



HAL
open science

Reconstruction 3D et production de carte dense de disparité en stéréovision non-alignée pour des applications industrielles de localisation 3D et d'analyse de surface

Jimmy Pelcat

► **To cite this version:**

Jimmy Pelcat. Reconstruction 3D et production de carte dense de disparité en stéréovision non-alignée pour des applications industrielles de localisation 3D et d'analyse de surface. Vision par ordinateur et reconnaissance de formes [cs.CV]. INSA de Rouen, 2012. Français. NNT : 2012ISAM0025 . tel-01821171

HAL Id: tel-01821171

<https://theses.hal.science/tel-01821171>

Submitted on 22 Jun 2018

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.



École Doctorale de Sciences Physiques et Mathématiques pour l'Ingénieur
THÈSE DE DOCTORAT DE L'INSA DE ROUEN

Discipline : Informatique
Spécialité : Systèmes de vision

présentée par

Jimmy Pelcat

pour obtenir le grade de

DOCTEUR DE L'INSA DE ROUEN

soutenue publiquement le 23 Février 2012

**Reconstruction 3D et production de carte dense de
disparité en stéréovision non-alignée pour des
applications industrielles de localisation 3D et
d'analyse de surface**

Jury :

Rapporteur	M. Fabrice MERIAUDEAU	Professeur, LE2I
Rapporteur	M. Majdi KHOUDEIR	Professeur, XLIM-SIC
Examineur	M. Xavier SAVIN	Directeur général, Visionic
Examineur	M. Jean-Luc CORRE	Ingénieur, NeoTec-Vision
Directeur de thèse	M. Abdelaziz BENSRAIR	Professeur, LITIS
Co-encadrant de thèse	M. Sébastien KRAMM	Docteur, LITIS

Résumé

En vision industrielle, de nombreuses applications de mesure et de contrôle qualité évoluent vers des problématiques tri-dimensionnelles. Les systèmes de stéréovision sont des solutions technologiques qui attirent les industriels par leur simplicité mécanique. Deux caméras statiques disposées à des endroits stratégiques peut s'avérer suffisantes pour répondre à cette problématique bien que les contraintes industrielles imposent de respecter des temps de traitement courts et des mesures précises. La diversité des applications nous amènent à envisager deux approches afin de répondre à deux types d'application.

La première technique consiste en la reconstruction 3D à partir de paires de points images qui se correspondent dans les deux images. Elle est destinée à répondre à la problématique de mesure 3D. Les méthodes de calibration monoculaire et de calcul 3D par triangulation sont la base de la reconstruction 3D. Nous étudions la précision de mesure et son évolution selon la pose du système de capture par rapport à la scène observée.

La seconde technique consiste à construire des images de disparité afin de répondre à des problématiques de construction de profil et d'analyse de défaut. La contrainte d'alignement des caméras, nécessaire pour accélérer le processus de mise en correspondance, implique d'utiliser des méthodes de calibration stéréoscopique et de rectification des images. Nous étudions l'impact de l'alignement sur la qualité de la rectification. La production de carte dense de disparité se base sur les techniques de stéréo-corrélation. Nous montrons les limites de l'utilisation d'un noyau de corrélation carré et proposons une alternative par production de deux cartes denses de disparité à partir de deux noyaux mono-directionnels, améliorant la mesure de disparité sur les zones de contours et d'occultations.

Mots Clés

Stéréovision, calibration, rectification, stéréo-corrélation, carte dense de disparité

Abstract

In industrial vision, many applications for measuring and quality control are moving to three-dimensional problems. Stereovision systems are technological solutions that attract industry by their mechanical simplicity. Two static cameras placed at strategic locations may be sufficient to address this problem although the industrial constraints imposed to respect a short processing time and precise measurements. The diversity of applications lead us to consider two approaches to resolve the two types of application.

The first technique consists in the 3D reconstruction from pairs of image points which correspond in both images. It is intended to address the problem of 3D measurement. The methods of monocular calibration and 3D triangulation are the basis of 3D reconstruction. We study the accuracy and its evolution according to the capture system pose compared to the observed scene.

The second technique is to construct disparity maps to address problems of building profile and defect analysis. The alignment constraint of cameras needed to accelerate the process of matching involves the use of methods of stereoscopic calibration and image rectification. We study the impact of alignment on the quality of the rectification. The production of dense disparity map is based on the stereo-correlation techniques. We show the limits of the use of a squared correlation kernel and propose an alternative production of two dense disparity maps from two mono-directional kernels, improving the measurement of disparity around edges and occlusions.

Keywords

Stereo vision, calibration, rectification, stereo correlation, dense disparity map

Remerciement

Je remercie tout d'abord M. Abdelaziz Bensrhair et M. Sébastien Kramm pour l'encadrement scientifique de cette thèse et pour les différents conseils pour sa bonne réalisation durant ces trois années.

Je remercie les rapporteurs MM. les professeurs Fabrice Mériaudeau et Majdi Khoudeir pour avoir accepté d'évaluer mon travail et pour les différentes remarques constructives pour la rédaction de ce manuscrit.

Je remercie la société Visionic pour avoir accepté ce projet scientifique dans leur contexte industriel. Je remercie, plus particulièrement, Xavier Savin, pour avoir accepté de représenter dans le jury l'utilisateur final des solutions apportées et pour avoir mis à ma disposition l'ensemble des ressources humaines et techniques pour mener à terme ce projet.

Je remercie Nicolas Balin, de Visionic, pour avoir fait le lien entre la problématique industrielle de Visionic et la problématique technique liée à sa réalisation.

Je remercie Florian Sella, de Visionic, pour son investissement global dans le projet, pour l'ensemble de ses conseils techniques, pour les différents échanges aussi bien scientifiques qu'amicaux.

Je remercie également MM. Gilles Da Palma et Baptiste Laroche, de Visionic, pour leur aide précieuse pour les différentes étapes de développement et de tests dans un environnement industriel mais surtout pour leur soutien et leur bonne humeur tout au long de la thèse.

Je remercie également la société NeoTec-Vision et plus particulièrement Jean-Luc Corre pour les différentes discussions constructives et pour avoir accepté de faire partie du jury.

Je remercie l'INSA de Rouen, l'école doctorale SPMII et l'ANRT pour avoir fait en sorte que ce projet puisse exister.

Je remercie l'ensemble des membres du LITIS. Je remercie les doctorants Bassem Besbes, Nadeen Salameh et Aminir Hadachi, le personnel administratif et technique, notamment Sandra Hague et Jean-François Brulard, et tous les autres membres que j'ai pu cotoyer par alternance.

Merci à tous ceux qui m'ont soutenu, dans le cadre familial et amical, et qui ont contribué à l'aboutissement de mes travaux de thèse.

Je terminerai avec une pensée toute particulière pour ma femme et mes deux enfants qui font que chaque jour est précieux.

Table des matières

1	Introduction générale	1
2	Problématique	3
2.1	La stéréovision	4
2.2	Les applications industrielles	5
2.2.1	Le "décaissage" et le "dévraccage"	5
2.2.2	Le recalage et la mesure 3D	7
2.2.3	Le contrôle d'aspect	8
2.3	Les solutions industrielles	9
2.3.1	La conoscopie holographique	9
2.3.2	La triangulation laser	10
2.3.3	La projection de franges	12
2.3.4	La stéréovision classique	14
2.3.5	La stéréovision par production de carte de disparité	16
2.3.6	La stéréovision par projection de franges	16
2.3.7	Le temps de vol	18
2.3.8	Avantages et inconvénients	20
2.4	Les deux approches dans leur contexte applicatif	21
2.4.1	Approche par reconstruction 3D	21
2.4.2	Approche par production de cartes de disparité	22
3	Reconstruction 3D Directe	25
3.1	Calibration monoculaire planaire	27
3.1.1	Définitions	28
3.1.2	Expression des paramètres	28
3.1.3	Estimation des paramètres	31
3.2	Sélection de points d'intérêt et appariement	32
3.3	Triangulation pour le calcul 3D	34
3.3.1	Conversion dans le repère monde	34

3.3.2	Calcul du point médian	36
3.4	Calibration spatiale	40
3.5	Expérimentation	42
3.5.1	Évaluation de la précision 3D	42
3.5.2	Influence de la précision 2D	47
4	Production de carte dense de disparité	53
4.1	Calibration stéréoscopique	55
4.2	Rectification des images	57
4.2.1	Expression algébrique de la problématique	58
4.2.2	Détermination des matrices de rectification	59
4.2.3	Évaluation de la rectification	64
4.3	Stéréo-corrélation	66
4.3.1	État de l'art	66
4.3.2	Appariement par stéréo-corrélation	66
4.3.3	Optimisation algorithmique	68
4.3.4	Amélioration qualitative	76
4.3.5	Implémentation	79
4.4	Experimentation	85
4.4.1	Projection de mouchetis	85
4.4.2	Correction de l'effet de dilatation	85
5	Évaluation pour des applications industrielles	93
5.1	Application de recalage 3D	95
5.1.1	Phase de calibration	95
5.1.2	Phase de production	96
5.1.3	Bilan	100
5.2	Application de décaissage	101
5.2.1	Phase de calibration	101
5.2.2	Cycle de production	103
5.2.3	Bilan	105
5.3	Application de contrôle d'aspect	107
5.3.1	Phase de calibration	107
5.3.2	Phase de production	107
5.3.3	Bilan	110
5.4	Application de contrôle de défaut	114
5.4.1	Acquisition	114
5.4.2	Phase de calibration	114

5.4.3	Analyse de défaut	115
5.4.4	Bilan	116
6	Conclusion	119
7	Annexes	121
7.1	Les capteurs de la technologie 3D	121
7.1.1	Capteurs de conoscopie holographique	121
7.1.2	Capteurs de triangulation laser	122
7.1.3	Capteurs avec projection de franges	123
7.1.4	Capteurs de stéréovision	123
7.1.5	Capteurs de stéréovision par projection de franges	124
7.1.6	Capteurs de temps de vol	125

Table des figures

2.1	Problématique du dévraccage par rapport au décaissage	5
2.2	Solution de palettisation par assistance robotique	6
2.3	Solution de recalage 3D	7
2.4	Solution de construction de profil de surface pour des composants électronique	8
2.5	Principe de la conoscopie holographie	9
2.6	Utilisation par déplacement linéaire	10
2.7	Schématisation de la triangulation laser	11
2.8	Exemple d'image, convertie sur 256 niveaux de gris, acquise avec un système de triangulation laser	12
2.9	Schématisation de la projection de franges	13
2.10	Conséquence de la surface de la pièce sur la déformation des franges	14
2.11	Schématisation de la stéréovision classique	15
2.12	Carte dense de disparité	17
2.13	Carte éparsse issue des travaux [Kra08] de Sébastien Kramm sur une scène autoroutière	17
2.14	Schématisation de la stéréovision par projection de franges	18
2.15	Schématisation de l'acquisition temps de vol	19
2.16	Triangulation	21
2.17	Notion de disparité pour un système aligné	23
3.1	Les différents repères	28
3.2	La relation caméra-image	29
3.3	La relation caméra-monde	30
3.4	Représentation de cas d'appariement problématiques	32
3.5	Segment qui minimise la distance inter-droite	34
3.6	Segment qui minimise la distance inter-droite	35
3.7	Le modèle de caméra	36
3.8	Représentation spatiale de la régression polynomiale	40
3.9	Expression des paramètres de positionnement des deux caméras	42

3.10	Éléments de calibration et de détection pour l'évaluation de la précision 3D	43
3.11	Évolution de l'erreur en distance (mm) pour le plan moyen	45
3.12	L'écart-type de l'erreur (mm) sur les trois axes et en distance pour deux résolutions de caméra	46
3.13	Évolution de l'écart-type de l'erreur (mm) sur les trois axes et en distance en fonction de l'angle moyen formé par les deux caméras et les points mesurés	47
3.14	Évolution de l'erreur (mm) en distance, pour trois plan, avec et sans régression	49
3.15	Évolution de l'écart-type de l'erreur (mm) sur les trois axes et en distance, avec et sans régression 3D	50
3.16	Évolution de l'écart-type de l'erreur (mm) en distance en fonction de l'erreur 2D, pour les deux résolutions de caméras, avec et sans régression 3D .	51
4.1	Construction du plan épipolaire	55
4.2	Conséquence du « Shearing »	62
4.3	Conséquences de la rectification : l'image rectifiée de gauche a des coordonnées négatives, la taille des deux images rectifiées est différente	63
4.4	Répartition de l'erreur de rectification - Les points verts représentent les erreurs inférieure au demi pixel, les points jaunes représentent les erreurs comprises entre 0.5px et 1px et les points rouges représentent les erreurs supérieures au pixel	65
4.5	Représentation des 27 comparaisons de fenêtre de corrélation de taille $k = 3$, dans lesquelles deux points fixes gauche et droite sont présents (les carrés rouges et jaunes correspondent respectivement à l'image de gauche et celle de droite)	69
4.6	Pseudo-algo du processus de production de carte dense de disparité	70
4.7	Initialisation de la première valeur puis propagation horizontale pour le calcul en ligne (cas où $d_i = 0$)	72
4.8	Initialisation de la première ligne puis propagation verticale pour le calcul en colonne	73
4.9	Mise à jour de l'image SSD et de la carte dense de disparité	74
4.10	Interpolation parabolique	77
4.11	Visualisation de l'effet de dilatation par rapport au contour supérieur des rouleaux en rouge	78
4.12	Justification de l'effet de dilatation avec 3 noyaux de taille différente	78
4.13	Architecture interne du MMX	82
4.14	Code source de la fonction pour construire l'image SSD d'origine	83
4.15	Influence du mouchetis pour un noyau 21x21 (utilisation du SSD avec validation croisée)	86

4.16	Correction de l'effet dilatation pour différents noyaux et comparaison avec les résultats issus de OpenCV (http://vision.middlebury.edu/stereo/data/)	87
4.17	L'écart en pourcentage du nombre de points appartenant à la forme rectangulaire et comparaison avec les performances de OpenCV	88
4.18	Évolution de l'effet de dilatation en fonction du noyau et comparaison avec OpenCV	89
4.19	Influence de la tolérance (http://vision.middlebury.edu/stereo/data/)	90
5.1	Sélection des points caractéristiques de la culasse	97
5.2	Traitements pour la localisation des points caractéristiques de la culasse	98
5.3	IHM : Calibration des paramètres intrinsèques des deux caméras	101
5.4	IHM : Calibration des paramètres extrinsèques des deux caméras	102
5.5	IHM : Calibration du modèle de régression 3D	103
5.6	IHM : Programme de détection et d'appariement des rouleaux	104
5.7	Première zone de recherche de rouleaux	104
5.8	Mise en correspondance des rouleaux des deux images après découpage en zone et utilisation de la contrainte d'ordre	105
5.9	Visualisation d'une surface mouchetée sans défaut	108
5.10	Visualisation d'un élément de 3mm d'épaisseur	109
5.11	Visualisation d'un élément de 3mm d'épaisseur avec un rapprochement de nos caméras	110
5.12	Visualisation d'éléments de 2mm d'épaisseur	111
5.13	Influence de l'utilisation de noyaux mono-directionnels	112
5.14	Visualisation de chocs de 1mm d'épaisseur	113
5.15	IHM : Programme d'acquisition des images des deux caméras	114
5.16	IHM : Calibration du système stéréoscopique	115
5.17	IHM : Construction de la carte dense de disparité pour une pièce étalon	116
5.18	IHM : Construction de la carte dense de disparité pour la pièce courante et recherche de défaut	117
7.1	Utilisation par déplacement linéaire	121
7.2	Le capteur Sick Ranger	122
7.3	Le capteur Photon Focus	122
7.4	Le capteur Automation Technology	123
7.5	Le capteur Vialux Z-Snapper	123
7.6	Le capteur Point Grey Bumblebee 2	124
7.7	Le capteur VisionRT	124
7.8	Le capteur GOM Atos III	125

7.9	Le capteur PMD Technologies CamCube	125
7.10	Le capteur Mesa Imaging SwissRanger	126

Liste des tableaux

2.1	Bilan : Avantages et inconvénients des solutions 3D et leur domaine d'application	20
4.1	Influence de l'angle inter-caméra sur la précision de la rectification	65
4.2	Comparaison du nombre d'opérations d'additions et de multiplications pour le cas classique et le cas avec optimisation	75
4.3	Quantification et confrontation du nombre d'opérations sur un exemple pour le cas classique et le cas avec optimisation	75
5.1	Erreur de précision sans régression (volume $20 \times 20 \times 20 \text{ cm}^3$)	99
5.2	Erreur de précision avec régression (volume $20 \times 20 \times 20 \text{ cm}^3$)	99
5.3	Erreur de précision avec régression (volume $18 \times 18 \times 18 \text{ cm}^3$)	99
5.4	Erreur de précision avec régression (volume $14 \times 14 \times 14 \text{ cm}^3$)	100
5.5	Erreur de précision avec régression (volume $10 \times 10 \times 10 \text{ cm}^3$)	100

Chapitre 1

Introduction générale

La vision industrielle est aujourd'hui un élément essentiel dans le domaine de la production pour des applications d'assistance aux opérateurs, de sécurisation, de contrôle, de mesure, ... Elle est une surcouche de la vision par ordinateur à laquelle elle associe un système d'acquisition, un système d'éclairage, un système mécanique et où l'ensemble de ces éléments suit un processus de synchronisation afin de répondre à une problématique industrielle.

Cette thèse présente notre contribution à l'intégration de la stéréovision dans le domaine industriel. Elle s'insère dans un partenariat avec la société Visionic dans le cadre de conventions industrielles de formation par la recherche.

La société Visionic¹ est une société spécialisée en informatique industrielle et en vision industrielle. Elle fait partie des leaders parmi les intégrateurs vision en France. Elle conçoit, réalise, et installe des machines spéciales pour du contrôle et de la mesure sans contact dans les domaines divers tels que l'automobile, la pharmaceutique, ...

Les acteurs de la vision industrielle sont confrontés à des problématiques de plus en plus complexes. La résolution d'applications tri-dimensionnelle est une demande récente des industries de production. Les intégrateurs vision et les fabricants de caméras se dirigent vers des solutions de stéréovision, soit par la conception de capteur évolué, soit par le développement d'outils logiciel.

Nous nous consacrons à deux approches de la stéréovision liées à deux contextes applicatifs distincts : la mesure tri-dimensionnelle et le contrôle tri-dimensionnel.

La mesure tri-dimensionnelle a pour objectif de fournir des coordonnées 3D de tout élément de la scène auquel nous nous intéressons sous la contrainte qu'il soit visible par les deux caméras. Nous nous focalisons dans ce cas à la problématique de calibrage fort pour estimer les paramètres intrinsèques et extrinsèques de nos caméras ainsi qu'au calcul 3D par triangulation.

1. Site internet de la société <http://www.visionic.fr>

Le contrôle tri-dimensionnel a pour objectif de fournir une information 3D de l'ensemble des éléments de la scène pour des problématiques de contrôle d'aspect. Nous nous intéressons à la production de carte dense de disparité représentative de la profondeur des éléments de la scène. Une implémentation industrialisable au sens « temps de cycle » nous amène à nous intéresser à la problématique d'alignement des caméras par détermination de l'orientation inter-caméra par estimation de la géométrie épipolaire. Elle nous permet d'apporter une rectification à nos images d'entrée, accélérant ainsi le processus de stéréo-corrélation.

Dans un premier chapitre, nous présenterons de manière détaillée, la problématique associée à cette thèse. Nous présenterons dans un premier temps les fondamentaux de la stéréovision, les différentes approches. Nous aborderons à la fois le contexte industriel et le contexte scientifique de la thèse. Nous citerons quelques solutions industrielles existantes. Nous distinguerons les deux types d'application industrielle et les deux approches distinctes qui permettent de résoudre chacune d'elle. Nous schématiserons les étapes de ces deux approches.

Dans un second chapitre, nous présentons la première approche que nous intitulerons « reconstruction 3D directe ». Nous aborderons les méthodes de calibrage fort. Nous rappellerons les techniques d'estimation des paramètres intrinsèques et extrinsèques des caméras. Nous détaillerons la partie de construction des droites caméra-objet et la mesure tri-dimensionnelle par intersection de droites. Nous montrerons comment nous améliorons la précision de notre système par un apprentissage 3D. Des résultats expérimentaux sont présentés montrant l'influence de la pose des caméras sur la précision de la mesure 3D.

Dans un troisième chapitre, nous présentons la seconde approche que nous intitulerons « production de carte dense de disparité ». Nous aborderons les méthodes d'estimation de la géométrie épipolaire et des matrices de rectification. Nous détaillerons les différentes étapes de stéréo-corrélation ainsi que les différents critères de corrélation associés. Nous présenterons notre approche pour optimiser la production de carte de disparité algorithmiquement et dans l'implémentation. Nous aborderons enfin notre méthode améliorant qualitativement et quantitativement nos cartes de disparité.

Dans un quatrième chapitre, nous présenterons des applications industrielles mettant en œuvre les deux méthodes de reconstruction 3D directe et de production de carte dense de disparité. Nous évaluerons la première méthode sur une application de recalage 3D et une application de décaissage puis la seconde méthode sur deux applications de contrôle d'aspect.

Chapitre 2

Problématique

Sommaire

2.1	La stéréovision	4
2.2	Les applications industrielles	5
2.2.1	Le "décaissage" et le "dévissage"	5
2.2.2	Le recalage et la mesure 3D	7
2.2.3	Le contrôle d'aspect	8
2.3	Les solutions industrielles	9
2.3.1	La conoscopie holographique	9
2.3.2	La triangulation laser	10
2.3.3	La projection de franges	12
2.3.4	La stéréovision classique	14
2.3.5	La stéréovision par production de carte de disparité	16
2.3.6	La stéréovision par projection de franges	16
2.3.7	Le temps de vol	18
2.3.8	Avantages et inconvénients	20
2.4	Les deux approches dans leur contexte applicatif	21
2.4.1	Approche par reconstruction 3D	21
2.4.2	Approche par production de cartes de disparité	22

Introduction

La stéréovision est une composante de la vision par ordinateur. Elle intéresse de nombreux scientifiques car elle offrirait à une machine les mêmes capacités que l'homme : apprécier la profondeur des éléments d'une scène. Néanmoins, le contexte environnemental, la précision de mesure et le temps de réactivité du système sont des contraintes qui nous amènent à poursuivre les recherches dans le domaine.

L'industrie est en attente de solutions de stéréovision mais pas à n'importe quelle condition. Le système doit être répétable, rapide, précis même dans des conditions difficiles (état poussiéreux, des pièces réfléchissantes, ...) et il doit s'intégrer à une chaîne de production existante.

2.1 La stéréovision

La stéréovision repose sur le principe de la triangulation. Si un point de l'espace est visible par deux caméras, nous pouvons déterminer ses coordonnées tri-dimensionnelles. La problématique de la stéréovision est double. Elle nécessite à la fois de déterminer les paires de points qui se correspondent, et également de calculer la valeur tri-dimensionnelle en se basant sur la géométrie du système stéréoscopique. La calibration du capteur de stéréovision est la condition indispensable de la réalisation de cette dernière.

On distingue deux approches de la calibration : la calibration monoculaire qui exprime la relation entre les points de l'espace et leur projection dans l'image et la calibration stéréoscopique qui exprime la relation entre les points correspondants dans les deux images.

Les aspects théoriques de la stéréovision sont grandement étudiés depuis maintenant trois décennies. De nombreux ouvrages [Fau93] [HM95] [HZ00] ont été publiés et sont des références. Ils permettent de poser la théorie de la géométrie de la stéréovision basée sur les principes de la géométrie projective.

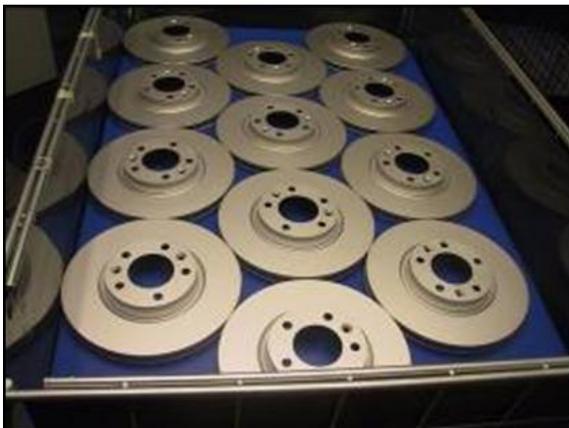
2.2 Les applications industrielles

Dans le domaine industriel, les applications sont diverses de par leur contexte environnemental, de par la scène et son contenu à observer, de par l'intégration à l'existant, de par le type de contrôle à réaliser. Les intégrateurs en vision industrielle tendent à maîtriser l'environnement par des caméras et un éclairage adaptés. Aujourd'hui, ce sont davantage les contrôles à réaliser qui sont sources de problématique et ils sont variés.

Nous citerons quelques applications 3D montrant cette diversité.

2.2.1 Le "décaissage" et le "dévraccage"

Le « (dé)caissage » (ou « (dé)palettisation ») et le « (dé)vraccage » sont des applications qui consiste à insérer ou à extraire des éléments d'un conteneur par guidage d'un robot industriel. Un système d'acquisition et de traitement vision permet de localiser en 2D ou en 3D un ensemble d'éléments semblables. Le « (dé)caissage » a pour particularité de présenter les éléments de manière ordonnée, présentant les mêmes primitives dans la ou les images acquises, ou du moins suffisamment proche. Le « (dé)vraccage » est une application où l'a priori sur la position et l'orientation des éléments n'existe plus. Les éléments restent semblables mais leur localisation nécessitera des traitements plus complexes gérant notamment les éventuelles superpositions.



(a) Décaissage



(b) Dévraccage

FIGURE 2.1 – Problématique du dévraccage par rapport au décaissage

Nous pouvons voir en figure 2.1, la problématique liée au dévraccage. En effet, pour une application de décaissage, les éléments sont ordonnés, présentent la même face de la pièce. Les primitives dans les images, telles une recherche de cercles, sont aisément détectables. Les éléments sont dissociés les uns des autres et sont disposés sur un même

plan. Au final, la saisie de chaque élément par le robot est grandement simplifiée et n'affectera pas les autres éléments.

En opposition, une application de dévissage implique des problématiques complexes. Notamment le phénomène de superposition a des conséquences sur la recherche de primitives qui peuvent être occulté. Les primitives sont présentes avec des positions aléatoires, une orientation variée, des tailles différentes. Cela rend difficile leur détection. La saisie par le robot des différents éléments implique des trajectoires complexes dues notamment à l'orientation. Aussi, l'applicatif doit tenir compte que la prise d'un élément risque d'agir sur les autres éléments et que ces mêmes éléments peuvent empêcher la saisie, notamment due à la superposition et la proximité entre les éléments.

Nous pouvons visualiser un exemple d'application de « palettisation » en figure 2.2. Les cartons en entrée sont présentés de manière pseudo-aléatoire. Le robot embarquant le système de capture, localise le carton, le saisit à l'aide d'un système de préhension adapté et le dispose sur une palette suivant une position et un ordonnancement prédéfinis. Les exigences pour ce type d'application sont la stabilité, la précision, le temps de cycle et un robot adapté par rapport à la préhension, aux trajectoires et à sa capacité de charge.

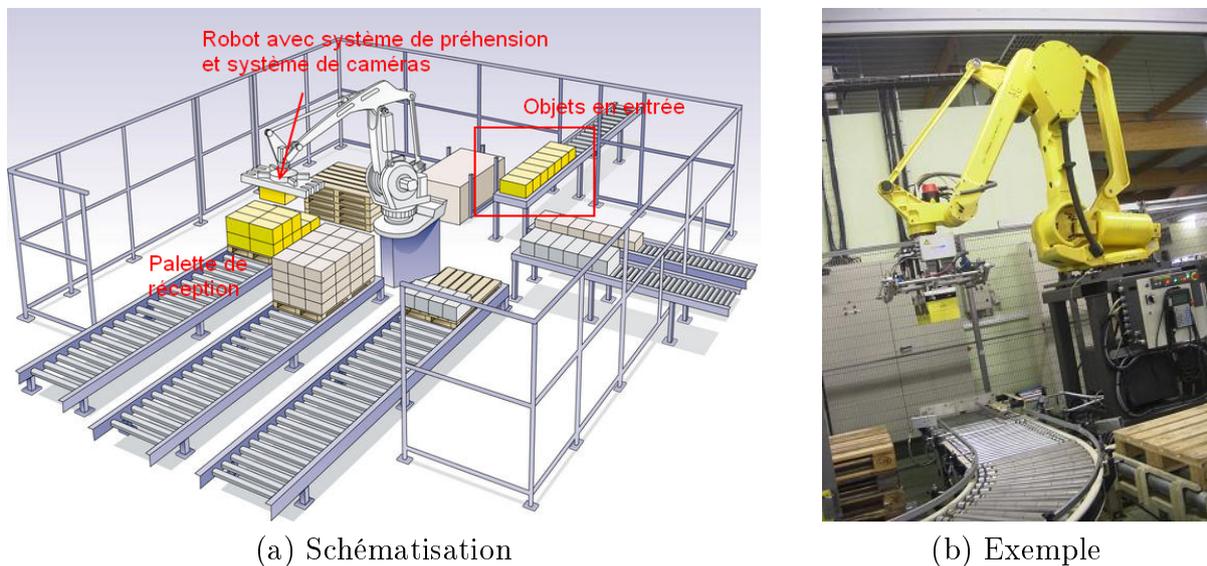


FIGURE 2.2 – Solution de palettisation par assistance robotique

Les éléments qui définissent la diversité applicative sont :

- Les outils pour positionner les objets à déplacer. Elle doit être adaptée à l'objet.
- Le système de préhension. Les robots utilisent essentiellement des pinces, mais aussi des mécanismes à ventouses ou magnétiques. Néanmoins, nous nous dirigeons vers

des systèmes proches de la main de l'homme car elle permettrait de saisir les objets dans toutes les orientations.

- La manière de déposer les objets dans sa nouvelle position et la trajectoire associée.

2.2.2 Le recalage et la mesure 3D

Les applications de recalage et de mesure 3D ont la même problématique : localiser des points 3D.

Le recalage 3D consiste à localiser une pièce afin de définir sa position et son orientation par rapport à un repère monde prédéfini. Ces applications sont utilisées pour que le système de vision puisse faire abstraction de la manière dont la pièce se présente aux systèmes d'acquisition. Elles sont généralement associées à des systèmes robotiques embarquant les caméras. Le recalage 3D permet de guider le robot vers la ou les zones de la pièce sur lesquelles nous souhaitons réaliser différents contrôles ou mesures. Méthodiquement, cela consiste à localiser dans l'espace un ensemble de points de référence de la pièce et, à partir de la connaissance métrique a priori que l'on a de la pièce, les différentes positions associées à des trajectoires du robot sont calculées.

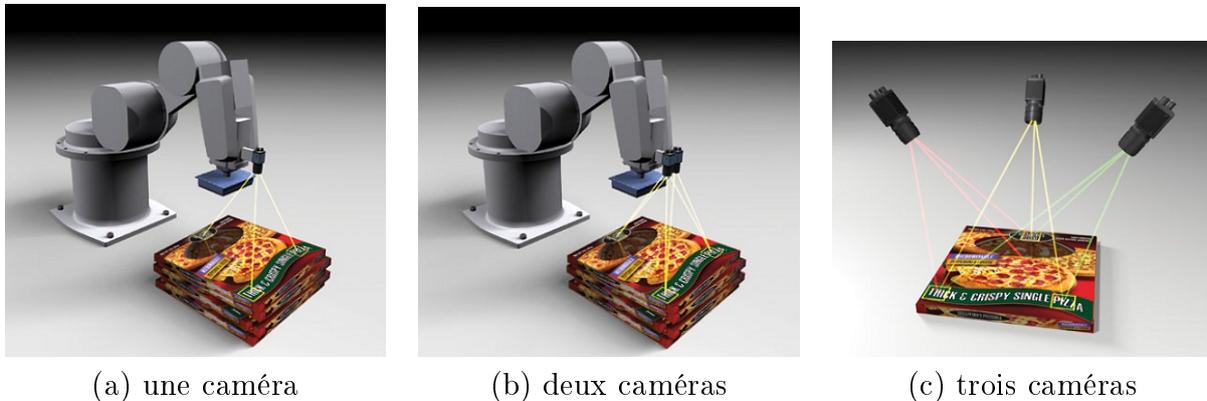


FIGURE 2.3 – Solution de recalage 3D

En figure 2.3, nous présentons une schématisation d'une application de recalage 3D dont les points caractéristiques sont des motifs à détecter dans les images, fournissant des informations d'échelle et d'orientation. Elle peut être réalisée de manière différente :

- avec un système à une caméra où l'information d'échelle et d'inclinaison des points caractéristiques caractérise l'information 3D,
- avec un système stéréoscopique qui facilite l'appariement des points caractéristiques,
- avec un système multi-caméra qui favorise la précision de mesure 3D.

La mesure 3D se base sur les mêmes principes. A partir d'un ensemble de points caractéristiques de la pièce, des mesures de points, de distance ou d'angles sont réalisées. Cette application ne se limite pas à une pièce mais s'étend à l'ensemble de la scène que l'on observe.

2.2.3 Le contrôle d'aspect

Le contrôle d'aspect peut être destiné à plusieurs types d'application mais cela consiste à construire une représentation de la surface d'un élément comme dans la figure 2.4.

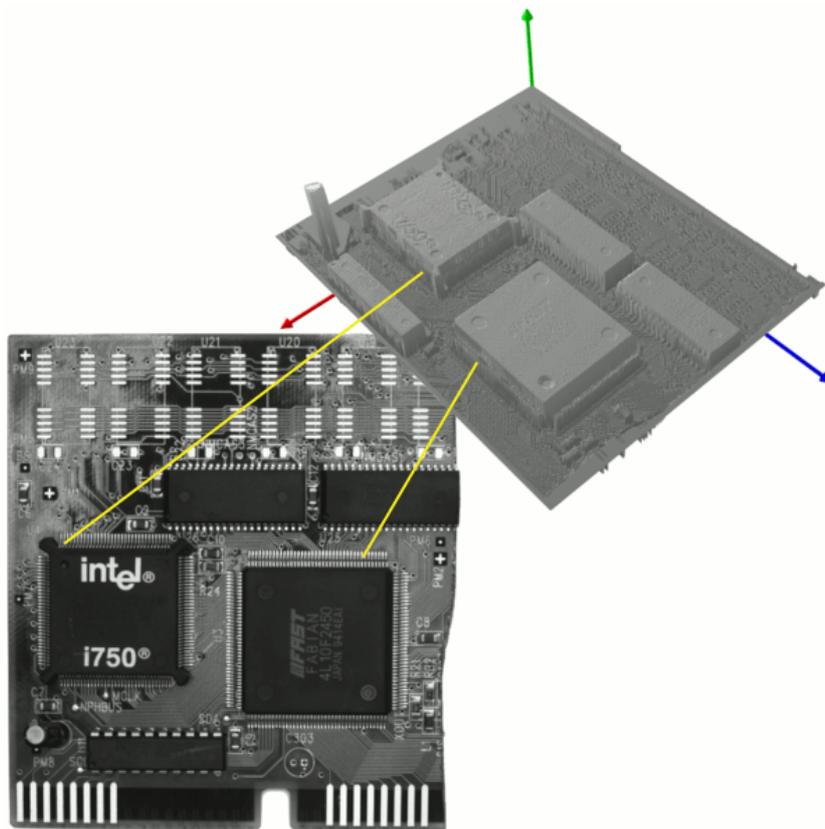


FIGURE 2.4 – Solution de construction de profil de surface pour des composants électronique

Les applications destinées à utiliser ce procédé servent à identifier la surface, soit pour vérifier la présence de variations attendues, soit pour détecter des éventuels surplus ou manque matière. Il y a deux moyens de réaliser la mesure :

- par l'utilisation d'outils de traitement pour réaliser des mesures ou des contrôles.
- par comparaison avec un modèle où nous réalisons une première représentation de surface sur une pièce de référence, puis à partir de la représentation des pièces à analyser, nous réalisons une comparaison.

2.3 Les solutions industrielles

Les fabricants de caméras et puis précisément de système de capture, amènent leur produits vers ces problématiques 3D. Soit par des caméras intelligentes qui intègrent des moyens de mesure tri-dimensionnelle, soit un capteur stéréoscopique mécaniquement aligné, soit deux caméras associées à un éclairage spécifique, soit deux caméras associées à des bibliothèques de traitement évoluées, ...

Nous distinguons plusieurs familles de la technologie 3D :

2.3.1 La conoscopie holographique

Dans un système conoscopique, un rayon laser est projeté sur une surface, puis la réflexion au travers du même faisceau passe à travers un cristal biréfringent et est envoyé sur un capteur CDD. La fréquence des motifs de diffraction peut être analysée et permet de déterminer la distance par rapport à cette surface. L'atout principal de l'holographie conoscopique est qu'un unique faisceau (aller-retour) est nécessaire pour effectuer la mesure, permettant de mesurer par exemple la profondeur d'un trou finement percé.

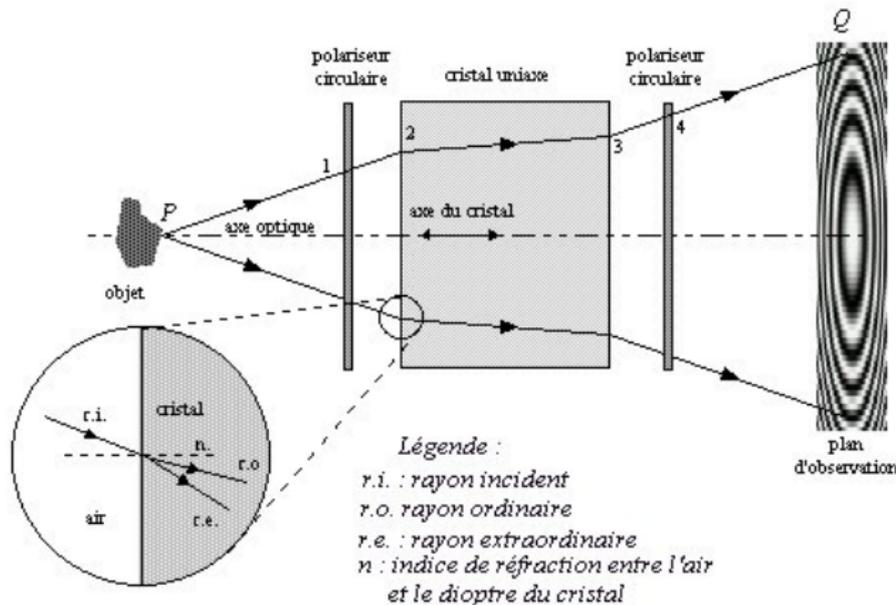


FIGURE 2.5 – Principe de la conoscopie holographique

Un système conoscopique est également appelé "capteur ponctuel en Z". Cela signifie qu'il donne la distance séparant le capteur de l'objet réfléchissant le rayon laser. Le signal se résume donc à une valeur. Le système est généralement associé à un système de déplacement tel qu'un axe linéaire. Cette association permet alors de générer un signal où la distance Z est fonction de la position du capteur le long de son axe de déplacement.

Chacune des mesures Z est séparée de la suivante par une valeur Δx égale au pas de déplacement du capteur le long de l'axe. Le signal acquis est ensuite analysé par des outils classiques du traitement du signal.

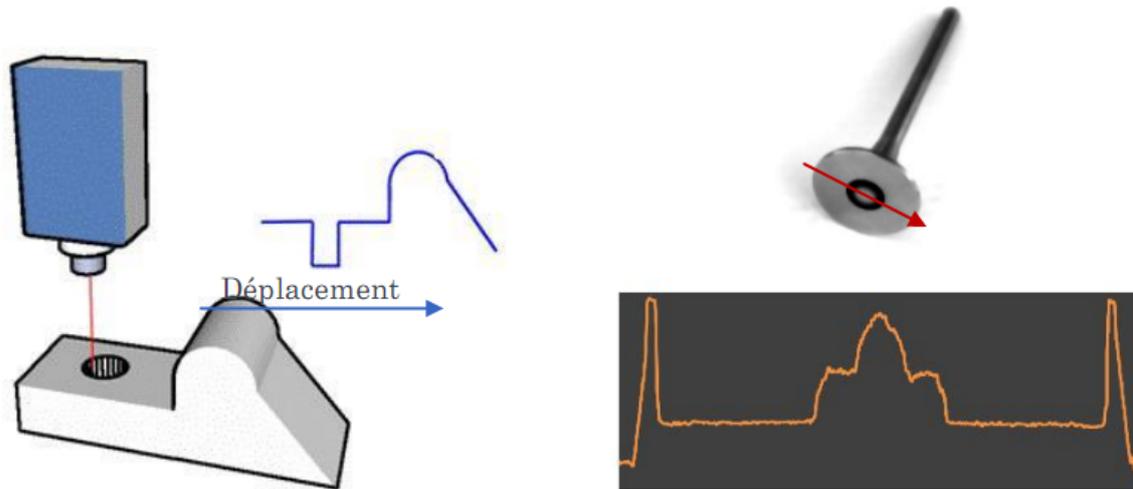


FIGURE 2.6 – Utilisation par déplacement linéaire

L'intervalle de mesure du Z est limité à quelques centimètres et dépend de la lentille utilisée. Il en va de même pour la précision de la mesure effectuée.

Nous noterons qu'il n'existe qu'une société qui propose un capteur de conoscopie holographique, la société israélienne Optimet qui a déposé un brevet sur cette technologie.

Les précisions obtenues sont inférieures, dans les cas les plus défavorables, au $1/10$ de millimètre ce qui fait de la conoscopie holographique un capteur de grande précision. L'axe du laser étant colinéaire à l'axe optique, les capteurs à conoscopie holographique permettent de mesurer sur des pentes inclinées jusqu'à 85° par rapport à l'axe optique. Enfin, un renvoi d'angle à 90° permet de mesurer des zones cachées. A noter qu'il existe des variantes ou des intégrations à base de miroirs rotatifs permettant d'obtenir un profil en Z voire un nuage de points 3D à partir d'une source de mesure ponctuelle en Z .

Les travaux de thèse de D. Gava [Gav98] ont été entrepris en association avec la société Optimet dans l'objectif de développer ce capteur 3D reposant sur la technologie conoscopique. Il y évoque notamment la problématique de calibration.

2.3.2 La triangulation laser

Un système de triangulation laser est composé d'une caméra matricielle et d'une ligne laser. Un déplacement de l'objet ou de la caméra est nécessaire pour acquérir une image.

L'épaisseur de la ligne laser est un élément important à maîtriser et définit le plus petit ΔZ observable. Car si l'épaisseur de la ligne laser est trop importante, un trop faible Δz ne suffira pas à déformer la ligne laser. Le signal acquis est une image 2D (ou 2D $1/2^1$) qui peut être travaillée par des outils de traitements 2D classiques.

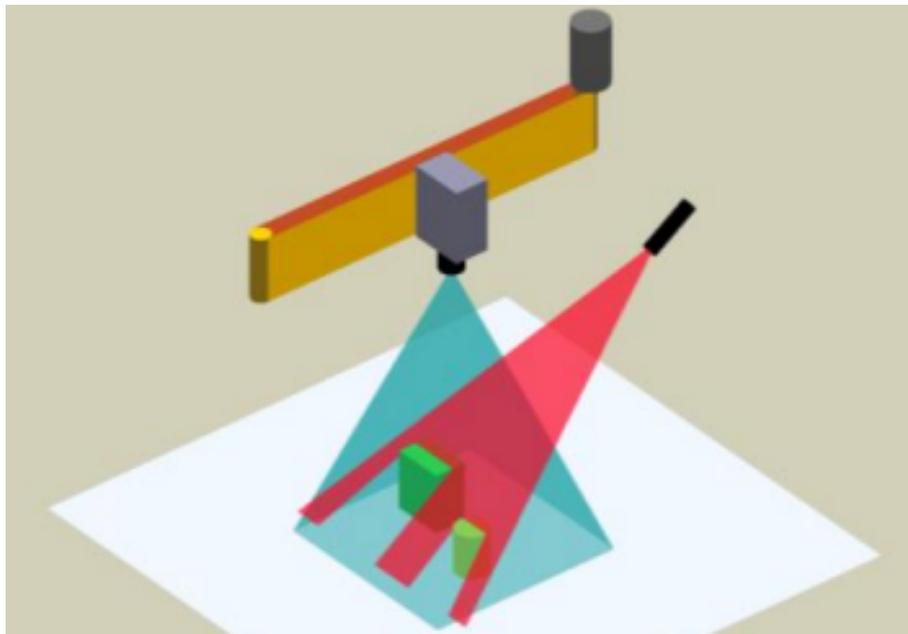


FIGURE 2.7 – Schématisation de la triangulation laser

Pour une position donnée, le système de triangulation laser est capable de calculer un profil linéaire. Une image 2D $1/2$ générée par triangulation laser est construite par l'accumulation de plusieurs de ces profils linéaires. Ainsi pour acquérir une image 2D $1/2$ de 500 lignes de 1000 pixels il faut que la caméra acquière 500 images de n lignes de 1000 pixels et qu'elle détermine pour chacune de ces images le profil linéaire de 1000 points formé par la ligne laser. La figure 2.8 montre une image acquise avec un système de triangulation laser.

Le déplacement est indispensable à l'acquisition. Si la pièce observée défile sous la caméra, cela ne pose pas de problème. Cependant si la pièce est statique, alors il est impératif de mettre en mouvement la caméra par un axe linéaire. Dans ce cas la mécanique et l'électronique à mettre en œuvre est contraignante. A noter que sur certaines applications robotisées, la caméra peut être déplacée par le robot ou bien un système de miroir rotatif permet de faire balayer le champ par le laser.

Les lasers utilisés sont généralement de classe 3B, c'est à dire de puissance d'émission comprise entre 25 mW et 500 mW. Les normes imposent de protéger les opérateurs de

1. Nous définissons une image 2D $1/2$ comme une image 2D dont le niveau de gris n'est pas une information d'intensité lumineuse mais une information de profondeur

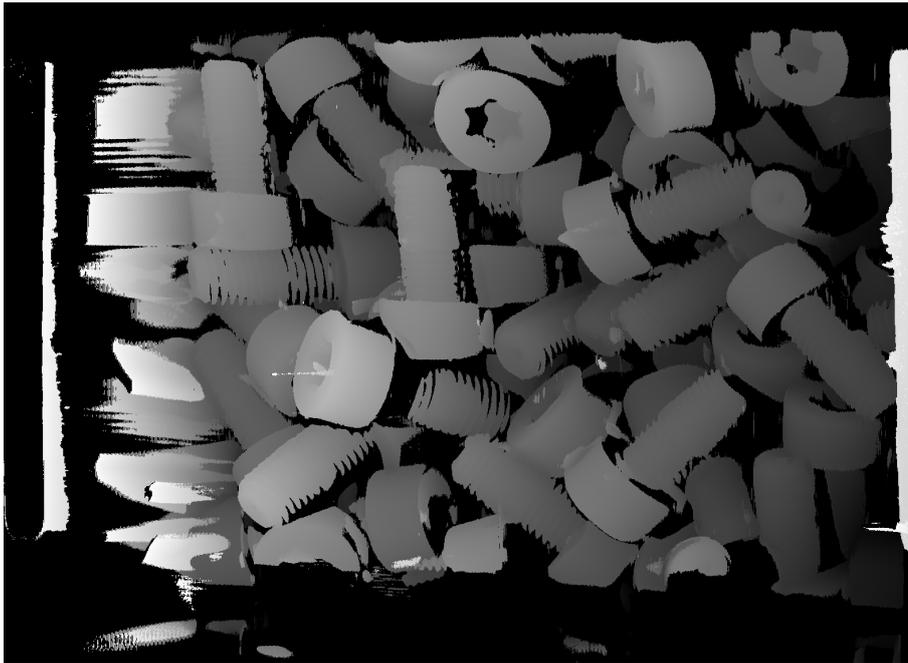


FIGURE 2.8 – Exemple d’image, convertie sur 256 niveaux de gris, acquise avec un système de triangulation laser

sorte qu’ils ne soient pas exposés directement ou à une réflexion directe du laser.

La précision en Z s’acquiert en augmentant l’angle entre l’axe du laser et l’axe optique de la caméra. Cependant plus on augmente l’angle, plus on génère des zones d’ombre, c’est à dire des zones dans lesquelles le Z n’est pas mesurable.

Enfin il est impératif de calibrer le système pour passer d’une information 2D $1/2$ à une information 3D.

2.3.3 La projection de franges

Un système de projection de franges est constitué d’une caméra et d’un projecteur, schématisé par la figure 2.9. Le projecteur projette une lumière structurée (généralement des franges ou une grille) sur l’objet à observer. La scène est acquise par la caméra. Une lumière est dite structurée lorsqu’elle est spatialement modulée en intensité comme par exemple la lumière provenant d’une source lumineuse devant laquelle on a placé une grille.

On peut déterminer la géométrie d’une surface grâce à ce genre d’illumination pour autant que la direction de cette dernière soit distincte de celle de l’axe optique de la caméra. Une structuration adéquate de la lumière permet de transmettre l’information qui, pour chaque pixel de la caméra, détermine sans ambiguïté un plan virtuel qui intersecte la surface de l’objet en ce point. La hauteur est déterminée ensuite par triangulation. L’information de hauteur étant véhiculée par la lumière, les zones d’ombres portées sont

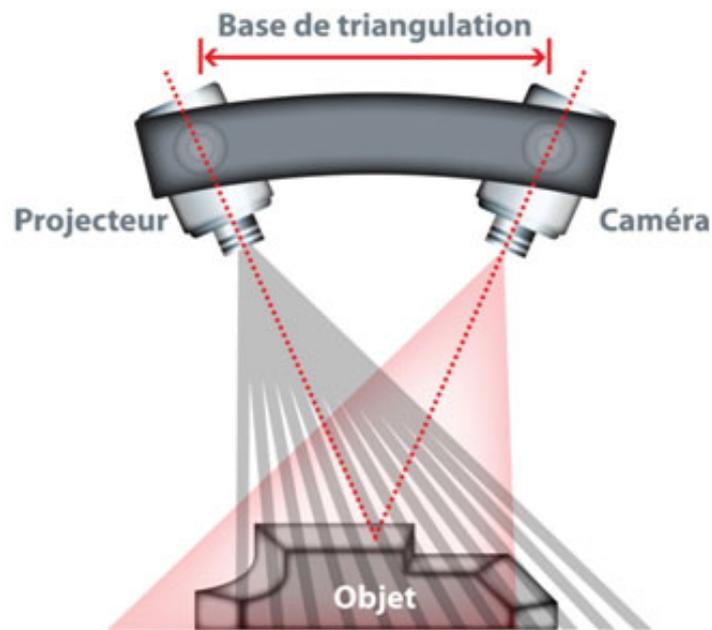


FIGURE 2.9 – Schématisation de la projection de franges

inexploitables.

Le principe de calcul de l'image 3D est similaire à celui de la triangulation laser excepté que le déplacement de la ligne laser et de la caméra est remplacé par la projection d'une grille. Il s'agit de calculer, pour les points mise en évidence par la grille, la déformation locale de la grille générée par la topologie de l'élément observé. Le signal de sortie est un nuage de points 3D.

La figure 2.9 montre la déformation des franges sur une pièce présentant une cavité.

Le système nécessite une calibration et présente une sensibilité importante aux variations de lumière. De plus la méthode est inutilisable sur les objets réfléchissants. Enfin l'objet doit être immobile durant toute la période d'acquisition.

Le système ne nécessite aucun mouvement contrairement aux systèmes de triangulation laser. Les temps d'acquisition sont relativement courts (40 à 200 ms). La résolution en Z des images obtenues est très élevée avec par ailleurs une fréquence d'acquisition pouvant aller jusqu'à 20 images par seconde. On remarquera cependant qu'aussi fin qu'il soit, le quadrillage de la grille projetée ne permet pas d'atteindre, en résolution, les performances d'un système de triangulation laser dans lequel le pas d'avancement du système permet un quadrillage très fin de l'objet analysé. Pour cela plusieurs acquisitions de la même scène peuvent être prises en faisant varier la grille projetée.

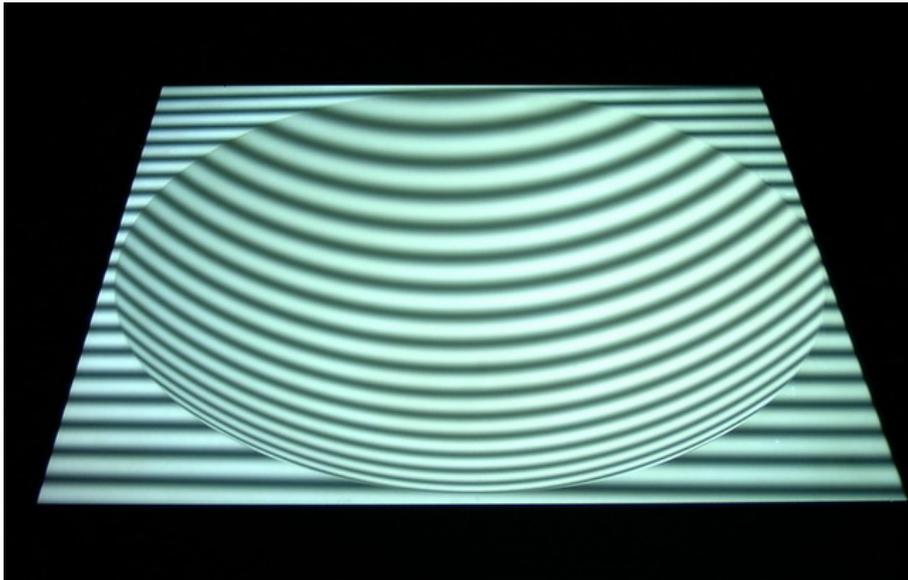


FIGURE 2.10 – Conséquence de la surface de la pièce sur la déformation des franges

Sai Siva Gorthi dans [GR10] fait un bilan des travaux utilisant la projection de franges. La méthode de résolution suit le même cheminement même si les méthodes utilisées sont diverses :

- Projection et acquisition
- Analyse des franges
- Construction d'une image de profondeur
- Calibration pour définir les correspondances entre les points image et les points 3D.

2.3.4 La stéréovision classique

Un système de stéréovision est caractérisé par l'acquisition de deux images de l'objet à observer sous deux angles différents. Après avoir acquis deux images d'un objet sous deux angles différents, les coordonnées images des points à mesurer sont déterminées sur chacune d'entre elles. L'appariement des points similaires est généralement faite automatiquement. Le résultat est une liste de coordonnées 3D.

Dans un premier temps chacun des deux points de vue (chacune des deux caméras par exemple) est calibré indépendamment de l'autre à l'aide de plusieurs mires avec des positions et des orientations variantes. Nous parlons ici de calibration monoculaire. Elle permet de déterminer les paramètres intrinsèques et extrinsèques de chaque caméra. Les paramètres intrinsèques de la caméra sont la projection du centre optique dans le repère image, la distance de la focale et les paramètres de distorsion de l'image.

Les paramètres extrinsèques sont les données de translation et de rotation entre le

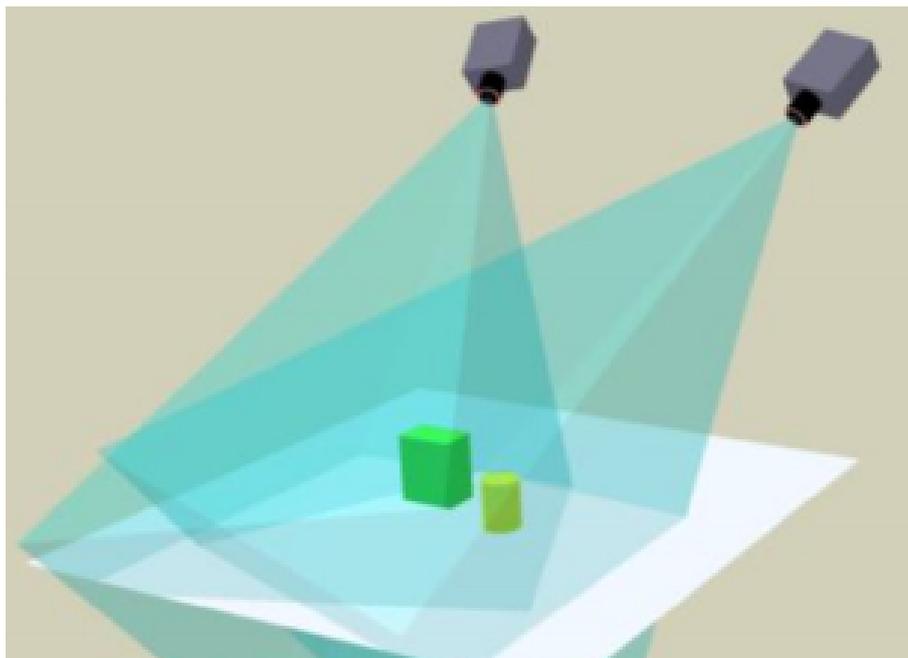


FIGURE 2.11 – Schématisation de la stéréovision classique

repère caméra et le repère monde. Ils permettent notamment de positionner chaque caméra dans un même repère donné.

Dans un second temps, chacune des deux images acquises est traitée par des outils classiques de traitement d'image de manière à produire une liste de points 2D caractéristiques des objets. Chaque image ayant produit chacune une liste de points, un appariement est nécessaire ensuite pour déterminer quels sont les points de la liste de gauche et de la liste de droite qui se correspondent. Cet appariement peut être fondé sur des *a priori* de la scène observée comme la préservation de l'ordre des éléments d'une image à l'autre. Cette contrainte impose le plus souvent d'avoir deux images similaires ou, autrement dit, de positionner les caméras très proches l'une de l'autre. Une fois les couples $(x, y)_{gauche}$ et $(x, y)_{droite}$ constitués, les modèles de calibration permettent de calculer les point $(x, y, z)_{monde}$ correspondants.

On voit ainsi apparaître l'inconvénient d'une telle méthode. Elle nécessite de pouvoir isoler dans l'image les éléments dont on veut calculer une information 3D. Si ces éléments ne présentent pas de géométrie ni d'aspect caractéristique alors les listes de points de gauche et de droite ne peuvent être extraites. Cela peut se produire pour des pièces en vrac ou pour des applications dont on ne connaît rien de l'image (images autoroutières). Un dernier inconvénient de cette méthode est la nécessité de calibrer le système stéréoscopique. La précision du système dépend de sa bonne calibration.

Néanmoins, les avantages sont nombreux. Le système d'acquisition est simple (deux

caméras) et ne nécessite aucun mouvement. Il présente une grande flexibilité : on peut choisir la caméra et l'optique. Il existe des systèmes vendus tout intégré et tout calibré. Enfin, le marché propose une grande diversité de solutions. A noter enfin que le système de stéréovision peut n'être constitué que d'une seule caméra amenée, par exemple, aux deux points de vue par un robot.

2.3.5 La stéréovision par production de carte de disparité

L'objectif est ici de générer une carte de disparité, c'est à dire une image 2D $1/2$ où les niveaux de gris sont représentatifs de l'éloignement des objets par rapport au système de capture. L'idée de cette méthode est de trouver toutes les paires de pixels (P_{gauche}, P_{droite}) correspondants suivant deux alternatives :

- une mise en correspondance dense, c'est à dire pour l'ensemble des points de l'image
- une mise en correspondance éparse, c'est à dire pour des primitives de l'image (des contours par exemple).

Cela nécessite une recherche bidimensionnelle qui est ramenée à une recherche monodimensionnelle grâce aux propriétés de la géométrie épipolaire et de rectification des images qui en découle. On calcule la disparité pour un pixel comme la distance relative, en abscisse, avec son correspondant. Une calibration inter-caméra est nécessaire et permet de définir les matrices de rectification.

En production, chaque image est rectifiée par sa matrice de rectification. La disparité de chaque pixel (ou primitive) est calculée pour donner l'image 2D $1/2$. Les figures 2.12 2.13 montrent l'image de disparité pour les deux alternatives.

Des outils classiques de traitement d'image 2D viennent ensuite analyser l'image. Le temps de calcul d'une image de disparité peut prendre quelques centaines de millisecondes.

2.3.6 La stéréovision par projection de franges

Un système de stéréovision par projection de franges est très similaire à un système classique de stéréovision. Les cas d'utilisation sont toutefois très différents. Un système classique de stéréovision intervient généralement dans des cas où il s'agit de localiser des éléments (décaissage ou dévracage). Dans ces problématiques, l'appariement des objets étudiés est rendu possible par leur géométrie ou d'autres *a priori*. La stéréovision par projection de franges intervient dans des problématiques où rien ne permet d'appairer des points 2D en vue d'en calculer leur coordonnées 3D. Il s'agit de problématiques telles que la mesure de déformations d'un matériau ou encore la modélisation 3D d'un objet avec pour objectif, par exemple, de le comparer à un fichier numérique. Sur un matériau lambda soumis à un étirement, il n'y a pas de point particulier qui puisse être repéré dans

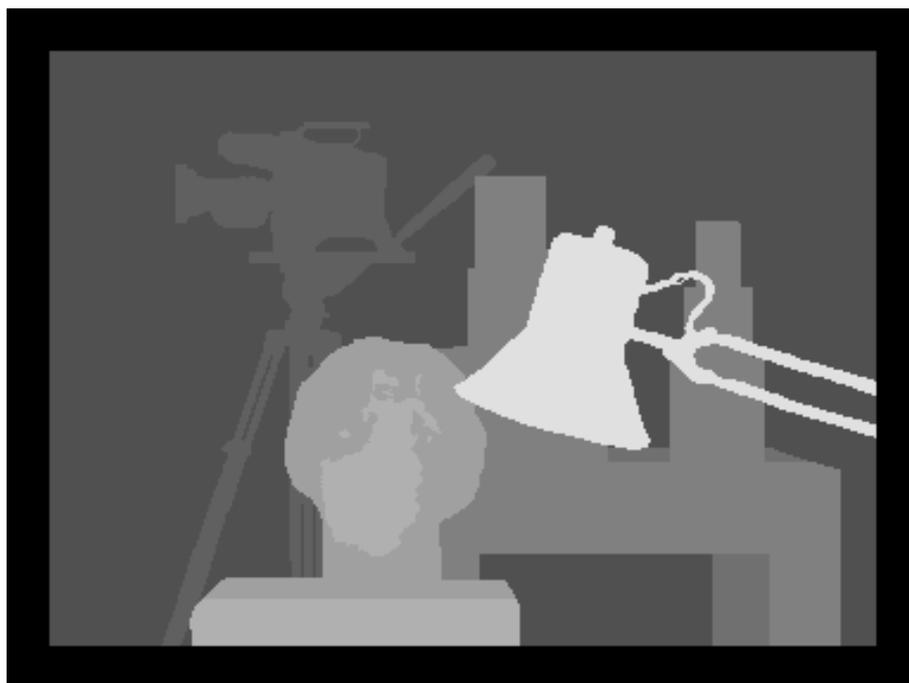


FIGURE 2.12 – Carte dense de disparité

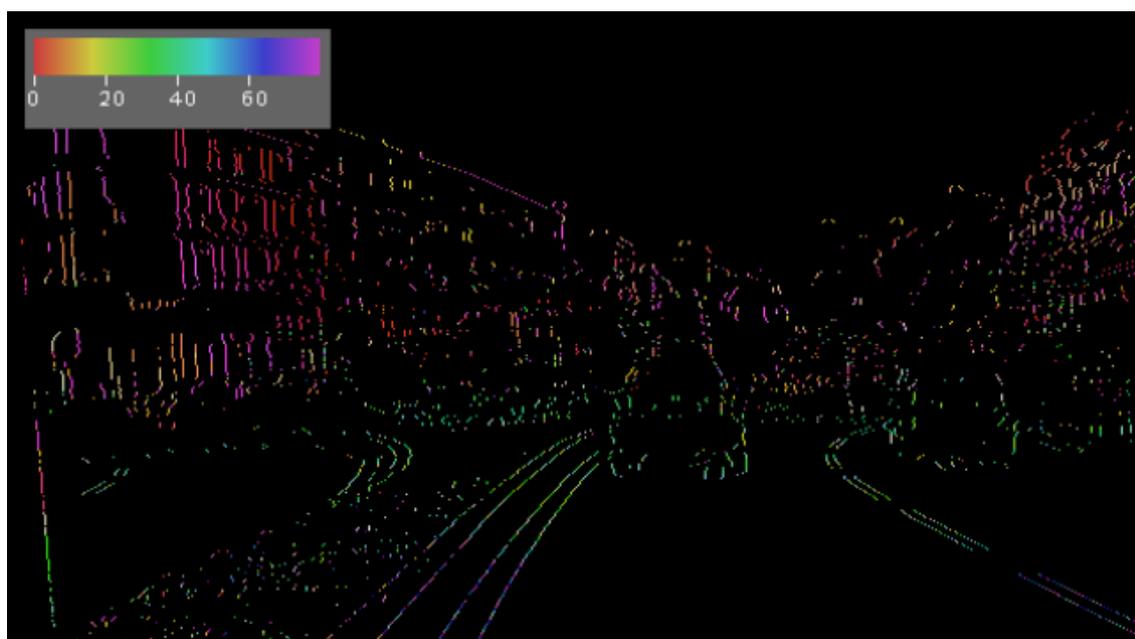


FIGURE 2.13 – Carte éparses issue des travaux [Kra08] de Sébastien Kramm sur une scène autoroutière

les images des deux caméras. Une solution serait de positionner des mires de couleur sur le matériau afin de créer des éléments d'appariement. Cette solution ne convainc pourtant pas car elle n'est pas sans contact, peut dégrader le matériau et n'est pas précise. La bonne solution consiste donc à projeter une grille dont les nœuds vont fournir des éléments

d'appariement. De même qu'un système classique de stéréovision, une calibration des paramètres intrinsèques et extrinsèques est nécessaire. Le signal de sortie est un nuage de points 3D.

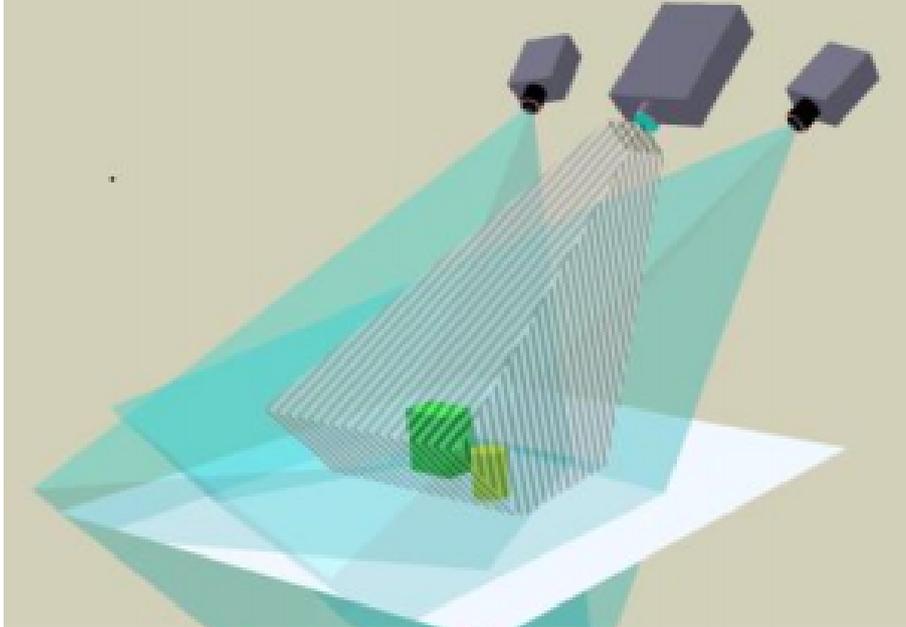


FIGURE 2.14 – Schématisation de la stéréovision par projection de franges

Les inconvénients sont les mêmes que ceux identifiés avec la méthode classique de stéréovision c'est à dire calibrer le système stéréoscopique. Nous retrouvons là aussi les mêmes avantages : simplicité du système, absence de mouvement et grand choix de solutions. Il existe des systèmes vendus tout intégré et tout calibré. Enfin, le marché propose une grande diversité de solutions.

Nous citerons notamment les travaux [SS03] qui utilise la projection de franges pour construire des carte dense de disparité.

2.3.7 Le temps de vol

La télémétrie par mesure de temps de vol est une méthode de mesure sans contact basée sur la mesure du temps de propagation aller et retour Δt de la lumière entre le capteur et la pièce à mesurer.

La distance mesurée est : $d = \frac{c\Delta t}{2g}$, avec c la vitesse de la lumière et g la constante de gravité.

L'information récupérée par le capteur est une image 2D. Le système est insensible aux conditions d'éclairages puisqu'il embarque son propre système. L'image acquise ne

présente pas de zones cachées.

La distance de mesure se trouve entre quelques dizaines de centimètres (en dessous le temps de vol est trop court pour être mesuré) à quelques mètres. Suivant la distance capteur-objet, la résolution en X et Y varie d'un peu plus de 1 mm à une dizaine de mm. La résolution en Z est de plus ou moins 20 mm. La faible résolution en Z fait de ce capteur un outil d'évaluation des distances mais absolument pas un outil de mesure en Z.

Le type d'application est essentiellement de l'absence présence d'objets ou de la mesure approximative de volume. Les résolutions de tels capteurs sont inférieures à 200 pixels par 200 pixels. Ceci est dû la taille importante des pixels du capteur. Cette taille est approximativement de $40\ \mu\text{m}$ par $40\ \mu\text{m}$ contre $5\ \mu\text{m}$ par $5\ \mu\text{m}$ pour des caméras traditionnelles. A titre de curiosité, un capteur temps de vol VGA (640 pixels par 480 pixels) aurait une taille supérieure à 25 mm par 19 mm, ce qui représente incontestablement une surface démesurément grande par rapport aux pavés CCD ou CMOS. A noter que la taille des pixels s'explique par la nécessité d'avoir une grande zone de récupération des photons émis par l'éclairage embarqué et renvoyé par l'objet observé.

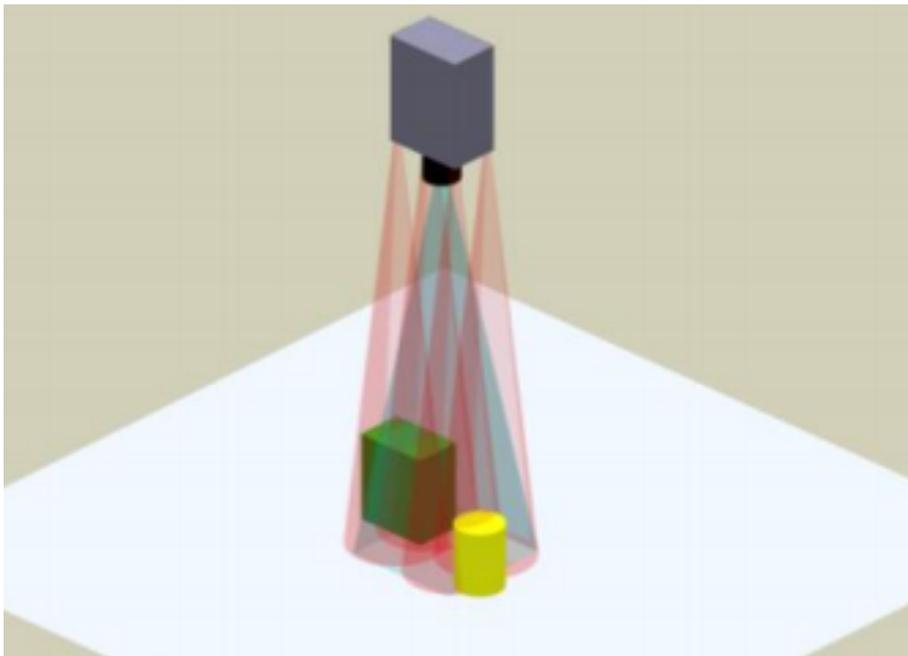


FIGURE 2.15 – Schématisation de l'acquisition temps de vol

Parmi les avantages, on trouve évidemment le côté tout intégré du capteur qui embarque son propre éclairage. Le capteur suffit en lui-même. Le système d'acquisition est simple et ne nécessite pas de calibration. L'inconvénient majeur est la faible dimension des capteurs.

2.3.8 Avantages et inconvénients

Les technologies 3D sont diverses de part leurs caractéristiques, leur mise en œuvre, leurs performances et également les domaines applicatifs auxquels elles sont destinées. Le tableau 2.1 énumère les différents avantages et inconvénients de l'ensemble des solutions 3D présentées ainsi que les domaines applicatifs de prédilection.

Solution 3D	Avantages	Inconvénients	Domaine applicatif
Conoscopie holographique	Précision de mesure	Temps d'acquisition dû au déplacement. Technologie peu répandue	Métrie
Triangulation laser	Précision de mesure	Temps d'acquisition dû au déplacement	Analyse de surface
Projection de franges	Temps d'acquisition court	Sensible aux variations de luminosité. Objets réfléchissants	Analyse de surface
Stéréovision classique	Flexibilité de système d'acquisition	Calibration indispensable. Nécessité d'avoir des objets caractérisables. Appariement de points	Mesure 3D
Stéréovision par production de carte de disparité	Flexibilité de système d'acquisition	Calibration indispensable	Analyse de surface
Stéréovision par projection de franges	Flexibilité de système d'acquisition	Objets réfléchissants	Analyse de surface
Temps de vol	Tout en un. Traitement temps réel	Résolution de mesure	Analyse de surface

TABLE 2.1 – Bilan : Avantages et inconvénients des solutions 3D et leur domaine d'application

2.4 Les deux approches dans leur contexte applicatif

Compte tenu de la diversité applicative en vision industrielle, deux applications distinctes sont ciblées et mettent en œuvre deux approches de stéréovision. Il est évident que les deux approches ne peuvent être mises en confrontation car elle répondent à des problématiques différentes. La présentation de ces deux méthodes ont fait l'objet d'une communication [PKBS10] lors d'une conférence.

2.4.1 Approche par reconstruction 3D

La reconstruction 3D directe est la première approche que l'on a associé à un système de capture stéréoscopique. Elle se base sur le principe que si l'on visualise un même point de l'espace sous deux angles différents, nous pouvons obtenir une information tridimensionnelle.

Principe

L'élément de base de la reconstruction 3D est la triangulation entre les deux objectifs et le points de l'espace à identifier. La figure 2.16 exprime cette triangulation.

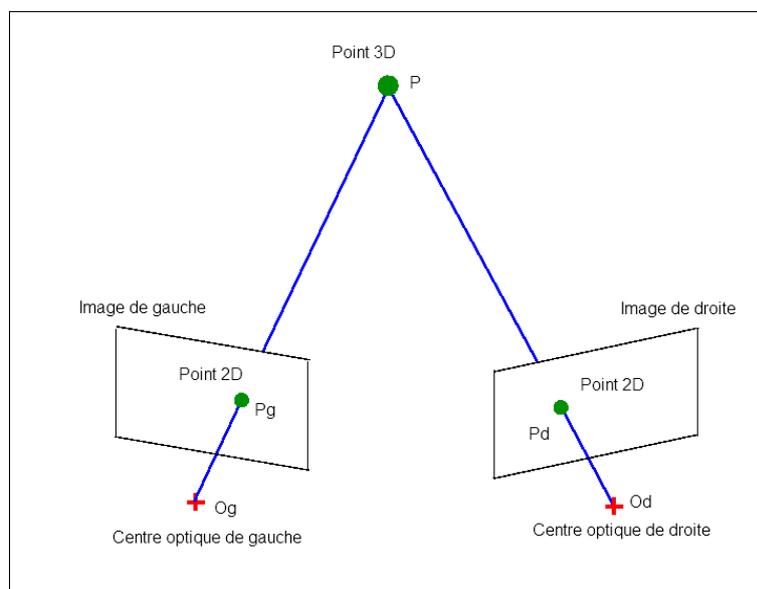


FIGURE 2.16 – Triangulation

La problématique est de construire les deux droites (O_gP) et (O_dP) ou encore (O_gP_g) et (O_dP_d) . Ainsi, un calibrage fort de chacune des caméras indépendamment l'une de l'autre nous permet d'estimer les paramètres intrinsèques et extrinsèques des deux caméras. Chaque droite est définie en trois étapes successives :

- Estimation de la pose caméra à partir des paramètres extrinsèques.

- Expression du point image dans le repère caméra à partir des paramètres intrinsèques.
- Expression du point image dans le repère monde à partir des paramètres extrinsèques.

Au final, les coordonnées des caméras et des points image dans le repère monde nous permettent de construire les deux droites (O_gP) et (O_dP) . Le résultat de la triangulation est le calcul de l'intersection des deux droites (O_gP) et (O_dP) ou plus précisément du point médian du segment qui minimise la distance inter-droite.

Application de recalage 3D

Dans l'objectif de valider la méthode par reconstruction 3D directe, nous évaluons la précision de mesure 3D sur une application de recalage robot. Cette application consiste à présenter une pièce industrielle devant le système d'acquisition stéréo dans une position aléatoire et, à partir d'un nombre limité de points caractéristiques, de localiser la pièce dans le repère monde défini. Ce type d'application a pour objectif, à partir de la connaissance de la position de la pièce, de réaliser un ensemble de contrôles et de mesures à différents emplacements de la pièce.

2.4.2 Approche par production de cartes de disparité

Principe

La production de carte dense de disparité est la seconde approche. Elle fournit une information de profondeur pour l'ensemble des points de l'image exprimé par la disparité, qui est l'écart relatif entre deux points image qui se correspondent. La figure 2.17 illustre cette notion de disparité. Elle montre également que plus un point 3D est proche du système de capture, plus la disparité augmente.

La problématique est donc de mettre en correspondance les points similaires des deux images. Afin de ne pas comparer l'ensemble des points de l'image de gauche avec l'ensemble des points de l'image de droite, nous utilisons les propriétés de la géométrie épipolaire basée sur les propriétés de la géométrie projective. Elle permet de réduire l'intervalle de recherche de correspondant. Nous utilisons la rectification des images déduite de la géométrie épipolaire afin que les correspondants soient sur la même ligne dans les deux images.

Pour faire l'appariement des points similaires, nous utilisons les techniques de stéréocorrélation basées sur la minimisation ou la maximisation d'un critère de corrélation. La carte dense en est le résultat.

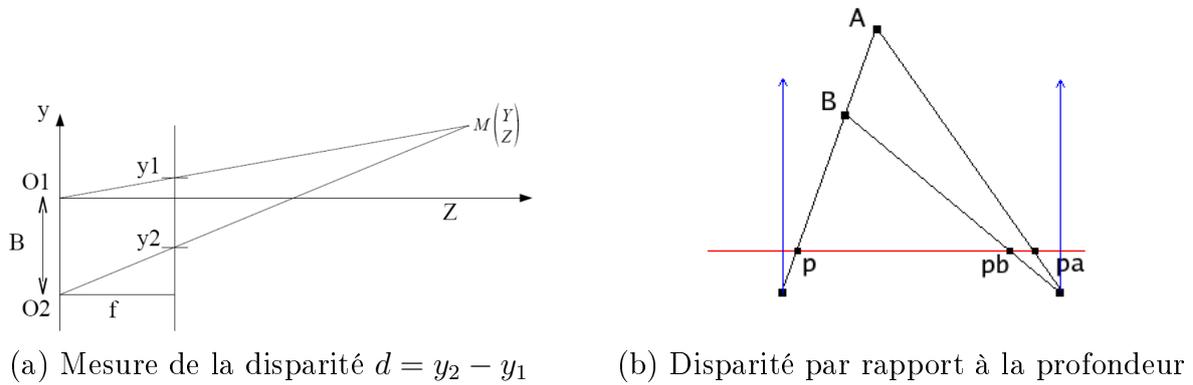


FIGURE 2.17 – Notion de disparité pour un système aligné

Application de détection de défaut

Dans l'objectif de valider la méthode par production de carte dense de disparité, nous évaluons la capacité du système à détecter des surplus ou des manques matière sur des pièces industrielles de manière quantitative et qualitative.

Conclusion

Dans ce chapitre, nous posons la problématique associée à la mesure 3D. Les différentes technologies 3D présentent chacune d'elles des avantages et des inconvénients.

Nous rappelons que le système de stéréovision doit être rapide, précis et intégrable. Si nous reprenons chacun de ces critères :

- Une solution rapide implique que les solutions nécessitant un déplacement du système d'acquisition sont inadaptées pour l'ensemble des contraintes mécaniques associées mais surtout pour le temps requis pour réaliser l'acquisition. Les cadences de production dans un contexte industriel rend ces solutions inutilisables.
- Une solution précise implique que la résolution de nos capteurs doit être suffisamment importante, éloignant pour le moment les caméras temps de vol car leur faible résolution implique soit une imprécision, soit un champ applicatif réduit.
- Une solution intégrable implique que le système d'acquisition soit flexible tout en gardant les qualités de précision.

L'ensemble de ses contraintes industrielles nous a amené à nous intéresser aux solutions de stéréovision. Néanmoins, la diversité applicative nous contraint à envisager deux alternatives : « la reconstruction 3D directe » et « la production de carte dense de disparité ».

Nous présenterons dans les deux prochains chapitres ces deux méthodes répondant par complémentarité à l'ensemble des applications 3D industrielles.

Chapitre 3

Reconstruction 3D Directe

Sommaire

3.1	Calibration monoculaire planaire	27
3.1.1	Définitions	28
3.1.2	Expression des paramètres	28
3.1.3	Estimation des paramètres	31
3.2	Sélection de points d'intérêt et appariement	32
3.3	Triangulation pour le calcul 3D	34
3.3.1	Conversion dans le repère monde	34
3.3.2	Calcul du point médian	36
3.4	Calibration spatiale	40
3.5	Expérimentation	42
3.5.1	Évaluation de la précision 3D	42
3.5.2	Influence de la précision 2D	47

Introduction

La reconstruction 3D directe consiste en la mesure tridimensionnelle d'un point M , exprimé dans un repère monde préalablement défini, à partir de ses projections dans les deux images.

Elle repose sur le principe de la triangulation 3D. En effet, une seule caméra ne peut fournir cette information car chaque point de l'image correspond à un rayon de l'espace entre le point 3D physique et le centre optique de la caméra mais nous ne pouvons déterminer quel point de ce rayon est réellement le point 3D. L'utilisation d'une seconde vue crée un second rayon avec le même point 3D et le centre optique de la seconde caméra. Il en découle que le point 3D correspond à l'intersection de ces deux rayons.

Signalons que l'autre approche pour réaliser la reconstruction 3D se base sur la connaissance de la matrice essentielle [LH81] exprimant la relation entre les deux repères caméra. Les travaux récents [LJ11] utilisent cette approche.

Nous retrouverons dans [Stu97] et [HZ00], une étude complète des techniques de triangulation 3D.

Nous noterons quelques travaux récents qui s'intéressent à la reconstruction 3D notamment dans le cas où les points images seraient bruités [VSCCTPF07] [LBDLF08]. Les travaux [MLGM09] proposent une méthode itérative pour estimer la position 3D. [CWC⁺01] tient compte des variations de la distance entre les caméras et le point 3D pour améliorer la précision de la reconstruction 3D.

Nous pouvons dès à présent identifier les problématiques de la reconstruction 3D : comment construire ces rayons uniquement à partir de la connaissance des images ? Comment garantir qu'un point de l'espace se projette bien dans les deux images ? Si tel est le cas, comment garantir que deux projections correspondent bien au même point 3D ? Ces interrogations expriment la difficulté associée à la reconstruction 3D.

Nous allons présenter dans ce chapitre les différentes étapes permettant de résoudre cette problématique qui passe par la calibration de nos deux caméras. Nous montrerons ensuite comment nous résolvons la problématique de la triangulation et comment nous améliorons la précision de la reconstruction 3D par une calibration supplémentaire.

Nous évaluerons enfin la précision de la mesure 3D selon différents critères pour le positionnement des caméras.

3.1 Calibration monoculaire planaire

La calibration monoculaire ou aussi appelée « calibrage fort » consiste à définir la fonction de projection d'une caméra exprimant ainsi la relation entre un point de l'espace et le point image associé. Elle est la base de la reconstruction 3D. Elle se décompose en deux transformations distinctes :

- la fonction entre le repère monde et le repère de la caméra exprimant les paramètres extrinsèques de la calibration.
- la fonction entre le repère caméra et le repère image exprimant les paramètres intrinsèques de la calibration

Dans notre problématique, nous nous intéressons au modèle caméra le plus utilisé, le modèle projectif ou sténopé qui inclus, pour paramètres intrinsèques, la distance focale, le point principal de l'image.

Nous noterons les travaux [Tsa87] [Bey92] qui introduisent des paramètres de distorsions basé sur un modèle non linéaire. Néanmoins, nous faisons abstraction de cette problématique dans cette étape de la calibration. En effet, nous proposons de réaliser une calibration spatiale de notre système stéréo, que nous développerons par la suite, incluant les effets non linéaires sur les pourtours des images issues de nos caméras.

Nous nous intéressons au modèle de caméra exprimé dans [Zha00], montrant une méthode pour estimer les différents paramètres de nos caméras à partir d'une liste de correspondances de points issus d'un même plan et de leur projection dans l'image. Les résultats de cette méthode montrent l'importance du choix des points de contrôle pour réaliser la calibration. L'utilisation de mire plane présentée en au moins deux orientations distinctes garantit une meilleur estimation de la calibration.

Nous noterons que de nombreux travaux se poursuivent dans la résolution de la calibration monoculaire dans une problématique de reconstruction 3D. Nous citerons des travaux récents qui s'intéressent au choix de l'élément de la scène nécessaire pour estimer les paramètres des caméras :

- [CZZ09] utilise des mires classiques et se base sur la méthode de Tsai.
- [LWZ08] utilise une mire sur une seule dimension.
- [WC09] utilise des triangles pour lesquels il réalise la mise en correspondance des trois côtés.
- [WJQW10] utilise un objet particulier.
- [CDY⁺08] utilise la projection d'une lumière structurée.
- [LWZ08] associe une lumière structurée et un motif imprimé.

L'ensemble de ses travaux évalue la qualité de la calibration monoculaire par le résultat de précision de la reconstruction 3D. [ZHZH07] va un peu plus loin en évaluant cette précision en fonction du volume décrit par les points de calibration.

3.1.1 Définitions

Nous nous intéressons au calibrage basé sur le modèle projectif linéaire et nous définissons ainsi les différents repères associés à la caméra, au plan image et au repère monde comme dans la figure 3.1.

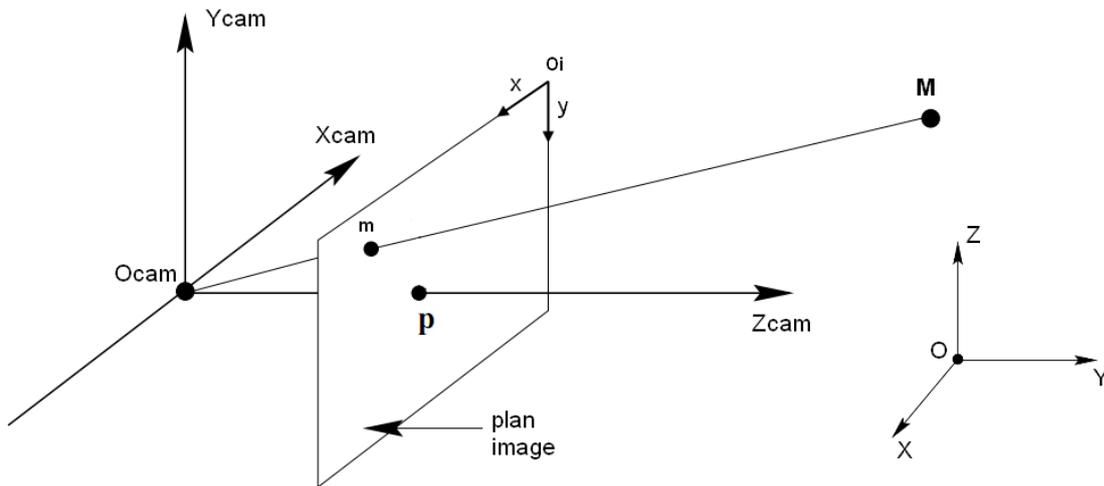


FIGURE 3.1 – Les différents repères

Suivant les notations, nous définissons :

- Le repère caméra $(O_{cam}, X_{cam}, Y_{cam}, Z_{cam})$
- Le repère image (O_i, x, y)
- Le repère monde (O, X, Y, Z)

Nous notons les propriétés suivantes :

- Tout point 3D $M(X, Y, Z)$, dans le repère monde, se projète en $m(x, y, 1)$ dans le repère image
- p est le projeté du centre optique O_{cam} de la caméra dans le plan image
- La distance $O_{cam}p$ est appelé la distance focale notée f

3.1.2 Expression des paramètres

La calibration monoculaire définit la relation entre les différents repères associé à la caméra. Les paramètres intrinsèques expriment la relation entre le repère caméra et le repère image. Les paramètres extrinsèques expriment la relation entre le repère caméra et le repère monde.

L'ensemble de ces paramètres sont estimés à partir de la matrice de projection.

3.1.2.1 La projection perspective

La projection perspective exprime la relation entre un point 3D de l'espace et la position de son projeté dans l'image. Elle représentée par la relation :

$$\lambda \begin{pmatrix} x \\ y \\ f \end{pmatrix} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (3.1)$$

qui s'écrit en coordonnées homogènes :

$$\begin{bmatrix} x \\ y \\ f \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.2)$$

3.1.2.2 Les paramètres intrinsèques

La matrice des paramètres intrinsèques exprime la relation un point image exprimé dans le repère image et ce même point exprimé dans le repère caméra. La figure 3.2 montre les paramètres d'échelle et de translation :

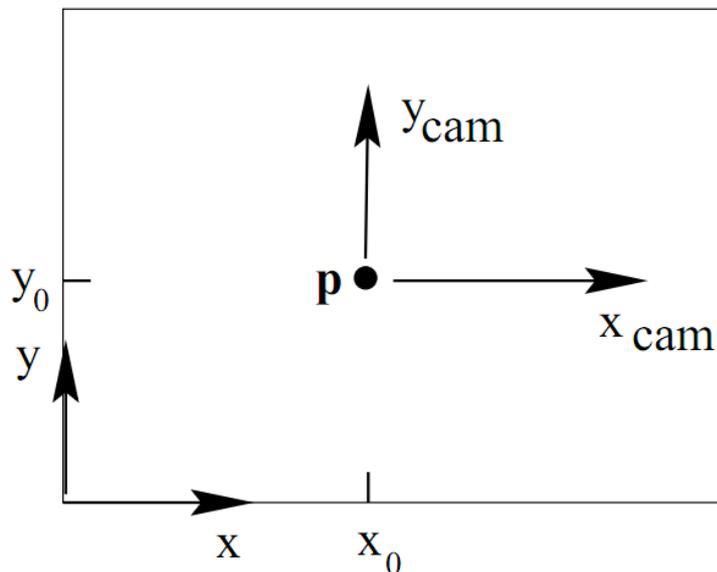


FIGURE 3.2 – La relation caméra-image

On obtient ainsi les relations suivantes :

$$\begin{cases} k_x x_{cam} = x - x_0 \\ k_y y_{cam} = y - y_0 \end{cases} \quad (3.3)$$

avec (x_0, y_0) les coordonnées du point principal dans le repère image et (k_x, k_y) les facteurs d'échelle qui sont liés à la dimension métrique des pixels de l'image.

Sous forme matricielle, la matrice K des paramètres intrinsèques s'écrit :

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K \begin{bmatrix} x_{cam} \\ y_{cam} \\ f \end{bmatrix} = \frac{1}{f} \begin{bmatrix} \alpha_x & 0 & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{cam} \\ y_{cam} \\ f \end{bmatrix} \quad (3.4)$$

avec $\alpha_x = fk_x$ et $\alpha_y = fk_y$

3.1.2.3 Les paramètres extrinsèques

La matrice des paramètres extrinsèques exprime la relation en translation et en rotation entre le repère caméra et le repère monde exprimée par la figure 3.3.

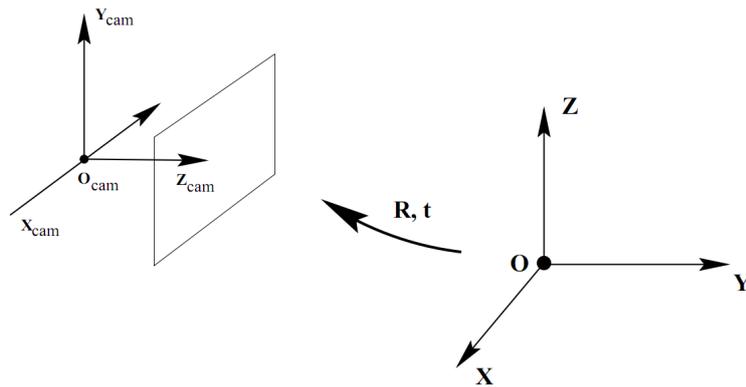


FIGURE 3.3 – La relation caméra-monde

Sous forme matricielle :

$$\begin{bmatrix} X_{cam} \\ Y_{cam} \\ Z_{cam} \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.5)$$

3.1.2.4 La matrice de projection

On en déduit directement la relation entre les points 3D et les points images par la relation matricielle :

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.6)$$

3.1.3 Estimation des paramètres

Lors de la calibration de notre caméra, nous ne possédons que les informations issues directement des images. C'est ainsi qu'à partir d'une liste de correspondances entre des points $M_i(X, Y, Z)$ dans le repère monde et des points $m_i(x, y)$ dans le repère image, nous pouvons estimer la matrice de projection de l'équation 3.6, chaque correspondance générant deux équations :

$$x_i = \frac{p_{11}X_i + p_{12}Y_i + p_{13}Z_i + p_{14}}{p_{31}X_i + p_{32}Y_i + p_{33}Z_i + p_{34}} \quad \text{et} \quad y_i = \frac{p_{21}X_i + p_{22}Y_i + p_{23}Z_i + p_{24}}{p_{31}X_i + p_{32}Y_i + p_{33}Z_i + p_{34}} \quad (3.7)$$

Pour estimer les paramètres intrinsèques et extrinsèques, la méthode par décomposition QR de la matrice de projection peut être utilisée.

Cependant, Zhang propose dans [Zha00] d'utiliser des mires planaires pour lesquels la coordonnée Z est nulle, permettant ainsi d'estimer les paramètres intrinsèques et extrinsèques directement à partir de correspondances ($M_i(X, Y, 0) \leftrightarrow m_i(x, y)$) sans passer par l'estimation de la matrice de projection. Cette méthode rend la mise en pratique de la calibration monoculaire beaucoup plus simple.

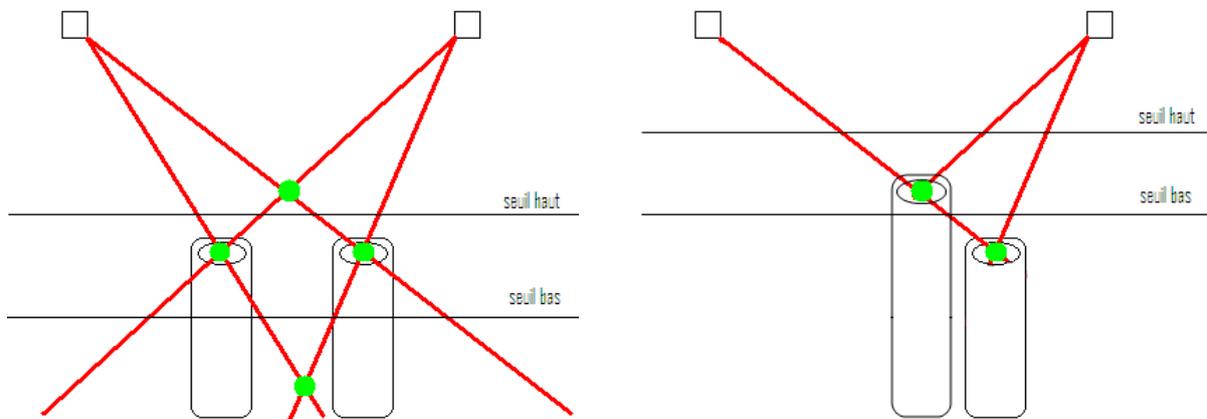
3.2 Sélection de points d'intérêt et appariement

Dans notre approche de la reconstruction 3D, nous supposons que nous sommes capables de localiser dans les deux images les projections des points 3D à mesurer. Le système de capture, par son positionnement, doit rendre l'ensemble des points 3D stéréo-visible.

L'occultation est au cœur de la problématique de l'appariement de point. Deux phénomènes en découlent :

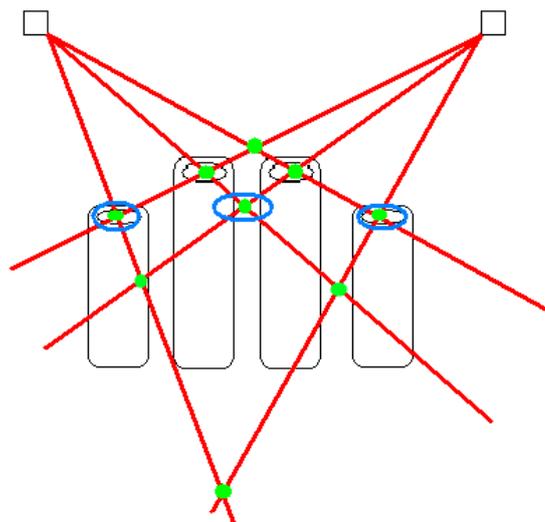
- nous ne pouvons détecter l'ensemble des points dans les deux images
- nous créons des erreurs d'appariement.

La figure 3.4 montre des cas schématisant la problématique.



(a) Comment gérer les appariements ?

(b) Comment gérer les occultations ?



(c) Conséquence de l'occultation

FIGURE 3.4 – Représentation de cas d'appariement problématiques

Dans le premier cas, les deux objets sont visibles mais il y a quatre appariements possibles. Dans ce cas de figure, l'appariement peut être résolu par des *a priori* d'ordre des objets dans la scène ou éventuellement par un seuillage sur une ou plusieurs composantes des points 3D calculés à la condition que les objets ne soient pas trop proches les uns des autres.

Le second cas exprime la problématique d'occultation qui entraîne un nombre différents de points 2D dans les deux images. Dans l'illustration présentée, les deux objets sont parfaitement alignés, auquel cas nous pouvons construire deux appariements et ils seront valides. Si l'alignement n'existait pas, ce qui est toujours le cas, alors l'un des deux sera un faux appariement et donc la question se pose : lequel choisir ? Nous pourrions résoudre ce cas par seuillage également avec l'*a priori* que ce sont les objets les plus proches qui occultent les objets les plus éloignés.

Le dernier cas présenté exprime également la problématique d'occultation mais, ici, l'appariement « créerait » des positions 3d inexistantes que nous sommes incapables d'identifier. En effet, dans les deux images, nous observons le même nombre d'objets mais ils ne correspondent pas aux mêmes objets physiques. Si nous nous basons sur la notion d'ordre des objets, nous allons créer trois appariements (en bleu) dont l'un sera un faux appariement, le point du milieu. Le seuillage ne pourra, non plus, le supprimer.

Ces cas problématiques montrent la difficulté à identifier les bons appariements lorsque nous souhaitons localiser des éléments semblables. La connaissance de la scène devient primordiale et le système d'acquisition doit être positionné de sorte que l'ensemble des éléments de la scène soit visible par les deux caméras.

3.3 Triangulation pour le calcul 3D

La triangulation consiste, à partir de la calibration de chaque caméra et d'une paire de points image qui se correspondent, d'estimer la position 3D de point de l'espace.

L'objectif est de définir la position de chaque caméra C_1 et C_2 et des deux points images respectivement P_1 et P_2 , dans un même repère, le repère monde. Nous construisons ensuite les deux droites (C_1P_1) et (C_2P_2) et nous calculons le point d'intersection des deux droites.

En pratique, le point d'intersection n'existe pas et il sera donc défini comme le point médian du segment qui minimise la distance inter-droite. La figure 3.5 schématise le principe de distance inter-droite.

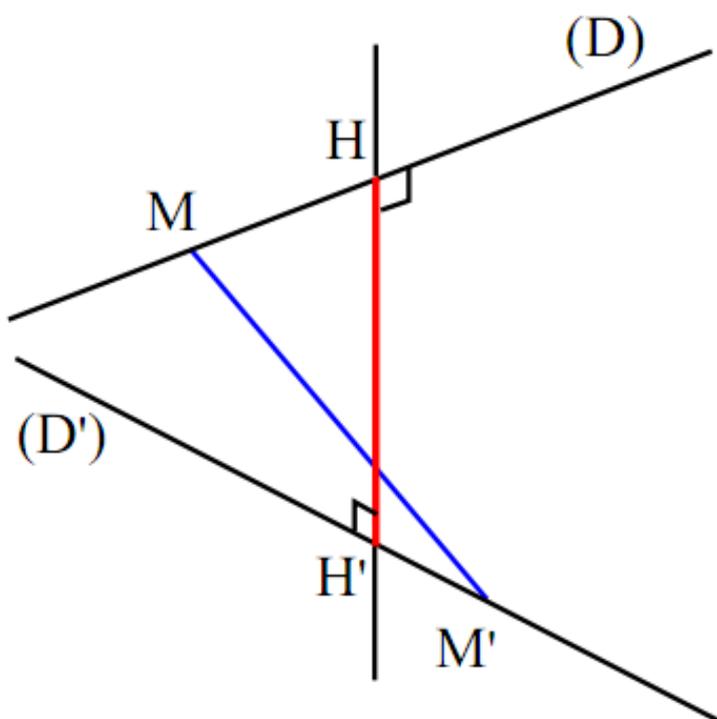


FIGURE 3.5 – Segment qui minimise la distance inter-droite

Étant donné deux droites (D) et (D') de l'espace, on appelle distance entre (D) et (D') la valeur minimale de la distance MM' où M est un point qui décrit (D) et M' un point qui décrit (D') . On démontrera que cette distance est la distance HH' correspondant à la perpendiculaire (HH') commune aux deux droites.

La figure 3.6 exprime l'intersection de droite par rapport à notre problématique.

3.3.1 Conversion dans le repère monde

La première problématique de la triangulation est de construire les deux droites de reconstruction 3D. Il est donc nécessaire que les centres optiques des caméras et les pro-

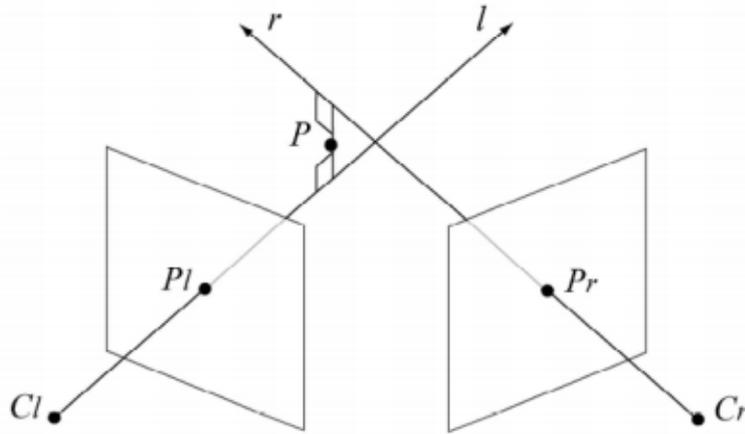


FIGURE 3.6 – Segment qui minimise la distance inter-droite

jections du point 3D dans les images soient exprimés dans le même repère.

3.3.1.1 Calcul de la pose caméra

Les paramètres extrinsèques exprimant la relation entre le repère caméra et le repère monde défini, nous permet de déduire directement la pose caméra. En effet la matrice de projection contient les informations de translation entre les deux repères. Plus précisément, le vecteur translation de la matrice de projection exprime la position, en coordonnées métriques, du point d'origine dans le repère monde par rapport au centre optique de la caméra.

On en déduit que la position de la caméra est le vecteur translation de la matrice de projection inverse. Et, la matrice de projection étant une matrice orthogonale, on définit la pose caméra comme le vecteur suivant :

$$T' = [-R]^T T \quad (3.8)$$

avec R la matrice de rotation et T le vecteur de translation de la matrice de projection.

3.3.1.2 Expression des points images dans le repère monde

Le point image est défini par les coordonnées (u, v) dans le repère image. L'objectif est d'exprimer les coordonnées de ce point dans le repère monde.

La résolution s'exécute en deux étapes :

- Expression des points image dans le repère caméra
- Expression des points images dans le repère monde

La première étape se base sur le modèle de la caméra (voir la figure 3.7) et les paramètres intrinsèques associés.

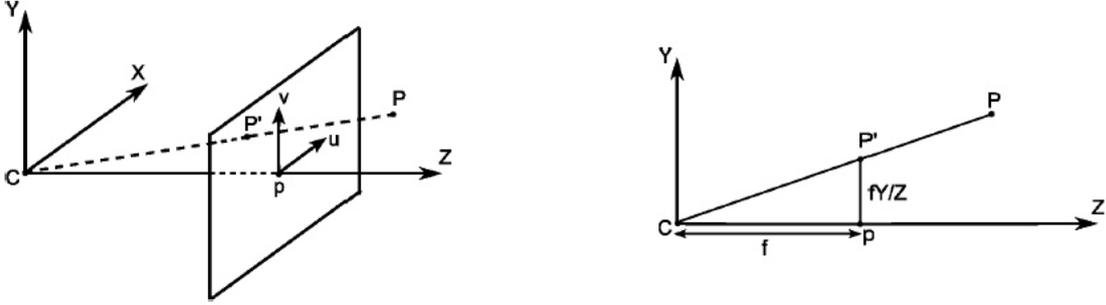


FIGURE 3.7 – Le modèle de caméra

En effet, on déduit directement la coordonnées Z par la distance focale. Les coordonnées X et Y sont définis par une simple translation au point principale de l'image (qui est la projection du centre optique dans le repère image) et l'application d'un facteur d'échelle de la valeur métrique (c_x, c_y) en abscisse et en ordonnée du pixel.

Ainsi, on exprime la conversion du point image du repère image vers le repère caméra à partir de la matrice modélisant les paramètres intrinsèques de la caméra par :

$$P_{image} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \implies P_{camera} = \begin{bmatrix} \frac{u-p_x}{c_x} \\ \frac{v-p_y}{c_y} \\ f \end{bmatrix} \quad (3.9)$$

avec $f = f_x p_x$ ou $f = f_y p_y$

La seconde étape est le passage des coordonnées caméra aux coordonnées monde. Nous réutilisons les paramètres extrinsèques de la caméra et calculons les coordonnées monde du point image par application de la matrice de projection inverse :

$$P_{monde} = \begin{bmatrix} P_x \\ P_y \\ p_z \end{bmatrix} = \begin{bmatrix} r'_{11} & r'_{12} & r'_{13} & t'_x \\ r'_{21} & r'_{22} & r'_{23} & t'_y \\ r'_{31} & r'_{32} & r'_{33} & t'_z \end{bmatrix} \begin{bmatrix} \frac{u-p_x}{c_x} \\ \frac{v-p_y}{c_y} \\ f \end{bmatrix} \quad (3.10)$$

3.3.2 Calcul du point médian

3.3.2.1 Définitions

Nous utilisons la représentation paramétrique, de paramètre t , d'une droite dans l'espace. Ainsi, La droite (D) passant par le point $O = (x_0, y_0, z_0)$ et de vecteur directeur

$\vec{u} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix}$ est l'ensemble des points M tels que :

$$\begin{cases} x = u_x t + x_O \\ y = u_y t + y_O \\ z = u_z t + z_O \end{cases} \quad (3.11)$$

Soient (D) et (D') deux droites non parallèles et Δ la perpendiculaire commune aux deux droites passant par $A \in (D)$ et $A' \in (D')$, la distance minimale entre les deux droites est AA' . En effet, pour tout points M_1 et M_2 appartenant respectivement à (D) et (D') , la relation de Charles nous amène à $\overrightarrow{M_1 M_2} = \overrightarrow{M_1 A_1} + \overrightarrow{A_1 A_2} + \overrightarrow{A_2 M_2}$. Comme $\overrightarrow{A_1 A_2}$ est orthogonal avec $\overrightarrow{M_1 A_1}$ et avec $\overrightarrow{A_2 M_2}$ alors les produits scalaires suivants sont nuls :

$$\begin{cases} \overrightarrow{A_1 A_2} \cdot \overrightarrow{M_1 A_1} = 0 \\ \overrightarrow{A_1 A_2} \cdot \overrightarrow{A_2 M_2} = 0 \end{cases} \quad (3.12)$$

et comme le produit scalaire est distributif par rapport à l'addition, alors $\overrightarrow{A_1 A_2}$ et $(\overrightarrow{M_1 A_1} + \overrightarrow{A_2 M_2})$ sont orthogonaux.

On en déduit que $M_1 M_2^2 = A_1 A_2^2 + \|\overrightarrow{M_1 A_1} + \overrightarrow{A_2 M_2}\|^2$ et donc on obtient $M_1 M_2 \geq A_1 A_2$

Au final, la droite qui minimise la distance inter-droite est la perpendiculaire commune aux deux droites.

3.3.2.2 Résolution du point médian

Posons :

- C_1 et C_2 les centres optiques des caméras,
- P_1 et P_2 les points images définis dans le repère monde,
- \vec{u}_1 et \vec{u}_2 les vecteurs directeurs des deux droites $(C_1 P_1)$ et $(C_2 P_2)$,
- \vec{n} un vecteur normal aux deux droites $(C_1 P_1)$ et $(C_2 P_2)$, $[M_1 M_2]$ le segment qui minimise la distance entre $(C_1 P_1)$ et $(C_2 P_2)$ tel que $M_1 \in (C_1 P_1)$ et $M_2 \in (C_2 P_2)$,
- M le point médian du segment $[M_1 M_2]$,
- d la longueur du segment $[M_1 M_2]$.

Nous caractérisons les deux droites par leur représentation paramétrique suivante :

$$\begin{cases} x_1 = u_{1x}t_1 + C_{1x} \\ y_1 = u_{1y}t_1 + C_{1y} \\ z_1 = u_{1z}t_1 + C_{1z} \end{cases} \quad (3.13)$$

et

$$\begin{cases} x_2 = u_{2x}t_2 + C_{2x} \\ y_2 = u_{2y}t_2 + C_{2y} \\ z_2 = u_{2z}t_2 + C_{2z} \end{cases} \quad (3.14)$$

avec $\vec{u}_1 = \frac{\overrightarrow{C_1P_1}}{\|C_1P_1\|}$ et $\vec{u}_2 = \frac{\overrightarrow{C_2P_2}}{\|C_2P_2\|}$.

\vec{n} étant un vecteur normal aux deux droites (C_1P_1) et (C_2P_2) alors $\vec{n} = \overrightarrow{C_1P_1} \wedge \overrightarrow{C_2P_2}$ et donc :

$$\vec{n} = \begin{bmatrix} (P_{1y} - C_{1y})(P_{2z} - C_{2z}) - (P_{1z} - C_{1z})(P_{2y} - C_{2y}) \\ (P_{1z} - C_{1z})(P_{2x} - C_{2x}) - (P_{1x} - C_{1x})(P_{2z} - C_{2z}) \\ (P_{1x} - C_{1x})(P_{2y} - C_{2y}) - (P_{1y} - C_{1y})(P_{2x} - C_{2x}) \end{bmatrix} \quad (3.15)$$

Nous aurions pu bien évidemment dire que \vec{n} est un vecteur normal à \vec{u}_1 et \vec{u}_2 et ainsi l'expression de \vec{n} serait simplifiée par :

$$\vec{n} = \begin{bmatrix} u_{1y}u_{2z} - u_{1z}u_{2y} \\ u_{1z}u_{2x} - u_{1x}u_{2z} \\ u_{1x}u_{2y} - u_{1y}u_{2x} \end{bmatrix} \quad (3.16)$$

L'objectif est maintenant de montrer que $\overrightarrow{M_1M_2}$ et \vec{n} sont colinéaires, c'est à dire qu'il existe un paramètre $k \in \mathbf{R}$ tel que :

$$\begin{cases} M_{2x} - M_{1x} = kn_x \\ M_{2y} - M_{1y} = kn_y \\ M_{2z} - M_{1z} = kn_z \end{cases} \quad (3.17)$$

Si nous reprenons l'expression des droites (C_1P_1) et (C_2P_2) , nous obtenons le système d'équations à 3 inconnus t_1, t_2 et k suivant :

$$\begin{cases} u_{2x}t_2 + C_{2x} - u_{1x}t_1 - C_{1x} = kn_x \\ u_{2y}t_2 + C_{2y} - u_{1y}t_1 - C_{1y} = kn_y \\ u_{2z}t_2 + C_{2z} - u_{1z}t_1 - C_{1z} = kn_z \end{cases} \quad (3.18)$$

En supposons que $\vec{n} \neq \vec{0}$, auquel cas, l'intersection des droites (C_1P_1) et (C_2P_2) existerait, le système d'équations se résume à résoudre le système matricielle suivant :

$$\begin{bmatrix} -u_{1x} & u_{2x} - n_x \\ -u_{1y} & u_{2y} - n_y \\ -u_{1z} & u_{2z} - n_z \end{bmatrix} = \begin{bmatrix} t_1 \\ t_2 \\ k \end{bmatrix} \begin{bmatrix} C_{1x} - C_{2x} \\ C_{1y} - C_{2y} \\ C_{1z} - C_{2z} \end{bmatrix} \quad (3.19)$$

Nous en déduisons la position 3D des points M_1, M_2 par :

$$M_1 = \begin{bmatrix} u_{1x}t_1 + C_{1x} \\ u_{1y}t_1 + C_{1y} \\ u_{1z}t_1 + C_{1z} \end{bmatrix} \quad (3.20)$$

et

$$M_2 = \begin{bmatrix} u_{2x}t_2 + C_{2x} \\ u_{2y}t_2 + C_{2y} \\ u_{2z}t_2 + C_{2z} \end{bmatrix} \quad (3.21)$$

Finalement, la reconstruction 3D pour les deux points correspondants P_1 et P_2 est définie par :

$$M = \frac{1}{2} \begin{bmatrix} M_{1x} + M_{2x} \\ M_{1y} + M_{2y} \\ M_{1z} + M_{2z} \end{bmatrix} \quad (3.22)$$

3.4 Calibration spatiale

L'objectif de la calibration spatiale est d'améliorer la précision de la mesure 3D pour l'ensemble des points du champ commun aux deux caméras. Nous proposons ici d'apporter une correction aux données 3D mesurées par apprentissage de point dont on connaît la position réelle dans l'espace.

Nous utilisons une régression polynomiale multiple d'ordre 2 avec interaction. Les données d'entrée sont les coordonnées 3D (X_m, Y_m, Z_m) mesurées par triangulation qui représentent les variables explicatives et les données de sortie sont les coordonnées 3D réelles (X_r, Y_r, Z_r) qui représentent les variables dépendantes.

Nous construisons donc le modèle suivant :

$$\begin{cases} X_r = x_0 + x_1X_m + x_2Y_m + x_3Z_m + x_4X_mY_m + x_5X_mZ_m + x_6Y_mZ_m + x_7X_m^2 + x_8Y_m^2 + x_9Z_m^2 \\ Y_r = y_0 + y_1X_m + y_2Y_m + y_3Z_m + y_4X_mY_m + y_5X_mZ_m + y_6Y_mZ_m + y_7X_m^2 + y_8Y_m^2 + y_9Z_m^2 \\ Z_r = z_0 + z_1X_m + z_2Y_m + z_3Z_m + z_4X_mY_m + z_5X_mZ_m + z_6Y_mZ_m + z_7X_m^2 + z_8Y_m^2 + z_9Z_m^2 \end{cases} \quad (3.23)$$

Chaque axe est modélisé par un plan formé d'un ensemble de paraboles dépendantes des deux autres axes. La figure 3.8 montre une représentation spatiale de la variable Z par rapport au deux axes X et Y. Cette représentation est également définie pour la variable X par rapport aux axes Y et Z ainsi que la variable Y par rapport à l'axe X et Z.

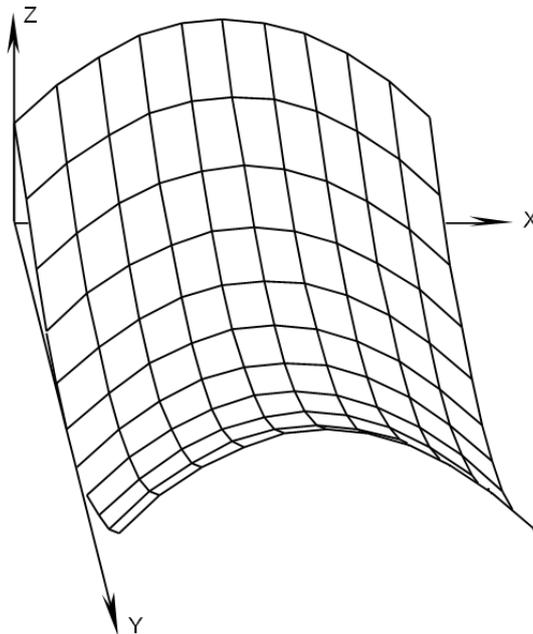


FIGURE 3.8 – Représentation spatiale de la régression polynomiale

Nous verrons dans la partie expérimentation que l'erreur de positionnement 3D pour les

trois axes décrit cette tendance, justifiant ainsi son utilisation pour améliorer la précision 3D.

3.5 Expérimentation

3.5.1 Évaluation de la précision 3D

Nous avons mis en place un système stéréoscopique configurable, nous apportant les quatre degrés de liberté suivants :

- La pose de la tête stéréoscopique dans l'espace 3D.
- L'écartement entre les deux caméras E
- L'angle formé par les axes optiques des deux caméras α
- L'angle entre les axes optiques et la normale au repère monde θ

La figure 3.9 présente ces différents paramètres pour configurer le positionnement des caméras.

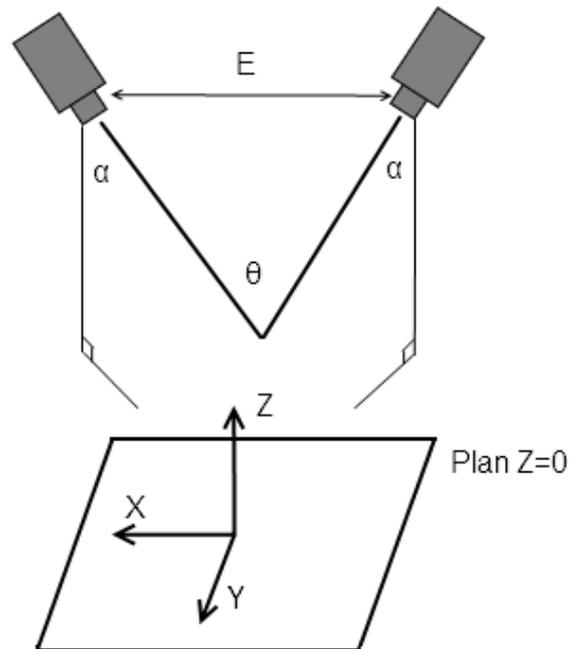


FIGURE 3.9 – Expression des paramètres de positionnement des deux caméras

Avec l'assistance d'un robot calibré sur le repère monde défini, nous présentons un motif à différentes positions de l'espace qui discrétisent un volume constant pour toutes les configurations du système stéréoscopique. L'objectif est de définir la précision de la reconstruction 3D directe par comparaison entre les positions 3D fournies par le robot et les positions 3D mesurées. Nous définissons les erreurs sur les trois axes et en distance par :

$$\begin{aligned}
E_x &= (X_{robot} - X_{mesure}) \\
E_y &= (Y_{robot} - Y_{mesure}) \\
E_z &= (Z_{robot} - Z_{mesure})
\end{aligned} \tag{3.24}$$

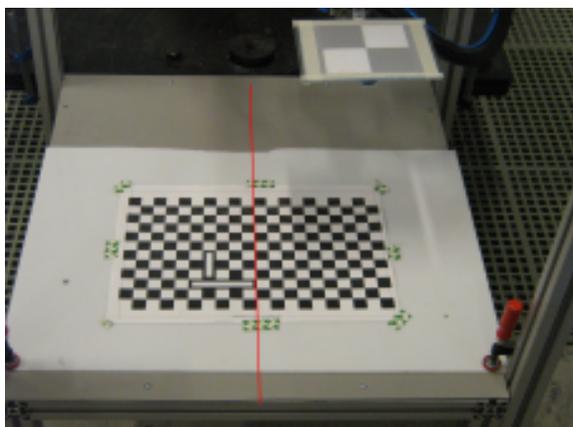
et

$$E_d = \sqrt{(X_{robot} - X_{mesure})^2 + (Y_{robot} - Y_{mesure})^2 + (Z_{robot} - Z_{mesure})^2} \tag{3.25}$$

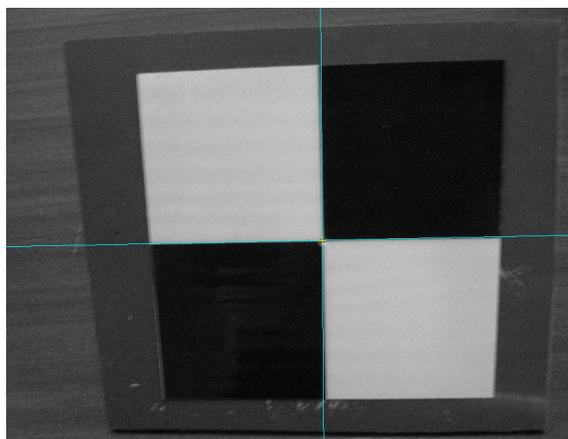
Le robot recale son repère sur le repère monde défini par la mire illustrée en figure 3.10(a). Le système définit les paramètres extrinsèques sur cette même mire pour obtenir la correspondance avec le robot.

En phase de déplacement, le robot présente le motif, en figure 3.10(b), avec une position que nous pouvons définir précisément. Néanmoins, l'orientation du motif étant invariant lors du déplacement du robot, le décalage sur les trois axes entre la position fournie par le robot et la position réelle du point central du motif est constante. Finalement, lors du calcul de l'erreur entre la position 3D fourni par le robot et la position 3D mesurée par notre système, la valeur moyenne de l'erreur correspondra à ce décalage.

C'est ainsi que nous évaluons la précision de mesure sur la valeur de l'écart-type de l'erreur. De plus, lors de la phase de calibration spatiale qui définit le modèle entre les points 3D robot et les points 3D mesurés, cette valeur moyenne et donc ce décalage y sera intégrée dans les composantes constantes de l'équation 3.23.



(a) Mire pour le repère monde



(b) Motif

FIGURE 3.10 – Éléments de calibration et de détection pour l'évaluation de la précision 3D

Nous dénombrons environ 500 points de mesure pour un volume de l'ordre 10 dm^3 . L'évaluation est réalisée sur trois écartements inter-caméra, trois angles "axes optiques-normale" et deux distances "caméras-scène". L'angle inter-caméra est adapté pour que le volume d'analyse soit dans le champ des deux caméras.

La précision est évaluée selon l'écart-type de l'erreur 3D. Nous faisons abstraction de la valeur moyenne de chaque axe car elles représentent un offset qui s'intègre dans la calibration spatiale. Elles ne modifient en rien notre capacité à fournir une mesure précise. Nous observons l'erreur en distance entre les points réels et les points mesurés mais également l'erreur en fonction des trois axes X, Y et Z.

3.5.1.1 Influence dans le volume

Nous visualisons sur la figure 3.11, la répartition de l'erreur en distance selon l'axe X et Y et pour un Z fixé au Z moyen tel que :

$$z = \frac{z_{max} - z_{min}}{2} \quad (3.26)$$

Ces résultats sont issus des caméras de résolution 768x576 et tiennent compte des différents écartements inter-caméra E et d'angles entre les axes optiques et la normale α .

L'évaluation de l'erreur en distance est définie pour un volume fixe de largeur 30 centimètres (en X) et de longueur 28 centimètres (en Y). L'axe optique des caméras a été centré manuellement vers le point central du volume.

Nous observons que pour l'ensemble des configurations (angle inter-caméra, écartement), l'erreur en distance est minimale pour les points au centres du volume qui correspondent à des points projetés dans le centre des deux images. Nous expliquons ce phénomène par la légère distorsion des images. En effet, la phase de triangulation 3D se base sur la construction de droite entre le centre optique et le point image exprimé dans le repère monde. Si nous tenions compte de la distorsion, nous devrions construire des courbes qui dépendent des paramètres de distorsion. Nous aurions pu estimer ces paramètres lors de la phase de calibration monoculaire mais nous avons fait le choix d'utiliser une calibration spatiale qui corrigera ce phénomène.

D'ailleurs, l'évolution de l'erreur suit la tendance de la figure 3.8 exprimant la régression polynomiale.

3.5.1.2 Influence de la résolution des caméras

Nous avons réalisés des essais avec deux paires de caméras de résolutions différentes, soit une paire de caméras de résolution 768x576 et une autre de résolution 1360x1024.

La figure 3.12 montre les erreurs pour chacun des axes et en distance pour les différentes positionnement des caméras. L'axe des abscisses sont des indices représentant les différentes configurations d'angle inter-caméra, d'écartement des caméras et d'angle avec la normale. Il n'y a pas de notion d'ordre dans les mesures, nous montrons uniquement la différence d'erreur entre les deux résolutions de caméra.

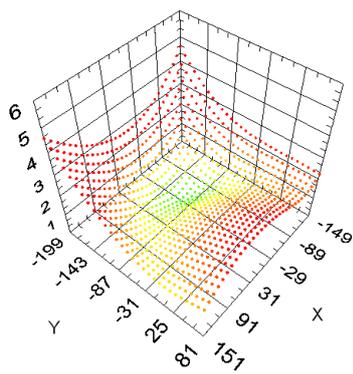
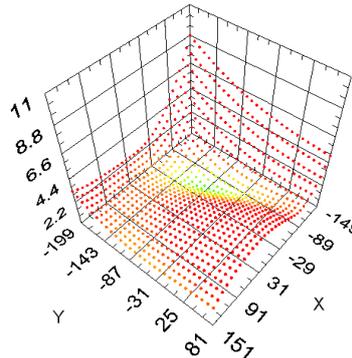
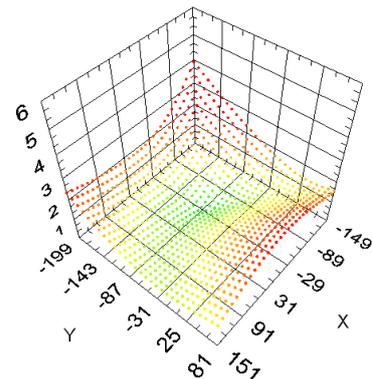
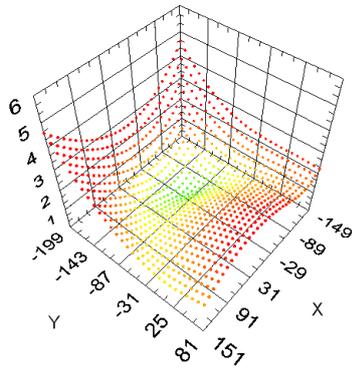
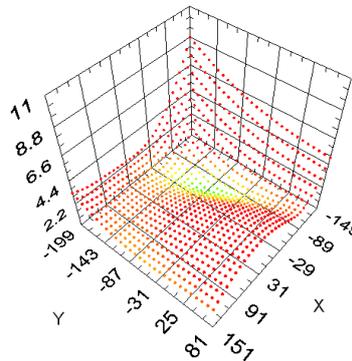
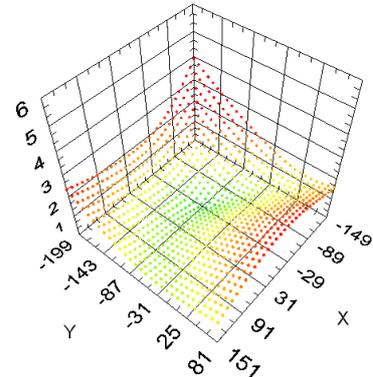
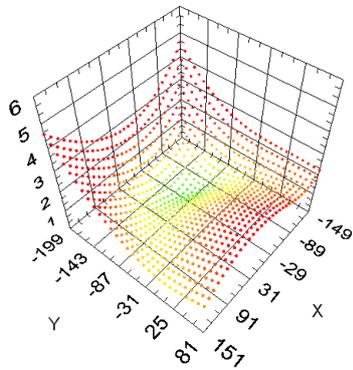
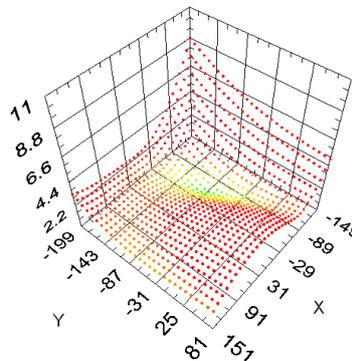
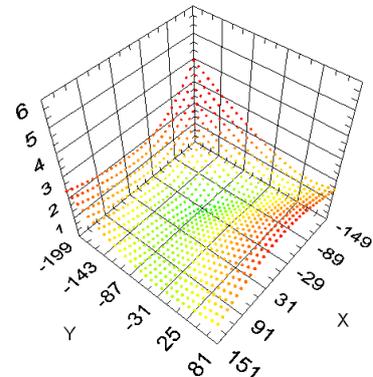
(a) $E = 400\text{mm}$ et $\alpha = 15^\circ$ (b) $E = 400\text{mm}$ et $\alpha = 30^\circ$ (c) $E = 400\text{mm}$ et $\alpha = 45^\circ$ (d) $E = 600\text{mm}$ et $\alpha = 15^\circ$ (e) $E = 600\text{mm}$ et $\alpha = 30^\circ$ (f) $E = 600\text{mm}$ et $\alpha = 45^\circ$ (g) $E = 800\text{mm}$ et $\alpha = 15^\circ$ (h) $E = 800\text{mm}$ et $\alpha = 30^\circ$ (i) $E = 800\text{mm}$ et $\alpha = 45^\circ$

FIGURE 3.11 – Évolution de l'erreur en distance (mm) pour le plan moyen

Cette évaluation ne tiens pas compte des éléments de calibration spatiale car elle pourrait dissimuler l'influence de la résolution des caméras que nous souhaitons mettre en avant.

Nous observons que, pour une résolution plus grande, les erreurs sont moindres pour l'ensemble des configurations, excepté pour l'erreur en X. Mais, en terme de précision calculée par la valeur de l'écart-type de l'erreur en distance, nous passons d'un écart-type

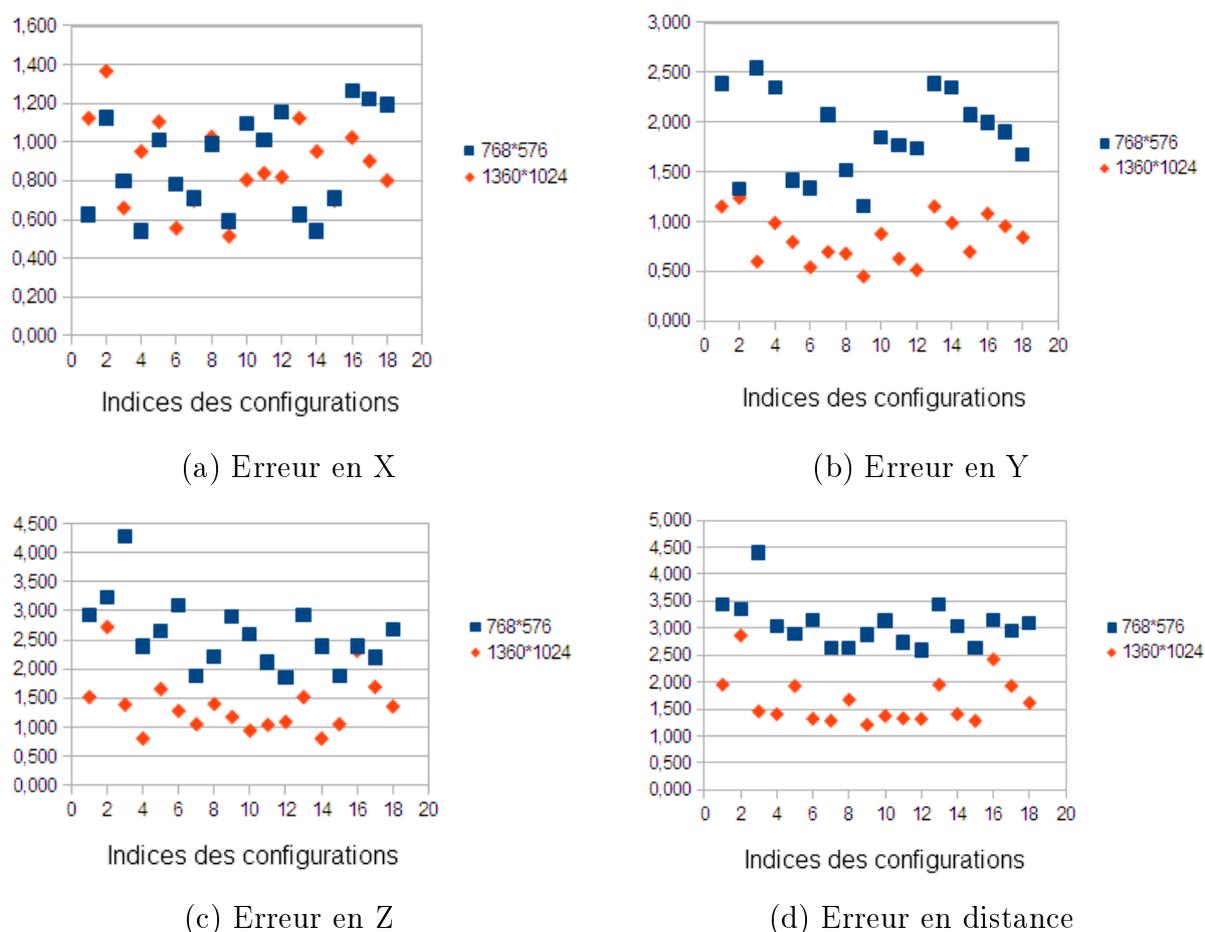


FIGURE 3.12 – L'écart-type de l'erreur (mm) sur les trois axes et en distance pour deux résolutions de caméra

compris entre $2.5mm$ et $3.5mm$ à un écart-type compris entre $1mm$ et $2mm$.

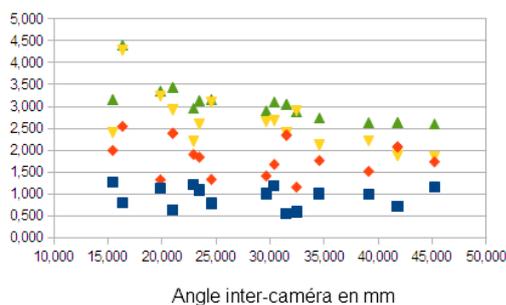
Ce phénomène s'explique par la précision de la détection du motif dans les deux images. Plus la résolution des caméras est grande, plus la localisation 2D est précise. Au final, une meilleure précision 2D minimise l'erreur de construction des droites pour la triangulation 3D.

3.5.1.3 Influence de l'angle inter-caméra

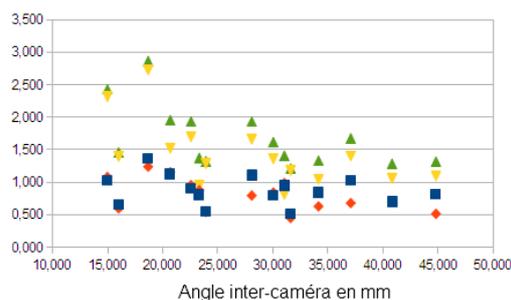
Nous voulons montrer ici l'influence de l'angle inter-caméra θ sur la précision de mesure. Ainsi, pour chaque configuration des caméras, nous mesurons l'angle θ_{moyen} moyen formé par l'ensemble des points du volume et les deux caméras, et nous évaluons les variations de l'écart-type selon cet angle.

La figure 3.13 montre les résultats pour les deux résolutions de caméras.

L'erreur sur les trois axes et en distance décroît lorsque l'angle inter-caméra tend vers 45° .



(a) Résolution 768x576



(b) Résolution 1360x1024

FIGURE 3.13 – Évolution de l'écart-type de l'erreur (mm) sur les trois axes et en distance en fonction de l'angle moyen formé par les deux caméras et les points mesurés

3.5.1.4 Influence de la correction par régression

Nous souhaitons évaluer l'influence de la correction par régression 3D sur la valeur de l'écart-type de l'erreur de précision. Dans ce cas, l'apprentissage du modèle de régression est réalisé à partir de l'ensemble des points de mesure et la précision est évaluée à partir des mêmes points.

Nous pouvons, dans premier temps, visualiser la tendance de l'erreur dans l'espace pour un Z fixé.

Nous observons que l'erreur ne suit plus la tendance sans régression, c'est à dire une erreur minimale pour les points au centre du volume. La tendance est liée au modèle de régression, l'erreur est aléatoire dans le volume.

La figure 3.15(a)(b)(c)(d) montre l'écart-type des erreurs pour les trois axes et en distance pour l'ensembles des configurations de caméras. Nous pouvons constater l'amélioration par rapport aux résultats de la figure 3.12.

La figure 3.15(e)(f) exprime la relation entre l'angle inter-caméra et la précision, avec et sans l'apport de la correction par régression.

Nous constatons une diminution conséquente de l'erreur 3D. Sans régression, les valeurs d'écart-type oscillait entre 2.5mm et 3.5mm pour la plus faible résolution et entre 1mm et 2mm pour la plus grande résolution. Dorénavant, les valeurs oscille entre 0.25mm et 0.50mm, et entre 0.15mm et 0.40mm sur l'ensemble des configurations des caméras. Nous conservons toujours la différence de précision entre les deux résolutions.

L'influence de l'angle est moins prononcé que pour le cas non corrigé.

3.5.2 Influence de la précision 2D

Dans cette dernière partie, nous souhaitons montrer l'influence de la mesure 2D de la position du motif dans les images sur la précision de reconstruction 3D. Dans cette

optique, nous avons appliqué une erreur gaussienne de valeur moyenne nulle sur les données d'entrée. En modifiant la variance de l'erreur gaussienne, nous observons l'impact sur l'écart-type de l'erreur en distance.

La figure 3.16 montrent les résultats pour les deux résolutions de caméra pour les cas avec et sans correction par régression 3D.

Le premier constat est que, dans tous les cas de résolution et de régression envisagés, plus l'erreur 2D augmente, plus l'erreur 3D augmente également, et cela pour l'ensemble des configurations de caméras.

Le second constat est que, l'évolution de l'erreur 3D par rapport à l'erreur 2D varie selon l'angle inter-caméra. En effet, plus l'angle inter-caméra augmente, plus l'erreur 3D augmente rapidement.

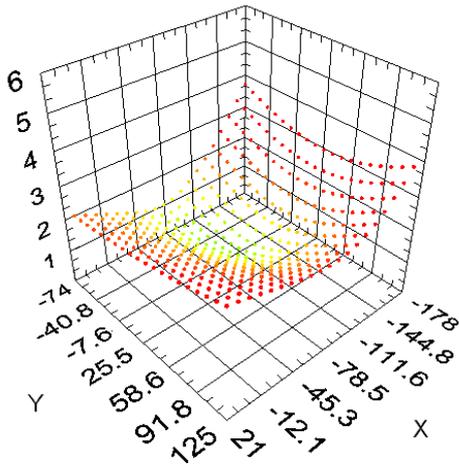
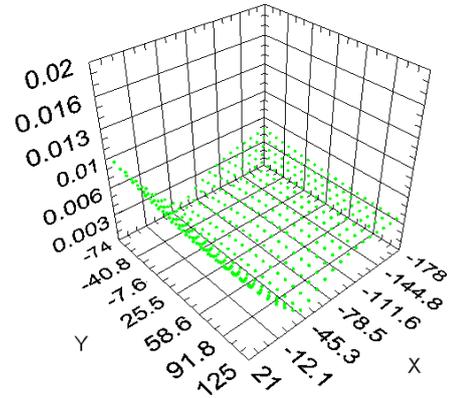
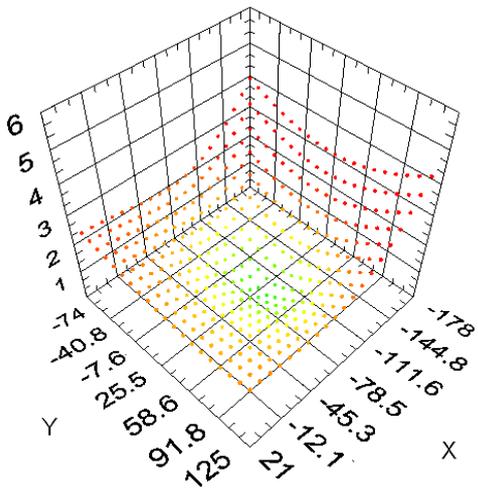
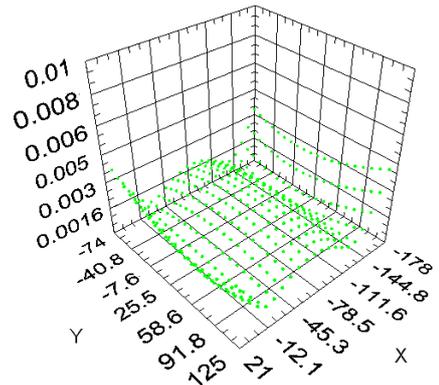
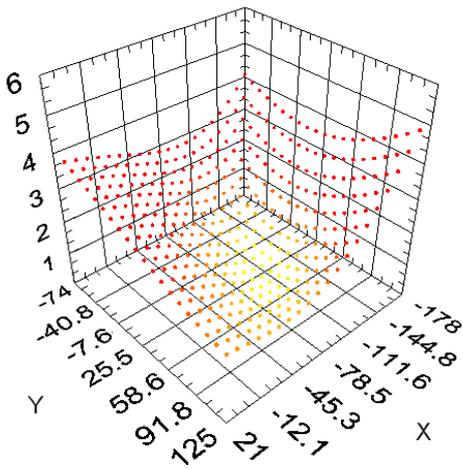
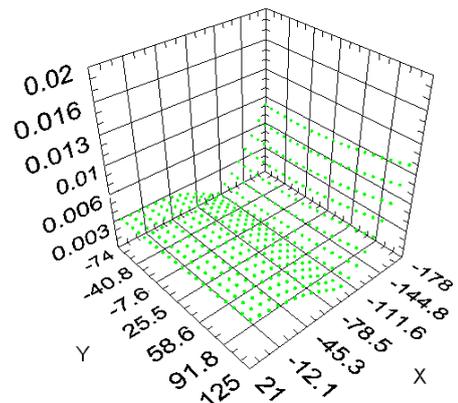
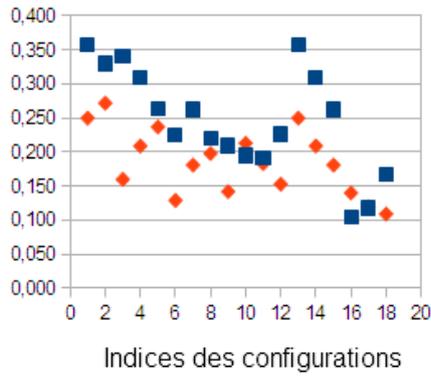
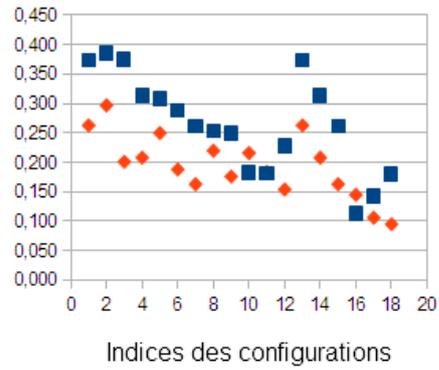
(a) $Z = Z_{min}$, sans régression(b) $Z = Z_{min}$, avec régression(c) $Z = Z_{moy}$, sans régression(d) $Z = Z_{moy}$, avec régression(e) $Z = Z_{max}$, sans régression(f) $Z = Z_{max}$, avec régression

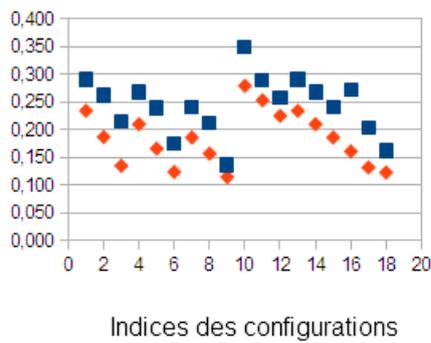
FIGURE 3.14 – Évolution de l'erreur (mm) en distance, pour trois plan, avec et sans régression



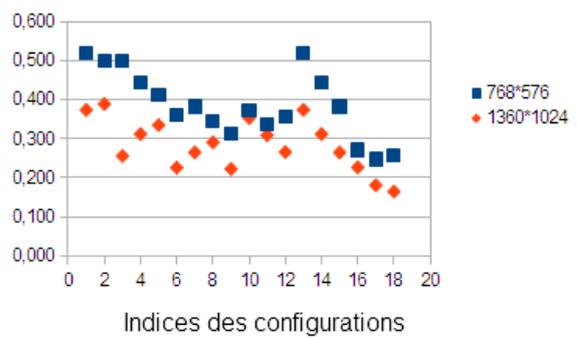
(a) Erreur en X



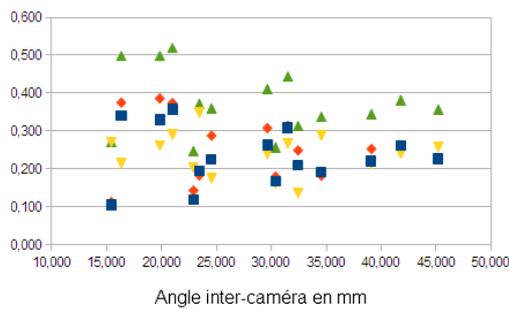
(b) Erreur en Y



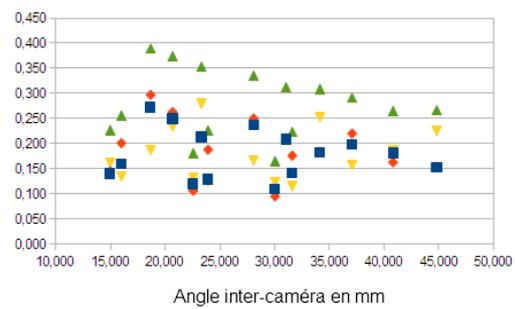
(c) Erreur en Z



(d) Erreur en distance

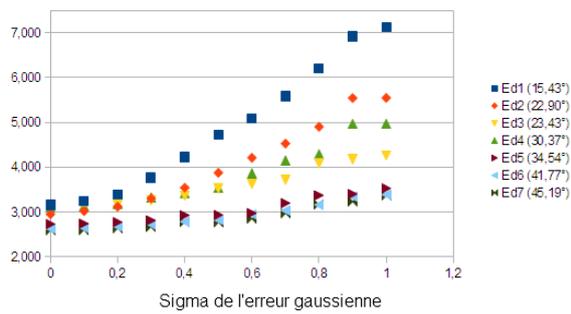


(e) Erreur selon l'angle θ , sans régression

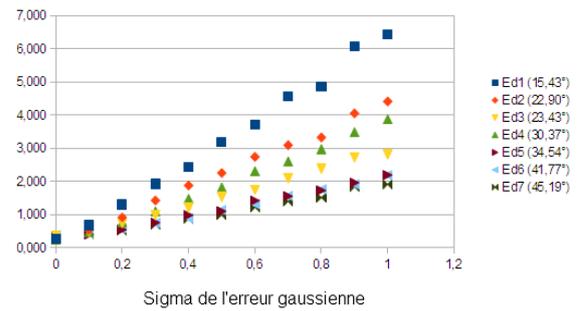


(f) Erreur selon l'angle θ , avec régression

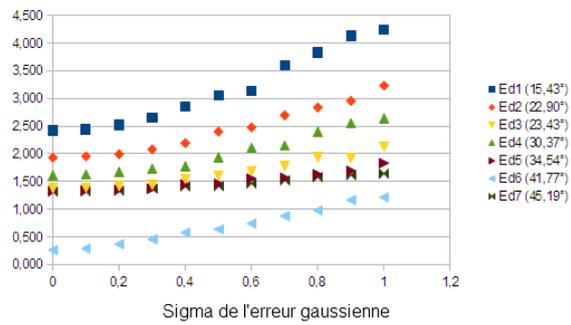
FIGURE 3.15 – Évolution de l'écart-type de l'erreur (mm) sur les trois axes et en distance, avec et sans régression 3D



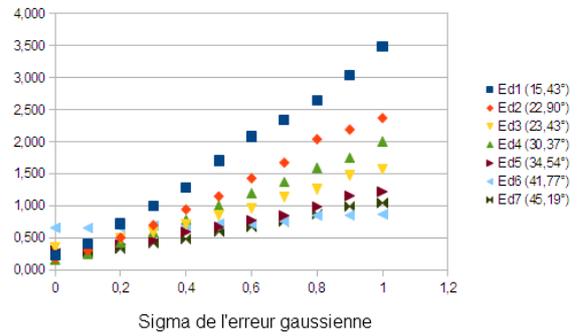
(a) Résolution 768x576 sans régression



(b) Résolution 768x576 avec régression



(c) Résolution 1360x1024 sans régression



(d) Résolution 1360x1024 avec régression

FIGURE 3.16 – Évolution de l'écart-type de l'erreur (mm) en distance en fonction de l'erreur 2D, pour les deux résolutions de caméras, avec et sans régression 3D

Conclusion

Nous avons présenté dans ce chapitre la méthode par reconstruction 3D directe qui est une solution pour localiser tout point de l'espace sous contrainte qu'il soit visible par nos deux caméras et que nous sommes capables de réaliser l'appariement de leur projection dans les deux images.

Nous avons présenté les différentes étapes de calibration monoculaire, de triangulation 3D et d'amélioration par régression 3D et d'évaluation de la mesure 3D. Puis nous avons évalué les performances de la reconstruction 3D.

La précision de la mesure 3D dépend de plusieurs paramètres. Tout d'abord, la précision de détection 2D des points caractéristiques dans les images garantit une meilleure précision 3D. L'utilisation de caméras de plus grande résolution est une solution.

Un grand angle formé par les centres optiques et le point 3D permet d'améliorer la précision 3D. Une augmentation de l'écartement ou un rapprochement des caméras permet d'augmenter cet angle. Néanmoins, les contraintes de bon appariement des points image limitent la pose des caméras pour éviter les phénomènes d'occultations. Aussi, la détection 2D peut devenir imprécise à cause de l'angle de perspective créé.

Au final, la configuration des caméras est un compromis où l'objectif est d'augmenter l'angle de la triangulation 3D tout en garantissant la stéréo-visibilité des points 3D analysés et leur bonne détection dans les deux images.

Notre proposition d'améliorer la précision 3D par l'utilisation d'une calibration spatiale s'avère être primordiale pour s'abstenir de la problématique liée à l'effet non-linéaire créé par la distorsion des images et pour s'abstenir des variations de l'angle de triangulation 3D dans le volume étudié.

Des résultats expérimentaux sur des applications industrielles seront présentés par la suite montrant des cas d'utilisation de la méthode par reconstruction 3D directe.

Chapitre 4

Production de carte dense de disparité

Sommaire

4.1	Calibration stéréoscopique	55
4.2	Rectification des images	57
4.2.1	Expression algébrique de la problématique	58
4.2.2	Détermination des matrices de rectification	59
4.2.3	Évaluation de la rectification	64
4.3	Stéréo-corrélation	66
4.3.1	État de l'art	66
4.3.2	Appariement par stéréo-corrélation	66
4.3.3	Optimisation algorithmique	68
4.3.4	Amélioration qualitative	76
4.3.5	Implémentation	79
4.4	Experimentation	85
4.4.1	Projection de mouchetis	85
4.4.2	Correction de l'effet de dilatation	85

Introduction

La production de carte de disparité consiste à définir une information tridimensionnelle de l'ensemble des points visibles dans l'espace étudié à partir de leur projection dans les deux images stéréoscopiques. Pour un point de l'espace, on définit cette information par sa position pixellique (u, v) dans l'image dite de référence, gauche ou droite, et la disparité entre les deux points images. La disparité est la distance en abscisse entre deux points correspondants et est inversement proportionnelle à l'éloignement du point de l'espace par rapport au système stéréoscopique.

Dans la littérature, on distingue deux approches pour définir l'information de l'image pour laquelle nous mesurons la disparité :

- soit l'ensemble des pixels de l'image qui aboutit à produire une carte dense de disparité
- soit des primitives de l'image qui aboutit à produire une carte éparse de disparité

La première approche consiste, pour chaque point de la première image, à trouver son correspondant dans la seconde image. La seconde nécessite de réaliser un pré-traitement des images pour identifier les primitives qui se distinguent soit en :

- des points d'intérêt dont nous retrouverons les plus populaires dans [HS88] [ST94] [SB95] [Low99] [BTG06]
- des contours, notamment avec [Can86] [SC92]

Nous citerons également l'utilisation de l'opérateur de déclivité dans [Kra08].

Dans notre problématique, nous nous intéressons à la production de carte dense de disparité.

Dans un premier temps, nous montrerons de manière détaillée les éléments nécessaires à la calibration de notre système de caméras. Puis nous présenterons une méthode pour construire les matrices de rectification qui sont déduites de la calibration inter-caméra.

Dans un second temps, nous développerons notre méthode pour construire nos cartes denses de disparité qui est basée sur la mise en correspondance par mesure de corrélation entre les pixels des deux images. Nous apportons des éléments pour accélérer le processus et pour améliorer la qualité de nos images résultats. Notre méthode fut l'objet d'un article [PKB11] montrant notre technique de mise en correspondance à partir de deux fenêtres de corrélation rectangulaires.

4.1 Calibration stéréoscopique

La géométrie épipolaire est la conséquence de l'expression de la géométrie projective appliquée à un système stéréoscopique. Elle exprime la relation en translation et en rotation entre les deux caméras.

La figure 4.1 exprime la géométrie épipolaire. On définit C_1 et C_2 les centres optiques des deux caméras. Soit un point P dans l'espace, on construit le plan épipolaire comme le plan passant par ces les points C_1 , C_2 et P . En conséquence, on définit les points p_1 et p_2 comme les projections du point P respectivement dans les plans image R_1 et R_2 , ainsi que les épiholes e_1 et e_2 comme les projections des centres optiques C_2 et C_1 respectivement dans les plans image R_1 et R_2 .

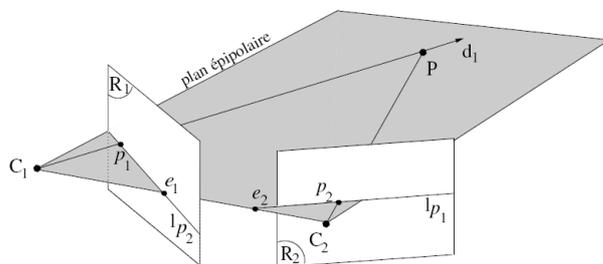


FIGURE 4.1 – Construction du plan épipolaire

Les propriétés de la géométrie projective montre la relation entre les deux points image p_1 et p_2 . On en déduit cette relation par une matrice homogène 3 x 3 appelée matrice fondamentale [LH81] telle que :

$$p_2^T F p_1 = 0 \quad (4.1)$$

La calibration du système stéréoscopique passe par l'estimation de la matrice fondamentale. Soit les points image $p_1 = (u_1, v_1, 1)$ et $p_2 = (u_2, v_2, 1)$ qui correspondent à la projection du même point P de l'espace, nous obtenons l'équation suivante :

$$u_1 u_2 f_{11} + u_1 v_2 f_{21} + u_1 f_{31} + v_1 u_2 f_{12} + v_1 v_2 f_{22} + v_1 f_{32} + u_2 f_{13} + v_2 f_{23} + f_{33} = 0 \quad (4.2)$$

La matrice fondamentale étant homogène, c'est à dire qu'elle est valable à un facteur non nul près, nous fixons $f_{33} = 1$. Au final, l'estimation de la matrice fondamentale nécessite au moins huit correspondances de points image.

En pratique, le système est résolu en sens des moindres carrés à partir d'un ensemble plus important de paires de correspondants. Il existe néanmoins des méthodes non linéaires [LF96] ou utilisant des estimateurs robustes afin de prendre en compte des éventuels

erreurs d'appariement des points image. Torr montre une étude sur les performances des estimateurs [TM97]. Pour estimer la matrice fondamentale, les plus utilisés sont le LMedS [Zha98] ou le RANSAC [FB81]). Les travaux [AS03] montrent les bonnes performances du LMedS et [Ste99] montrent les limites du RANSAC s'il y a de nombreuses erreurs d'appariement. Les travaux [MS04] montrent que jusqu'à 90% d'*outliers*, l'estimation de la matrice fondamentale reste valide.

Nous noterons que de nombreux travaux s'intéressent à la problématique d'auto-calibrage qui consiste à obtenir des points correspondants directement dans le contenu des images. Les descripteurs locaux, dont le plus connu est le SIFT [Low99], permettent de sélectionner des points particuliers dans les deux images et les estimateurs robustes permettent de réaliser l'appariement des points. Nous citerons les travaux [JFB11] qui utilisent le descripteur SURF [BTG06] dans une problématique « temps réel » montrant également les bonnes performances sur différentes configurations d'échelle, de luminosité, de déplacement.

[SK07] propose un comparatif sur un ensemble de méthode de résolution de la matrice fondamentale. Nous citerons notamment les travaux récents [ZPF06] qui utilisent une méthode linéaire et [BI12] qui proposent une méthode non-linéaire garantissant une bonne rectification des images.

Dans notre contexte où la calibration est réalisée en amont, l'utilisation de points de mires permet d'obtenir un appariement direct.

4.2 Rectification des images

La calibration d'un système stéréoscopique est essentielle car elle permet, à partir d'une liste de points correspondants, de déterminer la relation géométrique pour l'ensemble des points des deux images. Néanmoins, nous aborderons ici la rectification des images avec l'objectif d'aligner les points correspondants sur la même ligne dans les deux images. Cette technique coïncide avec le fait de vouloir accélérer le processus d'appariement de point et de calcul des disparités.

La rectification a commencée à être abordée dans la littérature à la fin des années 80. Mais c'est à la fin des années 90 que la méthode proposée dans [HAH99] devient la méthode de base de la rectification planaire. Il propose de séparer le calcul des deux matrices de rectification dont la première est calculée explicitement pour projeter le premier épipôle à l'infini et la seconde est déduite de la géométrie épipolaire en minimisant la valeur de la disparité.

Néanmoins on distingue trois approches de la rectification :

- Les méthodes par reparamétrage des coordonnées pixelliques dans un autre espace
- Les méthodes par projection des lignes épipolaires
- Les méthodes planaires

Les méthodes par reparamétrage sont destinées à des applications particulières et ont l'inconvénient d'apporter des déformations des lignes des images. Nous citerons notamment les travaux [RMC97] qui réalise une rectification cylindrique et les travaux [PKV99] [Ora01] qui réalisent un reparamétrage en coordonnées polaires. Ces méthodes sont destinées à une configuration particulière du système d'acquisition, notamment une translation en avant.

Les méthodes « lignes » consistent à projeter, de façon alignée, les lignes épipolaires. Ces méthodes nécessitent un traitement plus important que les autres méthodes. Elles ont fait l'objet de nombreux travaux tels [PD96] [CWT04] mais nous ne pouvons connaître leur impact sur la déformation projective des images résultats.

Les méthodes planaires sont les plus utilisées et ont fait l'objet de nombreux travaux [GF97] [IT99] [LZ99], avec notamment les méthodes [HAH99] qui minimise la disparité et [MW05] qui minimise la déformation projective.

Nous citerons également des travaux récents tels [WY05] qui prend en compte les effets de distorsion, [KMPF10] qui adapte l'échelle des images avant de mettre en pratique une rectification non-linéaire et [GNLY10] qui utilise les méthodes d'« optimisation par essais particuliers » dans le but de sélectionner les meilleurs appariements de points pour estimer la rectification.

Dans notre problématique, nous avons utilisé l'approche [MW05] car elle permet de conserver au mieux la forme des objets de la scène et donc de faciliter l'étape d'apparie-

ment qui s'en suit. Nous présenterons la première partie de cette dernière méthode que nous avons utilisée puis nous aborderons rapidement le principe de la minimisation de la déformation projective bien que nous puissions s'en abstenir si nos caméras sont proches de l'alignement.

Il est important de différencier le calcul des matrices de rectification qui permettront d'aligner nos deux images et l'application de la rectification qui consiste à appliquer une homographie à chacune des deux images. Dans cette partie liée à la calibration de notre système, nous montrons une méthode pour estimer les deux matrices de rectification à partir de la matrice fondamentale.

Le calcul se décompose en deux étapes :

- le calcul de la première matrice de rectification engendrant la parallélisation des lignes épipolaires dans la première image.
- le calcul de la seconde matrice de rectification engendrant l'alignement des lignes épipolaires entre les deux images

4.2.1 Expression algébrique de la problématique

D'un point de vue géométrique, la rectification consiste à faire une projection de l'image dans un nouveau plan. Algébriquement, on applique une homographie, matrice 3 x 3, à l'ensemble des points de l'image.

L'objectif est de déterminer les deux homographies H_1 et H_2 telles que, pour toutes paires de correspondants $p_1 = (u_1, v_1, 1)$ et $p_2 = (u_2, v_2, 1)$, leur projeté $p'_1 = (u'_1, v'_1, 1)$ et $p'_2 = (u'_2, v'_2, 1)$ respectent la condition $v'_1 = v'_2 = v'$.

La seconde condition est que les points p'_1 et p'_2 doivent respecter la contrainte de la géométrie épipolaire, qui se traduit par :

$$u'_1 u'_2 f_{11} + u'_1 v'_2 f_{21} + u'_1 f_{31} + v'_1 u_2 f_{12} + v'_1 v'_2 f_{22} + v'_1 f_{32} + u'_2 f_{13} + v'_2 f_{23} + f_{33} = 0 \quad (4.3)$$

ou encore :

$$u'_1 u'_2 f_{11} + v'(u'_1 f_{21} + u_2 f_{12} + f_{32} + f_{23}) + u'_1 f_{31} + v'^2 f_{22} + u'_2 f_{13} + f_{33} = 0 \quad (4.4)$$

Cette expression doit être vraie pour tout u'_1 , u'_2 et v' , elle est donc nulle si l'ensemble des termes qui la composent sont nuls également. Nous obtenons ainsi la contrainte $f_{32} + f_{23} = 0$. Par convention et comme la matrice fondamentale est vraie à un facteur d'échelle près, nous fixons $f_{32} = 1$ et $f_{23} = -1$.

La matrice fondamentale pour un système où l'ensemble des points correspondants sont sur la même ligne image se traduit par :

$$F_0 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \quad (4.5)$$

Nous obtenons finalement les deux expressions :

$$\begin{cases} p_2^T F p_1 = 0 \\ p_2'^T F_0 p_1' = 0 \end{cases} \quad (4.6)$$

Les points p_1' et p_2' étant les projetés des points p_1 et p_2 par les homographies H_1 et H_2 , nous pouvons écrire que $p_1' = H_1 p_1$ et $p_2' = H_2 p_2$. En conséquence, nous obtenons l'égalité :

$$(H_2 p_2)^T F_0 (H_1 p_1) = 0 \quad (4.7)$$

qui peut s'écrire :

$$p_2^T H_2^T F_0 H_1 p_1 = 0 \quad (4.8)$$

On en déduit la relation entre la matrice fondamentale F et les matrices de rectification H_1 et H_2 par :

$$F = H_2^T F_0 H_1 \quad (4.9)$$

4.2.2 Détermination des matrices de rectification

Pour calculer numériquement les paramètres des matrices de rectification, nous devons résoudre l'expression précédente car elle assure la contrainte d'alignement des points correspondants. Néanmoins, nous rappelons que l'objectif est d'amener ses points correspondants sur la même ligne image, qui se traduit par une parallélisation des lignes épipolaires.

Finalement, la détermination des matrices de rectification se décompose en deux étapes successives :

- Calcul de la première matrice de rectification H_1 entraînant la parallélisation des lignes épipolaires dans la première image.
- Calcul de la seconde matrice de rectification H_2 déduite de la relation 4.9.

4.2.2.1 Calcul de H_1

Cette première étape consiste à projeter l'épipole $e_1 = (e_u, e_v, 1)$ à l'infini tel que $H_1 e_1 = (e_u, 0, 0)$. Nous pouvons directement déduire l'expression matricielle suivante :

$$H_1 e_1 = \begin{bmatrix} 1 & 0 & 0 \\ -\frac{e_v}{e_u} & 1 & 0 \\ -\frac{1}{e_u} & 0 & 1 \end{bmatrix} \begin{bmatrix} e_u \\ e_v \\ 1 \end{bmatrix} = \begin{bmatrix} e_u \\ 0 \\ 0 \end{bmatrix} \quad (4.10)$$

4.2.2.2 Calcul de H_2

Cette seconde étape consiste à résoudre l'équation $F = H_2^T F_0 H_1$. Si nous développons le second terme, nous obtenons :

$$\begin{aligned} H_2^T F_0 H_1 &= \begin{bmatrix} h2_{11} & h2_{21} & h2_{31} \\ h2_{12} & h2_{22} & h2_{32} \\ h2_{13} & h2_{23} & h2_{33} \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ h1_{21} & 1 & 0 \\ h1_{31} & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} -h1_{31}h2_{21} + h1_{21}h2_{31} & h2_{31} & -h2_{21} \\ -h1_{31}h2_{22} + h1_{21}h2_{32} & h2_{32} & -h2_{22} \\ -h1_{31}h2_{23} + h1_{21}h2_{33} & h2_{33} & -h2_{23} \end{bmatrix} \end{aligned} \quad (4.11)$$

Les termes $h2_{11}$, $h2_{12}$ et $h2_{13}$ disparaissent de l'expression et sont donc fixés tels que la matrice H_2 s'écrit :

$$H_2 = \begin{bmatrix} 1 & 0 & 0 \\ h2_{12} & h2_{22} & h2_{32} \\ h2_{13} & h2_{23} & h2_{33} \end{bmatrix} \quad (4.12)$$

On en déduit le système à 9 équations suivant afin de résoudre les termes de la matrice H_2 à partir de la relation avec la matrice fondamentale à un facteur α près :

$$\left\{ \begin{array}{l} -h1_{31}h2_{21} + h1_{21}h2_{31} = \alpha f_{11} \\ \quad \quad \quad h2_{31} = \alpha f_{12} \\ \quad \quad \quad -h2_{21} = \alpha f_{13} \\ -h1_{31}h2_{22} + h1_{21}h2_{32} = \alpha f_{21} \\ \quad \quad \quad h2_{32} = \alpha f_{22} \\ \quad \quad \quad -h2_{22} = \alpha f_{23} \\ -h1_{31}h2_{23} + h1_{21}h2_{33} = \alpha f_{31} \\ \quad \quad \quad h2_{33} = \alpha f_{32} \\ \quad \quad \quad -h2_{23} = \alpha f_{33} \end{array} \right. \quad (4.13)$$

Le système est surdéterminé et nous pourrions déduire directement les termes de la matrice H_2 . Il est néanmoins plus judicieux de résoudre le système au sens des moindres carrés, réduisant par la même occasion, les éventuelles erreurs dans le calcul des termes de la matrice fondamentale.

Ainsi, nous devons résoudre la matrice H_2 ainsi que le facteur α . Nous nous permettons de fixer $h_{233} = 1$ car une homographie est valable à un facteur d'échelle près. On remarque que, en fixant h_{233} , le terme h_{211} fixé également à 1 initialement, devrait être adapté pour obtenir la bonne homographie. Néanmoins, la première ligne de la matrice H_2 n'intervient que sur la composante x des points images et, comme notre problématique est de rectifier nos images pour aligner les x, quelque soit la valeur de h_{211} , la contrainte d'alignement est respectée.

On en déduit les valeurs de H_2 par résolution du système matriciel suivant :

$$\begin{bmatrix} -h_{111} & 0 & 0 & h_{121} & 0 & -f_{11} \\ 0 & -h_{131} & 0 & 0 & h_{121} & -f_{21} \\ 0 & 0 & -h_{131} & 0 & 0 & -f_{31} \\ 0 & 0 & 0 & 1 & 0 & -f_{12} \\ 0 & 0 & 0 & 0 & 1 & -f_{22} \\ 0 & 0 & 0 & 0 & 0 & -f_{32} \\ -1 & 0 & 0 & 0 & 0 & -f_{13} \\ 0 & -1 & 0 & 0 & 0 & -f_{23} \\ 0 & 0 & -1 & 0 & 0 & -f_{33} \end{bmatrix} \begin{bmatrix} h_{221} \\ h_{222} \\ h_{223} \\ h_{231} \\ h_{232} \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ h_{121} \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (4.14)$$

4.2.2.3 Minimisation de la déformation projective

Lors de l'étape du calcul des matrices de rectification H_1 et H_2 , nous avons fixé les valeurs de la première ligne parce qu'elle n'intervenaient pas et donc n'avaient aucune conséquence sur le respect de la contrainte épipolaire. Mallon propose ainsi une solution qui utilise la première ligne des matrices de rectification pour minimiser la déformation des images rectifiées, indépendamment l'une de l'autre. Il propose ainsi d'appliquer une transformation telle que les nouvelles matrices de rectification R_1 et R_2 s'écrivent :

$$R_1 = R'_1 H_1 = \begin{bmatrix} a_1 & a_2 & a_3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} H_1 \quad \text{et} \quad R_2 = R'_2 H_2 = \begin{bmatrix} b_1 & b_2 & b_3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} H_2 \quad (4.15)$$

Ces transformations entraîne la création d'un effet « Shearing » sur les images comme l'exprime la figure 4.2.

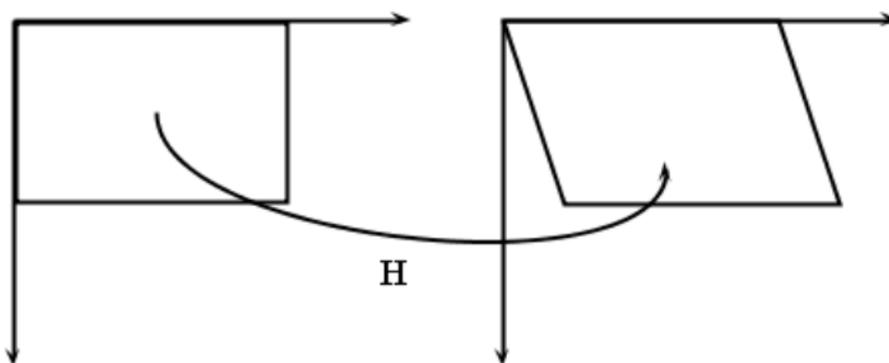


FIGURE 4.2 – Conséquence du « Shearing »

Les paramètres a_3 et b_3 n'agissant pas sur la déformation mais uniquement en translation, Mallon en déduit que les valeurs a_1 , a_2 et b_1 , b_2 sont le résultat de la minimisation des fonctions suivantes :

$$\begin{aligned} f(a_1, a_2) &= \sum_{i=1}^n (\sigma_1 - 1)^2 + (\sigma_2 - 1)^2 \\ f(b_1, b_2) &= \sum_{i=1}^{n'} (\sigma'_1 - 1)^2 + (\sigma'_2 - 1)^2 \end{aligned} \quad (4.16)$$

n et n' étant l'ensemble des points utilisés pour réaliser la minimisation, (σ_1, σ_2) et (σ'_1, σ'_2) sont respectivement les valeurs singulières du jacobien des matrices R_1 et R_2 .

Nous avons fait le choix de ne pas utiliser la minimisation de déformation projective car elle a peu d'influence si nos caméras sont très proche de l'alignement et que donc la déformation de nos images est déjà faible.

4.2.2.4 Recalage des matrices de rectification

La rectification des images par les matrices H_1 et H_2 garantit l'alignement des lignes images mais il peut se produire deux phénomènes, que l'on retrouve dans la figure 4.3 :

- Les points image résultats peuvent être de coordonnées négatives.
- La taille peut être différente entre les deux images rectifiées (en vert).

Le premier phénomène se résout par une translation des images rectifiées et ainsi les matrices de rectification deviennent :

$$H'_1 = \begin{bmatrix} 1 & 0 & t_{1x} \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} H_1 \quad \text{et} \quad H'_2 = \begin{bmatrix} 1 & 0 & t_{2x} \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} H_2 \quad (4.17)$$

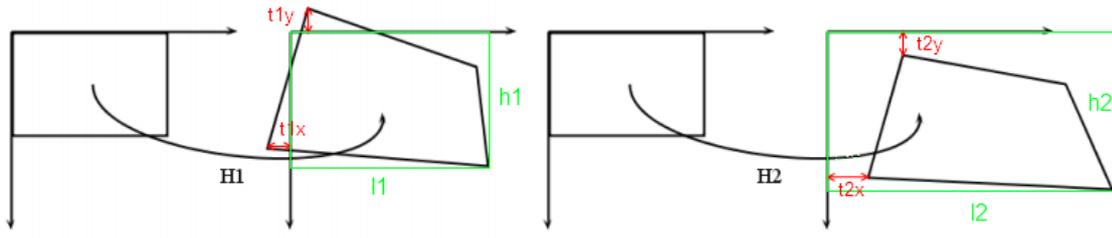


FIGURE 4.3 – Conséquences de la rectification : l'image rectifiée de gauche a des coordonnées négatives, la taille des deux images rectifiées est différente

La contrainte d'alignement nous oblige à appliquer la même translation t_y qui sera la valeur maximale entre t_{1y} et t_{2y} . Pour définir les valeurs de t_{1x} , t_{2x} , t_{1y} et t_{2y} , on applique les homographies H_1 et H_2 aux quatre coins des deux images de taille respective (l_1, h_1) et (l_2, h_2) tels que :

$$\begin{aligned}
 P_i^{hg} = P_i(0, 0) = H_i \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad P_i^{hd} = P_i(l_i, 0) = H_i \begin{bmatrix} l_i \\ 0 \\ 1 \end{bmatrix}, \\
 P_i^{bg} = P_i(0, h_i) = H_i \begin{bmatrix} 0 \\ h_i \\ 1 \end{bmatrix} \quad \text{et} \quad P_i^{bd} = P_i(l_i, h_i) = H_i \begin{bmatrix} l_i \\ h_i \\ 1 \end{bmatrix} \quad i=1,2
 \end{aligned} \tag{4.18}$$

Ainsi, nous obtenons :

$$\begin{cases} t_{ix} = -\min\left(\frac{P_i^{hg}(u)}{P_i^{hg}(w)}, \frac{P_i^{hd}(u)}{P_i^{hd}(w)}, \frac{P_i^{bg}(u)}{P_i^{bg}(w)}, \frac{P_i^{bd}(u)}{P_i^{bd}(w)}\right) \\ t_{iy} = -\min\left(\frac{P_i^{hg}(v)}{P_i^{hg}(w)}, \frac{P_i^{hd}(v)}{P_i^{hd}(w)}, \frac{P_i^{bg}(v)}{P_i^{bg}(w)}, \frac{P_i^{bd}(v)}{P_i^{bd}(w)}\right)_{i=1,2} \end{cases} \tag{4.19}$$

et

$$t_y = \max(t_{1y}, t_{2y}) \tag{4.20}$$

Pour définir les valeurs de longueur l et hauteur h uniques pour les deux images rectifiées, on applique dorénavant les homographies H'_1 et H'_2 respectivement aux quatre coins des deux images de la même manière que pour le calcul de la translation.

Ainsi, nous obtenons les longueur l_1 et l_2 et les hauteur h_1 et h_2 telles que :

$$\begin{cases} l_i = (\max - \min) \left(\frac{P'_i{}^{hg}(u)}{P'_i{}^{hg}(w)}, \frac{P'_i{}^{hd}(u)}{P'_i{}^{hd}(w)}, \frac{P'_i{}^{bg}(u)}{P'_i{}^{bg}(w)}, \frac{P'_i{}^{bd}(u)}{P'_i{}^{bd}(w)} \right) \\ h_i = (\max - \min) \left(\frac{P'_i{}^{hg}(v)}{P'_i{}^{hg}(w)}, \frac{P'_i{}^{hd}(v)}{P'_i{}^{hd}(w)}, \frac{P'_i{}^{bg}(v)}{P'_i{}^{bg}(w)}, \frac{P'_i{}^{bd}(v)}{P'_i{}^{bd}(w)} \right)_{i=1,2} \end{cases} \quad (4.21)$$

et nous définissons l et h telles que :

$$\begin{cases} l = \max(l_1, l_2) \\ h = \max(h_1, h_2) \end{cases} \quad (4.22)$$

En pratique, la rectification n'est pas une homographie de l'image d'entrée vers l'image de sortie mais on applique une homographie inverse. Pour chaque point des deux images de sortie, on applique les homographies inverses, respectivement H_1^{-1} et H_2^{-1} . La conséquence directe est que chaque point des images rectifiées est défini, nous n'avons pas de "trou" qui serait dû à la conversion en valeur entière des points rectifiés.

4.2.3 Évaluation de la rectification

Nous présentons les résultats de la rectification, c'est à dire nous évaluons l'erreur d'alignement sur un ensemble représentatif de points correspondants dans les deux images rectifiées. La procédure a été de faire l'évaluation sur les paires de points similaires utilisées pour l'estimation de la matrice fondamentale, soit un ensemble de quelques 3000 paires de points issus de la procédure de calibration. Les conditions de test sont définis par :

- deux caméras de même résolution 768x576
- un écartement caméra de l'ordre de 20 centimètres
- une distance entre les caméras et l'origine du repère monde de l'ordre du mètre

La mesure d'erreur est la différence de coordonnées v entre un point gauche rectifié par la matrice de rectification gauche et son correspondant à droite rectifié par la matrice de rectification droite. Soit les points correspondants $P1_i$ et $P2_i$, soit H_1 et H_2 les matrices de rectification, l'erreur de rectification se traduit par :

$$E_i = \left(\frac{P1_i(u).h1_{21} + P1_i(v).h1_{22} + P1_i(w).h1_{23}}{P1_i(u).h1_{31} + P1_i(v).h1_{32} + P1_i(w).h1_{33}} - \frac{P2_i(u).h2_{21} + P2_i(v).h2_{22} + P2_i(w).h2_{23}}{P2_i(u).h2_{31} + P2_i(v).h2_{32} + P2_i(w).h2_{33}} \right)^2 \quad (4.23)$$

Les résultats suivants montrent l'influence de l'angle inter-caméra sur la précision de la rectification :

Plus le système de caméra est proche de l'alignement, plus on minimise l'erreur de rectification. Cela se vérifie avec le tableau 4.1. Néanmoins nous justifions l'erreur de

Angle	$erreur < 0.5px$	$0.5px \leq erreur < 1px$	$1px \leq erreur$	$erreur_{max}$
30°	1503 pts (52.0%)	587 pts (20.3%)	798 pts (27.6%)	$> 5px$
$\simeq 0^\circ$	2756 pts (88.6%)	333 pts (10.7%)	22 pts (0.7%)	$1.42px$

TABLE 4.1 – Influence de l’angle inter-caméra sur la précision de la rectification

rectification par l’éventuelle erreur dans l’estimation de la matrice fondamentale dont les causes seraient :

- l’imprécision 2D sur les points de mire
- la distorsion des images non prise en compte dans l’estimation linéaire de la matrice fondamentale

On peut observer la répartition des erreurs de rectification sur la figure 4.4 pour les deux angles évalués.

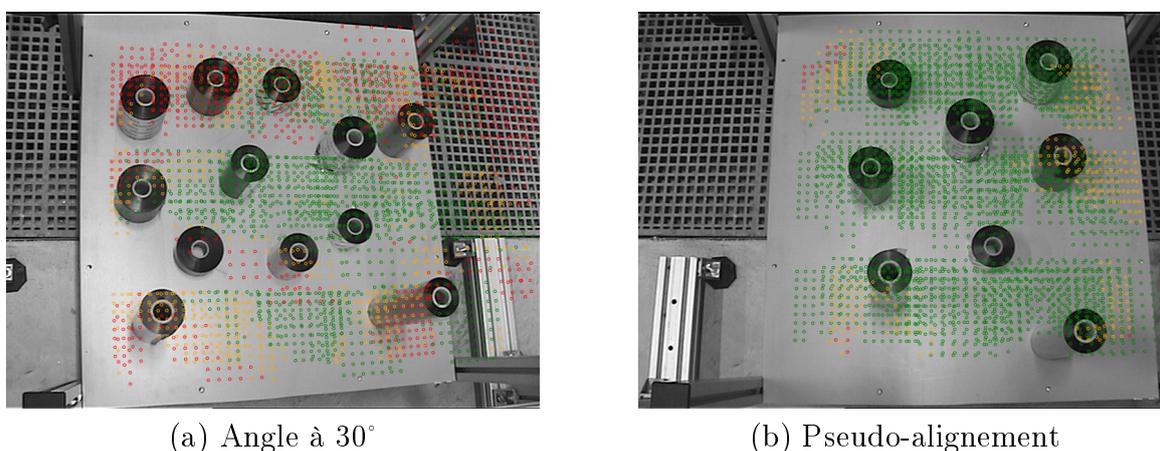


FIGURE 4.4 – Répartition de l’erreur de rectification - Les points verts représentent les erreurs inférieure au demi pixel, les points jaunes représentent les erreurs comprises entre $0.5px$ et $1px$ et les points rouges représentent les erreurs supérieures au pixel

Nous observons que l’erreur d’alignement des points image après rectification, est minimale au centre de l’image et augmente lorsque que l’on se déplace vers les pourtours des images. Cela se justifie que la rectification planaire ne prend pas en compte les effets de distorsion des images. Néanmoins, nous admettons que les caméras seront disposées de sorte que le champ analysé sera centré dans les deux images.

Nous pouvons également en conclure que, pour la production de carte dense de disparité, nous devons privilégier un système de caméra au plus proche de l’alignement qui aura pour conséquence de limiter les erreurs de rectification mais également de limiter la déformation projective des images.

4.3 Stéréo-corrélation

4.3.1 État de l'art

La mise en correspondance est réalisée par mesure de ressemblance entre deux pixels. Nous distinguons néanmoins différentes méthodes pour réaliser la mise en correspondance :

- Les méthodes locales
- Les méthodes globales
- Les méthodes mixtes
- Les méthodes à passages multiples

Les travaux [Cha05] fait une comparaison complète des méthodes de mise en correspondance basée sur [SS02] et [BBHM03].

Nous citerons des travaux récents mettant en œuvre les différentes approches de mise en correspondances avec [GIK03] [GM04a] [CJ04] pour les méthodes locales, [GM04b] [SLKS05] [GY05] [BG07] pour les méthodes globales, [AD04] [LALS04] [BHM04] pour les méthodes mixtes, [WQ04] [KLCL05] pour les méthodes à passages multiples.

Les méthodes locales et les méthodes globales sont les plus utilisées dans la littérature bien que les méthodes mixtes et à passages multiples se développent de plus en plus. Nous citerons notamment dans la seconde catégorie, les méthodes par coupure de graphe dites "*graph cuts*" [BVZ98] [IG98] [Roy99]. Les méthodes par coupure de graphe consistent à construire un graphe où les pixels sont les nœuds et les arcs sont les mesures de similarité entre les pixels. L'objectif est de construire un ensemble de sous-graphes correspondant à des régions distinctes de la scène. Ce sont des méthodes itératives qui, dans notre contexte industriel, ne sont pas exploitables car les temps de traitement sont de plusieurs dizaines de secondes pour des images de faible résolution et sur un intervalle de disparité réduit [KZ01].

C'est ainsi que nous avons privilégié les méthodes locales appelées généralement méthodes corrélatives qui sont très variées et qui font l'objet de nombreux travaux récents [SMM04] [LYX08] [HH09] [HZW⁺10] [LAC11].

4.3.2 Appariement par stéréo-corrélation

La mise en correspondance est l'étape essentielle à la production de carte dense de disparité. Elle consiste à déterminer les points qui se correspondent dans les deux images, c'est à dire faire correspondre les deux projections dans les images d'un même point de l'espace.

Nous nous plaçons dorénavant dans le contexte où nos images sont alignées et que l'ensemble des correspondants sont sur la même ligne dans les deux images. Pour construire

notre carte dense de disparité, nous devons, pour chaque point de l'image de gauche, trouver le point droit qui lui correspond. La disparité est déduite par la différence relative entre les deux points correspondants. L'intervalle de recherche de correspondance peut se réduire sur un intervalle limité dans l'image de droite à partir de la connaissance de la scène observée.

La stéréo-corrélation est un moyen de mesurer la ressemblance entre deux pixels. Elle consiste à mesurer la ressemblance entre les points issus des deux images par comparaison de leur voisinage respectif. Cette mesure est quantifiée par le calcul d'un score de corrélation.

4.3.2.1 Les indices de corrélation

Le choix du critère de corrélation est important pour obtenir le meilleur appariement. Il existe différents indices de corrélation dont la complexité est inversement proportionnelle au nombre de faux appariements. Le temps de traitement et la nature des images expliquent le choix d'un critère par rapport à un autre.

Le critère SSD (*Sum of Squared Differences*) est la somme des différences au carré de l'ensemble des pixels du noyau (fenêtre de corrélation de taille k avec $k = 2n + 1$, $n \in \mathbb{Z}$) respectivement dans l'image de gauche et dans l'image de droite :

$$SSD_{u,v}(d) = \sum_{i=-\frac{k-1}{2}}^{\frac{k-1}{2}} \sum_{j=-\frac{k-1}{2}}^{\frac{k-1}{2}} (I_g(u+i, v+j) - I_d(u+i+d, v+j))^2 \quad (4.24)$$

Une déclinaison du critère SSD est le critère SAD (*Sum of Absolute Differences*), qui est la somme des différences absolues de l'ensemble des pixels du noyau respectivement dans l'image de gauche et dans l'image de droite.

$$SAD_{u,v}(d) = \sum_{i=-\frac{k-1}{2}}^{\frac{k-1}{2}} \sum_{j=-\frac{k-1}{2}}^{\frac{k-1}{2}} |I_g(u+i, v+j) - I_d(u+i+d, v+j)| \quad (4.25)$$

Les fonctions x^2 et $|x|$ ayant la même monotonie sur \mathbb{R} , la contrainte de minimisation sur le critère SSD et SAD entrainera le même résultat. La différence réside dans les intervalles de valeurs des deux critères, soit $[0 \cdots 65025]$ pour le SSD et $[0 \cdots 255]$ pour le SAD.

Le critère ZSSD (*Zero-mean Sum of Squared Differences*) est la somme des différences au carré des écarts avec la moyenne de l'ensemble des pixels du noyau respectivement dans l'image de gauche et dans l'image de droite :

$$ZSSD_{u,v}(d) = \sum_{i=-\frac{k-1}{2}}^{\frac{k-1}{2}} \sum_{j=-\frac{k-1}{2}}^{\frac{k-1}{2}} ((I_g(u+i, v+j) - I_g(u, v)) - (I_d(u+i+d, v+j) - I_d(u+d, v)))^2 \quad (4.26)$$

Nous citerons le critère ZNSSD qui normalise le score de corrélation par les variations locales d'intensité. Nous citerons également les critères CC et ZNCC basés sur la corrélation croisée.

Les critères ZSSD, ZNSSD et ZNCC présentent les meilleurs résultats bien que le critère SSD ou SAD peuvent s'avérer suffisant dans un contexte industriel où on tend à maîtriser l'éclairage de la scène et ils ont l'avantage d'être plus rapides à calculer.

4.3.2.2 La validation croisée

L'étape de validation croisée permet de minimiser les erreurs de mise en correspondance et de supprimer les points erronés dans la carte disparité issus principalement des points occultés dans l'une des deux images. La procédure est de construire deux cartes de disparité avec comme image de référence, respectivement la première et la seconde image. La validation est la fusion des deux images de disparité par comparaison avec prise en compte du décalage. C'est à dire que pour chaque pixel de la première carte, on vérifie que la valeur de disparité d_1 est l'opposé de la disparité dans la seconde carte au même pixel décalé de d_1 . On vérifie donc que :

$$d_1(u, v) = -d_2(u + d_1(u, v), v) \quad (4.27)$$

4.3.3 Optimisation algorithmique

Nous proposons une technique d'optimisation du calcul des scores de corrélation.

Pour illustrer l'optimisation de la production des carte dense de disparité, nous nous intéressons au critère de corrélation SSD. Nous noterons dorénavant SD_i comme la différence au carré entre deux pixels ayant une disparité d_i :

Lors de la construction de carte dense de disparité, des opérations identiques sont réalisées de nombreuses fois. On dénombre que pour deux points fixés gauche et droite, leur SD intervient dans k^3 calculs de SSD. La figure 4.5 montre l'exemple pour un noyau de taille $k = 3$. Dans ce cas, deux points fixes dans l'image de gauche et l'image de droite interviennent dans 27 calculs de SSD.

Le principe de base de notre méthode est de construire une image SSD et une image de disparité qui sont mises à jour à chaque disparité d_i considérée. Ainsi, sur un intervalle

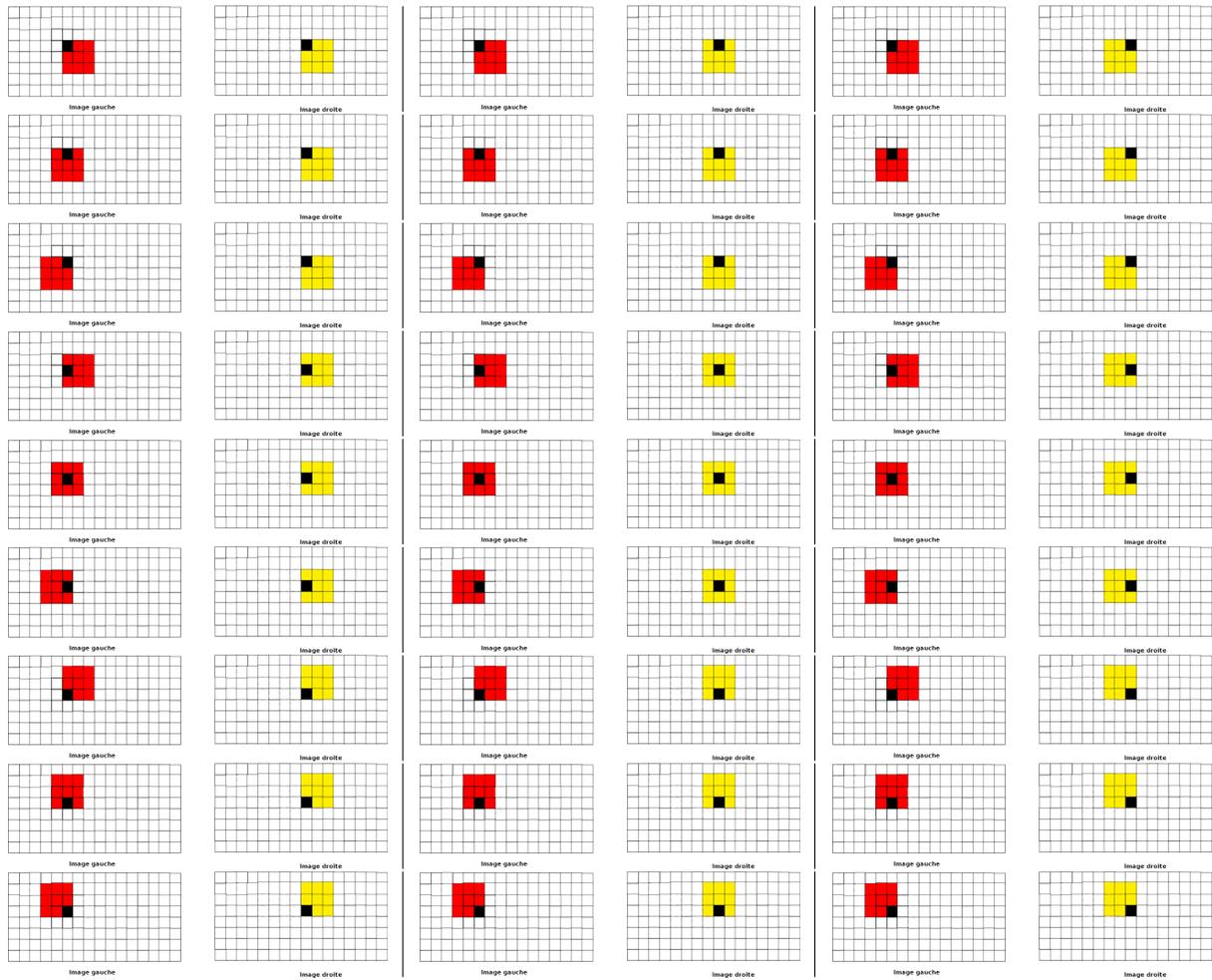


FIGURE 4.5 – Représentation des 27 comparaisons de fenêtre de corrélation de taille $k = 3$, dans lesquelles deux points fixes gauche et droite sont présents (les carrés rouges et jaunes correspondent respectivement à l'image de gauche et celle de droite)

de disparité $[d_{min} \cdots d_{max}]$, nous construisons une image SSD_i correspondant au résultat du calcul du SSD pour une disparité fixée d_i (voir fig. 4.6).

En résumé, pour chaque disparité d_i , nous construisons une image SSD_i à partir de l'image de gauche et de l'image de droite décalée de d_i , l'image SSD et l'image de disparité sont actualisées, respectivement par les valeurs d'erreur de l'image SSD_i et la valeur d_i , pour l'ensemble des points $p(x, y)$ si $SSD_i(x, y) < SSD(x, y)$.

Nous proposons une méthode qui se base sur deux principes : la séparabilité du calcul de SSD_i et la mémorisation du SD précédent pour calculer le SD suivant.

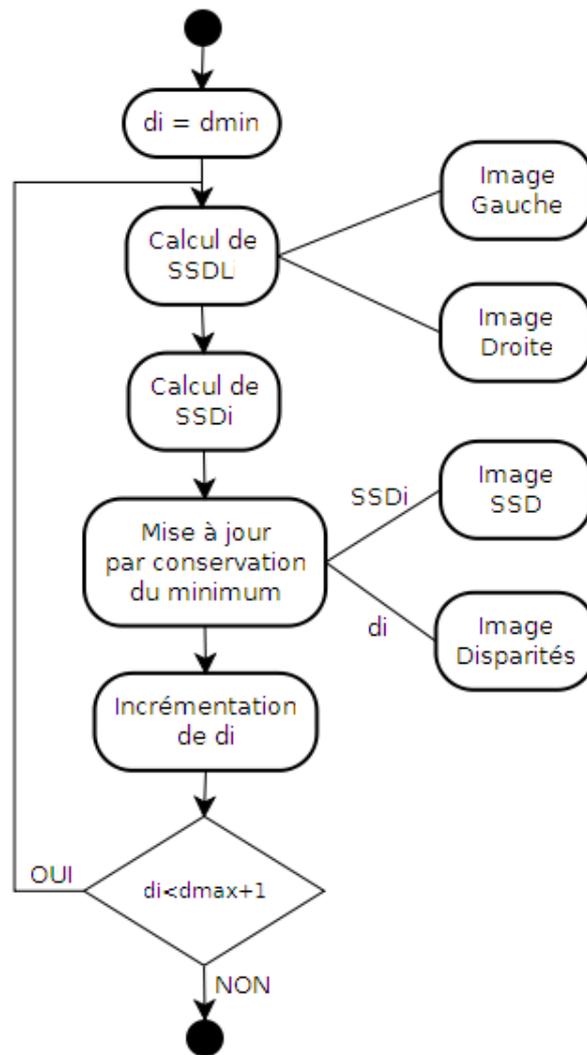


FIGURE 4.6 – Pseudo-algo du processus de production de carte dense de disparité

4.3.3.1 Séparabilité

Pour optimiser le traitement, notre méthode met en évidence la séparabilité du calcul d'une image de SSD_i . En effet, au lieu de calculer l'indice de corrélation sur toute la fenêtre, on décompose l'opération en une opération en ligne suivi d'une opération en colonne. La décomposition du calcul du SSD_i en deux opérations nous amène à construire une image temporaire $SSDL_i$, résultat de l'opération en ligne. Ainsi, nous exprimons $SSDL_i$ et SSD_i par :

$$SSDL_i(x, y) = \sum_{j=x-\frac{k-1}{2}}^{x+\frac{k-1}{2}} SD_i(j, y) \quad (4.28)$$

et

$$SSD_i(x, y) = \sum_{j=y-\frac{k-1}{2}}^{y+\frac{k-1}{2}} SSDL_i(x, j) \quad (4.29)$$

4.3.3.2 Mémorisation

Les méthodes corrélatives [DMM04] [MMHM02] [Vek03] mettent en œuvre la réutilisation des scores préalablement calculés sur tout le noyau. La séparabilité permet également de conserver le calcul intelligent des $SSDL_i$ avec mémorisation du calcul précédent afin de réduire le nombre d'opérations. Il intervient à la fois sur le calcul en ligne et sur le calcul en colonne. Notre méthode se différencie par la propagation du calcul de $SSDL_i$ d'une ligne à l'autre car nous admettons que les bords de l'image, de la largeur du noyau, soient entachés d'erreurs. Les images étant enregistrées dans un buffer mono-dimensionnel (les lignes sont les unes derrière les autres), la propagation horizontale en découle logiquement.

Pour le calcul en ligne (voir fig. 4.7), on calcule le $SSDL_i$ du premier pixel et les pixels suivants sont calculés suivant :

$$SSDL_i(x+1, y) = SSDL_i(x, y) - SD_i(x - \frac{k-1}{2}, y) + SD_i(x+1 + \frac{k-1}{2}, y) \quad (4.30)$$

L'illustration de l'algorithme pour le calcul de $SSDL_i$ reflète le calcul pour une disparité $d_i = 0$. Néanmoins, pour les autres disparités, le principe est le même mais on ajoute un décalage de la valeur de la disparité sur l'une des deux images en fonction du signe de la disparité.

Pour le calcul en colonne (voir fig. 4.8), on calcule le SSD_i pour la première ligne de l'image puis on propage le calcul des pixels suivants tel que :

$$SSD_i(x, y+1) = SSD_i(x, y) - SSDL_i(x, y - \frac{k-1}{2}) + SSDL_i(x, y+1 + \frac{k-1}{2}) \quad (4.31)$$

4.3.3.3 Mise à jour

Le calcul de SSD_i est l'image de corrélation pour la disparité d_i dont les valeurs sont comparées à l'image SSD . Pour les points de SSD_i dont la valeur est inférieure à ceux de SSD , nous faisons une mise à jour de la carte de disparité avec la valeur de disparité courante d_i et une mise à jour de l'image SSD (voir fig. 4.9).

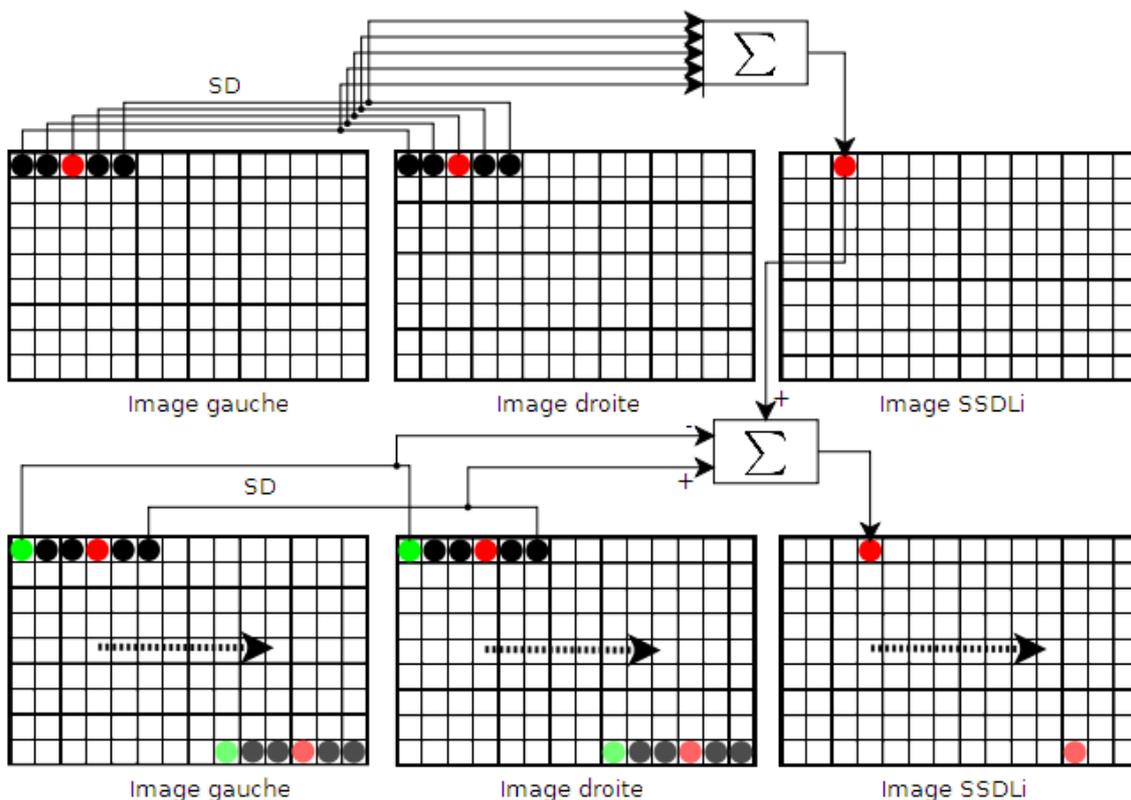


FIGURE 4.7 – Initialisation de la première valeur puis propagation horizontale pour le calcul en ligne (cas où $d_i = 0$)

4.3.3.4 Conséquences

L'apport de notre méthode réside dans notre quasi indépendance à la taille du noyau. En effet, la taille du noyau n'intervient qu'à l'initialisation de la première valeur lors de l'opération en ligne et à l'initialisation de la première ligne lors de l'opération en colonne. Au final, les temps de traitement pour la production de notre carte dense de disparité sont uniquement proportionnels à la taille de nos images et à l'intervalle de disparité considéré correspondant au nombre de SSD_i construites et de mises à jour de SSD .

Nous avons quantifié le nombre d'opérations en additions et en multiplications et l'avons comparé au cas où nous utiliserions directement le critère de corrélation par comparaison sur deux noyau carrés sans utiliser la séparabilité ni la mémorisation.

Nous posons :

- L et H, respectivement la largeur et hauteur des images
- k la taille du noyau carré
- l'opération d'addition \oplus
- l'opération de multiplication \otimes

Nous estimons le nombre d'opérations pour une disparité fixe et nulle. Nous admettons que, pour chaque valeur de disparité, le nombre d'opérations reste du même ordre. Au

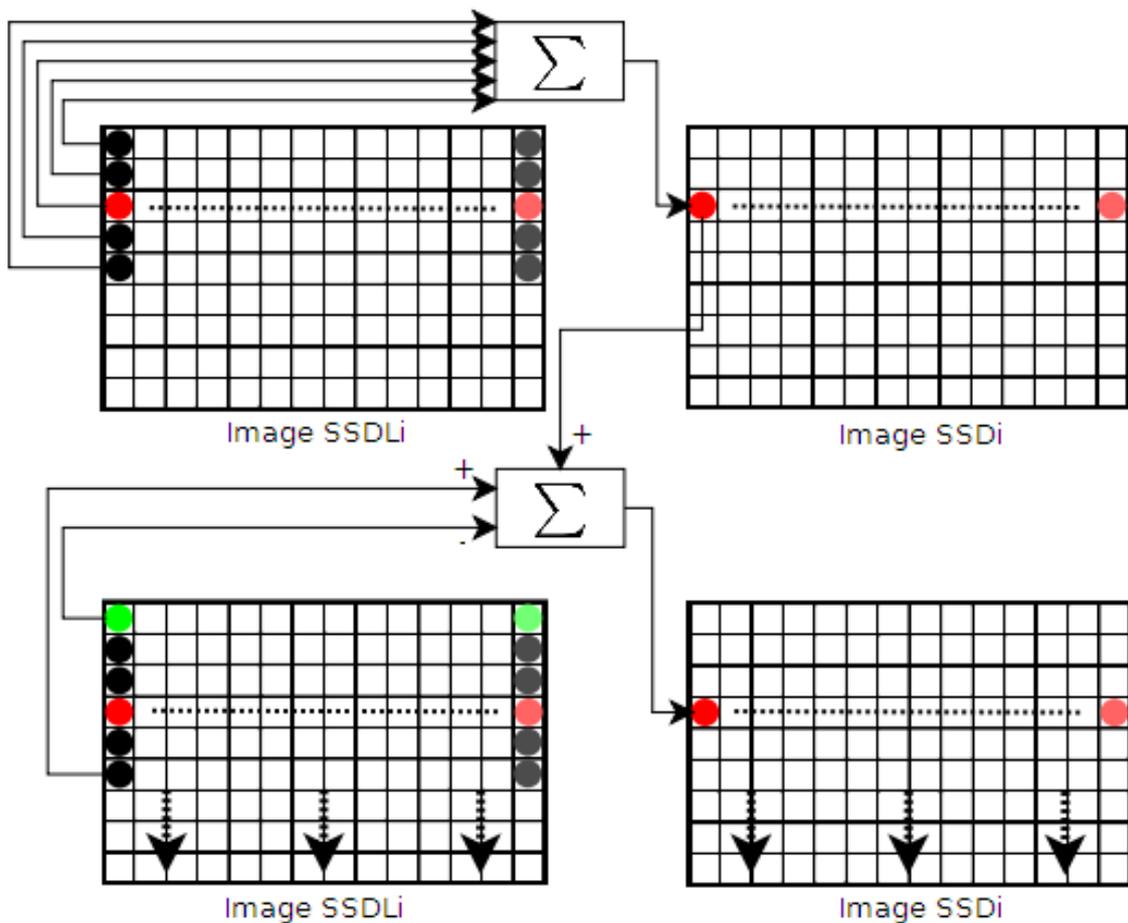


FIGURE 4.8 – Initialisation de la première ligne puis propagation verticale pour le calcul en colonne

final, nous nous intéressons à la construction d'une image SSD_i sans tenir compte de l'étape de mise à jour de la carte de disparité et de l'image d'erreur qui est commune dans les deux cas.

Sans optimisation : Pour une disparité nulle, nous calculons le score de corrélation sur une fenêtre de taille k pour l'ensemble des paires de points de mêmes coordonnées dans les deux images. Si nous faisons en sorte de garantir l'existence de la fenêtre de corrélation, nous dénombrons au final $(L - k)(H - k)$ calcul de score de corrélation.

Un score de corrélation SSD consiste à additionner les différences au carré de l'ensemble des pixels de la fenêtre de corrélation, soit $k^2 - 1$ additions de k^2 différences au carré. La différence est équivalente à une opération d'addition et le carré représente une opération de multiplication.

Au final, la construction d'une image SSD_i nous amène à réaliser le nombre d'opérations suivant :

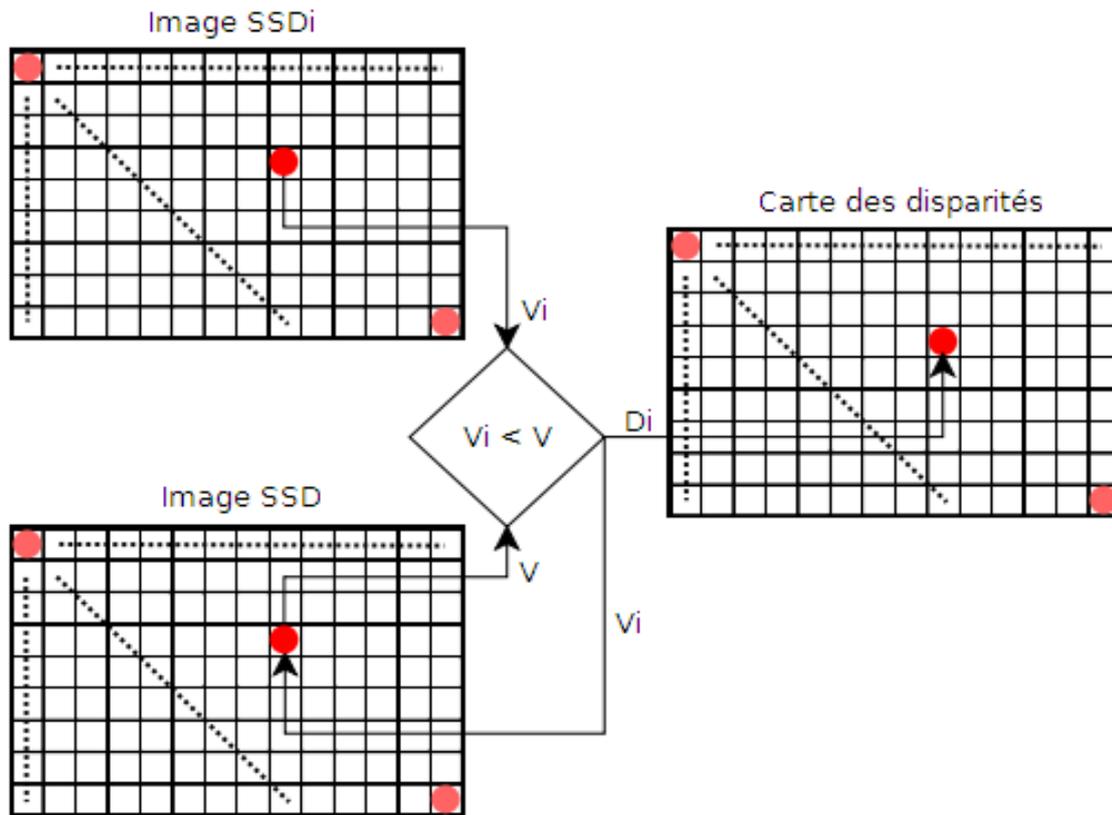


FIGURE 4.9 – Mise à jour de l'image SSD et de la carte dense de disparité

$$N = (L - k)(H - k)((k^2 - 1) \oplus + (k^2)(\oplus + \otimes)) \quad (4.32)$$

Avec optimisation : Nous décomposons le nombre d'opérations suivant les quatre différentes étapes suivantes :

- Calcul horizontal
 - initialisation de la première valeur
 - propagation horizontale
- Calcul vertical
 - initialisation des premières valeurs
 - propagation verticale

L'initialisation de la première valeur en horizontal consiste à calculer un score de corrélation sur un vecteur de taille k , soit $(k-1)$ additions de k différences au carré, soit $(k - 1) \oplus + k(\oplus + \otimes)$ opérations.

La propagation horizontale consiste à calculer les scores de corrélation pour l'ensemble des pixels "calculables" par l'utilisation de la valeur précédente à laquelle on ôte une différence au carré et on en ajoute une autre. A chaque pixel, nous réalisons donc deux

additions intégrant deux différences au carré, soit $(LH - k - 1)(4\oplus + 2\otimes)$ opérations pour l'ensemble des pixels.

L'initialisation des premières valeur en vertical consiste à calculer l'ensemble des scores de corrélation d'une ligne image sur un vecteur de taille k . La séparabilité implique que les différences au carré sont déjà intégré dans le calcul du score et nous opérons dorénavant uniquement avec des additions simples. Donc, cette étape implique que, pour chaque pixel, nous réalisons $(k-1)$ additions, soit $L(k-1)\oplus$ opérations pour l'ensemble des pixels de la ligne.

La propagation horizontale consiste à calculer le score de corrélation pour l'ensemble des pixels par l'utilisation de la valeur précédente à laquelle on ôte une valeur et on ajoute une autre. Pour l'ensemble des pixels, nous réalisons $L(H - k - 1)(2\oplus)$ opérations.

Au final, la construction d'une image SSD_i nous amène à réaliser le nombre d'opérations suivant :

$$N_{opt} = (k-1)\oplus + k(\oplus + \otimes) + (LH - k - 1)(4\oplus + 2\otimes) + L(k-1)\oplus + L(H - k - 1)(2\oplus) \quad (4.33)$$

Évaluation Nous pouvons décomposer les additions et les multiplications pour les deux cas telles que :

Optimisation	nombre d'opération \oplus	nombre d'opération \otimes
Sans	$(L - k)(H - k)(2k^2 - 1)$	$(L - k)(H - k)k^2$
Avec	$6L(H - k - 3) - 2k - 5$	$2(LH - 1) - k$

TABLE 4.2 – Comparaison du nombre d'opérations d'additions et de multiplications pour le cas classique et le cas avec optimisation

Pour illustrer l'impact de l'optimisation algorithmique du calcul d'une image d'erreur SSD pour une disparité, nous présentons dans le tableau 4.3 les valeurs numériques du nombre d'additions et de multiplications dans le cas d'image de résolution $l = 800$ et $H = 600$ et un noyau de taille $k = 11$.

Milliers d'opérations	opération \oplus	opération \otimes
Sans optimisation	111997	56231
Avec optimisation	2813	960
Rapport	39.8	58.6

TABLE 4.3 – Quantification et confrontation du nombre d'opérations sur un exemple pour le cas classique et le cas avec optimisation

En conclusion, la méthode optimisée nous apporte un gain conséquent en temps de traitement et nous permet de perdre la dépendance à la taille du noyau.

4.3.4 Amélioration qualitative

Les cartes denses de disparité ainsi construites mettent en évidence plusieurs problématiques :

- nous avons des "sauts" de disparité, représentant le passage d'une valeur de disparité à une autre
- nous avons de nombreuses erreurs d'appariement en zone homogène,
- nous avons de nombreuses erreurs d'appariement sur les zones proches des contours, ce qui se traduit par un effet de dilatation (voir fig. 4.11(b) à fig. 4.11(d)).

4.3.4.1 Précision subpixelique

Lors de la production de carte dense de disparité, on définit un intervalle de recherche $[d_{min}..d_{max}]$ et donc notre méthode fournit une carte de $(d_{max}-d_{min}+1)$ valeurs différentes. Par exemple, pour un intervalle de disparité $[10..50]$, la résolution des cartes sera de 41 valeurs de niveaux de gris différents. Qualitativement, on a des discontinuités lors du passage d'une disparité d à une disparité $d + 1$ (ou $d - 1$).

En réalité, la disparité entre 2 points qui se correspondent, n'est pas une valeur entière. On met donc en place des méthodes d'estimation de la valeur réelle de la disparité.

Pour rappel, 2 points (x, y) et $(x + disparit, y)$ se correspondent s'ils minimisent un score de corrélation (SSD par exemple). L'idée est de conserver les valeurs du SSD entre (x, y) et $(x + disparit - 1, y)$ et entre (x, y) et $(x + disparit + 1, y)$. On définit une fonction qui passe par les 3 valeurs de SSD et on cherche la valeur de disparité flottante qui minimise ou maximise cette fonction.

Si nous reprenons l'algorithme utilisé pour la production de carte dense de disparité, cette opération nécessite soit de conserver l'ensemble des cartes SSD_i , soit d'utiliser trois cartes SSD_i mais avec une gestion plus complexe que nous développerons par la suite.

Afin d'améliorer la résolution de nos images de disparité, nous avons mis en place la fonction d'interpolation parabolique dont les résultats sont meilleurs que la fonction d'interpolation dite « du toit » [Dev97]. En effet, si deux points (x, y) et $(x + d_e, y)$ minimisent le SSD, on calcule d_f tel que :

$$d_f = d_e + 0.5 \frac{SSD(d_e + 1) - SSD(d_e - 1)}{2 \cdot SSD(d_e) - SSD(d_e + 1) - SSD(d_e - 1)} \quad (4.34)$$

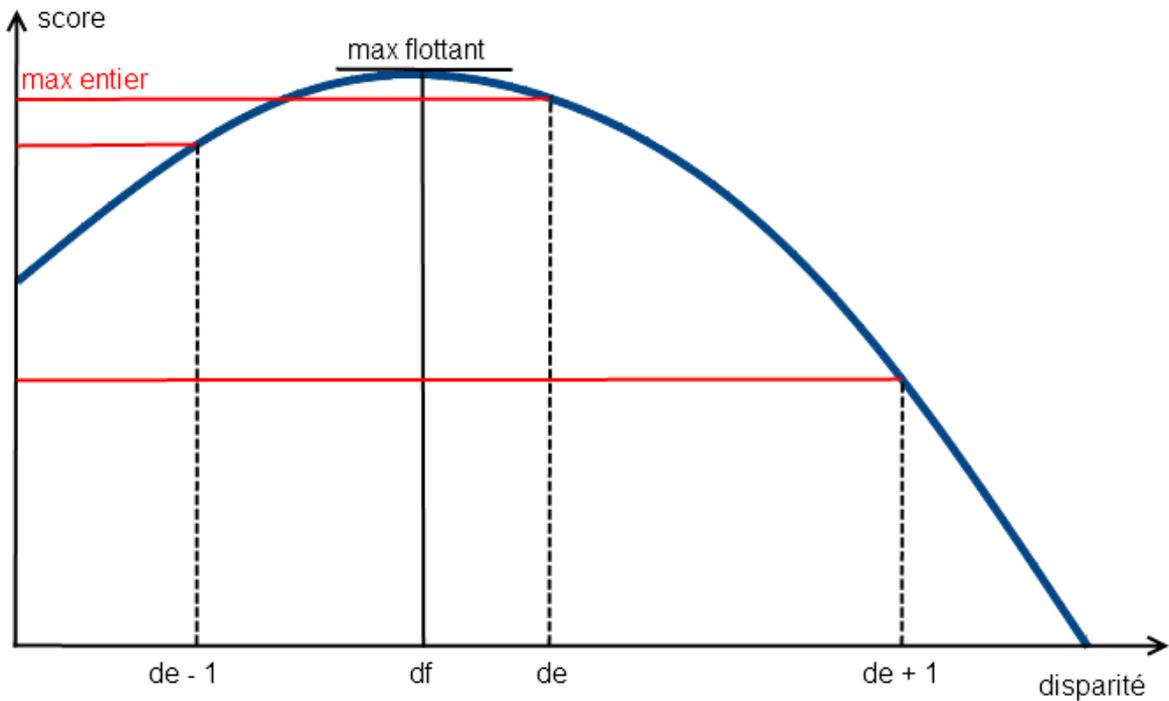


FIGURE 4.10 – Interpolation parabolique

4.3.4.2 Zones homogènes

Nous proposons d'utiliser la projection d'une image bruitée sur la scène afin de créer de la texture sur les éléments homogènes et cela de manière stéréoscopique. L'utilisation d'une lumière structurée est courante pour les applications où la scène ne présente pas suffisamment de texture pour satisfaire la stéréo-corrélation. Nous retrouvons notamment des travaux utilisant la projection de mouchetis [KWZK95] [RKS96] et plus récemment [HS06] [MJYY06]

4.3.4.3 L'effet de dilatation

L'effet de dilatation exprime l'erreur de calcul de la disparité près des zones de contours. En effet, plus le noyau est grand, plus l'effet de dilatation est important (voir fig. 4.11(b) à fig. 4.11(d)). Cet effet est particulièrement présent à proximité des contours avec une grande transition et présentant des zones homogènes. Les figures 4.12(a) et 4.12(b) montrent trois appariements de points pour trois noyaux différents. La problématique est que ces appariements engendreront la même valeur de disparité mais cette valeur sera affectée à un pixel différent, le point central du noyau considéré. Dans l'exemple, les points, plus ou moins éloignés de l'objet (rouleau) en fonction de la taille du noyau, auront une disparité proche de celle de l'objet lui-même alors qu'il devraient avoir la disparité du support.

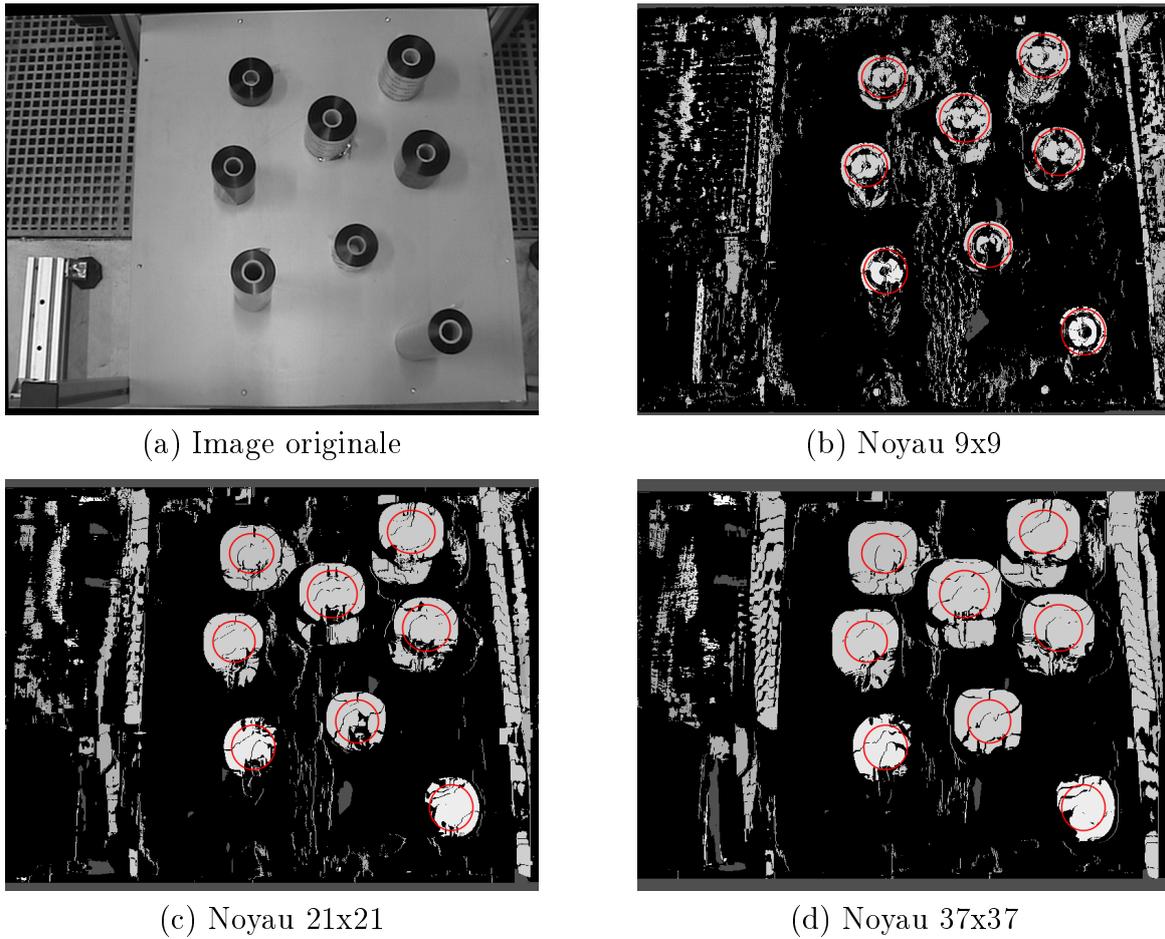


FIGURE 4.11 – Visualisation de l'effet de dilatation par rapport au contour supérieur des rouleaux en rouge

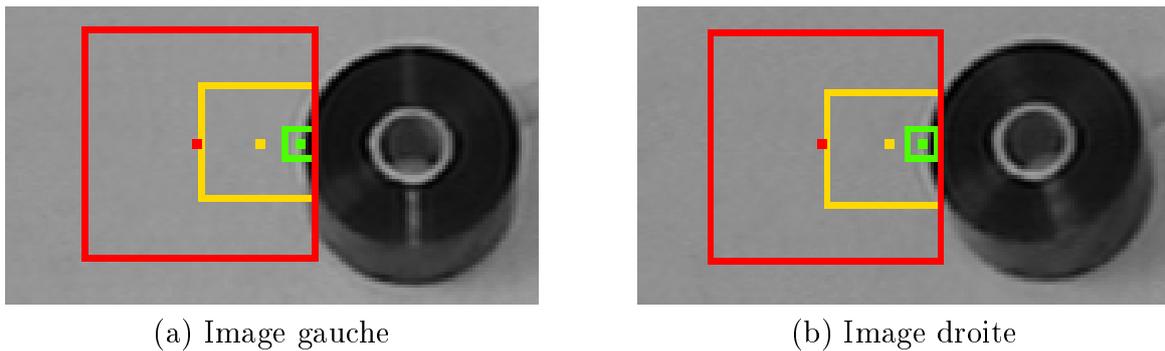


FIGURE 4.12 – Justification de l'effet de dilatation avec 3 noyaux de taille différente

Pour réduire l'effet de dilatation qui est proportionnel à la taille du noyau, nous proposons de reconsidérer l'utilisation d'une fenêtre englobante carrée. En effet, elle engendre un effet de dilatation dans toutes les directions. C'est ainsi que notre approche se base sur la construction de deux cartes de disparité, l'une utilisant un noyau "ligne" et l'autre

un noyau "colonne". Ces deux cartes sont produites de gauche vers la droite et nous ne mettons pas en œuvre le principe de validation croisée qui sera effectuée à l'étape suivante. L'objectif principal est que l'effet de dilatation sera uniquement horizontal pour le cas en ligne et vertical pour le cas en colonne.

La carte dense de disparité finale est une combinaison de ces deux images par comparaison des valeurs de disparité. Si la disparité dans les deux images est identique alors nous conservons cette disparité sinon on considère que nous n'avons pas de disparité. Au final, c'est la fusion des deux cartes qui réalise une validation croisée.

```

pour  $y = 0$  à hauteur faire
┌
  pour  $x = 0$  à largeur faire
├
│   si  $ImgDisp_{ligne}(x, y) = ImgDisp_{colonne}(x, y)$  alors
│   │    $ImgDisp(x, y) = ImgDisp_{ligne}(x, y)$ 
│   sinon
│   │    $ImgDisp(x, y) = 0$ 
└

```

Nous proposons également de conserver une certaine tolérance sur l'effet de dilatation. En effet, l'appariement, uniquement sur un noyau mono-dimensionnel, engendre davantage d'erreur que sur un noyau bi-dimensionnel dans la mesure où sa taille est k fois inférieure. La tolérance représente la hauteur du noyau "colonne" et également la largeur du noyau "ligne". Finalement, si on accepte un effet de dilatation de T pixels alors on utilisera un noyau "rectangulaire" de taille (T, k) et un autre noyau "rectangulaire" de taille (k, T) . (k étant la taille du noyau original). La tolérance n'a que très peu d'influence sur le temps de traitement dû à notre quasi indépendance à la taille du noyau lors du calcul du SSD_i .

Au final, le choix de la tolérance est un compromis entre un bon appariement nécessitant une fenêtre de corrélation de taille suffisante et une minimisation de l'effet de dilatation.

4.3.5 Implémentation

4.3.5.1 Approche subpixellique

La production de carte dense de disparité de base nous fournit, pour l'ensemble des points de l'image, une valeur de disparité entière ainsi qu'un score de corrélation qui lui est associé. Néanmoins, l'approche subpixellique nécessite que nous conservions les scores de corrélation pour la disparité précédente et pour la disparité suivante.

Nous pouvons considérer deux approches pour mettre en application l'amélioration subpixellique :

- par mémorisation des images d'erreur SSD_i

- par gestion de trois images d’erreur

La première approche consiste à conserver l’ensemble des images SSD_i . En conséquence, l’image de disparité finale devra être analysée en totalité. C’est à dire que pour chaque pixel de l’image de disparité finale, on récupère la disparité d_i ainsi les valeurs d’erreur des images SSD_{i-1} , SSD_i et SSD_{i+1} à la même position que le pixel courant. Cette approche a l’avantage d’être mise en place plus facilement mais a la contrainte mémoire de conserver autant d’image qu’il y a de disparité à considérer.

La seconde approche consiste à n’utiliser que trois images d’erreur : l’image SSD globale et les deux images des erreurs pour les disparités inférieure et supérieure. La gestion est plus difficile mais le gain en mémoire est considérable. Nous avons choisi cette approche que nous allons expliciter.

Nous définissons tout d’abord les différentes images nécessaires à la réalisation de l’amélioration subpixelique dans le cadre de la seconde approche :

- les trois images d’erreur finales inférieure, globale et supérieur, respectivement SSD_{inf} , SSD et SSD_{sup}
- les trois images d’erreur courantes SSD_{i-1} , SSD_i et SSD_{i+1}
- l’image de disparité finale $Disp$
- l’image de masque courante $Mask$

Nous décomposons l’approche en trois étapes : l’initialisation, la production et la finalisation.

L’initialisation permet de gérer la première disparité. En effet, pour la première disparité d_0 , nous ne connaissons pas l’image d’erreur précédente car elle n’existe pas. Nous imposons que, pour une valeur de disparité égale à d_0 , la valeur de disparité subpixelique reste inchangée.

Dans un premier temps, nous calculons l’image d’erreur SSD_{d_0} dans l’image courante SSD_{i-1} , que nous recopions ensuite dans les images SSD_{inf} , SSD et SSD_{sup} . Puis nous considérons la disparité suivante d_1 et nous calculons l’image d’erreur SSD_{d_1} dans l’image courante SSD_i . Cette dernière nous permet de connaître l’erreur suivante par rapport à la disparité d_0 , nécessaire à la mise à jour des images d’erreur globales.

La seconde étape consiste à poursuivre sur les disparités suivantes. La problématique se trouve sur la gestion des trois images d’erreur courantes. En effet, à chaque incrémentation de la disparité, l’image SSD_i calculée pour d_0 devient l’image SSD_{i-1} lors du calcul pour la disparité d_1 . Pour cela, les images d’erreur courantes SSD_{i-1} , SSD_i et SSD_{i+1} fonctionnent à la manière dans buffer circulaire.

Aussi, comme la mise à jour des images d'erreur finales impose de connaître l'image d'erreur courante SSD_{i+1} , alors SSD_{inf} , SSD et SSD_{sup} sont actualisées par la disparité d_i après le calcul de l'erreur avec la disparité d_{i+1} .

Cette étape est donc réalisée jusqu'à la disparité $d_{max} - 1$.

Au final, pour chaque disparité d_i :

- Nous construisons l'image de masque courante $Mask$ qui est une image binaire qui activera les pixels dont l'erreur dans SSD_i est inférieur à l'erreur dans SSD .
- Nous actualisons simultanément l'image de disparité globale $Disp$ avec la valeur de disparité d_i .
- Nous calculons l'image courante SSD_{i+1} pour la disparité suivante $(d_i + 1)$.
- Nous actualisons les images globales SSD_{inf} , SSD et SSD_{sup} à partir du masque et des images courante SSD_{i-1} , SSD_i et SSD_{i+1} .
- Nous recommençons l'opération avec la disparité suivante $(d_i + 1)$ en décalant les images SSD_{i-1} , SSD et SSD_{i+1} . Ainsi l'image SSD d'avant devient l'image SSD_{i-1} , l'image SSD_{i+1} d'avant devient l'image SSD et enfin l'image SSD_{i-1} d'avant devient l'image SSD_{i+1} . Cette dernière sera modifiée à l'étape suivante.

La dernière étape de finalisation consiste à gérer la dernière valeur de disparité d_{max} . A la manière de l'étape précédente, nous construisons l'image de masque $Mask$, nous actualisons l'image de disparité $Disp$. Pour la dernière disparité, la disparité suivante n'existe pas et, de la même manière qu'à l'étape d'initialisation, nous admettons que la dernière disparité est inchangée. Ainsi, nous actualisons les images globales SSD_{inf} , SSD et SSD_{sup} , à partir du masque, avec la même image SSD_{sup} .

L'image $Disp$ contient les valeurs entières de disparité, les images SSD_{inf} , SSD et SSD_{sup} contiennent respectivement les scores de corrélation précédent, courant et suivant. Ainsi, à partir de l'équation 4.34, nous pouvons calculer la valeur de disparité flottante pour l'ensemble des pixels de l'image de disparité.

4.3.5.2 Optimisation multiprocesseur

Pour l'implémentation de notre algorithme de production de carte dense de disparité, nous construisons nos deux cartes de disparité, respectivement avec un noyau en ligne et un noyau en colonne, sur deux cœurs de processeur différents. Une série de mesures nous permet d'estimer un gain de 1,8 en temps de calcul sur l'ensemble du traitement.

4.3.5.3 Optimisation XMMX

Le MMX est un jeu d'instructions pour les microprocesseur de type x86. L'architecture du MMX est présentée dans la figure 4.13.

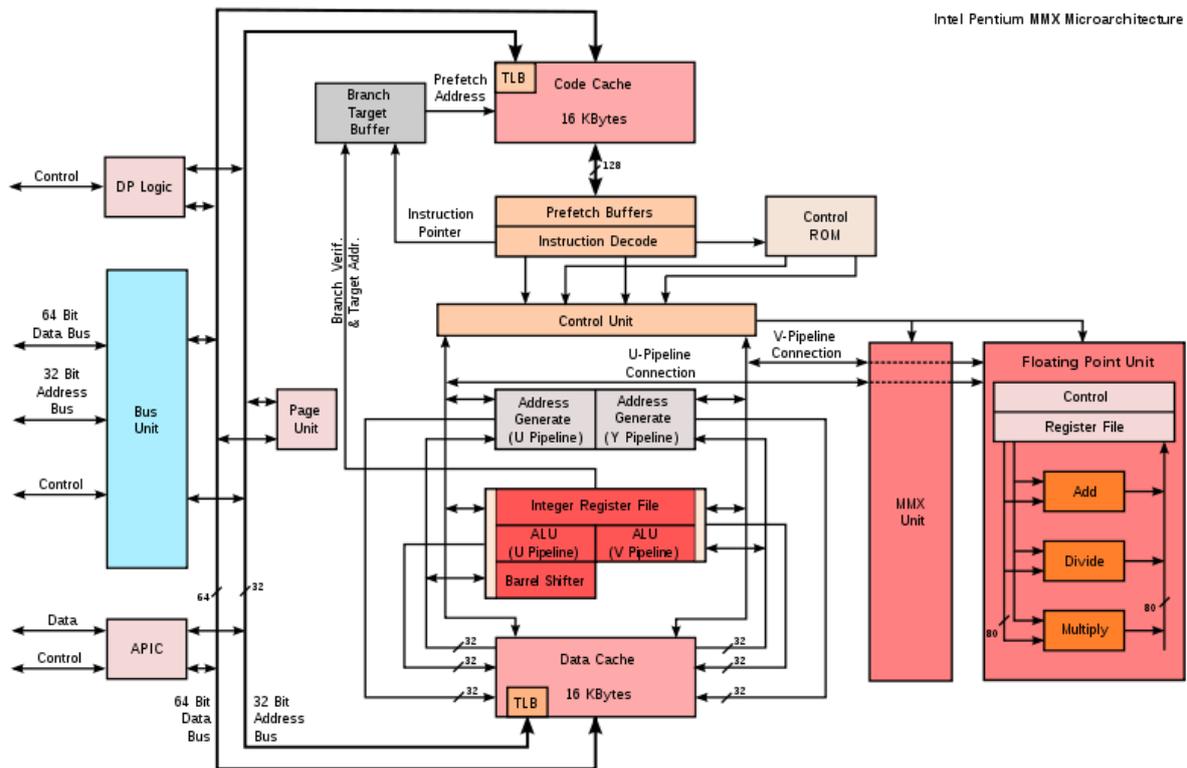


FIGURE 4.13 – Architecture interne du MMX

Cette architecture permet de manipuler plusieurs valeurs simultanément. Le terme de SIMD (Single Instruction Multiple Data) est généralement utilisé. Plusieurs extensions d'instructions existent, nommé SSE (Streaming SIMD Extensions), de la version 1 à la version 5 aujourd'hui. Ces instructions sont notamment utilisées dans le traitement du son et de l'image car cela permet d'accélérer considérablement les algorithmes sous la condition qu'ils soient parallélisables.

La version étendue, nommé XMMX, permet de réaliser une opération sur 16 octets de données en une seule instruction.

Dans nos travaux, nous utilisons le jeu d'instructions SSE en version 4 qui intègre les registres étendus XMMX.

Pour exemple, lors de la phase d'initialisation des trois images de disparité (à des valeurs nulles), le XMMX permet de traiter 16 octets de données en une seule instruction. Le pixel étant représenté sur un octet, on peut traiter les images de disparité par bloc de 16 pixels, soit un gain de 16 par rapport à une implémentation classique.

Nous utilisons ici l'optimisation XMMX à son maximum mais le gain est évidemment dépendant du type de données à traiter.

Nous montrons un exemple avec l'instruction « pcmpeqb » en figure 4.14 avec la fonction qui initialise l'image SSD. En effet, lors de la mise à jour de l'image globale *SSD*, nous cherchons à insérer les nouvelles valeurs issues de SSD_i qui minimisent les valeurs courantes de *SSD*. Il est donc nécessaire que les valeurs de *SSD* soient de valeurs maximales avant les différentes mises à jour.

```

template<class T> void ComputeInitializationMax(vector<T> & v)
{
    // pointeur sur la première valeur du vecteur
    T * ptrV = &v[0];
    // calcul du nombre d'opération :
    // count=(taille vecteur / 16) * (nb d'octet pour une donnée)
    const int count = (v.size()* sizeof(T)) >> 4 ;

    // nous mettons l'adresse du pointeur dans le registre eax
    __asm mov    eax,    ptrV
    // nous mettons le compteur d'opération dans le registre ecx
    __asm mov    ecx,    count
    /* instruction XMMX qui fait un traitement de comparaison octet
       par octet. Ici, on compare xmm0 avec lui-même. Au final xmm0
       contiendra que des bits à 1. Et donc chaque donnée de taille
       sizeof(T) sera à sa valeur maximale */
    __asm pcmpeqb xmm0,    xmm0

    // label
    __asm boucle:
    /* Nous mettons la valeur de xmm0 à l'adresse eax (pointeur
       sur les données)*/
    __asm movdqu [eax],    xmm0
    // Nous nous déplaçons sur la première valeur des 16 octets suivants
    __asm add    eax,    16
    // On décrémente le compteur d'opérations
    __asm dec    ecx
    /* Nous vérifions si le compteur est non nul, auquel cas on
       recommence */
    __asm jnz    boucle
}

```

FIGURE 4.14 – Code source de la fonction pour construire l'image SSD d'origine

La fonction « ComputeInitializationMax » est une fonction templâtée afin de montrer la dépendance aux données manipulées. Le nombre d'opération, dans cette exemple, est dépendant du compteur d'opération « count ». En effet, si nous voulons initialiser une image en niveau de gris, les données sont sur un octet et comme l'instruction « pcmpeqb » traite seize octets alors, en une seule opération, nous initialiserons seize pixels.

En opposition, l'initialisation de l'image SSD contient des valeurs plus importantes.

En effet, le score SSD est compris entre 0 et $k^2 255^2$. Nous pouvons montrer que, pour un noyau carré compris entre 3 et 257, les données de l'image SSD sont sur quatre octets. Ainsi, en une seule opération, nous pourrions initialiser $\frac{16}{4} = 4$ données.

Dans notre problématique, chaque étape de la construction des images SSD_i ne peut pas toujours être parallélisée par des instructions XMMX.

En effet, la phase de construction des images $SSDL_i$, résultat de la propagation horizontale, ne peut pas être parallélisée. Pour modifier une valeur de l'image $SSDL_i$ à la position (x,y) , nous avons besoin de connaître la valeur à la position $(x-1,y)$ dans la même image. Nous ne pouvons donc pas calculer deux valeurs successives lors d'une même opération.

Par contre, lors de la propagation verticale, nous pouvons calculer plusieurs valeurs successives d'une même ligne car cela ne nécessite que la connaissance préalable des valeurs de la ligne supérieure et qui ont déjà été calculées.

Enfin l'ensemble des opérations n'effectuant aucune opération entre des pixels d'une même image peut être traité par des opérations XMMX comme l'initialisation des images $Disp$ à des valeurs nulles et les images SSD_{inf} , SSD et SSD_{sup} à des valeurs maximales et aussi la mise à jour des images SSD_i , SSD_{inf} , SSD , SSD_{sup} et $Disp$.

4.4 Experimentation

4.4.1 Projection de mouchetis

Nous montrons dans un premier temps l'évolution entre la production de carte dense de disparité à partir de l'image originale (voir fig. 4.15(a)) et celle à partir de l'image issue de la scène sur laquelle nous avons projeté une image de bruit (voir fig. 4.15(b)). Les images rectifiées utilisées sont de taille 488 par 489 (238632 pixels) et l'intervalle de disparité est $[-40..30]$, soit un total de 70 valeurs de disparité à considérer.

Nous pouvons voir sur la figure 4.15(c) et 4.15(d), l'apport du mouchetis sur notre scène par rapport aux zones homogènes. On dénombre pour l'image originale 140863 points non appariés soit 59%, contre 105471 pour l'image mouchetée, soit 44,2%.

Les images sont le résultat de la production de carte dense de disparité à partir de notre méthode sans prise en compte de l'effet de dilatation. Afin de supprimer les erreurs d'appariement, nous avons utilisé le procédé de validation croisée.

4.4.2 Correction de l'effet de dilatation

Pour évaluer notre algorithme visant à limiter l'effet de dilatation, nous le confrontons aux résultats obtenus avec la librairie OpenCV¹. La fonction de production de carte de disparité intégrée dans OpenCV est basée sur l'indice de corrélation SAD (*Sum of Absolute Differences*). Cet indice est différent du nôtre mais, dans une problématique de minimisation, le SAD et le SSD ont des résultats équivalents.

La différence entre les deux méthodes réside dans le fait que nous n'avons pas l'effet de dilatation de la surface. Nous conservons donc la forme rectangulaire et nous n'avons pas d'estimation de la disparité sur les zones d'occultation. (voir fig. 4.16)

Nous avons quantifié le nombre de points qui représentent la forme rectangulaire afin d'évaluer le pourcentage de conservation des formes (voir fig. 4.17). Nous avons évalué nos performances selon le critère de tolérance et nous obtenons des écarts, avec le nombre de points issus de l'image de vérité terrain, inférieurs au demi pourcent (soit une cinquantaine de points) alors que la méthode de OpenCV dépasse les 2% (autour de 400 points d'écart).

Aussi, l'effet de dilatation augmente proportionnellement pour les cartes denses de disparité issues de OpenCV alors que notre algorithme est insensible à cette problématique. (voir fig. 4.18)

On peut noter l'influence de la tolérance (voir fig. 4.19). Plus la tolérance augmente, moins la carte de disparité est bruitée mais moins l'estimation des contours est précise.

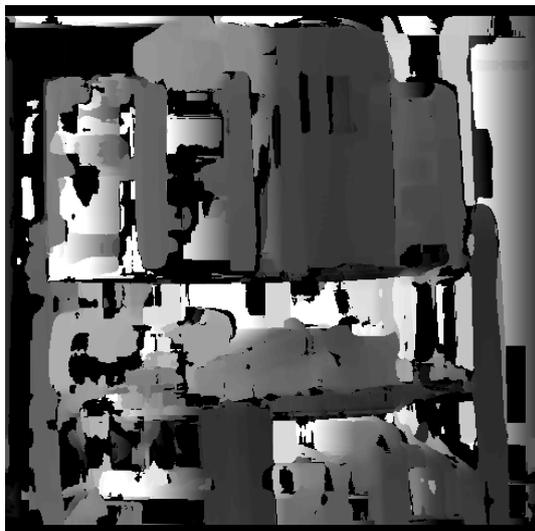
1. OpenCV est une librairie opensource de référence en traitement d'image et vision par ordinateur : <http://opencv.willowgarage.com>



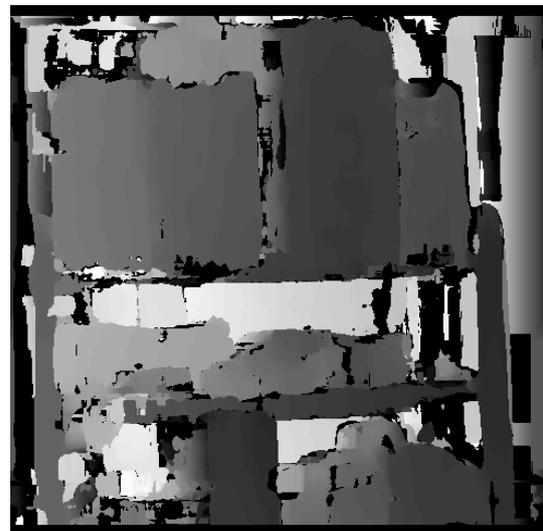
(a) Original



(b) Mouchetis



(c) Disparités (Original)

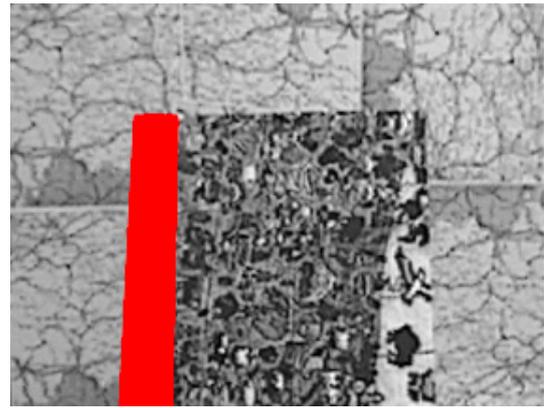


(d) Disparités (Mouchetis)

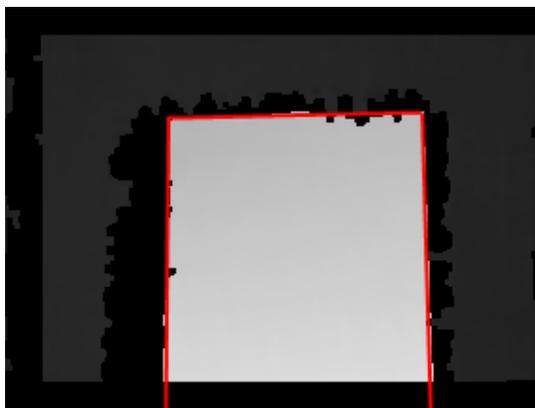
FIGURE 4.15 – Influence du mouchetis pour un noyau 21x21 (utilisation du SSD avec validation croisée)



(a) Vérité terrain



(b) Occultation

(c) Notre carte ($k = 31$)(d) OpenCV ($k = 31$)

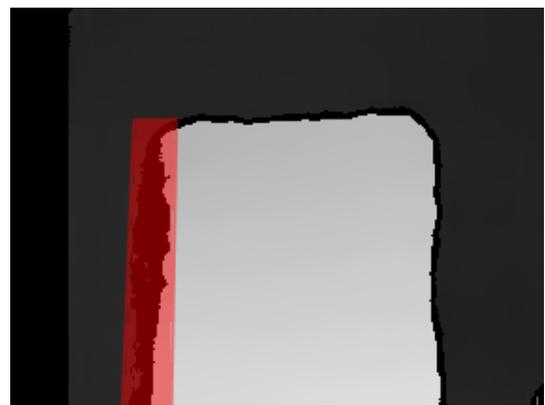
(e) Notre carte (contours)



(f) OpenCV (contours)



(e) Notre carte (occultation)



(f) OpenCV (occultation)

FIGURE 4.16 – Correction de l'effet dilatation pour différents noyaux et comparaison avec les résultats issus de OpenCV (<http://vision.middlebury.edu/stereo/data/>)

	Tolérance	Noyau k = 21		Noyau k = 31		Noyau k = 41	
Vérité terrain		18832 pts		17422 pts		16014 pts	
Fusion	3	18633 pts	1,06%	17248 pts	1,00%	15947 pts	0,42%
	5	18864 pts	0,17%	17494 pts	0,41%	16167 pts	0,96%
	7	19151 pts	1,69%	17725 pts	1,74%	16316 pts	1,89%
OpenCV		19370 pts	2,86%	17786 pts	2,09%	16337 pts	2,02%

FIGURE 4.17 – L'écart en pourcentage du nombre de points appartenant à la forme rectangulaire et comparaison avec les performances de OpenCV

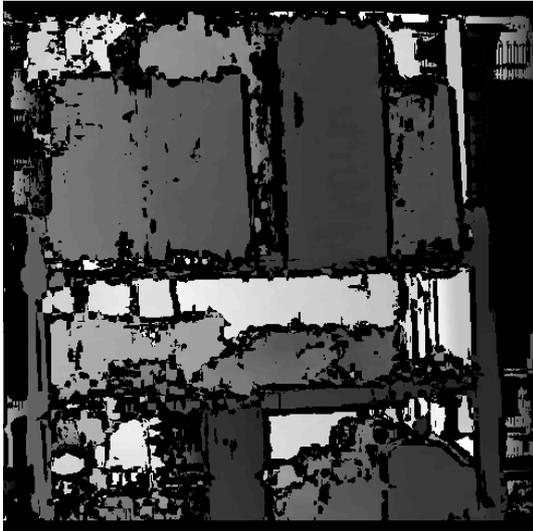
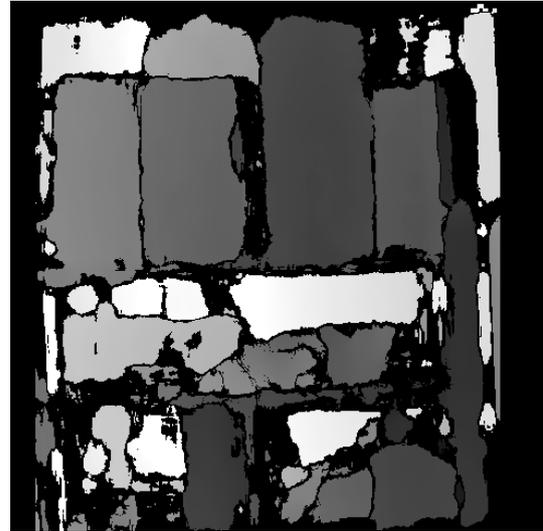
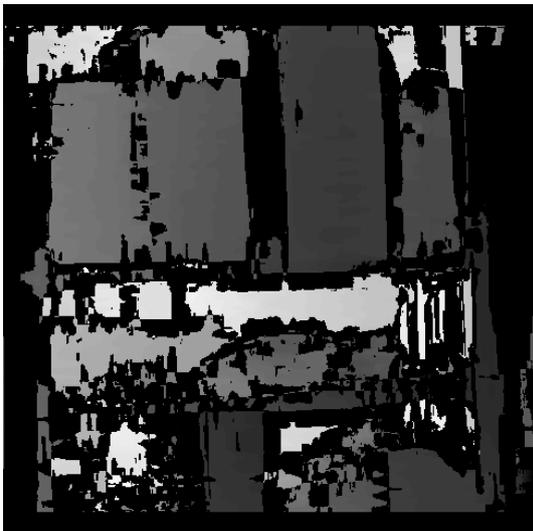
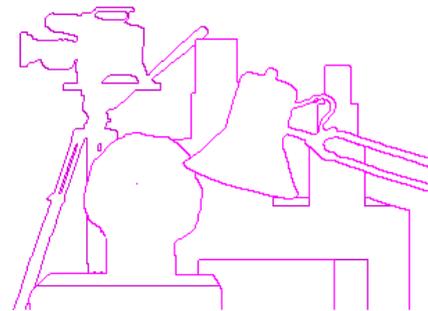
(c) Notre carte ($k = 21$)(d) OpenCV ($k = 21$)(e) Notre carte ($k = 41$)(f) OpenCV ($k = 41$)

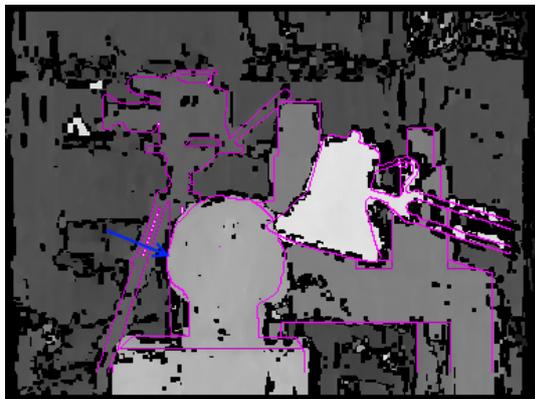
FIGURE 4.18 – Évolution de l'effet de dilatation en fonction du noyau et comparaison avec OpenCV



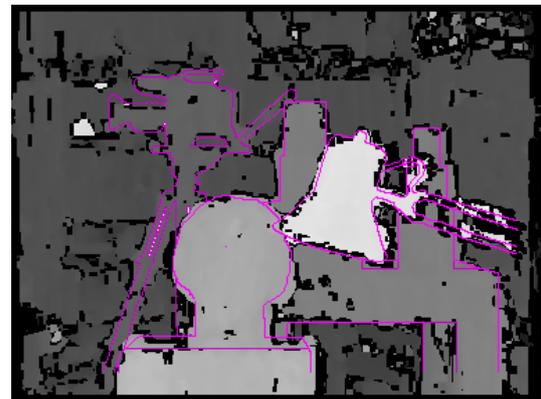
(a) Vérité terrain



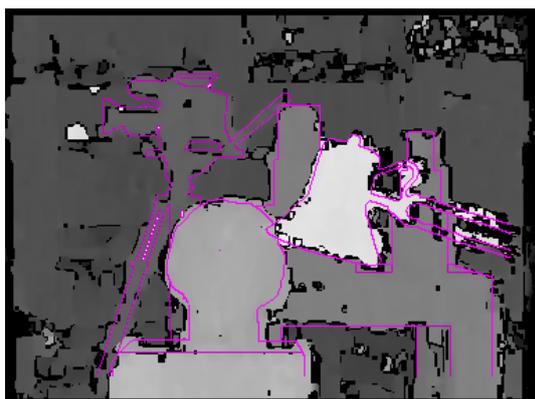
(b) Les contours



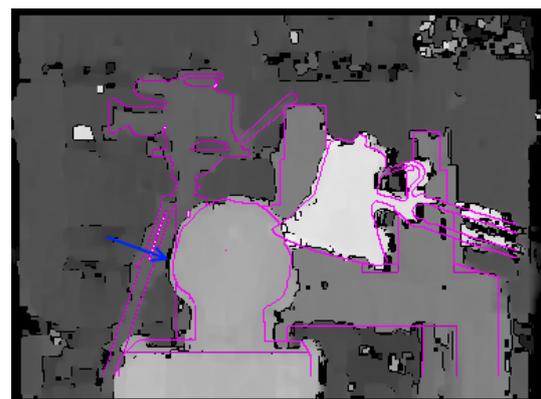
(c) Tolérance de 3



(d) Tolérance de 5



(e) Tolérance de 7



(f) Tolérance de 9

FIGURE 4.19 – Influence de la tolérance (<http://vision.middlebury.edu/stereo/data/>)

Conclusion

Différentes approches pour produire des cartes denses de disparité existent mais les méthodes par stéréo-corrélation permettent d'avoir un bon compromis entre la qualité des images et le temps de traitement.

Nous avons présenté dans ce chapitre notre méthode de production de carte dense de disparité qui est un moyen de construire une représentation d'une scène dans sa profondeur. Nous avons présenté les différentes étapes de calibration stéréo, de rectification, de stéréo-corrélation et d'amélioration qualitative et quantitative.

Le calcul du score de corrélation est grandement accéléré :

- dans l'algorithme par les propriétés de séparabilité du calcul du score de corrélation et de récursivité entre eux,
- dans l'implémentation par la construction en parallèle des deux cartes denses de disparité en ligne et en colonne et par l'utilisation d'instructions XMMX.

La qualité de notre carte dense de disparité est améliorée :

- par l'utilisation d'une source lumineuse augmentant la texture de la scène qui garantit une meilleure mise en correspondance,
- par l'interpolation de la disparité qui assure une continuité sur la surface des objets,
- par l'utilisation de deux noyaux mono-directionnels garantissant la conservation de la forme des objets apparaissant au premier plan.

Des résultats expérimentaux sur des applications industrielles seront présentés par la suite montrant l'intérêt d'utiliser la méthode par production de carte dense de disparité.

Chapitre 5

Évaluation pour des applications industrielles

Sommaire

5.1	Application de recalage 3D	95
5.1.1	Phase de calibration	95
5.1.2	Phase de production	96
5.1.3	Bilan	100
5.2	Application de décaissage	101
5.2.1	Phase de calibration	101
5.2.2	Cycle de production	103
5.2.3	Bilan	105
5.3	Application de contrôle d’aspect	107
5.3.1	Phase de calibration	107
5.3.2	Phase de production	107
5.3.3	Bilan	110
5.4	Application de contrôle de défaut	114
5.4.1	Acquisition	114
5.4.2	Phase de calibration	114
5.4.3	Analyse de défaut	115
5.4.4	Bilan	116

Introduction

Dans ce chapitre, nous avons mis en place les deux méthodes de reconstruction 3D directe et de production de carte dense de disparité pour des applications industrielles.

Ces deux méthodes sont destinées à des applications différentes dans la mesure où la première fournit des positions tri-dimensionnelles sur des points pré-définis de l'espace alors que la seconde ne fournit qu'une représentation de la profondeur. Il est donc évident qu'une carte de disparité en tant que telle ne permet pas d'obtenir une mesure 3D.

Ainsi, nous allons dans un premier temps présenter deux applications nécessitant l'utilisation de la reconstruction 3D : le recalage 3D et le décaissage.

Puis, nous présenterons deux applications nécessitant la production de carte dense de disparité afin d'analyser les variations de surface d'un objet.

5.1 Application de recalage 3D

Dans cette partie, nous vérifions nos résultats de test sur une application industrielle de recalage, c'est à dire localiser une pièce de production (culasse) dans l'espace monde prédéfini. Lors de la première phase de test, le robot nous présentait un motif à différentes position 3D et le système stéréoscopique fixe estimait ces positions. Dans un contexte industrielle, pour les applications de production notamment, l'encombrement, le poids, ou d'autres paramètres encore, impliquent que la pièce est très souvent présentée de manière aléatoire devant un système de vision qui effectue la mesure.

C'est ainsi que pour mettre en évidence la précision de la reconstruction 3D sur ce type d'application industrielle, il est nécessaire de figer la pièce à localiser et de rendre mobile le système de capture.

Ainsi, les deux caméras sont embarquées sur le robot qui se déplace en discrétisant un volume constant.

5.1.1 Phase de calibration

Dans un premier temps, chaque caméra est calibrée indépendamment afin d'estimer leurs paramètres intrinsèques. Nous utilisons une mire de calibration fournissant 183 points mesurables que nous présentons devant chacune des caméras avec cinq orientations différentes. Nous utilisons deux caméras de résolution voisine pour lesquelles nous devons connaître la valeur métrique du pixel pour réaliser la calibration :

- une caméra (gauche) DALSA de résolution 1360x1024 avec une mesure pixellique de $4.65\mu m$
- une caméra (droite) BASLER de résolution 1392x1040 avec une mesure pixellique de $4.65\mu m$

Dans un second temps, nous posons une mire de calibration fixe sur une surface plane. Le robot définit le repère monde à partir de cette mire. Ensuite, le robot embarquant les caméras se positionne dans la position de référence, la position moyenne du volume, afin de définir les paramètres extrinsèques de chaque caméra par rapport à la même mire de calibration.

La calibration des deux caméras nous fournissent ainsi les matrices des paramètres intrinsèques et extrinsèques suivantes :

Pour la caméra gauche :

$$K = \begin{bmatrix} 1707.71 & 0 & 714.30 \\ 0 & 1705.70 & 501.64 \\ 0 & 0 & 1 \end{bmatrix} \text{ et } M = \begin{bmatrix} 0.982 & 0.018 & 0.188 & -122.141 \\ 0.017 & -1.000 & 0.004 & 21.219 \\ 0.188 & -0.001 & -0.982 & 924.012 \end{bmatrix} \quad (5.1)$$

et pour la caméra droite :

$$K = \begin{bmatrix} 1733.92 & 0 & 725.35 \\ 0 & 1733.38 & 480.97 \\ 0 & 0 & 1 \end{bmatrix} \text{ et } M = \begin{bmatrix} 0.994 & 0.009 & -0.108 & 114.173 \\ 0.008 & -1.000 & -0.008 & 35.579 \\ -0.108 & 0.007 & -0.994 & 928.659 \end{bmatrix} \quad (5.2)$$

Nous pouvons en déduire la configuration des caméras. Les caméras sont positionnées dans le repère monde en :

$$C_g = \begin{bmatrix} -122.141 \\ 21.219 \\ 924.012 \end{bmatrix} \text{ et } C_d = \begin{bmatrix} 114.173 \\ 35.579 \\ 928.659 \end{bmatrix} \quad (5.3)$$

Nous en déduisons l'écartement inter-caméra :

$$E = \sqrt{(C_g(x) - C_d(x))^2 + (C_g(y) - C_d(y))^2 + (C_g(z) - C_d(z))^2} = 236.8mm \quad (5.4)$$

5.1.2 Phase de production

La phase de production consiste à détecter les points caractéristiques de la culasse, à faire l'appariement inter-image de ces points et enfin calculer la position 3D pour chaque paire de points.

En pratique, à la position de référence, nous conservons la position robot Pr_{ref} et la position mesurée Pm_{ref} . Pour chaque points de l'espace prise par le robot, nous mesurons la position 3D par reconstruction 3D directe et nous calculons l'erreur comme la différence entre la position de référence mesurée et la position courante mesurée mais décalé de la translation du robot avec la position de référence robot :

$$\begin{cases} E_x(P_i) = (P_i(x) - Pm_{ref}(x) - (Pr_{ref}(x) - Pr_{courante}(x)))^2 \\ E_y(P_i) = (P_i(y) - Pm_{ref}(y) - (Pr_{ref}(y) - Pr_{courante}(y)))^2 \\ E_z(P_i) = (P_i(z) - Pm_{ref}(z) - (Pr_{ref}(z) - Pr_{courante}(z)))^2 \end{cases} \quad (5.5)$$

$$E_{distance} = \sqrt{E_x^2 + E_y^2 + E_z^2}$$

L'évaluation est réalisée sur une culasse, une pièce métallique dont la surface réfléchissante et peu texturée est une contrainte importante. Le recalage 3D nécessite de définir un ensemble de points caractéristiques de la pièce que l'on puisse localiser dans les deux images de manière précise et répétable. Des méthodes d'éclairage nous permettent de mettre en exergue certains éléments de la pièce. En l'occurrence, les culasses présentent

des cavités et l'utilisation d'un éclairage rasant permet d'illuminer la partie supérieure de la pièce et d'assombrir ces cavités. Cela permet de faire ressortir des formes caractéristiques que nous pouvons détecter par des traitements adaptés.

5.1.2.1 Localisation de points caractéristiques

Nous avons défini trois points caractéristiques qui représentent le nombre minimum de points nécessaires pour localiser une pièce dans l'espace. Nous utilisons des outils de traitement d'image issu de la librairie propriétaire Vision Pro de la société Cognex. L'éclairage rasant nous permet de mettre en évidence trois zones de la culasse qui seront traitées par des outils adaptés. La figure 5.1 nous les présente.

Les zones doivent être suffisamment grandes pour être adaptées aux différentes positions des caméras.

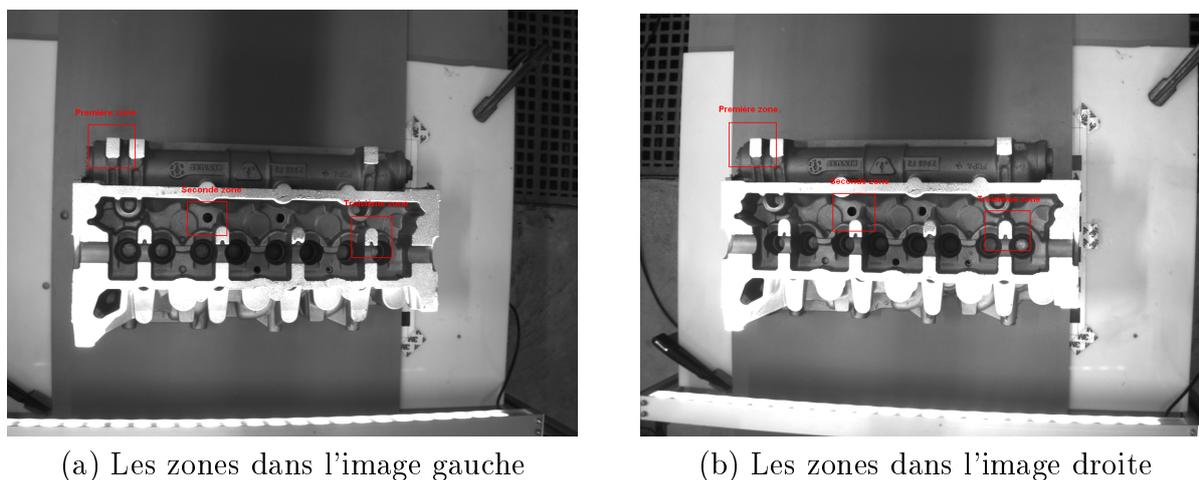


FIGURE 5.1 – Sélection des points caractéristiques de la culasse

Les premier et troisième points caractéristiques de la culasse sont détectés à partir d'un traitement de recherche de blob. Pour assurer l'unicité du blob, le traitement nous permet de filtrer les blobs par rapport à leur surface (le nombre de pixels) et leur circularité. Nous avons défini le centre de masse comme le point caractéristique du blob.

Le second point caractéristique de la culasse est détecté à partir d'une recherche d'ellipse. La recherche d'ellipse fonctionne par la recherche d'un ensemble de transition le long d'une ellipse. Elle est le résultat de la meilleure ellipse passant par l'ensemble de points de transition, plus précisément en minimisant l'erreur entre les transitions et l'ellipse à détecter. Le traitement nous permet de supprimer un ensemble de points dont l'erreur seraient trop importante. Nous avons défini le centre de l'ellipse comme le point caractéristique de l'ellipse.

Nous observons le résultat de la détection des trois points caractéristiques dans la figure 5.2.

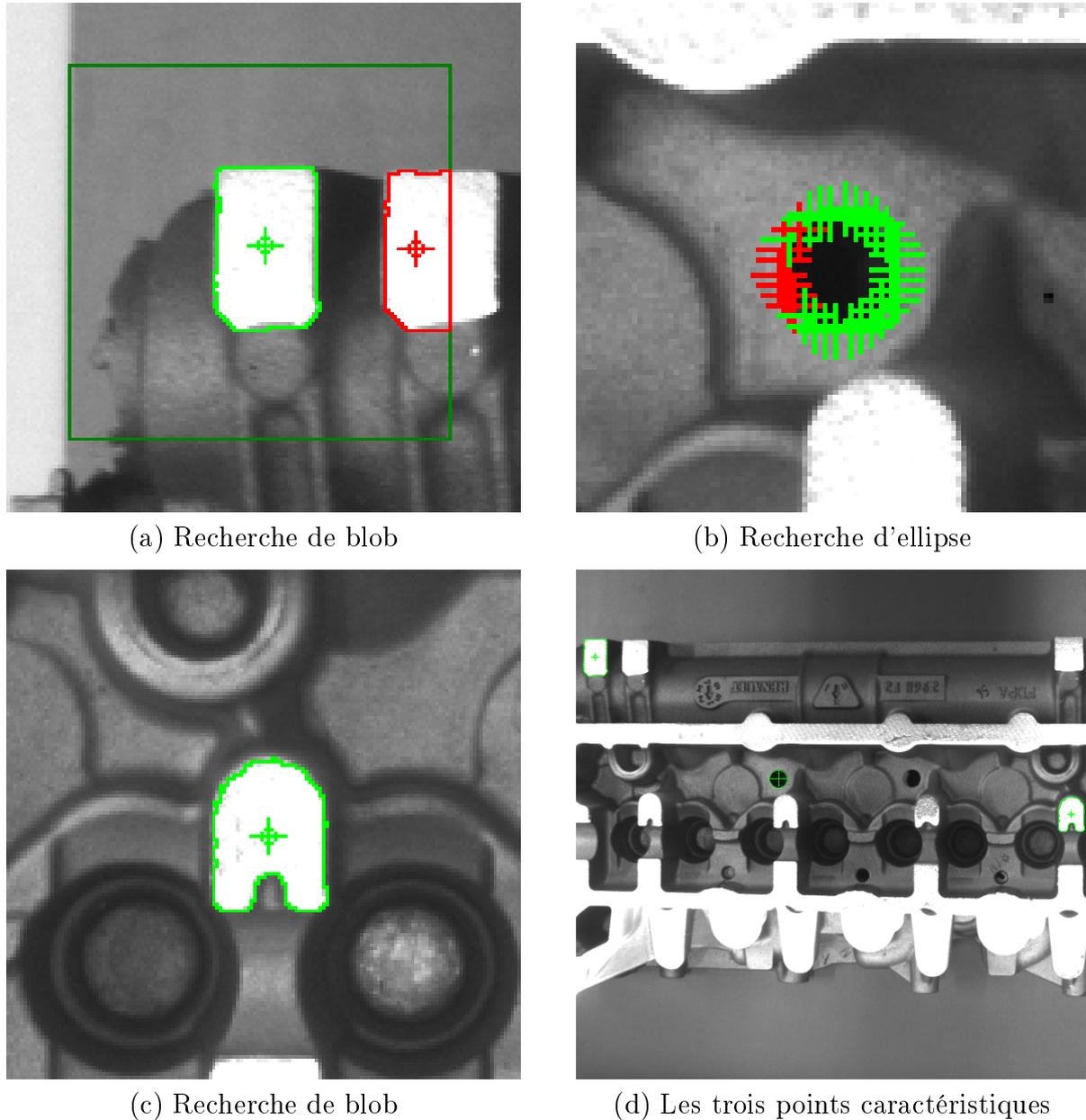


FIGURE 5.2 – Traitements pour la localisation des points caractéristiques de la culasse

Le pseudo alignement par rapport à l'axe des y dans les deux images ainsi que l'ordre des traitements, rendent automatique l'appariement des points caractéristiques.

Au final, pour chaque image nous construisons trois couples de point image, ainsi que la paire du point moyen qui sera la donnée d'entrée pour estimer la précision du système de reconstruction 3D.

5.1.2.2 Reconstruction 3D

Nous avons déplacé le système de caméras embarqué sur le robot le long d'un volume de dimension $300 \times 200 \times 200 \text{ mm}^3$ pour un ensemble de 816 positions.

Nous avons évalué la précision 3D par mesure de l'écart-type entre la valeur mesurée et la valeur attendu sur l'ensemble des points caractéristiques, puis pour l'ensemble des points moyens. Nous évaluons aussi cette précision pour des volumes réduits.

Nous présentons les statistiques d'erreur pour les trois axes et pour l'erreur en distance entre le point mesuré et le point réel avec visualisation des erreurs minimales et maximales. Nous montrons nos résultats avec et sans la correction de l'erreur par régression 3D.

Erreur en mm	Minimum	Maximum	Moyenne	Ecart-type
X	-4.508	3.282	0.000	1.581
Y	-3.180	2.302	0.000	1.168
Z	-3.723	3.122	0.000	0.928
Distance	0.310	5.726	1.968	0.920

TABLE 5.1 – Erreur de précision sans régression (volume $20 \times 20 \times 20 \text{ cm}^3$)

Erreur en mm	Minimum	Maximum	Moyenne	Ecart-type
X	-0.666	0.782	0.000	0.228
Y	-0.383	0.695	0.000	0.139
Z	-0.689	0.816	0.000	0.211
Distance	0.028	1.014	0.301	0.159

TABLE 5.2 – Erreur de précision avec régression (volume $20 \times 20 \times 20 \text{ cm}^3$)

Les tables 5.1 et 5.2 montrent l'influence de la correction apportée par la régression 3D. Le gain en précision est de l'ordre de 6. Au final, l'erreur moyenne en distance est de 0.301 mm pour un écart type de 0.159 mm . Pour tout point du volume, l'erreur est inférieure au millimètre.

Erreur en mm	Minimum	Maximum	Moyenne	Ecart-type
X	-0.604	0.532	0.000	0.190
Y	-0.276	0.282	0.000	0.092
Z	-0.442	0.469	0.000	0.152
Distance	0.016	0.679	0.228	0.124

TABLE 5.3 – Erreur de précision avec régression (volume $18 \times 18 \times 18 \text{ cm}^3$)

Les tables 5.2, 5.3, 5.4 et 5.5 montrent l'influence du volume décrit par le robot sur la précision de la mesure 3D. Le volume de déplacement du robot est, en pratique, en

Erreur en mm	Minimum	Maximum	Moyenne	Ecart-type
X	-0.428	0.333	0.000	0.137
Y	-0.155	0.138	0.000	0.056
Z	-0.247	0.302	0.000	0.090
Distance	0.017	0.439	0.152	0.083

TABLE 5.4 – Erreur de précision avec régression (volume 14x14x14 cm^3)

Erreur en mm	Minimum	Maximum	Moyenne	Ecart-type
X	-0.391	0.285	0.000	0.138
Y	-0.126	0.122	0.000	0.049
Z	-0.280	0.219	0.000	0.079
Distance	0.031	0.401	0.144	0.082

TABLE 5.5 – Erreur de précision avec régression (volume 10x10x10 cm^3)

corrélation avec les variations de positionnement aléatoire que la culasse pourrait définir. Au final, les résultats montrent que moins la position de la culasse est susceptible de varier, meilleure sera la précision 3D.

5.1.3 Bilan

Dans cette application de recalage 3D, nous conservons la précision de mesure 3D présentée lors de l'expérimentation de la méthode de reconstruction 3D directe, bien que l'erreur de précision 2D de l'analyse de blob et de recherche de cercle est susceptible d'exister.

La précision du recalage 3D dépend également du nombre de points caractéristiques utilisés pour définir le positionnement de la culasse. Nous supposons que plus le nombre de points augmentent, meilleure est l'imprécision.

5.2 Application de décaissage

Dans cette partie, nous présentons une application industrielle de décaissage sur des rouleaux d'impression présents dans une caisse. Une démonstration de localisation de rouleaux et de saisie par un système robotique a été mise en place lors d'un salon industrielle [PV09] reposant sur les mêmes méthodes de calibration.

L'objectif ici est d'obtenir une application de démonstration dont les éléments de calibration et de production peuvent être modifiés et enregistrés.

Nous avons décomposé l'application en cinq phases permettant de réaliser les différentes calibrations et la localisation 3D.

5.2.1 Phase de calibration

5.2.1.1 Calibrations des paramètres intrinsèques

La calibration des paramètres intrinsèques des deux caméras est réalisée à partir d'images de mires acquises hors-production.

Nous sauvegardons la calibration des paramètres intrinsèques par enregistrement des matrices internes, de la taille des images et de la valeur métrique des pixels.

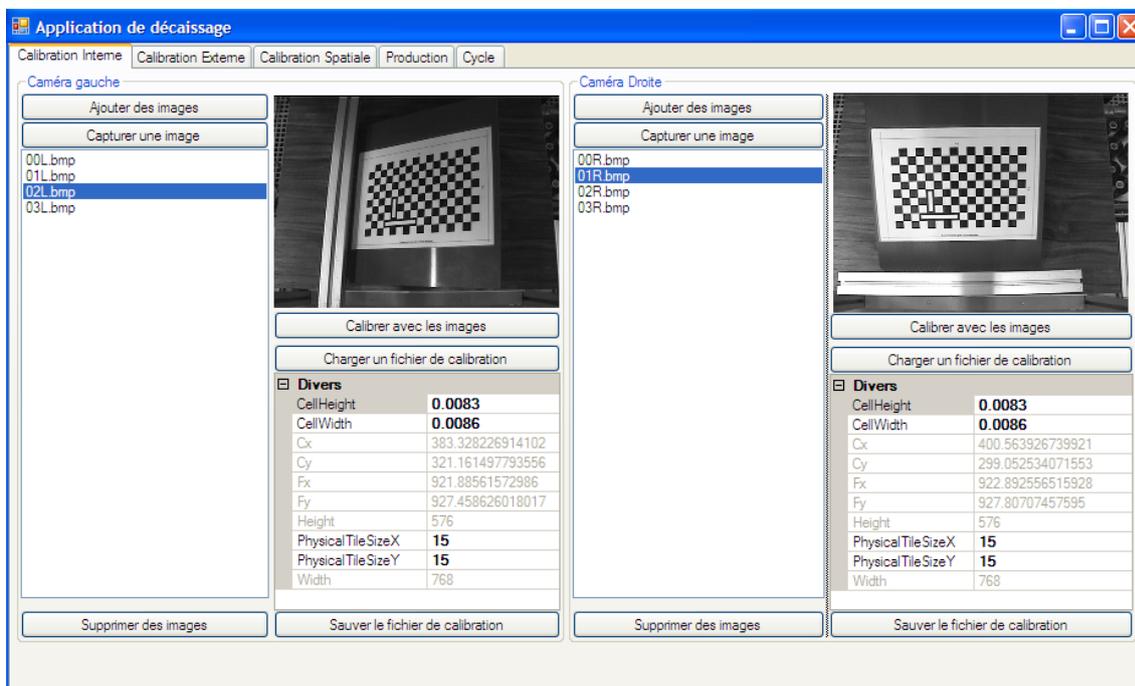


FIGURE 5.3 – IHM : Calibration des paramètres intrinsèques des deux caméras

5.2.1.2 Calibrations des paramètres extrinsèques

La calibration des paramètres extrinsèques est réalisée à partir d'une mire unique acquise par les deux caméras. Le robot définit son repère à partir de cette même mire.

Nous sauvegardons la calibration des paramètres extrinsèques par enregistrement des matrices de projection.

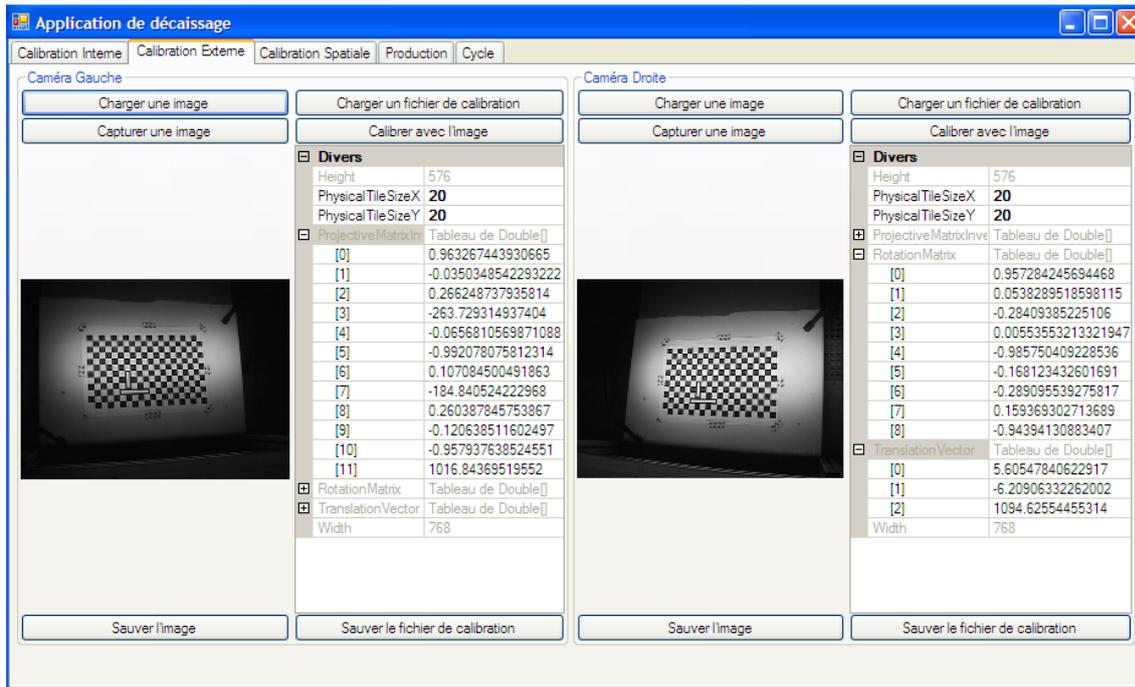


FIGURE 5.4 – IHM : Calibration des paramètres extrinsèques des deux caméras

5.2.1.3 Calibration spatiale

La dernière étape de calibration est l'estimation du modèle entre les points 3D mesurés et les points 3D réels. Les points 3D mesurés sont définis à partir de la localisation 2D des rouleaux, des éléments de calibration monoculaires et de la triangulation 3D. Les points 3D réels correspondent aux positions du préhenseur du robot.

La réalisation de la calibration spatiale se base sur la méthode d'évaluation de la précision de la reconstruction 3D que nous avons utilisée. En effet, nous déplaçons un motif que nous estimons "facilement" détectable dans les deux images. L'ensemble des paires de points 2D nous permet de construire le modèle de régression 3D entre la mesure 3D et les positions du robot.

Nous sauvegardons la calibration spatiale par enregistrement des 30 paramètres de système d'équation 3.23.

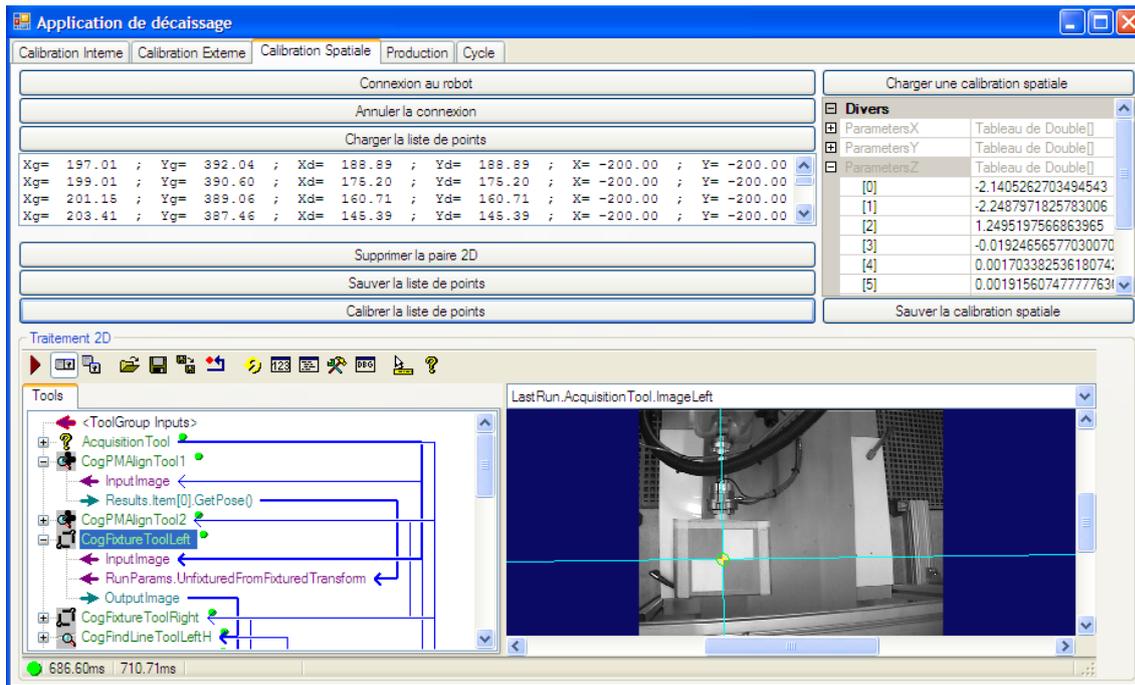


FIGURE 5.5 – IHM : Calibration du modèle de régression 3D

5.2.2 Cycle de production

La production est l'étape de recherche des primitives des rouleaux que nous souhaitons localiser ainsi que l'appariement des primitives des deux images appartenant au même rouleau. Nous avons ensuite défini une procédure pour réaliser le décaissage des rouleaux par le robot.

5.2.2.1 Traitements 2D

La localisation dans les images des rouleaux passe par la définition d'une primitive commune que nous sommes capables d'identifier de manière précise et répétable. Nous avons défini le centre de l'ellipse formée par le mandrin du rouleau comme le point caractéristique de chaque rouleau.

L'appariement des points caractéristiques des rouleaux se base sur la connaissance *a priori* de la scène. En effet nos caméras sont disposées de sorte que les rouleaux soient ordonnés de la même manière dans les deux images. Aussi, la connaissance de l'organisation des rouleaux dans la caisse nous amène à réaliser la localisation des rouleaux sur plusieurs zones, c'est à dire une zone par ligne de rouleaux.

Pour chaque zone de recherche, nous nous aidons de la contrainte d'ordre de la ligne de rouleaux pour réaliser l'appariement directement.

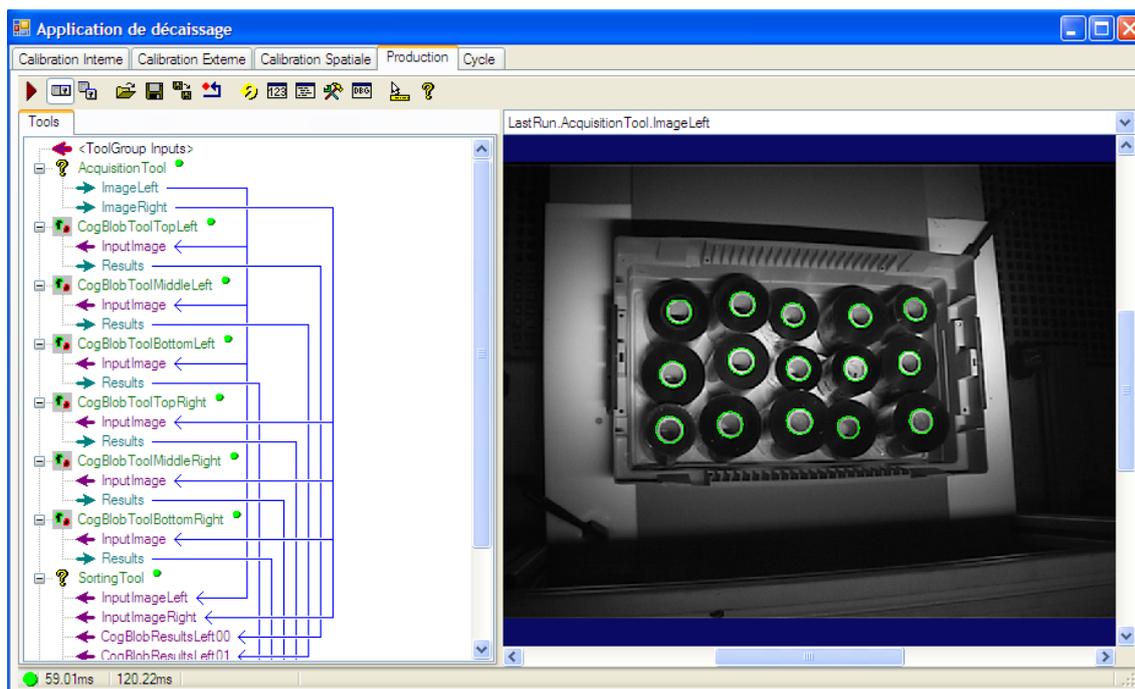
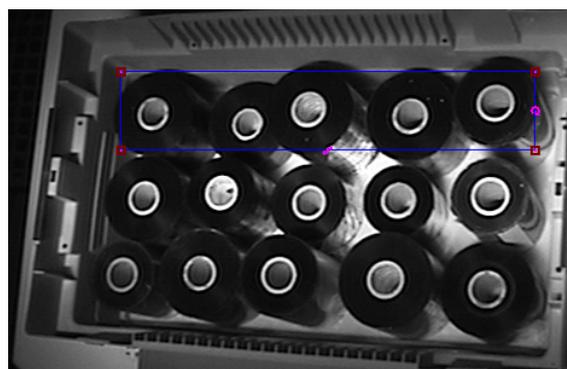


FIGURE 5.6 – IHM : Programme de détection et d'appariement des rouleaux



(a) Zone dans l'image gauche



(b) Zone dans l'image de droite



(c) Détection de rouleaux à gauche



(d) Détection des rouleaux à droite

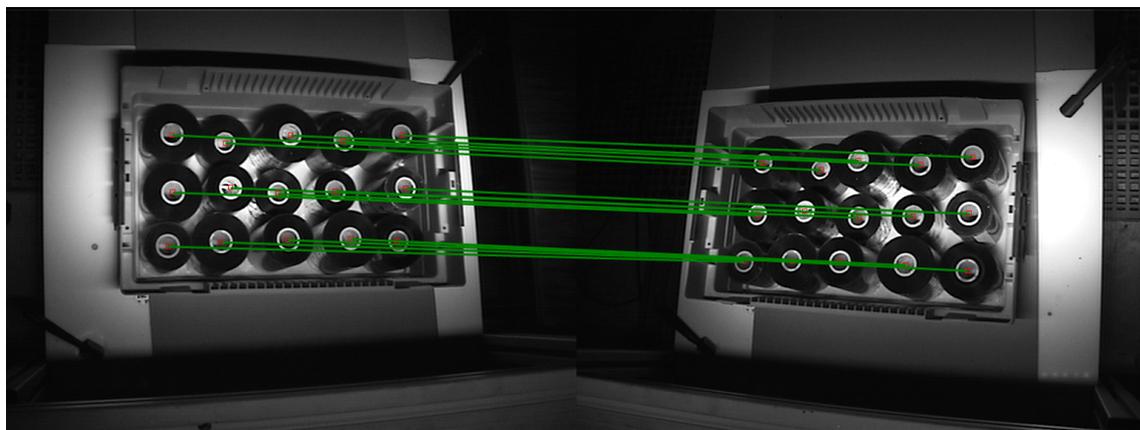
FIGURE 5.7 – Première zone de recherche de rouleaux



(a) Image de gauche



(b) Image de droite



(c) Les appariements

FIGURE 5.8 – Mise en correspondance des rouleaux des deux images après découpage en zone et utilisation de la contrainte d'ordre

5.2.2.2 Gestion du cycle

Le cycle de production consiste à interagir entre l'application de vision et le robot. Pour cela, une procédure a été mise en place pour réaliser le décaissage des rouleaux. Le robot gère le comptage des rouleaux, nous lui fournissons donc le nombre de rouleaux à décaisser. Le robot fait des demandes de position, le système de vision lance le programme de localisation des rouleaux dans les images, d'appariement des points caractéristiques des rouleaux et de la localisation 3D des rouleaux. Le robot récupère la position du rouleau le plus haut et réalise le déplacement pour saisir le rouleau et le placer sur un support.

5.2.3 Bilan

Cette application a été mise en place comme démonstration utilisant la reconstruction 3D pour évaluer une application de décaissage. Les tolérances de précision étaient de l'ordre du millimètre pour que le préhenseur du robot puisse pénétrer à l'intérieur du

rouleau.

L'application est évolutive et peut être exportée à n'importe quel type d'objet sous contrainte que les éléments soient stéréo-visibles et que l'on puisse réaliser l'appariement.

5.3 Application de contrôle d'aspect

Dans cette partie, nous évaluons la production de carte de disparité dense pour la constitution de profil représentatif de la surface d'un objet afin de détecter d'éventuels surplus ou manque de matière.

La production de carte dense de disparité est sensible aux conditions d'acquisition et aux conditions environnementales. A partir de cette connaissance, il était important de bien définir ces conditions.

Premièrement nous avons observé que l'angle formé par les deux axes optiques des caméras est proportionnel à la déformation de rectification de nos images. Aussi, l'erreur d'alignement des points image après rectification, est minimale au centre de l'image et augmente lorsque que l'on se déplace vers les pourtours des images. Enfin, la configuration pseudo-aligné nous oblige à augmenter la distance caméras-scène ou diminuer l'écartement inter-caméra pour que la surface analysée soit dans le champ des deux caméras. Ces contraintes nous amènent à positionner les caméras dans une configuration au plus proche de l'alignement avec un écartement inter-caméra faible et une distance caméras-surface adaptée pour que la surface soit le plus centrée dans les deux images.

Deuxièmement, nous avons également observé que la production de carte dense nécessite d'analyser des surfaces suffisamment texturée pour garantir un appariement des points image. La projection de mouchetis est utilisée pour répondre à cette problématique.

5.3.1 Phase de calibration

La calibration du système de caméra a été réalisée à partir de mires stéréo-visibles, nous permettant d'obtenir une liste de paires de points image qui se correspondent dans les deux images. En pratique, nous présentions cinq mires fournissant 183 points chacune, avec une orientation différente.

L'ensemble des paires de points nous permet d'estimer la matrice fondamentale et les matrices de rectification.

5.3.2 Phase de production

La phase de production se décompose en deux phases :

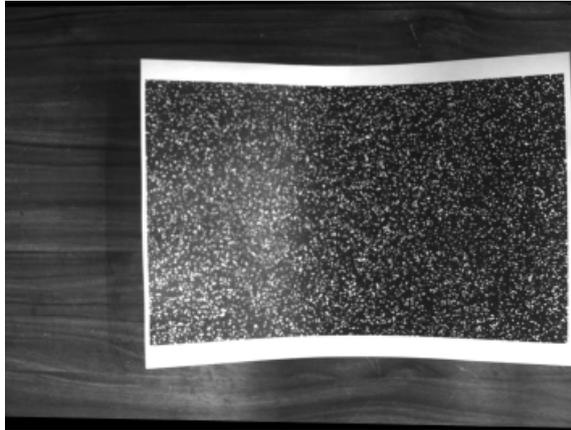
- La rectification des deux images d'entrée
- La production de carte dense de disparité

Pour l'évaluation, nous avons fait abstraction de la projection de mouchetis par l'utilisation d'une surface recouverte d'une feuille imprimée d'une image bruitée.

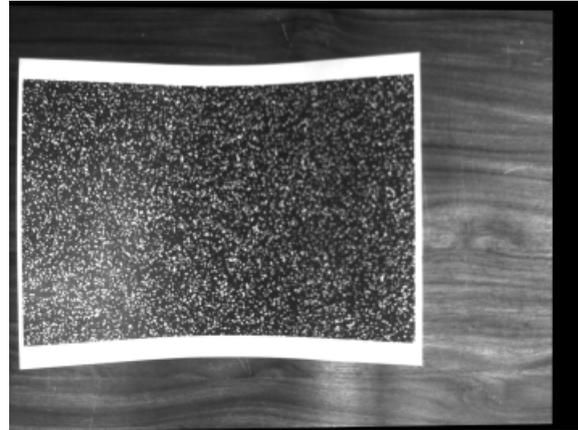
Nous avons réalisé quatre configurations pour la pose des caméras en faisant varier la distance caméras-surface, l'écartement des caméras.

5.3.2.1 Première configuration

Dans un premier temps, nous avons réalisé le profil d'une surface présentant une courbure pour valider l'utilisation de l'image de bruit. Nous pouvons visualiser en figure 5.9 la carte dense de disparité ainsi que la représentation tridimensionnelle de la surface.



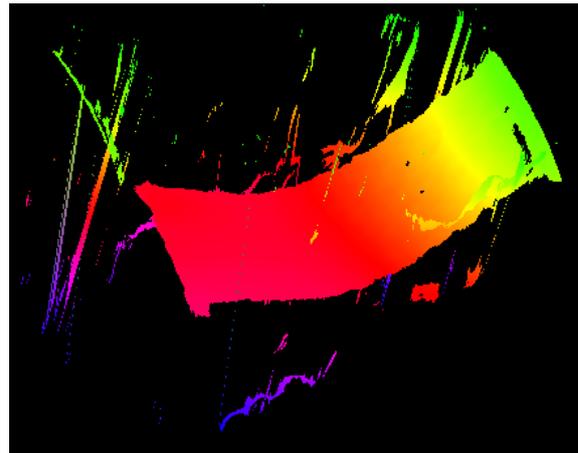
(a) Image gauche rectifiée



(b) Image droite rectifiée



(c) Carte de disparité



(d) Visualisation 3D

FIGURE 5.9 – Visualisation d'une surface mouchetée sans défaut

Dans un second temps, nous avons superposé un élément moucheté d'une épaisseur de $3mm$. Nous pouvons visualiser le résultat en figure 5.10.

L'élément est détectable mais l'amplitude est faible.

5.3.2.2 Deuxième configuration

Dans cette configuration, nous avons rapproché le système d'acquisition de la surface, tout en adaptant l'écartement inter-caméra afin de visualiser la partie de la surface pour laquelle nous avons ajouté des éléments de hauteurs différentes.

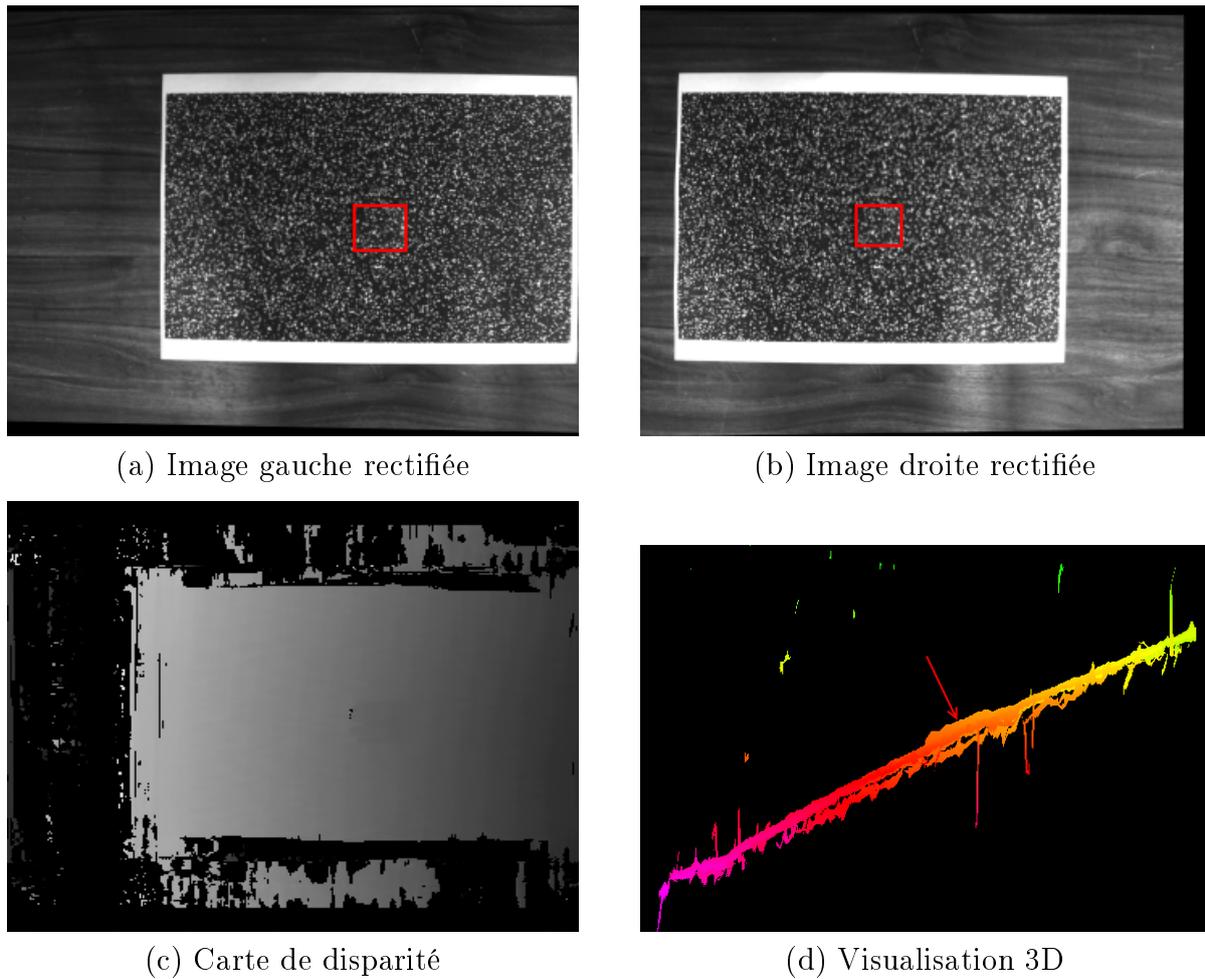


FIGURE 5.10 – Visualisation d'un élément de 3mm d'épaisseur

Pour la même scène observé qu'en première configuration, nous pouvons voir en figure 5.11 que l'amplitude de l'élément carré a fortement augmenté. Sa détection est grandement facilitée.

Pour une seconde scène observée, nous avons placé deux autres éléments d'épaisseur moindre, de l'ordre de $2mm$, que nous sommes toujours en mesure de visualiser. Les résultats sont en figure 5.12.

5.3.2.3 Troisième configuration

Dans une troisième configuration, nous souhaitons mettre en évidence l'apport de l'utilisation de noyaux mono-directionnels.

Nous pouvons voir en figure 5.13 que l'utilisation deux noyaux mono-directionnels conserve mieux les deux formes carré par rapport à la méthode utilisant un noyau carré.

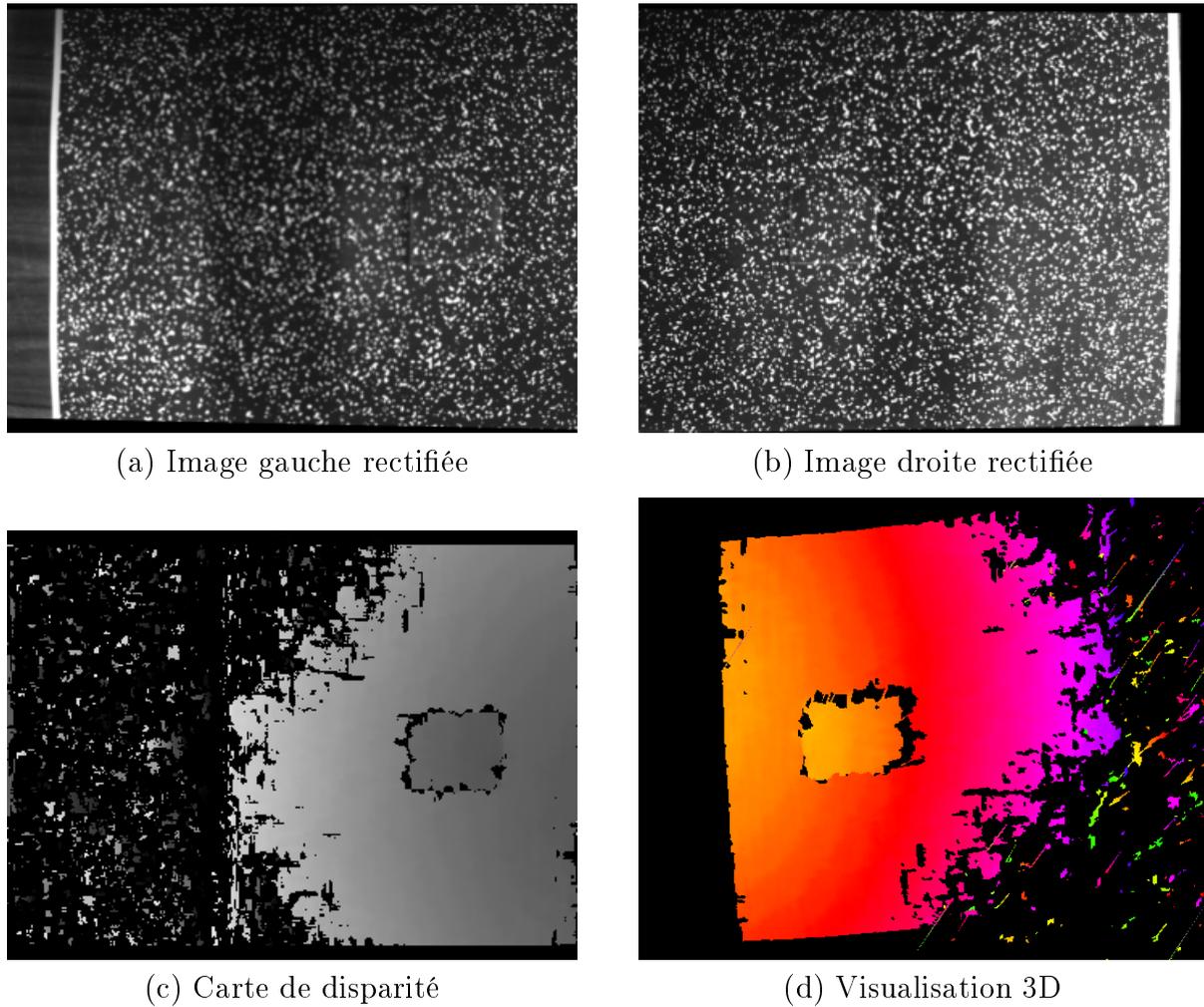


FIGURE 5.11 – Visualisation d'un élément de 3mm d'épaisseur avec un rapprochement de nos caméras

5.3.2.4 Quatrième configuration

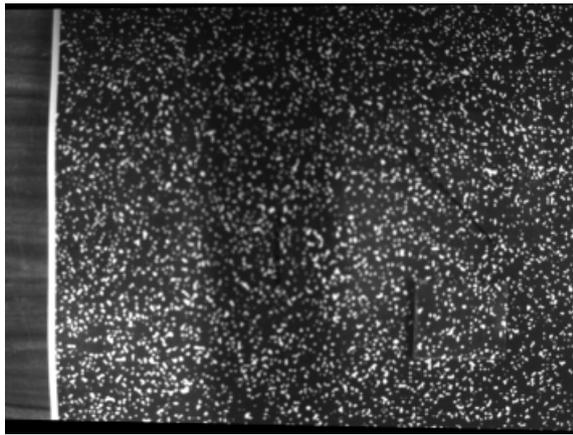
Dans la quatrième configuration, nous avons simulé des chocs sur la surface. Nous visualisons les trois chocs sur la figure 5.14.

La dimension des chocs est de l'ordre du millimètre.

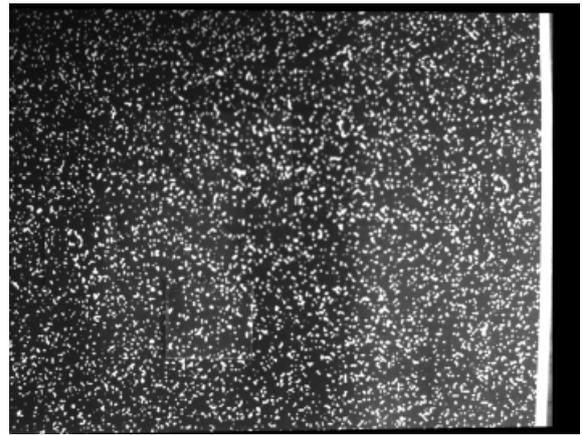
5.3.3 Bilan

Nous montrons dans ces premiers résultats que la production de carte dense de disparité nous fournit uniquement une représentation de la profondeur de la scène observée qui ne nous permet de donner une valeur métrique des éléments. Néanmoins, à partir de la connaissance métrique des éléments, la visualisation nous fournit une information de notre capacité à les détecter.

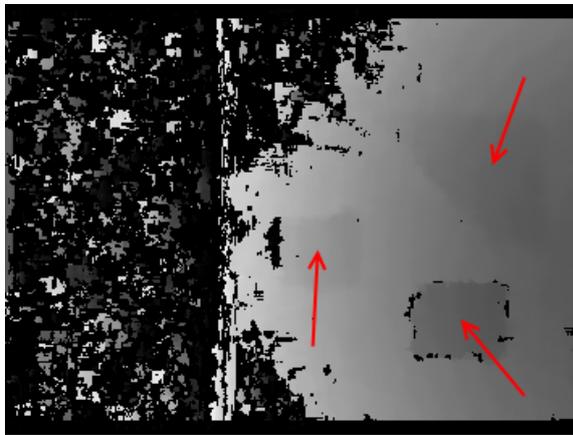
Nous remarquons que plus nous rapprochons nos caméras de la scène, meilleure est la



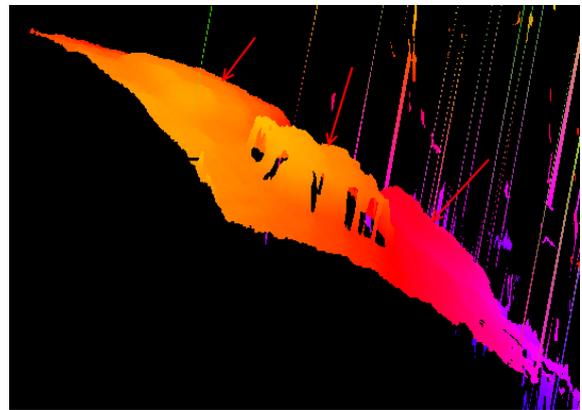
(a) Image gauche rectifiée



(b) Image droite rectifiée



(c) Carte de disparité

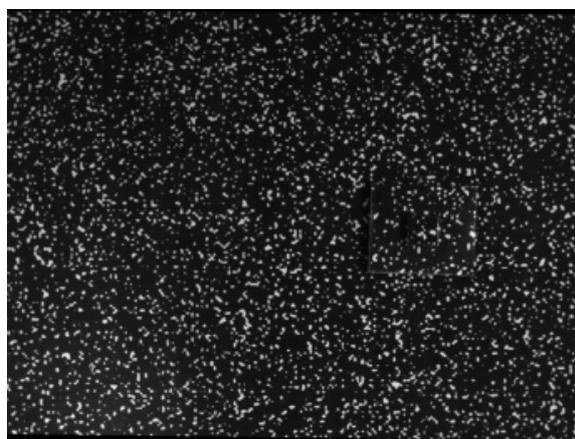


(d) Visualisation 3D

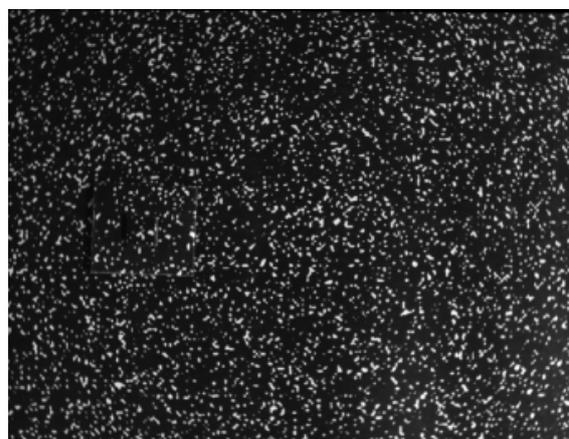
FIGURE 5.12 – Visualisation d'éléments de 2mm d'épaisseur

précision de notre carte de disparité puisqu'elle nous permet de visualiser des éléments de plus faible amplitude. Néanmoins la surface analysée, commune aux deux caméras, s'en retrouve réduite.

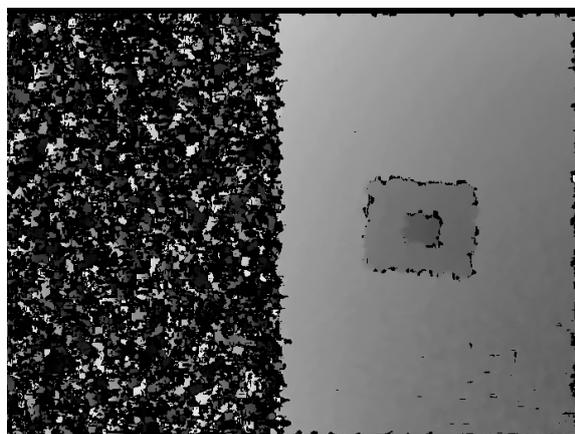
Enfin, l'utilisation de noyaux mono-dimensionnels conserve davantage la dimension des éléments du fait de la réduction du phénomène de dilatation. Ceci peut présenter un intérêt si l'on veut dimensionner ces éléments.



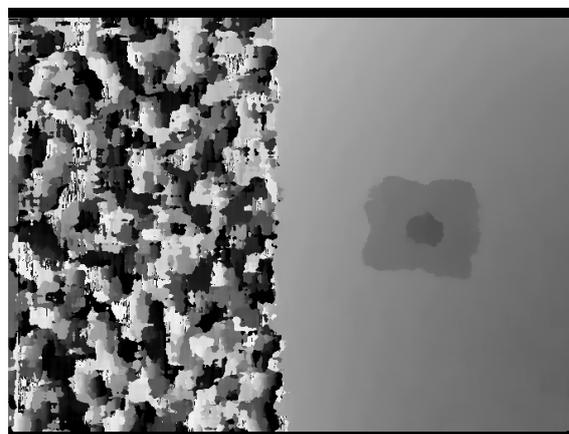
(a) Image gauche rectifiée



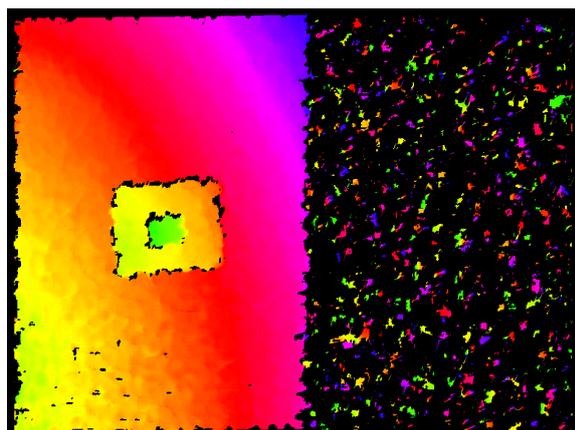
(b) Image droite rectifiée



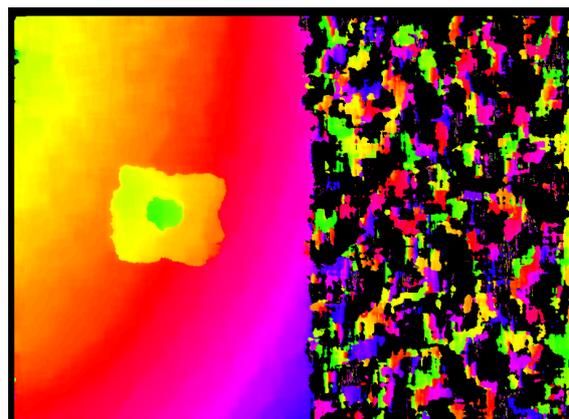
(c) Notre carte de disparité



(c) Carte de disparité avec un noyau carré

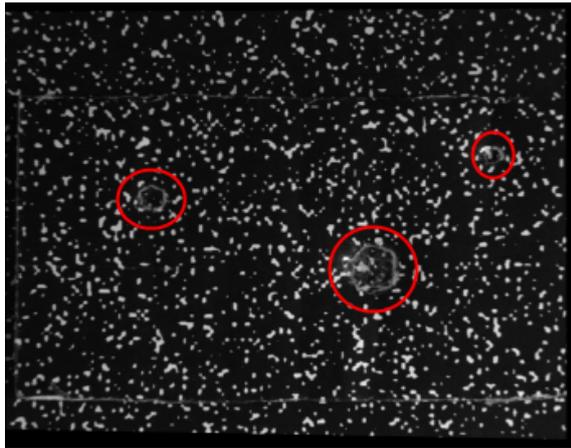


(d) Notre visualisation 3D

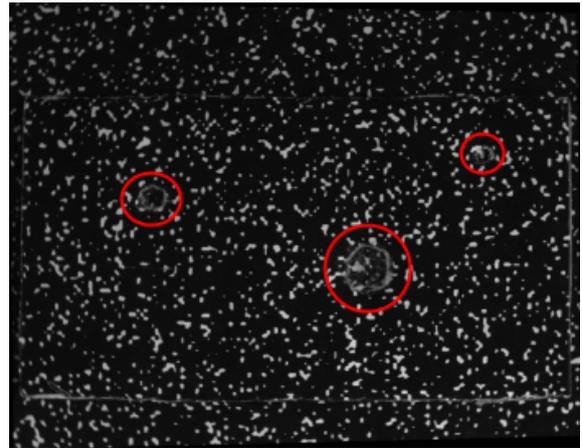


(d) Visualisation 3D avec un noyau carré

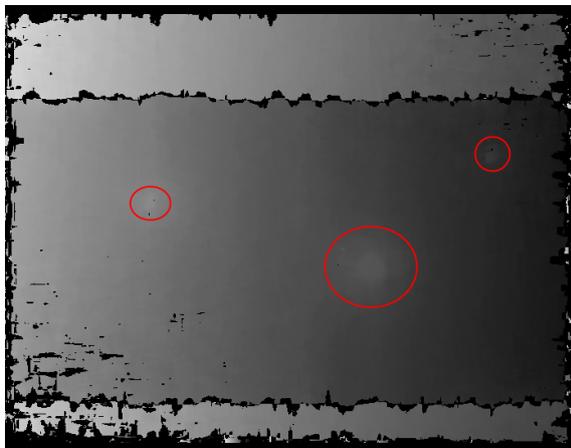
FIGURE 5.13 – Influence de l'utilisation de noyaux mono-directionnels



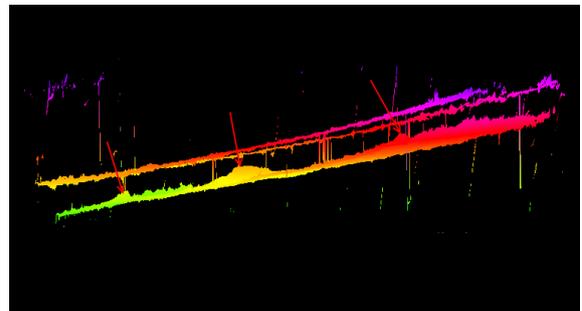
(a) Image gauche rectifiée



(b) Image droite rectifiée



(c) Carte de disparité



(d) Visualisation 3D

FIGURE 5.14 – Visualisation de chocs de 1mm d'épaisseur

5.4 Application de contrôle de défaut

Dans cette partie, nous présentons une application d'analyse de défaut sur une canette. L'objectif d'obtenir une application de démonstration dont les éléments de calibration et de production peuvent être modifiés et enregistrés.

Nous avons décomposé l'application en trois phases permettant de réaliser la calibration, l'apprentissage du profil et l'analyse de défaut.

5.4.1 Acquisition

L'acquisition des deux images est réalisée successivement. Il est donc nécessaire que la scène observée soit statique entre les deux prises de vue.

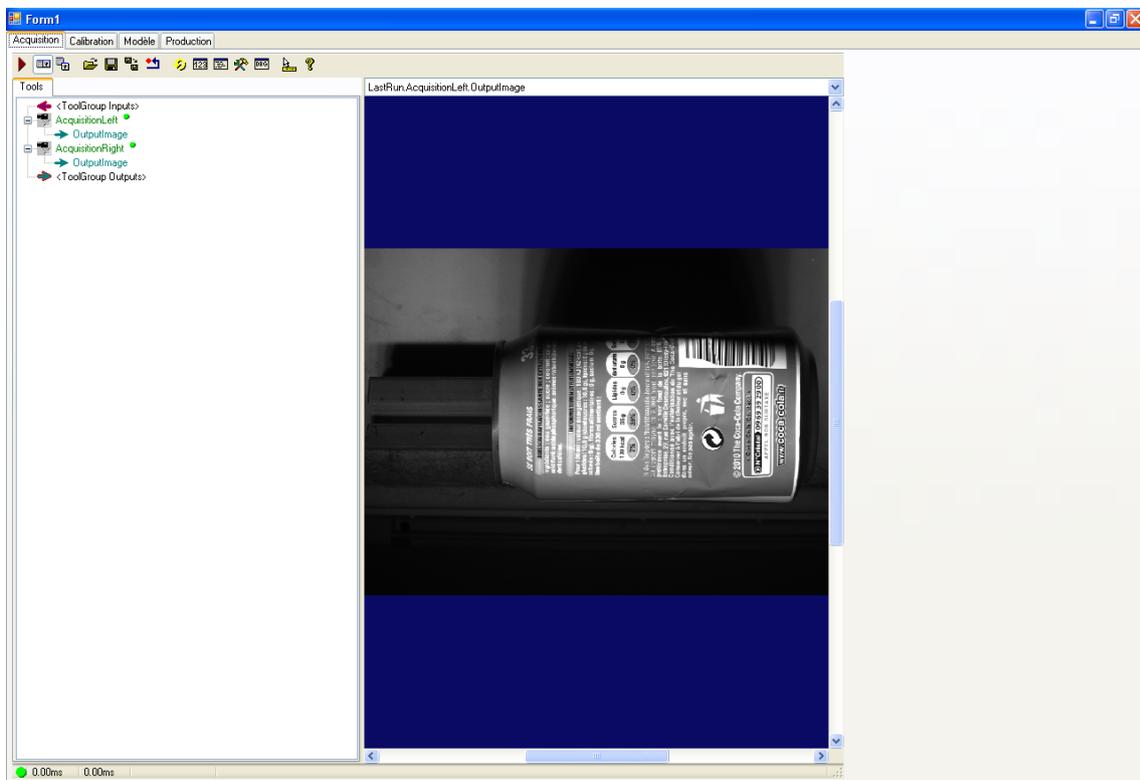


FIGURE 5.15 – IHM : Programme d'acquisition des images des deux caméras

5.4.2 Phase de calibration

La calibration est réalisée à partir de mires acquises au même instant par les deux caméras afin de définir un ensemble de points correspondants dans les deux images. Nous avons déplacé la mire sur cinq orientations différentes.

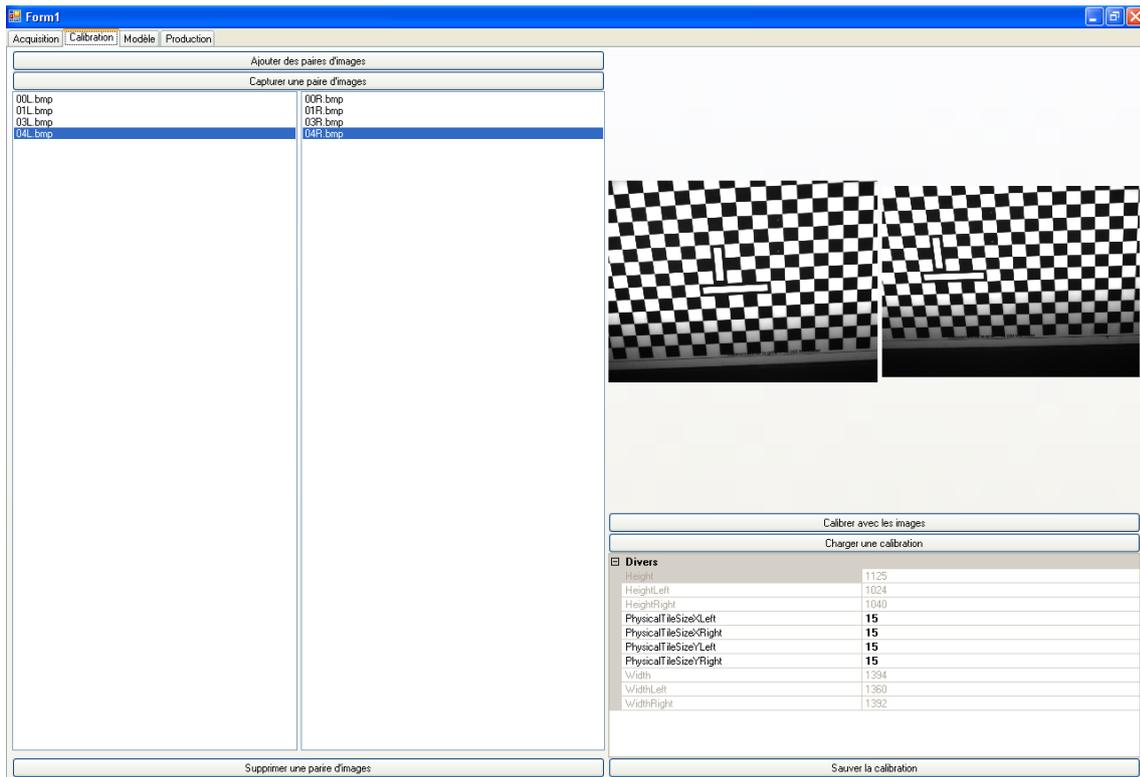


FIGURE 5.16 – IHM : Calibration du système stéréoscopique

La calibration permet de définir les paramètres de la matrice fondamentale et des matrices de rectification pour les deux images. Seules ces dernières sont nécessaires pour enregistrer la calibration.

5.4.3 Analyse de défaut

L'objectif de l'application est de détecter des variations de la surface de la canette. La méthode utilisée est de construire une carte dense de disparité sur une canette-étalon, c'est une canette qui servira de modèle car elle ne présente aucune déformation. En conséquence, pour chaque nouvelle canette analysée, nous construisons une nouvelle carte dense de disparité et la comparons à l'image modèle.

5.4.3.1 Apprentissage du profil

Après avoir acquis les deux images, nous réalisons la rectification et construisons la carte dense de disparité du modèle. Nous paramétrons la carte selon la taille du vecteur noyau, la tolérance, l'intervalle de disparité et le critère de corrélation.

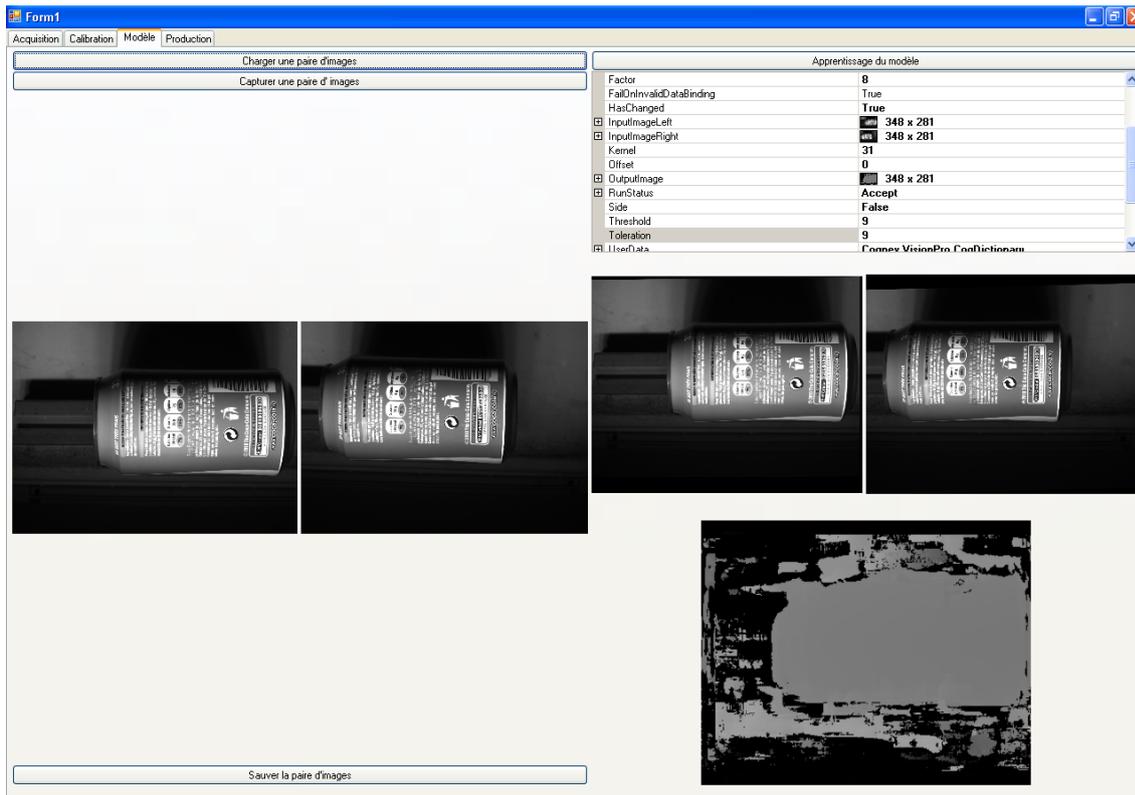


FIGURE 5.17 – IHM : Construction de la carte dense de disparité pour une pièce étalon

5.4.3.2 Comparaison des profils

L'étape de comparaison de la nouvelle carte dense de disparité avec celle du modèle impose que la canette soit indexée, c'est à dire qu'elle soit positionnée de la même manière que le modèle. La comparaison consiste à mesurer l'écart absolu de disparité entre l'image modèle et l'image courante pour l'ensemble des pixel.

Nous observons sur la figure 5.18, la déformation de la canette. L'erreur a été amplifié pour pouvoir visualiser la déformation.

5.4.4 Bilan

Cette méthode est courante en vision industrielle où on réalise une mesure sur une pièce étalon et de comparer chaque nouvelle pièce au modèle.

Cette application a été mise en place comme démonstration utilisant la production de carte dense de disparité pour évaluer la déformation de pièce quelconque. La première contrainte est d'avoir une pièce présentant suffisamment de texture, sans laquelle une projection de mouchetis serait requis. La seconde contrainte est d'avoir un moyen de garantir que les pièces soit placées de la même manière.



FIGURE 5.18 – IHM : Construction de la carte dense de disparité pour la pièce courante et recherche de défaut

Conclusion

Nous avons présenté dans ce chapitre les deux méthodes de reconstruction 3D directe et de production de carte de disparité associé à des applications spécifiques.

Les applications de recalage 3D et de décaissage nécessitent de localiser dans un repère monde pré-défini des points caractéristiques. Cette mesure 3d est réalisée par la méthode de reconstruction 3D de manière précise sous contrainte que nos objets soient stéréovisibles, détectables par des outils 2D précis et que l'on puisse réaliser l'appariement.

Les applications de d'analyse de surface nécessitent de connaître l'état de surface des objets dans l'objectif d'avoir une tendance de la déformation des objets. Cette représentation est réalisée par la méthode de production de carte dense de disparité sous contrainte que la scène soit texturée naturellement ou artificiellement par la projection d'une lumière mouchetée.

Au final, nous montrons que les deux méthodes sont adaptées à de nombreuses applications mais que la maîtrise de l'environnement est primordiale, que ce soit par rapport à la pose des caméras, de l'éclairage, et des pièces à localiser ou analyser. La performance globale est directement liée à ces paramètres.

Chapitre 6

Conclusion

Nous avons présenté dans ce manuscrit nos travaux sur la stéréovision dans un contexte industriel, nous intéressant à deux problématiques, la mesure 3D et le contrôle 3D, basées sur deux approches de la stéréovision, la reconstruction 3D directe et la production de carte dense de disparité.

La reconstruction 3D directe

Concernant la reconstruction 3D directe, nous avons présenté les étapes nécessaires à la constitution d'une mesure 3D à partir de la connaissance deux points image correspondants dans les deux images. La calibration monoculaire de nos caméras, estimée à partir de points de mire non coplanaires, est une étape importante pour définir le modèle de triangulation 3D, basé sur la construction de deux droites dans le repère monde représentant les rayons de projection du point 3D dans les deux images.

L'amélioration de la précision de la mesure 3D repose sur un apprentissage de correspondances entre des positions réelles de l'espace et de positions mesurées par la triangulation 3D, et la construction d'un modèle de régression 3D. Nous avons montré que cette approche nous permet de corriger les erreurs de mesure 3D et de nous abstenir d'estimer les paramètres de distorsion de nos caméras.

La production de carte dense de disparité

La production de carte dense de disparité a été réalisée à partir d'un système stéréo non aligné, nous amenant à estimer la relation géométrique entre nos caméras à partir de la seule connaissance des images. L'utilisation de points de mires permet de calibrer notre système directement.

La rectification, dont nous avons détaillé la construction des matrices basée sur la première partie de travaux de Mallon, permet d'obtenir la contrainte d'alignement des

images afin d'accélérer le processus de mise en correspondance. La minimisation de la déformation projective passe par la contrainte d'avoir un système stéréo au préalable proche de l'alignement, minimisant par la même occasion les erreurs de rectification.

La mise en correspondance passe par des méthodes corrélatives rapides sous la contrainte que notre scène présente des éléments texturés naturellement ou par la projection d'une source lumineuse adaptée. Nous avons optimisé la construction de l'image d'erreur pour chaque disparité par la séparation du calcul en deux étapes, en ligne puis en colonne et par l'utilisation d'instructions « bas-niveau ».

Nous avons montré que le choix de la fenêtre de corrélation avait une influence sur la qualité des cartes de disparité. Nous avons proposé une approche alternative basée sur l'utilisation de noyaux mono-directionnels. Elle permet de conserver la nature des éléments au premier plan.

Les perspectives

La reconstruction 3D directe est valide pour un système stéréo mais pourrait être étendue à des solutions multi-caméras pouvant entraîner une amélioration de la précision 3D mais complexifiant l'appariement des points image. Nous envisagerions, pour un système à N caméras, de construire un ensemble de triangulation 3D pour chaque combinaison de paires de caméras, soit $\frac{N(N-1)}{2}$ calculs 3D. La mesure 3D finale serait le point moyen de l'ensemble des point 3D construits.

La production de carte dense de disparité, destinée aux applications d'analyse de surface, fournit une image dont le niveau de gris est une information de profondeur. Une analyse de cette image requiert la mise en place d'outils de traitement d'image qui pourraient être mis en œuvre pour détecter des variations de la surface des objets. La quantification d'un surplus ou d'un manque matière pourrait être réalisée par la réutilisation des résultats de calibration monoculaire. En effet la connaissance des paramètres des caméras nous donnerait une relation entre les points de l'image de disparité et les points 3D de l'espace.

Chapitre 7

Annexes

7.1 Les capteurs de la technologie 3D

7.1.1 Capteurs de conoscopie holographique

7.1.1.1 Les produits Optimet

Le ConoProbe est un capteur ponctuel en z permettant une fréquence d'acquisition comprise entre 850Hz et 3000Hz. Le système est doté de 31 calibrations pré-stockées permettant de faire varier la distance de mesure. La précision est de l'ordre du micron. Toujours basé sur le principe de la conoscopie holographique, le ConoLine permet d'obtenir un profil 3D sur une ligne de mesure.

La figure 7.1 montrent la robustesse des solutions d'optimet par rapport :

- à la distance de mesure par l'utilisation de lentille adaptées
- à la surface de contact avec la pièce
- à la mesure indirecte par l'utilisation d'un miroir réfléchissant
- à l'angle d'incidence pouvant atteindre 85°

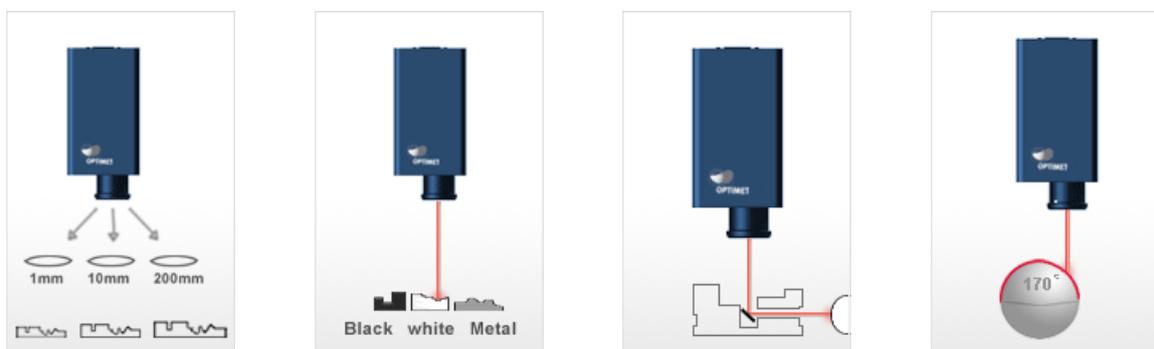


FIGURE 7.1 – Utilisation par déplacement linéaire

7.1.2 Capteurs de triangulation laser

7.1.2.1 Sick Ranger

La caméra 3D Ranger est dotée d'un capteur 1536 pixels par 512 pixels. Elle permet de réaliser simultanément des acquisitions d'images 2D et 3D (MultiScan). Cette caméra acquiert jusqu'à 35 000 profils par seconde, chacun contenant jusqu'à 1536 coordonnées 3D. Cela représente plus de 50 millions de pixels 3D par seconde. Les images en sorties sont au format 8 bits ou 16 bits. Sick propose également un système pré-calibré intégrant directement un laser.



FIGURE 7.2 – Le capteur Sick Ranger

7.1.2.2 Photon Focus

Caméra 3D Photon Focus : Cette caméra fonctionne sur le même principe que la caméra 3D de Sick a une vitesse également très élevée (jusqu'à 150 images par seconde en pleine résolution). Elle est dotée d'un capteur de 1020 pixels par 1024 pixels. Là aussi la caméra permet le MultiScan et le format de sortie est du 8 bits ou du 16 bits.



FIGURE 7.3 – Le capteur Photon Focus

7.1.2.3 Automation Technology

La gamme 3D de Automation Technology a la particularité de proposer des capteurs de haute résolution (jusqu'à 2352 pixels par 1728 pixels) pour des vitesses remarquablement élevées. Le tableau suivant liste des performances pour les 3 caméras de la gamme C3.



FIGURE 7.4 – Le capteur Automation Technology

7.1.3 Capteurs avec projection de franges

7.1.3.1 Vialux Z-Snapper

Ce système de projection de franges permet d'acquérir des images VGA à plus de 40 images par seconde afin d'en extraire un nuage de points 3D.



FIGURE 7.5 – Le capteur Vialux Z-Snapper

7.1.4 Capteurs de stéréovision

7.1.4.1 Point Grey Bumblebee 2

Doté de 2 caméras couleur ou monochromes VGA (640 pixels par 480 pixels) ou X VGA (1024 pixels par 768 pixels), ce système tout intégré et tout calibré permet des fréquences

d'acquisition de 48 images par seconde en VGA et 20 images par seconde en XVGA. A noter qu'un SDK (Software Development Kit) fourni avec le système permet de calculer en temps réel des cartes de disparité avec les images acquises par le système.



FIGURE 7.6 – Le capteur Point Grey Bumblebee 2

7.1.4.2 VisionRT

Le capteur VisionRT est un système de stéréovision intégrant une lumière structurée de type mouchetis permettant de créer de la texture sur la surface analysée et ainsi permet de garantir l'unicité de l'ensemble des points de la surface.

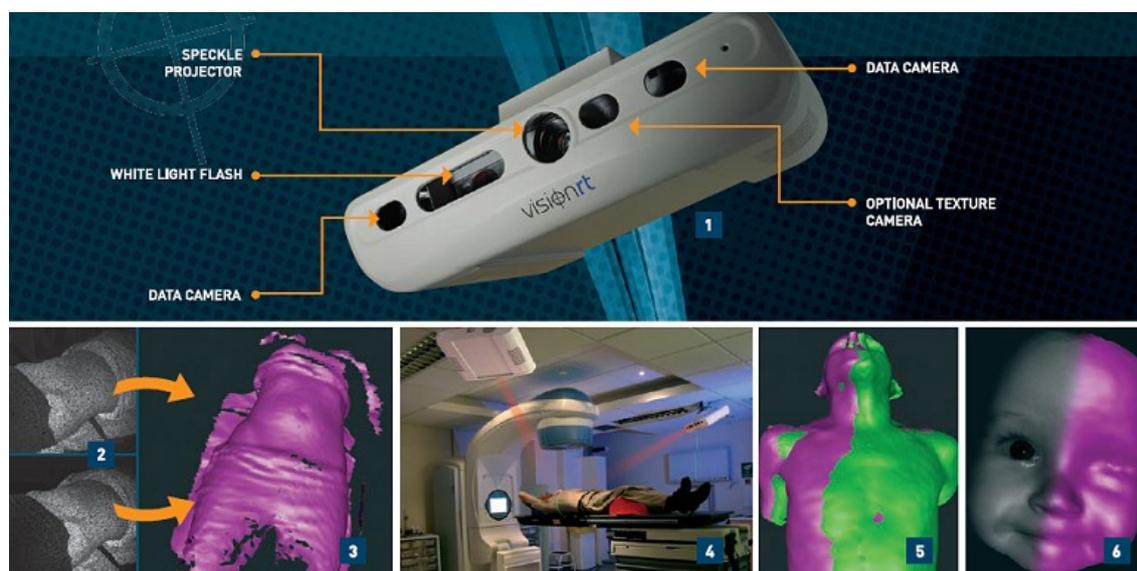


FIGURE 7.7 – Le capteur VisionRT

7.1.5 Capteurs de stéréovision par projection de franges

7.1.5.1 GOM Atos III

Après acquisition des deux images, le système calcule immédiatement les coordonnées 3D de jusqu'à 4 millions de points dans chaque mesure. Du volant jusqu'au véhicule

complet - le scanner mobile "ATOS" 3D de GOM numérise les objets de 10 mm à 10 mètres avec un espacement des points de 0.15 mm à 1 mm. Le système atteint une précision de 0.02 à 0.13 mm (1 :15,000 du volume mesuré). La haute densité de points permet une analyse de CAO détaillée des surfaces, des arêtes et des éléments géométriques. De plus, le logiciel fournit une polygonisation rapide et une exportation complète de données (ex. STL) pour les traitements à la demande, bien adaptés pour la rétro conception.



FIGURE 7.8 – Le capteur GOM Atos III

7.1.6 Capteurs de temps de vol

7.1.6.1 PMD Technologies CamCube

Le capteur CamCube de PMD Technologies existent sous différentes résolutions 200x200, 176x144 et 160x120 permettant respectivement des cadences d'acquisition de 40, 60 et 80 images par seconde. Des travaux récents [MLK08] [BHS08] et [EKB08] mettent en œuvre les capteurs PMD Technologiques pour des applications diverses.

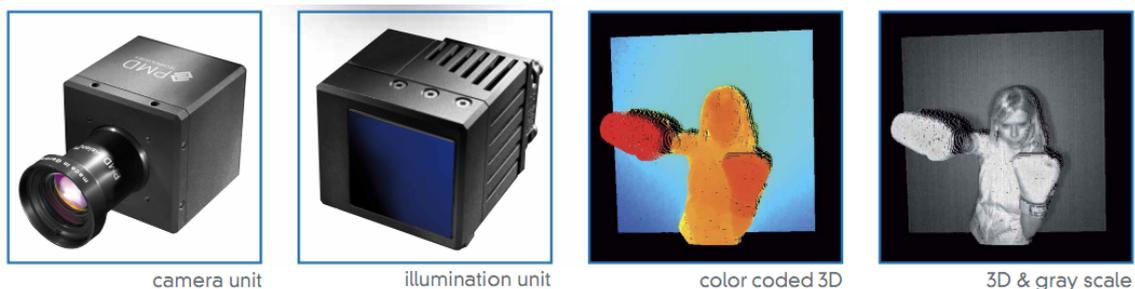


FIGURE 7.9 – Le capteur PMD Technologies CamCube

7.1.6.2 Mesa Imaging SwissRanger

Le SwissRanger SR4000 est un capteur de résolution 176x144 et permettent une fréquence d'acquisition de 50fps. Il est un des capteurs les plus utilisé dans le domaine de la recherche comme dans les travaux [SST08] [YK08] et [YCT10]



FIGURE 7.10 – Le capteur Mesa Imaging SwissRanger

Bibliographie

- [AD04] M Agrawal and L S Davis. Window-based, discontinuity preserving stereo. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2004 CVPR 2004*, 00(C) :66–73, 2004.
- [AS03] Xavier Armangué and Joaquim Salvi. Overall view regarding fundamental matrix estimation. *Image and vision computing*, 21 :205–220, 2003.
- [BBHM03] Myron Z. Brown, Darius Burschka, Gregory D. Hager, and Senior Member. Advances in computational stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25 :993–1008, 2003.
- [Bey92] Horst A Beyer. *Accurate calibration of CCD-cameras*, pages 96–101. IEEE, 1992.
- [BG07] Michael Bleyer and Margrit Gelautz. Graph-cut-based stereo matching using image segmentation with symmetrical treatment of occlusions. *Image Commun.*, 22 :127–143, February 2007.
- [BHM04] Roland Brockers, Marcus Hund, and Bärbel Mertsching. A fast cost relaxation stereo algorithm with occlusion detection for mobile robot applications. In *VMV'04*, pages 47–53, 2004.
- [BHS08] P. Jenke B. Huhle and W. Straßer. On-the-fly scene acquisition with a handy multisensor-system. *Int. J. of Intelligent Systems Technologies and Applications*, 2008.
- [BI12] Atsuhiko Banno and Katsushi Ikeuchi. Estimation of f-matrix and image rectification by double quaternion. *Inf. Sci.*, 183 :140–150, January 2012.
- [BTG06] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. SURF : Speeded Up Robust Features. In *proceedings of European Conference on Computer Vision (ECCV'06), Graz, Austria*, pages 404–417, 2006.
- [BVZ98] Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 648–655, 1998.

- [Can86] J Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8 :679–698, June 1986.
- [CDY⁺08] Haihua Cui, Ning Dai, Tianran Yuan, Xiaosheng Cheng, and Wenhe Liao. Calibration algorithm for structured light 3d vision measuring system. In *Proceedings of the 2008 Congress on Image and Signal Processing, Vol. 2 - Volume 02*, pages 324–328, Washington, DC, USA, 2008. IEEE Computer Society.
- [Cha05] Sylvie Chambon. *Mise en correspondance stéréoscopique d’images couleur en présence d’occultation*. PhD thesis, Université Paul Sabatier, 2005.
- [CJ04] Gustavo Carneiro and Allan D. Jepson. Pruning local feature correspondences using shape context. In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR’04) Volume 3 - Volume 03*, ICPR ’04, pages 16–19, Washington, DC, USA, 2004. IEEE Computer Society.
- [CWC⁺01] Y.-Q. Cheng, X. G. Wang, R. T. Collins, E. M. Riseman, and A. R. Hanson. Three-dimensional reconstruction of points and lines with unknown correspondence across images. *Int. J. Comput. Vision*, 45 :129–156, November 2001.
- [CWT04] Zezhi Chen, Chengke Wu, and Hung Tat Tsui. A new image rectification algorithm. *Pattern Recognition Letters*, 24(1-3) :251–260, 2004.
- [CZZ09] Shanshan Chen, Wuheng Zuo, and Lijun Zheng. Camera calibration via stereo vision using tsai’s method. In *Proceedings of the 2009 First International Workshop on Education Technology and Computer Science - Volume 03*, ETCS ’09, pages 273–277, Washington, DC, USA, 2009. IEEE Computer Society.
- [Dev97] Frédéric Devernay. *Vision stéréoscopique et propriétés différentielles des surfaces*. PhD thesis, Ecole Polytechnique, 1997.
- [DMM04] L. Di Stefano, M. Marchionni, and S. Mattoccia. A fast area-based stereo matching algorithm. *Image and Vision Computing*, 22(12) :983–1005, 2004.
- [EKB08] J. Hornegger E. Kollorz and A. Barke. Gesture recognition with a time-of-flight camera. *Int. J. of Intelligent Systems Technologies and Applications*, 2008.
- [Fau93] Olivier Faugeras. *Three-Dimensional Computer Vision : A geometric Viewpoint*. MIT Press, Cambridge, 1993.

- [FB81] Martin A. Fischler and Robert C. Bolles. Random sample consensus : A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6) :381–395, 1981.
- [Gav98] Didier Gava. *Vision conoscopique 3D : calibration et reconstruction*. PhD thesis, Université René Descartes - Paris V, 1998.
- [GF97] Brigitte Géraud and Gilles Foulon. Une technique de rectification sans contrainte. In *colloque GRETSI, Grenoble, France*, pages 415–418, 1997.
- [GIK03] Abdul Ghafoor, Rao Naveed Iqbal, and Shoab Khan. Image matching using distance transform. In *Proceedings of the 13th Scandinavian conference on Image analysis, SCIA'03*, pages 654–660, Berlin, Heidelberg, 2003. Springer-Verlag.
- [GM04a] Bogdan Georgescu and Peter Meer. Point matching under large image deformations and illumination changes. *IEEE TRANS. PATTERN ANAL. MACHINE INTELL*, 26(6) :674–688, 2004.
- [GM04b] S Gutierrez and J L Marroquin. Robust approach for disparity estimation in stereo vision. *Image and Vision Computing*, 22(3) :183–195, 2004.
- [GNLY10] Hongwei Gao, Ben Niu, Bin Li, and Yang Yu. An improved image rectification algorithm based on particle swarm optimization. In *Proceedings of the 6th international conference on Advanced intelligent computing theories and applications : intelligent computing, ICIC'10*, pages 587–594, Berlin, Heidelberg, 2010. Springer-Verlag.
- [GR10] Sai Siva Gorthi and Pramod Rastogi. Fringe projection techniques : Whether we are? *OPTICS AND LASERS IN ENGINEERING*, 48(2) :133–140, FEB 2010.
- [GY05] Minglun Gong and Yee-Hong Yang. Near real-time reliable stereo matching using programmable graphics hardware. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 924–931, Washington, DC, USA, 2005. IEEE Computer Society.
- [HAH99] R. Hartley, L. Agapito, and E. Hayman. Camera calibration and the search for infinity. In *proceedings of IEEE International Conference on Computer Vision (ICCV'99), Corfu, Greece*, volume 1, pages 510–517, 1999.
- [HH09] Tao Hu and May Huang. A new stereo matching algorithm for binocular vision. In *Proceedings of the 2009 International Conference on Hybrid*

-
- Information Technology*, ICHIT '09, pages 42–44, New York, NY, USA, 2009. ACM.
- [HM95] Radu Horaud and Olivier Monga. *Vision par ordinateur : Outils fondamentaux*. Hermes, 1995.
- [HS88] C. Harris and M. Stephens. A combined corner and edge detector. In *proceedings of 4th Alvey Conference*, pages 147–151, 1988.
- [HS06] Shinichiro Hirooka and Hideo Saito. Calibration free virtual display system using video projector onto real object surface. *IEICE Trans Inf Syst*, E89-D(1) :88–97, 2006.
- [HZ00] Richard I. Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN : 0521623049, 2000.
- [HZW⁺10] Martin Humenberger, Christian Zinner, Michael Weber, Wilfried Kubinger, and Markus Vincze. A fast stereo matching algorithm suitable for embedded real-time systems. *Comput. Vis. Image Underst.*, 114 :1180–1202, November 2010.
- [IG98] H. Ishikawa and D. Geiger. Segmentation by grouping junctions. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 125–131, 1998.
- [IT99] Francesco Isgrò and Emanuele Trucco. Projective rectification without epipolar geometry. In *proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'99), Fort Collins, CO, USA*, pages 1094–1099, 1999.
- [JFB11] Gauri M. Jog, Habib Fathi, and Ioannis Brilakis. Automated computation of the fundamental matrix for vision based construction site applications. *Adv. Eng. Inform.*, 25 :725–735, October 2011.
- [KLCL05] Jae Chul Kim, Kyoung Mu Lee, Byoung Tae Choi, and Sang Uk Lee. A dense stereo matching using two-pass dynamic programming with generalized ground control points. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2 - Volume 02*, CVPR '05, pages 1075–1082, Washington, DC, USA, 2005. IEEE Computer Society.
- [KMPF10] S. Kumar, C. Micheloni, C. Piciarelli, and G. L. Foresti. Stereo rectification of uncalibrated and heterogeneous images. *Pattern Recogn. Lett.*, 31 :1445–1452, August 2010.

- [Kra08] Sébastien Kramm. *Production de cartes éparses de profondeur avec un système de stéréovision embarqué non-aligné*. PhD thesis, Université de Rouen, 2008.
- [KWZK95] S.B. Kang, J. Webb, L. Zitnick, and T. Kanade. A multibaseline stereo system with active illumination and real-time image acquisition. *ICCV*, pages 88–93, 1995.
- [KZ01] V. Kolmogorov and R. Zabih. Visual correspondence with occlusions using graph cuts. *International Conference on Computer Vision*, pages 508–515, 2001.
- [LAC11] Sébastien Lefebvre, Sébastien Ambellouis, and François Cabestaing. A 1d approach to correlation-based stereo matching. *Image and Vision Computing*, 29(9) :580 – 593, 2011.
- [LALS04] Carlos Leung, Ben Appleton, Brian C. Lovell, and Changming Sun. An energy minimisation approach to stereo-temporal dense reconstruction. In *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 4 - Volume 04*, ICPR '04, pages 72–75, Washington, DC, USA, 2004. IEEE Computer Society.
- [LBDLF08] Ricardo Landa-Becerra and Luis G. De La Fraga. Triangulation using differential evolution. In *Proceedings of the 2008 conference on Applications of evolutionary computing, Evo'08*, pages 359–364, Berlin, Heidelberg, 2008. Springer-Verlag.
- [LF96] Quang-Tuan Luong and Olivier Faugeras. The fundamental matrix : theory, algorithms, and stability analysis. *International Journal of Computer Vision*, 17(1) :43–75, 1996.
- [LH81] H.C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(10) :133–135, 1981.
- [LJ11] My-Ha Le and Kang-Hyun Jo. Building detection and 3d reconstruction from two-view of monocular camera. In *Proceedings of the Third international conference on Computational collective intelligence : technologies and applications - Volume Part I, ICCCI'11*, pages 428–437, Berlin, Heidelberg, 2011. Springer-Verlag.
- [Low99] David Lowe. Object recognition from local scale-invariant features. In *proceedings of IEEE International Conference on Computer Vision (ICCV'99), Corfu, Greece*, pages 1150–1157, 1999.
- [LWZ08] Hong Lei, Zhenzhong Wei, and Guangjun Zhang. A simple global calibration method based on 1d target for multi-binocular vision sensor. In

- Proceedings of the 2008 International Symposium on Computer Science and Computational Technology - Volume 02*, pages 290–294, Washington, DC, USA, 2008. IEEE Computer Society.
- [LYX08] Guie Luo, Xinrong Yang, and Qi Xu. Fast stereo matching algorithm using adaptive window. In *Proceedings of the 2008 International Symposiums on Information Processing*, pages 25–30, Washington, DC, USA, 2008. IEEE Computer Society.
- [LZ99] Charles T. Loop and Zhengyou Zhang. Computing rectifying homographies for stereo vision. In *proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'99)*, Fort Collins, CO, USA, pages 1125–1131, 1999.
- [MJYY06] L.B. Meng, G.C. Jin, X.F. Yao, and H.Y. Yeh. 3d full-field deformation monitoring of fiber composite pressure vessel using 3d digital speckle correlation method. *Polymer Testing*, 25(1) :42 – 48, 2006.
- [MLGM09] Zhang Mandun, Qi Lichao, Cui Guodong, and Yu Ming. A triangulation method in 3d reconstruction from image sequences. In *Proceedings of the 2009 Second International Conference on Intelligent Networks and Intelligent Systems*, ICINIS '09, pages 306–308, Washington, DC, USA, 2009. IEEE Computer Society.
- [MLK08] Martin Lambers Marvin Lindner and Andreas Kolb. Sub-pixel data fusion and edge-enhanced distance refinement for 2d/3d images. *Int. J. of Intelligent Systems Technologies and Applications*, 2008.
- [MMHM02] K. Mühlmann, D. Maier, J. Hesser, and R. Männer. Calculating dense disparity maps from color stereo images, an efficient implementation. *IJCV*, 47(1-3) :79–88, 2002.
- [MS04] Lionel Moisan and Bérenger Stival. A probabilistic criterion to detect rigid point matches between two images and estimate the fundamental matrix. *Int. J. Comput. Vision*, 57 :201–218, May 2004.
- [MW05] J. Mallon and P.F. Whelan. Projective rectification from the fundamental matrix. *Image and Vision Computing*, 23(7) :643–650, 2005.
- [Ora01] Daniel Oram. Rectification for any epipolar geometry. In *proceedings of British Machine Vision Conference (BMVC'01)*, 2001.
- [PD96] D. V. Papadimitriou and T. J. Dennis. Epipolar line estimation and rectification for stereo image pairs. *IEEE Transactions on Image Processing*, 5(4), 1996.

- [PKB11] Jimmy Pelcat, Sebastien Kramm, and Abdelaziz Bensrhair. Production de carte dense de disparité dans un contexte industriel par stéréo-corrélation d'images sur deux noyaux mono-directionnels. In *ORASIS - Congrès des jeunes chercheurs en vision par ordinateur*, Praz-sur-Arly, France, 2011. INRIA Grenoble Rhône-Alpes.
- [PKBS10] Jimmy Pelcat, Sebastien Kramm, Abdelaziz Bensrhair, and Florian Sella. Research and development of stereovision techniques within an industrial environment. July 2010.
- [PKV99] Marc Pollefeys, R. Koch, and L. Van Gool. A simple and efficient rectification method for general motion. In *proceedings of IEEE International Conference on Computer Vision (ICCV'99), Corfu, Greece*, volume 1, pages 496–501, 1999.
- [PV09] J. Pelcat and Visionic. Application 3d de guidage robot. 3ème édition du salon des équipements et des solutions en vision, Octobre 2009.
- [RKS96] V. Rodehorst, A. Koschan, and K. Spiller. Color stereo vision using hierarchical block matching and active color illumination. *International Conference on Pattern Recognition*, pages 835–839, 1996.
- [RMC97] Sebastien Roy, Jean Meunier, and Ingemar J. Cox. Cylindrical rectification to minimize epipolar distortion. In *proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'97), Puerto Rico*, pages 393–399, june 1997.
- [Roy99] S. Roy. Stereo without epipolar lines : A maximum flow formulation. *International Journal of Computer Vision*, 1(2) :1–15, 1999.
- [SB95] S. M. Smith and J. M. Brady. Susan - a new approach to low level image processing. *International Journal of Computer Vision*, 23 :45–78, 1995.
- [SC92] Jun Shen and Serge Castan. An optimal linear operator for step edge detection. *CVGIP : Graphical Model and Image Processing*, 54(2) :112–133, 1992.
- [SK07] Yasuyuki Sugaya and Kenichi Kanatani. Highest accuracy fundamental matrix computation. In *Proceedings of the 8th Asian conference on Computer vision - Volume Part II, ACCV'07*, pages 311–321, Berlin, Heidelberg, 2007. Springer-Verlag.
- [SLKS05] Jian Sun, Yin Li, Sing Bing Kang, and Heung-Yeung Shum. Symmetric stereo matching for occlusion handling. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recog-*

- nition (CVPR'05) - Volume 2 - Volume 02*, CVPR '05, pages 399–406, Washington, DC, USA, 2005. IEEE Computer Society.
- [SMM04] Luigi Di Stefano, Massimiliano Marchionni, and Stefano Mattocchia. A fast area-based stereo matching algorithm. *Image and Vision Computing*, 22(12) :983 – 1005, 2004. <ce :title>Proceedings from the 15th International Conference on Vision Interface</ce :title>.
- [SS02] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1) :7-42, 2002.
- [SS03] D. Scharstein and R. Szelisk. High-accuracy stereo depth maps using structured light. *CVPR*, 1 :195–202, 2003.
- [SST08] J. Davis S. Schuon, C. Theobalt and S. Thrun. High-quality scanning using time-of-flight depth superresolution. *IEEE CVPR Workshop on Time-Of-Flight Computer Vision*, 2008.
- [ST94] Jiambo Shi and Carlo Tomasi. Good features to track. In *proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94), Seattle, WA, USA*, pages 593–600, 1994.
- [Ste99] Charles V. Stewart. Robust parameter estimation in computer vision. *SIAM Review*, 41(3) :513–537, 1999.
- [Stu97] Peter Sturm. *Vision 3d non calibrée : contribution à la reconstruction projective et étude des mouvements critiques pour l'auto-calibrage*. PhD thesis, Institut National Polytechnique De Grenoble, France, 1997.
- [TM97] P.H.S. Torr and D.W. Murray. The development and comparison of robust methods for estimating the fundamental matrix. *International Journal of Computer Vision*, 24 :271–300, 1997. 10.1023/A :1007927408552.
- [Tsa87] R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *International Journal of Robotics and Automation*, 3(4) :323–344, 1987.
- [Vek03] O. Veksler. Fast variable window for stereo correspondence using integral images. *CVPR*, pages 556–561, 2003.
- [VSCCTPF07] Israel Vite-Silva, Nareli Cruz-Cortés, Gregorio Toscano-Pulido, and Luis Gerardo Fraga. Optimal triangulation in 3d computer vision using a multi-objective evolutionary algorithm. In *Proceedings of the 2007 EvoWorkshops 2007 on EvoCoMnet, EvoFIN, EvoIASP, EvoINTERACTION, EvoMUSART, EvoSTOC and EvoTransLog : Applications of Evolutionary Computing*, pages 330–339, Berlin, Heidelberg, 2007. Springer-Verlag.

- [WC09] Qizhi Wang and Xinyu Cheng. The simple camera calibration approach based on a triangle and depth estimation from monocular vision. In *Proceedings of the 2009 IEEE/RSJ international conference on Intelligent robots and systems, IROS'09*, pages 316–320, Piscataway, NJ, USA, 2009. IEEE Press.
- [WJQW10] Ruiyan Wang, Guang Jiang, Long Quan, and Chengke Wu. Camera calibration using lengths of corresponding line segments. In *Proceedings of the 2010 Fourth Pacific-Rim Symposium on Image and Video Technology, PSIVT '10*, pages 162–167, Washington, DC, USA, 2010. IEEE Computer Society.
- [WQ04] Yichen Wei and Long Quan. Region-based progressive stereo matching. In *CVPR (1)'04*, pages 106–113, 2004.
- [WY05] Hsien-Huang P. Wu and Yu-Hua Yu. Projective rectification with reduced geometric distortion for stereo vision and stereoscopic video. *J. Intell. Robotics Syst.*, 42 :71–94, January 2005.
- [YCT10] D. Chan S. Thrun Y. Cui, S. Schuon and C. Theobalt. 3d shape scanning with a time-of-flight camera. *CVPR*, 2010.
- [YK08] C. Theobalt S. Thrun Y.M. Kim, D. Chan. Design and calibration of a multi-view tof sensor fusion system. *CVPR Workshop on Time-of-flight Computer Vision*, 2008.
- [Zha98] Zhengyou Zhang. Determining the epipolar geometry and its uncertainty : A review. *International Journal of Computer Vision*, 27(2) :161–195, 1998.
- [Zha00] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Pattern Analysis and Machine Intelligence*, 22(11) :1330–1334, 2000.
- [ZHZH07] Jing Zhou, Yingming Hao, Feng Zhu, and Lei He. The effect of camera calibration space on visual pose's precision. In *Proceedings of the Third International Conference on Natural Computation - Volume 02, ICNC '07*, pages 194–198, Washington, DC, USA, 2007. IEEE Computer Society.
- [ZPF06] H. X. Zhong, Y. J. Pang, and Y. P. Feng. A new approach to estimating fundamental matrix. *Image Vision Comput.*, 24 :56–60, January 2006.