWALL, I. et BOLAS, M. (2004). Synthetic aperture confocal imaging. *In ACM Transactions on Graphics (ToG)*, volume 23, pages 825–834. ACM.
- LIN, I.-C., YEH, J.-S. et OUHYOUNG, M. (2002). Extracting 3d facial animation parameters from multiview video clips. *IEEE Computer Graphics and Applications*, 22(6):72–80.
- LONG, G., KNEIP, L., LI, X., ZHANG, X. et YU, Q. (2015). Simplified mirror-based camera pose computation via rotation averaging. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1247–1255.
- LOWE, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110.
- MA, R., CHEN, J. et SU, Z. (2010). Mi-sift : mirror and inversion invariant generalization for sift descriptor. *In Proceedings of the ACM International Conference on Image and Video Retrieval*, pages 228–235. ACM.
- MARCHAND, E. et CHAUMETTE, F. (2017). Visual servoing through mirror reflection. *In Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 3798–3804. IEEE.
- MARIOTTINI, G. L., SCHEGGI, S., MORBIDI, F. et PRATTICHIZZO, D. (2009). Planar catadioptric stereo : single and multi-view geometry for calibration and localization. *In Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 1510–1515. IEEE.
- MARIOTTINI, G. L., SCHEGGI, S., MORBIDI, F. et PRATTICHIZZO, D. (2012). Planar mirrors for image-based robot localization and 3-d reconstruction. *Mechatronics*, 22(4): 398–409.

- MARTINS, N. et DIAS, J. (2004). Camera calibration using reflections in planar mirrors and object reconstruction using volume carving method. *The Imaging Science Journal*, 52(2):117–130.
- MIKOLAJCZYK, K., TUYTELAARS, T., SCHMID, C., ZISSERMAN, A., MATAS, J., SCHAFFALITZKY, F., KADIR, T. et VAN GOOL, L. (2005). A comparison of affine region detectors. *International journal of computer vision*, 65(1-2):43–72.
- MITSUMOTO, H., TAMURA, S., OKAZAKI, K., KAJIMI, N. et FUKUI, Y. (1992). 3-d reconstruction using mirror images based on a plane symmetry recovering method. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (9):941–946.
- MOISAN, L. et STIVAL, B. (2004). A probabilistic criterion to detect rigid point matches between two images and estimate the fundamental matrix. *International Journal of Computer Vision*, 57(3):201–218.
- MOREL, J.-M. et YU, G. (2009). Asift : A new framework for fully affine invariant image comparison. *SIAM journal on imaging sciences*, 2(2):438–469.
- NENE, S. A. et NAYAR, S. K. (1998). Stereo with mirrors. In *Computer Vision, 1998. Sixth International Conference on*, pages 1087–1094. IEEE.
- NGUYEN, T.-N., HUYNH, H.-H. et MEUNIER, J. (2018). 3d reconstruction with time-of-flight depth camera and multiple mirrors. *IEEE Access*, 6:38106–38114.
- NOBUHARA, S., KASHINO, T., MATSUYAMA, T., TAKEUCHI, K. et FUJII, K. (2016). A single-shot multi-path interference resolution for mirror-based full 3d shape measurement with a correlation-based tof camera. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 342–350. IEEE.
- PANAHANDEH, G. et JANSSON, M. (2011). Imu-camera self-calibration using planar mirror reflection. In *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on*, pages 1–7. IEEE.
- RADU, O. *et al.* (2006). *Catadioptric stereo based on structured light projection*. Universitat de Girona.
- RESHETOUSKI, I., MANAKOV, A., SEIDEL, H.-P. et IHRKE, I. (2011). Three-dimensional kaleidoscopic imaging. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 353–360. IEEE.
- RODRIGUES, R., BARRETO, J. P. et NUNES, U. (2010). Camera pose estimation using images of planar mirror reflections. In *European Conference on Computer Vision*, pages 382–395. Springer.
- SALVI, J., PAGES, J. et BATLLE, J. (2004). Pattern codification strategies in structured light systems. *Pattern recognition*, 37(4):827–849.
- SEMPLE, J. G. et KNEEBONE, G. T. (1998). *Algebraic projective geometry*. Oxford University Press.
- SEN, P., CHEN, B., GARG, G., MARSCHNER, S. R., HOROWITZ, M., LEVOY, M. et LENSCH, H. (2005). Dual photography. In *ACM Transactions on Graphics (TOG)*, volume 24, pages 745–755. ACM.

- SHEN, H.-L. et CHENG, Y. (2011). Calibrating light sources by using a planar mirror. *Journal of Electronic Imaging*, 20(1):013002.
- SONG, J., LU, X., WANG, W. et CHEN, C. (2016). Isift : Improving the performance of sift for mirror images. In *Computer and Communications (ICCC), 2016 2nd IEEE International Conference on*, pages 742–745. IEEE.
- STURM, P. et BONFORT, T. (2006). How to compute the pose of an object without a direct view? In *Asian Conference on Computer Vision*, pages 21–31. Springer.
- SU, M., MA, Y., ZHANG, X., WANG, Y. et ZHANG, Y. (2017). Mbr-sift : A mirror reflected invariant feature descriptor using a binary representation for image matching. *PLoS one*, 12(5):e0178090.
- TAGAWA, S., MUKAIGAWA, Y. et YAGI, Y. (2012). 8-d reflectance field for computational photography. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 2181–2185. IEEE.
- TAHARA, T., KAWAHARA, R., NOBUHARA, S. et MATSUYAMA, T. (2015). Interference-free epipole-centered structured light pattern for mirror-based multi-view active stereo. In *3D Vision (3DV), 2015 International Conference on*, pages 153–161. IEEE.
- TAN, K.-H., HUA, H. et AHUJA, N. (2004). Multiview panoramic cameras using mirror pyramids. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(7):941–946.
- TRIGGS, B., McLAUCHLAN, P. F., HARTLEY, R. I. et FITZGIBBON, A. W. (1999). Bundle adjustment—a modern synthesis. In *International workshop on vision algorithms*, pages 298–372. Springer.
- WU, H.-H. P. (2007). Rectification of stereoscopic video for planar catadioptric stereo systems. *IEEE transactions on circuits and systems for video technology*, 17(6):686–698.
- XIE, L., TIAN, Q. et ZHANG, B. (2014). Max-sift : Flipping invariant descriptors for web logo search. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 5716–5720. IEEE.
- XU, Z., WANG, Y. et YANG, C. (2015). Multi-camera global calibration for large-scale measurement based on plane mirror. *Optik-International Journal for Light and Electron Optics*, 126(23):4149–4154.
- YING, X., PENG, K., HOU, Y., GUAN, S., KONG, J. et ZHA, H. (2013). Self-calibration of catadioptric camera with two planar mirrors from silhouettes. *IEEE transactions on pattern analysis and machine intelligence*, 35(5):1206–1220.
- YING, X., PENG, K., REN, R. et ZHA, H. (2010). Geometric properties of multiple reflections in catadioptric camera with two planar mirrors. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1126–1132. IEEE.
- ZHANG, H., SHAO, L. et WONG, K.-Y. K. (2012). Self-calibration and motion recovery from silhouettes with two mirrors. In *Asian Conference on Computer Vision*, pages 1–12. Springer.

- ZHANG, Z. (1999). Flexible camera calibration by viewing a plane from unknown orientations. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, pages 666–673. Ieee.
- ZHAO, Y. et LI, Y. (2017). Camera self-calibration from projection silhouettes of an object in double planar mirrors. *JOSA A*, 34(5):696–707.
- ZHAO, Y., ZHANG, X., WANG, Y. et XU, L. (2016). Camera self-calibration based on circular points with two planar mirrors using silhouettes. *Multimedia Tools and Applications*, 75(13):7981–7997.
- ZHONG, H., SZE, W. et HUNG, Y. (2006). Reconstruction from plane mirror reflection. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 1, pages 715–718. IEEE.



LISTE DES PUBLICATIONS

Conférence : ICIP - International Conference on Image Processing

Titre : AMIFT : AFFINE-MIRROR INVARIANT FEATURE TRANSFORM

Auteurs : Nouredine Mohtaram, Amina Radgui, Guillaume Caron, El Mustapha Mouaddib

Année : 2018

Résumé : In this paper, we propose a descriptor for image matching under multiple mirror reflections. Indeed, existing adaptations of SIFT for the mirror transformation are not successful when object and mirrors orientations are not constrained. Hence, we propose to combine MIFT and Affine-SIFT descriptors as the Affine Mirror Invariant Feature Transform (AMIFT). The experimental results and given evaluation show that our proposed descriptor outperforms MIFT and ASIFT on both synthetic and real images datasets.

Conférence : ORASIS- Congrès des jeunes chercheurs en vision par ordinateur

Titre : Reconstruction 3D d'objet à partir d'une image catadioptrique multi-plan

Auteurs : Nouredine Mohtaram, Amina Radgui, Guillaume Caron, El Mustapha Mouaddib, Driss Aboutajdine

Année : 2017

Résumé : L'étude porte sur un Système Stéréo Catadioptrique multiPlan (SSCP) constitué d'une unique caméra perspective et deux miroirs plans judicieusement positionnés. En positionnant un objet entre les miroirs plans et la caméra, cinq vues en sont observées simultanément. Dans cet article, nous abordons la modélisation et l'étalonnage géométrique de SSCP. Les points d'intérêt détectés dans la vue directe servent de référence pour la recherche de correspondances sous la contrainte d'une homographie minimisant l'erreur photométrique entre voisinages. La correspondance mène à la reconstruction 3D des points dont les résultats démontrent la qualité de la méthode de recherche de correspondances et que la reconstruction 3D avec une image d'un SSCP est possible.



RÉSUMÉ

Ce travail de thèse s'intéresse au problème de la reconstruction 3D dense complète d'objets sans recul. Nous avons conçu et développé un système de reconstruction tridimensionnelle d'objet réel basé caméra avec deux miroirs plans : c'est un système Stéréo Catadioptrique Planaire (SCP). En premier lieu, nous modélisons le système SCP par un réseau de caméras virtuelles afin de l'étalonner. Ensuite, nous formulons la mise en correspondance des points caractéristiques détectés dans les images reflétés en utilisant une variante de la méthode ASIFT qui est adaptée aux miroirs plans et que l'on dénomme AMIFT. L'élimination des fausses correspondances est assurée par la méthode Symmetric-RANSAC proposée dans cette thèse. Pour reconstruire la surface de l'objet, et non quelques points épars, une étape de mise en correspondance dense est par la suite nécessaire. Nous proposons alors une approche de recherche de pixels correspondants intégrant la géométrie projective par le biais d'homographies locales. Cette méthode permet d'estimer la transformation géométrique qui lie l'image de l'objet à l'une de ses inter-réflexions sur les miroirs plans en minimisant une fonction de coût avec la technique d'optimisation de Branch-and-Bound. Cela nous permet d'adapter la reconstruction 3D dense, fondamentalement basée sur la triangulation de correspondances entre images. Enfin, nous appliquons ce pipeline de reconstruction 3D sur des images de réflexions multiples afin de vérifier l'hypothèse de reconstruction 3D complète à partir d'un système SCP. La performance du système proposé est validée par des expérimentations sur des images synthétiques et les résultats obtenus montrent la qualité de la reconstruction 3D.

Mots-clés : Système catadioptrique planaire, étalonnage, caméras virtuelles, géométrie épipolaire, mise en correspondance, reconstruction 3D.

Abstract

This PhD work focuses on the problem of complete dense 3D reconstruction of objects without recoil. We have conceived and developed a tridimensional reconstruction system of real objects based on a camera with two planar mirrors; baptised as a Planar Catadioptric Stereo (PCS) system. We first model the PCS system by a network of virtual cameras to perform calibration. Then, we formulate the problem of characteristic points' correspondences detected on the reflected images by using a variant of the ASIFT method. This adaptation which we behold as AMIFT on the mirror planes. Putative point correspondences are further refined with outlier rejection using our method of Symmetric RANSAC proposed in this thesis. To reconstruct a proper dense object surface, and not just a sparse projection of points, a dense correspondence technique is consequently required. This method estimates the geometric transformation linking the image object with one of it's inter reflections on the mirror planes by minimising a branch and bound cost function. This allows us to adapt the 3D dense reconstruction, fundamentally based on the triangulation of image correspondences. We apply this 3D reconstruction pipeline on multiple catadioptric images in order to verify the undermining hypothesis using a PCS system. Our methodology is validated using experimental results on synthetic images to illustrate the quality of the 3D reconstruction.

Keywords : Planar catadioptric system, calibration, virtual cameras, epipolar geometry, matching, 3D reconstruction.