



HAL
open science

Approche coopérative pour l'acquisition et l'observation de formes tridimensionnelles

Thierry Molinier

► **To cite this version:**

Thierry Molinier. Approche coopérative pour l'acquisition et l'observation de formes tridimensionnelles. Modélisation et simulation. Université de Bourgogne, 2009. Français. NNT: . tel-00465787

HAL Id: tel-00465787

<https://theses.hal.science/tel-00465787>

Submitted on 21 Mar 2010

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

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

UNIVERSITÉ DE BOURGOGNE
ÉCOLE DOCTORALE ENVIRONNEMENT - SANTÉ / STIC (E2S)

THÈSE

présentée par

Thierry MOLINIER

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ

discipline Instrumentation et Informatique de l'Image

**APPROCHE COOPÉRATIVE
POUR L'ACQUISITION ET L'OBSERVATION
DE FORMES TRIDIMENSIONNELLES**

Soutenue le 1er juillet 2009

JURY

Malik MALLEM	Professeur	Evry	Rapporteur
Christophe DOIGNON	MCF HDR	Strasbourg	Rapporteur
Joaquim SALVI	Professeur	Girona	Examineur
Andrea FUSIELLO	Professeur	Verona	Examineur
David FOFI	MCF HDR	Le Creusot	Encadrant
Patrick GORRIA	Professeur	Le Creusot	Directeur de thèse
Fabrice MERIAUDEAU	Professeur	Le Creusot	Président

Résumé

Les progrès technologiques ont permis d'abaisser la frontière entre image réelle et image virtuelle à tel point que les différencier devient difficile. Si la réalité virtuelle évolue au rythme des progrès technologiques, la réalité augmentée, passage du virtuel vers le réel, n'en est qu'à ses débuts. Le but de cette thèse est d'assurer la transition entre le monde virtuel et réel grâce à un système procam (projecteur caméra) pour améliorer l'apparence visuelle d'un objet. La position de l'objet est estimée en projetant une mire codée pour déterminer des points de correspondance entre les caméras et le projecteur sur un objet sans texture (les détecteurs classiques de points d'intérêt sont inefficaces sur de tels objets). Nous combinons l'étalonnage du système et ces points pour estimer la position 3D de l'objet, et ensuite le recalons avec un modèle de points plus fin, issu d'un scanner ou d'un modèle de CAO. La synthèse de vue est réalisée et l'image synthétisée est projetée sur l'objet. De tels outils permettent de générer des textures virtuelles, temporaires (il suffit d'éteindre le projecteur pour enlever la texture de l'objet), qui peuvent décrire l'usure de la texture ou l'objet sous son apparence d'origine.

Mots-clefs : Lumière structurée, étalonnage, reconstruction 3D, projection de texture, réalité augmentée

Abstract

Technologic progres can reduce the barrier between real image and virtual image so far it is really difficult to recognize them. When virtual reality is well done, augmented reality, from virtual world to real world, is still developped. We want to project a virtual texture onto a 3D real object. By using cameras and projector, we estimate its real pose in the virtual reference by projecting a coded pattern to find points of correspondence between cameras and projector onto the textureless object, where common detector of points of correspondence are useless. We combine the calibration of the system and these points to estimate the 3D position of the object, and then register with a fine model issued from scanners. Once the registration step is done, the view synthesis can be performed and then this virtual image is projected adequately onto the real object. Such technologies can synthesize virtual and temporary textures suited to real objects without any degradation. Such textures can be easily modified, and can describe the deterioration of texture along the time or show real objects in their original state.

Keywords : Structured light, calibration, 3D reconstruction, projective texture, augmented reality

Remerciements

Ces travaux de thèse ont été réalisés au sein du Laboratoire d'Électronique, Informatique et Image (Le2i), Unité Mixte de Recherche (UMR CNRS 5158), Université de Bourgogne, et pendant six mois au sein du Laboratoire Computer Vision and Robotic (VICOROB), unité 3D perception lab, Université de Girona, Espagne.

J'aimerais en premier lieu remercier mes parents, ma famille qui m'ont soutenu tout le long de mes études.

Je tiens également à remercier Messieurs Patrick Gorria, David Fofi et Joaquim Salvi pour leurs conseils avisés et leurs nombreuses contributions à l'ensemble de ces travaux. Je tiens à leur témoigner ici le plaisir que j'ai eu à travailler avec eux pendant ces trois années, que ce soit au Laboratoire Le2i au Creusot ou au laboratoire VICOROB, à Girona, Catalogne Espagne.

Je remercie Messieurs Fabrice Meriaudeau, directeur de l'équipe du Creusot, Jean Marie Bilbao, directeur du Le2i et ainsi que les membres du laboratoire Vicorob, pour m'avoir accueilli dans leurs laboratoires.

Je tiens à remercier toutes les personnes qui ont contribué de près ou de loin à cette expérience tant scientifique qu'humaine : Carles, tant pour m'avoir supporté sur les deux sites que pour les discussions sur le recalage 3D, Alexandra pour sa compréhension de Labview, Marina, Birgit, François pour m'avoir appris les rudiments du catalan et de jeu de cartes, Yohan pour ce surnom qui me poursuit toujours, et tous les thésards que j'ai croisé, de Gérone au Creusot, Benjamin, Thomas, Youssef, Mathias, Nico, Pierre Emmanuel...

TABLE DES MATIÈRES

1	Introduction	1
1.1	Motivations	2
1.2	Contributions	3
1.3	Organisation du document	4
2	État de l’art sur les systèmes Projecteur-Caméra	6
2.1	Introduction	7
2.2	Plaquage de Texture	8
2.2.1	Texture 2D	8
2.2.2	Texture 3D	9
2.2.2.1	Textures solides	9
2.2.2.2	Textures géométriques	10
2.3	Correction géométrique	11
2.3.1	Caractéristiques de la caméra et du projecteur	12
2.3.2	Projection sur un plan	13
2.4	Correction visuelle	15
2.4.1	Multi projecteur	15
2.4.2	Projection sur des Objets réels	18
2.4.3	Projection temps réel	21
2.5	Synthèse	23
2.5.1	Les projections sur des plans	23
2.5.2	Les projections sur des formes tridimensionnelles	24
2.5.3	Suggestion	25
3	Géométrie et Étalonnage d’un Système Projecteur-Caméra	26
3.1	Étalonnage	27
3.1.1	Modèle / Système	27
3.1.1.1	Le modèle linéaire	27
3.1.1.2	Distorsion	29
3.1.2	Caméra	30

3.1.3	Estimation de l'homographie entre le plan caméra et la mire	31
3.1.4	Contraintes sur les paramètres intrinsèques	32
3.1.5	Étalonnage de la caméra	33
3.1.5.1	Initialisation	33
3.1.5.2	Optimisation non linéaire	35
3.1.6	Stéréo-étalonnage	36
3.1.7	Projecteur	36
3.1.7.1	Méthode de Faugeras	37
3.1.7.2	Optimisation	40
3.1.7.3	Stéréovision ou triangulation	41
3.1.7.4	Extraction des points d'étalonnage du projecteur	42
3.1.8	Système à n caméra(s) et m projecteur(s)	44
3.2	Géométrie épipolaire	45
3.2.1	Théorie	45
3.2.2	Géométrie épipolaire pour 2 vues : la matrice fondamentale	47
3.2.2.1	Estimation de matrices fondamentales	49
3.2.3	Géométrie trifocale pour trois vues : le tenseur trifocal	51
3.2.4	Relations entre le tenseur trifocal et la matrice fondamentale	52
3.3	Précision du système	53
3.3.1	Synthèse de vue	55
3.4	Synthèse	57
4	Lumière Structurée et Synthèse de vue	58
4.1	Lumière structurée	58
4.1.1	Mire Binaire	59
4.1.1.1	Construction de la mire	61
4.1.1.2	Points de correspondance	63
4.1.1.3	Les points aberrants	64
4.1.2	Mire Couleur	65
4.1.2.1	Séquence de De Bruijn	66
4.1.2.2	Stratégie de codage	67
4.1.2.3	Décodage et Points de correspondance	69
4.2	Synthèse de vue	73
4.2.1	Reconstruction complète du modèle de référence	74
4.2.2	Projection de texture sur un objet blanc	76
4.2.2.1	Recalage	76
4.2.2.1.1	Position initiale	76
4.2.2.1.2	Recalage d'une vue sur le modèle reconstruit	77
4.2.2.2	Synthèse de vue	81

4.2.2.3	Texturation	82
4.3	Synthèse	83
5	Résultats expérimentaux	84
5.1	Système étalonné	84
5.1.1	Reconstruction 3D	85
5.1.2	Synthèse de vue	86
5.1.2.1	Synthèse de la mire	87
5.1.2.2	Erreur de rétroprojection	87
5.1.3	Amélioration	88
5.1.4	Projection sur objet réel	89
5.2	Lumière Structurée	91
5.2.1	Résultats quantitatifs	91
5.2.1.1	Résultats dans le monde réel	91
5.2.1.2	Résultats dans le monde virtuel	93
5.2.1.3	Résultats qualitatifs	94
5.2.1.4	Comparaison de reconstruction 3D suivant la mire de lumière structurée	98
5.3	Projection de texture	99
5.3.1	Reproduction	100
5.3.2	Reproduction complète	101
5.3.3	Changement de couleur	104
5.3.4	Surimpression : altération de texture d'un objet	105
5.3.5	Autres modèles	106
5.3.5.1	Vue des 3 quarts	110
5.3.5.2	Erreur de recalage	111
5.4	Synthèse	111
6	Conclusions et perspectives	112
6.1	Conclusion	112
6.2	Perspectives	113
6.3	Publications	114
6.4	Collaboration scientifique	115
	Bibliographie	116
	Table des figures	124
	Liste des tableaux	129

A	Vision active	131
A.1	Mire uniforme	131
A.2	Mire intégrée	132
B	Mosaïque des synthèses de vues et objet réel texturé pour le modèle Garfield	135

Chapitre 1

Introduction

Du premier film des frères Lumière en noir et blanc au dernier film hollywoodien rempli d'effets spéciaux, la projection de films au cinéma a vu la qualité des images présentées s'améliorer au fil du temps. Avec l'émergence de l'informatique, la notion d'image virtuelle a fait son apparition. Les progrès hardware et software en informatique permettent de générer des images de synthèse en réalité virtuelle d'une qualité telle qu'il devient difficile de différencier image réelle et image virtuelle à l'intérieur d'un écran d'ordinateur ou d'un écran de cinéma. Ce désir de s'affranchir de la barrière entre réel et virtuel a abouti à la fusion des deux types d'images. Une fois la technique des images 2D maîtrisée, l'attention s'est portée sur les images 3D. L'essor de l'informatique a permis de générer des images de plus en plus complexes jusqu'à la maîtrise de cette nouvelle technologie. L'étape suivante a consisté à s'adapter au support physique et ainsi dépasser la projection sur un support plan ou légèrement incurvé.

Projeter une image 2D virtuelle sur un objet 3D réel requiert de créer un lien physique entre le réel et le virtuel. Ce besoin se fait sentir dans l'animation virtuelle, comme les spectacles de son et lumière, l'illumination de monuments avec des images. On peut citer l'exemple de la polychromie de la cathédrale d'Amiens (figure 1.1).

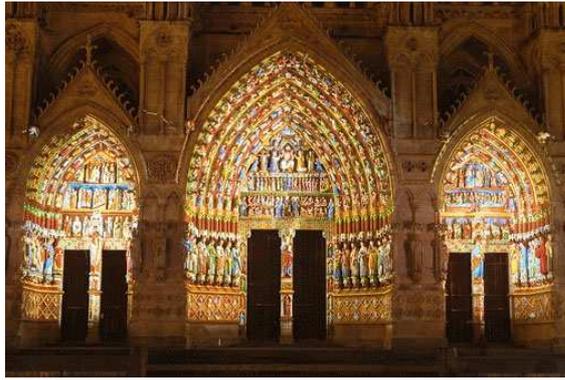


Figure 1.1 – Polychromie du portail de la cathédrale d’Amiens

Si la reproduction de formes géométriques est maîtrisée au travers de techniques comme la stéréolithographie ou l’usinage, ces techniques se concentrent uniquement sur le rendu géométrique et non sur l’aspect esthétique. Si les reproductions d’objets réels sont texturées, toute modification de la texture reste lourde : il nécessite soit de nettoyer l’objet avant d’appliquer une nouvelle texture, soit de commander une nouvelle reproduction. Caméra et projecteur permettent de passer d’un monde à l’autre, du support numérique au monde réel et donc d’appliquer une texture de manière dynamique et flexible.

1.1 Motivations

Dans un premier temps, nous nous intéressons aux outils que nous allons utiliser, caméras et projecteurs, et plus précisément à leur étalonnage. La connaissance de leurs paramètres intrinsèques (ou internes, comme le focus, la distorsion ...) et extrinsèques (position et orientation) vont nous assurer une maîtrise sur notre système pour passer des images au monde réel.

Une fois le système étalonné et ainsi les images contrôlées, notre attention se porte sur l’objet. Estimer sa position par rapport au projecteur grâce au système requiert l’emploi de technique de lumière structurée. Cette technique consiste à projeter une image codée sur l’objet et ensuite de décoder l’image perçue. Elle permet de positionner l’objet dans l’espace virtuel du système. L’absence d’informations sur l’objet qui est sans texture pousse à l’emploi de techniques actives [Salvi et al., 2004].

L’étalonnage du système et l’estimation de la position de l’objet dans l’espace virtuel du projecteur permettent d’estimer la position de l’objet dans le plan image du projecteur. On applique une texture sur l’objet et synthétise l’image de l’objet "vue" par le projecteur. La rétroprojection de cette image dans le monde réel constitue la dernière étape.

1.2 Contributions

Les contributions de nos travaux dans le domaine de la réalité augmentée sont :

- **Étalonnage d'un système de caméras et projecteur** : Nous proposons une méthode pour étalonner deux caméras et un projecteur décomposée en deux étapes :
 1. le stéréo étalonnage des caméras (avec estimation de la distorsion).
 2. l'étalonnage du projecteur par triangulation de mires projetées. Cette méthode peut être étendue à un système comportant plus de caméras et projecteurs. Le nombre minimum de caméras pour utiliser notre méthode étant de deux caméras auxquelles s'ajoute un projecteur.

D'autres systèmes ProCam existent mais ne comportent qu'une seule caméra. L'étalonnage nécessite alors d'utiliser une information extérieure supplémentaire pour dépasser une reconstruction à l'échelle [Raskar and Beardsley, 2001], en faisant coïncider des marqueurs physiques et virtuels [Hamada and Sato, 2007], en utilisant des surfaces arbitraires (planes ou incurvées)...

- **Création d'une mire codée et son décodage** : La création automatique, pseudo-aléatoire d'une mire codée est ensuite proposée. La caractéristique de cette mire réside dans l'unicité de tout motif présent dans l'image. Ce motif est composé d'un ensemble de cinq éléments sur cinq, unique et de couleur binaire. Ce chiffre de cinq éléments code toute l'image projetée par le projecteur. Ces motifs peuvent se chevaucher et respecter la contrainte d'unicité. Cette mire permet d'estimer des points de correspondance entre deux images : identifier ces points revient à identifier un même motif sur les deux images.
- **Synthèse de vue et projection** : La contribution majeure réside dans la méthode globale proposée sans intervention humaine qui consiste à poser un objet devant un projecteur, projeter une mire pour déterminer la position de l'objet par rapport au projecteur de synthétiser une image qui se projette de manière adéquate sur cet objet réel. Notre contribution méthodologique consiste en l'intégration d'outils de vision adaptés à la projection de texture virtuelle sur un objet réel.

1.3 Organisation du document

Le chapitre 2 expose l'état de l'art des systèmes projecteur-caméra (ou ProCam) dévolus à la texturation virtuelle d'objets réels. Cette méthode de texturation, appelée texturation projective, n'est qu'une partie des techniques de texturation qui ont été développées. Nous aborderons ensuite les possibilités que peuvent apporter de tels systèmes, de la correction géométrique à la correction visuelle, ainsi que les différents systèmes développés au cours des dix dernières années.

Le chapitre 3 présente le système ProCam utilisé, deux caméras et un projecteur, et définit les paramètres nécessaires pour étalonner ce système. Cet étalonnage se décompose en deux étapes successives : la première étape consiste à étalonner une ou plusieurs caméras à l'aide d'une mire et d'estimer plusieurs homographies entre le plan de la mire et le plan caméra. La seconde étape est l'étalonnage du projecteur, effectué par triangulation de points projetés par le projecteur et par la mise en correspondance 3D / 2D. Une fois l'étalonnage fort du système complet (ou *hard calibration*, tous les paramètres sont connus), la géométrie épipolaire du système est estimée et les premiers résultats de synthèse de vue calculés.

Le chapitre 4 décrit l'estimation de la position relative de l'objet par rapport à notre système étalonné. Pour cela deux techniques de lumière structurée sont exposées, l'une basée sur le codage binaire et l'autre sur une séquence de couleur. Ce code caché dans la mire virtuelle est appliqué par le projecteur sur l'objet et le décodage des images perçues par plusieurs caméras permet de reconstruire la surface de l'objet. Afin d'obtenir une image de meilleure qualité, on recalcule un modèle 3D plus fin sur notre reconstruction précédente en utilisant des techniques dérivées de l'ICP (*Iterative Closest Point*). La projection de ce modèle recalé sur le plan image du projecteur permet alors de synthétiser l'image vue par une caméra placée au même endroit que le projecteur et ayant les mêmes paramètres. Cette caméra montre l'image "vue" par le projecteur.

Le chapitre 5 expose les différents résultats obtenus. Résultats quantitatifs et qualitatifs sont calculés pour estimer la précision de chaque étape de la méthode, de l'étalonnage du système au recalage de nuages de points pour parvenir à la synthèse de vue désirée. L'image de synthèse ainsi générée dépend de la qualité de l'objet 3D qui est rétroprojeté sur le plan image du projecteur. Un modèle fin dont le maillage respecte les singularités de la surface est alors nécessaire pour texturer certaines zones particulières de l'objet et la mettre en évidence. Ce modèle fin est obtenu soit par connaissance antérieure, tel que le modèle CAO du modèle dans l'industrie, soit par acquisition par un scanner.

Nous présentons au cours du chapitre 6 nos conclusions et les perspectives liées à un tel système.

Chapitre 2

État de l'art sur les systèmes Projecteur-Caméra

Les progrès en réalité virtuelle ont permis de modéliser une scène réelle sur un écran d'ordinateur à tel point que distinguer réalité et virtualité devient presque impossible, en respectant les contraintes du monde réel. Après la représentation des formes tridimensionnelles, le rendu a été la source d'une grande attention pour obtenir une apparence réaliste. La projection de texture sur des objets dans le monde virtuel est désormais possible grâce à des bibliothèques 3D mais ces progrès restent confinés dans ce monde. L'apparition de salles immersives de réalité, telles les systèmes CAVE (*Cave Automatic Virtual Environment*) [Cruz-Neira et al., 1992] développés dès le début des années quatre-vingt-dix, permettent une première immersion dans la réalité virtuelle. L'utilisateur se retrouve plongé dans une pièce où les murs servent de support d'écran. Dès la fin des années quatre-vingt-dix, les applications de projection dynamique de texture dans le monde réel sont envisagées [Raskar et al., 1998] [Underkoffler et al., 1999] avec l'apparition du couple projecteur-caméra.

2.1 Introduction

Réalité virtuelle et réalité augmentée sont deux approches qui méritent d'être distinguées :

- La réalité virtuelle immerge l'utilisateur dans un univers virtuel basé sur un monde créé sur écran d'ordinateur. La réalité augmentée superpose au monde réel des images de synthèse.
- La réalité augmentée, AR ou RA comme indiquée sur la figure 2.1, est une discipline en plein essor [Azuma, 1995],[Bimber and Raskar, 2005]. La reproduction réaliste de l'espace qui nous entoure nécessite des outils toujours plus performants. La réalité augmentée consiste à prendre un environnement réel comme les salles immersives ou les simulateurs et à y ajouter une information supplémentaire. Le but ultime de tel système est d'arriver à un résultat tel que séparer le virtuel du réel ne sera plus possible.

Milgram introduit le concept de réalité mixte [Milgram et al., 1994], entre la réalité virtuelle et la réalité augmentée (figure 2.1). Il introduit ce terme pour identifier les systèmes qui sont principalement synthétiques avec des images réelles comme des textures réelles projetées sur des objets virtuels, introduction suivi d'une taxonomie.

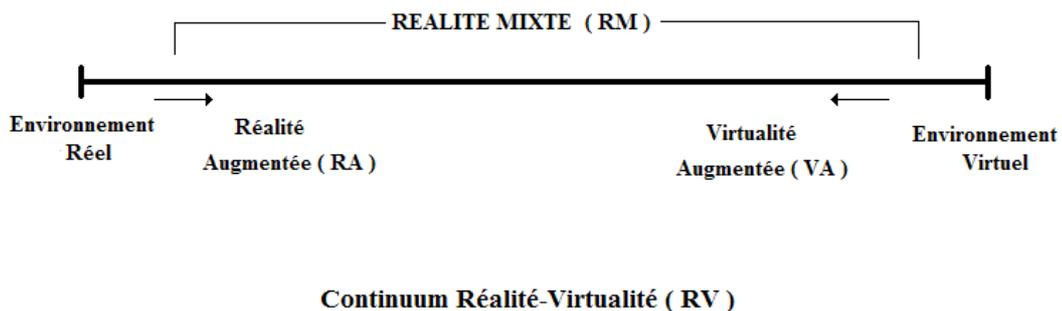


Figure 2.1 – Continuum Réel Virtuel [Milgram et al., 1994]

Trois concepts sont présentés :

- Fidélité de la reproduction.
- Extension de la métaphore de la présence.
- Extension de la connaissance du monde.

Le premier concept se réfère à la qualité de la reproduction, qui peut aller de l'image binaire grossière aux images qui comptent plusieurs millions de couleurs et que l'on qualifie de réalistes. Le deuxième concept prend en compte l'immersion de l'utilisateur dans le système : il se retrouve plongé dans un univers d'images virtuelles. Enfin le dernière concept se réfère à la

contrainte pour la réalité augmentée de ne pas se contenter de la surimposition d'image virtuelle sur une scène réelle, mais de s'assurer aussi de la justesse du recalage sur cette scène.

Les trois concepts présentés par Milgram nous permettent d'introduire le plan de ce chapitre : Le plaquage de texture, la correction géométrique et la correction visuelle.

2.2 Plaquage de Texture

2.2.1 Texture 2D

Le plaquage de texture [Heckbert, 1986] est la méthode pour obtenir un rendu réaliste de la surface de l'objet quand les détails de cette surface sont trop complexes pour être modélisés directement. La première approche pour appliquer une texture sur une surface 3D a été de plaquer directement une texture 2D sur cette surface. La définition de texture est très large : elle implique sa définition basique (apparence tissu, bois, pierre ...), un motif finement détaillé répété sur l'image ou encore une image multidimensionnelle qui est plaquée dans un espace multidimensionnel.

Nous citons deux techniques principales de texture 2D : le *projective mapping* et le *bump mapping*.

La technique de *projective mapping* a été originalement créée pour simuler les effets de lumière et d'ombre [Segal et al., 1992] mais cette technique a été très rapidement généralisée dans le domaine du rendu d'image car elle permet de synthétiser la vue prise par un appareil photographique. A cette fin Segal utilise quatre systèmes de coordonnées :

- un système de coordonnées homogènes (x, y, z, w) centré sur le point de vue.
- les coordonnées 2D de l'écran, obtenues suivant la relation $x^s = \frac{x}{w}$ et $y^s = \frac{y}{w}$.
- un système de coordonnées homogènes (x^l, y^l, z^l, w^l) , centré sur la source de lumière.
- les coordonnées de texture, représentées par : $x^t = \frac{x^l}{w^l}$ et $y^t = \frac{y^l}{w^l}$

En combinant ces coordonnées et en utilisant des contraintes pour déterminer si la surface est visible, masquée ou illuminée, il peut reconstruire une nouvelle vue. Pour réaliser un plaquage de texture projectif ou *projective mapping*, il faut spécifier la position et l'orientation de la caméra virtuelle. Ce type de modèle nécessite de connaître tout son environnement et toutes les interactions.

Une autre technique de plaquage 2D est la technique de *Bump mapping* [Blinn, 1978]. Une perturbation est ajoutée à chaque normale de la surface avant chaque estimation de l'illumination pour chaque pixel de cette surface. Le résultat (figure 2.2) est plus riche, plus détaillé et le rendu est plus réaliste.

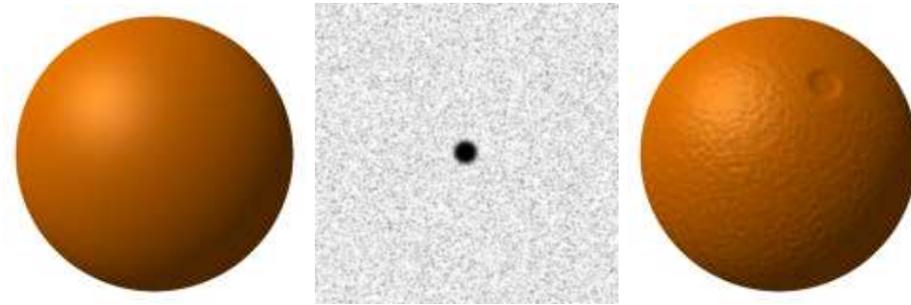


Figure 2.2 – Image obtenue sans bump mapping, image de bump mapping, image obtenue avec le bump mapping (GNU Free Documentation License)

Deux problèmes majeurs montrent les limites de cette technique :

- le premier problème rencontré est la distorsion de la texture sur la plupart des surfaces : on peut citer le cas classique de la sphère et du problème des pôles.
- le deuxième problème est l'absence d'information 3D. Il est impossible de décrire correctement des surfaces comme une surface rugueuse ou bosselée, car la géométrie influe alors autant que la couleur sur la texture.

2.2.2 Texture 3D

Si la texture 2D reste la méthode la plus commune pour plaquer une texture, cette technique est limitée. L'ajout d'une troisième dimension à la texture est un vaste champ de recherche [Dischler and Ghazanfarpour, 2001], que l'on peut regrouper en deux grandes familles.

2.2.2.1 Textures solides

Les textures solides ont été introduites pour corriger ce problème de distorsion et de discontinuité de la texture 2D. Elles consistent à définir directement la texture dans un espace 3D et non plus 2D. Introduites par Garder [Gardner, 1985], même si cette technique n'a pas été originellement nommée ainsi, elles permettent de synthétiser les paysages (arbre, montagne) et

principalement les nuages. Les textures sont définies comme le produit et la somme d'ondes sinusoïdales avec une amplitude et phase modulées le long de l'axe z . Définir les paramètres de ces textures (amplitude, fréquence) reste un problème entier. Ces textures solides peuvent être assimilées à des textures procédurales, car elles peuvent être définies par une fonction plutôt que par des données, que ce soit par une approche analytique [Perlin and Hoffert, 1989], une simulation physique, par exemple de l'aspect bois [Buchanan, 1998] ou encore une texture *solid antialiasing* [Peachey, 1988], qui permet de compenser l'effet visuel indésirable de *aliasing* ou recouvrement spectral qui s'affiche sous la forme de dentelures ou contours brisés (figure 2.3). Les textures solides sont adaptées pour représenter les matériaux qui sont réellement 3D, comme le bois ou le marbre.

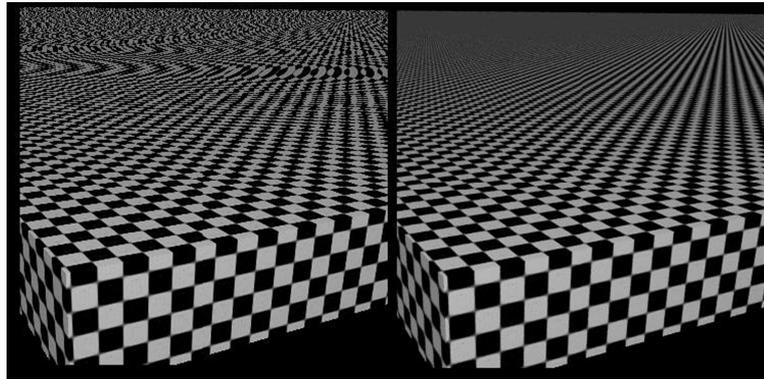


Figure 2.3 – Effet de crênelage, compensé ou non

2.2.2.2 Textures géométriques

Si les textures solides rajoutent une information de couleur dans l'espace 3D, les textures géométriques ajoutent une troisième dimension sous la forme d'une réelle géométrie apparente. Elles permettent de rendre des phénomènes complexes comme le feu, le coton, qui ne peuvent pas être rendus réalistement par les textures 2D ou solides. L'extension du *bump mapping*, appelée *displacement texture* [Cook, 1984] permet de modifier la position 3D des points suivant leur normale. Elle permet d'ajouter une troisième dimension.

On peut répartir les techniques de texture géométrique suivant :

- *Hypertexture*, encore appelée *shape texturing*, est une technique 3D basée sur le principe de la modulation de densité. La cible doit être explicitement définie comme une densité (figure 2.4).
- *texel mapping* : le maillage en triangles est abandonné au profit d'une variation de densité avec une fonction de réflectance. L'élément de base est le texel, un parallélépipède défini par



Figure 2.4 – Exemple de sphère hypertexturée

une variation de densité en son sein, un repère local orthogonal indiquant sa position et une fonction de réflectance pour son ombre. Cette couche de texel est appliquée sur la surface comme une texture 2D mais avec une épaisseur (figure 2.5).

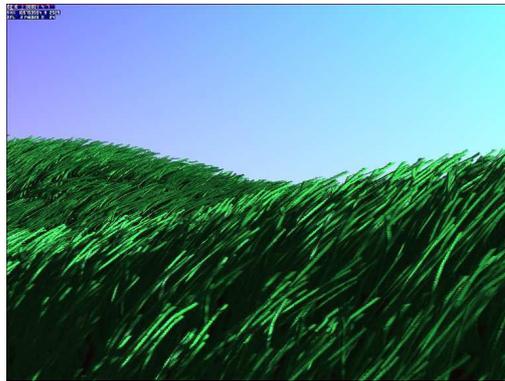


Figure 2.5 – Texture représentée par des texels

- ou encore de techniques basées sur des modèles géométriques explicites. Fleischer *et al.* présentent une texture basée sur un modèle géométrique explicite [Fleischer et al., 1995]. Cette technique permet de modéliser les détails de la surface comme des écailles, des plumes, des cornes (figure 2.6). La texture est générée à partir d'éléments géométriques qui suivent une simulation de développement cellulaire biologique avec une contrainte pour maintenir les cellules à la surface.

2.3 Correction géométrique

La salle de cinéma est le cas simple de projection d'images sur un écran. Cet écran est positionné de telle sorte que l'image ne soit pas déformée. Mais il arrive que le mur et le projecteur

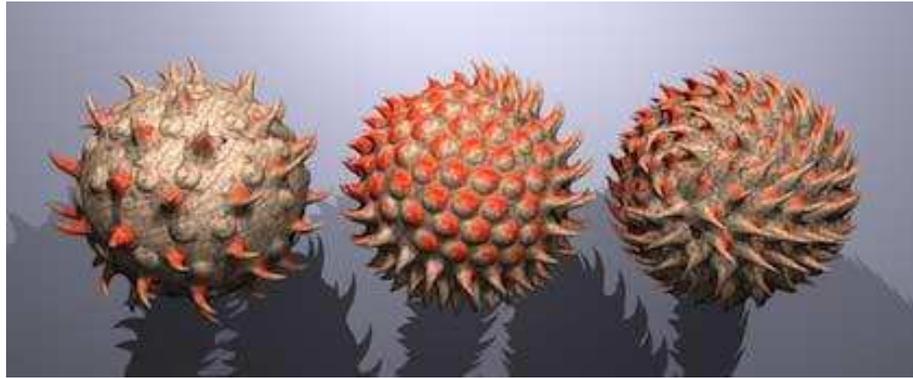


Figure 2.6 – Différents types d'écaïlle suivant l'équation [Fleischer et al., 1995]

ne soient pas positionnés de manière adéquate et le résultat est une image déformée. La projection d'image sur une surface (réelle ou non) repose généralement sur quelques hypothèses préalables, comme la planéité de la surface, et une position relative entre cette surface et le projecteur pour obtenir une image nette et non déformée. Si tel n'est pas le cas, il faut alors corriger l'image projetée pour obtenir visuellement une image non déformée. Ces corrections nécessitent une maîtrise du système projectif, dont nous rappelons brièvement les caractéristiques.

2.3.1 Caractéristiques de la caméra et du projecteur

Caméra et projecteur peuvent être décrits par le modèle sténopé (figure 2.7), en considérant le projecteur comme une caméra fonctionnant en sens inverse pour le trajet des rayons lumineux. La seule différence entre la caméra et le projecteur est la direction de projection : de la scène 3D vers une image 2D pour la caméra et d'une image 2D vers une scène 3D dans le cas du projecteur. Ce modèle schématise la caméra par un plan rétinien R et un centre optique C .

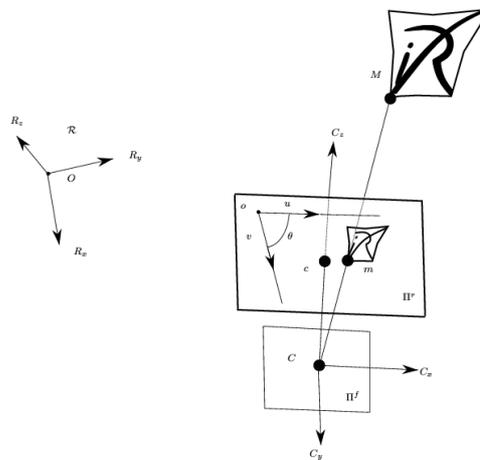


Figure 2.7 – modèle *pinhole* de la caméra

La formule suivante décrit la relation entre un point 3D M et sa projection m sur le plan de la caméra.

$$M = \begin{pmatrix} wX \\ wY \\ wZ \\ w \end{pmatrix} = P \cdot m = P \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (2.1)$$

La matrice P est la matrice de projection, w étant la coordonnée homogène ou facteur d'échelle. Elle se décompose en deux parties $P = K \cdot \begin{bmatrix} R & T \end{bmatrix}$. La matrice K représente les paramètres intrinsèques de la caméra, ses paramètres internes, comme la distance focale, la position de son point principal. R et T représentent la matrice de rotation et de translation de la caméra par rapport à l'origine du repère choisi, et sont les paramètres extrinsèques ou externes de la caméra. Le modèle sténopé de la caméra est présenté dans le chapitre 3.

2.3.2 Projection sur un plan

Une déformation de l'image due à un mauvais positionnement de l'écran est un défaut qu'un système projecteur-caméra peut corriger [Sukthankar et al., 2001] [Raskar and Beardsley, 2001]. Le système caméra-projecteur peut transformer l'image pour l'adapter au couple camera-écran. L'image transmise au projecteur peut être ainsi transformée de manière à corriger les distorsions liées aux positions relatives de l'écran et du projecteur. Raskar propose un système portable (figure 2.8) qui présente l'avantage de ne pas nécessiter d'étalonnage.

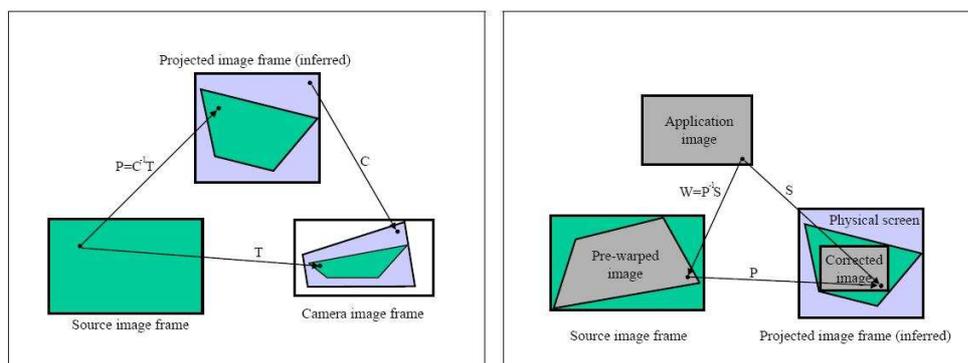


Figure 2.8 – Correction de la distorsion due au positionnement de l'écran [Sukthankar et al., 2001]

Ainsi la position, l'orientation et les paramètres des caméras et du projecteur sont inconnus. Trop de paramètres sont a priori inconnus mais Suthankar soulignant que l'écran est plat, le

problème se résume à l'estimation d'une homographie entre un point (x, y) du plan du projecteur et le point (X, Y) projeté sur l'écran.

L'équation se résume alors à :

$$\begin{pmatrix} wx \\ wy \\ w \end{pmatrix} = p \cdot \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = \begin{pmatrix} p_1 & p_2 & p_3 \\ p_4 & p_5 & p_6 \\ p_7 & p_8 & p_9 \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \quad (2.2)$$

avec comme condition $\|p\| = 1$

Quatre est alors le nombre minimum de points pour déterminer la matrice p , en supposant que toute combinaison de trois points parmi ces quatre ne soit pas colinéaire. Le problème reste donc de déterminer ces points communs. Suthankar utilise les quatre coins de l'image comme estimation initiale de l'homographie et ensuite l'affine en projetant une mire. Beardsley [Beardsley et al., 2005] utilise une méthode similaire pour calculer l'homographie de son système portatif.

Si la méthode précédente s'accommode d'un système non étalonné (et donc de plus d'inconnues), Raskar emprunte une voie différente pour adapter l'image à l'écran [Raskar and Beardsley, 2001] et obtenir un effet similaire (figure 2.9). A partir d'un système étalonné dont la position horizontale et verticale du projecteur est déterminée par un capteur d'inclinaison, il adapte l'image projetée pour obtenir une image de format 4 :3 suivant la ligne horizontale la plus longue possible comprise dans la zone de projection.

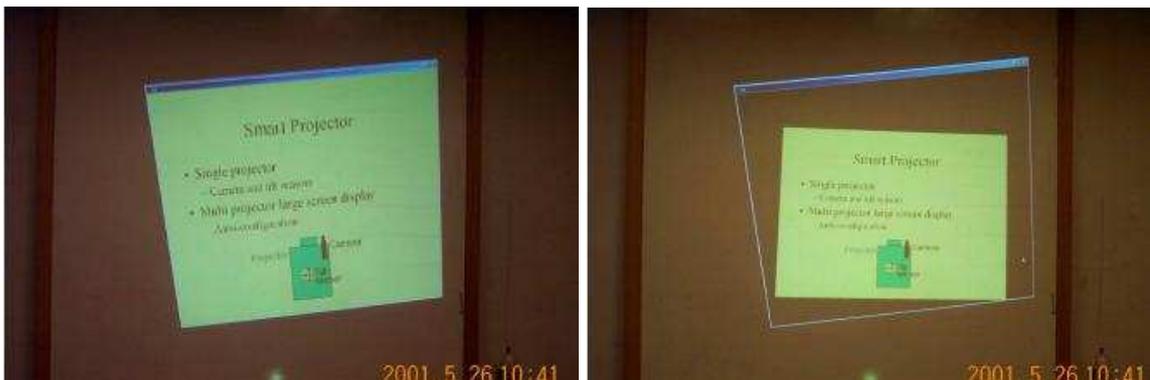


Figure 2.9 – Image droite englobée dans la zone de projection [Raskar and Beardsley, 2001]

2.4 Correction visuelle

Les techniques de réalité virtuelle sont aussi appliquées pour améliorer le rendu des scènes projetées. Quelques rappels sont nécessaires.

La technique de lancé de rayon est une technique de synthèse d'image qui consiste à parcourir le trajet inverse de la lumière de la scène vers l'oeil. Si elle permet de traduire les phénomènes simples que sont la réflexion et la réfraction de la lumière, en revanche elle ne peut pas prendre en compte les phénomènes physiques comme la dispersion lumineuse. La technique de lancé de rayon est combinée avec la technique de radiosité qui permet d'évaluer les phénomènes de transfert de lumière entre les différents éléments d'une scène.

Kajiya [Kajiya, 1986] a posé les bases d'une nouvelle technique posant la technique de lancé de rayon grâce à une équation intégrale qui généralise les équations de rendus de solution de type équation de Monte-Carlo.

$$L(x \rightarrow x) = E(x \rightarrow x) + \int_x f_r(x \rightarrow x) \cdot L(x \rightarrow x) \cdot V(x, x) \cdot G(x, x) dx \quad (2.3)$$

$L(x' \rightarrow x'')$ est la radiance totale provenant de x' vers x''

$E(x' \rightarrow x'')$ est la radiance émise de x' vers x''

f_r est la BRDF (*Bidirectional Radiosity Function*), équation qui définit comment la lumière se réfléchit sur une surface opaque

$V(x', x'')$ est le terme visible : 1 si x est visible de x' , sinon 0

$G(x', x'')$ est le terme de géométrie

Raskar applique cette équation pour corriger l'intensité de la projection de lumière blanche sur un carré vert et de lumière verte sur un carré blanc de telle sorte que l'on ne puisse plus distinguer le carré vert sous lumière blanche du carré blanc sous lumière verte (figure 2.10).

2.4.1 Multi projecteur

L'usage de plusieurs projecteurs se heurte principalement à deux problèmes : les variations photométriques et le problème de continuité de l'image.

Les variations photométriques peuvent être dues aux projecteurs eux mêmes : deux projecteurs du même constructeur ont une différence de radiosité ou des fonctions de transfert différentes.

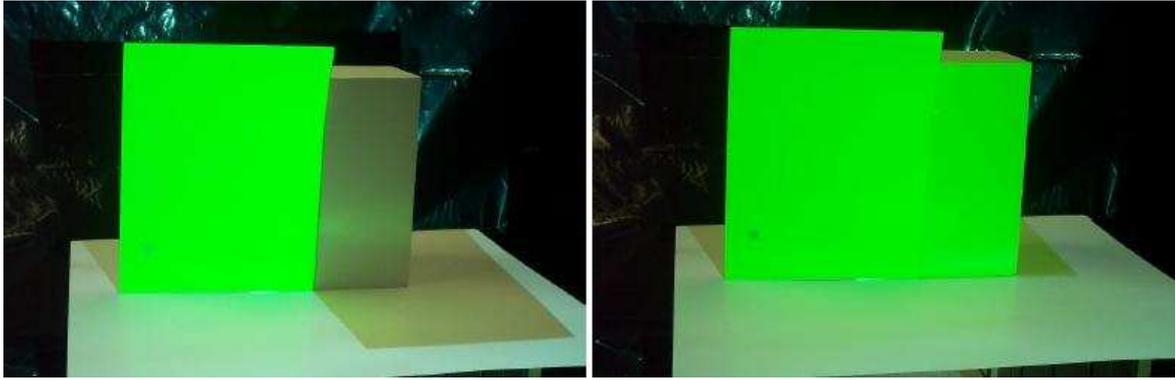


Figure 2.10 – Correction d'intensité de la lumière

Majumder *et al.* propose une méthode pour corriger en temps réel ces variations pour obtenir une image corrigée (figure 2.11).



Figure 2.11 – Image non corrigée, Image photométrique uniforme. [Majumder et al., 2003]

Mais ces erreurs peuvent être dues aussi à des imperfections sur l'écran. Bimber [Grundhöfer and Bimber, 2008] ou Nayar [Nayar et al., 2003] définissent complètement le modèle radiométrique d'un système projecteur-caméra et atténuent les dégradations de l'image dues à la surface de projection en modifiant l'image projetée pour compenser radiométriquement ces imperfections. La figure 2.12 montre les différentes images obtenues pour compenser ces imperfections.

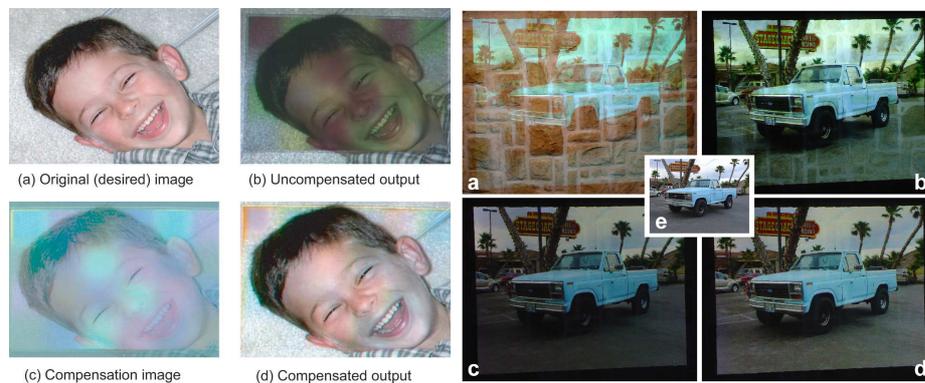


Figure 2.12 – De gauche à droite : [Nayar et al., 2003],[Grundhöfer and Bimber, 2008]

De nombreux auteurs se sont intéressés au recalage pour plusieurs projecteurs. Raskar [Raskar et al., 1999] reconstruit les surfaces sur lesquelles des images seront projetées, la jonction de ces différents plans étant résolue par une pondération des surfaces qui se chevauchent pour assurer leur continuité. Tardif *et al.* proposent une méthode pour projeter de manière adéquate une image sur un mur inconnu. Leur méthode s'adapte aussi bien au cas d'un système mono que multi-projecteur [Tardif et al., 2000]. Ce système, composé de projecteur(s) et d'une caméra, ne nécessite pas un étalonnage fort du système mais l'estimation de la position relative des caméras, réalisée par la technique de lumière structurée.

L'emploi de plusieurs projecteurs permet de dépasser les limites propres à un seul projecteur : l'amélioration de la qualité de l'image ou l'augmentation de la taille de l'image globale en sont deux avantages. La résolution de l'image pour des systèmes multi-projecteur est assimilée à celle du projecteur de la zone concernée. Jaynes propose de tirer partie des zones de recouvrement pour augmenter la résolution et le contraste de la zone d'affichage [Jaynes and Ramakrishnan, 2003]. Les images projetées sont décomposées au niveau sub-pixel dans les zones de recouvrement. Le système complet peut obtenir une résolution supérieure à celle des projecteurs.

Les systèmes multi-projecteur ont d'autres avantages. Majumder et Welsch proposent de tirer partie d'un tel système [Majumder and Welch, 2001] pour obtenir l'effet de profondeur ou DOF (*Depth of Field Effect*) grâce à la superposition optique sur les zones de recouvrement. Par l'intermédiaire de deux projecteurs dont les projections se recouvrent parfaitement, la figure 2.13 reproduit l'effet de champ de profondeur de l'oeil humain et reproduit une caractéristique de la vision humaine en simulant la zone de focus de l'oeil humain sur une scène globale plus floue.



Figure 2.13 – Focus et défocus de deux projecteurs [Majumder and Welch, 2001]

2.4.2 Projection sur des Objets réels

Verlinden *et al* [Verlinden et al., 2003] proposent un système de réalité augmentée pour ajouter l'information de texture à des objets créés par prototypage rapide. Ce système, présenté figure 2.14, baptisé WARP (*Workbench for Augmented Rapid Prototyping*) se compose d'une table rotative et d'un projecteur. Pour régler l'image sur l'objet, une silhouette du modèle est projetée. Cette étape manuelle permet d'aligner correctement l'objet par rapport à l'image, tout en plaçant son origine sur le centre de la table de rotation.

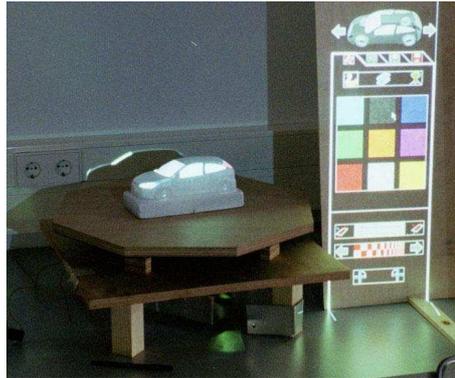


Figure 2.14 – Système proposé par verlinden

Matkovic [Matkovic et al., 2005] propose un autre système de réalité augmentée. Il décompose son système (figure 2.15) en deux entités, la table 3D, qui joue le rôle de support et de décors, et un pinceau virtuel qui ajoute la texture sur l'objet. La scène ainsi recomposée est constituée de l'objet peint, du fond de la scène projetée et de marqueurs physiques.



Figure 2.15 – Système de réalité augmentée [Matkovic et al., 2005]

D'autres systèmes de réalité augmentée se contentent de projeter des textures sur des surfaces planes. Le système de salle immersive ou CAVE en est un exemple, mais on peut aussi citer le système DOME (*Digital Object Media Environment*, figure 2.16) développé par Webb et Jaines [Webb and Jaines, 2005].

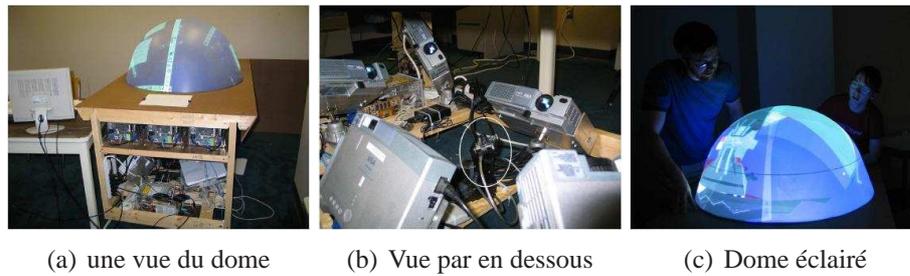


Figure 2.16 – Système DOME

Raskar *et al.* ont développé plusieurs systèmes pour projeter de la texture sur des objets réels. Le temple de Tajh Mahal est l'oeuvre la plus connue [Raskar et al., 2000], mais on peut citer également le système de peinture dynamique [Bandyopadhyay et al., 2001]. En déterminant un jeu de points 3D connus sur l'objet et les pixels correspondants sur l'image du projecteur, la matrice de projection 3×4 peut être déterminée [Faugeras, 1993] à un facteur près, matrice qui peut être ensuite décomposée pour déterminer les paramètres intrinsèques et extrinsèques. Une fois le projecteur étalonné, l'image projetée s'adapte à la géométrie de l'objet (figure (2.17)).

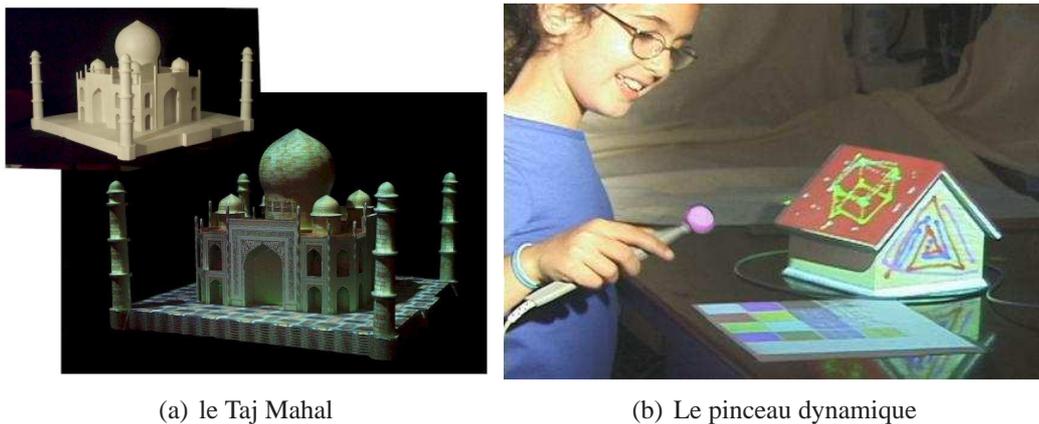


Figure 2.17 – Outils développés par l'équipe de Raskar ,[Raskar et al., 2000] [Bandyopadhyay et al., 2001].

Ashdown et Robinson combinent le concept de table dynamique et l'emploi de plusieurs projecteurs [Ashdown and Robinson, 2003]. La figure 2.18 démontre l'usage d'un projecteur en basse résolution et le second en haute résolution suivant la position des documents projetés sur la table. Le problème de marqueurs est résolu par l'emploi de stylos dynamiques. Contrairement aux systèmes multi-projecteur, ce système n'aborde pas les problèmes de luminance et de chrominance, la disposition des projecteurs éliminant le problème de recouvrement.

Berard traque les mouvements sur la table avec un système totalement différent [Bérard, 2003]. Il combine un seul projecteur avec deux caméras. Les mouvements imprimés aux documents virtuels sur la table sont simulés à l'aide de marqueurs réels, des jetons. Ces jetons sont traqués

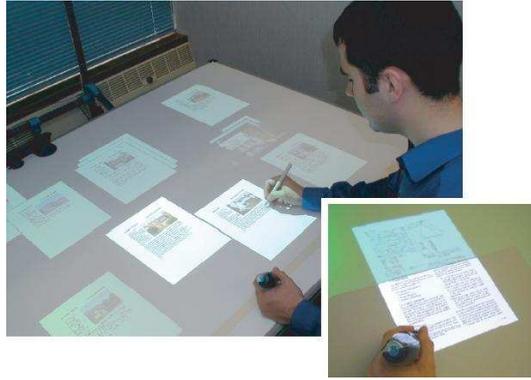


Figure 2.18 – Table dynamique développée par Ashdown [Ashdown and Robinson, 2003]

par détection de couleur. Un modèle de couleur des jetons est extrait d'un jeu de pixels des jetons. Les auteurs ont choisi de travailler dans l'espace de couleur (r, g) au lieu de l'espace (R, G, B) avec la relation $r = \frac{R}{R+G+B}$ et $g = \frac{G}{R+G+B}$. Ce changement d'espace permet de limiter l'impact des variations d'intensité de lumière sur la détection de jeton. Le modèle de couleur est alors représenté par son centre bi-dimensionnel m et la matrice de covariance C . Dans l'image chaque pixel est substitué par sa distance de Mahalanobis :

$$d = \sqrt{(p - m)^t \cdot C^{-1} \cdot (p - m)} \quad (2.4)$$

avec p étant le pixel (r, g) . Un seuil est ensuite choisi empiriquement pour transformer la carte de distance en une carte binaire. Un critère de taille sur le nombre de pixels nécessaires est ajouté pour déterminer ou non la présence de jeton, sa taille étant inférieure à la taille théorique d'un jeton, celui-ci étant souvent partiellement occulté. Une amélioration de ce système a été présentée par Gupta *et al.*, en utilisant une seule caméra et utilisant un algorithme de *tracking* sur les jetons pour maintenir l'étalonnage de son système [Gupta and Jaynes, 2005].

Déterminer la zone d'intérêt ou de projection concerne aussi bien les tables que les panneaux. Ainsi Borkowski [Borkowski et al., 2003] propose le PDS (*Portable Display Screen*), ou écran d'affichage portable. L'écran étant supposé plat, la situation est similaire à la détermination d'une homographie estimée à partir de quatre points minimum. Borkowski utilise les bords de son écran (figure 2.19). Suite à un pré-traitement pour enlever des régions de couleur, les bords de l'ardoise sont détectés par un gradient magnitude calculé par convolution avec un gradient gaussien avec un σ approprié (racine carré de la variance), suivant la formule

$$\text{Gradient}(\sigma, x, y) = \sqrt{(L_x(\sigma, x, y))^2 + (L_y(\sigma, x, y))^2} \quad (2.5)$$

L_x et L_y étant les dérivées premières gaussiennes. Les points de fort contraste sont utilisés dans la transformée de Hough pour déterminer les bords. En limitant la zone de recherche dans l'image et en estimant l'emplacement futur des bords de l'ardoise dans l'image, l'application peut être en temps réel. Enfin pour déterminer la projection de l'image du projecteur sur l'ardoise, les quatre points nécessaires à l'estimation de l'homographie plan à plan sont les quatre intersections des bords détectés.



Figure 2.19 – de gauche à droite : détection des bords grâce au maxima de la transformée de Hough, projection sur l'ardoise [Borkowski et al., 2003]

2.4.3 Projection temps réel

Jhonson et Fuchs proposent un système de projection en temps réel sur des surfaces complexes [Johnson and Fuchs, 2007]. Une fois le système étalonné et la géométrie de l'écran définie (un coin de mur par exemple), pour assurer le suivi en temps réel de l'image, il prédit l'image du projecteur pour trouver des points d'intérêt avec la vue de la caméra. Cette prédiction n'est possible que si seuls les paramètres externes de l'étalonnage du projecteur varient. Ainsi l'image projetée s'adapte à l'écran suivant la position du projecteur (figure 2.20). Le projecteur peut donc bouger, mais pas le couple de caméras. Ce système ne permet pas de corriger les erreurs photométriques dues à l'écran.

Un autre système en temps réel s'appuie sur le *tracking* pour corriger un problème d'occultation de l'image projetée par la présence de l'orateur devant l'écran (figure 2.21). Ce système [Audet and Cooperstock, 2007], composé de deux caméras et deux projecteurs, permet d'effacer la présence de l'orateur devant l'écran. Le premier projecteur, qui projette la scène, n'illumine pas l'orateur alors que le deuxième projecteur comble l'*ombre* de l'orateur sur l'image. Les premiers systèmes, comme celui développé par Jaynes [Jaynes et al., 2001], se limitait à effacer l'ombre détectée sur l'image et les contours de l'orateur restaient perceptibles sur l'écran (figure 2.21).

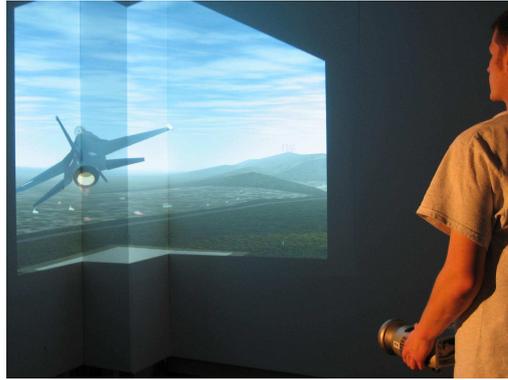


Figure 2.20 – Projection en temps réel sur une surface complexe



Figure 2.21 – Avant et Après, remove shadow

Jones *et al.* ont développé un système de projection 3D [Jones et al., 2007]. Leur système (figure 2.22) utilise un projecteur et un miroir posé sur une table rotative à haute vitesse. La vitesse de rotation et la fréquence de projection étant synchronisées, on obtient l'illusion d'une projection 3D ou à 360°.

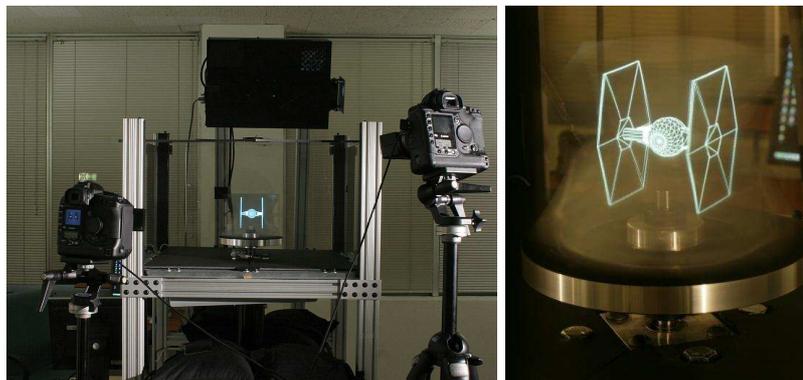


Figure 2.22 – Système développé par Jones *et al.*

Les modèles projectifs et perspectifs entraînent des défauts de perception de l'image. Projeter des images perspectives régulières exagère la perspective horizontale et provoque des étirements quand le point de vue s'élève. Projeter une image perspective, qui apparaîtrait correcte pour l'observateur si le miroir était diffus, exagère la perspective horizontale et crée des clés de voûte. Utiliser l'algorithme MCOP (*Multiple Center Of Projection*) produit des images de

perspective corrigée, quelle que soit la position de l'observateur. La figure 2.23 expose les différents résultats obtenus suivant le modèle de perspective. Des modèles plus simples pour projeter l'image sur le plan donnent des résultats distordus suivant que la méthode choisie privilégie la perspective horizontale [Cossairt et al., 2007] ou verticale [Raskar et al., 1998]. L'algorithme MCOP permet d'obtenir une image sur le plan avec une perspective visuellement correcte.

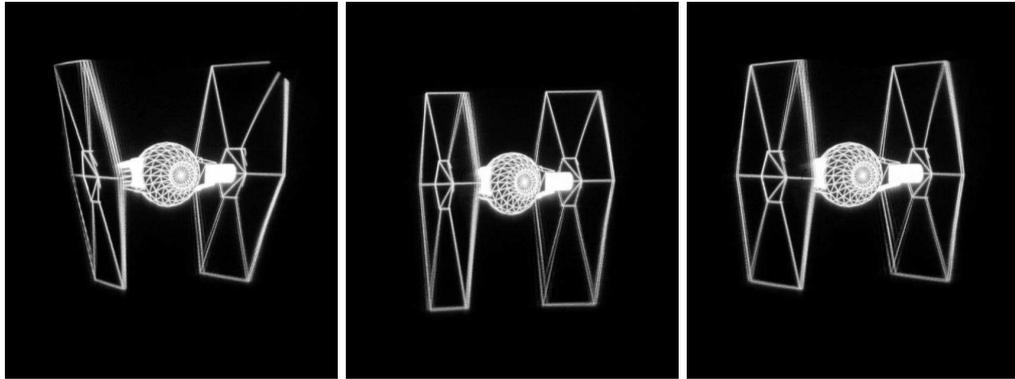


Figure 2.23 – Différentes images obtenues suivant le modèle de perspective

2.5 Synthèse

La réalité augmentée, par ses applications dans des domaines comme l'archéologie ou la médecine, bénéficie d'une attention continue de la communauté scientifique. Les systèmes projecteur caméra, à travers différentes exploitations comme les salles immersives, tables dynamiques ... nécessitent encore des développements pour résoudre les derniers problèmes.

2.5.1 Les projections sur des plans

La projection d'images sur une surface plane est un sujet traité, bien que l'amélioration de l'inter-activité de systèmes (table dynamique,...) nécessite encore des travaux. Le calcul d'homographie entre deux images est également maîtrisée : la seule difficulté est l'obtention d'au moins quatre points de correspondance entre les deux images pour pouvoir estimer l'homographie. Deux solutions sont envisagées pour lever cette difficulté :

- l'emploi de marqueurs physiques comme des jetons.
- des points caractéristiques de l'image (coins de l'image projetée).

L'extension de cette technique aux systèmes multi-projecteurs, le problème de recouvrement et d'étalonnage photométrique est également maîtrisé. Tous ces systèmes restent néanmoins

statiques et limités à un système de projection plane, tout comme les systèmes de simulation immersive de type CAVE.

Les systèmes temps réel développés récemment se limitent à "des surfaces simples". Le système de Jhonson se limite à des plans, et si l'application se corrige en temps réel suivant les mouvements aléatoires du projecteur, la contrainte d'immobilité du jeu de caméras et donc d'un repère absolu intemporel demeure. De même le système de projection 3D de Jones se limite au mouvement rotatif du miroir : à l'instant donné t la position relative du miroir plan par rapport au projecteur est connue.

2.5.2 Les projections sur des formes tridimensionnelles

Des systèmes de simulation virtuelle, comme le système DOME, regroupent les éléments précédents. Il combine les problèmes de multi-projection, donc de recouvrement et d'étalonnage photométrique, ainsi que de projection sur un dôme. Ce système ne s'affranchit pas de l'écran de projection, ni de sa position relative par rapport aux projecteurs.

Seule la projection sur des surfaces simples (plane ou courbe) a été abordée. L'exemple le plus connu de projection sur un objet est le temple de Taj Mahal texturé par Raskar *et al.*. Ce projet combine les problèmes de texture sur un objet 3D et ceux liés à un système multi-projecteur. Si le problème d'étalonnage photométrique est étudié en utilisant le modèle fourni par Kajiya [Kajiya, 1986], l'estimation de la position relative de l'objet par rapport au projecteur est réalisée manuellement par la projection directe de croix (figure 2.24) sur certains points 3D caractéristiques du modèle physique pour estimer la matrice de projection de perspective du projecteur et donc facilement identifiables sur un modèle numérique.

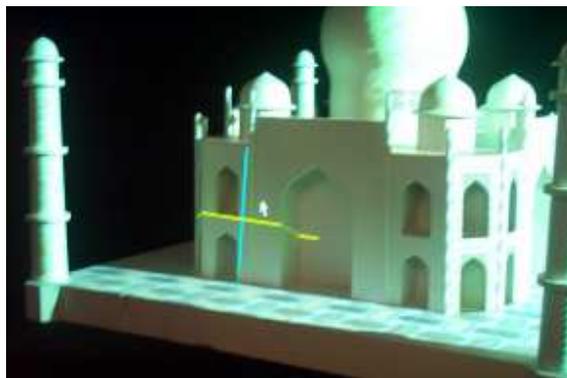


Figure 2.24 – Détermination de points caractéristiques sur le temple de Taj Mahal [Raskar and Beardsley, 2001]

D'autres systèmes ont été mis au point, comme celui développé par Verlinden ou Matkovic, mais se concentrent sur l'aspect fonctionnel du système. L'utilisation de marqueurs physiques jouent le rôle des points de correspondance 2D et 3D. Projeter le contour de l'objet pour qu'il coïncide avec la silhouette de l'objet est une solution envisageable.

2.5.3 Suggestion

Le but de cette thèse est d'éliminer toute étape manuelle ou intervention humaine dans l'ajustement de l'image projetée sur l'objet tridimensionnel réel. Aucune information sur la forme de l'objet n'est utilisée. La seule hypothèse que nous formulerons sera que l'objet est sans texture, de préférence blanc, restriction que l'on peut minimiser eu égard à de nouveaux travaux basés sur la compensation radiométrique [Zollmann and Bimber, 2007], qui ont montré qu'il était possible de compenser des défauts de texture de l'écran pour afficher la texture que l'on veut percevoir.

Ajuster l'image projetée revient à déterminer la position relative de l'objet par rapport au projecteur. Celui-ci va nous permettre de résoudre ce problème au travers de la technique de lumière structurée, qui consiste en la projection d'image codée. Ce code permet de résoudre notre problème grâce à l'estimation de points de correspondance entre l'image projetée et l'image perçue par une caméra. L'objet ainsi replacé dans le repère du projecteur, la projection de texture est désormais possible. Ces techniques de lumière structurée nous autorisent à des scènes dynamiques. L'annexe A expose deux méthodes "temps réel" dite de lumière structurée imperceptible, qui cachent la lumière structurée dans la séquence de projection de la scène.

Chapitre 3

Géométrie et Étalonnage d'un Système Projecteur-Caméra

Notre objectif est de synthétiser l'image obtenue par une caméra positionnée à la place du projecteur. Cette synthèse d'image réclame une connaissance précise du système projecteur-caméra. Ce chapitre va présenter l'étalonnage d'un système composé d'un projecteur et de deux caméras, méthode qui peut être généralisée à un système multi-projecteur et multi-caméras.

Ce chapitre sera décomposé de la manière suivante :

- Les différents paramètres de la caméra seront donnés.
- Les méthodes d'étalonnage retenues seront exposées.
- Les différents outils de la géométrie épipolaire seront définis.

3.1 Étalonnage

L'étalonnage de caméras est une étape importante. On distingue en général deux grandes méthodes d'étalonnage :

- Les algorithmes d'auto-étalonnage ne font pas ou peu d'hypothèses sur la structure particulière de la scène. Ils essaient d'estimer les paramètres intrinsèques en exploitant les contraintes imposées sur ces paramètres par les relations épipolaires ou trilineaires. Pour un jeu de caméras, une minimisation numérique dans l'espace des matrices projectives, qui transforme le jeu original de caméras en un nouveau qui satisfasse les contraintes des paramètres intrinsèques, est possible. Ces méthodes, dites d'auto-étalonnage, se décomposent en deux étapes ; la première consistant à extraire des points d'intérêts de la scène [Harris and Stephens, 1988], la seconde en l'étalonnage lui-même [Pollefeys, 1999]. Ces méthodes permettent aussi de modifier les paramètres intrinsèques pendant la mesure.
- La deuxième méthode consiste à contourner la difficulté de déterminer les points de correspondance, en utilisant une mire ; l'extraction de points de correspondance est alors facilitée mais nécessite l'intervention d'un opérateur humain.

Que l'on soit en étalonnage ou auto-étalonnage, les paramètres à estimer restent les mêmes. Le modèle utilisé est le modèle retenu par Bouguet [Bouguet, 2004], méthode elle-même inspirée par la modélisation de Heikkila [Heikkila and Silven, 1997].

3.1.1 Modèle / Système

Le modèle choisi pour représenter une caméra dépend de nombreux facteurs, comme la précision, la prise en compte ou non de la distorsion, l'accès aux paramètres... le modèle le plus simple est celui proposé par Hall [Hall et al., 1982] qui propose une relation linéaire entre le point 3D de la scène et sa projection 2D sur la caméra (figure 3.1).

3.1.1.1 Le modèle linéaire

Nous utilisons le modèle sténopé pour décrire le comportement géométrique de la caméra : il s'agit d'une modélisation simple et linéaire du processus de formation des images au sein d'une caméra. Ce modèle suppose que le système optique de la caméra, sa lentille, respecte les conditions de Gauss. Il suffit d'exprimer les relations de passage du repère monde au repère caméra,

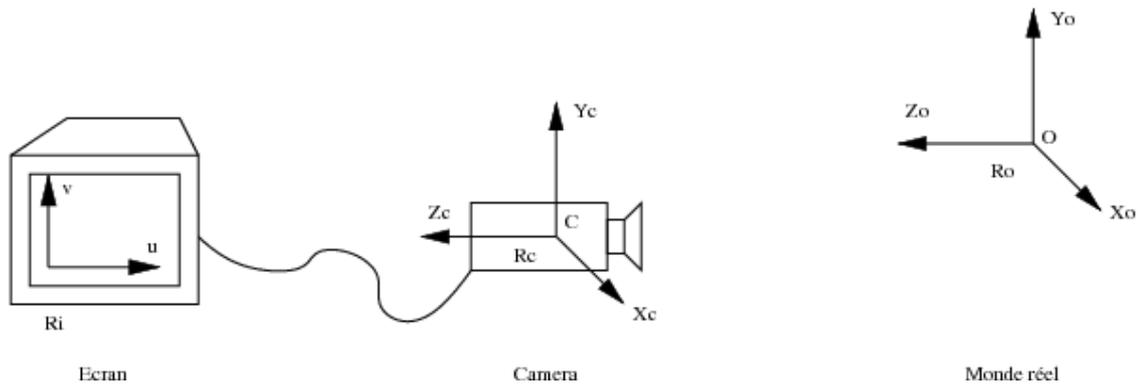


Figure 3.1 – Repères employés pour étalonner une caméra, dessinée par Jydidier en octobre 2005.

d'exprimer la projection du repère caméra dans le plan image et d'appliquer la transformation affine qui conduit aux coordonnées de l'image.

Ainsi trois repères sont nécessaires pour décrire le processus :

- $(R_O, \vec{X}_O, \vec{Y}_O, \vec{Z}_O)$: qui est le repère associé à l'espace de travail. O est l'origine de ce repère,
- $(R_C, \vec{X}_C, \vec{Y}_C, \vec{Z}_C)$: qui est le repère associé à la caméra. C est le centre optique de la caméra,
- (R_i, \vec{u}, \vec{v}) : qui est le repère associé à l'image visualisée.

Les paramètres employés dans ce modèle sont divisés en deux catégories : les paramètres intrinsèques qui sont internes à la caméra, et les paramètres extrinsèques qui varient suivant la position et l'orientation de la caméra dans l'espace de travail. Les paramètres intrinsèques sont :

- f_c : la distance focale en pixels
- cc : le point principal
- α_c le facteur d'obliquité : il définit l'angle entre les axes horizontal et vertical des pixels
- k_c : les coefficients de distorsion radiale et tangentielle.

Les paramètres extrinsèques sont :

- $R_{3 \times 3}$: matrice de rotation permettant de passer du repère lié à l'espace de travail au repère lié à la caméra,
- t_x, t_y et t_z : les composantes du vecteur de translation permettant de passer du repère lié à l'espace de travail au repère lié à la caméra.

Étalonner la caméra consiste à déterminer la valeur numérique des paramètres de ce modèle. Il est toutefois possible de les regrouper de manière différente, suivant la forme sous laquelle ce modèle doit ensuite être exploité.

Soit $\mathbf{X} = \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}$ un point 3D dans le repère associé à la caméra, que nous allons projeter sur l'image de la caméra en respectant les paramètres définis précédemment :

Par commodité, nous normalisons les coordonnées :

$$x_n = \begin{bmatrix} \frac{X_c}{Z_c} \\ \frac{Y_c}{Z_c} \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} \quad (3.1)$$

Les coordonnées finales x_p, y_p de la projection de \mathbf{X} sur le plan image de la caméra s'expriment alors par :

$$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \begin{bmatrix} f_c(1) & \alpha_c \cdot f_c(1) & cc(1) \\ 0 & f_c(2) & cc(2) \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{n(1)} \\ x_{n(2)} \\ 1 \end{bmatrix} \quad (3.2)$$

Néanmoins ce modèle ne traduit que la relation d'un modèle idéal de la boîte noire, mais n'intègre pas les erreurs engendrées par les aberrations de la lentille, comme la distorsion.

3.1.1.2 Distorsion

Brown est le premier à introduire la notion de distorsion dans son modèle *Plumb Bob* (polynôme radial et minceur du prisme) [Brown, 1971]. D'autres modèles, comme celui proposé par Weng [Weng et al., 1992], différencient trois types de distorsions : distorsion radiale, distorsion de non-centrage, distorsion de minceur de prisme, les deux dernières distorsions étant source de distorsion radiale et tangentielle. On peut donc regrouper les distorsions en deux groupes :

- la distorsion radiale, provoquée par les erreurs de courbure radiale de la lentille.
- la distorsion tangentielle causée par le non centrage, ou centrage imparfait des composants de la lentille et autres défauts industriels dans la création de la lentille.

Cette distorsion est exprimée dans le vecteur k_c et s'exprime suivant la relation :

$$x_d = \begin{bmatrix} x_d(1) \\ x_d(2) \end{bmatrix} = (1 + k_c(1)r^2 + k_c(2)r^4 + k_c(5)r^6)x_n + dx \quad (3.3)$$

avec $r^2 = x^2 + y^2$ et dx le vecteur de distorsion tangentielle $dx = \begin{bmatrix} 2kc(3)xy + kc(4)(r^2 + 2x^2) \\ kc(3)(r^2 + 2y^2) + 2kc(4)xy \end{bmatrix}$

Le pixel $\begin{bmatrix} x_p \\ y_p \end{bmatrix}$ peut alors s'exprimer directement par :

$$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \begin{bmatrix} f_c(1) & \alpha_c \cdot f_c(1) & cc(1) \\ 0 & f_c(2) & cc(2) \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{d(1)} \\ x_{d(2)} \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} x_{d(1)} \\ x_{d(2)} \\ 1 \end{bmatrix} \quad (3.4)$$

Il ne reste plus qu'à exprimer les coordonnées du point dans le repère du monde par rapport à celui de la caméra : $X_c = R\mathbf{X} + T$

Au final la relation entre un point du repère monde et sa projection sur l'image est :

$$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} R & t \end{bmatrix} \begin{bmatrix} x_d \\ 1 \end{bmatrix} \quad (3.5)$$

3.1.2 Caméra

Les caméras ne justifient pas toujours un modèle optique aussi sophistiqué. En effet, il est admis que le pixel de la caméra est carré, et ainsi on fait l'hypothèse que le facteur d'obliquité est nul, soit $\alpha_c = 0$. La méthode que nous utilisons ne prend pas en compte ce facteur. Le modèle complet du vecteur de distorsion k_c (estimer la sixième composante de la distorsion radiale et la distorsion tangentielle) n'est pas nécessaire. Pour des caméras standards (non grand angle de vue) il n'est pas nécessaire de pousser le modèle de distorsion radiale au delà de la quatrième composante. La composante tangentielle de la distorsion peut être ignorée. Cette simplification est justifiée par le fait que la majorité des lentilles actuelles n'ont pas d'imperfection sur leur centre.

N'estimer que les quatre premières composantes du vecteur de distorsion k_c et ignorer la composante tangentielle correspond au modèle utilisé par Zhang [Zhang, 1999].

Un autre modèle de distorsion pour de bons systèmes optiques ou *narrow field of view lenses* est de limiter l'estimation de la distorsion radiale au second ordre. Dans ce modèle seul le premier paramètre est estimé, alors que les quatre autres composantes du vecteur sont de valeur nulle. Ce modèle est employé lorsque peu d'images sont disponibles pour l'étalonnage. Un

modèle plus complexe serait impossible à employer car le modèle proposerait plus d'inconnues que d'équations. Le facteur de distorsion et celui d'obliquité ne sont pas les seules paramètres que l'on peut simplifier pour alléger le modèle. Lorsque peu d'images sont disponibles pour l'étalonnage, il est difficile d'estimer la position du point principal cc . Il est alors commun d'estimer sa position au centre de l'image. En effet la position du point principal influe très peu sur la qualité de la reconstruction [Gomez et al., 2005].

L'étalonnage se fera en deux étapes :

- initialisation
- optimisation non linéaire.

L'initialisation propose une solution approchée qui ne tient pas compte de la distorsion de l'image. L'optimisation non linéaire minimise l'erreur de rétroprojection totale, dans le sens des moindres carrés, en tenant compte de tous les paramètres (9 degrés de liberté pour les paramètres intrinsèques et $6 \times image$ degrés de liberté pour les paramètres extrinsèques).

L'optimisation est réalisée par un gradient descendant itératif avec une implémentation explicite (forme approchée) de la matrice Jacobienne.

L'initialisation a été en partie inspirée par l'article de Zhang. L'estimation initiale des homographies planes est identique, mais l'estimation de la forme des paramètres internes est légèrement différente : l'orthogonalité des points de fuite est explicite et les coefficients de distorsion ne sont pas estimés dans la phase d'initialisation pour estimer une solution approchée. L'estimation finale du voisinage maximum est identique. Le modèle des paramètres intrinsèques utilisé est celui de Heikkil et Silven [Heikkila and Silven, 1997], avec deux coefficients de distorsion tangentielle supplémentaires.

3.1.3 Estimation de l'homographie entre le plan caméra et la mire

Comparée aux méthodes classiques qui nécessitent un équipement comme des plans orthogonaux, la méthode proposée par Zhang est simple et flexible. En effet elle ne nécessite que plusieurs images d'une mire sous différents points de vue.

La relation entre un point 3D $M = \begin{bmatrix} X & Y & Z \end{bmatrix}$ et sa projection $m = \begin{bmatrix} u & v \end{bmatrix}$ sur le plan de la caméra est :

$$s\tilde{m} = \mathbf{K} \begin{bmatrix} R & t \end{bmatrix} \tilde{M} \quad (3.6)$$

K est une matrice 3×4 qui projette les coordonnées normalisées de l'image sur le plan de la caméra, \tilde{m} est la notation pour le vecteur m en coordonnées homogènes en ajoutant une troisième coordonnée $\tilde{m} = \begin{bmatrix} u & v & 1 \end{bmatrix}$. s est un facteur scalaire arbitraire non-nul. Par commodité nous utiliserons l'abréviation A^{-t} au lieu de $(A^t)^{-1}$ ou $(A^{-1})^t$.

Dans le repère monde, on ajuste le repère à la mire de telle sorte que $Z = 0$ (figure 3.2), et on note la i^{me} colonne de R par r_i .

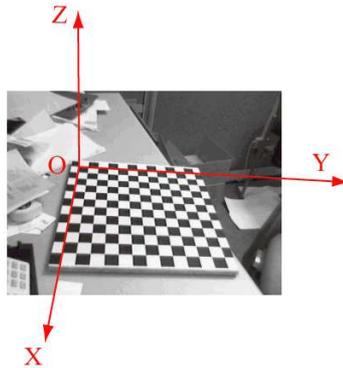


Figure 3.2 – Repère du monde et mire

L'équation s'écrit alors :

$$s\tilde{m} = \mathbf{K} \begin{bmatrix} r_1 & r_2 & r_3 & t \end{bmatrix} \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} r_1 & r_2 & t \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}. \quad (3.7)$$

Pour simplifier la notation, nous continuons à écrire M à la place de $M = \begin{bmatrix} X & Y \end{bmatrix}^t$.

L'équation peut alors s'interpréter comme l'expression de l'homographie reliant le point M et sa projection m :

$$s\tilde{m} = H\tilde{M} \quad (3.8)$$

avec $H = \mathbf{K} \begin{bmatrix} r_1 & r_2 & t \end{bmatrix}$.

3.1.4 Contraintes sur les paramètres intrinsèques

On note $H = \begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix}$.

L'équation peut alors s'écrire sous la forme

$$h_i^t B h_j = v_{ij}^t b \quad (3.12)$$

$$\text{avec } v_{ij} = \left[h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3} \right]^t$$

Les deux contraintes issues de l'estimation d'une homographie peuvent s'écrire sous la forme :

$$\begin{bmatrix} v_{12}^t \\ (v_{11} - v_{22})^t \end{bmatrix} b = 0 \quad (3.13)$$

Si n images du modèle sont disponibles, en regroupant les différentes équations, on obtient : $Vb = 0$, avec V une matrice $2n \times 6$. Il faut donc au minimum 3 images, soit $n \geq 3$, pour obtenir en général une solution unique b . La solution de $Vb = 0$ est connue, il s'agit des vecteurs propres de $V^t V$ associés aux plus petites valeurs propres.

Une fois b estimée, on peut alors en déduire K grâce à la formule $B = \lambda A^{-t} A$:

$$\begin{aligned} cc(2) &= \frac{(B_{12}B_{13} - B_{11}B_{23})}{(B_{11}B_{22} - B_{12}^2)} \\ \lambda &= B_{33} - \frac{B_{13}^2 + cc(2) \cdot (B_{12}B_{13} - B_{11}B_{23})}{B_{11}} \\ f_c(1) &= \sqrt{\frac{\lambda}{B_{11}}} \\ f_c(2) &= \sqrt{\frac{\lambda B_{11}}{(B_{11}B_{22} - B_{12}^2)}} \\ \alpha(c) &= -\frac{B_{12} \cdot f_c(1) \cdot f_c(2)}{\lambda} \\ cc(1) &= \alpha(c) \cdot cc(2) - \frac{B_{13} f_c(1)^2}{\lambda} \end{aligned} \quad (3.14)$$

Il reste enfin à estimer l'homographie entre le modèle et le plan suivant l'équation $s\tilde{m} = K \begin{bmatrix} R & t \end{bmatrix} \tilde{M}$. Cette équation est vérifiée si les points mesurés ne sont pas bruités. On détecte, malgré les précautions prises, un bruit que l'on peut modéliser par un bruit gaussien de moyenne nulle et de matrice de covariance Λ_{m_i} .

La technique présentée utilise la méthode de l'estimation du voisinage maximum [Myung, 2003]. Cette estimation de H est obtenue alors par la minimisation de :

$$\sum_i (m_i - \hat{m}_i)^t \Lambda_{m_i}^{-1} (m_i - \hat{m}_i) \quad (3.15)$$

avec $\hat{m}_i = \frac{1}{h_3^t M_i} \begin{bmatrix} \bar{h}_1^t M_i \\ \bar{h}_2^t M_i \end{bmatrix}$ et \bar{h}_1 la ième ligne de H

En pratique on suppose que $\Lambda = \sigma^2 I$. Cette hypothèse est raisonnable si les points sont extraits indépendamment selon la même procédure. Le problème ci-dessus devient alors un problème de minimisation non linéaire aux moindres carrés, soit $\min_H \sum_i \|m_i - \hat{m}_i\|$. Cette minimisation non linéaire est résolue par l'algorithme de Levenberg-Marquardt. L'hypothèse d'un bruit indépendant sur les points permet d'obtenir l'initialisation :

Soit $x = \begin{bmatrix} \hat{h}_1^t & \hat{h}_2^t & \hat{h}_3^t \end{bmatrix}^t$.

L'équation peut alors s'écrire :

$$\begin{bmatrix} \tilde{M}^t & 0^t & -u\tilde{M}^t \\ -v\tilde{M}^t & 0^t & \tilde{M}^t \end{bmatrix} x = Lx = 0 \quad (3.16)$$

Une solution de cette équation est le vecteur propre associé à la plus petite valeur propre de la matrice $L^t L$.

3.1.5.2 Optimisation non linéaire

Dans cette seconde étape de l'étalonnage des caméras, le facteur de distorsion est pris en compte. On cherche à minimiser la projection des points M suivant le modèle fourni, et de minimiser la distance entre ses projections et les points distordus \tilde{m} :

$$\sum_i \sum_j \|\tilde{m}_{ij} - \hat{m}(KK, R_i, t_i, M_j)\| \quad (3.17)$$

La méthode retenue pour minimiser l'équation 3.17 est la méthode du gradient descendant, avec un minimum empirique de 15 itérations.

3.1.6 Stéréo-étalonnage

Étalonner plusieurs caméras requiert l'estimation des paramètres intrinsèques et extrinsèques. Elle se déroule en deux phases :

1. Initialisation : Le stéréo-étalonnage est une extension de la méthode décrite précédemment. On étalonne les deux caméras séparément, et l'estimation des paramètres des deux caméras vont permettre l'initialisation de la phase suivante. Pour estimer les paramètres extrinsèques om et T caractérisant la position relative de la caméra droite par rapport à la caméra gauche, on utilise la relation unissant un point M de l'espace 3D dont les coordonnées sont respectivement X_r dans la caméra de droite et X_l dans celle de gauche :

$$X_R = R.X_L + T \quad (3.18)$$

2. Optimisation : Tous les paramètres sont recalculés et leurs incertitudes sont minimisées. Pour une même scène on utilise les équations des deux caméras dans un même système lié à la caméra gauche (figure 3.3).

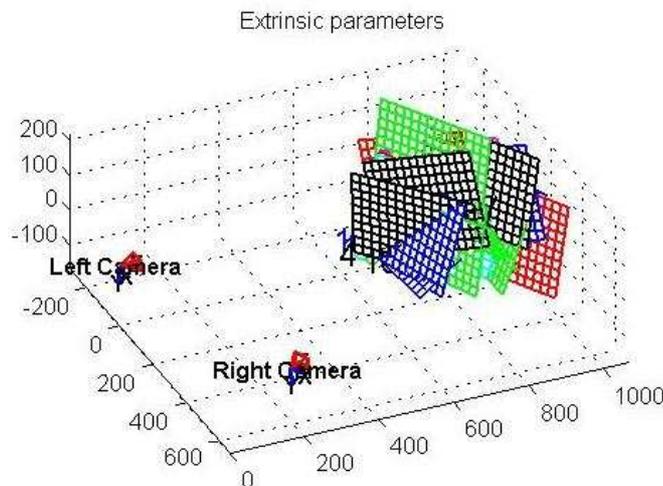


Figure 3.3 – Modélisation 3D de la position et de l'orientation des caméras, ainsi que des mires

3.1.7 Projecteur

L'étalonnage du projecteur n'est pas simple car il ne fournit pas d'image de la scène. Cette absence d'image de la scène nous prive d'informations nécessaires pour étalonner directement le projecteur. La solution retenue consiste à utiliser les caméras, qui sont désormais étalonnées,

pour obtenir les informations manquantes (un jeu de points de correspondance relié au projecteur). Il faut remarquer que le principal inconvénient d'une telle méthode est d'inclure dans notre estimation des paramètres du projecteur les incertitudes liées à l'estimation des paramètres des caméras.

Une autre solution aurait été d'utiliser la méthode de Raskar [Raskar and Beardsley, 2001] développée pour étalonner son système dual caméra projecteur, méthode qui évite l'emploi d'outils externes au projecteur, comme des caméras étalonnées à une position connue mais aussi de marqueurs physiques [Borkowski et al., 2003] sur l'écran ou de détecter explicitement les bords du rectangle de l'image à une position connue [Bérard, 2003]. Un capteur d'inclinaison, qui détermine l'axe vertical, complète alors le système dual caméra projecteur. Si la caméra est étalonnée en suivant le modèle de Zhang [Zhang, 1999], le projecteur est étalonné grâce à la projection d'une mire et l'estimation d'homographies H à partir de points de correspondances entre les points m_1 de la caméra et les points m_2 du projecteur [Hartley, 1995], reliés par la relation $m_2 \cong m_1$, avec \cong signifiant égalité à un facteur près.

Deux plans distincts sont utilisés, deux homographies distinctes sont déterminées. On estime la position des épipoles par l'équation $k.H_1^{-1}.e_2 = H_2^{-1}.e_2$, k étant un facteur inconnu. L'estimation des matrices de projection P_1 et P_2 qui se limitent à une reconstruction métrique (reconstruction à une échelle inconnue) permet de lever la dernière inconnue.

La méthode de Faugeras Toscani [Faugeras and Toscani., 1986] estime les paramètres intrinsèques et extrinsèques d'une caméra à partir d'un jeu de points de correspondance entre des points 3D et leur projection sur le plan de la caméra. Salvi *et al.* [Salvi et al., 2002] y ont ajouté la notion de distorsion radiale [Tsai, 1987], qui introduit les erreurs de courbure radiale de la lentille.

La méthode employée se décompose en deux parties :

- Une première estimation approximative des paramètres, qui utilise la méthode de Faugeras.
- Une estimation par itération qui inclut la distorsion et utilise la première étape comme initialisation.

Une étape préalable à l'utilisation du modèle de Faugeras itératif est l'estimation de la position 3D des points perçues par les deux caméras.

3.1.7.1 Méthode de Faugeras

On rappelle que la matrice d'étalonnage des paramètres internes s'écrit :

$$K = \begin{bmatrix} f_c(1) & \alpha_c \cdot f_c(1) & cc(1) \\ 0 & f_c(2) & cc(2) \\ 0 & 0 & 1 \end{bmatrix} \quad (3.19)$$

α_c , le paramètre d'obliquité, est considéré nul, car on suppose le pixel orthogonal. Le modèle de Faugeras peut se décrire par l'équation :

$$\begin{bmatrix} wx \\ wy \\ w \end{bmatrix} = \begin{bmatrix} f_c(1) & 0 & cc(1) & 0 \\ 0 & f_c(2) & cc(2) & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} f_c(1) \cdot r_{11} + cc(1) \cdot r_{31} & f_c(1) \cdot t_x + cc(1) \cdot t_z \\ f_c(2) \cdot r_{21} + cc(2) \cdot r_{31} & f_c(2) \cdot t_y + cc(2) \cdot t_z \\ r_{31} & t_z \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.20)$$

Cette équation peut s'écrire sous forme matricielle : $\begin{bmatrix} wx \\ wy \\ y \end{bmatrix} = A \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$ avec $A = \begin{bmatrix} A_1 A_{14} \\ A_2 A_{24} \\ A_3 A_{34} \end{bmatrix}$

Faugeras écrit cette équation sous la forme :

$$\begin{aligned} A_1^w P_w + A_{14} - {}^l X_u (A_3^w P_w + A_{34}) &= 0 \\ A_2^w P_w + A_{24} - {}^l Y_u (A_3^w P_w + A_{34}) &= 0 \end{aligned} \quad (3.21)$$

soit

$$\begin{aligned} {}^l X_u &= \frac{A_1}{A_{34}} {}^w P_w + \frac{A_{14}}{A_{34}} - \frac{A_3}{A_{34}} {}^w P_w {}^l X_u = T_1^w P_w + C_1 - T_2^w P_w {}^l X_u \\ {}^l Y_u &= \frac{A_2}{A_{34}} {}^w P_w + \frac{A_{24}}{A_{34}} - \frac{A_3}{A_{34}} {}^w P_w {}^l Y_u = T_3^w P_w + C_2 - T_2^w P_w {}^l Y_u \end{aligned} \quad (3.22)$$

En introduisant la notation $X = [T_1 \ T_2 \ T_3 \ C_1 \ C_2]$, l'équation se met sous la forme : $B = QX$ avec

$$Q = \begin{bmatrix} \dots & & & & & \\ {}^w P_w^t & -{}^l X_u^w P_w^t & 0_{1 \times 3} & 1 & 0 & \\ 0_{1 \times 3} & -{}^l Y_u^w P_w^t & {}^w P_w^t & 0 & 1 & \\ \dots & & & & & \end{bmatrix} \text{ et } B = \begin{bmatrix} \dots \\ {}^l X_u \\ {}^l Y_u \\ \dots \end{bmatrix}.$$

On en déduit la valeur de X : $X = (Q_t Q)^{-1} Q^t B$

On peut alors extraire les paramètres de la caméra en utilisant les équations précédentes :

$$\begin{aligned} T_1 &= \frac{r_3}{t_z} \cdot cc(1) + \frac{r_1}{t_t} f_c(1) \\ T_2 &= \frac{r_3}{t_z} \\ T_3 &= \frac{r_3}{t_z} \cdot cc(2) + \frac{r_2}{t_t} f_c(2) \\ C_1 &= cc(1) + \frac{t_x}{t_z} \cdot f_c(1) \\ C_2 &= cc(2) + \frac{t_y}{t_z} \cdot f_c(2) \end{aligned} \quad (3.23)$$

La matrice de rotation est orthonormée. Considérant que $r_3 = 1$, on déduit que :

– les paramètres intrinsèques s'expriment ainsi :

$$\begin{aligned} t_z &= \frac{1}{\|T_2\|} \\ cc(1) &= \frac{T_1 T_2^t}{\|T_2\|^2} \\ cc(2) &= \frac{T_1 T_3^t}{\|T_2\|^2} \\ f_c(1) &= -\frac{\|T_1^t \wedge T_2^t\|}{\|T_2\|^2} \\ f_c(2) &= -\frac{\|T_2^t \wedge T_3^t\|}{\|T_2\|^2} \end{aligned} \quad (3.24)$$

– les paramètres extrinsèques de la matrice de rotation

$$\begin{aligned} r_1 &= -\frac{\|T_2\|}{\|T_1^t \wedge T_2^t\|} \left(T_1 - \frac{T_1 T_2^t}{\|T_2\|^2} T_2 \right) \\ r_2 &= -\frac{\|T_2\|}{\|T_2^t \wedge T_3^t\|} \left(T_3 - \frac{T_2 T_3^t}{\|T_2\|^2} T_2 \right) \\ r_3 &= \frac{T_2}{\|T_2\|} \end{aligned} \quad (3.25)$$

– le vecteur de translation

$$\begin{aligned} t_x &= -\frac{\|T_2\|}{\|T_1^t \wedge T_2^t\|} \left(C_1 - \frac{T_1 T_2^t}{\|T_2\|^2} \right) \\ t_z &= -\frac{\|T_2\|}{\|T_2^t \wedge T_3^t\|} \left(C_2 - \frac{T_2 T_3^t}{\|T_2\|^2} \right) \\ r_3 &= \frac{1}{\|T_2\|} \end{aligned} \quad (3.26)$$

3.1.7.2 Optimisation

La seconde étape diffère par le fait que la notion de distorsion est introduite dans le modèle de Faugeras Toscani. L'équation 3.22 devient alors :

$$\begin{aligned} {}^c X_d + {}^c X_d k_1 r^2 &= f \frac{r_{11}^w X_w + r_{12}^w Y_w + r_{13}^w Z_w + t_x}{r_{31}^w X_w + r_{32}^w Y_w + r_{33}^w Z_w + t_z} \\ {}^c Y_d + {}^c Y_d k_1 r^2 &= f \frac{r_{21}^w X_w + r_{22}^w Y_w + r_{23}^w Z_w + t_y}{r_{31}^w X_w + r_{32}^w Y_w + r_{33}^w Z_w + t_z} \end{aligned} \quad (3.27)$$

$$\text{avec } r = \sqrt{{}^c X_d^2 + {}^c Y_d^2}$$

Il faut transformer les coordonnées métriques en coordonnées pixels : ${}^l X_d = -k_u^c X_d + cc(1)$ et ${}^l Y_d = -k_v^c X_d + cc(2)$ avec (k_u, k_v) les paramètres de transformation.

En introduisant la notation, on obtient alors $x = (\alpha, \beta, \gamma, t_x, t_y, t_z, k_u, k_v, cc(1), cc(2), k_1)^t$

$$\begin{aligned} U(x) &= f \frac{r_{11}^w X_w + r_{12}^w Y_w + r_{13}^w Z_w + t_x}{r_{31}^w X_w + r_{32}^w Y_w + r_{33}^w Z_w + t_z} - \frac{{}^l X_d - cc(1)}{-k_u} \\ &\quad - k_1 \left(\left(\frac{{}^l X_d - cc(1)}{-k_u} \right)^2 + \left(\frac{{}^l Y_d - cc(2)}{-k_v} \right)^2 \right) \cdot \frac{{}^l X_d - cc(1)}{-k_u} \end{aligned} \quad (3.28)$$

$$\begin{aligned} V(x) &= f \frac{r_{21}^w X_w + r_{22}^w Y_w + r_{23}^w Z_w + t_y}{r_{31}^w X_w + r_{32}^w Y_w + r_{33}^w Z_w + t_z} - \frac{{}^l Y_d - cc(2)}{-k_v} \\ &\quad - k_1 \left(\left(\frac{{}^l X_d - cc(1)}{-k_u} \right)^2 + \left(\frac{{}^l Y_d - cc(2)}{-k_v} \right)^2 \right) \cdot \frac{{}^l Y_d - cc(2)}{-k_v} \end{aligned}$$

Pour résoudre ce système, il est nécessaire d'appliquer les équations précédentes aux n points utilisés pour l'étalonnage. On introduit alors comme notation la fonction symbolique $G(x)$ et de sa matrice de dérivées partielles J telle que :

$$G(x_{k-1}) = \begin{pmatrix} U_1(x_{k-1}) \\ V_1(x_{k-1}) \\ \dots \\ V_n(x_{k-1}) \end{pmatrix} \text{ et } J = \begin{pmatrix} \frac{\partial U_1(x_{k-1})}{\partial \alpha} & \frac{\partial U_1(x_{k-1})}{\partial \beta} & \dots & \frac{\partial U_1(x_{k-1})}{\partial k_1} \\ \frac{\partial V_1(x_{k-1})}{\partial \alpha} & \frac{\partial V_1(x_{k-1})}{\partial \beta} & \dots & \frac{\partial V_1(x_{k-1})}{\partial k_1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial V_n(x_{k-1})}{\partial \alpha} & \frac{\partial V_n(x_{k-1})}{\partial \beta} & \dots & \frac{\partial V_n(x_{k-1})}{\partial k_1} \end{pmatrix}$$

Les paramètres du projecteur sont alors obtenus en appliquant une pseudo-inverse à chaque itération de l'équation suivante :

$$\Delta x_k = - (J^t J)^{-1} J^t G(x_{k-1}) \quad (3.29)$$

avec $x_k = x_{k-1} + \Delta x_k$

3.1.7.3 Stéréovision ou triangulation

La figure 3.4 montre la situation d'un système de deux caméras étalonnées : un même point P apparaît sur les deux images, les points de coordonnées (x_p^1, y_p^1) et (x_p^2, y_p^2) .

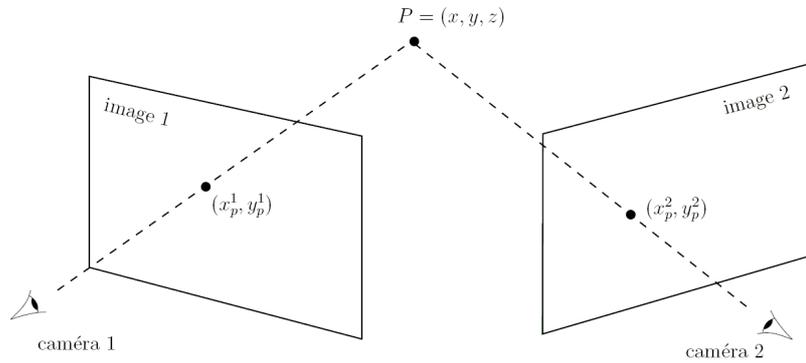


Figure 3.4 – stéréovision

L'équation reliant le point P à l'une de ses images est :

$$\begin{bmatrix} x_p^k \\ y_p^k \\ 1 \end{bmatrix} = A^k \cdot P \quad (3.30)$$

, avec $k = 1, 2$ On note la matrice d'étalonnage $A : A^k = [a_{ij}^k]$ avec $i, j = 1 \dots 3$

L'équation se développe alors sous la forme :

$$\begin{aligned}
(a_{1,1}^k - x_p^k \cdot a_{3,1}^k) \cdot x + (a_{1,2}^k - x_p^k \cdot a_{3,2}^k) \cdot y + (a_{1,3}^k - x_p^k \cdot a_{3,3}^k) \cdot z + (a_{1,4}^k - x_p^k \cdot a_{3,4}^k) &= 0 \\
(a_{2,1}^k - x_p^k \cdot a_{3,1}^k) \cdot x + (a_{2,2}^k - x_p^k \cdot a_{3,2}^k) \cdot y + (a_{2,3}^k - x_p^k \cdot a_{3,3}^k) \cdot z + (a_{2,4}^k - x_p^k \cdot a_{3,4}^k) &= 0 \\
(a_{1,1}^k - x_p^k \cdot a_{2,1}^k) \cdot x + (a_{1,2}^k - x_p^k \cdot a_{2,2}^k) \cdot y + (a_{1,3}^k - x_p^k \cdot a_{2,3}^k) \cdot z + (a_{1,4}^k - x_p^k \cdot a_{2,4}^k) &= 0
\end{aligned} \tag{3.31}$$

On peut alors mettre cette équation de 6 équations à 6 inconnues sous la forme ;

$$U.P = 0 \tag{3.32}$$

$$\text{avec } U = \begin{bmatrix}
(a_{1,1}^1 - x_p^1 \cdot a_{3,1}^1) & (a_{1,2}^1 - x_p^1 \cdot a_{3,2}^1) & (a_{1,3}^1 - x_p^1 \cdot a_{3,3}^1) & (a_{1,4}^1 - x_p^1 \cdot a_{3,4}^1) \\
(a_{2,1}^1 - x_p^1 \cdot a_{3,1}^1) & (a_{2,2}^1 - x_p^1 \cdot a_{3,2}^1) & (a_{2,3}^1 - x_p^1 \cdot a_{3,3}^1) & (a_{2,4}^1 - x_p^1 \cdot a_{3,4}^1) \\
(a_{1,1}^1 - x_p^1 \cdot a_{2,1}^1) & (a_{1,2}^1 - x_p^1 \cdot a_{2,2}^1) & (a_{1,3}^1 - x_p^1 \cdot a_{2,3}^1) & (a_{1,4}^1 - x_p^1 \cdot a_{2,4}^1) \\
(a_{1,1}^2 - x_p^2 \cdot a_{3,1}^2) & (a_{1,2}^2 - x_p^2 \cdot a_{3,2}^2) & (a_{1,3}^2 - x_p^2 \cdot a_{3,3}^2) & (a_{1,4}^2 - x_p^2 \cdot a_{3,4}^2) \\
(a_{2,1}^2 - x_p^2 \cdot a_{3,1}^2) & (a_{2,2}^2 - x_p^2 \cdot a_{3,2}^2) & (a_{2,3}^2 - x_p^2 \cdot a_{3,3}^2) & (a_{2,4}^2 - x_p^2 \cdot a_{3,4}^2) \\
(a_{1,1}^2 - x_p^2 \cdot a_{2,1}^2) & (a_{1,2}^2 - x_p^2 \cdot a_{2,2}^2) & (a_{1,3}^2 - x_p^2 \cdot a_{2,3}^2) & (a_{1,4}^2 - x_p^2 \cdot a_{2,4}^2)
\end{bmatrix}$$

Ce système peut être résolu par la méthode des moindres carrés.

3.1.7.4 Extraction des points d'étalonnage du projecteur

L'étalonnage du projecteur ne peut utiliser la même technique que celle des caméras, car le projecteur ne fournit pas d'image de la scène. Par contre l'image projetée est parfaitement connue car elle peut être générée pour répondre à nos besoins.

La première difficulté est d'extraire l'image projetée de la scène observée par la caméra : du format classique RGB (Red, Green, Blue) pour définir l'image nous convertissons l'image au format HSL (Hue, Saturation, Luminance) et nous utilisons l'image de luminance : Plus le pixel est blanc, plus il est éclairé. Un seuillage basé sur la luminance nous permettra d'extraire la mire projetée de la scène. Nos points d'intérêt dans l'image projetée doivent donc se situer dans les zones les plus lumineuses de l'image Une mire binaire (figure 3.5) convient donc à nos besoins.

1. La conversion d'une image RGB en HSL est la première étape de l'étalonnage de notre projecteur.
2. Un seuillage de l'image avec un seuil choisi empiriquement permet de sélectionner les zones d'intérêt de l'image.

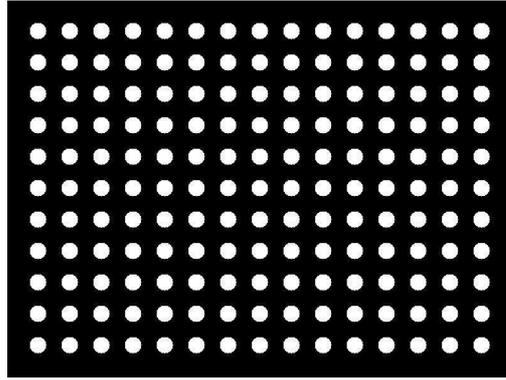


Figure 3.5 – Mire d'étalonnage du projecteur

3. La détection des centres des disques utilise l'algorithme de la distance de Danielsson [Danielsson, 1980], qui détecte des formes grâce à une *danielsson distance map*, image dont le pixel code la distance.
4. La numérotation des disques se fait suivant la taille du rayon, de la surface et du paramètre des cercles. Il ne correspond donc pas à celui de la création de la mire qui numérote les disques de la gauche vers la droite, puis de haut en bas. Cette numérotation est capitale car elle détermine la correspondance entre points 3D et points 2D. Une erreur de correspondance (au minimum deux dans le meilleur des cas) aurait un effet de cascade en décalant la numérotation des points d'intérêt et entraînerait ainsi une erreur d'étalonnage du projecteur et nous empêcherait d'atteindre notre objectif final de texturation d'objet.

La figure 3.6 reprend les différentes étapes de la détermination des points de correspondance.

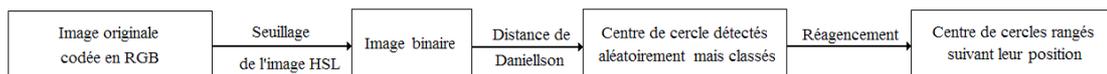


Figure 3.6 – Différentes étapes d'étalonnage du projecteur

La numérotation des cercles se base sur l'hypothèse que les angles de rotation de la matrice de passage du plan image du projecteur au plan de projection sont faibles, ainsi on peut faire les hypothèses suivantes :

- Les n points d'abscisse la plus faible font partie de la première colonne de la mire, la mire étant composé de $n \times m$ disques.
- Les m points d'ordonnée la plus faible font partie de la première ligne, et ainsi de suite.
- Le premier point de la i_{me} ligne correspond bien au i_{me} élément de la première colonne.

La figure 3.7 reprend le même système de numérotation que celui utilisé lors de la création de la mire.

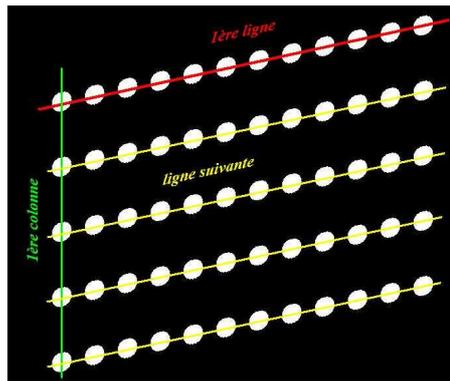


Figure 3.7 – Identification des lignes et de la première colonne sur la mire

La figure 3.8 montre la détection et l'appariement des points projetés vus par les caméras.

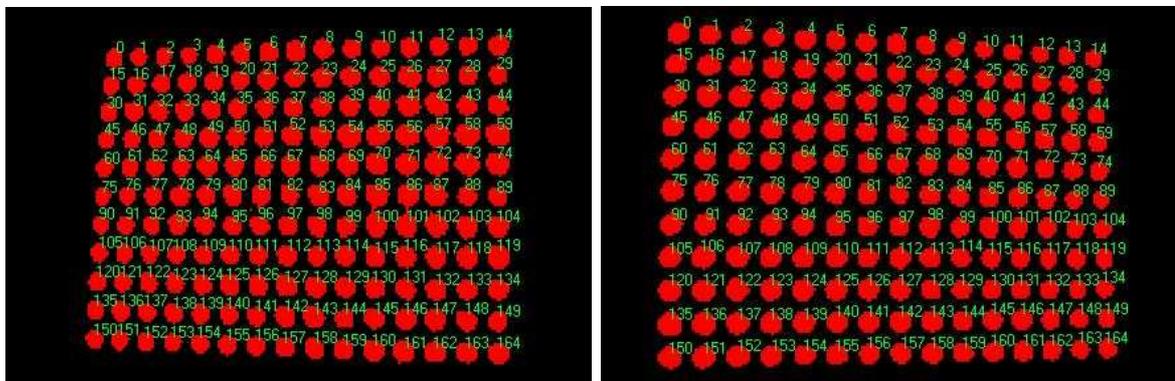


Figure 3.8 – Identification des points de correspondance de la mire projetée pour l'étalonnage du projecteur

3.1.8 Système à n caméra(s) et m projecteur(s)

L'étalonnage d'un système composé de deux caméras et d'un projecteur vient d'être décrit. Cette méthode peut évidemment se généraliser à un système composé d'un nombre plus élevé de caméras et de projecteurs. En effet on peut généraliser le système pour trois caméras ou plus, en employant la même méthode de stéréo-étalonnage en prenant soin d'avoir les matrices de rotation et translation entre les différents référentiels de nos systèmes de deux caméras, voire conserver le même référentiel pour chaque binôme de caméras afin de minimiser les erreurs d'étalonnage et ne pas les additionner pour chaque nouveau binôme. L'étalonnage du projecteur nécessite deux caméras du système.

Si l'utilisation d'un seul projecteur ne limite pas notre méthode, par contre l'emploi d'une seule caméra n'est pas possible avec la méthode proposée : il faudrait soit ajouter une caméra

pour étalonner le projecteur, soit utiliser une autre méthode d'étalonnage de système mono-caméra/mono-projecteur comme la méthode du quadrangle [Chen and Kak, 1987], qui permet d'étalonner un système laser. Il faudrait donc considérer le motif projeté comme plusieurs lasers plans. Une deuxième méthode qui se montre moins exhaustive, consisterait à ajouter une contrainte de distance du plan par rapport au système pour résoudre les équations [Raskar and Beardsley, 2001], [Hamada and Sato, 2007].

3.2 Géométrie épipolaire

3.2.1 Théorie

L'application de la géométrie projective à la stéréovision n'est pas nouvelle et remonte au 19ème siècle pour la photographie. La vision assistée par ordinateur a ravivé l'intérêt pour le sujet par les travaux importants dans le domaine des invariants et de la reconstruction projectifs par [Faugeras and Papadopoulos, 1995] et [Hartley and Zisserman, 1994].

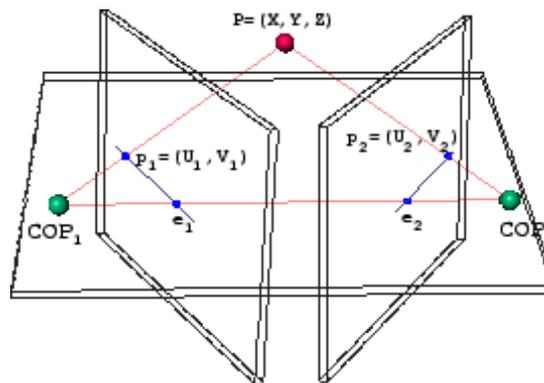


Figure 3.9 – Géométrie épipolaire pour deux vues

La figure 3.9 montre la situation de la formation d'image pour la stéréovision. Trois points, les deux centres des caméras CO_1 et CO_2 et le point 3D P forment un plan épipolaire et les intersections de ce plan avec les deux plans image forment les lignes épipolaires. La ligne reliant les deux centres de la projection coupe les plans image aux points e_1 et e_2 , appelés épipoles.

Le point 3D projette dans les deux plans image les points p_1 et p_2 qui sont représentés en coordonnées homogènes par $\begin{bmatrix} u_1 & v_1 & 1 \end{bmatrix}^t$ et $\begin{bmatrix} u_2 & v_2 & 1 \end{bmatrix}^t$ respectivement. La relation principale de la géométrie épipolaire est :

$$p_1^t F p_2 = 0 \quad (3.33)$$

F est la matrice fondamentale, de taille 3×3 avec 9 paramètres, ne possède que sept degrés de liberté.

Elle définit la géométrie des correspondances entre deux vues d'une manière compacte, codant la géométrie intrinsèque de la caméra et le mouvement relatif extrinsèque entre deux vues. La structure de la scène est éliminée de l'évaluation de F et peut être estimée dans une étape suivante. Connaissant la matrice F , l'identification d'un point dans une image permet l'estimation de la ligne épipolaire correspondante dans l'autre image.

Hartley propose une technique [Hartley, 1995] pour déterminer les paramètres de la matrice fondamentale quand on observe au moins 8 points. L'extension de l'expression dans l'équation 3.34 donne une contrainte linéaire sur F par point observé (équation 3.33). La combinaison de N équations à N inconnues correspond à un système linéaire de la forme $Af = 0$.

$$u_1 u_2 f_{11} + u_1 v_1 f_{12} + u_1 f_{13} + v_1 u_2 f_{21} + v_2 u_2 f_{22} + v_1 f_{23} + u_2 f_{31} + v_2 f_{32} + f_{33} = 0 \quad (3.34)$$

On résout un système linéaire en utilisant plus de 8 points dans une minimisation des moindres carrés $\min \|Af\|^2$ sujet à la contrainte $\|f\| = 1$.

Cette contrainte fixe la norme de la matrice fondamentale. La contrainte de rang 2 doit également être imposée [Hartley and Zisserman, 2004].

La matrice fondamentale F est récupérée indépendamment de la structure et peut être utile seule, par exemple pour l'estimation du mouvement d'un robot [Armangué et al., 2003]. Hartley utilise la dérivation des équations de Kruppa pour déterminer les paramètres intrinsèques d'un système de prise de vue. Il devient alors possible de récupérer les coordonnées 3D euclidiennes de la structure.

Il est intéressant d'étudier la stabilité de telles techniques. Il faut considérer le cas où les centres de projection des deux images sont près les uns des autres. La dégénérescence se produit lorsque les centres se recouvrent. Ce cas se produit si le mouvement entre ces centres est une pure rotation. La dégénérescence se produit également quand tous les points 3D dans la scène sont coplanaires. La linéarisation du calcul de la matrice fondamentale crée ces dégénérescences. Une méthode pour surmonter ces dégénérescences est fournie par [Torr and Murray, 1997].

La formulation épipolaire linéaire de la géométrie montre la sensibilité au bruit dans les mesures d'image 2D, comparée aux approches non linéaires. En effet chaque point peut correspondre à

n'importe quel point de la ligne épipolaire. Ainsi, les solutions tendent à produire des erreurs résiduelles élevées selon les lignes épipolaires et la reconstruction qu'elle permet reste sensible au bruit et peu précise.

3.2.2 Géométrie épipolaire pour 2 vues : la matrice fondamentale

Si les méthodes linéaires pour estimer la géométrie épipolaire se caractérisent par un temps de calcul faible, leur faible résistance face aussi bien à des points aberrants qu'au bruit de l'image réduit leur intérêt. Il faut alors se tourner vers les méthodes itératives pour obtenir des résultats de bonne qualité. Les méthodes itératives peuvent se classer en deux groupes : les méthodes qui réduisent la distance entre les points et leur ligne épipolaire et celles basées sur le gradient. Armangue *et al.* présentent une revue générale des différentes méthodes [Armangue and Salvi, 2003] mais nous ne nous intéresserons à la méthode qui donne sensiblement les meilleurs résultats, la méthode Least-Median-Squares (LMedS) [Zhang, 1996]. Elle estime les paramètres en résolvant le problème de minimisation non linéaire :

$$\min(\text{median}_i r_i^2) \quad (3.35)$$

L'estimateur doit atteindre la plus petite valeur du médian des résidus au carré, estimée pour l'ensemble des n points de correspondance. L'algorithme se décompose ainsi :

1. Un nombre m de jeu de sept points, soit le minimum requis pour estimer la matrice fondamentale, est choisi.
2. Pour chaque jeu, la matrice fondamentale F_j est estimée.
3. Pour chaque matrice F_j , le médian des résidus au carré, noté M_j , est estimé :

$$M_j = \text{median}_{i=1\dots n} \left[d^2 \left(\tilde{m}_i F_j \tilde{m}_i^t \right) + d^2 \left(\tilde{m}_i^t F_j \tilde{m}_i \right) \right] \quad (3.36)$$

4. Conserver la matrice F_j qui minimise M_j

La figure 3.10 présente l'algorithme de la méthode LMedS.

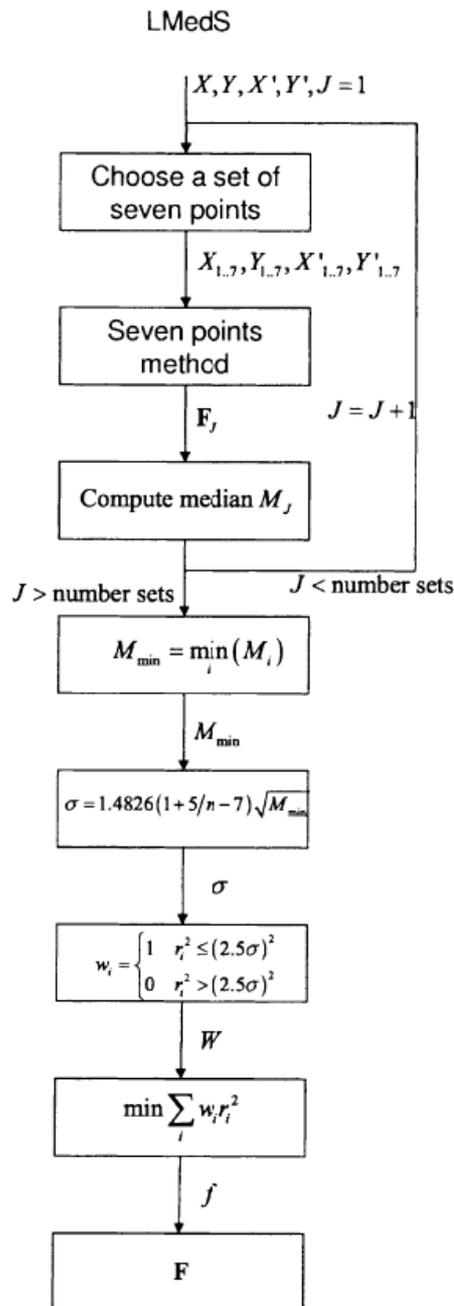


Figure 3.10 – Estimation de la matrice fondamentale [Armangue and Salvi, 2003]

La méthode se révélant peu robuste face au bruit gaussien, une étape supplémentaire est ajoutée pour compenser cette faiblesse. La notion de poids w_i est introduite et estimée par

$$w_i = \begin{cases} 1 & \text{si } r_i^2 \leq 2,5.\hat{\sigma}^2 \\ \text{sinon} & 0 \end{cases} \quad (3.37)$$

avec $\hat{\sigma}$ la déviation standart,

$$\hat{\sigma} = 1.42826 \frac{1}{(n-p)} \sqrt{M_j}$$

On optimise ensuite la matrice fondamentale F en résolvant le problème de moindres carrés pondérés :

$$\min \sum_i w_i.r_i^2 \quad (3.38)$$

Il reste donc à déterminer le nombre m de jeux de points de correspondance et la méthode pour choisir chaque jeu, soit sept points, nombre de points minimum pour estimer la matrice fondamentale.

Pour déterminer m , on considère l'échantillon comme valable s'il contient p bonnes correspondances. Si on suppose qu'un jeu de points de correspondance contient au pire un pourcentage de ϵ de points aberrants, la probabilité qu'au moins une des m combinaisons de points soit bonne est de :

$$P = 1 - [1 - (1 - \epsilon)^p]^m \quad (3.39)$$

Si on veut que P soit proche de 1, on peut alors estimer m en choisissant des valeurs pour p et ϵ :

$$m = \frac{\log(1 - P)}{\log[1 - (1 - \epsilon)^p]} \quad (3.40)$$

3.2.2.1 Estimation de matrices fondamentales

L'estimation de matrices fondamentales (entre les deux caméras et entre une caméra et un projecteur) nécessite des points de correspondances entre les images. De tels points sont déjà connus : ceux utilisés lors de l'étalonnage du système, à savoir les points des mires pour la

matrice fondamentale entre les caméras et les points de la mire projetée pour les matrices fondamentales entre le projecteur et une caméra.

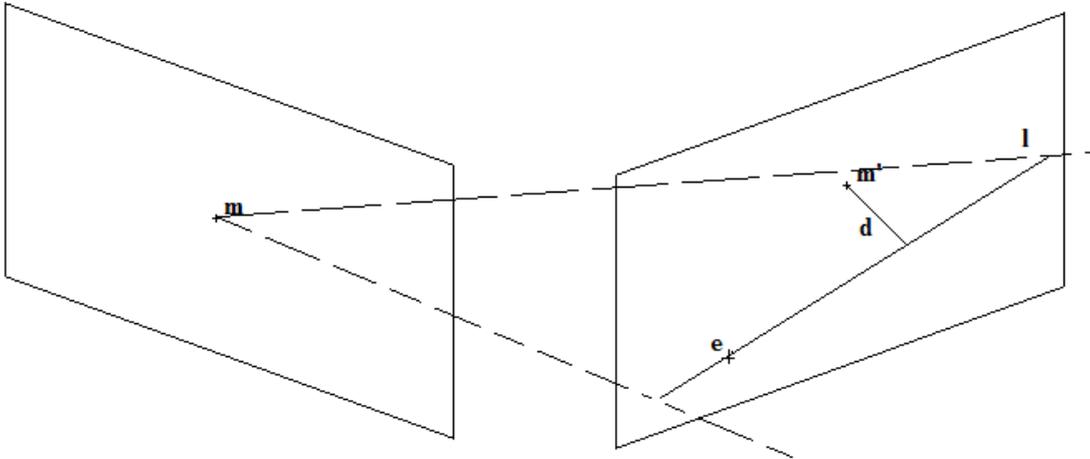


Figure 3.11 – Géométrie épipolaire pour deux vues

Pour estimer la précision de ces matrices, la distance épipolaire d sera estimée : elle mesure la distance entre un point m et sa ligne épipolaire l (figure 3.11). Cette ligne épipolaire est estimée à partir d'un point de correspondance et de la matrice fondamentale, par la relation :

$$l = F.m \quad (3.41)$$

Le tableau 3.1 présente les distances moyennes, minimum et maximum épipolaires pour chaque doublet du système . Des erreurs de même ordre de grandeur sont observées.

objet	caméras d et g	caméra g et projo	caméra d et projo
distance moyenne en pixel	0.4323	0.4911	0.451
distance minimum	0.0000	0.0015	0.0000
distance maximum	3.454	2.4419	2.1291

TAB. 3.1 – Estimation de distance épipolaire en mm

La distorsion de la lentille des caméras ne peut être ignorée. On corrige les positions des points d'intérêt avant l'étalonnage, et seulement ensuite les matrices fondamentales peuvent être estimées. La correction de la distorsion n'est pas la seule étape pour améliorer l'estimation des matrices. La normalisation introduite par Hartley [Hartley, 1995], en normalisant les points et recentrant leur point de gravité, permet d'éviter certains problèmes (mauvais conditionnement des points, etc ...).

3.2.3 Géométrie trifocale pour trois vues : le tenseur trifocal

La géométrie épipolaire (figure 3.12) décrit les relations de correspondance entre les projections d'un point 3D sur plusieurs plans images. La matrice fondamentale ne décrit que la correspondance entre un point et une ligne pour deux images. Le tenseur trifocal est un cube $3 \times 3 \times 3$ qui décrit la géométrie épipolaire pour trois vues comme la matrice fondamentale la décrit pour deux. Il est composé de vingt sept éléments mais ne possède que dix huit degrés de liberté car il doit satisfaire des contraintes internes. Il permet de mettre en correspondance ligne et point entre une image et les deux autres : trois points, trois lignes, deux points et une ligne, etc... [Hartley and Zisserman, 2004].

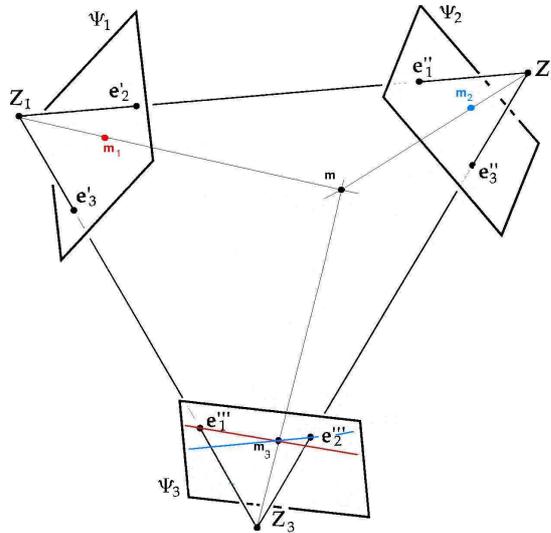


Figure 3.12 – Géométrie épipolaire pour trois vues

e sont les épipoles, Z le centre de projection du plan image 2. La figure 3.12 décrit comment, à partir de deux correspondances de deux images différentes, les deux lignes épipolaires correspondantes sur la troisième image peuvent être estimées. Les coordonnées du point de correspondance sur la troisième image, qui se trouve à l'intersection des deux lignes épipolaires, peuvent ainsi être calculées [Ressl, 2003].

Les matrices de projection peuvent s'écrire sous la forme :

$$\begin{aligned}
 P &= \begin{bmatrix} I_d & | & 0 \end{bmatrix} \\
 P' &= K' \begin{bmatrix} R' & | & T' \end{bmatrix} \\
 P'' &= K'' \begin{bmatrix} R'' & | & T'' \end{bmatrix}
 \end{aligned} \tag{3.42}$$

La caméra gauche est choisie comme origine de notre système. Ainsi la matrice de rotation T_{left} et le vecteur de translation R_{left} peuvent être initialisés $T_{left} = 0$ et $R_{left} = I_d$. Mais la matrice d'étalonnage de la caméra ne peut être la matrice identité I_d . Le formalisme de l'estimation du tenseur trifocal impose que $P = \begin{bmatrix} I_d & | & 0 \end{bmatrix}$, une homographie sera appliquée sur la première image K^{-1} , K matrice d'étalonnage de la caméra droite. La convention d'Einstein est observée : les indices répétés dans les positions de contravariants et de covariants impliquent qu'il fasse faire la somme suivant cet indice. Ainsi par exemple $x'^i = \sum_j a'_j \cdot x^j = a^i_j \cdot x^j$.

Le tenseur trifocal peut alors se déterminer suivant :

$$T_i^{kj} = v'^j b_i^k - v''^k a_i^j \quad (3.43)$$

en écrivant les matrices sous la forme $P' = \begin{bmatrix} a_i^j & | & v' \end{bmatrix}$ et $P'' = \begin{bmatrix} b_i^j & | & v'' \end{bmatrix}$

La notation tensorielle se généralise à n vues (tenseur n -focal) sans changer de formalisme ni d'écriture.

3.2.4 Relations entre le tenseur trifocal et la matrice fondamentale

On peut extraire les matrices fondamentales F_{31} et F_{21} du tenseur trifocal [Hartley and Zisserman, 2004], ainsi que les épipoles. Tenseur trifocal et matrices fondamentales peuvent tous les deux permettre d'estimer la position d'un point x'' sur la troisième image si ses deux points de correspondance, x et x' , sont connus sur les deux images. La relation peut alors se décrire sous deux équations :

$$x'' = (F_{31}x) \times (F_{32}x') \quad (3.44)$$

ou bien

$$x''^k = x^k l'_j T_i^{jk} \quad (3.45)$$

avec l' la ligne perpendiculaire à la ligne épipolaire à $l = [l_1, l_2, l_3]$ et passant par x' , soit $l' = [l_2, -l_1, -x_1 \cdot l_2 + x_2 \cdot l_1]$

Mais l'inverse est également possible : on peut exprimer le tenseur trifocal à partir des matrices fondamentales [Shen et al., 1998]. A partir des matrices fondamentales on peut déterminer

les matrices d'étalonnage à une homographie près : nous profitons de cette incertitude pour respecter les équations 3.42 et définir la première matrice de projection par $P = \begin{bmatrix} I_d & | & 0 \end{bmatrix}$. Ensuite nous estimons P' et P'' par :

$$P' = [m'|e'] \quad (3.46)$$

avec e' obtenu suivant $F^t e' = 0$ et $F = [e']_x m'$, avec $[a]_x b = a \times b$

En résumé pour déduire le tenseur trifocal à partir des matrices fondamentales, Shen utilise la méthode suivante :

1. Estimation des matrices fondamentales à partir des points de correspondances (figure 3.10) basé sur la méthode LMedS [Zhang, 1996]
2. Estimation des matrices de projection à partir des matrices fondamentales (Equation 3.46)
3. Estimation du tenseur trifocal à partir des matrices de projection (Equation 3.43)

3.3 Précision du système

L'étalonnage du système représente la première étape dans notre travail. La figure 3.13 montre le système étalonné : les positions des caméras et du projecteur sont estimées, ainsi que celles des mires utilisées.

On rappelle succinctement la méthode utilisée pour étalonner le système :

1. le stéréo étalonnage des caméras est estimée,
2. les paramètres intrinsèques et extrinsèques du projecteur sont déterminés.

Les paramètres intrinsèques de la caméra de gauche estimées sont :

- distance focale : $f_c = \begin{bmatrix} 401.50925 & 398.61448 \end{bmatrix} \pm \begin{bmatrix} 3.23408 & 3.45448 \end{bmatrix}$
- Point principal : $cc = \begin{bmatrix} 189.02623 & 111.38242 \end{bmatrix} \pm \begin{bmatrix} 4.25028 & 3.57726 \end{bmatrix}$
- obliquité : $\alpha_c = [0.00000] \pm [0.00000] \Rightarrow$ pixel carré
- distorsion : $kc = \begin{bmatrix} -0.22948 & -0.66675 & -0.00219 & 0.01327 & 0.00000 \end{bmatrix} \pm \begin{bmatrix} 0.04837 & 0.37344 & 0.00151 & 0.00222 & 0.00000 \end{bmatrix}$

Les paramètres intrinsèques de la caméra de droite estimées sont :

- distance focale : $f_c = \begin{bmatrix} 403.36713 & 400.55476 \end{bmatrix} \pm \begin{bmatrix} 3.31348 & 3.49065 \end{bmatrix}$

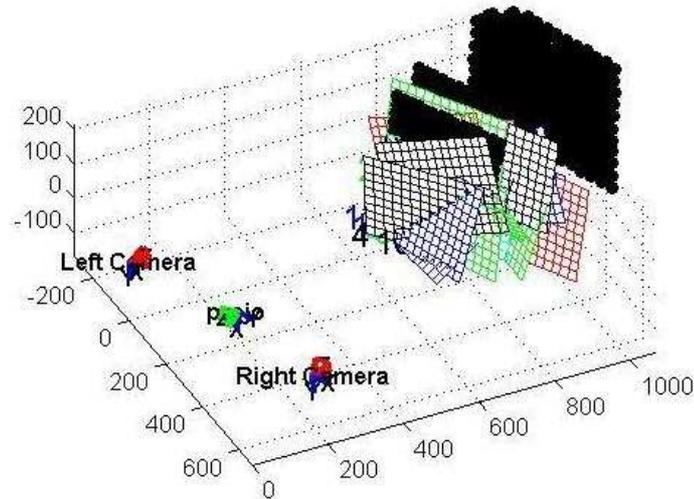


Figure 3.13 – Étalonnage du système : Les caméras sont représentées en rouge, le projecteur en vert, les mire pour étalonner les caméras en couleur, et les mire du projecteur sont en noir. Les distances sont exprimées en mm

- Point principal : $cc = \begin{bmatrix} 182.47323 & 111.56900 \end{bmatrix} \pm \begin{bmatrix} 4.55293 & 3.48012 \end{bmatrix}$
- obliquité : $\alpha_c = \begin{bmatrix} 0.00000 \end{bmatrix} \pm \begin{bmatrix} 0.00000 \end{bmatrix} \Rightarrow$ pixel carré
- distorsion : $kc = \begin{bmatrix} -0.29190 & 0.05254 & -0.00610 & 0.00328 & 0.00000 \end{bmatrix} \pm \begin{bmatrix} 0.03216 & 0.13226 & 0.00136 & 0.00249 & 0.00000 \end{bmatrix}$

les paramètres extrinsèques (position de la caméra de droite par rapport à la gauche) :

- Vecteur de rotation : $om = \begin{bmatrix} 0.01882 & 0.79574 & -0.22490 \end{bmatrix} \pm \begin{bmatrix} 0.01110 & 0.01462 & 0.00547 \end{bmatrix}$
- Vecteur de translation : $T = \begin{bmatrix} -535.35318 & 51.22905 & 182.03962 \end{bmatrix} \pm \begin{bmatrix} 3.22017 & 2.07910 & 7.50052 \end{bmatrix}$

les paramètres extrinsèques et intrinsèques du projecteur sont

- $R = \begin{bmatrix} -0.8697 & -0.1428 & -0.4725 \\ -0.0189 & -0.9469 & 0.3209 \\ -0.4932 & 0.2880 & 0.8209 \end{bmatrix}$
- $T = \begin{bmatrix} 302.1010 & 16.5098 & 493.6736 \end{bmatrix}$
- $au = -2.1899e + 003$
- $av = -2.1733e + 003$
- $u0 = 518.2337$
- $v0 = 845.3355$
- $k1_{projo} = 0.0241$
- $iter = 8$

La caméra de gauche est choisie comme repère de notre système pour tous nos tests. Cela nous permet de retrouver les hypothèses de matrices d'étalonnage pour l'estimation du tenseur trifocal.

Aux erreurs près d'estimation des paramètres intrinsèques des caméras, nous retrouvons sensiblement les mêmes résultats d'un test à l'autre, lorsque nous modifions la disposition du système.

L'ordre de grandeur est similaire entre nos mesures manuelles assez grossières et les valeurs estimées par l'ordinateur pour les paramètres extrinsèques. La figure 3.13 montre aussi que nous retrouvons le projecteur placé au milieu des deux caméras, les trois composants orientés vers les mires.

La distorsion a été estimée pour les caméras (tangentielle et radiale) et pour le projecteur (radiale). Si on compare la distorsion radiale, on remarque que le coefficient de distorsion radiale des caméras est dix fois supérieur à celui du projecteur. Cette distorsion est corrigée lors de la triangulation et la rétroprojection de points 3D sur le plan projecteur.

3.3.1 Synthèse de vue

On considère le projecteur comme une caméra placée au même endroit (position et orientation). On peut alors estimer l'image *vue* par le projecteur comme celle prise par cette caméra qui posséderait les mêmes paramètres intrinsèques et extrinsèques que le projecteur. Le tenseur trifocal nous permet alors d'estimer la position d'un point dans la troisième image (l'image du projecteur) si la position des points de correspondance sur les deux images (première et deuxième) est connue.

La figure 3.14 modélise cette relation et l'équation 3.47 permet d'estimer la nouvelle vue :

$$p^i s_j^\mu T_i^{kj} \cong p''^k \quad (3.47)$$

p et p'' sont les points sur la première et troisième image, s est une ligne sur la deuxième image, perpendiculaire à la ligne épipolaire qui est la projection du point p avec la matrice fondamentale F_{21} , qui décrit la géométrie épipolaire entre les deux images (première et deuxième). Il est préférable d'utiliser les lignes perpendiculaires aux lignes épipolaires pour éviter le cas de configurations critiques [Hartley and Zisserman, 2004].

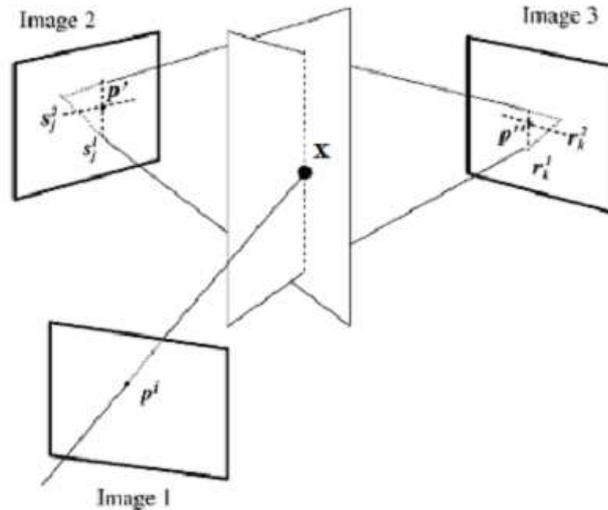


Figure 3.14 – Transfert de point des deux premières images vers la troisième image.

La première étape de la méthode est d'estimer les points de correspondance entre les deux images des caméras. Puisqu'une mise en correspondance dense est préférable pour synthétiser une nouvelle image, la carte de disparité devrait être estimée. Les images devraient être alors rectifiées, c'est à dire qu'il faudrait transformer les images pour faire en sorte qu'un point de la première image et sa ligne épipolaire correspondante sur le deuxième image se trouvent sur la même ligne : le résultat est que les lignes épipolaires sont horizontales et les épipoles se trouvent à l'infini. L'image synthétisée peut être considérée comme une troisième image et l'emploi du tenseur trifocal devient alors possible : une nouvelle vue peut être synthétisée par la méthode de transfert de point. Cette nouvelle vue peut être ensuite projetée sur un objet réel.

Considérons une correspondance entre les deux images (première et deuxième) avec les points $x \leftrightarrow x'$. Soit la ligne l' passant par le point x' , alors le point correspondant x'' sur la troisième image peut être estimé par le transfert du point x de la première image sur la troisième image par l'équation

$$x''^k = x^i l'_j T_i^{jk} \quad (3.48)$$

Le choix de la ligne l' est donc primordiale. Pour éviter des cas de dégénérescence, on choisit la ligne perpendiculaire à la ligne épipolaire. Ce transfert de point se décompose en trois étapes :

1. Estimer la matrice fondamentale F_{21} et corriger la correspondance $x \leftrightarrow x'$ par $x \leftrightarrow \hat{x}'$.
2. Estimer la ligne l' passant par \hat{x}' et perpendiculaire à la ligne $l'_e = F_{21}\hat{x}'$.
3. Le point de transfert est alors : $x''^k = \hat{x}'^i l'_j T_i^{jk}$.

3.4 Synthèse

Ce chapitre a permis d'estimer tous les outils qui nous seront nécessaire par la suite :

1. La géométrie épipolaire pour trois vues a été définie et permet de synthétiser des images si les points de correspondance sont fournis.
2. La méthode employée pour étalonner le système. Elle se décompose en deux étapes :
 - Étalonnage des caméras : l'association d'un modèle dérivé de celui de Zhang et un soin particulier apporté à l'estimation des points d'intérêt de la mire pour éviter de propager une erreur en estimant individuellement leurs positions permettent d'obtenir de bons résultats.
 - Étalonnage du projecteur : L'estimation de la position des points d'intérêt de la mire projetée sur les images des caméras par la distance de Daniellson permet suite à une triangulation d'obtenir un jeu de points de correspondance entre les points 2D de la mire et les points 3D estimées. Ce jeu permet d'étalonner le projecteur grâce à l'algorithme de Faugeras itératif. L'erreur de triangulation des points 3D, imputable à l'erreur d'étalonnage des caméras, se retrouve dans l'erreur d'étalonnage du projecteur.

Le problème de trouver des points de correspondance précis est mis en évidence dès les étapes d'étalonnage du système et d'estimation de la géométrie épipolaire. Pour minimiser l'erreur de détection des points 2D, chaque point est estimé séparément. Ce soin s'explique par le fait qu'ils servent non seulement à l'étalonnage du système, mais aussi à l'estimation de la géométrie épipolaire. La détection de points de correspondance sur un objet tridimensionnel est abordé dans le chapitre suivant.

Chapitre 4

Lumière Structurée et Synthèse de vue

Le système est étalonné, les paramètres intrinsèques et extrinsèques des caméras et projecteurs sont estimés, ainsi que les outils utiles à la mise en correspondance comme la matrice fondamentale. Estimer des points de correspondance est désormais possible, à condition de déterminer un point d'intérêt sur la première image pour estimer son point correspondant sur la deuxième image grâce à la géométrie épipolaire.

La détection de points d'intérêt dans une image est un sujet largement traité. En passant du détecteur de Harris, à l'image de disparité et aux problèmes d'occultation, la détection de points d'intérêt est un sujet maîtrisé. La difficulté réside dans la corrélation de ces points d'intérêt, c'est à dire de détecter non seulement le même point sur deux images, mais aussi de l'identifier comme le même point.

Nous présentons deux mires dans ce chapitre : la première mire est une mire binaire, noir et blanc, pour l'adapter à une stratégie de lumière imperceptible (annexe A), la deuxième mire, couleur, est choisie pour sa précision et sa résolution.

4.1 Lumière structurée

Notre système se heurte à deux difficultés pour déterminer des points de correspondance sur deux images :

- L'objectif est d'observer des objets réels dépourvus de texture, par exemple totalement blancs.
- La corrélation de points d'intérêt pour estimer des points de correspondance, qui donne de meilleurs résultats avec des images prises avec des points de vue proches, s'oppose à la

nécessité d'avoir des points de vues éloignés qui donnent une meilleure précision pour la reconstruction.

On considère que le projecteur peut s'employer de deux manières différentes. La première manière, dite passive, consiste à n'utiliser le projecteur que "en lumière texturante". La deuxième manière, dite active, consiste à utiliser le projecteur pour projeter une image codée dont on pourra tirer des informations utiles en lumière structurée.

Pour contourner le problème de détermination des points de correspondance, l'emploi de la technique de la lumière structurée s'impose. Elle consiste à projeter une mire codée à partir de laquelle on peut extraire des points de correspondance. Battle *et al* [Battle et al., 1998] présentent un état de l'art sur cette technique (figure 4.1).

Même si nous nous limitons à des objets statiques pour l'instant, la possibilité de faire un jour bouger les objets ne doit pas être écartée. Les techniques de voisinage spatial, qui se caractérisent entre autre par des techniques dites *one shot*, soit à une seule prise de vue, sont adaptées aux scènes dynamiques. Le deuxième critère à prendre en compte est la profondeur du pixel, à savoir sur combien de bits le pixel est codé, et donc s'il représente la couleur, le niveau de gris ou le noir et blanc. Nous avons choisi une image binaire pour sa simplicité, plus facile à segmenter et à intégrer dans une technique de lumière structurée imperceptible (annexe A). Le dernier critère est la stratégie de codage : le motif de code se répète dans l'image (codage périodique) ou non (codage absolu).

4.1.1 Mire Binaire

La mire choisie possède trois caractéristiques : binaire, à une prise de vue et de codage absolu. Elle fera donc partie de la famille des mires structurés M-arrays.

Une *perfect map* M de facteurs $(r, s; u, v)$, entiers positifs, est définie comme un tableau binaire de période $r \times s$ dans laquelle tous les tableaux de taille $u \times v$ n'apparaissent qu'une seule fois [Etzion, 1988]. Si seul le tableau $u \times v$ composé de zéros n'apparaît pas, alors on parle de *M-array* ou *Pseudorandom array*. Toutefois, la construction de tels tableaux nécessite le respect de certaines contraintes (équation 4.1).

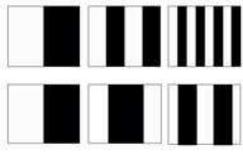
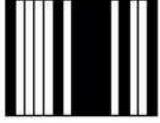
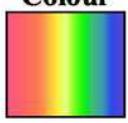
Time-multiplexing	Binary codes 	Posdamer et al. Inokuchi et al. Minou et al. Trobina Valkenburg and McIvor Skocaj and Leonardis Rocchini et al.	√ √ √ √ √ √ √	√ √ √ √ √ √ √	√ √ √ √ √ √ √
	n-ary codes	Caspi et al. Horn and Kiryati	√ √	√ √	√ √
	Gray code + Phase shifting	Bergmann Sansoni et al. Wiora Gühring	√ √ √ √	√ √ √ √	√ √ √ √
	Hybrid methods	Kosuke Sato Hall-Holt and Rusinkiewicz	√ √	√ √	√ √
Spatial Neighborhood	Non-formal codification	Maruyama and Abe Durdle et al. Ito and Ishii Boyer and Kak Chen et al.	√ √ √ √ √	√ √ √ √ √	√ √ √ √ √
	De Bruijn sequences 	Hügli and Maître Monks et al. Vuylsteke and Oosterlinck Salvi et al. Lavoie et al. Zhang et al.	√ √ √ √ √ √	√ √ √ √ √ √	√ √ √ √ √ √
	M-arrays 	Morita et al. Petriu et al. Kiyasu et al. Spoelder et al. Griffin and Yee Davies and Nixon Morano et al.	√ √ √ √ √ √ √	√ √ √ √ √ √ √	√ √ √ √ √ √ √
Direct coding	Grey levels	Carrihill and Hummel Chazan and Kiryati Hung	√ √ √	√ √ √	√ √ √
	Colour 	Tajima and Iwakawa Smutny and Pajdla Geng Wust and Capson Tatsuo Sato	√ √ √ √ √	√ √ √ √ √	√ √ √ √ √
Scene applicability		<i>Static</i> <i>Moving</i>			
Pixel depth		<i>Binary</i> <i>Grey levels</i> <i>Colour</i>			
Coding strategy		<i>Periodical</i> <i>Absolute</i>			

Figure 4.1 – Classification des techniques de lumière structurée proposé par [Salvi et al., 2004]

$$\begin{aligned}
 i) \quad & rs = 2^{uv} \\
 ii) \quad & r > uorr = u = 1 \\
 iii) \quad & s > voru = v = 1
 \end{aligned}
 \tag{4.1}$$

Si l'unicité est un facteur intéressant pour nos travaux, en revanche le fait que toutes les combinaisons soient codées à l'intérieur de la *map M* est superflu. Nous nous contenterons de *perfects submaps*, qui conservent la propriété de cellule unique mais sans coder toutes les combinaisons [Griffin et al., 1992]. Une fois la mire construite, nous ne nous intéressons pas si toutes les combinaisons possibles ont été utilisées ou non.

4.1.1.1 Construction de la mire

Le motif doit être centré sur le pixel testé, donc un chiffre impair pour la taille du motif est nécessaire. On appelle cellule tout carré de cinq par cinq éléments de base dans la mire créée qui constitue le motif élémentaire de la mire. La taille de 5×5 est choisie car un motif de taille 3×3 ne permet pas de coder entièrement l'image projetée par le projecteur.

La construction de la mire se fait de la manière suivante :

1. Une cellule noire est choisie pour le coin gauche supérieur de l'image (un carré noir).
2. Une liste aléatoire de toutes les combinaisons possibles d'une colonne de 5 unités est créée. On ajoute la première combinaison possible et on vérifie que tout carré 5 par 5 est unique. Si l'unicité n'est pas respectée, on vérifie avec la deuxième combinaison... Lorsque l'unicité est respectée, on recommence toute l'opération : une nouvelle liste de combinaison, ajout de colonne et vérification jusqu'à atteindre la limite de l'image.
3. La seconde étape consiste à ajouter une ligne d'unité à l'image : une liste de combinaison d'une ligne de 5 unités est créée : on vérifie l'unicité. La méthode est la même que décrite précédemment.
4. On complète la ligne en ajoutant unité par unité, la liste se limitant à 2 choix (blanc ou noir)
5. Une fois la ligne complétée, on commence une nouvelle ligne jusqu'à remplir intégralement l'image.

La figure 4.2 représente la méthode décrite précédemment.

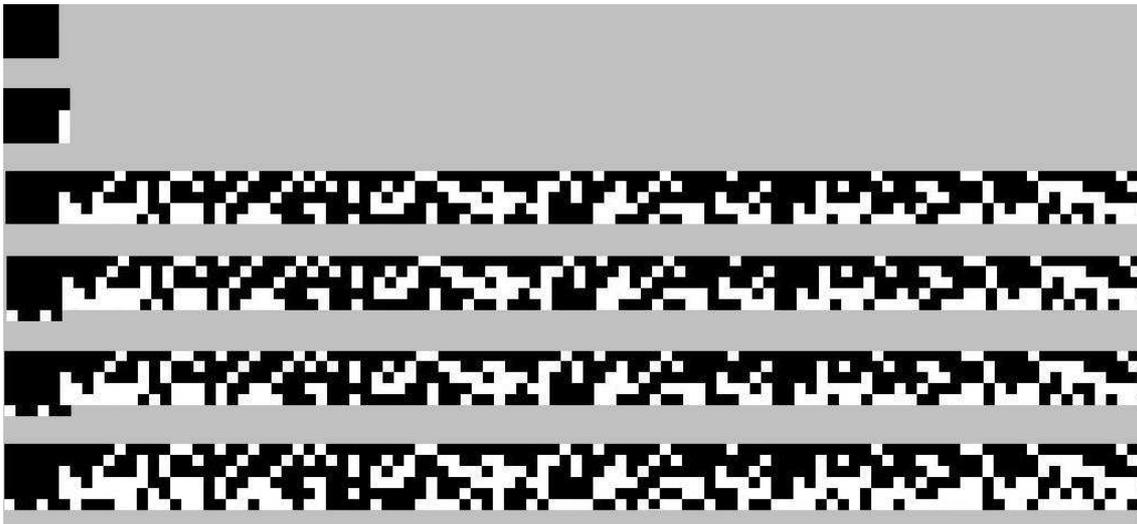


Figure 4.2 – Création de la mire binaire, étape par étape

La figure 4.3 montre une mire possible et précise que l'unicité des cellules est respectée, même dans le cas de chevauchement de 2 carrés.



Chaque cellule de 5 par 5 éléments est unique

Figure 4.3 – Mire binaire pseudo-aléatoire

Identifier des points de correspondance consiste à reconnaître la même cellule sur deux images. Nous allons utiliser le voisinage autour du point d'intérêt sous la forme de cellule ou matrice 5×5 pour identifier cette cellule.

4.1.1.2 Points de correspondance

Pour reconnaître deux cellules identiques, nous allons employer un critère de corrélation sur le voisinage de deux images, couplé à la géométrie épipolaire. A partir d'un point sur une première image, le point de correspondance sur la deuxième image se trouve le long de la ligne épipolaire. La mire structurée permet de lever l'ambiguïté sur la ligne épipolaire et de trouver ce point de correspondance. La géométrie épipolaire ne sert qu'à limiter la zone de recherche dans la deuxième image.

Les points de correspondance dans les trois images requièrent que les matrices fondamentales soient estimées. La figure 4.4 expose le processus complet pour trouver ces points.

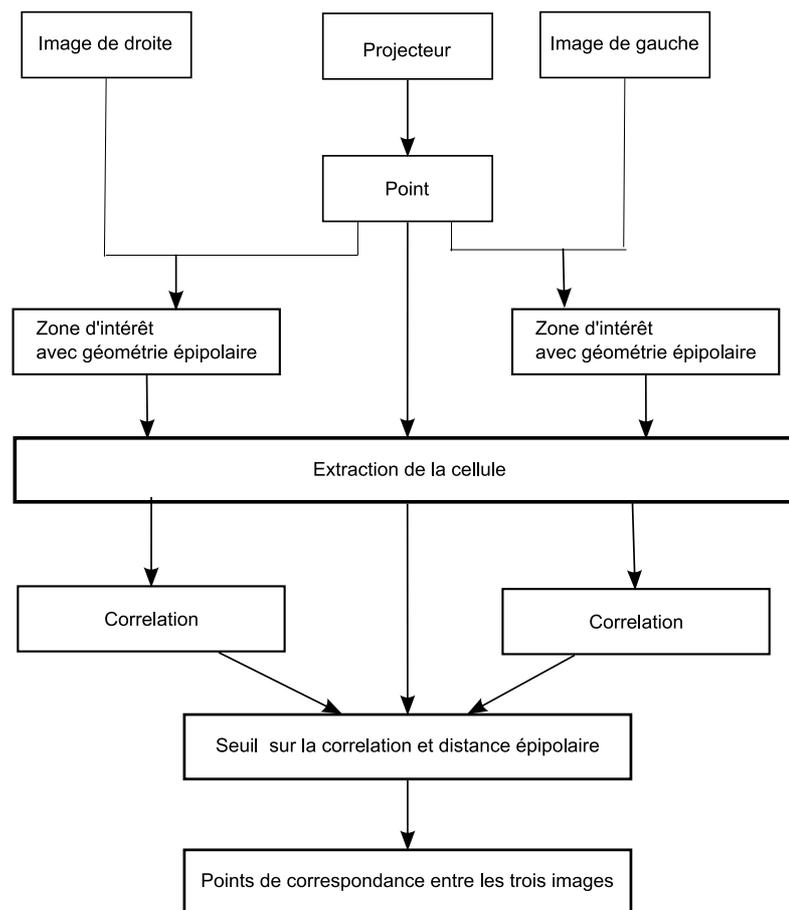


Figure 4.4 – Méthode pour extraire les points de correspondance de la mire binaire

La ligne épipolaire permet de limiter la zone de recherche de la cellule. Nous allons estimer la corrélation entre le voisinage d'un point de la mire sur l'image du projecteur [Garcia Campos, 2001] et celui d'un point le long de la ligne épipolaire. Notre problème revient à comparer deux cellules, notées A et B , des deux images. On estime la corrélation (équation 4.2) entre les cellules 5×5 des deux images.

$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{(\sum_m \sum_n (A_{mn} - \bar{A})^2)(\sum_m \sum_n (B_{mn} - \bar{B})^2)}} \quad (4.2)$$

avec \bar{A} et \bar{B} les moyennes des cellules A et B . La cellule ou voisinage est définie comme une matrice 5×5 , dont l'élément central est le pixel testé et chaque élément de la matrice un pixel voisin. On estime que le point de corrélation maximum coïncide avec le point de correspondance. La figure 4.5 montre sur l'image de gauche la mire, le point sélectionné et son voisinage, et sur l'image de droite la ligne épipolaire correspondante ainsi que le point de correspondance estimé.

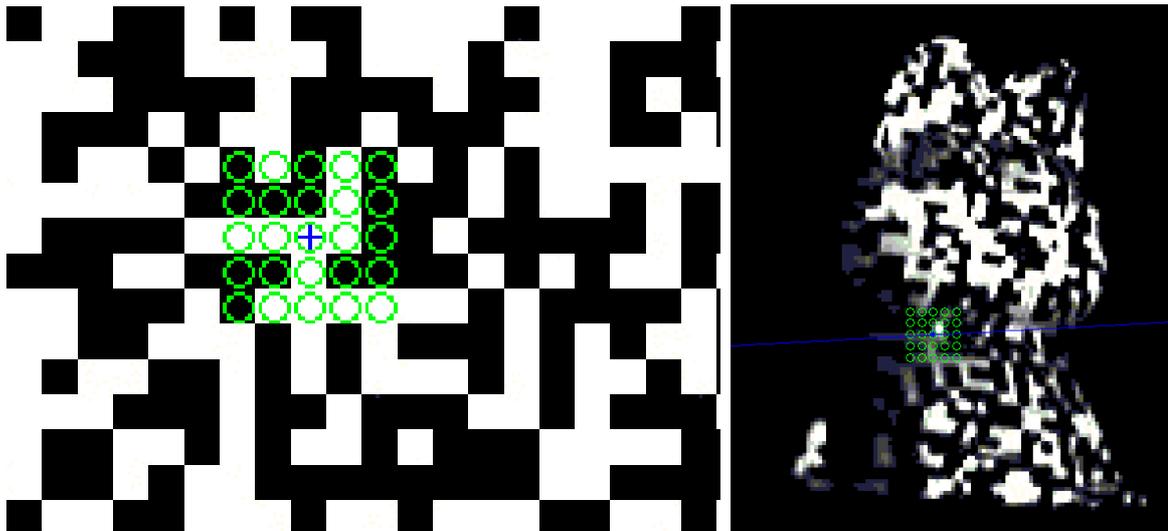


Figure 4.5 – A gauche Point de correspondance centrée sur une cellule, à droite cellule centrée sur la ligne épipolaire

4.1.1.3 Les points aberrants

Les points aberrants sont des points dont la mise en correspondance et la reconstruction 3D sont mauvaises. Nous fixons des seuils pour éliminer tous les points aberrants :

- Le premier paramètre est la corrélation. Nous avons écrit que le maximum de corrélation correspond généralement au point de correspondance, mais aucun seuil n'a encore été fixé pour ce maximum. Une valeur est choisie empiriquement pour éliminer les points aberrants. La valeur de corrélation étant comprise entre 0 et 1, une valeur de 0.8 satisfait nos contraintes.
- Le second paramètre concerne la géométrie épipolaire. La ligne épipolaire permet de limiter la zone de recherche dans la seconde image et la distance entre la ligne épipolaire et le point correspondant peut être estimé. Grâce aux estimations d'erreur de la géométrie épipolaire, nous pouvons imposer un seuil épipolaire d'un pixel. La figure 4.6 montre les points de correspondance trouvés sur les images des caméras de droite et de gauche.

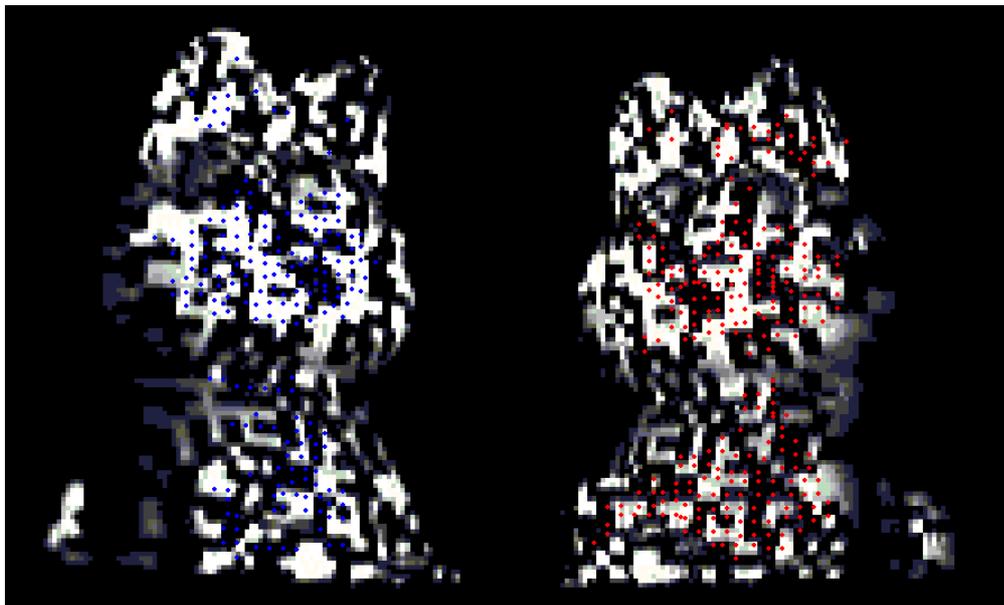


Figure 4.6 – Points de correspondance entre les caméras

4.1.2 Mire Couleur

Une deuxième technique de codage de mire est employée pour accroître la précision de la reconstruction 3D du modèle et permettre ainsi un recalage de vues. Cette mire a été développée par Pages [Pagès et al., 2005]. Nous présentons succinctement le principe de la mire et son décodage. Cette mire (figure 4.8) ne s'appuie pas sur un motif unique mais sur une séquence de bande de couleurs. Son codage repose sur le principe que tout triplet de bandes de couleur constitue une unique combinaison de trois couleurs successives. Ainsi on ne cherche plus à identifier un carré mais trois bandes de couleurs.

4.1.2.1 Séquence de De Bruijn

La stratégie pour coder la mire couleur repose sur les séquences de De Bruijn : les voisins sont définis par des séquences pseudo-aléatoires. Un bref rappel mathématique explique la nature de cette séquence.

Une séquence de De Bruijn d'ordre m sur un alphabet de n symboles est un chemin circulaire de longueur $n \times m$ qui contient chaque élément de longueur m exactement une seule fois. Ce type de séquence peut être obtenue en parcourant des circuits hamiltoniens sur des graphes de De Bruijn [Fredricksen, 1982]. Par exemple dans la figure 4.7 tous les mots de longueur $m - 1$ sont inclus dans les sommets. Un circuit eulérien est un chemin qui part et revient du même sommet et ne passe qu'une seule fois par les arêtes. En regroupant les labels des arêtes d'un tel circuit, on obtient une séquence de De Bruijn d'ordre m

1000010111101001

Un chemin hamiltonien est un chemin qui part et revient du même sommet en passant par tous les autres sommets une seule fois, une séquence de De Bruijn d'ordre $m-1$ est obtenue

00101110

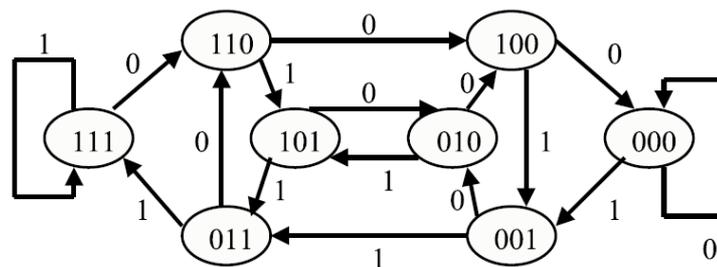


Figure 4.7 – Chemin Hamiltonien et Eulérien sur un graphe de Bruijn

Des séquences pseudo-aléatoires ont été utilisées pour coder des mires, code basé sur des colonnes, des lignes ou des grilles.

4.1.2.2 Stratégie de codage

La stratégie de codage s'appuie sur la volonté de combinée les avantages des méthodes *multi-split* et *stripe pattern* :

- dans le canal intensité, l'image est codée suivant une *peak based pattern*. La méthode de codage *multi-split* introduit des bandes noires entre les bandes de couleur. Deux bandes consécutives peuvent avoir la même couleur. Les bandes noires permettent l'emploi de détecteurs de pic d'intensité. Le problème de correspondance est résolu pour les points se situant au centre de chaque bande de couleur, correspondant à un pic d'intensité.
- dans l'espace RGB le codage s'appuie sur le *stripe pattern*. Cette méthode n'introduit pas de bandes sombres entre les bandes colorées. Deux bandes adjacentes ne peuvent posséder la même couleur. Les frontières de ces bandes sont estimées pour déterminer les correspondances.

La stratégie développée par Pages ne requiert que $n = 4$ différentes valeurs de teinte pour coder une mire de 128 lignes avec une propriété de fenêtre $m = 3$. Notons W_m et W_h respectivement la largeur en pixels d'une bande d'intensité moyenne et forte, $S = 1, \dots, 2^m$ le jeu d'indices des bandes. Alors le passage de l'abscisse de la mire vers le numéro d'indice de bande est défini par

$$\begin{aligned} & \text{stripe} : X \rightarrow S \\ \text{stripe}(x) &= 2((x - 1) \div (W_m + W_h)) + \begin{cases} 1 & ((x - 1) \bmod (W_m + W_h)) + 1 < W_m \\ 2 & \text{sinon} \end{cases} \end{aligned} \quad (4.3)$$

avec \div symbole de division entière. L'équation suivante détermine le niveau d'intensité L d'une bande, qui a deux valeurs possibles, haute ou moyenne.

$$\begin{aligned} & \text{Int} : S \rightarrow L \\ \text{Int}(x) &= \begin{cases} \text{haute} & x \bmod 2 = 0 \\ \text{moyenne} & \text{sinon} \end{cases} \end{aligned} \quad (4.4)$$

La mire est divisée en n périodes $P = 1, \dots, n$ de n^m bandes chacune.

Pour déterminer dans quelle période une bande appartient la fonction suivante est définie :

$$\begin{aligned} & \text{period} : S \rightarrow P \\ \text{period}(s) &= ((s - 1) \div 2n^{m-1}) + 1 \end{aligned} \quad (4.5)$$

Soit $H = 1, \dots, n$ le jeu d'indices de la valeur de teinte choisi. Alors la valeur de teinte d'une bande donnée est définie par

$$Couleur : S \rightarrow H$$

$$Couleur(s) = \begin{cases} period(s)Int(s) = moyenne \\ db [(s - 1) \bmod 2n^2 + 1/2] \text{ sinon} \end{cases} \quad (4.6)$$

avec db un vecteur ordonné qui contient la séquence de De Bruijn avec une propriété de fenêtre de 2, comme celle suivi par [Salvi et al., 2004]

$$db = [1, 1, 2, 1, 3, 1, 4, 2, 2, 3, 2, 4, 3, 3, 4, 4] \quad (4.7)$$

Chaque bande avec une certaine teinte et un certain niveau d'intensité peut être identifiée via un jeu de labels $B = 1, 2, 3, 4, 5, 6, 7, 8$ par la fonction

$$Label : H \times L \rightarrow BLabel(h, l) = \begin{cases} hl = haute \\ h + nl = moyen \end{cases} \quad (4.8)$$

La séquence de bandes générée par la stratégie de codage proposée et la séquence de De Bruijn se représentent par l'équation :

$$\begin{aligned} P_1 &= [5, 1, 5, 1, 5, 2, 5, 1, 5, 4, 5, 2, 5, 2, 5, 3, 5, 2, 5, 4, 5, 3, 5, 3, 5, 4, 5, 4] \\ P_2 &= [6, 1, 6, 1, 6, 2, 6, 1, 6, 4, 6, 2, 6, 2, 6, 3, 6, 2, 6, 4, 6, 3, 6, 3, 6, 4, 6, 4] \\ P_3 &= [7, 1, 7, 1, 7, 2, 7, 1, 7, 4, 7, 2, 7, 2, 7, 3, 7, 2, 7, 4, 7, 3, 7, 3, 7, 4, 7, 4] \\ P_4 &= [8, 1, 8, 1, 8, 2, 8, 1, 8, 4, 8, 2, 8, 2, 8, 3, 8, 2, 8, 4, 8, 3, 8, 3, 8, 4, 8, 4] \end{aligned} \quad (4.9)$$

avec P_i la séquence de bandes correspondant à la période i . On rappelle que les éléments 1, 2, 3, 4 correspondent aux bandes d'intensité fortes et 1 indique la première teinte ... 5, 6, 7, 8 correspondent aux intensités moyennes.

La figure 4.8 montre la mire codée suivant la stratégie énoncée, avec $n = 4$ teintes qui sont réparties de manière homogène dans l'espace de teinte de telle sorte que $H = 0^\circ, 90^\circ, 180^\circ, 270^\circ$

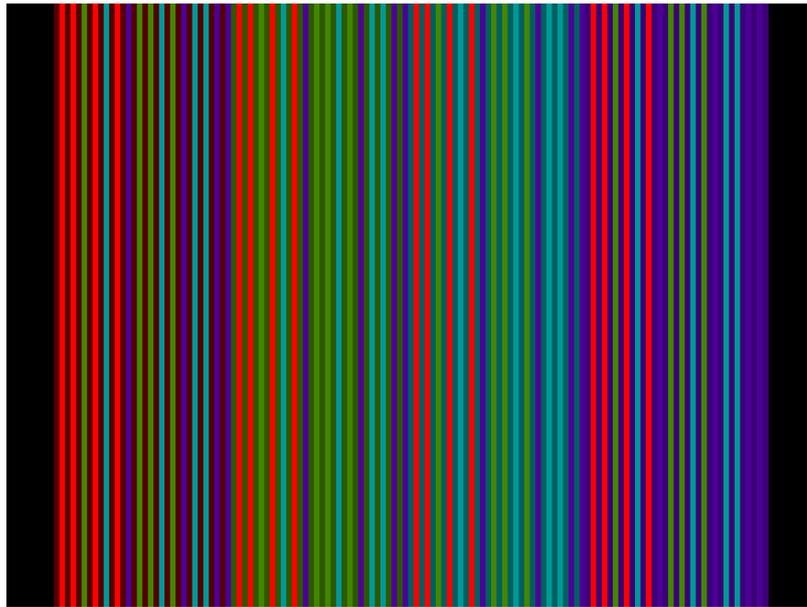


Figure 4.8 – Mire couleur de Pages

4.1.2.3 Décodage et Points de correspondance

Les bandes sont identifiées suivant la ligne de l'image (figure 4.9). Chaque ligne est analysée.

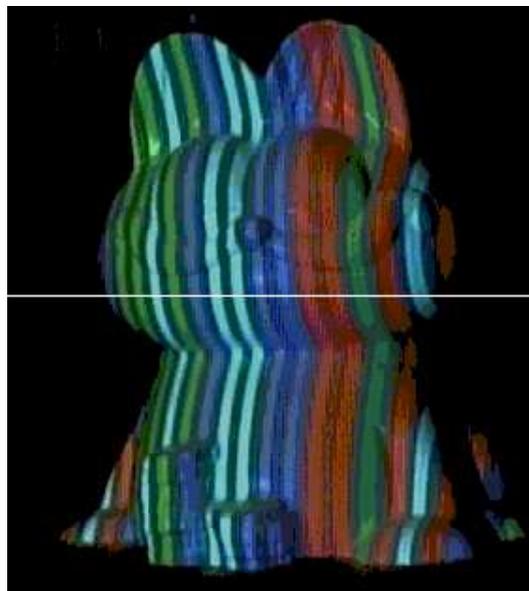


Figure 4.9 – Mire couleur de Pages projetée sur un modèle réel : une ligne est extraite de cette image

Les figures 4.10 et 4.11 permettent d'analyser les pics de couleurs RGB.

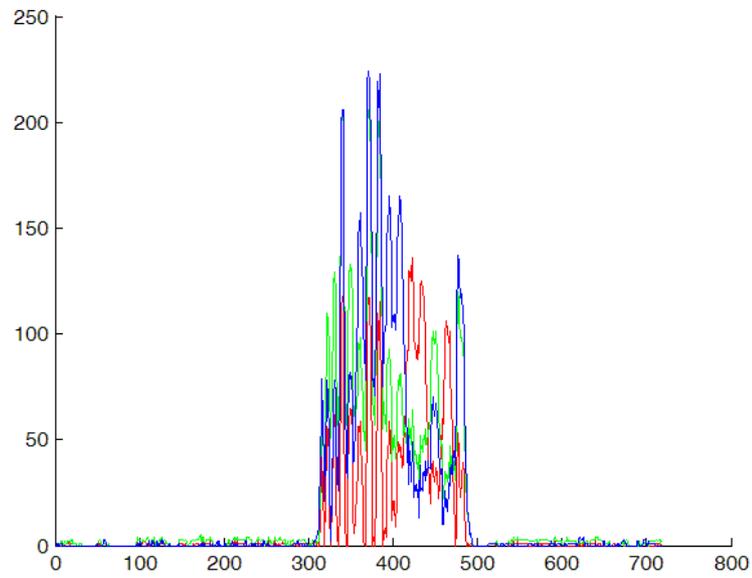


Figure 4.10 – Courbes RGB de la ligne analysée pour extraire les pics

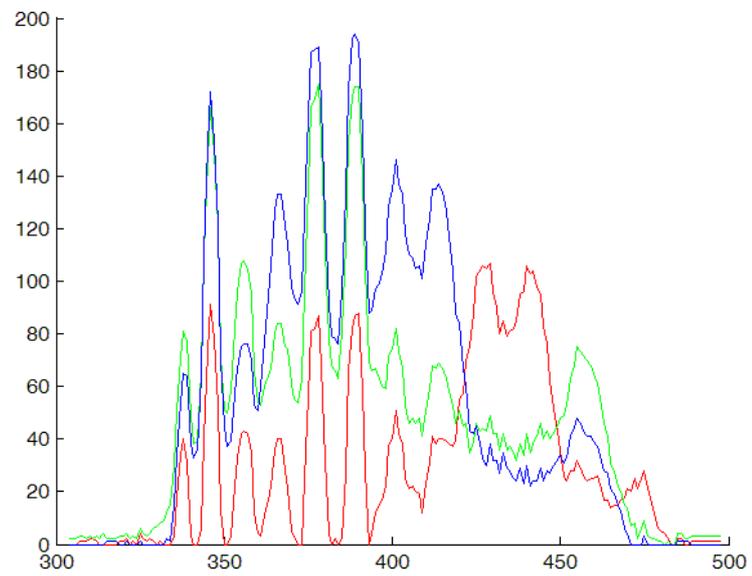


Figure 4.11 – Courbes RGB de la ligne analysée pour extraire les pics sur une plage de valeurs affinée

Blais et Rioux ont introduit un filtre numérique linéaire dérivatif [F. Blais, 1986] :

$$g(i) = \sum_{c=1}^{o/2} (f(i+c) - f(i-c)) \quad (4.10)$$

avec f la fonction original , ici le M canal de la ligne image, g est le filtre dérivé linéaire d'ordre o et i indique l'index qui est filtré. Cet opérateur est appliqué deux fois pour obtenir la dérivée seconde du canal. Cette segmentation permet de différencier les zones de forte et moyenne intensité. Le centre de ces zones est estimé avec une précision sub-pixel en utilisant un détecteur de sommet [Trucco et al., 1998].

Les bords des bandes sont localisés à l'aide du sommet détecté : la position sub-pixel des frontières environnantes correspondent aux deux maxima les plus proches de la fonction $g = dR^2 + dG^2 + dB^2 + dM^2$ avec dR, dG, dB et dM les dérivés premières du filtre et M le canal de la ligne. Les maxima sont détectés en utilisant le même filtre, les maxima correspondants à une valeur nulle de la dérivé.

La valeur de la teinte de chaque bande est estimée comme la teinte moyenne de tous les pixels inclus dans la bande estimée. Quatre teintes sont projetées, la teinte de chaque région peut être identifiée par des règles simples basées sur les couleurs RGB :

$$\left\{ \begin{array}{ll} \max(R, G, B) = R \text{ et } G < 0.5.R \text{ et } B < 0.5.R & \Rightarrow H = 0^\circ \\ \max(R, G, B) = G \text{ et } B < 0.5.G & \Rightarrow H = 90^\circ \\ \max(R, G, B) = R & \Rightarrow H = 180^\circ \\ \max(R, G, B) = G & \Rightarrow H = 270^\circ \end{array} \right. \quad (4.11)$$

Nous n'utilisons que deux images (caméra et projecteur) pour déterminer les points de correspondance entre ces deux images. La géométrie épipolaire pourrait nous permettre d'améliorer la détection des points aberrants en estimant le point de correspondance sur la troisième image. Il suffit ensuite d'utiliser le code de la lumière structurée pour vérifier la justesse de la mise en correspondance. On peut aussi synthétiser l'image de la caméra non utilisée et comparer la synthèse de vue avec l'image de la caméra pour constater toute erreur visible.

Les figure 4.12 et 4.13 montre l'identification des points d'intérêt sur l'image puis la reconstruction 3D suite à l'identification de ces points par le codage de la séquence par de Bruijn.

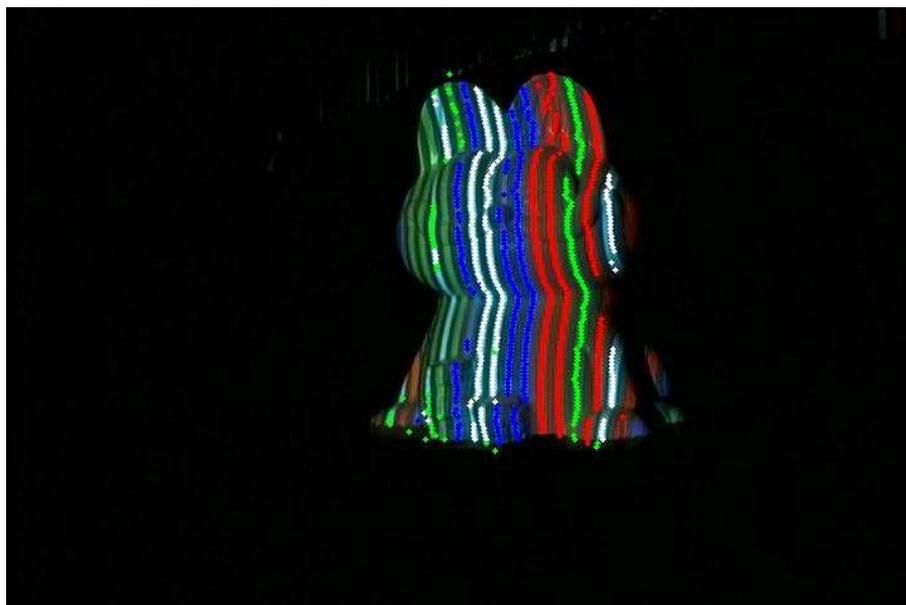


Figure 4.12 – Identification des points d'intérêt sur un objet texturé par la mire de Pages

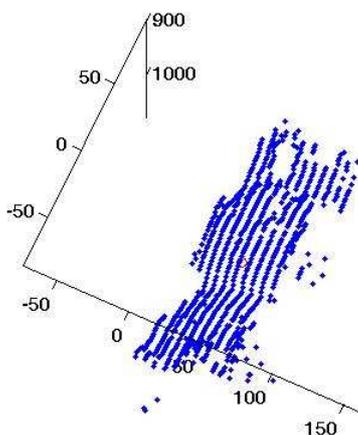


Figure 4.13 – Reconstruction 3D des points de correspondance de la mire de Pages

4.2 Synthèse de vue

Les deux techniques de lumière structurée que nous avons présentées dans le chapitre 4 font toutes les deux parties des techniques à une seule prise de vue. Cette limitation est imposée par la perspective de pouvoir utiliser la méthode dite de lumière structurée imperceptible (annexe A). Cette technique consiste à projeter la mire structurée de telle sorte qu'elle soit invisible à l'oeil nu [Fofi et al., 2004]. Cette limitation nous empêche d'utiliser des méthodes plus performantes pour la reconstruction 3D telles que la méthode multi-phases [Potsdamer and Altschuler, 1982] qui avec plusieurs images multiplie le nombre de points de correspondance et ainsi les points 3D reconstruits.

Cette restriction influe sur le nombre de points 3D estimés et sur la qualité de la reconstruction 3D. Le nombre de points étant estimés uniformément dans l'espace sans tenir compte de la complexité plus ou moins grande de la surface de l'objet, les détails de la surface, s'ils sont trop fins, ne sont pas détectés avec les méthodes employées.

Une reconstruction 3D partielle d'un objet a été estimée, avec plus ou moins de points et de précision suivant la méthode utilisée. La limite que nous nous sommes imposées, celle d'utiliser une technique à une prise de vue, entraîne cette reconstruction incomplète. Une projection de texture sur l'objet est possible mais le résultat reste incomplet et ne nous satisfait pas.

Néanmoins notre système permet de créer un modèle de référence à partir de plusieurs vues qui seront recalées. La figure 4.14 montre la reconstruction et le recalage de quatre vues pour le modèle du chat.

On constate l'absence de reconstruction valide des oreilles car l'objet est trop fin. Le nez de ce modèle, trop petit, échappe au système. Une fois ce modèle reconstitué avec plus de vues (annexe B), la texturation dense pourra être complétée par le recalage de notre reconstruction partielle avec un nuage de points plus dense pour obtenir une texturation dense.

Une fois l'étape de recalage effectuée, avec notre système précédemment étalonné, il ne reste plus qu'à projeter ce nuage sur le plan image du projecteur. Une fois la position des points 2D estimée sur le plan image, on peut utiliser les informations de texture pour la génération de texture, qui représente la dernière étape, indépendante qu'on peut renouveler.

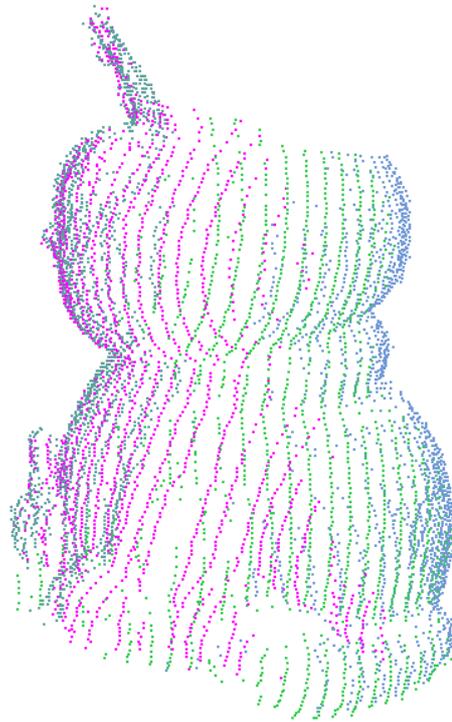


Figure 4.14 – Recalage

4.2.1 Reconstruction complète du modèle de référence

Nous avons montré dans le chapitre précédent que nous savons reconstruire la surface visible d'un objet par notre système. Nous n'obtenons pas une reconstruction homogène de la surface mais une série de lignes 3D reconstruites. Cette discontinuité peut engendrer des problèmes de convergence dans l'étape de recalage. Pour éviter ce problème et obtenir une meilleure visualisation de la surface (voire une meilleure reconstruction 3D de la surface), notre nuage de points va être interpolé pour combler les espaces vides entre nos lignes. L'interpolation d'un nuage de points désorganisé est compliquée, mais nos points peuvent être organisés suivant les franges projetées par la mire. En effet les lignes de la mire sont perpendiculaires aux franges, donc nous réalisons une interpolation suivant une ligne perpendiculaire aux franges et une unique interpolation est estimée pour une courbe 3D.

La figure 4.15 expose les résultats de la reconstruction 3D d'une surface avant et après l'étape d'interpolation.

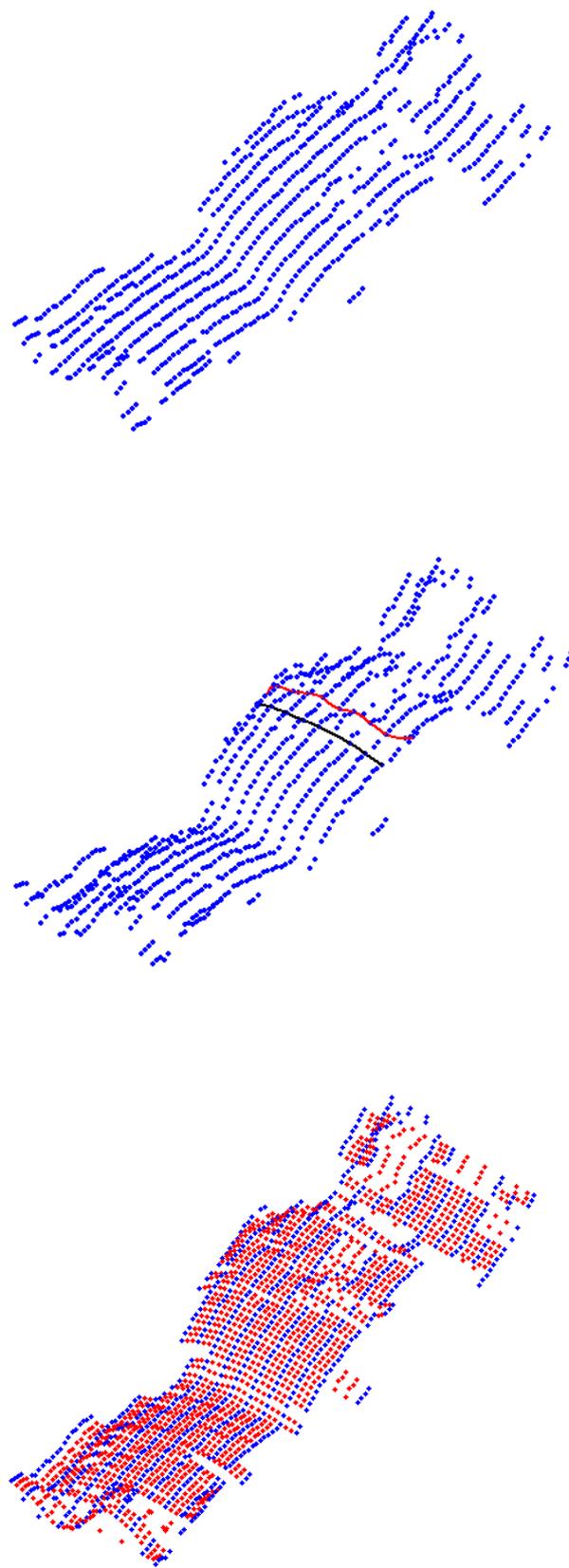


Figure 4.15 – a. Reconstruction 3D de la surface b. 2 splines sont estimées c. Reconstruction et interpolation de la surface

Cinq points sur une même ligne sont nécessaires pour pouvoir interpoler une spline courbe correctement. Si ce seuil n'est pas atteint, l'interpolation n'est pas réalisée. L'équation 4.12 décrit l'équation générale de la spline cubique :

$$\begin{aligned}x(t) &= a_x t^3 + b_x t^2 + c_x t + d_x \\y(t) &= a_y t^3 + b_y t^2 + c_y t + d_y \\z(t) &= a_z t^3 + b_z t^2 + c_z t + d_z\end{aligned}\tag{4.12}$$

avec $x(t)$, $y(t)$, $z(t)$ les coordonnées paramétrées de points 3D consécutifs, $a_x \dots d_z$ les coefficients de la spline, t paramètre de la fonction, qui varie de 0, position initiale, à 1, position finale. Pour résoudre ce système d'équations, outre les conditions aux valeurs extrêmes de t , il faut rajouter les conditions de continuité aux bornes ainsi que celle de continuité de la première dérivé.

4.2.2 Projection de texture sur un objet blanc

4.2.2.1 Recalage

La position de l'objet dans le repère du projecteur est estimée à travers la reconstruction 3D réalisée par la technique de lumière structurée. Pour projeter la texture générée sur l'objet, la position de cet objet doit être estimée (distance et orientation par rapport au projecteur). Nous avons choisi d'utiliser un modèle complet de l'objet (modèle obtenu par un recalage de plusieurs vues), bien qu'on puisse utiliser la technique de lumière structurée par phase pour obtenir un modèle plus dense (cette technique n'est pas à une prise de vue). La texture de cet objet, si on veut le reproduire, peut également être acquise lors de cette phase d'acquisition 3D [Lathuilière et al., 2006].

La technique de recalage se décompose en deux étapes :

- Déterminer la position initiale du nuage de points complet par rapport à notre reconstruction 3D.
- Recaler finement les deux nuages de points.

4.2.2.1.1 Position initiale Connaître la position initiale influe sur le type de méthode de recalage que l'on peut employer. On distingue deux grandes méthodes [Salvi et al., 2007]. Les techniques de recalage (figure 4.16) peuvent se classer en deux catégories les méthodes grossières et les méthodes fines :

- les techniques dites grossières ne nécessitent pas la connaissance de la position initiale.
- les techniques dites fines minimisent l'erreur de recalage à partir d'une connaissance a priori de la position initiale.

Les algorithmes de recalage ne donnent des résultats satisfaisants que si la position initiale et la position finale, celle que l'on recherche, sont proches l'une de l'autre. Si une erreur de translation entre les deux positions est corrigée par translation des centres de gravité, une erreur de rotation risque d'être plus difficilement corrigée si elle est trop importante. L'estimation initiale doit être aussi précise que possible. Cette position initiale du nuage de points du modèle scanné est référencée suivant le repère lié au projecteur, donc la position initiale peut être estimée car nous plaçons l'objet devant le projecteur.

Des tests préliminaires de recalage manuelle en indiquant des points de recalage entre le modèle scanné et notre reconstruction 3D ont montré la possibilité de projeter de manière adéquate une texture virtuelle sur l'objet réel. La prochaine étape est d'éliminer l'intervention humaine pour rendre le processus automatique.

4.2.2.1.2 Recalage d'une vue sur le modèle reconstruit La technique de recalage consiste à estimer le mouvement (une translation et une rotation) entre les deux nuages de points pour les faire coïncider. Le premier nuage de points est supposé parfait, de bonne précision, sans aucun point aberrant. Le deuxième nuage de points doit être recalé sur le premier : ce second nuage de points est le nuage de points reconstruit par la technique de lumière structurée : il est partiel, possède moins de points, ne concerne qu'une seule vue, et peut contenir des points aberrants.

Analyse en Composante principale La méthode ACP, ou Analyse en Composantes Principales, est une méthode de recalage grossier qui utilise la direction de l'axe principal du volume créé par un nuage de point pour les recalcr. Si la zone commune est suffisante (50% est le seuil critique pour obtenir un résultat), les deux axes doivent presque coïncider et le recalage doit être bon. Néanmoins cette méthode simple, rapide par rapport à d'autres méthodes qui demandent l'estimation de courbure ou signature [Chua and Jarvis, 1997], peut servir d'initialisation à une méthode fine.

Chung et Lee proposent une méthode [Chung et al., 1998] qui utilise les vecteurs du nuage de points : ils estiment la matrice de covariance pour chaque nuage :

Registration strategy	Linear							
	Iterative							
Robustness								
Motion Estimation	Not defined/Other							
	Eigenvectors							
	Least Squares							
Kind of Correspondence	Surface							
	Curves							
	Points							

Coarse Registration	Point Signature	Chua97	√		√		√	√
		Ho99	√			√	√	√
		Chua00	√			√	√	√
	Spin Image	Johnson99	√		√		√	√
		Carmichael99	√		√		√	√
		Huber99	√		√		√	√
	PCA	Huber02	√		√		√	√
		Chung		√		√		√
		kim02		√		√		√
	RANSAC-Based Darces	kim03		√		√		√
		Chen98	√		√		√	√
	Hung99	Hung99	√		√		√	√
		Tarel98		√		√		√
	Line-based	Wyngaerd02		√		√	√	√
		Stamos03		√		√	√	√
Chen05			√		√	√	√	
Genetic Alg	Brunnstrom96	√		√		√	√	
Principal Curvature	Feldmar94	√			√		√	

Fine Registration	ICP	Besl92	√		√		
		Kapoutsis98	√		√	√	
		Yamany98	√		√	√	
		Trucco99	√		√	√	√
		Greenspan01	√		√	√	
		Jost02	√		√	√	
		Sharp02	√		√	√	
		Zinsser03	√		√	√	√
	Chen	Chen91	√			√	
		Gagnon94		√		√	
		Pulli99		√		√	
	Signed Distance Fields	Rusinkiewicz01	√			√	
		Masuda01		√		√	√
	Genetic Algorithms	Masuda02		√		√	√
		Chow03	√		√	√	√
Silva04	√		√	√	√		

Registration strategy	Pair-wise registration						
	Multi-view registration						
Efficient search	k-d trees						
Minimization distance	distance point-point						
	distance point-plane						
Robustness							

Figure 4.16 – Classification des techniques de recalage fin et grossier proposé par [Salvi et al., 2007]

$$Cov = \frac{1}{N} \sum_{i=0}^{N-1} (p_i - \bar{p})(p_i - \bar{p})^t \quad (4.13)$$

avec N nombre de point, \bar{p} le centre de masse du nuage de point et p_i le p_{ime} point.

La direction U_i de l'axe principal est déduit d'une décomposition en valeur singulière :

$$Cov_i = U_i D_i U_i^t \quad (4.14)$$

La matrice de rotation est alors extraite par le produit des matrices des vecteurs absolus :

$$R = U_1 U_2^{-1} \quad (4.15)$$

le vecteur de translation est estimé par la distance entre les centres de masse, en respect avec le même axe :

$$t = \bar{\mu}_2 - R \bar{\mu}_1 \quad (4.16)$$

ICP Iterative Closest Point Besl et McKay ont introduit la méthode ICP [Besl and McKay, 1992], qui consiste à minimiser la distance entre les points de pseudo correspondance, soit le point le plus proche. L'estimation initiale sert à déplacer le nuage de points vers sa nouvelle position. Ensuite on estime la distance entre un point m_i du premier nuage de points vers son voisin le plus proche p_i du second nuage de points, afin de minimiser la distance entre tous ces points. On répète la méthode tant que la distance entre les points de pseudo-correspondance est supérieur à un seuil.

$$f = \frac{1}{N_p} \sum_{i=1}^{N_p} \|\vec{m}_i - R(\vec{q}_R) \vec{p}_i - \vec{t}\| \quad (4.17)$$

Pour minimiser l'équation 4.17, une matrice symétrique $Q(\sum_{pm})$ est introduite

$$Q(\sum_{pm}) = \begin{bmatrix} tr(\sum_{pm}) & \Delta^t \\ \Delta & \sum_{pm} + \sum_{pm}^t - tr(\sum_{pm}) I_3 \end{bmatrix} \quad (4.18)$$

avec tr la trace, $\Delta = [A_{23} \ A_{31} \ A_{12}]^t$ extraite de la matrice antisymétrique $A_{ij} = (\sum_{pm} - \sum_{pm}^t)$, I_3 la matrice identité $x \times 3$ et \sum_{pm} la matrice de covariance des points m et p

$$\sum_{pm} = \frac{1}{N_p} \sum_{i=1}^{N_p} [\vec{p}_i \vec{m}_i] - \vec{\mu}_p \vec{\mu}_m \quad (4.19)$$

Le vecteur absolu unitaire $\vec{q}_R = [q_0 \ q_1 \ q_2 \ q_3]^t$ de la valeur absolue maximale de la matrice Q est sélectionné comme la meilleure rotation exprimée en quaternion. L'équation 4.20 donne alors la matrice de rotation

$$R = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 - q_0q_1) & q_0^2 - q_1^2 - q_2^2 - q_3^2 \end{bmatrix} \quad (4.20)$$

Le vecteur de translation a la même forme que l'équation 4.16. La figure 4.17 montre le recalage de deux vues sur un modèle de référence.

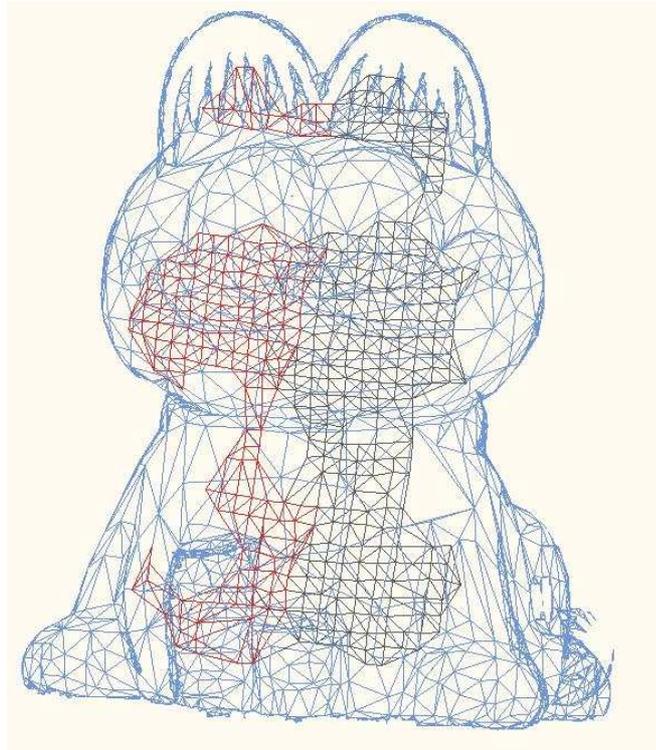


Figure 4.17 – Recalage de notre reconstruction 3D incomplète sur le modèle de référence

4.2.2.2 Synthèse de vue

Une fois le nuage de points recalé dans le repère du projecteur, la synthèse de vue peut être estimée. La synthèse de vue consiste à synthétiser l'image que verrait une caméra placée à l'endroit exact du projecteur et possédant les mêmes paramètres extrinsèques et intrinsèques.

La première étape consiste à estimer la projection des points 3D sur le plan image du projecteur grâce à la formule :

$$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} R & T \end{bmatrix} \begin{bmatrix} x_d \\ 1 \end{bmatrix} \quad (4.21)$$

$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix}$ est le point 2D, \mathbf{K} est la matrice des paramètres intrinsèques, R et T sont les vecteurs rotation et translation, $\begin{bmatrix} x_d \\ 1 \end{bmatrix}$ est le point 3D. La figure 4.18 montre la synthèse du nuage de points sur le plan image du projecteur.

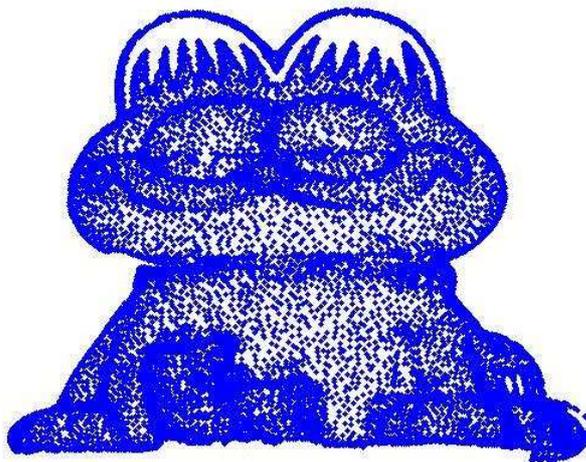


Figure 4.18 – Synthèse de vue sans texture

4.2.2.3 Texturation

Une fois la position des points 2D estimée, il suffit d'estimer la couleur de chaque face en faisant la moyenne des couleurs RGB des trois points qui composent les sommets de chaque face. La figure 4.19 montre l'image synthétisée que le projecteur va afficher. On synthétise donc l'image d'une caméra positionné à la place du projecteur.



Figure 4.19 – Synthèse de vue

4.3 Synthèse

Nous avons abordé la problématique de la mise en correspondance à travers deux méthodes d'acquisition active : une mire binaire et une mire couleur. Estimer des points de correspondance consiste à identifier le codage inséré dans la mire projetée, que ce soit l'élément unique de la mire binaire ou une séquence de trois couleurs.

La détection du motif unique projeté par la mire constitue l'enjeu principal de ce chapitre, une détection précise et fiable du motif étant un gage d'un appariement de points de correspondance valide, et donc d'une reconstruction 3D précise.

Deux mires ont été testées :

- La première mire, binaire, est codée de telle sorte que toute cellule composée de cinq carrés par cinq est unique dans toute l'image, même en cas de chevauchement. L'identification de cet élément permet de déterminer les points de correspondance.
- La deuxième mire est composée d'une séquence de bandes de couleurs où chaque trio de bandes de couleurs est unique grâce à la séquence de De Bruijn. Estimer la position des bandes, et en particulier celle des pics, et puis identifier leur couleur permet d'identifier les points de correspondance.

Chacune de ces mires présentent des avantages et des désavantages l'une par rapport l'autre :

- si la mire binaire requiert une maîtrise de l'éclairage moins importante la rendant plus facile à mettre en application, la difficulté d'identifier précisément la position du centre de sa cellule rend sa précision faible, d'où une reconstruction 3D bruitée. On reconnaît la forme de l'objet, mais ses détails passent inaperçus et rendent un recalage incertain.
- La mire couleur se caractérise par sa vitesse d'exécution. On ne traite les lignes qu'une fois (alors que l'on doit re-parcourir chaque ligne épipolaire pour la mire binaire), la précision subpixel de la détection des pics par la méthode RB (ou d'autres [Forest et al., 2004]). Par contre la mise en place de l'étalonnage couleur et la maîtrise de l'éclairage demande beaucoup d'attention.

Se limitant à une technique de lumière structurée à une prise de vue (d'autres techniques auraient permis une reconstruction globale de l'objet, comme la projection de phases), la reconstruction 3D obtenue reste incomplète à la vue de notre objectif final, la projection de texture virtuelle sur un objet. Le chapitre suivant présente la dernière partie de notre méthode, le recalage de deux nuages de points et la synthèse de vue de l'image du point de vue du projecteur.

Chapitre 5

Résultats expérimentaux

Ce chapitre regroupe les résultats expérimentaux obtenus lors des différentes étapes de notre méthode. L'étalonnage, la géométrie épipolaire, le recalage sont estimés pour évaluer leur précision. La dernière étape, synthèse de vue et rétroprojection de l'image sur l'objet, n'offre que des résultats qualitatifs.

La procédure expérimentale est un système étalonné *fortement*, tous les paramètres (extrinsèques et intrinsèques) ainsi que les outils de géométrie épipolaire sont estimés suivant la méthode décrite dans le chapitre 3. Ce système est composé d'un projecteur DLC et deux caméras (webcam pour la mire binaire, caméras *CDD* pour la mire couleur).

La résolution du système est définie comme :

- projecteur : 1200×728
- webcam : 320×240
- caméra *CDD* : 768×580

5.1 Système étalonné

Une fois le système étalonné, plusieurs applications sont possibles à partir d'un jeu de points de correspondance entre deux images. On peut alors estimer :

- La reconstruction 3D de ces points.
- La synthèse de vue : on peut estimer la position du point de correspondance de l'image du projecteur, appelé par commodité troisième image.

Ces applications nécessitent toutes les deux des points de correspondance. Des jeux de points de correspondance ont été déterminés lors de l'étalonnage du système.

5.1.1 Reconstruction 3D

Pour estimer la précision de notre système au regard de la reconstruction 3D, nous allons reconstruire la mire projetée en utilisant soit le couple de caméras, soit un couple caméra-projecteur. La figure 5.1 présente cette reconstruction.

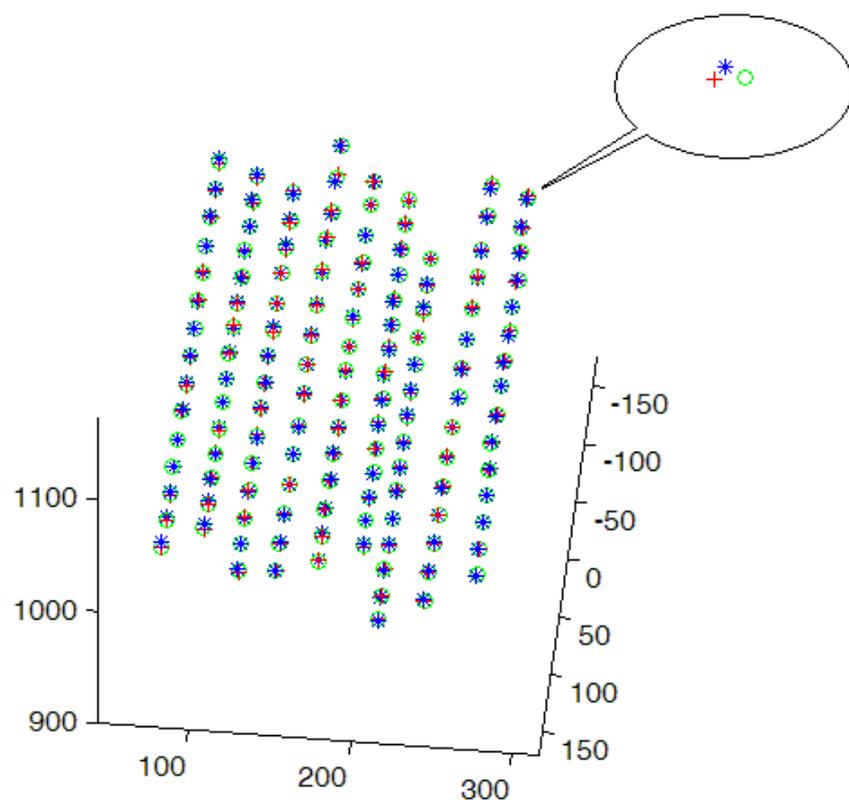


Figure 5.1 – Reconstruction 3D de la mire

Les erreurs de reconstruction entre les différents doublets sont exposées dans la table 5.1. Les couples *projo-caméra g*, *projo-caméra d* et *caméra d - caméra g* font référence au couple de projecteur et de caméra (une caméra avec un projecteur, ou les deux caméras) utilisé pour trianguler la position 3D des points.

couple	distance moyenne en mm
projo-camera d vs projo-camera g	1,76
projo-camera g vs camera g - camera d	1,01
projo-camera d vs camera g - camera d	1,01

TAB. 5.1 – Estimation de distance entre des points 3D employés lors de l'étalonnage

L'erreur la plus importante apparaît pour le couple projecteur-caméra car l'erreur d'étalonnage des caméras est propagée sur celle du projecteur, alors que l'erreur pour le couple caméra-projecteur contre les deux caméras est sensiblement la même.

5.1.2 Synthèse de vue

La synthèse de vue consiste à générer de nouvelles vues d'une scène à partir d'autres vues. Ces nouvelles vues créées correspondent à des points de vue virtuels où aucune caméra n'a pris d'image. Les premiers tests se sont limités à des objets simples, comme un cube. Les images générées sont simples, car nous ne nous intéressons qu'aux points d'intérêt, comme les centres de disque (mire du projecteur) ou les sommets de carrés (mire des caméras).

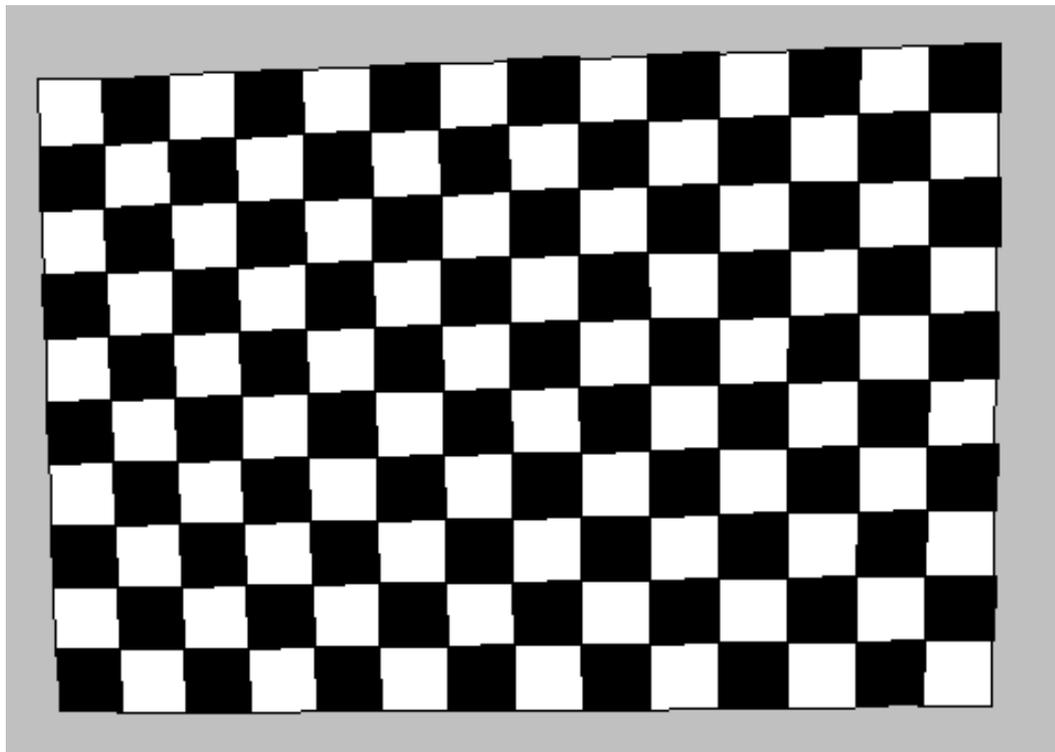


Figure 5.2 – Synthèse de vue de la mire employée pour l'étalonnage des caméras

La mire (figure 5.2) est placée orthofrontalement devant le projecteur : nous supposons que l'image "vue" par le projecteur doit être composée de carrés. En examinant l'image projetée, on remarque que la zone droite de l'image est plus distordue que celle de gauche. Nous supposons que cette déformation de l'image est due à la distorsion de la lentille que nous avons ignorée et dont nous constatons l'impact visuel. La déformation droite de l'image, plus forte, provient sans doute du fait que notre système est étalonné avec la caméra gauche comme origine. Sa matrice de projection P est supposée de la forme $P = \begin{bmatrix} I & | & 0 \end{bmatrix}$. On applique une homographie sur toutes les matrices de projection pour satisfaire l'écriture de P et le formalisme épipolaire (voir section 3.2)

5.1.2.1 Synthèse de la mire

Nous synthétisons la mire du projecteur. Elle est connue (car construite), donc une comparaison entre la mire et sa simulation est possible (figure 5.3).

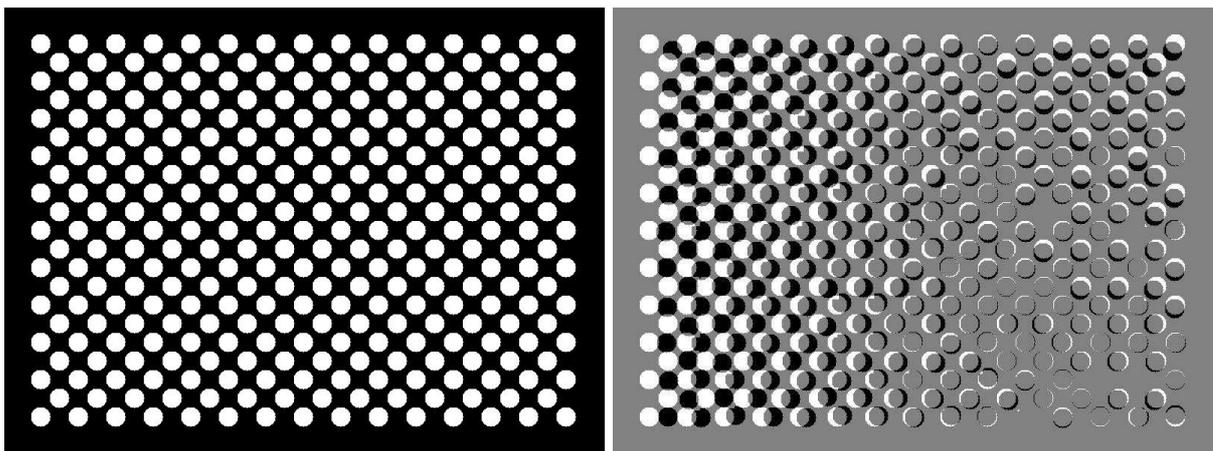


Figure 5.3 – Synthèse de vue de la mire employée pour étalonner le projecteur et comparaison entre la synthèse de vue et l'image originale

La figure 5.3 montre le même type d'obliquité que la figure 5.2. On remarque à nouveau la différence de distorsion entre la partie droite et gauche de l'image. Nous estimons l'erreur de rétroprojection de la mire dans la section suivante.

5.1.2.2 Erreur de rétroprojection

L'erreur de rétroprojection est la distance entre un point connu de la troisième image (dans notre cas l'image du projecteur) et le point de correspondance estimée à partir du tenseur trifocal et des points de correspondance de la première et deuxième image des caméras. Lors de l'étalonnage

du projecteur, nous estimons la position des centres des disques pour les trois images, donc nous évaluons la précision de la méthode de transfert. Le tableau 5.2 montre les résultats obtenus :

Mesure	dist mean	dist max	dist min	std deviation
distance en pixel	9	31	0.8	7.5

TAB. 5.2 – Erreur en pixel

Habed a proposé un état de l’art et une comparaison [Habed, 2000] entre les méthodes connues de synthèse de vue employant deux autres vues. Si l’erreur moyenne (tableau 5.2) est acceptable et du même ordre de grandeur obtenue par Habed, la déviation observée sur la partie droite de l’image ne l’est pas. L’erreur de rétroprojection passe de 3 pixels pour la zone droite à 40 pixels pour la zone gauche de l’image.

5.1.3 Amélioration

Nous avons testé différentes méthodes pour réduire cette erreur : changement d’origine, normalisation de Hartley, mais l’amélioration reste négligeable. Ce type d’erreur apparaît décentrée est radiale [Weng et al., 1992]. Nous supposons alors que le facteur de distorsion est le responsable de la déformation de l’image (voir section 3.1.1.2). Ce facteur est déterminé lors de l’étalonnage. On peut alors estimer la position corrigée des points de correspondance sur les images des caméras sans la distorsion.

La correction de la distorsion suivant le vecteur estimé k_c est précédée par la normalisation des points. L’équation 5.1 décrit la matrice de normalisation M

$$M = \begin{bmatrix} 1 & -\alpha_c & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{f_c(1)} & 0 & \frac{-cc(1)}{f_c(1)} \\ 0 & \frac{1}{f_c(2)} & \frac{-cc(2)}{f_c(2)} \\ 0 & 0 & 1 \end{bmatrix} \quad (5.1)$$

Il faut alors ajuster le tenseur trifocal [Hartley and Zisserman, 2004] aux points de correspondance normalisés.

$$T_i^{kj} = (F^{-1})_i^r \cdot G_s^j \cdot H_t^k \cdot T_r^{st} \quad (5.2)$$

avec F et G les homographies appliquées respectivement sur les images des caméras et $H = I$ l’homographie sur le projecteur (on considère la distorsion due au projecteur nulle).

les résultats apparaissent meilleurs suite à la correction de la distorsion (voir table 5.3).

Mesure	distance moyenne	distance max	distance min	std deviation
distance in pixel	2	6	0.2	1.2

TAB. 5.3 – Erreur de rétroprojection suite à la correction de la distorsion

La figure 5.4 montre l'impact de la correction de la distorsion dans la méthode de transfert.

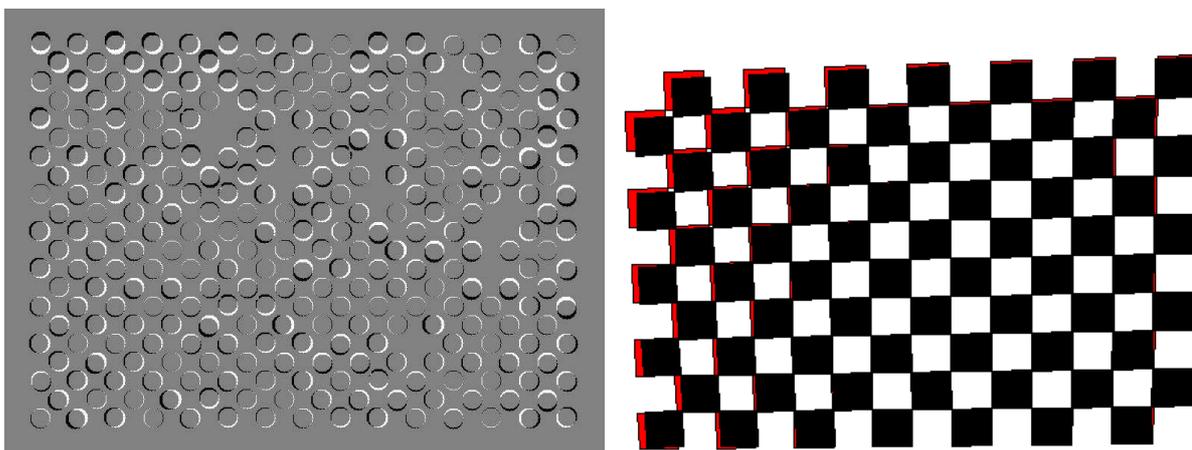


Figure 5.4 – Fusion entre l'image et son estimation après correction de la distorsion et correction de la distorsion sur la synthèse de vue de la mire papier

5.1.4 Projection sur objet réel

La position des sommets du cube sur les images des caméras est récupérée manuellement. Le point de correspondance sur l'image du projecteur est estimé suivant la méthode de transfert, et les faces peuvent alors être dessinées sur l'image. La figure 5.5 présente le résultat obtenu.

La synthèse de vue par la géométrie épipolaire est réalisée sur une maquette de maison. Les points de correspondance sont déterminés manuellement sur les images des caméras et leurs correspondants sont estimés grâce à la géométrie épipolaire. La figure 5.6 montre le résultat.

La précision de l'étalonnage du système a été testée à travers deux applications, la reconstruction 3D et la synthèse de vue, qui toutes deux nécessitent des points de correspondance. La section suivante s'intéresse à l'estimation de ces points.

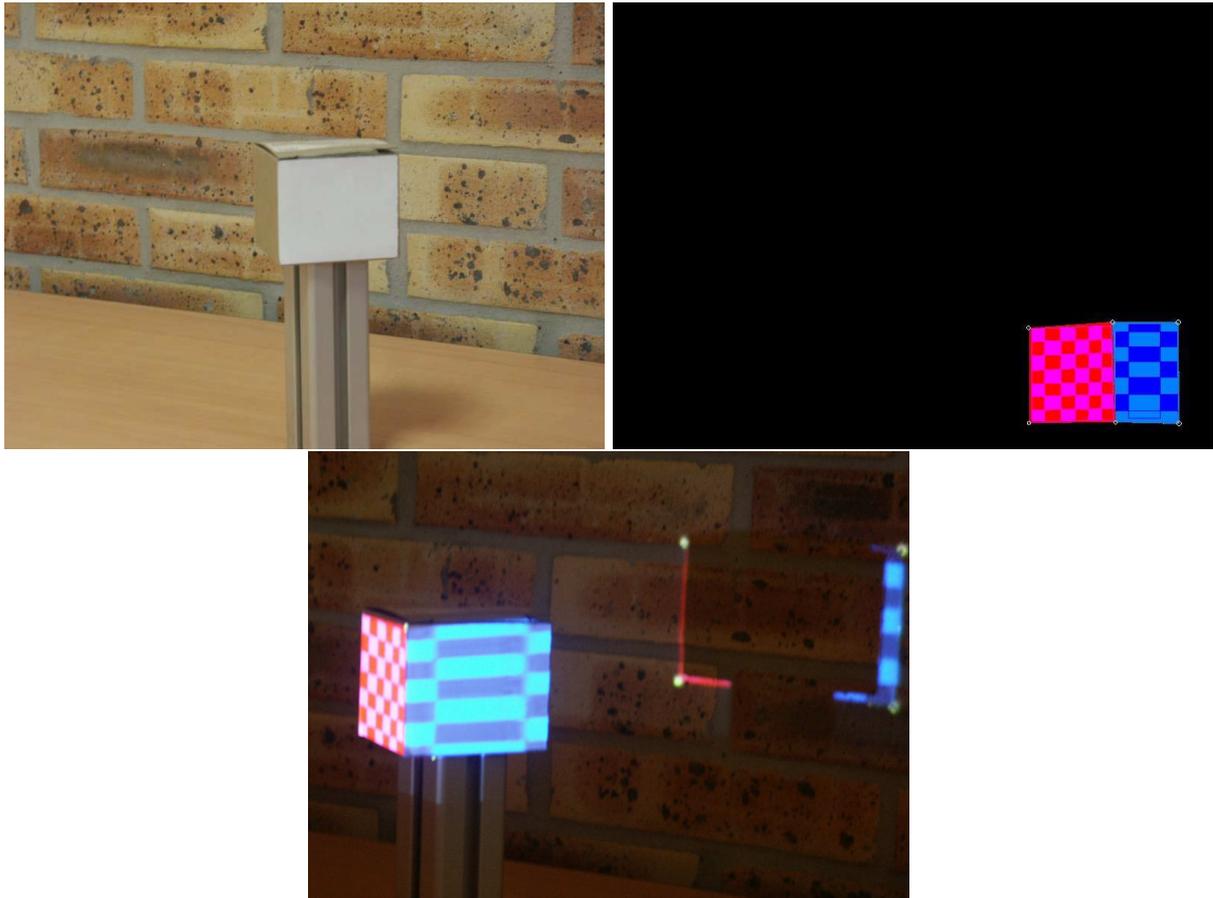


Figure 5.5 – De droite à gauche : image de la caméra, image synthétisée, *texture mapping* sur objet réel



Figure 5.6 – Condamnation de la porte de garage de la maquette par une texture brique

5.2 Lumière Structurée

Une simulation complète du système est implémentée : caméras virtuelles ainsi que projecteur sont ainsi définis. Des points 3D connus sont créés dans l'espace et projetés sur les images des caméras et projecteur. La robustesse de la méthode est testée. Un bruit gaussien de moyenne nulle et de variance σ est ajouté aux points de correspondance sur les deux caméras mais pas sur le projecteur ; en effet la mire du projecteur est parfaitement connue car elle est issue de notre implémentation.

Deux erreurs sont estimées (figure 5.7) :

- La distance D entre les points 3D ajoutés et les points 3D triangulés à partir des points 2D bruités.
- La distance d entre les points 2D de la mire du projecteur et les points rétroprojetés.

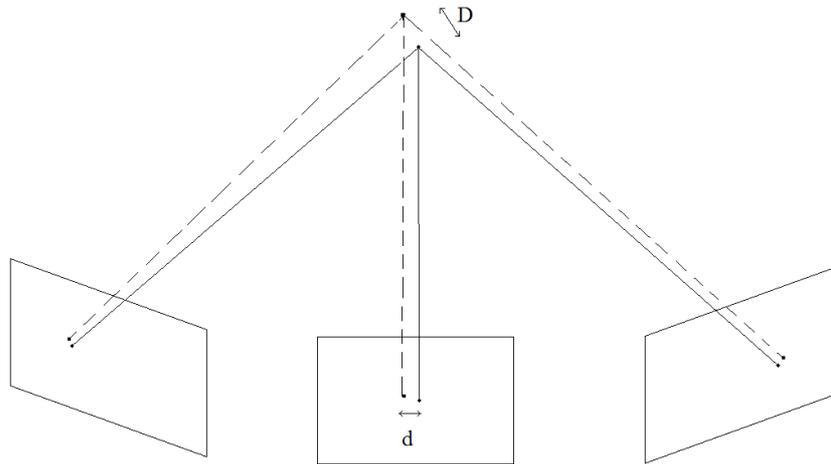


Figure 5.7 – Estimation de l'erreur due au bruit gaussien

5.2.1 Résultats quantitatifs

5.2.1.1 Résultats dans le monde réel

La figure 5.8 présente le pourcentage de points valides (points appariés qui sont des points de correspondance) selon le seuil de corrélation. Sa valeur reste de 70% pour un seuil aux environs de 0.7. Une valeur inférieure à ce seuil ne permet que d'augmenter sensiblement le pourcentage de points aberrants.

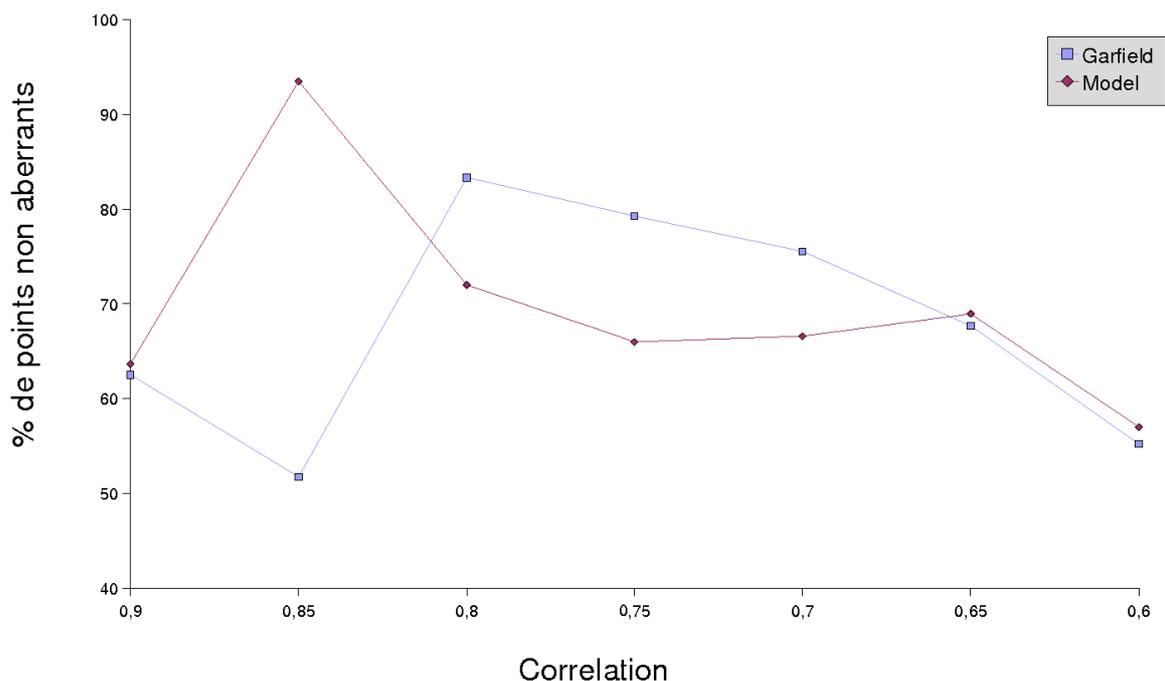


Figure 5.8 – Pourcentage de points valides après un seuil sur la corrélation

On estime la corrélation pour chaque point de la mire. La table 5.4 expose la relation entre le seuil de corrélation et le pourcentage de points valides.

Garfield							
corrélation	0,9	0,85	0,8	0,75	0,7	0,65	0,6
Points après seuil	8	29	66	135	233	405	770
nombre de points 3D	5	15	55	107	176	274	425
Pourcentage de points corrélés	62	52	83	79	75	68	55
Vase							
corrélation	0,9	0,85	0,8	0,75	0,7	0,65	0,6
Points après seuil	22	92	157	235	344	528	888
nombre de points 3D	14	86	113	155	229	364	506
Pourcentage de points corrélés	64	93	72	66	66	68	57

TAB. 5.4 – Erreurs de correspondance pour deux modèles

Plus le seuil est bas plus le nombre de points sélectionnés est important, mais le risque de sélectionner des points aberrants augmente aussi. Une troisième condition est ajoutée sur la

distance 3D des points. Si deux points sont voisins sur la mire du projecteur, alors la distance entre les points 3D correspondants ne peut pas être supérieure à une distance seuil, comme par exemple la distance moyenne entre tous les points 3D reconstruits.

Une fois le nombre de points de correspondance évalué, on peut estimer l'erreur de rétroprojection. La figure 5.9 expose cette erreur, erreur symbolisée par la distance entre le point réel et le point 3D projeté sur le plan image du projecteur. L'erreur moyenne est estimée à 2,7 pixels. Cette erreur apparaît trop importante pour notre objectif final (projeter l'image de synthèse sur un objet). Cette erreur provient de l'erreur de position des points de correspondance sur les images des caméras qui ne sont pas parfaitement centré sur le motif.

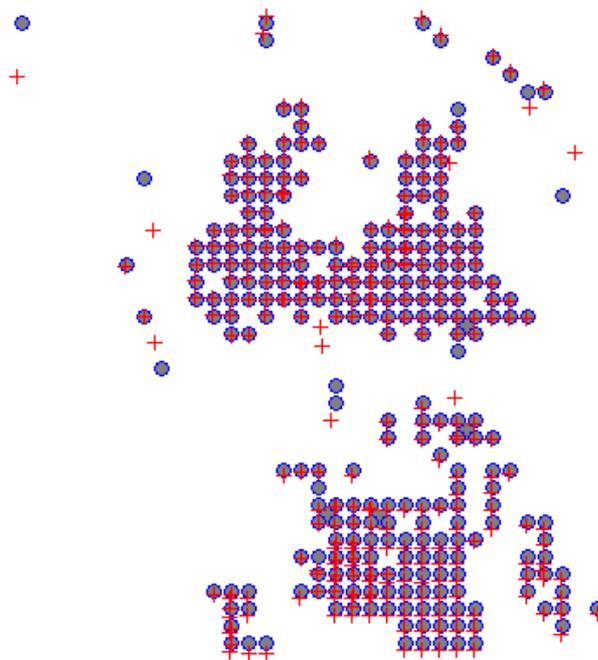


Figure 5.9 – Erreur de rétroprojection entre les point réels (cercle bleu), et les point 3D projetés sur le plan image du projecteur (croix rouge)

5.2.1.2 Résultats dans le monde virtuel

Le tableau 5.5 présentent la robustesse de la méthode face à un bruit gaussien. L'*erreur 3D* est la distance en mm entre les points 3D virtuels et les points triangulés, et l'*erreur de mire* est la distance en pixel entre les points 2D virtuels de la mire et les points 2D retroprojetés.

σ	0	0.1	0.5	1
erreur 3D	0	0.26	1.26	2.56
erreur de mire	0	0.57	2.73	5.39

TAB. 5.5 – Robustesse de la méthode au bruit gaussien

5.2.1.3 Résultats qualitatifs

Nous obtenons plus de points avec la technique de lumière structurée couplée avec la géométrie épipolaire (figure 5.10), qu'avec un détecteur classique comme [Harris and Stephens, 1988] pour un objet dépourvu de texture. Le choix des deux seuils, sur la corrélation et la distance épipolaire conditionne la qualité des résultats : un juste milieu doit être trouvé entre l'identification des point aberrants et l'obtention de points de correspondance.

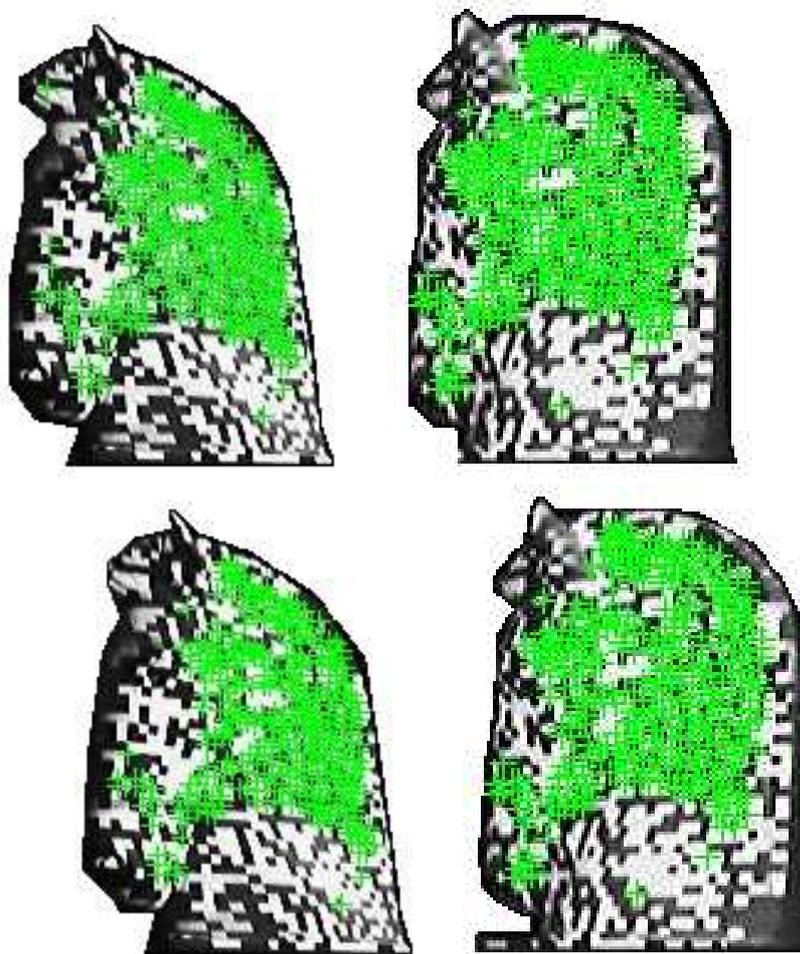


Figure 5.10 – Points de correspondance déterminés avec les seuils de corrélation sur les deux premières images, points de correspondance déterminés avec les seuils de corrélation et de distance épipolaire sur les deux dernières images

La figure 5.11 montre la reconstruction 3D du modèle. Les points aberrants peuvent être identifiés comme des points éloignés du nuage de points, par rapport à la distance moyenne entre les points 3D.

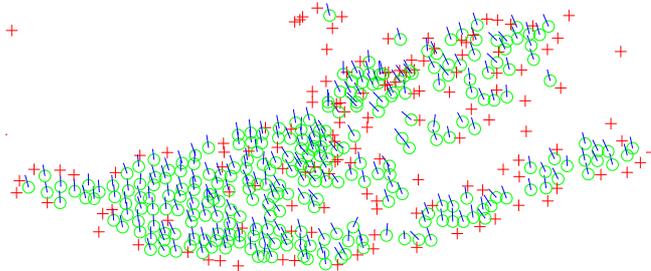


Figure 5.11 – Reconstruction 3D et normales

La figure 5.12 montre l'identification d'un plan. Les deux premières images montrent les points de correspondance estimés après le seuil de corrélation, les deux dernières images montrent les points après le seuil sur la distance épipolaire. Une partie du plan n'est pas identifiée car l'erreur de distance épipolaire est trop grande sur ce test. Cette mauvaise évaluation de la géométrie épipolaire est corrigée sur la figure 5.14.

Enfin la figure 5.13 montre l'identification d'un plan suivant les normales.

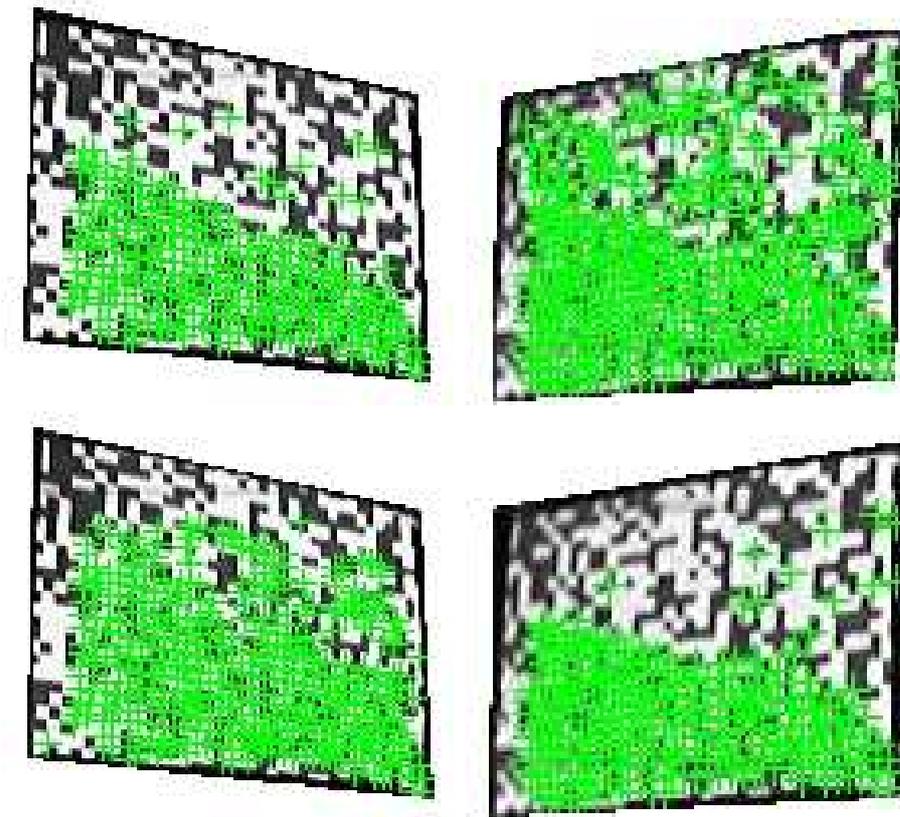


Figure 5.12 – Points de correspondance après le seuil de corrélation, puis de distance épipolaire

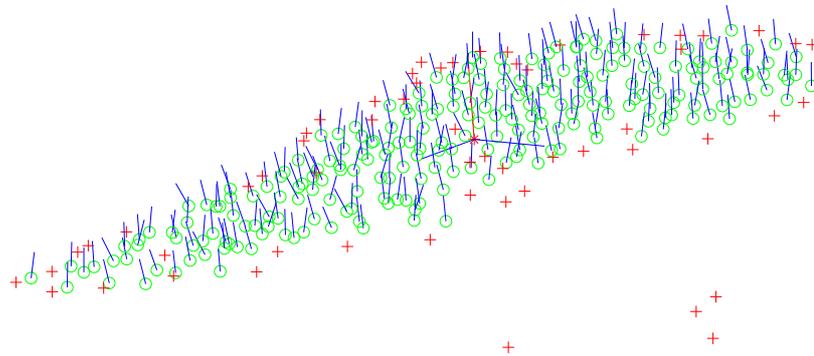


Figure 5.13 – Reconstruction 3D de la surface et estimation des normales

La figure 5.14 montre la projection de textures différentes sur deux plans parallèles. Les deux plans sont reconstruits et ensuite identifiés. La correspondance entre les points 3D et 2D est connue, la zone de la mire correspondant au plan peut donc être reconnue et une texture particulière peut ainsi être projetée.

Deux raisons expliquent l'absence de texture sur les bords des plans :

- La première raison est l'absence d'information due aux occultations créées par la position relative des caméras par rapport au projecteur.
- La deuxième raison est l'absence de capture de cellule complète sur les bords des plans et ainsi l'impossibilité d'identifier des points de correspondance.

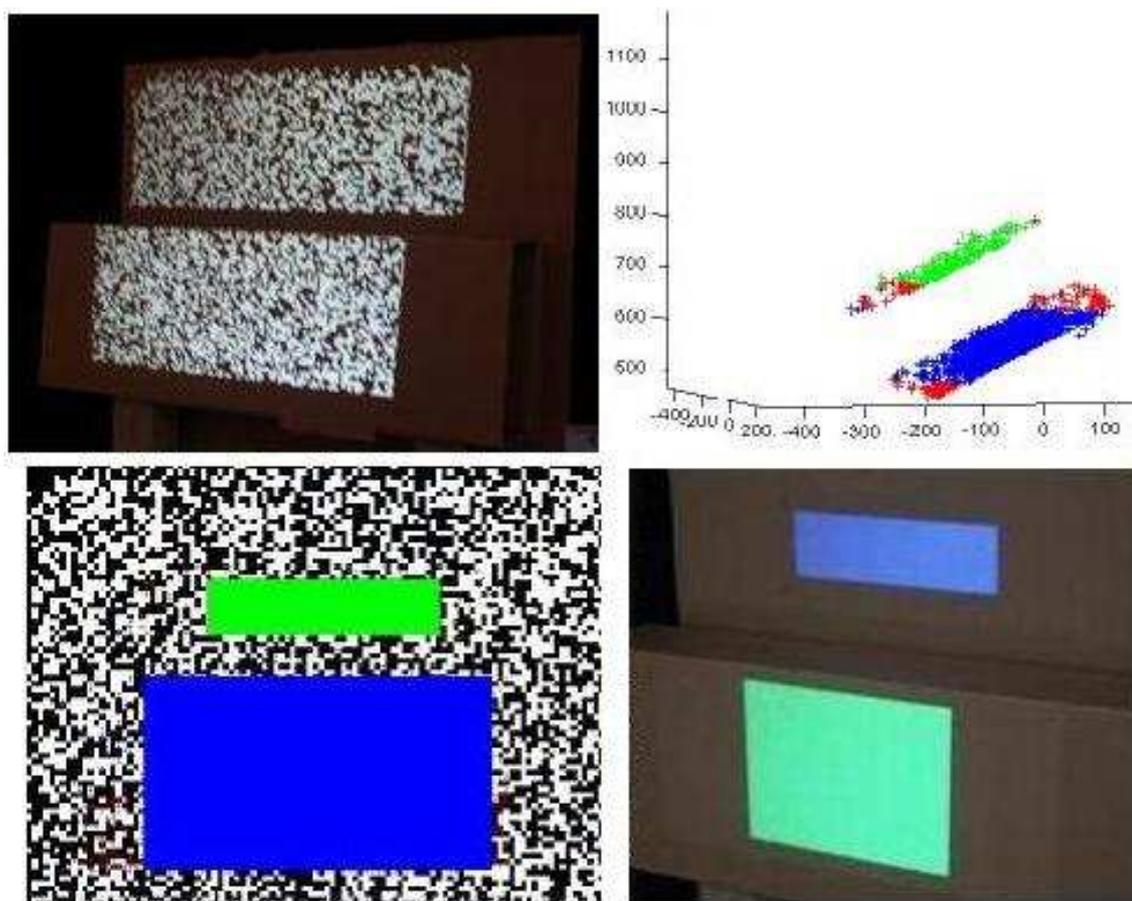


Figure 5.14 – Gauche à droite : Deux plans avec la mire, reconstruction et identification des plans, zone de correspondance entre les points 2D et 3D, projection de la texture

Le test suivant concerne un objet réel, le modèle du vase avec la même méthode, mais cette fois nous avons réalisé une triangulation de Delaunay des points 2D de correspondance de la mire. Seule la partie centrale de l'objet peut être texturée. Cette partie de l'objet est la seule zone visible commune aux trois points de vue de notre système. La figure 5.15 montre les résultats obtenus avec différents seuils de corrélation. Plus le seuil est faible, plus le nombre de points de correspondance validés est élevé. Le pourcentage de points aberrants augmente lui aussi.

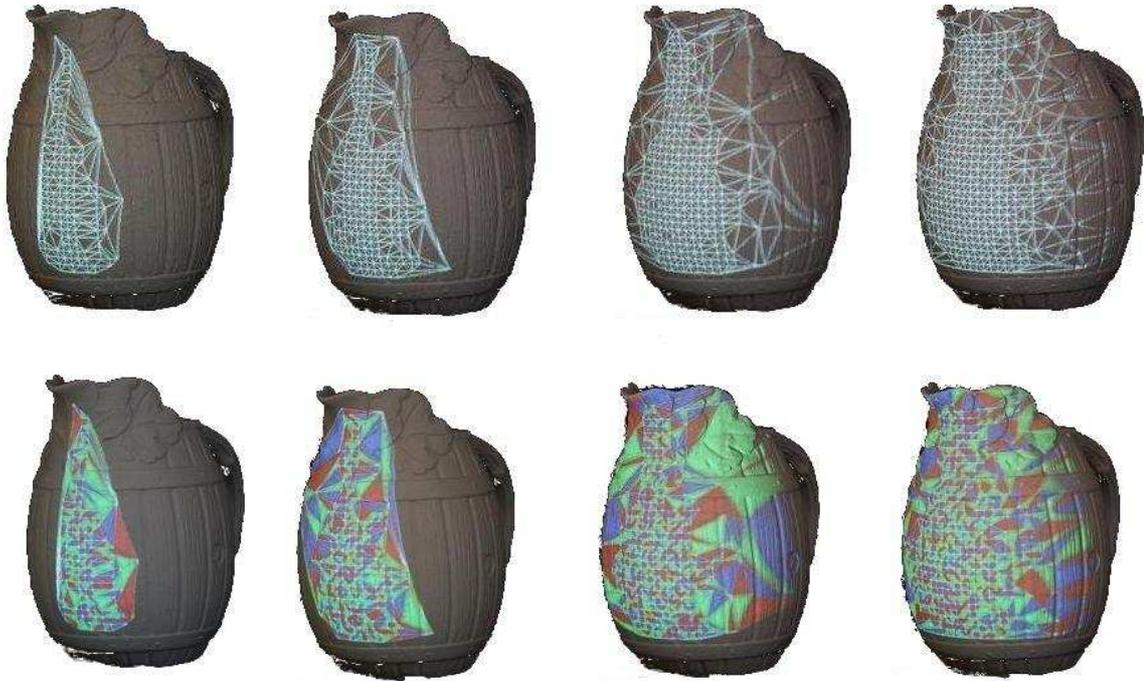


Figure 5.15 – Triangulation de Delaunay des points de correspondance, avec ou sans couleur de facette

5.2.1.4 Comparaison de reconstruction 3D suivant la mire de lumière structurée

Deux reconstructions 3D du même modèle sont exposées dans la figure 5.16 : la reconstruction de gauche est obtenue grâce à la mise en correspondance issue de la mire binaire, alors que celle de droite est estimée par la mire couleur. La mire couleur est plus difficile à mettre en place car elle requiert un contrôle de l'éclairage ambiant, mais ces contraintes sont compensées par le gain en nombre de points plus important, une répartition de ces points plus homogènes et une précision 3D accrue.

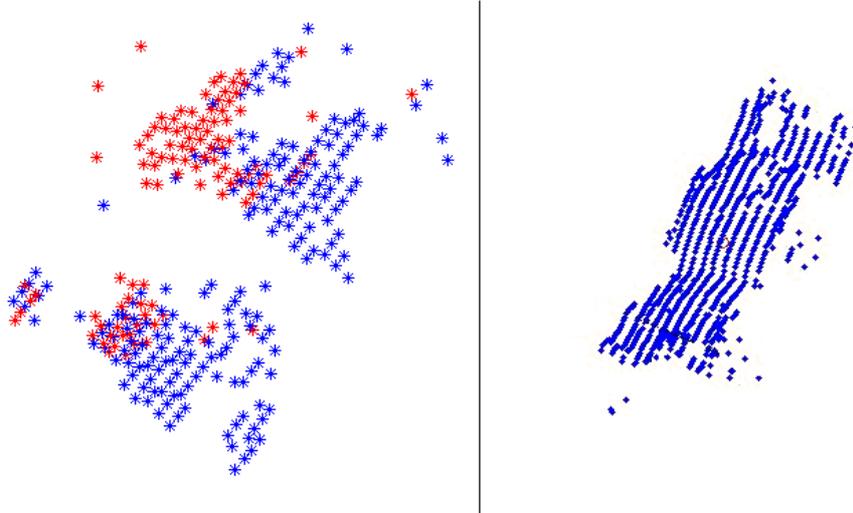


Figure 5.16 – Comparaison des reconstructions 3D : à gauche la mire binaire, à droite la mire couleur

5.3 Projection de texture

Notre objectif initial, projeter de manière adéquate une texture sur un objet, est atteint. La position relative de l'objet restant statique par rapport au projecteur, tous les calculs précédents pour estimer la projection de texture sur l'objet reste valide. On peut donc dissocier la projection de texture et la reconstruction 3D. Nous disposons de deux modèles de forme identique, l'un étant texturé mais pas l'autre (figure 5.17).



Figure 5.17 – Deux modèles de forme identique, un seul étant texturé

5.3.1 Reproduction

La figure 5.18 présente la reproduction de la texture réelle de l'objet de référence sur un objet dépourvu de texture. La texture projetée s'adapte sur l'objet.



Figure 5.18 – Modèle réel et texture virtuelle

La bouche, les yeux ainsi que la pile de pièces de monnaie au bas du modèle sont visuellement bien différenciés du reste du corps et de sa couleur uniforme.

La figure 5.19 montre notre modèle texturé, avec des agrandissements sur des zones particulières du modèle. L'agrandissement en bas à gauche de la figure montre la pile de monnaie au pied du modèle texturé de manière adéquate et sans bavure sur les bords. Le deuxième agrandissement (en haut à droite de la figure) expose une partie du visage. Si le nez et l'oeil sont bien texturés, on remarque une bavure de la texture de la bouche sur le visage du modèle. Cette bavure s'explique par la position relative du projecteur par rapport à la limite de la bouche. La surface en cet endroit est pratiquement parallèle à la direction des rayons lumineux issus du projecteur. Toute erreur due à l'étalonnage, au recalage ou au modèle est amplifiée.



Figure 5.19 – Agrandissement sur les différentes parties du modèle

5.3.2 Reproduction complète

Nous allons réaliser la texturation de l'objet sous différents points de vue pour permettre une reconstruction 3D de l'objet et simuler le mouvement de cet objet. Seules deux vues sur douze seront montrées dans ce paragraphe, les onze vues restantes étant mise en annexe B. Le tableau 5.6 montre l'erreur de distance de recalage : elle estime la distance entre un point de notre reconstruction 3D et le point 3D le plus proche du modèle.

Le diagramme 5.20 permet de mieux visualiser l'erreur de recalage. On remarque une augmentation de l'erreur maximum de recalage pour les dernières vues. Elle s'explique par un ou plusieurs points aberrants persistants et n'est pas révélatrice d'un mauvais recalage. A contrario l'erreur moyenne de recalage est plus explicite : si elle reste faible pour presque toutes les vues, elle se montre en revanche importante pour la vue 10. La zone de recalage disponible pour cette vue est le dos arrondi du modèle, soit une surface courbe sur laquelle le recalage est pauvre. Les douze vues synthétisées et celles de l'objet texturée sont disponible dans l'annexe B.

Numéro de vue	erreur minimale	erreur maximale	erreur moyenne
1	0.28	8.66	3.5
2	0.14	10.09	1.90
3	0.14	8.6	1.94
4	0.26	23.9	6.8
5	0.07	11.2	1.8
6	0.13	6.4	1.7
7	0.09	9.8	1.9
8	0.34	13.59	5.26
9	0.15	5.87	1.86
10	0.44	39.27	17.42
11	0.08	35.16	2.5
12	0.07	25.66	2.37
13	0.044	35.91	3.19

TAB. 5.6 – Estimation de l'erreur de recalage en mm sur le modèle du chat

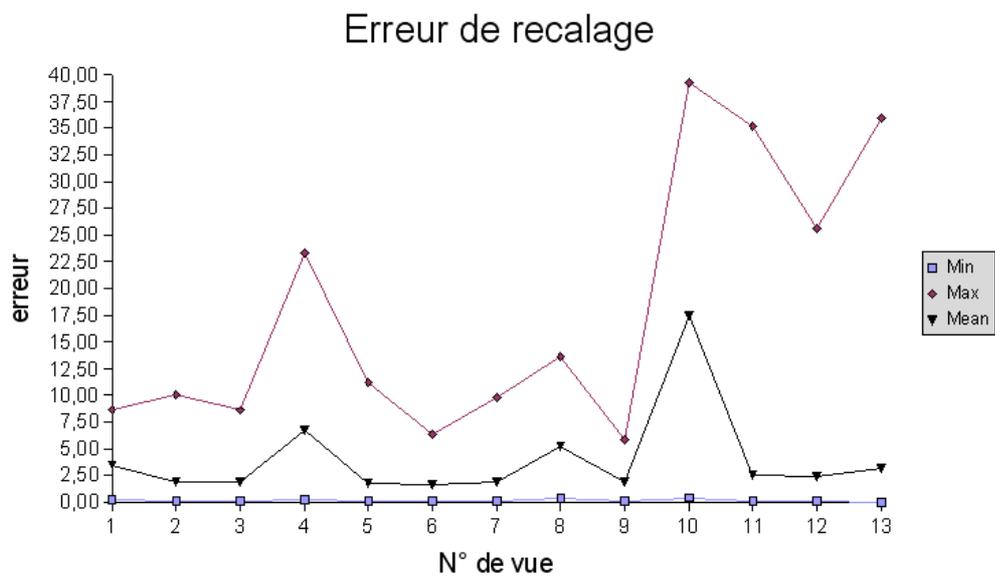


Figure 5.20 – Diagramme d'erreur en mm de recalage

La figure 5.21 montre deux séries images, pour la vue 3 et 10, qui exposent l'image synthétisée et l'objet réel avec sa texture virtuelle. Si la vue 3 est correcte, par contre la vue 10 est mauvaise de part un pauvre recalage.

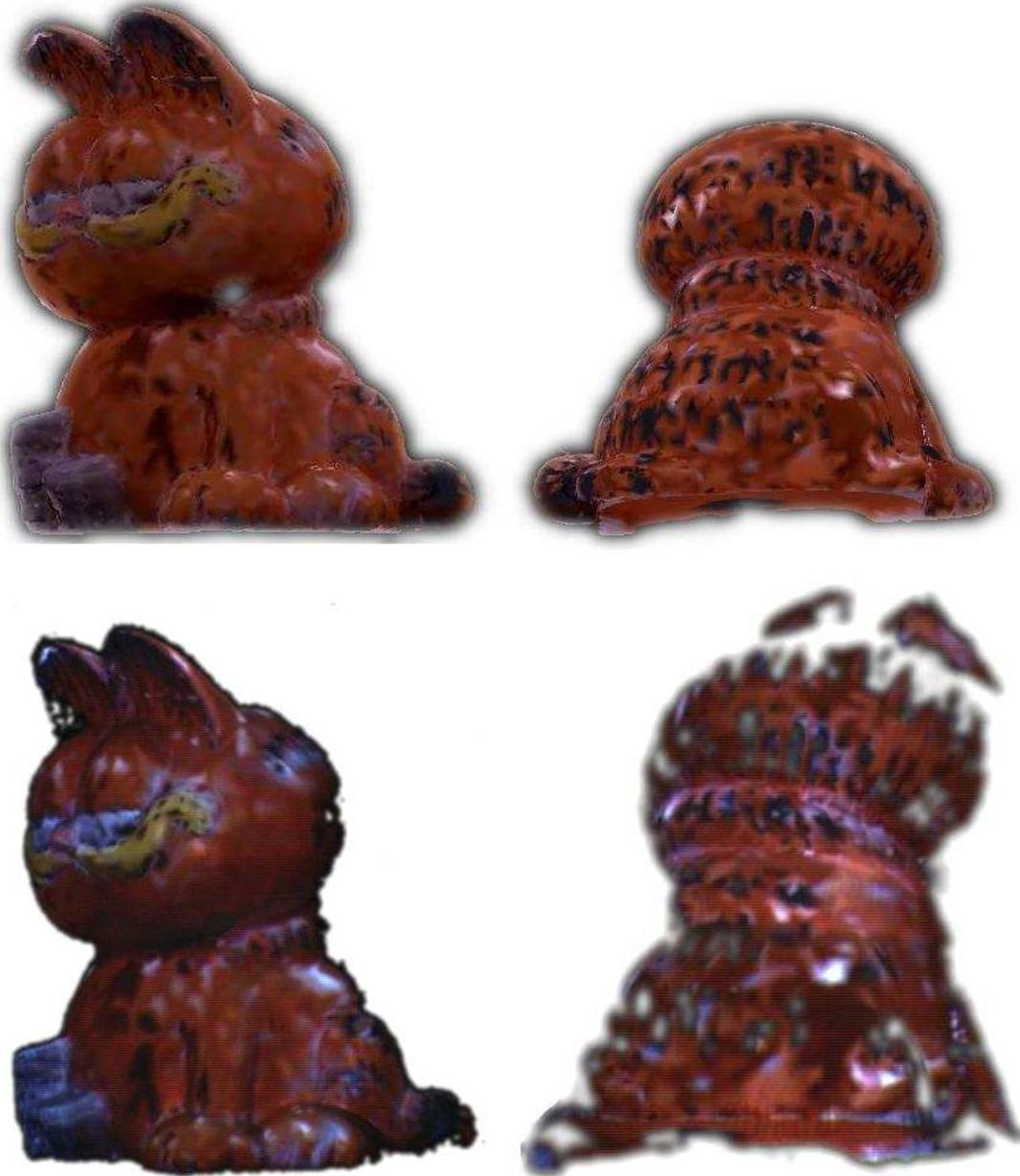


Figure 5.21 – 2 vues de garfiled : de gauche à droite : synthèse de vue, objet réel et texture virtuelle, haut à bas : vue 3 et 10

5.3.3 Changement de couleur

L'avantage d'avoir une texture virtuelle est de pouvoir la modifier sans modification de la géométrie 3D. Géométrie et texture peuvent être dissociées. Ainsi par segmentation on identifie des plages de couleur (dans ce cas l'orange, le blanc et le jaune) et modifie ces couleurs.

La figure 5.22 montre un tel résultat.



Figure 5.22 – Modèle réel et texture virtuel, segmentation synthèse de vue

5.3.4 Surimpression : altération de texture d'un objet

La figure 5.23 montre les résultats d'une autre application de surimpression, basée sur la segmentation des couleurs. Le modèle que nous voulons texturer est un objet qui possède déjà une texture. Les différentes images servent à mettre en exergue différentes parties du modèles : que ce soit le corps, la bouche, ou les yeux.



Figure 5.23 – Surimpression : la texture de l'objet est modifiée pour mettre en valeur certaines parties

Le paragraphe suivant montre les résultats similaires obtenues avec la même méthode mais pour des modèles différents.

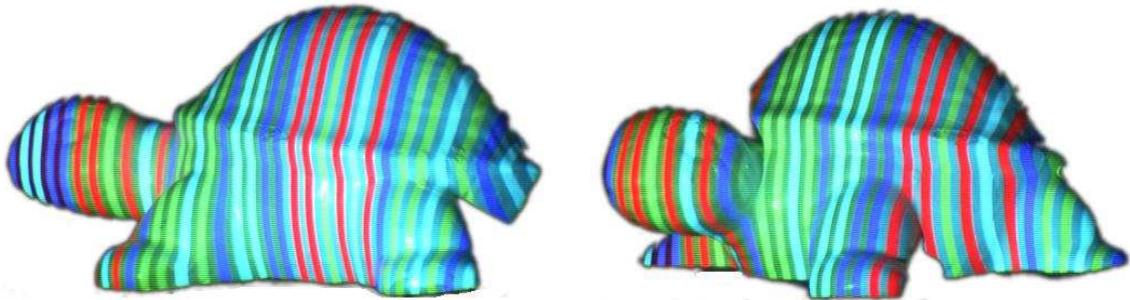
5.3.5 Autres modèles

La figure 5.24 expose les quatre modèles employés. Le premier modèle est présenté pour permettre une comparaison de taille avec les autres modèles. Dans ce paragraphe nous allons montrer les résultats obtenus pour les trois nouveaux objets dépourvus de texture, le jardinier, le nain de jardin et la tortue.

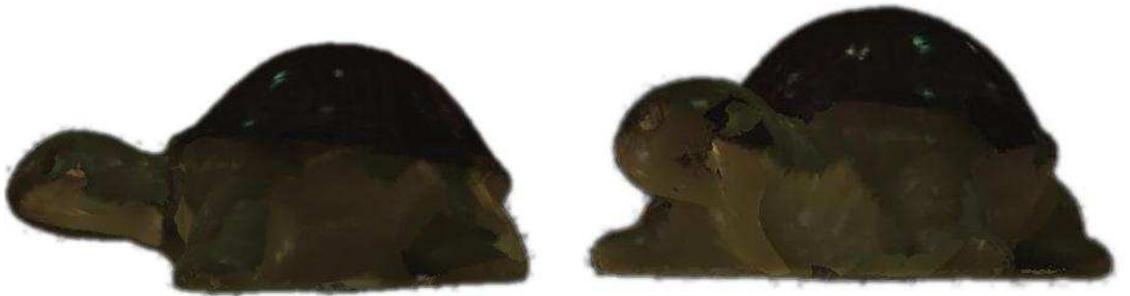


Figure 5.24 – Quatre objets dépourvus de texture

La figure 5.25 montre la vue de l'objet avec la mire et la série d'images que nous retrouverons pour tous nos objets : la vue synthétisée du projecteur et enfin l'objet réel texturé. La tortue permet de vérifier si un objet en profondeur, la tête de l'objet et son corps ne sont pas à la même profondeur, peut être texturé avec la méthode proposée.



(a) Objets texturés par la mire de Pages



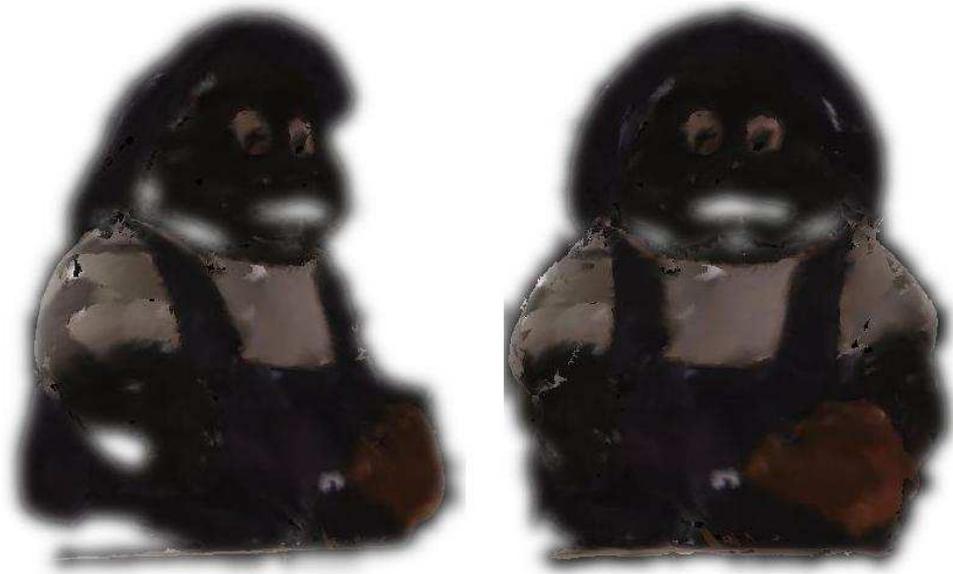
(b) Synthèses de vue



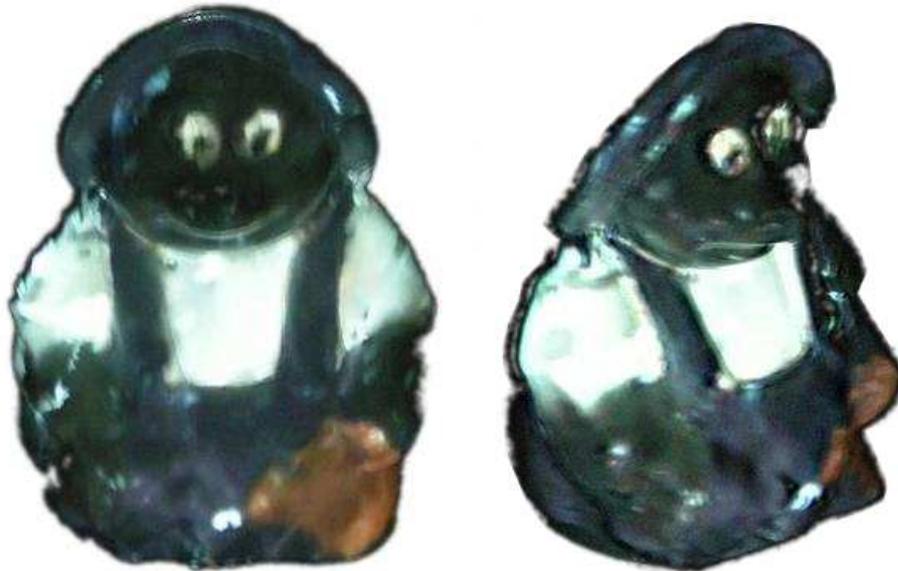
(c) Objets texturés

Figure 5.25 – tortue

Le deuxième objet testé est le jardinier, un objet compact (figure 5.26). Le recalage doit être fin pour bien différencier les différentes zones du modèle (tête, corps, bretelle et pantalon). La figure 5.26 montre que telle est bien le cas.



(a) Synthèses de vue



(b) Objets texturés

Figure 5.26 – Jardinier

Les figures 5.27 montrent la possibilité d'appliquer cette technique sur des objets plus fins, en respectant la perspective. De fait la taille minimale des objets que nous pouvons texturer avec notre méthode correspond à trois bandes de la mire couleur, ses trois bandes étant la condition minimale à l'identification de la mire sur l'objet.



(a) Synthèses de vue



(b) Objets texturés

Figure 5.27 – nain

5.3.5.1 Vue des 3 quarts

La figure 5.28 montre les limites de notre méthode. Plutôt que d'observer les objets texturés virtuellement du point de vue du système, les modèles sont observés de côté. On observe que si la texturation est adéquate, elle reste partielle car soumise au point de vue du projecteur : toute surface cachée au projecteur demeure sans texture. Une solution serait de rajouter un projecteur pour couvrir ces surfaces cachées mais cet ajout engendre de nouveaux problèmes, tels que la compensation radiométrique entre les projecteurs et les zones de recouvrement, problèmes traités pour une surface de projection plane [Majumder et al., 2003].



Figure 5.28 – Vue de côté

5.3.5.2 Erreur de recalage

Le tableau 5.7 montre l'erreur de recalage pour les trois objets précédemment présentés. Les valeurs sensiblement les mêmes pour les trois objets montrent l'insensibilité de la méthode face à différents objets.

objet	erreur minimale	erreur maximale	erreur moyenne
tortue de face	0.15	11.87	2.96
tortue de 3 quart	0.12	15.83	2.69
nain de jardin de face	0.15	14.40	2.75
nain de jardin de 3 quart	0.11	10.00	2.34
taupe de face	0.14	8.67	2.18
taupe de 3 quart	0.10	12.24	2.7

TAB. 5.7 – Estimation de l'erreur en mm de recalage

5.4 Synthèse

La synthèse de vue peut se décomposer en trois étapes :

- Recalage entre la reconstruction 3D et un modèle plus fin.
- Rétroprojection du modèle sur le plan image du projecteur.
- Texturation de l'image.

Le recalage, nécessite une attention particulière car de la précision du recalage découle l'adéquation de la texture projetée sur l'objet. Une estimation assez grossière de la position de l'objet par rapport au projecteur, suivi d'un recalage fin basé sur la méthode ICP fournit des résultats satisfaisants. La deuxième étape dépend de la qualité de l'étalonnage du système effectué préalablement. Enfin, la dernière étape, totalement indépendante des deux étapes précédentes, assure la finition de l'image en ajoutant la texture au maillage de points projetés sur l'image.

Une fois la position de l'objet estimé, la projection du modèle dans le plan image permet d'estimer la position des points 2D sur l'image du projecteur. L'application de texture à ce nuage de points se fait de manière autonome par rapport à la géométrie. On peut ainsi changer la couleur ou mettre en avant certaines parties du modèle facilement.

Chapitre 6

Conclusions et perspectives

Ce chapitre présente les conclusions et quelques perspectives ouvertes par nos travaux. Les contributions scientifiques sont abordées ainsi qu'une liste de publications de nos travaux est présentée.

6.1 Conclusion

La frontière entre le monde réel et le monde virtuel tend à s'amenuiser avec les progrès technologiques. La réalité augmentée permet de projeter des images virtuelles dans le monde réel, à travers un projecteur. Dans le premier chapitre de cette thèse, nous avons présenté les principaux systèmes de réalité augmentée, composée de caméra(s) et projecteur(s), qui permettent de dépasser l'emploi standard du projecteur. Les principaux axes de développement sont les systèmes multi-projecteurs pour circonvier les problèmes de chevauchement, la compensation radiométrique et la correction d'images. Mais tous ces systèmes se contentent de projeter leur image sur un écran plat ou légèrement incurvé et n'adaptent pas l'image à la zone de projection.

Nous proposons un système flexible pour projeter une image ou texture sur un objet de forme géométrique quelconque. Ce système se compose de deux caméras et d'un projecteur, mais toutes les étapes décrites précédemment peuvent être généralisées à plus de composants.

L'étalonnage est la première étape à réaliser, étape indépendante de l'objet. Elle permet de déterminer les paramètres intrinsèques et extrinsèques des différents composants du système et se décompose en deux étapes : d'abord les caméras sont étalonnées, et seulement par la suite

le projecteur l'est aussi. Une fois le système étalonné, toutes les coordonnées exprimées le sont dans le repère lié au projecteur. Cette étape assure le lien entre le monde réel et virtuel.

Une fois le système étalonné, tout est prêt. Il faut alors assurer la transition vers le monde virtuel et projeter cet objet réel dans le repère virtuel du projecteur. Pour projeter de manière adéquate une image sur l'objet, il faut déterminer la position et l'orientation de cette surface dans le référentiel du projecteur. L'emploi de la technique de lumière structurée va permettre de reconstruire les surfaces visibles du point de vue du projecteur par la projection de mire pseudo-aléatoire, image dans laquelle un code est inséré et qui permet ainsi d'identifier des points de correspondance entre l'image du projecteur et celle perçue par les caméras qui voient l'objet réel texturé par cette mire. Cette mise en correspondance permet ensuite par triangulation de reconstruire un nuage de points et ainsi la surface dans le monde virtuel.

Nous utilisons une reconstruction complète de l'objet, obtenue soit par un scanner 3D soit par notre système grâce à la prise de plusieurs vues, comme dans le cas du modèle de garfield, où douze images ont été prises. L'étape suivante est le recalage entre la reconstruction partielle et le modèle plus complet. Un rapprochement grossier, comme la méthode ACP, suivi d'un recalage plus fin qui nécessite une estimation d'une matrice de rotation et de translation pour l'initialisation d'une méthode basée sur l'ICP sont les deux étapes.

La dernière étape de notre méthode est la synthèse de vue. Elle consiste à rétroprojeter le modèle recalé dans le repère du projecteur sur le plan image du projecteur, grâce aux paramètres intrinsèques estimés lors de l'étalonnage du système. Nous avons testé notre méthode sur différents objets et différentes positions et la texturation virtuelle de l'objet apparaît visuellement adéquate, l'étape de recalage apparaissant comme la plus critique.

6.2 Perspectives

Des évolutions à notre méthode sont envisageables :

1. Seule la surface visible du point de vue du projecteur peut être texturée donc l'emploi d'un seul projecteur ne permet pas de texturer entièrement un objet réel 3D. L'emploi de plusieurs projecteurs permettrait de compenser sur les occultations, mais engendreraient les problèmes classiques de recouvrement entre projecteur et nécessiteraient un étalonnage photométrique, qui a été réalisé pour le cas d'écran plat.
2. Statique à Dynamique : la mosaïque de synthèse de vue du modèle garfield laisse présager une application en temps réel : la technique de lumière structurée imperceptible combinée

avec l'utilisation d'un environnement optimisé pourra permettre la texturation en temps réel d'objet réel et donc aussi d'objets en mouvement.

3. La correction radiométrique pour projeter des couleurs en fonction du rendu souhaité.

6.3 Publications

Ces travaux ont fait l'objet de publications dans :

– 2 revues

- "Trifocal Tensor as a Tool for Modelling an Imperceptible Structured Light Sensor", T. Molinier, D. Fofi, F. Meriaudeau, R. SEULIN, *Journal of Optical Technology*, Optical Society of America, 74 (4), pp. 282-287, April 2007,
- "Auto-Calibrage à partir de Deux Vues et de la Projection d'Un Motif Lumineux", T. Molinier, D. Fofi, *J3eA, Journal sur l'enseignement des sciences et technologies de l'information et des systèmes*, Hors-série spécial "Eclairage Structuré pour la Vision Active", EDP Sciences, 4 (3 - 002), 2005,

– 6 conférences internationales

- "Projector View Synthesis and Virtual Texturing", T. Molinier, D. Fofi, J. Salvi, P. Gorria, F. Meriaudeau, The 2nd International Topical Meeting on Optical Sensing and Artificial Vision, Saint-Petersburg, Russia, May 2008.
- "2D Virtual Texture on 3D Real Object with Coded Structured Light", T. Molinier, D. Fofi, J. Salvi, P. Gorria, *SPIE Electronic Imaging - Image Processing : Machine Vision Applications*, January 2008.
- "Detection of Surfaces for Projection of Texture", T. Molinier, D. Fofi, P. Gorria, J. Salvi, *IEEE/SPIE 8th International Conference on Quality Control by Artificial Vision (QCAV'2007)*, Le Creusot, France, May 2007.
- "Novel View Synthesis for Projective Texture Mapping on Real 3D Objects", T. Molinier, D. Fofi, P. Gorria, *SPIE Electronic Imaging - Machine Vision Applications in Industrial Inspection XIV*, San José, USA, January 2006,
- "Self-Calibration of a Trinocular Sensor with Imperceptible Structured Light and Varying Intrinsic Parameters", T. Molinier, D. Fofi, F. Meriaudeau, *7th International Conference on Quality Control by Artificial Vision*, Nagoya, Japan, May 2005,
- "Trifocal Tensor Estimation Applied to Imperceptible Structured Light", T. Molinier, D. Fofi, Ralph SEULIN, F. Meriaudeau, *The International Topical Meeting on Optical Sensing and Artificial Vision*, Saint Petersburg, Russia, October 2004,

- **1 conférence nationale** "Projection de Texture sur des Objets 3D Réels", T. Molinier, D. Fofi, P. Gorria, J. Salvi, F. Meriaudeau, Colloque du Groupe de Recherche sur le Traitement du Signal et des Images (GRETSI), Troyes, France, Septembre 2007,
- **1 workshop international** "Automatic Texture Mapping on Real 3D Model", T. Molinier, D. Fofi, P. Gorria, J. Salvi, IEEE International Workshop on Projector-Camera Systems, in conjunction with CVPR'2007, Minneapolis, USA, June 2007.

6.4 Collaboration scientifique

Cette thèse s'est déroulée sur les années 2004 - 2008. Grâce aux liens unissant l'université de Bourgogne à celle de Gérone, cette thèse s'est partagée sur deux centres. La majeure partie s'est passée sur le site du laboratoire Le2i du Creusot et deux trimestres ont eu lieu dans le laboratoire VICOROB de l'université de Gérone, au printemps 2006 et 2007. Ces stages m'ont permis de profiter des connaissances du laboratoire VICOROB et de Mr Salvi au sujet des techniques de lumière structurée.

Bibliographie

- [Armangué et al., 2003] Armangué, X., Araújo, H., and Salvi, J. (December 2003). A review on egomotion by means of differential epipolar geometry applied to the movement of a mobile robot. *Pattern Recognition*, pp. 2927-2944, Vol. 36, Issue 12,.
- [Armangué and Salvi, 2003] Armangué, X. and Salvi, J. (2003). Overall view regarding fundamental matrix estimation. volume 21, pages 205–220.
- [Ashdown and Robinson, 2003] Ashdown, M. and Robinson, P. (2003). Experiences implementing and using personal projected displays. Mark Ashdown and Peter Robinson. Experiences Implementing and Using Personal Projected Displays. In *IEEE International Workshop on Projector-Camera Systems (Procams 2003)*, 2003.
- [Audet and Cooperstock, 2007] Audet, S. and Cooperstock, J. (2007). Shadow removal in front projection environments using object tracking. *IEEE Projector-Camera Systems (ProCams)*.
- [Azuma, 1995] Azuma, R. T. (1995). A survey of augmented reality. *ACM SIGGRAPH '95 (Los Angeles)*.
- [Bandyopadhyay et al., 2001] Bandyopadhyay, D., Raskar, R., and Fuchs, H. (2001). Dynamic shader lamps : Painting on movable objects. *Proc. IEEE and ACM Int. Symposium on Augmented Reality*.
- [Batlle et al., 1998] Batlle, J., Mouaddib, E., and Salvi, J. (1998). Recent progress in coded structured light as a technique to solve the correspondence problem : a survey. In *Pattern Recognition*,, pages 31(7) :963–982.
- [Beardsley et al., 2005] Beardsley, P., Forlines, C., Raskar, R., and VanBaar, J. (2005). Hand-held projectors for mixing physical and digital textures. In *CVPR '05 : Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, page 112, Washington, DC, USA. IEEE Computer Society.
- [Besl and McKay, 1992] Besl, P. J. and McKay, N. D. (1992). A method for registration of 3d shapes. *IEEE Trans. PAMI* 14, 239-256.
- [Bimber and Raskar, 2005] Bimber, O. and Raskar, R. (2005). *Spatial Augmented Reality : Merging Real and Virtual Worlds*. A. K. Peters, Ltd., Natick, MA, USA.

- [Blinn, 1978] Blinn, J. F. (1978). Simulation of wrinkled surfaces. In *SIGGRAPH '78 : Proceedings of the 5th annual conference on Computer graphics and interactive techniques*, pages 286–292, New York, NY, USA. ACM Press.
- [Borkowski et al., 2003] Borkowski, S., Riff, O., and Crowley, J. L. (2003). Projecting rectified images in an augmented environment. In *International workshop on Projector-Camera Systems ProCams*. IEEE Computer Society Press.
- [Bouguet, 2004] Bouguet, J.-Y. (2004). Camera calibration toolbox for matlab, caltech.
- [Bérard, 2003] Bérard, F. (2003). The magic table : Computer-vision based augmentation of a whiteboard for creative meetings.
- [Brown, 1971] Brown, D. C. (1971). Close-range camera calibration. *Photogrammetric Engineering*, 37 :855–866.
- [Buchanan, 1998] Buchanan, J. W. (1998). Simulating wood using a voxel approach. *Comput. Graph. Forum*, 17(3) :105–112.
- [Chen and Kak, 1987] Chen, C. and Kak, A. (1987). Modelling and calibration of a structured light scanner for 3d robot vision. In : *IEEE conference on robotics and automation*.
- [Chua and Jarvis, 1997] Chua, C. S. and Jarvis, R. (1997). Point signatures : A new representation for 3d object recognition. *Int. J. Comput. Vision*, 25(1) :63–85.
- [Chung et al., 1998] Chung, D., Yun, I., and Lee, S. (1998). Registration of multiple range views using the reverse calibration technique. *Pattern Recognit.*, vol. 31, no.4.
- [Cook, 1984] Cook, R. L. (1984). Shade trees. In *SIGGRAPH '84 : Proceedings of the 11th annual conference on Computer graphics and interactive techniques*, pages 223–231, New York, NY, USA. ACM Press.
- [Cossairt et al., 2007] Cossairt, O. S., Napoli, J., Hill, S. L., Dorval, R. K., and Favolora, G. E. (2007). Occlusion-capable multiview volumetric three-dimensional display. *Appl. Opt.*, 46(8) :1244–1250.
- [Cotting et al., 2004] Cotting, D., Naef, M., Gross, M., and Fuchs, H. (2004). Embedding imperceptible patterns into projected images for simultaneous acquisition and display. In *ISMAR '04 : Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 100–109, Washington, DC, USA. IEEE Computer Society.
- [Cruz-Neira et al., 1992] Cruz-Neira, C., Sandin, D. J., DeFanti, T. A., Kenyon, R. V., and Hart, J. C. (1992). The cave : audio visual experience automatic virtual environment. *Commun. ACM*, 35(6) :64–72.
- [Danielsson, 1980] Danielsson, P.-E. (1980). Euclidean distance mapping. In *Computer Graphics and Image Processing*, pages 14, pp. 227–248.

- [Dischler and Ghazanfarpour, 2001] Dischler, J.-M. and Ghazanfarpour, D. (2001). A survey of 3d texturing. *Computers & Graphics*, 25(10).
- [Etzion, 1988] Etzion, T. (1988). Constructions for perfect maps and pseudorandom arrays. In *IEEE Transactions on information theory* 34(5), pages 1308–1316.
- [F. Blais, 1986] F. Blais, M. R. (1986). Real-time numerical peak detector.
- [Faugeras, 1993] Faugeras, O. (1993). *Three-dimensional computer vision : a geometric viewpoint*. MIT Press, Cambridge, MA, USA.
- [Faugeras and Papadopoulo, 1995] Faugeras, O. and Papadopoulo, T. (January 1998. 20 1070, Cambridge, Massachusetts, USA, 1995). A nonlinear method for estimating the projective geometry of 3 views. In Sixth International Conference on Computer Vision, pages 477–484.
- [Faugeras and Toscani., 1986] Faugeras, O. D. and Toscani., G. (June 1986). The calibration problem for stereo. In *In Proc. Conf. Computer Vision and Pattern Recognition, Miami, FL.,* pages 15–20.
- [Fleischer et al., 1995] Fleischer, K. W., Laidlaw, D. H., Currin, B. L., and Barr, A. H. (1995). Cellular texture generation. *Proceedings of SIGGRAPH, Computer Graphics*, 29(4) :239–248.
- [Fofi et al., 2004] Fofi, D., Sliwa, T., and Voisin, Y. (2004). A comparative survey on invisible structured light. In *Machine Vision Applications in Industrial Inspection XII. Edited by Price, Jeffery R. ; Meriaudeau, Fabrice. Proceedings of the SPIE, Volume 5303, pp. 90-98,*
- [Forest et al., 2004] Forest, J., Salvi, J., Cabruja, E., and Pous, C. (2004). Laser stripe peak detector for 3d scanners, a fir filter approach. In *17th International Conference on Pattern Recognition, ICPR 2004, vol 3, Cambridge (Reino Unido)*, pages 646–649.
- [Fredricksen, 1982] Fredricksen, H. (1982). A survey of full length nonlinear shift register cycle algorithms. *Society of Industrial and Applied Mathematics Review*, 24(2) :195–221.
- [Garcia Campos, 2001] Garcia Campos, R. (2001). A proposal to estimate the motion of an underwater vehicle through visual mosaicking. In *IA Electrònica, Informàtica i Automàtica, University of girona, Spain.*
- [Gardner, 1985] Gardner, G. Y. (1985). Visual simulation of clouds. In *SIGGRAPH '85 : Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pages 297–304, New York, NY, USA. ACM Press.
- [Gomez et al., 2005] Gomez, J.-F. V., Simon, G., and Berger, M.-O. (2005). Calibration errors in augmented reality : A practical study. In *ISMAR '05 : Proceedings of the Fourth IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 154–163, Washington, DC, USA. IEEE Computer Society.

- [Griffin et al., 1992] Griffin, P., Narasimhan, L., and Yee, S. (1992). Generation of uniquely encoded light patterns for range data acquisition. In *Pattern Recognition 25(6)*, pages 609–616.
- [Grundhöfer and Bimber, 2008] Grundhöfer, A. and Bimber, O. (2008). Real-time adaptive radiometric compensation. *IEEE Transactions on Visualization and Computer Graphics*, 14(1) :97–108.
- [Gupta and Jaynes, 2005] Gupta, S. and Jaynes, C. (2005). Active pursuit tracking in a projector-camera system with application to augmented reality. In *CVPR '05 : Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, page 111, Washington, DC, USA. IEEE Computer Society.
- [Habed, 2000] Habed, B. (2000). Novel view synthesis : a comparative analysis study. (*Vision Interface, Montrea(Canada)*).
- [Hall et al., 1982] Hall, E. L., Tio, J. B. K., McPherson, C. A., and Sadjadi, F. A. (1982). Measuring curved surfaces for robot vision. *Computer*, 15(12) :42–54.
- [Hamada and Sato, 2007] Hamada, K. and Sato, J. (2007). Calibration and image generation of mobile projector-camera systems. In *ICIAP 07 : Proceedings of the 14th International Conference On Image Analysis and Processing*, pages 93–98, Washington, DC, USA. IEEE Computer Society.
- [Harris and Stephens, 1988] Harris and Stephens, M. (1988). A combined corner and edge detector. *Proceedings of the 4th Alvey Vision Conference*, pages 147–151.
- [Hartley and Zisserman, 1994] Hartley, R. and Zisserman, A. (May 1994). The algebraic approach to invariance. ESPRIT Workshop.
- [Hartley, 1995] Hartley, R. I. (1995). In defence of the 8-point algorithm.
- [Hartley and Zisserman, 2004] Hartley, R. I. and Zisserman, A. (2004). *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN : 0521540518, second edition.
- [Heckbert, 1986] Heckbert, P. (1986). Survey of texture mapping. In *IEEE Computer Graphics and Applications*, pages 56–67.
- [Heikkila and Silven, 1997] Heikkila, J. and Silven, O. (1997). A four-step camera calibration procedure with implicit image correction. In *CVPR '97 : Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, page 1106, Washington, DC, USA. IEEE Computer Society.
- [Jaynes and Ramakrishnan, 2003] Jaynes, C. and Ramakrishnan, D. (2003). Super-resolution composition in multi-projector displays. In *IEEE Workshop on Projector-Camera Systems, In Conjunction with International Conference on Computer Vision, Nice, France*.

- [Jaynes et al., 2001] Jaynes, C., Webb, S., Steele, R. M., Brown, M., and Seales, W. B. (2001). Dynamic shadow removal from front projection displays. In *VIS '01 : Proceedings of the conference on Visualization '01*, pages 175–182, Washington, DC, USA. IEEE Computer Society.
- [Johnson and Fuchs, 2007] Johnson, T. and Fuchs, H. (2007). Real-time projector tracking on complex geometry using ordinary imagery. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0 :1.
- [Jones et al., 2007] Jones, A., McDowall, I., Yamada, H., Bolas, M., and Debevec, P. (2007). Rendering for an interactive 360° light field display. *ACM Trans. Graph.*, 26(3) :40.
- [Kajiya, 1986] Kajiya, J. T. (1986). The rendering equation. In *SIGGRAPH '86 : Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 143–150, New York, NY, USA. ACM Press.
- [Lathuilière et al., 2006] Lathuilière, A., Mansouri, A., Voisin, Y., Marzani, F., and Gouton, P. (2006). Stereoscopic system for 3d reconstruction using multispectral camera and lcd projector. *The Imaging Science Journal, Maney Publishing*, 54 (1) :19–28.
- [Majumder et al., 2003] Majumder, A., Jones, D., McCrory, M., Papka, M., and Stevens, R. (2003). Using a camera to capture and correct spatial photometric variation in multi-projector displays. In *Proc. IEEE International Workshop Projector-Camera Systems*.
- [Majumder and Welch, 2001] Majumder, A. and Welch, G. (2001). Computer graphics optique : Optical superposition of projected computer graphics. In *Proc. Fifth Immersive Projection Technology Workshop, in conjunction with Seventh Eurographics Workshop Virtual Environments*.
- [Matkovic et al., 2005] Matkovic, K., Psik, T., Wagner, I., and Gracanin, D. (2005). Dynamic texturing of real objects in an augmented reality system. pages 245–248. *Proceeding of IEEE Virtual Reality 2005*.
- [Milgram et al., 1994] Milgram, P., Takemura, H., and et al. (1994). Augmented reality : A class of displays on the reality-virtuality continuum. *SPIE Proceedings : Telemanipulator and Telepresence Technologies*, 2351 :282–292.
- [Myung, 2003] Myung, I. J. (2003). Tutorial on maximum likelihood estimation. *J. Math. Psychol.*, 47(1) :90–100.
- [Nayar et al., 2003] Nayar, S., Peri, H., Grossberg, M., and Belhumeur, P. (2003). A projection system with radiometric compensation for screen imperfections. In *Proc. ICCV Workshop on Projector-Camera Systems (PROCAMS)*.
- [Pagès et al., 2005] Pagès, J., Salvi, J., and Forest., J. (August 2005). Optimized de bruijn patterns for one-shot shape acquisition. In *Image and Vision Computing*, pages 23(8), pp 707–720.

- [Peachey, 1988] Peachey, D. (1988). Antialiasing solid textures. In *ACM , SIGGRAPH, 'Functional Based Modeling' CourseNotes*.
- [Perlin and Hoffert, 1989] Perlin, K. and Hoffert, E. M. (1989). Hypertexture. In *SIGGRAPH '89 : Proceedings of the 16th annual conference on Computer graphics and interactive techniques*, pages 253–262, New York, NY, USA. ACM Press.
- [Pollefeys, 1999] Pollefeys, M. (1999). *Self-calibration and metric 3D reconstruction from uncalibrated image sequences*. PhD thesis, ESAT-PSI, K.U.Leuven.
- [Potsdamer and Altschuler, 1982] Potsdamer, J. and Altschuler, M. (1982). "surface measurement by space-encoded projected beam system", vol. 18. In *in Computer Graphics and Image Processing*,, pages 1–17.
- [Raskar and Beardsley, 2001] Raskar, R. and Beardsley, P. (2001). A self-correcting projector. volume 2, pages II–504–II–508 vol.2.
- [Raskar et al., 1999] Raskar, R., Brown, M. S., Yang, R., Chen, W.-C., Welch, G., Towles, H., Seales, W. B., and Fuchs, H. (1999). Multi-projector displays using camera-based registration. In *IEEE Visualization*, pages 161–168.
- [Raskar et al., 2000] Raskar, R., Low, K.-L., and Welch, G. (2000). Shader lamps : Animating real objects with image-based illumination. (TR00-027).
- [Raskar et al., 1998] Raskar, R., Welch, G., Cutts, M., Lake, A., Stesin, L., and Fuchs, H. (1998). The office of the future : A unified approach to image-based modeling and spatially immersive displays. *Computer Graphics*, 32(Annual Conference Series) :179–188.
- [Ressl, 2003] Ressl, C. (2003). *Geometry, Constraints and Computation of the Trifocal Tensor*. PhD thesis, Institut für Photogrammetrie und Fernerkundung, Germany.
- [Salvi et al., 2002] Salvi, J., Armangu, X., and Batlle, J. (2002). A comparative review of camera calibrating methods with accuracy evaluation. J. Salvi, X. Armangu, J. Batlle, A comparative review of camera calibrating methods with accuracy evaluation, *Pattern Recognition* 35 (7) (2002) 1617–1635.
- [Salvi et al., 2007] Salvi, J., Matabosch, C., Fofi, D., and Forest, J. (2007). A review of recent range image registration methods with accuracy evaluation. volume 25, pages 578–596, Newton, MA, USA.
- [Salvi et al., 2004] Salvi, J., Pagés, J., and Batlle., J. (April 2004). Pattern codification strategies in structured light systems. In *Pattern Recognition*,, pages 37(4), pp 827–849.
- [Segal et al., 1992] Segal, M., Korobkin, C., van Sidenfelt, R., Foran, J., and Haeberli, P. (1992). Fast shadows and lighting effects using texture mapping. (Computer Graphics).
- [Shen et al., 1998] Shen, P.-Y., Wang, W., Wu, C., Quan, L., and Mohr, R. (1998). From fundamental matrix to trifocal tensor. In Melter, R. A., Wu, A. Y., and Latecki, L. J., editors, *Proc.*

- SPIE Vol. 3454, p. 340-347, Vision Geometry VII, Robert A. Melter ; Angela Y. Wu ; Longin J. Latecki ; Eds., volume 3454 of Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference, pages 340–347.*
- [Sukthankar et al., 2001] Sukthankar, R., Stockton, R., and Mullin, M. (2001). Smarter presentations : Exploiting homography in camera-projector systems. In *Proceedings of the International Conference on Computer Vision*.
- [Tardif et al., 2000] Tardif, J.-P., Roy, S., and Trudeau, M. (2000). Multi-projectors for arbitrary surfaces without explicit calibration nor reconstruction.
- [Torr and Murray, 1997] Torr, P. and Murray, D. (1997). The development and comparison of robust methods for estimating the fundamental matrix. In *International Journal of Computer Vision*, pages 271-300, v 24, n3.
- [Trucco et al., 1998] Trucco, E., Fischer, R., Fitzgibbon, A., and Naidu, D. (1998). Calibration, data consistency and model acquisition with laser stripers. In *International Journal of Computer Integrated Manufacturing 11*, 293-310.
- [Tsai, 1987] Tsai, R. Y. (Aug. 1987). A versatile camera calibration technique for highaccuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, vol. 3, pp. 323– 344.
- [Underkoffler et al., 1999] Underkoffler, J., Chak, D., Santos, G. S., Laszlo, J., and Ishii, H. (1999). The luminous room : some of it, anyway. In *SIGGRAPH '99 : ACM SIGGRAPH 99 Conference abstracts and applications*, page 167, New York, NY, USA. ACM Press.
- [Verlinden et al., 2003] Verlinden, J., de Smit, A., Peeters, A., and van Gelderen, M. (2003). Development of a flexible augmented prototyping system. in *Journal of WSCG*, vol 11(3), 2003, pp. 496-503.
- [Watson, 1986] Watson, A. B. (1986). Temporal sensitivity. In *Handbook of Perception and Human Performance*, K. Boff, L. Kaufman and J. Thomas editors, New-York.
- [Webb and Jaines, 2005] Webb, S. and Jaines, C. (2005). The dome : A portable multi-projector visualization system for digital artifacts. pages 245–248. *Proceeding of IEEE Virtual Reality 2005*.
- [Weng et al., 1992] Weng, J., Cohen, P., and Herniou, M. (1992). Camera calibration with distortion models and accuracy evaluation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(10) :965–980.
- [Zhang, 1996] Zhang, Z. (1996). Determining the epipolar geometry and its uncertainty : A review.
- [Zhang, 1999] Zhang, Z. (1999). Flexible camera calibration by viewing a plane from unknown orientations. In *ICCV99*, pages 666–673.

[Zollmann and Bimber, 2007] Zollmann, S. and Bimber, O. (2007). Imperceptible calibration for radiometric compensation. pages 61–64.

Table des figures

1.1	Polychromie du portail de la cathédrale d'Amiens	2
2.1	Continuum Réel Virtuel [Milgram et al., 1994]	7
2.2	Image obtenue sans bump mapping, image de bump mapping, image obtenue avec le bump mapping (GNU Free Documentation License)	9
2.3	Effet de crènelage, compensé ou non	10
2.4	Exemple de sphère hypertexturée	11
2.5	Texture représentée par des texels	11
2.6	Différentes types d'écaille suivant l'équation [Fleischer et al., 1995]	12
2.7	modèle <i>pinhole</i> de la caméra	12
2.8	Correction de la distorsion due au positionnement de l'écran [Sukthankar et al., 2001]	13
2.9	Image droite englobée dans la zone de projection [Raskar and Beardsley, 2001]	14
2.10	Correction d'intensité de la lumière	16
2.11	Image non corrigée, Image photométrique uniforme. [Majumder et al., 2003]	16
2.12	De gauche à droite : [Nayar et al., 2003],[Grundhöfer and Bimber, 2008]	16
2.13	Focus et défocus de deux projecteurs [Majumder and Welch, 2001]	17
2.14	Système proposé par verlinden	18

2.15	Système de réalité augmentée [Matkovic et al., 2005]	18
2.16	Système DOME	19
2.17	Outils développés par l'équipe de Raskar ,[Raskar et al., 2000] [Bandyopadhyay et al., 2001].	19
2.18	Table dynamique développé par Ashdow [Ashdown and Robinson, 2003]	20
2.19	de gauche à droite : détection des bords grâce au maxima de la transformée de Hough, projection sur l'ardoise [Borkowski et al., 2003]	21
2.20	Projection en temps réel sur une surface complexe	22
2.21	Avant et Après, remove shadow	22
2.22	Système developpé par Jones <i>et al.</i>	22
2.23	Différentes images obtenues suivant le modèle de perspective	23
2.24	Détermination de points caractéristiques sur le temple de Taj Mahal [Raskar and Beardsley, 2001]	24
3.1	Repères employés pour étalonner une caméra, dessinée par Jydidier en octobre 2005.	28
3.2	Repère du monde et mire	32
3.3	Modélisation 3D de la position et de l'orientation des caméras, ainsi que des mires	36
3.4	stéréovision	41
3.5	Mire d'étalonnage du projecteur	43
3.6	Différentes étapes d'étalonnage du projecteur	43
3.7	Identification des lignes et de la première colonne sur la mire	44
3.8	Identification des points de correspondance de la mire projetée pour l'étalonnage du projecteur	44
3.9	Géométrie épipolaire pour deux vues	45

3.10	Estimation de la matrice fondamentale [Armangue and Salvi, 2003]	48
3.11	Géométrie épipolaire pour deux vues	50
3.12	Géométrie épipolaire pour trois vues	51
3.13	Étalonnage du système : Les caméras sont représentées en rouge, le projecteur en vert, les mires pour étalonner les caméras en couleur, et les mires du projecteur sont en noir. Les distances sont exprimées en mm	54
3.14	Transfert de point des deux premières images vers la troisième image.	56
4.1	Classification des techniques de lumière structurée proposé par [Salvi et al., 2004]	60
4.2	Création de la mire binaire, étape par étape	62
4.3	Mire binaire pseudo-aléatoire	62
4.4	Méthode pour extraire les points de correspondance de la mire binaire	63
4.5	A gauche Point de correspondance centrée sur une cellule, à droite cellule centrée sur la ligne épipolaire	64
4.6	Points de correspondance entre les caméras	65
4.7	Chemin Hamiltonien et Eulérien sur un graphe de Bruijn	66
4.8	Mire couleur de Pages	69
4.9	Mire couleur de Pages projetée sur un modèle réel : une ligne est extraite de cette image	69
4.10	Courbes RGB de la ligne analysée pour extraire les pics	70
4.11	Courbes RGB de la ligne analysée pour extraire les pics sur une plage de valeurs affinée	70
4.12	Identification des points d'intérêt sur un objet texturé par la mire de Pages	72
4.13	Reconstruction 3D des points de correspondance de la mire de Pages	72
4.14	Recalage	74

4.15 a. Reconstruction 3D de la surface b. 2 splines sont estimées c. Reconstruction et interpolation de la surface	75
4.16 Classification des techniques de recalage fin et grossier proposé par [Salvi et al., 2007]	78
4.17 Recalage de notre reconstruction 3D incomplète sur le modèle de référence . . .	80
4.18 Synthèse de vue sans texture	81
4.19 Synthèse de vue	82
5.1 Reconstruction 3D de la mire	85
5.2 Synthèse de vue de la mire employée pour l'étalonnage des caméras	86
5.3 Synthèse de vue de la mire employée pour étalonner le projecteur et comparaison entre la synthèse de vue et l'image originale	87
5.4 Fusion entre l'image et son estimation après correction de la distorsion et correction de la distorsion sur la synthèse de vue de la mire papier	89
5.5 De droite à gauche : image de la caméra, image synthétisée, <i>texture mapping</i> sur objet réel	90
5.6 Condamnation de la porte de garage de la maquette par une texture brique . . .	90
5.7 Estimation de l'erreur due au bruit gaussien	91
5.8 Pourcentage de points valides après un seuil sur la corrélation	92
5.9 Erreur de rétroprojection entre les point réels (cercle bleu), et les point 3D projetés sur le plan image du projecteur (croix rouge)	93
5.10 Points de correspondance déterminés avec les seuils de corrélation sur les deux premières images, points de correspondance déterminés avec les seuils de corrélation et de distance épipolaire sur les deux dernières images	94
5.11 Reconstruction 3D et normales	95
5.12 Points de correspondance après le seuil de corrélation, puis de distance épipolaire	96

5.13	Reconstruction 3D de la surface et estimation des normales	96
5.14	Gauche à droite : Deux plans avec la mire, reconstruction et identification des plans, zone de correspondance entre les points 2D et 3D, projection de la texture	97
5.15	Triangulation de Delaunay des points de correspondance, avec ou sans couleur de facette	98
5.16	Comparaison des reconstructions 3D : à gauche la mire binaire, à droite la mire couleur	99
5.17	Deux modèles de forme identique, un seul étant texturé	99
5.18	Modèle réel et texture virtuelle	100
5.19	Agrandissement sur les différentes parties du modèle	101
5.20	Diagramme d'erreur en mm de recalage	102
5.21	2 vues de garfiled : de gauche à droite : synthèse de vue, objet réel et texture virtuelle, haut à bas : vue 3 et 10	103
5.22	Modèle réel et texture virtuel, segmentation synthèse de vue	104
5.23	Surimpression : la texture de l'objet est modifiée pour mettre en valeur certaines parties	105
5.24	Quatre objets dépourvus de texture	106
5.25	tortue	107
5.26	Jardinier	108
5.27	nain	109
5.28	Vue de côté	110
A.1	Principe de la lumière structurée imperceptible	131
A.2	Motif projeté	132
A.3	Compression d'image [Cotting et al., 2004]	134

A.4	Image de profondeur par mire binaire [Cotting et al., 2004]	134
A.5	Tracking du bras [Cotting et al., 2004]	134

Liste des tableaux

3.1	Estimation de distance épipolaire en mm	50
5.1	Estimation de distance entre des points 3D employés lors de l'étalonnage	86
5.2	Erreur en pixel	88
5.3	Erreur de rétroprojection suite à la correction de la distorsion	89
5.4	Erreurs de correspondance pour deux modèles	92
5.5	Robustesse de la méthode au bruit gaussien	94
5.6	Estimation de l'erreur de recalage en mm sur le modèle du chat	102
5.7	Estimation de l'erreur en mm de recalage	111
B.1	synthese de vue	136
B.2	suite synthese de vue	137
B.3	Texture virtuel	138
B.4	suite Texture virtuel	139

Annexe A

Vision active

A.1 Mire uniforme

La vision active pour la reconstruction 3D à partir de multiples images 2D nécessite le concours d'un motif projeté sur la scène observée. Malheureusement, cette projection empêche de pouvoir obtenir les informations de texture et de couleur que l'on désire plaquer sur une reconstruction 3D. Le but de ce paragraphe est d'exposer le principe de notre système trinoculaire qui utilise la technique de la lumière structurée imperceptible pour contourner ce problème, ainsi que les premiers résultats obtenues, via des hypothèses couramment admises sur les matrices d'étalonnage des capteurs.

Notre capteur trifocal se compose de deux caméras et d'un projecteur (figure A.1). La source lumineuse projette, à fréquence élevée, un motif structuré (diapositive ou image) suivi de son motif complémentaire, de sorte que la résultante forme une lumière uniforme (figure A.2).

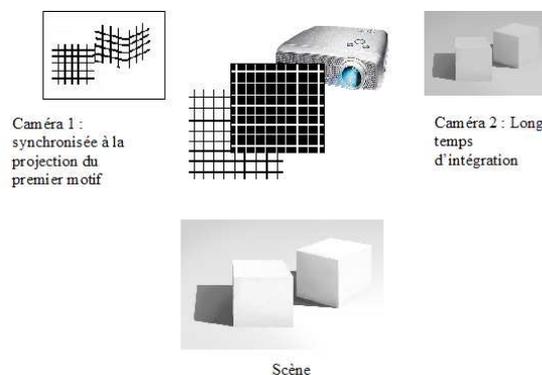


Figure A.1 – Principe de la lumière structurée imperceptible

L'une des caméras, que nous appellerons dans l'annexe caméra 1, est synchronisée à la projection du motif et dispose d'un temps d'intégration rapide ; la seconde caméra (caméra 2) est configurée avec un temps d'intégration plus long. En conséquence, la caméra 1 voit la scène avec le motif projeté alors que la caméra 2 voit la scène sous une illumination uniforme et permet donc les traitements de la vision monoculaire classique.

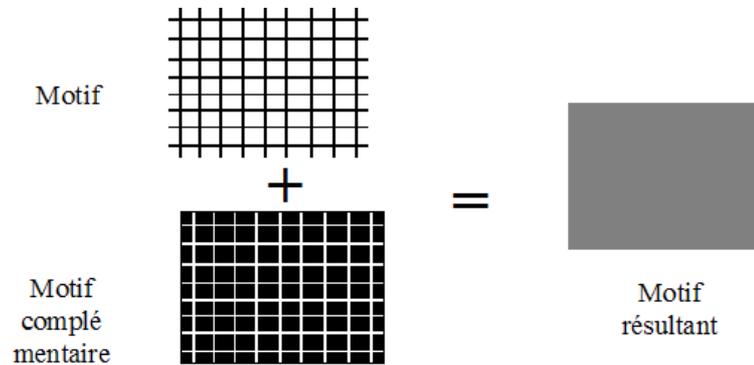


Figure A.2 – Motif projeté

Le principe psychophysique de la lumière structurée imperceptible repose sur l'effet visuel de papillotement lors de la projection d'un motif et de son complémentaire : en dessous de la FCF (Fréquence Critique de Fusion, fixée à 75 Hz selon [Watson, 1986]), l'oeil perçoit l'alternance entre les deux images. Au-delà de cette fréquence critique, l'oeil ne voit plus que la fusion du motif et de son complémentaire, soit une lumière uniforme.

A.2 Mire intégrée

Outre la projection de la mire et de son opposé, Cotting [Cotting et al., 2004] propose une nouvelle méthode de lumière structurée imperceptible en intégrant une mire binaire arbitraire dans une image couleur ordinaire.

La mire n'est visible que pour une caméra synchronisée et exposée pour un court instant. Il exploite la mire de modulation du micro-miroir utilisé par le projecteur pour générer l'intensité pour chaque pixel et couleur.

Ce micro miroir, à la base de la technologie DLP (Digital Light Processing) est composé d'un large CMOS IC, appelé DMD, dont la surface est constellée d'une dense population de miroirs $16\mu m \times 16\mu m$.

Chacun de ces micro miroir peut être électrostatiquement déplacé vers une de ces deux positions possibles ($10\mu m.s$). L'intensité du pixel relié à un micro-miroir est relié au temps que ce micro-miroir est orienté vers l'écran. Il estime la séquence miroir pour les 256 valeurs de rouge et les regroupe en 3 classes : blanche, noire et indéfinie.

Cotting tire partie du fort contraste de ratio (on peut passer d'un pixel illuminé à sombre en un bref instant) pour projeter une mire imperceptible dans l'espace de temps de $125ms$, très inférieur à la résolution temporelle de l'oeil humain.

D'autres approches existent, comme l'emploi de kits de découverte DMD qui donnent accès au chip DLP et leur contrôleur. Un projecteur infra-rouge est une autre solution pour créer des mires imperceptibles, mais ce type de projecteur n'est destiné que pour des scènes statiques.

Cotting estime les séquences de miroir *flip* qui varient dans le temps et l'espace et mesure ainsi de multiples séquences d'un jeu représentatif de pixel sur la surface d'affichage. Comme les trois canaux sont indépendants, seul le canal rouge est abordé. Les 256 valeurs peuvent être divisées en 3 classes : noir, blanc et non fiables. Les miroirs sont actifs ou non dans ce laps de temps. Cotting étudie le comportement des pixels durant la période de $0.4 ms$ à $0.5 ms$, appelé *binary image exposure period*, ou BIEP, pour permettre la classification des séquences de miroir, ensuite il procède à une classification basé sur la caméra pour déterminer le temps d'exposition et mesure la valeurs des pixels pour toutes les intensités possibles.

On peut éclairer la scène en considérant que le canal rouge ne fonctionne que dans la classe blanche. Inversement si on veut que la caméra ne perçoive aucune lumière, on n'utilise le canal rouge que dans la classe sombre. L'algorithme d'intégration introduit le fait que la luminosité de chaque pixel rouge peut maintenant provenir approximativement pour moitié de l'image original, ainsi les images projetés montreront une résolution réduite.

La figure A.3 montre la compression de tous les canaux.



Figure A.3 – Compression d’image [Cotting et al., 2004]

La figure A.4 présente le résultat de la LSI : on peut extraire une image de profondeur de la scène.

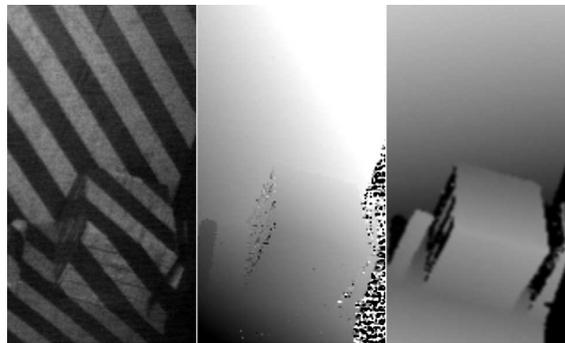


Figure A.4 – Image de profondeur par mire binaire [Cotting et al., 2004]

Une autre application de ce système est un système de *tracking* A.5, la zone rouge représentant le bras et la zone bleue l’ombre du bras.

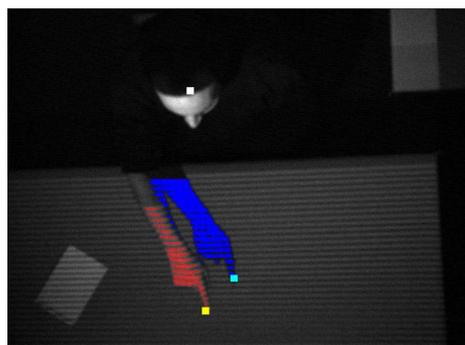
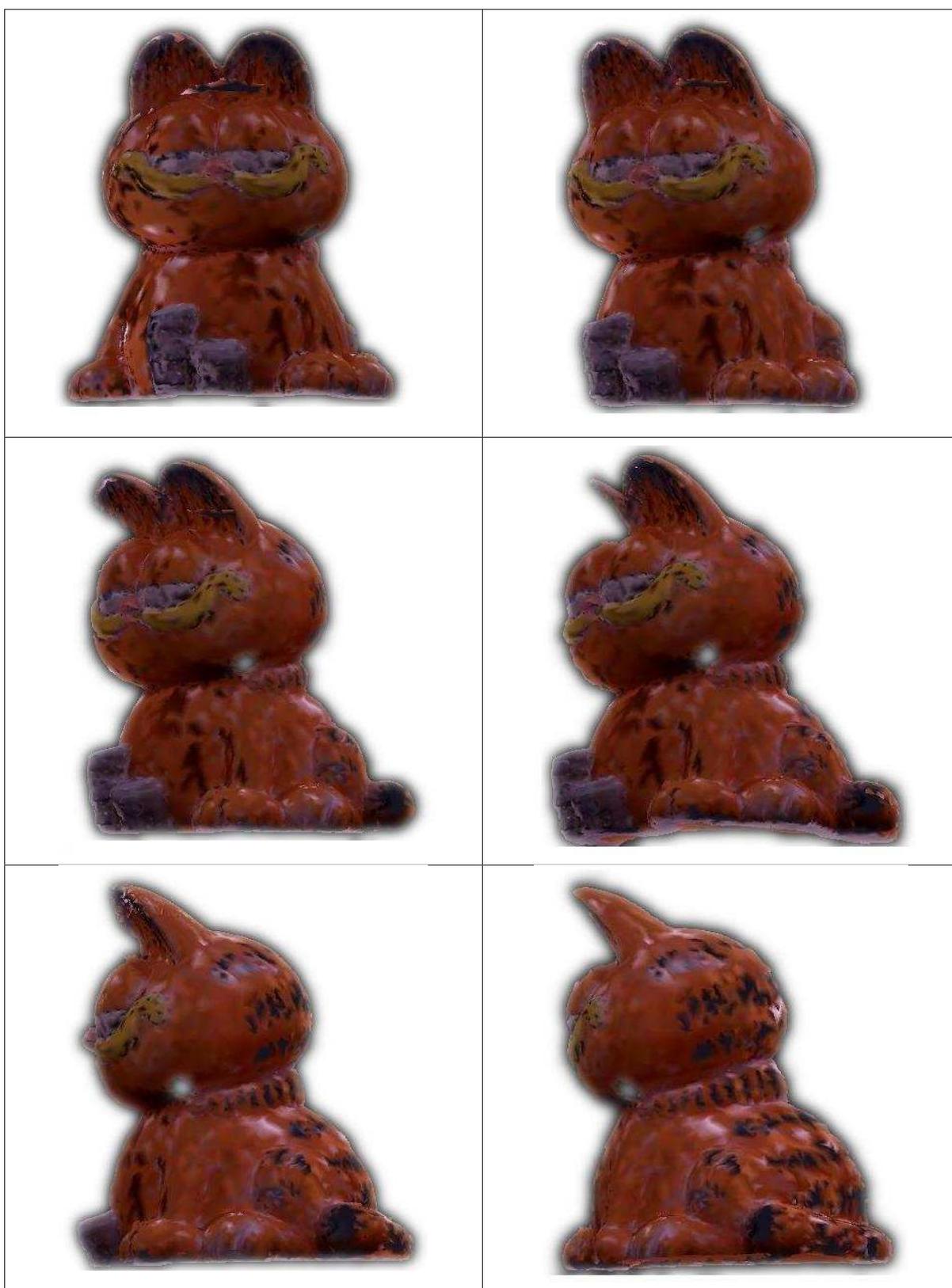


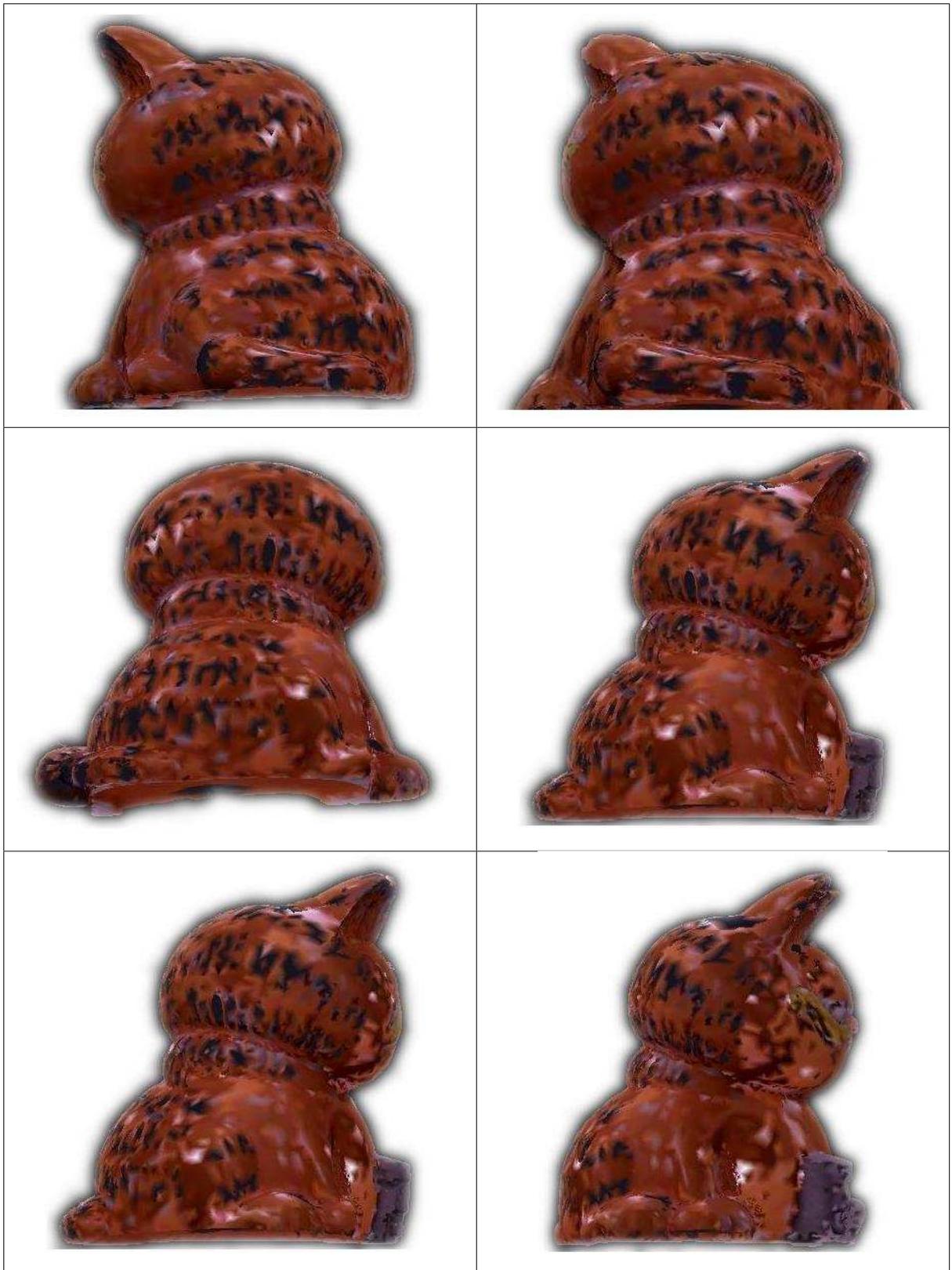
Figure A.5 – Tracking du bras [Cotting et al., 2004]

Annexe B

Mosaïque des synthèses de vues et objet réel texturé pour le modèle Garfield



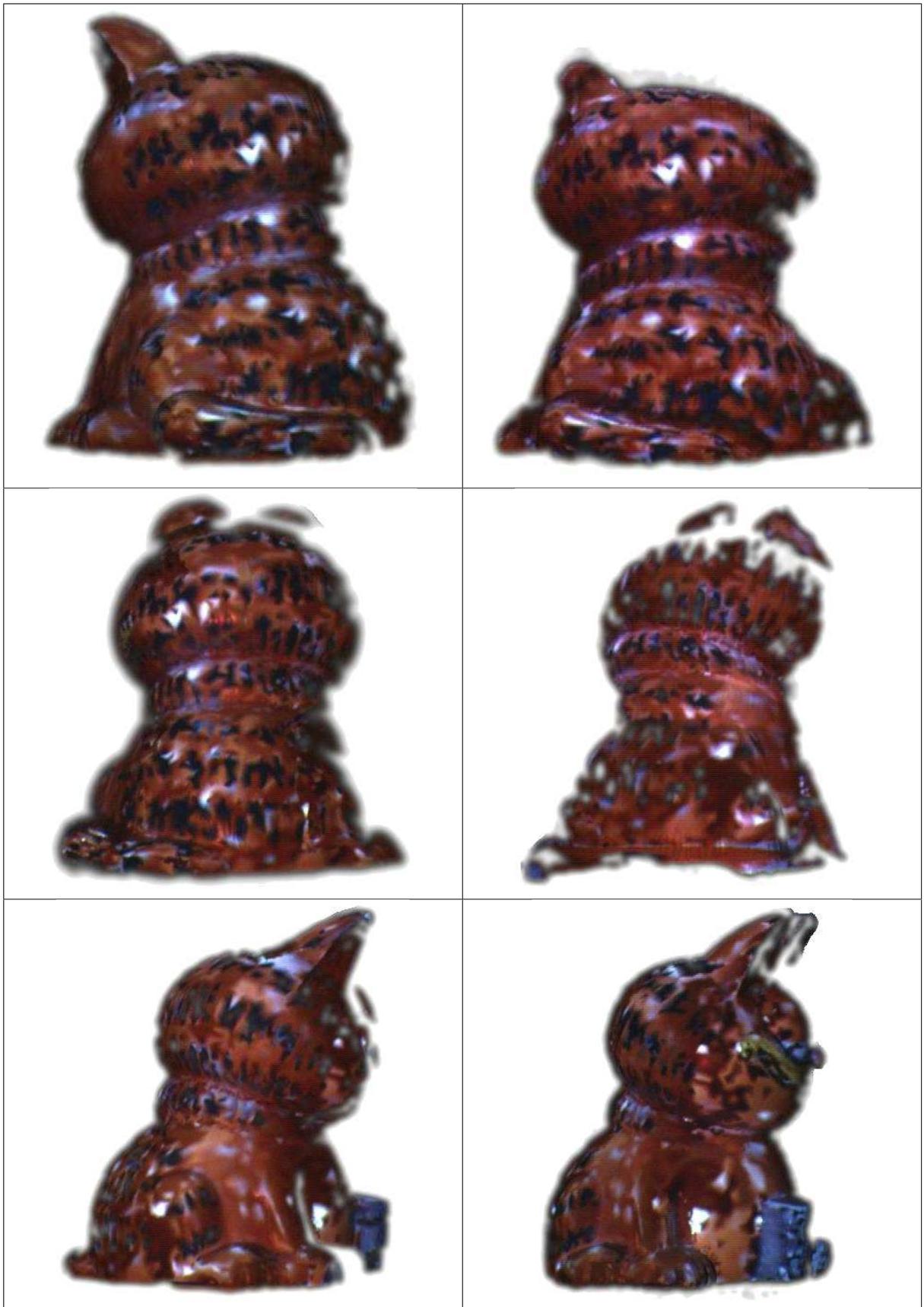
TAB. B.1 – synthese de vue



TAB. B.2 – suite synthese de vue



TAB. B.3 – Texture virtuel



TAB. B.4 – suite Texture virtuel