

N° d'ordre:

THÈSE

présentée à

**L'UNIVERSITÉ PARIS-IX DAUPHINE
U.F.R. MATHÉMATIQUES DE LA DÉCISION**

pour obtenir le titre de

DOCTEUR EN SCIENCES

Spécialité

MATHÉMATIQUES APPLIQUÉES

par

Jean-Philippe TAREL

Sujet de la thèse :

**ESTIMATION GÉOMÉTRIQUE
ET APPARIEMENT EN
MODÉLISATION AUTOMATIQUE**

Soutenue le 13 juin 1996 devant le jury composé de :

MM.	Rachid	DERICHE	Président
	Michel	DHOME	Rapporteurs
	Roger	MOHR	
	André	GAGALOWICZ	Examineurs
	Jean-Michel	MOREL	

*À Karine
ma soeur
unique,*

Remerciements

Je tiens à exprimer ma gratitude et mes plus vifs remerciements aux membres du jury :

- à Rachid Deriche qui m’a fait l’honneur d’être le président de ma thèse, et dont j’apprécie tout particulièrement les travaux,
- à Roger Mohr et Michel Dhome pour m’avoir fait l’honneur d’accepter d’être les rapporteurs, et dont les avis et conseils m’ont été très enrichissants,
- à André Gagalowicz, mon directeur de recherche, qui m’a accueilli dans son laboratoire à l’INRIA, et dont j’apprécie le soutien constant qui m’a permis d’aboutir dans mon travail,
- et enfin à Jean-Michel Morel pour m’avoir fait l’honneur de bien vouloir faire partie du jury.

Ma profonde gratitude va à Nozha Boujemaa qui, grâce à de nombreuses discussions, échanges d’idées et relectures, a beaucoup influencé mon travail. Je voudrais également remercier tout particulièrement Jean-Marc Vézien pour son aide amicale sans faille et ses judicieux conseils qui m’ont permis de faire mes premiers pas en stéréovision. En outre, je remercie Vannary Meas-Yedid avec laquelle la collaboration a été fructueuse et amicale.

Mes remerciements ne seraient pas complets s’ils ne se tournaient pas vers tous ceux qui animent les locaux du laboratoire et plus généralement de l’INRIA, en contribuant au climat d’amitié qui y règne, particulièrement : René Boswinkel, Sonia Bouzidi, Maïté Carrive, Henri Chajmowicz, Giovanni Cornetti, Richard Lengagne, Chahab Nastar, Arghyro Paouri, Xavier Provot, Sabine Randriamasy, Véronique Serfaty, Latifa Taleb.

Enfin, je tiens à exprimer ma reconnaissance aux personnes très chères à mon coeur qui, par leur affection, m’ont soutenu durant ces longues années de travail.

Notations

\mathbb{R} ensemble des nombres réels. Les lettres minuscules sont généralement réservées aux nombres.

i, j, k et l indices.

J matrice de taille $k \times l$. Les lettres majuscules sont généralement réservées aux vecteurs et matrices.

J^t transposée de la matrice J .

J^{-1} inverse de la matrice J .

J^* notation sous la forme d'un vecteur colonne de la matrice J .

X (ou X' , X'' , Y et Y') point dans \mathbb{R}^p . Si X est un point 3D, il a pour coordonnées $\begin{pmatrix} x \\ y \\ z \end{pmatrix}$.

Si X un points 2D, il a pour coordonnées $\begin{pmatrix} x \\ y \end{pmatrix}$.

U point dans une image de coordonnées $\begin{pmatrix} u \\ v \end{pmatrix}$.

\mathcal{J} matrice notée en coordonnées homogènes qui se décompose par ligne en $\mathcal{J} = \begin{pmatrix} J_1 \\ \vdots \\ J_k \end{pmatrix}$. De

manière générale, les objets notés en coordonnées homogènes sont caligraphiés.

$\mathcal{X} = \begin{pmatrix} X \\ 1 \end{pmatrix}$ notation en coordonnées homogènes du point X .

$\mathcal{V} = \begin{pmatrix} V \\ 0 \end{pmatrix}$ notation en coordonnées homogènes du vecteur V .

∇ opérateur différentiel 'nabla'.

X_{ii} avec $i = 1, \dots, n$ ensemble de n points indicés par i .

N_{ii} avec $i = 1, \dots, n$ ensemble de n normales indicées par i .

B_i bruit gaussien centré d'écart type σ .

$F()$, $G()$ transformations géométriques de \mathbb{R}^p dans \mathbb{R}^p : translation, rotation, déplacement rigide, similitude, étirement, affinité, homographie.

T translation de description minimale en 3D: $T(t_x, t_y, t_z)$.

R (ou W , Z) matrice d'une rotation, de description minimale (p_1, p_2, p_3) , où les p_i sont par exemple les paramètres d'Euler-Rodrigues.

(R, T) déplacement rigide constitué d'une rotation de matrice R et d'une translation T .

O tenseur d'ordre 3, o forme triadique associée, ω forme trilineaire polaire associée à o .

s, s' changements d'échelle.

D, D', Λ, Λ' matrices diagonales.

Q matrice de covariance croisée. \mathcal{Q} notation de la matrice de covariance croisée en coordonnées homogènes.

\mathcal{H} représente la matrice en coordonnées homogènes d'une homographie, dont les lignes sont notées H_i .

\mathcal{L} représente la matrice en coordonnées homogènes d'une transformation perspective, dont les lignes sont notées L_i .

P plan.

$d(X, X')$ distance euclidienne entre les points X et X' .

I_d matrice identité.

\odot produit contracté de deux tenseurs.

$\|X\|$ norme vectorielle ou matricielle de X .

A aire, G barycentre, I matrice d'inertie. \mathcal{I} matrice d'inertie en coordonnées homogènes. Dans le cas d'un ensemble $\{X_i\}_i$ de n points $A = n$, $G = \frac{1}{n} \sum_{i=1}^n X_i = \overline{X}_i$, et $I = \frac{1}{n} \sum_{i=1}^n X_i X_i^t = \overline{X_i X_i^t}$.

R, T, u_0, v_0, a_u, a_v paramètres de la calibration d'une caméra, ou une décomposition partielle de la matrice perspective \mathcal{L} . (u_0, v_0) est le centre de l'image en pixels. (a_u, a_v) sont les inverses des tailles horizontale et verticale d'un pixel en mm .

\equiv égalité vectorielle ou matricielle à un facteur réel près.

M^k torseurs des moments géométriques d'ordre k .

$M_{k,l}$ moment d'ordre (k, l) en 2D.

$var(x)$ variance de la variable aléatoire x .

$I(u, v)$ intensité de l'image au pixel de coordonnée (u, v) .

u_{ik}, w_{ik} degrés d'appartenance de la i -ème donnée à la k -ème classe.

\dot{X}, \ddot{X} dérivées première et seconde par rapport au temps t .

$\frac{\partial f(u,v)}{\partial u}$ dérivée partielle de la fonction f par rapport à u .

\mathfrak{R} repère.

C origine d'un repère \mathfrak{R} .

Δx petit écart sur x .

n nombre d'éléments.

c nombre de classes.

m facteur flou.

θ angle.

e, e' fonctions d'erreur.

$\epsilon = \pm 1$.

φ coefficient de confiance.

Φ vecteur force.

Introduction

En physique, poser un problème consiste souvent à le formuler comme l'estimation des paramètres qui minimisent la fonction d'énergie du système considéré. Par analogie, la vision par ordinateur suscite des problèmes qui se ramènent souvent à une estimation par minimisation couplée à un choix combinatoire.

Tout d'abord, il est certes difficile de définir un problème de vision par ordinateur, sans être trop réducteur. De par ses applications diverses -le domaine médical, la réalité augmentée, la cartographie, la télédétection et la robotique- elle présente des buts qui paraissent fort éloignés. En outre, on ne connaît pas encore, par exemple, toutes les applications possibles de ce domaine encore jeune. L'ensemble des problèmes que la vision par ordinateur tente de résoudre n'est donc pas encore délimité.

Pourtant, sans prétendre répondre à cette difficile question de la problématique de la vision par ordinateur, et suite à notre travail effectué dans le cadre du développement d'un système de vision stéréoscopique qui produit une modélisation géométrique comme photométrique, nous avons constaté que, souvent, la difficulté fondamentale de ce type de problème résidait dans la nécessité d'estimer le modèle sous-jacent aux données en même temps que de déterminer le sous-ensemble des données sur laquelle l'estimation doit être réalisée. Ce sous-ensemble peut être directement une partie des données brutes, ou une partie des combinaisons possibles entre les données.

Ainsi, une fois un problème de vision caractérisé, il se réduit souvent à la question suivante : comment découpler l'estimation des paramètres de la modélisation sous-jacente, de la détermination de la partie des données sur laquelle elle s'effectue ?

Modélisation polyédrique par stéréoscopie

Notre travail s'inscrit dans le cadre du développement d'un système de vision qui cherche à obtenir une modélisation de ce qui est observé des caméras CCD par extraction d'informations de plus en plus abstraites. On considère généralement trois niveaux d'abstraction des informations : les caractéristiques des images, la description géométrique, et enfin l'interprétation sémantique.

Nous nous attachons dans ce document à obtenir une description géométrique de la scène. L'accroissement grandissant des simulations informatiques nécessite des outils de modélisation de l'environnement qui soient robustes et efficaces, pour leur fournir des données réalistes. En particulier, les aspects géométriques de la modélisation à partir d'images restent difficiles à obtenir.

Par rapport à d'autres approches [BB82, Aya88, Gro93], notre système de modélisation se révèle original, puisqu'il tente d'analyser son environnement tant sur ses aspects géométriques que photométriques. Ce système a donc pour objectif d'extraire suffisamment d'informations géométriques sur la forme et la position des objets observés pour poursuivre l'analyse par l'obtention d'informations de nature plus photométrique, telles les caractéristiques des sources lumineuses et l'albédo des surfaces.

Comme les informations de nature photométrique s'estiment sur des éléments de surface, et que l'environnement de travail est une scène d'intérieur de type polyédrique, les concepteurs initiaux de ce système ont choisi d'extraire la géométrie 3D sous la forme de facettes d'albédo constant à partir d'un couple d'images stéréoscopique. Dans ces conditions, la projection d'une facette est une région dont l'intensité varie linéairement dans l'image. L'analyse commence donc par extraire des régions dans les images, pour pouvoir ensuite reconstruire directement des primitives 3D de type surfacique. Ceci présente l'avantage d'extraire, en plus de la géométrie de la facette, sa texture (voir figure 0.1).

De la reconstruction en facettes texturées, l'analyse progresse encore en abstraction, vers une description géométrique. Lorsqu'une modélisation des objets observés est disponible, la position de ces derniers est alors retrouvée dans la scène à partir de la reconstruction. Dans le cas contraire, cette dernière est modélisée comme le recouvrement d'un ensemble de prototypes géométriques, des parallélépipèdes par exemple, pour aboutir à un modèle CAO complété d'informations photométriques.

Ce système d'analyse de scènes polyédriques, à partir d'images stéréoscopiques est schématisé par la figure 0.2. Il procède en sept étapes [JMN⁺96]:

1. Une calibration géométrique (chapitre 2) et radiométrique (annexe A) des caméras qui est assez précise pour ne pas ajouter de difficultés aux tâches postérieures.
2. Une segmentation de chaque image en régions qui ont une intensité constante [RG91, BSG94] ou variant linéairement [AMG93] (section 5.4).
3. Une mise en correspondance 2D des régions segmentées entre les images gauche et droite qui se fonde sur l'utilisation des nombreux attributs propres à chaque région [SVC89, Ran92].
4. Une reconstruction, selon des critères géométriques, photométriques ou mixte [Véz95] (chapitre 3), des facettes 3D issues des paires de régions que l'on suppose images d'une même facette.
5. Un recalage 3D des reconstructions prises à différents points de vue, ou, si la géométrie d'un objet vu est connue (construite à partir d'un autre capteur ou interactivement comme dans l'annexe B) un recalage entre son modèle géométrique et la reconstruction de la scène (chapitre 4). Si la précision du résultat n'est pas suffisante, cette étape est complétée par un raffinement du recalage 3D à partir des images (annexe C).
6. Une modélisation géométrique par un ensemble de prototypes déformables, lorsque suffisamment de données géométriques ont été accumulées par la reconstruction sur un même objet (chapitre 5).

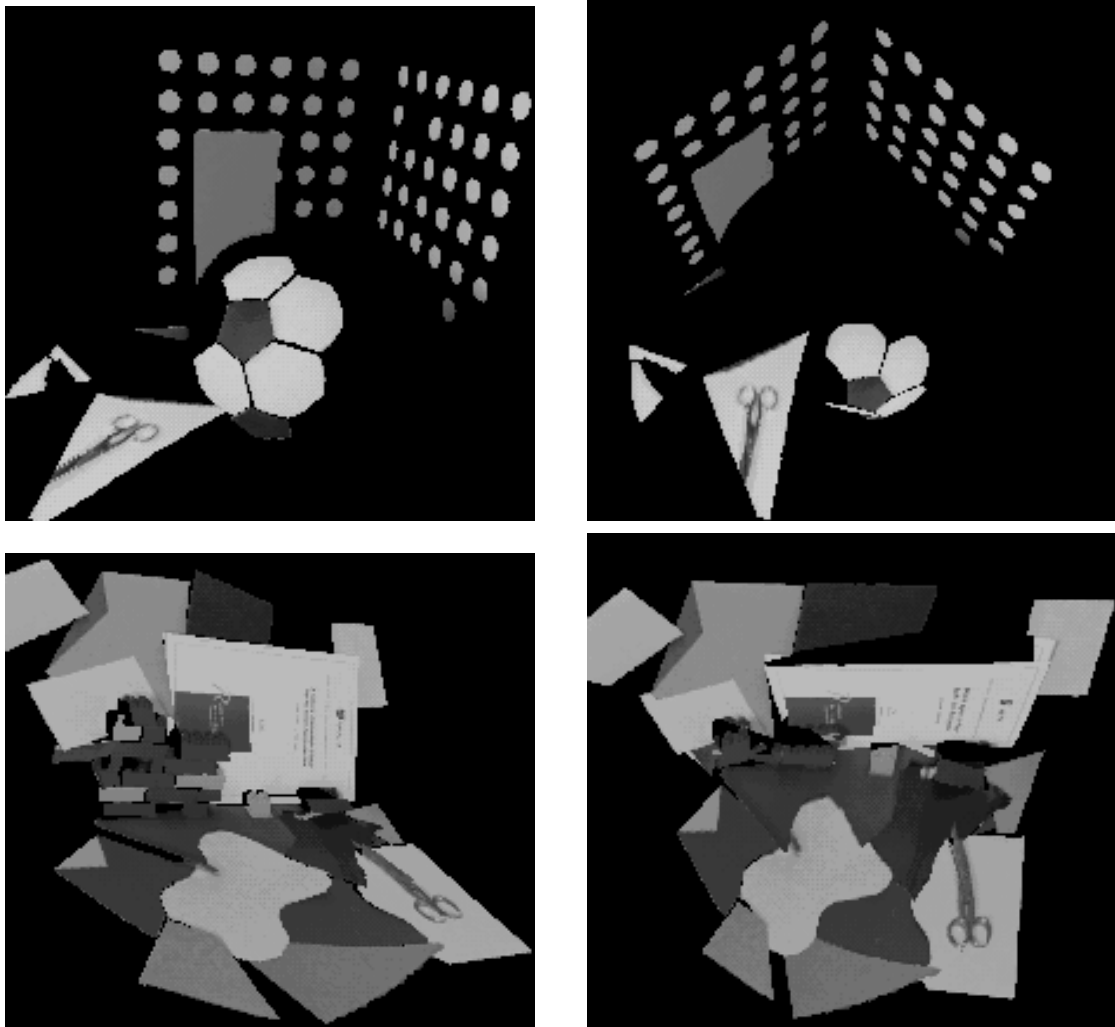


FIG. 0.1 - *Reconstruction en facettes texturées de deux scènes.*

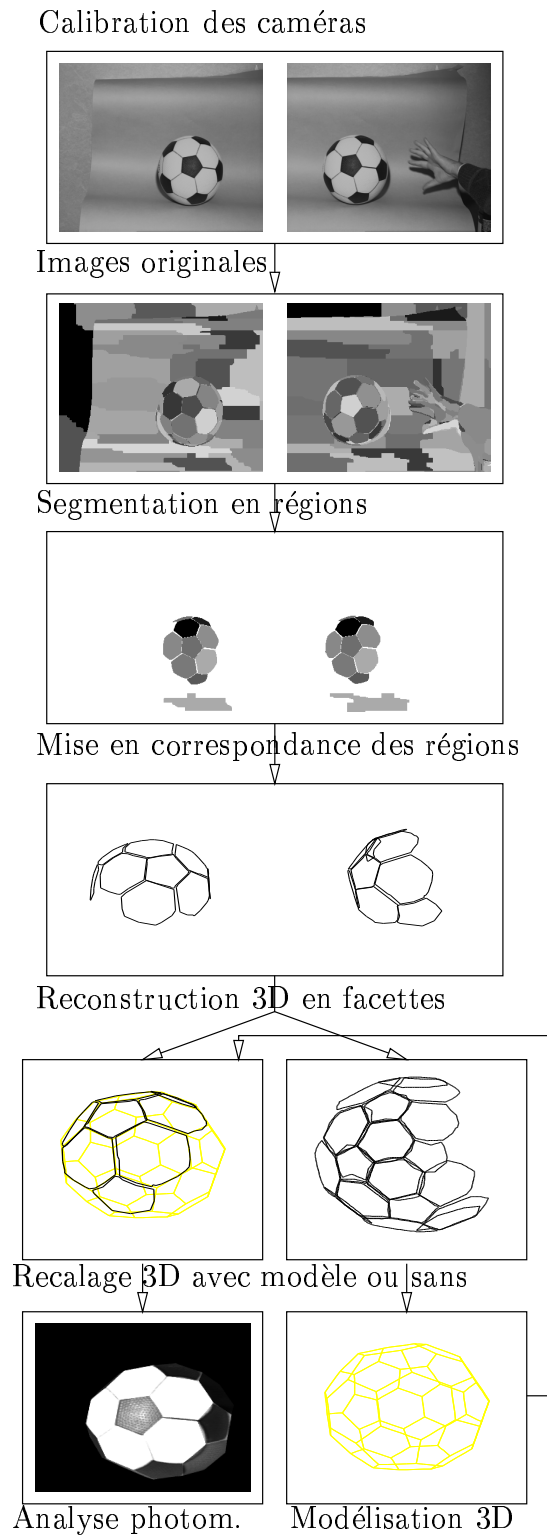


FIG. 0.2 - *Le processus d'analyse stéréoscopique.*

7. Une analyse photométrique des facettes pour retrouver leur albédo et la position de la source lumineuse équivalente [SAMLG93].

Au cours de chaque étape du processus d'analyse que nous avons traité, nous nous sommes attachés à séparer au minimum l'aspect géométrique et photométrique, dans l'esprit qui a animé le développement de ce système, car la complémentarité de ces deux aspects est souvent utile pour simplifier les problèmes, et aboutir à des solutions plus robustes ou plus précises.

Pour chaque algorithme développé, nous avons aussi essayé d'exploiter au mieux les images que peut nous fournir la synthèse d'images à partir de modélisations géométriques et photométriques assez réalistes. En effet, tout en sachant que cela ne permet que des tests préliminaires, l'utilisation de ces images synthétiques aident à l'analyse globale des problèmes qui mélangent difficultés géométriques, imprécision de discrétisation, et complexité photométrique.

Estimation géométrique et appariement

Dans le cadre de la constitution du processus d'analyse stéréoscopique précédent, nous expliquons la méthodologie que nous avons suivi et qui consiste à analyser chaque problème comme une estimation couplée à un appariement, dans le but de le poser de façon à séparer au maximum les deux types de déterminations, ou à se ramener à un problème d'estimation seul.

Nous faisons donc l'hypothèse de base que non seulement la modélisation du problème existe, mais qu'elle nous est connue. Mais il n'est pas toujours nécessaire de connaître la modélisation sous-jacente pour faire de l'estimation [Vio95]. Néanmoins, cette connaissance permet de gagner en rapidité et en précision, car l'algorithme exploite alors directement les contraintes dues à la modélisation cadre.

Puisque nous cherchons à produire un modèle géométrique de la scène, les données que nous traitons sont principalement de nature géométrique bi- ou tri-dimensionnelles. Nous parlerons donc essentiellement d'estimation géométrique. Dès lors, les paramètres qui sont estimés consistent alors essentiellement en des transformations géométriques sur les données.

Dans les cas les plus simples où données et modèle sont complets, il existe de nombreuses techniques récapitulées dans le cadre particulier de notre application.

Toutefois, l'estimation doit être souvent réalisée sur une partie des données seulement car elles présentent des données aberrantes. Ainsi, les méthodes d'estimation robustes sont très utiles comme elles sont conçues pour estimer et séparer les données pertinentes des aberrantes dans un même temps. Pourtant, les deux qualités nécessaires à ce type d'algorithme -la précision du résultat et la robustesse face à la diversité des données- ne sont pas forcément compatibles. En effet, un gain en précision se fait souvent au détriment de la robustesse de l'algorithme face à la diversité des jeux de données.

Cette conséquence devient d'autant plus critique quand l'estimation est couplée à une recherche d'appariements. Nous avons donc préféré supposer que les données peuvent seulement être incomplètes, mais qu'il n'y avait pas ou peu de données parasites. On fait donc l'hypothèse que le modèle sous-jacent aux données est complet. Ceci permet de résoudre de

façon relativement robuste des problèmes qui, auquel cas, seraient très difficiles. De plus, la classification floue nous a fourni un bon cadre pour traiter des situations où plusieurs estimations géométriques et recherches d'appariements de même nature sont à faire en un même temps.

Ainsi, tout en veillant à la robustesse et à la précision des algorithmes produits, nous illustrons notre approche par l'exemple, en traitant, du plus simple au plus compliqué, une succession de problèmes : dans le premier, l'appariement y est fait de manière préliminaire; dans le deuxième, l'appariement n'est pas nécessaire pour faire l'estimation; enfin, bien que couplé avec l'estimation, le problème se ramène à une minimisation. Dans la dernière catégorie, nous traitons, plus particulièrement, les problèmes qui nécessitent d'estimer un ou plusieurs modèles sous-jacents de même nature géométrique, soit sur une partie des données à sélectionner, soit sur une partition de ces données à constituer.

Plan du document

La méthodologie que nous avons mise en place pour traiter de l'estimation d'une transformation géométrique entre deux ensembles de points qui sont appariés, non-appariés, ou appariables, est exposée dans le premier chapitre, ainsi que des rappels sur les outils mathématiques qui y sont associés. Nous avons aussi évalué dans le *Chapitre 1: couplage entre l'estimation géométrique et l'appariement* la précision qu'il est possible d'obtenir dans un certain nombre de cas.

Ensuite nous présentons l'application de cette méthodologie à quatre des sept étapes du système de vision stéréoscopique fondé sur les régions :

- *Chapitre 2: calibration géométrique de la caméra.* L'utilisation d'une mire permet par sa structure bien spécifique de découpler l'estimation de la transformation perspective de l'appariement entre les cibles disposées à sa surface et leurs images. Nous avons choisi comme forme de cible une ellipse, car l'estimation de ses caractéristiques géométriques peut être réalisée de façon très précise sur l'ensemble de la cible sans nécessiter d'appariement locaux, malgré la discrétisation de l'image.
- *Chapitre 3: Reconstruction géométrique/photométrique de facettes 3D.* Nous avons cherché à nous placer dans les conditions stéréoscopiques qui permettent d'éviter l'appariement des contours des deux régions projections de la facette, pour estimer son support. Cela nous a conduits à travailler dans l'espace dit des disparités qui présente l'avantage de permettre d'effectuer la reconstruction géométrique comme photométrique de façon simple, robuste et précise.
- *Chapitre 4: Recalage rigide 3D.* Le recalage de deux objets 3D qui présentent une partie commune nécessite de coupler estimation et appariement. Ce recalage rigide, pour des ensembles de facettes 3D, peut se faire de façon combinatoire en travaillant, non pas sur les données brutes, mais sur l'ensemble des paires de facettes. La combinaison de cette méthode de recherche combinatoire sur les appariements avec un algorithme de recalage itératif où l'appariement est fait avec le plus proche voisin, permet d'allier la

robustesse de la détermination de la partie des données concernées avec la précision du résultat obtenu.

- *Chapitre 5: Modélisation par des prototypes déformables.* Comme le découplage des estimations s'avère impossible, notre approche de résolution combine itérativement une étape d'appariement vers le plus proche voisin, et une étape d'estimation de la transformation géométrique optimale qui réalise l'appariement. Comme l'ensemble de la reconstruction doit être modélisé par des prototypes géométriques, nous proposons d'unifier les méthodes de recalage avec les approches de type classification pour résoudre ce problème de façon précise et de manière assez robuste. Plus généralement, nous montrons que la sous-classe des problèmes d'estimation géométrique couplée à la segmentation de l'ensemble des données est un problème bien posé.

Chapitre 1

Couplage entre l'estimation et l'appariement

1.1 Introduction

Comment estimer une transformation géométrique entre la totalité ou une partie de deux ensembles des données? Cela est une question de base de la vision par ordinateur [BSG90, HJL⁺89].

De nombreux problèmes de vision par ordinateur peuvent être vus comme des cas particuliers. Par exemple, faire la calibration d'une caméra consiste à estimer la transformation géométrique entre l'espace 3D de la scène et l'espace 2D de l'image. La reconstruction d'une primitive en est un autre exemple, puisque cela revient à calculer la transformation 3D que doit subir une primitive canonique pour que ses deux projections se confondent avec son image stéréoscopique. Beaucoup d'algorithmes de vision résolvent cette question de base, pour un type particulier de transformation géométrique (rigide, affine, homographique, orthographique ou perspective), et pour un type de données géométriques (points, segments, courbes, faces ou maillages). Sans omettre les contraintes, propres à chaque problème, qui nécessitent un traitement particulier, il est souvent intéressant d'analyser un problème de vision par ordinateur sous ses aspects.

Expliciter cette question et analyser un ensemble de problèmes dans ce cadre permet de reconnaître un certain nombre de difficultés communes aux problèmes de la vision par ordinateur. C'est en s'appuyant sur les différentes facettes révélées par les problèmes de calibration, segmentation, reconstruction, recalage, et modélisation, qu'il nous a été possible de reconnaître cette problématique commune. En effet, toutes les facettes du problème n'apparaissent pas forcément sur un seul sujet, même lorsque l'on travaille sur le recalage, qui illustre pourtant assez bien la problématique ici posée. Car, une fois dépassées, les points délicats de l'estimation robuste et précise, la difficulté fondamentale à surmonter est l'obtention couplée d'un appariement et d'une estimation géométrique.

Lorsque l'on dispose de l'appariement, comme une donnée a priori, ou comme le résultat d'une étape préliminaire, la problématique se réduit à un problème d'estimation dont traite la théorie mathématique de la régression de modèle. Au contraire, on ne dispose pas ou peu d'un cadre mathématique adéquat pour répondre à la question lorsque la correspondance

est inconnue. Peut-on alors se passer de la mise en correspondance? Certes, la réponse est positive pour des transformations géométriques particulières. De fait, dans le cas général, il apparaît très vite que l'on ne peut se passer de la mise en correspondance. Qu'est-il possible de faire face au problème d'estimation géométrique lorsqu'il est couplé à l'obtention de l'appariement?

Nous présentons dans la première section, notre approche pour tenter de répondre à cette question. Dans les sections suivantes, les outils disponibles pour traiter ces différents types de problèmes sont récapitulés. Dans ces sections, la présentation se veut la plus unifiée et la plus générale possible. Néanmoins, nous avons dû nous restreindre à la primitive la plus utilisée qui est le point. Dans la section 1.3, les algorithmes pour estimer les transformations les plus usuelles lorsque l'appariement est connu entre les deux ensembles de primitives sont décrits. De même, la section 1.4 suivante réunit un ensemble de types de transformations, où il existe un algorithme global pour la calculer même si la mise en correspondance est inconnue. Enfin, dans la dernière section, nous décrivons quelques outils généraux que nous allons utiliser tout au long des prochains chapitres, pour faire face à certains des problèmes où l'estimation est inséparable de l'appariement couplés mais qui sont néanmoins bien posés.

1.2 Séparer l'estimation et l'appariement

1.2.1 Différents niveaux de difficultés

En vision par ordinateur, il est souvent possible de résoudre une même tâche en la posant mathématiquement comme un problème d'estimation seul, si l'appariement est connu, si l'estimation peut se faire sans les appariements, ou bien si l'appariement se formule comme une estimation. On peut donc constituer quatre catégories de problèmes :

- **Problèmes d'estimation pure :** la correspondance est connue a priori, ou obtenue par un traitement préliminaire. Même si la solution n'est pas toujours explicite, puisque le problème se formule comme la minimisation d'une erreur entre les données du premier ensemble transformé et ceux du deuxième, il est alors toujours possible de concevoir un algorithme itératif pour estimer la transformation optimale. Un exemple typique d'estimation avec appariement est le problème de la calibration de caméra avec une mire, qui est traité dans le chapitre suivant.
- **Problèmes d'estimation sans appariement :** la correspondance est inconnue, mais si la transformation entre les deux ensembles est linéaire, en s'appuyant sur les moments géométriques de chacun d'eux, il est possible de retrouver explicitement cette transformation. La reconstruction d'une facette à partir de ses projections en régions est un exemple de problème que nous avons pu faire rentrer, sous certaines conditions étudiées dans le chapitre 3, dans cette catégorie.
- **Problèmes d'appariement formulés comme une estimation :** l'absence de données aberrantes donne la possibilité de formuler simplement ces problèmes comme la minimisation d'une fonction d'erreur non analytique. Car, à la différence du cas 1, l'appariement est ici implicitement donné par une fonction de type plus proche voisin.

Une sous-classe intéressante est l'ensemble des problèmes où toutes les données doivent être recouvertes par différentes instances déformables d'un prototype géométrique. Le chapitre 5 présente des exemples de ce type de problèmes, comme la segmentation en régions des images et le recouvrement de données 3D incomplètes ou occultées.

- **Problèmes où l'estimation est couplée à l'appariement** : une partie seulement des données doit être transformée. Il faut donc déterminer cette partie en même temps que la transformation est estimée. Constituer l'appariement est alors obligatoire pour arriver à un résultat. Pour les trois classes précédentes, il est supposé que la ou les transformations s'appliquent sur la totalité des deux ensembles de points. Par contre, dans la dernière catégorie entre tous les autres problèmes, en particulier ceux qui présentent des données aberrantes et incomplètes par rapport au modèle sous-jacent. Un exemple de ce type de problème est traité dans le chapitre 4 sur le recalage 3D rigide.

1.2.2 Comment posons-nous un problème?

Nous résumons à présent la démarche que nous avons suivie pour traiter les différents sujets exposés dans chacun des chapitres suivants. Notre angle d'attaque consiste à déterminer ce qui relève de l'estimation et de l'appariement, pour analyser combien ces deux aspects du problème sont couplés.

Intrinsèquement, résoudre un problème couplé est plus difficile que faire une simple estimation. Pour chaque problème de vision, on a donc intérêt à poser le problème de façon à se ramener à l'une des deux premières catégories présentées dans la section précédente. En effet, nous disposons alors d'un cadre mathématique général pour y faire face : théorie de l'estimation, des invariants, algèbre multilinéaire. De plus, dans ce cadre, des outils puissants pour faire de l'estimation robuste et précise existent. Nous avons donc intérêt à les utiliser.

Un certain nombre d'algorithmes pour faire de l'estimation géométrique sont rappelés ou référencés dans les sections suivantes. Plus précisément, la section 1.3 récapitule les algorithmes qui sont disponibles actuellement pour estimer différents types usuels de transformations géométriques entre deux ensembles de points appariés. Ces estimations se basent sur la minimisation de la distance euclidienne entre les points appariés (équation (1.1)).

La présence de données parasites induit toujours des difficultés importantes. Par exemple, sans appariement, mais en l'absence de points aberrants, il est parfois possible de retrouver la transformation entre deux ensembles de données. La section 1.4 récapitule, dans le cas de deux ensembles de points, les types de transformations que l'on peut retrouver ainsi que les algorithmes associés. Ce type d'estimation n'est pas encore bien formalisé mathématiquement, puisque l'on n'a plus de critère d'erreur à minimiser (voir équation 1.16), mais nous montrons que, pour certaines transformations, les solutions obtenues coïncident avec celles où l'appariement est connu. De fait, elles minimisent aussi l'erreur aux moindres carrés.

Les deux premières catégories de problèmes supposent donc l'absence de données aberrantes, ce qui leur permet, en utilisant l'ensemble des données, d'être robuste aux bruits. Il est important de noter la différence entre une donnée bruitée et une donnée parasite. Une donnée bruitée contient toujours un peu d'information utile à l'estimateur, que l'on peut tenter d'extraire grâce à un modèle sous-jacent de ce bruit. Par contre une donnée parasite,

est par définition, beaucoup plus dure à modéliser. Notre approche consiste donc implicitement à nous placer dans des situations où le minimum de données doivent être considérées comme aberrantes.

Plus généralement, l'absence de données aberrantes indique que le modèle sous-jacent du problème est suffisamment complet pour expliquer les données. Dans ce cas, on peut toujours le poser globalement comme un problème d'estimation. Mais, à la différence de la première catégorie de problèmes, il n'y a pas de formule analytique de l'erreur à minimiser. Cette formule est seulement implicite. L'équation (1.27), de la section 1.6 est un exemple de ce type de fonction, car elle fait intervenir, pour chaque point, le plus proche voisin contenu dans le modèle. Pour autant, le fait que l'erreur ne soit pas explicite empêche en rien la résolution numérique qui est alors itérative. Il est alors possible de procéder par un algorithme qui, à chaque itération, réalise deux étapes, l'une d'appariement, et l'autre d'estimation. Ce découpage à l'avantage de permettre d'utiliser directement les algorithmes d'estimation avec appariement connu.

Que l'appariement soit fait avec le plus proche voisin implique une sensibilité de la méthode de résolution aux conditions initiales. Si l'initialisation n'est pas facile, il est donc préférable de poser un problème de façon à avoir une expression de l'erreur explicite plutôt que implicite. Néanmoins, un des avantages de ces algorithmes est qu'ils résolvent des cas non traitables par ceux qui font l'estimation globalement sans avoir l'appariement. En effet, comme ils recherchent un appariement local, à chaque étape, tout en gardant une contrainte globale, ils ne sont gênés ni par les occultations, ni par les données manquantes, puisque ces parties sont automatiquement interpolées. Nous introduirons, dans la section 1.6, quelques algorithmes utilisés pour résoudre ce type de problèmes, appliqués au cas où les données sont constituées de points.

La dernière catégorie de problèmes est la plus délicate à traiter. Géométriquement, il existe une caractérisation des points parasites : c'est un point positionné loin de tous les autres. Un cadre mathématique utile est alors l'estimation robuste avec par exemple les M-estimateurs [HRRS86] ou les filtres médian [SW92]. Ces méthodes ont tendance à être sensibles à l'initialisation et sont donc souvent fragiles face à la diversité des jeux de données. De plus, apparier est une tâche de nature combinatoire, qui se relève souvent de complexité algorithmique prohibitive pour des ensembles de grande taille. L'addition des difficultés de l'estimation robuste avec celles de l'appariement font que cette catégorie reste souvent très problématique.

1.3 Estimation avec appariement

Dans cette partie, nous disposons de deux ensembles de n points $\{X_i\}_{1,\dots,n}$ et $\{X'_i\}_{1,\dots,n}$, où $i = 1, \dots, n$ qui sont appariés pour un même indice i . L'ensemble $\{X'_i\}_{1,\dots,n}$ est le transformé par F de l'ensemble $\{X_i\}_{1,\dots,n}$. Ceci se traduit par la relation suivante :

$$X'_i = F(X_i) + B_i$$

où B_i est un bruit supposé gaussien centré. Dans le cadre des problèmes de vision, les

transformations F les plus utiles sont :

- une translation $F(X) = X + T$,
- un déplacement rigide, soit une rotation R combinée avec une translation $F(X) = RX + T$,
- une similitude $F(X) = sRX + T$ où s est un scalaire,
- un étirement, c'est-à-dire la combinaison d'un changement d'échelle sur chaque axe avec un déplacement rigide, soit $F(X) = RDX + T$ où D est une matrice diagonale,
- une affinité $F(X) = LX + T$, qui est une combinaison d'une transformation linéaire L est d'une translation.
- une homographie qui est représentée par \mathcal{H} , une matrice $(p + 1) \times (p + 1)$. L'image X' d'un point X est $\mathcal{X}' \equiv \mathcal{H}\mathcal{X}$, en coordonnées homogènes ($\mathcal{X} = \begin{pmatrix} X \\ 1 \end{pmatrix}$) avec \equiv l'égalité à un facteur près.

On remarque que ces transformations définissent des transformations de plus en plus générales. Elles vont de l'espace \mathbb{R}^p vers \mathbb{R}^p .

Étant donné deux ensembles de points, estimer au mieux la transformation entre les deux ensembles bruités par un bruit gaussien revient à minimiser la fonctionnelle suivante :

$$e(F) = \sum_{i=1}^n d^2(F(X_i), X'_i) \quad (1.1)$$

où $d(X, X')$ est la distance euclidienne entre les points X et X' .

Nous allons maintenant récapituler, pour les différents types de transformations, la méthode la plus générale pour estimer la meilleure transformation minimisant cette fonctionnelle (1.1). Nous nous sommes focalisés sur les méthodes non itératives qui présentent l'avantage évident de ne pas nécessiter de point initial proche de la solution exacte. Pour chaque type de transformation, nous indiquerons une application et éventuellement reporterons le lecteur aux parties du document qui utilisent ces calculs.

1.3.1 Translation

Nous désirons trouver la translation T qui minimise :

$$e(T) = \sum_{i=1}^n \|X_i + T - X'_i\|^2$$

Une condition nécessaire pour que cette fonction e soit minimum est que $\nabla e = 0$. Généralement, cette condition n'est pas suffisante, mais dans le cas présent, elle est nécessaire et

suffisante. De plus, il y a toujours unicité de la solution. Par une simple dérivation, on obtient la meilleure translation, comme la différence des centres de gravité des deux ensembles :

$$T = \frac{1}{n} \sum_{i=1}^n X'_i - \frac{1}{n} \sum_{i=1}^n X_i = \bar{X}'_i - \bar{X}_i$$

Nous avons rappelé ce cas très simple car l'estimation de la translation peut souvent être séparée lors de celle de la transformation F . C'est le cas du déplacement rigide, de la similitude, de l'étirement et de l'affinité. Alors, pour ces transformations, on réécrit la fonctionnelle (1.1) en introduisant G , défini par $F(X) = G(X) + T$, où T est la translation :

$$e(G, T) = \sum_{i=1}^n \|G(X_i) + T - X'_i\|^2 \quad (1.2)$$

Pour les transformations qui nous intéressent, G est linéaire, en conséquence, on déduit par dérivation par rapport à T que :

$$T = \frac{1}{n} \sum_{i=1}^n X'_i - \frac{1}{n} \sum_{i=1}^n G(X_i) = \bar{X}'_i - G\bar{X}_i \quad (1.3)$$

Ainsi, il est avantageux de travailler dans les repères centrés sur les barycentres de chaque ensemble, soit $Y'_i = X'_i - \bar{X}'_i$ et $Y_i = X_i - \bar{X}_i$. Ainsi, dans ce chapitre, nous noterons par Y les coordonnées dans le repère centré. Dans ce dernier, le problème de minimisation s'écrit alors simplement, puisque la translation n'apparaît plus :

$$e(G) = \sum_{i=1}^n \|G(Y_i) - Y'_i\|^2 \quad (1.4)$$

Par là-même, le problème de moindres carrés de départ (1.2) se réduit à résoudre les deux étapes suivantes :

- trouver $G()$ qui minimise e dans (1.4),
- puis, calculer la translation avec (1.3),

Ce schéma de calcul est appliqué dans chacune des quatre sections qui suivent.

1.3.2 Déplacement rigide/Rotation

Dans cette section, $G()$ est une rotation de matrice R . Nous présentons l'algorithme pour résoudre le problème de moindres carrés, dans le cas d'une rotation, par la méthode de la décomposition en valeurs singulières (SVD: Singular Value Decomposition). Nous avons préféré présenter celui-ci plutôt que ceux fondés sur les quaternions, car il est utilisable dans un espace de dimension quelconque. Pour avoir la démonstration du résultat et de ses propriétés, il faut se reporter à [Kan94]. Néanmoins, en pratique, dans les cas particuliers de la dimension 2 et 3, nous utilisons l'algorithme avec les complexes et les quaternions [FH83] respectivement. L'algorithme avec SVD se résume de la façon suivante :

- Calculer la matrice de covariance croisée des deux ensembles de points, dans les repères centrés sur les barycentres $Q = \frac{1}{n} \sum_{i=1}^n Y'_i Y_i^t = \overline{Y'_i Y_i^t}$ où l'exposant t indique la transposition.

- Trouver la décomposition en valeurs singulière de Q , $Q = ZD'W^t$, où Z et W sont des rotations et D' est diagonale.
- La meilleure rotation est alors $R = ZW^t$.
- La meilleure translation est $T = \overline{X}'_i - R\overline{X}_i$

À condition que l'ensemble $\{X_i\}_{1,\dots,n}$ ne soit pas dégénéré, la solution au problème est unique. Un ensemble est dégénéré, si ses points sont coplanaires, ou inclus dans un sous-espace de \mathbb{R}^p . Dès que l'on ne peut plus ordonner strictement la diagonale de D' , la décomposition en valeurs singulières n'est plus unique. Pourtant, la rotation optimale résultante est identique pour toutes les décompositions de Q .

L'algorithme d'obtention d'un déplacement optimal sert de base aux méthodes de recalage rigide 2D et 3D. Il est en particulier utilisé dans notre étape (section 4.2.3) de raffinement du recalage 3D d'un modèle par rapport à une reconstruction (chapitre 4).

1.3.3 Similitude/Rotation et homothétie

Pour une transformation de type similitude, G est le produit d'un scalaire positif s , avec une rotation R . Puisque $G(Y) = sRY$, la fonction à minimiser est :

$$e(s, R) = \sum_{i=1}^n \|sRY_i - Y'_i\|^2 \quad (1.5)$$

La dérivée partielle de $e(s, R)$ par rapport à s est nécessairement nulle, pour minimiser cette équation. Cette contrainte permet d'obtenir s en fonction de la rotation :

$$s = \frac{\sum_{i=1}^n Y_i^t R Y_i}{\sum_{i=1}^n Y_i^t Y_i} \quad (1.6)$$

Cette formule se réécrit en introduisant la matrice de covariance croisée $Q = \frac{1}{n} \sum_{i=1}^n Y_i' Y_i^t$ et la matrice d'inertie du première ensemble $I = \sum_{i=1}^n Y_i Y_i^t$, de la façon suivante :

$$s = \frac{\text{trace}(RQ^t)}{\text{trace}(I)} \quad (1.7)$$

Puisque s n'est qu'un facteur, on peut appliquer la décomposition en valeurs singulières utilisée dans la section précédente sans modifications. En effet, les rotations Z et W ne sont pas affectées par ce scalaire s , contrairement à D' . La rotation R est donc égale à ZW^t , si $Q = ZD'W^t$. Par substitution, dans (1.7), on vérifie que la valeur de s obtenue est positive, puisque D' est positive tout comme I . Comme dans la section précédente, l'unicité de la solution optimale est assurée pour les ensembles de points non dégénérés, s étant positif.

Le résumé de notre algorithme pour les similitudes est donc le suivant :

- Calculer la matrice de covariance croisée centrée $Q = \overline{Y}'_i Y_i^t$ et la matrice de covariance centrée $I = \overline{Y}_i Y_i^t$.
- Trouver la SVD de Q , $Q = ZD'W^t$

- La meilleure rotation est alors $R = ZW^t$.
- Le meilleur facteur d'échelle est $s = \frac{\text{trace}(D')}{\text{trace}(I)}$.
- La meilleure translation est $T = \overline{X}_i' - sR\overline{X}_i$.

Cet algorithme n'est pas utilisé en vision, bien qu'il permette de concevoir des méthodes de recalage avec échelle inconnue. Nous présentons une méthode de recalage qui s'appuie sur une généralisation de l'algorithme ICP dans le chapitre 5.

1.3.4 Étirement/Rotation et changement d'échelle

Le cas des étirements est plus complexe que le précédent, car il n'a pas de solution générale connue en dimension quelconque. L'ensemble des étirements ne forme en effet pas un groupe de transformations. Néanmoins, nous décrivons la solution du problème dans le cas 2D que nous avons dérivé. En dimension 2, on note θ l'angle de la rotation $R = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$, et $D = \begin{pmatrix} \lambda & 0 \\ 0 & \mu \end{pmatrix}$. Dans le repère des barycentres, la fonction à minimiser est :

$$e(\lambda, \mu, \theta) = \sum_{i=1}^n (\lambda x_i \cos \theta - \mu y_i \sin \theta - x_i')^2 + (\lambda x_i \sin \theta + \mu y_i \cos \theta - y_i')^2 \quad (1.8)$$

où λ et μ sont des scalaires positifs. À partir de la dérivée partielle de $e(\lambda, \mu, \theta)$ par rapport à λ et μ , nous déduisons la valeur des facteurs d'échelle en fonction de l'angle de rotation θ :

$$\lambda = \frac{\overline{x_i' x_i} \cos \theta + \overline{y_i' x_i} \sin \theta}{\overline{x_i^2}} \quad (1.9)$$

$$\mu = \frac{\overline{y_i' y_i} \cos \theta - \overline{x_i' y_i} \sin \theta}{\overline{y_i^2}} \quad (1.10)$$

Par substitution de λ et μ dans la dérivée partielle de $e(\lambda, \mu, \theta)$ par rapport à θ , il vient que θ doit vérifier l'égalité suivante :

$$b \cos 2\theta - a \sin 2\theta = 0$$

avec $a = \frac{1}{2} \left(\frac{\overline{x_i' x_i}^2}{\overline{x_i^2}} - \frac{\overline{y_i' x_i}^2}{\overline{x_i^2}} - \frac{\overline{x_i' y_i}^2}{\overline{y_i^2}} + \frac{\overline{y_i' y_i}^2}{\overline{y_i^2}} \right)$ et $b = -\frac{\overline{x_i' x_i} \overline{y_i' x_i}}{\overline{x_i^2}} + \frac{\overline{x_i' y_i} \overline{y_i' y_i}}{\overline{y_i^2}}$. En conséquence, nous proposons l'algorithme suivant pour les étirements 2D :

- Calculer la matrice de covariance croisée centrée : $Q = \begin{pmatrix} \overline{x_i' x_i} & \overline{x_i' y_i} \\ \overline{y_i' x_i} & \overline{y_i' y_i} \end{pmatrix}$.
- Le meilleur angle de rotation est $\frac{1}{2} \arctan \frac{b}{a} + k\frac{\pi}{2}$. k est déterminé du fait que λ et μ doivent être positifs dans l'étape suivante.
- Les meilleurs facteurs d'échelle sont donnés par (1.9) et (1.10).

- La meilleure translation est $T = \overline{X}'_i - \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} \lambda & 0 \\ 0 & \mu \end{pmatrix} \overline{X}_i$.

L'existence et l'unicité de la solution optimale est assurée dès que trois points au moins sont non alignés.

Ce type de transformation est utilisé en synthèse d'images, non en vision. Elle a pourtant des applications intéressantes. Ainsi, nous présentons, dans le chapitre 5, une méthode de recalage de rectangles sur des images, qui utilise cet algorithme.

1.3.5 Affinité/Transformation linéaire

Une affinité, après séparation de sa partie translation, est une transformation linéaire, $G(Y) = LY$ avec L une matrice quelconque. Alors, la solution est très simple à obtenir. La solution optimale peut être écrite de façon directe par :

$$L = \overline{Y}'_i Y_i^t \overline{Y}_i Y_i^t{}^{-1} \quad (1.11)$$

Il est remarquable de voir, une nouvelle fois, apparaître la matrice de covariance croisée $Q = \overline{Y}'_i Y_i^t$ qui joue un rôle central dans ces problèmes d'estimation de transformations avec correspondance. Pour qu'une solution existe, la matrice de covariance centré $I = \overline{Y}_i Y_i^t$ doit être inversible. Ainsi, pour les ensembles de points $\{X_i\}_{1,\dots,n}$ non dégénérés, la solution au problème de moindres carrés affine et linéaire existe toujours et elle est unique.

En introduisant la représentation homogène $\mathcal{X} = \begin{pmatrix} X \\ 1 \end{pmatrix}$ d'un point X et la représentation homogène $\mathcal{A} = \begin{pmatrix} L & T \\ 0 & 1 \end{pmatrix}$ d'une transformation affine, l'algorithme peut se résumer à :

- Calculer la matrice de covariance croisée en coordonnées homogènes $\mathcal{Q} = \overline{\mathcal{X}}'_i \mathcal{X}_i^t$ et la matrice de covariance en coordonnées homogènes $\mathcal{I} = \overline{\mathcal{X}}_i \mathcal{X}_i^t$.
- La meilleure transformation est $\mathcal{A} = \mathcal{Q}\mathcal{I}^{-1}$.

Cet algorithme de régression linéaire sur-déterminé est très souvent utilisé en vision par ordinateur, directement dans l'espace physique 3D, l'espace image 2D ou des espaces abstraits. On peut en voir une utilisation dans le chapitre 5.

1.3.6 Homographie

Les coordonnées homogènes introduites dans la section précédente permettent de représenter une homographie sur un espace de dimension p par une matrice \mathcal{H} de dimension $(p+1) \times (p+1)$. La fonctionnelle (1.1) qui permet de minimiser l'erreur entre les coordonnées \mathcal{X}'_{ij} de \mathcal{X}'_i et celles de X_i transformées par \mathcal{H} , s'écrit donc :

$$e(\mathcal{H}) = \sum_{i=1}^n \sum_{j=1}^p \left(\frac{H_j \mathcal{X}_i}{H_{p+1} \mathcal{X}_i} - \mathcal{X}'_{ij} \right)^2 \quad (1.12)$$

où H_1, H_2, \dots, H_{p+1} sont les $p+1$ lignes de l'homographie \mathcal{H} . Puisque cette équation n'est pas linéaire en tous les termes de \mathcal{H} , on pose la fonctionnelle linéaire e' (1.13) dont la minimisation

fournit une solution généralement assez proche de la solution optimale de e (1.12). Dans le cas où la plus grande précision est recherchée, la solution de e' peut servir de point initial pour une méthode de minimisation itérative de e .

$$e'(\mathcal{H}) = \sum_{i=1}^n \sum_{j=1}^p (H_j \mathcal{X}_i - \mathcal{X}'_{ij} H_{p+1} \mathcal{X}_i)^2 \quad (1.13)$$

L'équation (1.13) est un problème de minimisation aux moindres carrés, linéaire. Soit \mathcal{H}^* le vecteur constitué des termes de la matrice \mathcal{H} . Ce vecteur \mathcal{H}^* est supposé de norme 1. Cette condition est ajoutée pour que le problème soit bien posé et que la solution soit unique. Avec les notations précédentes, l'équation (1.13) se réécrit comme un problème de minimisation sous la contrainte $\|\mathcal{H}^*\| = 1$ de :

$$e'(\mathcal{H}^*) = \sum_{i=1}^{np} (J_i \mathcal{H}^*)^2 \quad (1.14)$$

Dans cette formule \mathcal{H}^* est un vecteur à $(p+1)^2$ éléments, et J est une matrice $(p+1)^2 \times np$. La solution optimale est donc le vecteur propre de la matrice $J^t J$ associé à la plus petite valeur propre.

Comme dans le cas affine, la solution existe seulement si l'ensemble $\{X_i\}_{1,\dots,n}$ n'est pas dégénéré, c'est-à-dire s'il n'est pas inclus dans un sous-espace de \mathbb{R}^p . On peut montrer que la condition nécessaire et suffisante est réalisée par l'existence d'au moins deux points non coplanaires avec les autres points [BMV93].

Nous pouvons donc résumer l'algorithme dans le cas homographique comme :

- Calculer la matrice J définie par les équations (1.13) et (1.14).
- L'homographie optimale est donnée par le vecteur propre associé à la plus petite valeur propre de la matrice $J^t J$.

Ce type d'algorithme est souvent utilisé en auto-calibration, ou lorsque l'on cherche à retrouver l'homographie qui permet à une reconstruction projective de devenir euclidienne [BMV93, ZLF96]. On peut voir une autre utilisation de cet algorithme dans le chapitre 5.

1.3.7 Projection perspective

Dans les sections précédentes, les transformations vont de \mathbb{R}^p vers \mathbb{R}^p . Souvent la donnée des points est partielle, et les problèmes de vision combinent alors les transformations précédentes avec une projection. Suivant le problème, cette projection est centrale, orthogonale ou orthographique (orthogonale avec un facteur s' d'échelle inversement égal à la profondeur moyenne de l'objet). Par exemple pour p égal à deux, les projections associées réduites sont :

$$L_{orthogonale} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

et

$$\mathcal{L}_{centrale} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$\mathcal{L}_{orthographique} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & s' \end{pmatrix}$$

en coordonnées homogènes pour les deux dernières.

La transformation perspective est la combinaison d'une affinité avec la projection centrale réduite. Elle va de \mathbb{R}^{p+1} vers \mathbb{R}^p , et se représente par une matrice homogène de taille $(p+1) \times (p+2)$.

En examinant de façon approfondie les formules de la section précédente sur les homographies, on s'aperçoit qu'il est possible d'appliquer les mêmes calculs aux transformations perspectives. On peut reprendre directement l'algorithme de cette section, à la seule différence que cette fois-ci : \mathcal{X} n'est plus un vecteur de taille $p+1$ mais $p+2$, \mathcal{H}^* est un vecteur à $(p+1)(p+2)$ éléments, et J est une matrice $(p+1)(p+2) \times np$.

- Calculer la matrice J définie par les équations (1.13) et (1.14).
- L'homographie optimale est le vecteur propre associé à la plus petite valeur propre de la matrice $J^t J$

Cet algorithme, ou des dérivés, est souvent utilisé pour faire la calibration des caméras, à partir de points 3D sur une mire et de leurs images 2D. Travailler dans des espaces de dimensions différentes rend l'algorithme très sensible aux erreurs. Le prochain chapitre décrit donc en détails comment nous avons abordé ce problème et ces difficultés particulières.

1.3.8 Conclusion

Nous n'avons pas cherché à être exhaustif, mais les cas étudiés recouvrent une partie importante des transformations utilisées en vision. À titre d'exemple, l'algorithme de calcul de la projection orthographique optimale (combinaison d'une affinité et de la projection orthogonale réduite), cas non traité ici, peut facilement être déduit du cas de l'affinité.

En conséquence, dans le cas où l'on connaît la correspondance entre les points, on dispose d'un ensemble d'outils mathématiques assez puissant pour répondre à beaucoup de nos problèmes, même s'il reste encore des questions en suspens comme le calcul de l'étirement optimal au sens des moindres carrés pour une dimension supérieure ou égale à trois. Une autre question non résolue jusqu'à présent est de trouver un algorithme non itératif pour le problème du calcul optimal d'un déplacement rigide combiné avec une projection centrale réduite pour un nombre sur-déterminé de points. Tsai [Tsa87] a proposé une solution à ce problème partiellement équivalent à la calibration d'une caméra lorsque les paramètres intrinsèques sont connus. Mais la solution est approximative et peu satisfaisante du point de vue mathématique.

Ce problème consiste à trouver le déplacement rigide $\mathcal{D} = \begin{pmatrix} R & T \\ 0 & 1 \end{pmatrix}$ en coordonnées homogènes qui minimise la fonction suivante :

$$e(R, T) = \sum_{i=1}^n \sum_{j=1}^p \left(\frac{D_j \mathcal{X}_i}{D_{p+1} \mathcal{X}_i} - \mathcal{X}'_{ij} \right)^2 \quad (1.15)$$

Le calcul optimal et explicite de la combinaison d'une projection centrale ou orthogonale réduite avec les déplacements rigides, les similitudes et les étirements sont des problèmes encore ouverts dont la résolution fournira un outil utile pour résoudre de nombreux problèmes de vision. Il existe pour l'instant que des solutions itératives à ces problèmes dans le cas général de n points. Pour 3 points ou 4 points, différentes méthodes explicites sont récapitulées dans [HLON94, PHYT95].

1.4 Estimation sans appariement

Dans la partie précédente, nous avons fait l'hypothèse que les deux ensembles de points sont appariés. Or cette hypothèse est contraignante, puisqu'il est souvent difficile d'obtenir ces correspondances. Nous nous sommes donc intéressés aux techniques existantes pour calculer la transformation F lorsque l'on ne dispose pas de ces appariements. Il n'est alors plus possible de minimiser le critère (1.1) de la section 1.3. Toutefois, nous allons reformuler au mieux les résultats établis précédemment, pour voir s'ils ne peuvent pas être étendus au cas sans correspondance.

Sinon, on peut se fonder sur les caractéristiques géométriques globales des deux ensembles, en supposant que le deuxième des ensembles est exactement le transformé par F du premier. Alors :

$$X'_i = F(X_i) \quad (1.16)$$

Pour un ensemble de données géométriques, les caractéristiques géométriques globales les plus utilisés sont les tenseurs des moments. Le tenseur d'ordre 1 est le barycentre, celui d'ordre 2, la matrice d'inertie. De façon générale, on définit à l'ordre m le tenseur des moments d'un ensemble de points $\{X_i\}_{1,\dots,n}$ par :

$$M^m(X_i) = \sum_i X_i (\odot X_i)^{m-1} \quad (1.17)$$

Comme nous allons le voir, la combinaison de certaines de ces caractéristiques géométriques permet de retrouver explicitement la transformation entre les deux ensembles.

De manière duale, d'autres combinaisons sont invariantes par différents types de transformations et ces invariants permettent de vérifier a posteriori si effectivement les deux ensembles sont l'image par cette transformation l'un de l'autre. Le cadre mathématique adéquat est alors la théorie des invariants géométriques [Rei93, Moh93].

1.4.1 Translation

En se reportant à la section 1.3.1, puisque la meilleure translation est la différence des centres de gravité des deux ensembles, on s'aperçoit que le calcul de la translation optimale ne nécessite pas la connaissance de l'appariement des points entre les deux ensembles. En conséquence, sans correspondance et lorsque la transformation est affine, l'unique translation optimale est toujours :

$$T = \frac{1}{n} \sum_{i=1}^n X'_i - \frac{1}{n} \sum_{i=1}^n X_i = \bar{X}'_i - \bar{X}_i$$

De façon analogue à la situation avec appariements, la translation peut souvent être découplée de la transformation F (déplacement rigide, similitude, étirement, et affinité). On décompose ces transformations F en $F(X) = G(X) + T$, où T est la translation. Pour chaque point d'indice i , on écrit :

$$X'_i = F(X_i) = G(X_i) + T \quad (1.18)$$

Pour les transformations qui nous intéressent, G est linéaire, en conséquence, en faisant la moyenne sur tous les points de l'équation (1.18), on déduit que :

$$T = \frac{1}{n} \sum_{i=1}^n X'_i - \frac{1}{n} \sum_{i=1}^n G(X_i) = \bar{X}'_i - G\bar{X}_i \quad (1.19)$$

Cette formule (1.19) coïncide avec celle (1.3) de la section 1.3.1. Ainsi, travaillant dans les repères centrés sur les barycentres de chaque ensemble, soit $Y'_i = X'_i - \bar{X}'_i$ et $Y_i = X_i - \bar{X}_i$, la transformation (1.18) entre les deux ensembles s'écrit après soustraction de (1.19) :

$$Y'_i = G(Y_i) \quad (1.20)$$

En conséquence, le problème sans appariement se réduit à résoudre les deux étapes suivantes :

- trouver G qui apparie $\{Y'_i\}_{1,\dots,n}$ avec $\{Y_i\}_{1,\dots,n}$ par (1.20),
- puis, calculer la translation avec (1.19),

Ce schéma de calcul est utilisé dans les sections suivantes.

1.4.2 Déplacement rigide/Rotation

Dans le cas où la transformation est une rotation de matrice R , une technique simple construit un repère propre sur chaque ensemble à partir de leur matrice d'inertie [LD89]. Soit I' et I les matrices d'inertie centrales des deux ensembles $\{X'_i\}_{1,\dots,n}$ et $\{X_i\}_{1,\dots,n}$, ou matrices de covariance des deux ensembles $\{Y'_i\}_{1,\dots,n}$ et $\{Y_i\}_{1,\dots,n}$. Une matrice d'inertie est définie comme :

$$I = \frac{1}{n} \sum_{i=1}^n Y_i Y_i^t$$

La méthode est généralement présentée de la façon suivante, puisque chaque paire de points appariés (Y'_i, Y_i) vérifie :

$$Y'_i = RY_i \quad (1.21)$$

On déduit, en faisant la moyenne des produits de (1.21) par elle-même transposée, que :

$$I' = RIR^t \quad (1.22)$$

Cette équation se résout par diagonalisation des matrices d'inertie. Les matrices diagonales Λ' et Λ sont positives à diagonale décroissante. Alors, on a :

$$\begin{aligned} I &= \Omega\Lambda\Omega^t \\ I' &= \Omega'\Lambda'\Omega'^t \end{aligned}$$

Ω et Ω' définissent un repère propre sur chaque ensemble. Ces repères, une fois appariés, permettent de calculer la rotation entre les deux ensembles de points associés. En effet par substitution de la forme propre de I et I' dans (1.21), on déduit qu'une solution est $R = \Omega'\Omega^t$. Les autres solutions se déduisent par inversions du sens des axes propres.

En établissant le lien avec la solution optimale quand la correspondance est connue, nous allons montrer que la solution avec correspondance appartient à l'ensemble des solutions du problème sans correspondance. Ainsi, la méthode des repères propres minimise aussi $e(R) = \sum_{i=1}^n \|RY_i - Y'_i\|^2$ en présence d'un bruit gaussien.

En effet, dans la section 1.3.5, nous avons montré que la solution générale au problème linéaire est $L = QI^{-1}$. Dans le cas présent, la transformation L est contrainte à être une rotation; en conséquence, on doit avoir l'égalité suivante :

$$L^tL = I_d$$

Décomposons Q en sa forme SVD : $Q = ZD'W^t$, et I en sa forme propre : $I = \Omega\Lambda\Omega^t$. Puisque $L = QI^{-1}$, on a :

$$L^tL = \Omega\Lambda^{-1}\Omega^tW D'^2W^t\Omega\Lambda^{-1}\Omega^t = I_d$$

Soit après simplification :

$$(\Omega^tW)D'^2(\Omega^tW)^t = \Lambda^2$$

Si la diagonale de Λ a des termes tous différents, cette équation implique l'égalité de Ω avec W . En inversant le rôle des ensembles $\{X_i\}_{1,\dots,n}$ et $\{X'_i\}_{1,\dots,n}$, on obtient par un raisonnement analogue que $\Omega' = Z$. En conséquence, la solution optimale $R = ZW^t$ obtenue dans la section 1.3.5 est identique à la solution $R = \Omega'\Omega^t$ obtenue par la méthode des repères propres.

Toutefois, l'algorithme avec correspondance fondé sur la décomposition SVD donne toujours une unique solution, alors que la méthode avec les repères débouche sur plusieurs solutions pas toujours départageables. En effet, deux repères peuvent s'apparier suivant plusieurs configurations dont le nombre dépend de la dimension p de l'espace de travail, et de la longueur des vecteurs de base donnée par les valeurs propres. Ces longueurs sont des invariants par déplacement rigide, elles peuvent donc aussi être utilisées pour constater que les deux ensembles ne sont pas simplement l'image l'un de l'autre par un déplacement. On

montre, puisque le signe de la direction de chaque vecteur propre est inconnu, que 2^p est le nombre de configurations possibles d'appariement de deux bases, et que 2^{p-1} est le nombre de configurations possibles d'appariement de deux bases directes, si toutes les valeurs propres sont différentes.

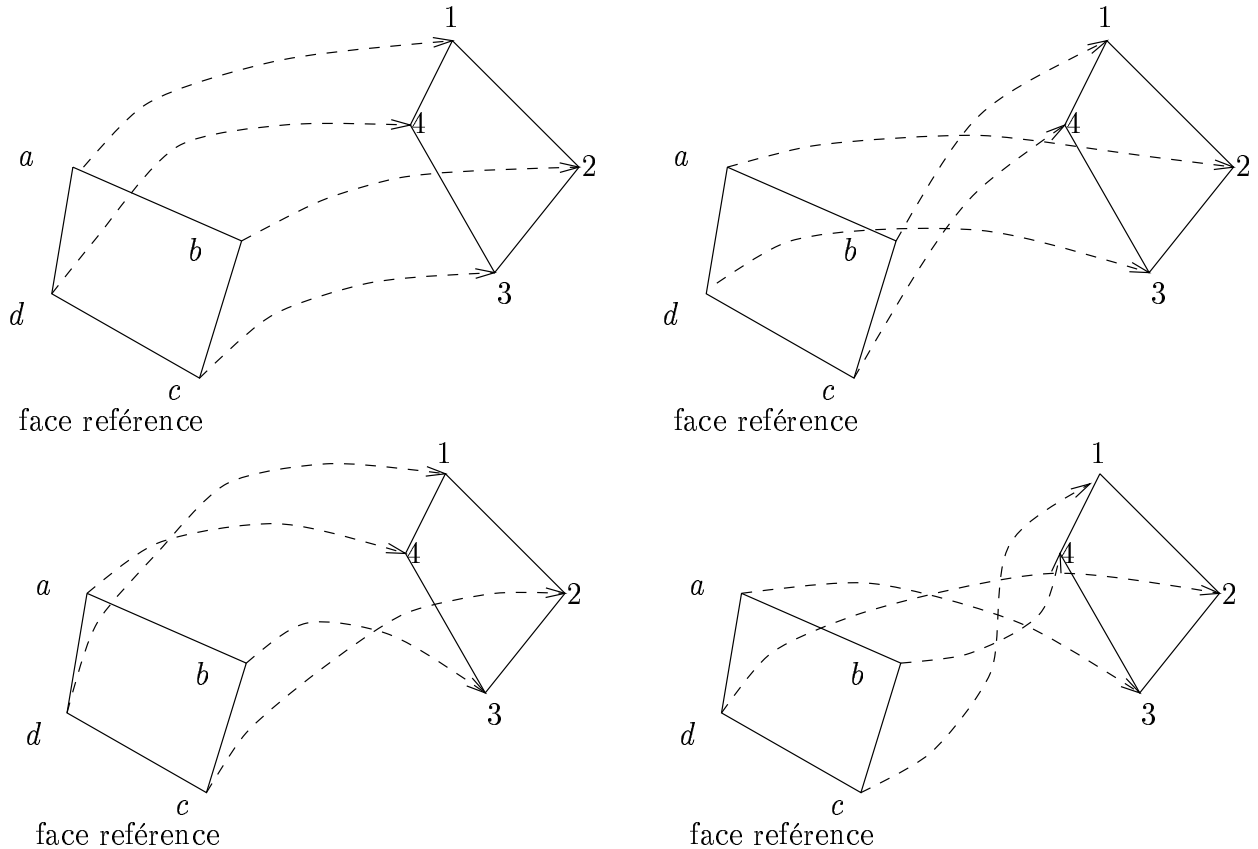


FIG. 1.1 - Mises en correspondances possibles d'une facette 3D réduite à ses axes propres (ellipse visualisée par son rectangle englobant).

Par exemple, deux configurations d'appariement direct des repères propres sont possibles pour $p = 2$, et quatre pour $p = 3$. C'est le cas d'une facette 3D, comme le montre la figure 1.1. Cette propriété est utilisée dans le chapitre 4 sur le recalage 3D (section 4.2.1). Quand les valeurs propres sont distinctes, le nombre de configurations d'appariement reste fini. Ce nombre peut être éventuellement réduit par l'utilisation des moments d'ordre 3. Dans ces situations, en posant C_k la matrice associée à une configuration d'appariement de repères directs, l'ensemble des solutions est :

$$R = \Omega' C_k \Omega^t$$

L'algorithme se résume donc de la façon suivante dans le cas où le nombre de solutions est fini :

- Calculer les matrices d'inertie des deux ensembles de points, dans les repères centrés

sur les barycentres $I = \overline{Y_i Y_i^t}$ et $I' = \overline{Y'_i Y'_i{}^t}$.

- Trouver une décomposition en valeurs propres de $I = \Omega \Lambda \Omega^t$ et de $I' = \Omega' \Lambda' \Omega'^t$, où Ω et Ω' sont des rotations et Λ et Λ' sont à diagonale décroissante. Trouver les 2^{p-1} matrices C_k d'appariement de deux bases directes.
- Les meilleures rotations sont $R = \Omega' C_k \Omega^t$.
- La meilleure translation est $T = \overline{X'_i} - R \overline{X_i}$

Pour associer de façon dénombrable les éléments de la diagonale de Λ à ceux de Λ' , il faut que les moments d'inertie principaux soient différents, sinon l'association s'avère impossible, et la rotation est définie à une rotation près autour de l'axe perpendiculaire aux vecteurs propres dont les valeurs propres sont identiques. Pour traiter ce type de situations, il faut utiliser une information géométrique supplémentaire, comme le tenseur des moments d'ordre 3 de chaque ensemble. Malheureusement, la théorie de la réduction des tenseurs d'ordre 3 est peu développée mathématiquement en dimension quelconque (voir section 1.4.5), à la différence de ceux d'ordre 2 (diagonalisation des matrices symétriques). Pourtant, nous avons remarqué en généralisant l'algorithme de réduction de Gauss aux tenseurs d'ordre trois, que cette réduction est souvent numériquement possible. Une alternative, peut-être plus simple, est d'utiliser au lieu de la seule matrice d'inertie, une fonction matricielle. Par exemple, on peut l'obtenir l'a montré [Slu95] en occultant l'ensemble par un disque de rayon r centré sur le barycentre, et en calculant la matrice d'inertie de la partie non-occultée en fonction de r .

1.4.3 Similitude/Rotation et homothétie

Le cas de la similitude se déduit assez facilement en adaptant la démonstration précédente. En effet, pour une similitude, la transformation L est de la forme sR , en conséquence, on a la contrainte :

$$L^t L = s^2 I_d$$

Puisque, $L = QI^{-1}$, $Q = ZD'W^t$, et $I = \Omega \Lambda \Omega^t$ on déduit que :

$$(\Omega^t W) D'^2 (\Omega^t W)^t = s^2 \Lambda^2$$

On a le même type de relation que dans la section précédente. Alors, si la diagonale de Λ a des termes tous différents, on a $\Omega = W$ et $\Omega' = Z$. La solution optimale pour la rotation est toujours $R = ZW^t$. De plus, on déduit $s\Lambda = D' = s^{-1}\Lambda'$, d'où la formule qui s'interprète naturellement géométriquement :

$$s = \sqrt{\frac{1}{p} \text{trace}(\Lambda' \Lambda^{-1})}$$

L'algorithme se résume donc de la façon suivante dans le cas où le nombre de solutions est fini :

- Calculer les matrices d'inertie des deux ensembles de points, dans les repères centrés sur les barycentres $I = \overline{Y_i Y_i^t}$ et $I' = \overline{Y'_i Y'_i{}^t}$.

- Trouver la décomposition en valeurs propres de $I = \Omega\Lambda\Omega^t$ et de $I' = \Omega'\Lambda'\Omega'^t$, où Ω et Ω' sont des rotations et Λ et Λ' sont à diagonale décroissante. Trouver les 2^{k-1} matrices C_k d'appariement de deux bases directes.
- Le meilleur facteur d'échelle est $s = \sqrt{\frac{1}{p}\text{trace}(\Lambda'\Lambda^{-1})}$.
- Les meilleures rotations sont $R = \Omega'C_k\Omega^t$.
- La meilleure translation est $T = \bar{X}'_i - R\bar{X}_i$

Comme dans la section précédente, il y a un nombre fini de solutions, sauf si I et I' ont des valeurs propres identiques.

1.4.4 Étirement/Rotation et changement d'échelle

L'utilité pratique du recalage à un étirement près a été bien comprise par Faber [FS88] au moins pour des applications 3D dans le domaine du médical. Nous n'avons en revanche pas rencontré d'algorithmes équivalents en 2D.

On ne dispose plus comme dans la section précédente d'un lien avec le problème avec correspondance. De fait, comment trouver à partir des attributs géométriques des deux ensembles de points, l'étirement qui transforme l'un en l'autre?

Faber a proposé un algorithme qui utilise des moments 3D d'ordre 2, 3, 4 et 5 pour retrouver la transformation. L'approche pour obtenir cet algorithme est similaire à celle de Cyganski [PCO85] qui contracte astucieusement, par des produits bien choisis, les tenseurs des moments d'ordre 2, 3, 4 et 5 pour obtenir des tenseurs combinés qui sont simplement des vecteurs covariants ou contravariants. À partir de ces vecteurs, retrouver la transformation linéaire est alors simple, puisque par définition ils se transforment comme des vecteurs normaux, ou comme des vecteurs par l'application transposée, suivant qu'ils sont covariants ou contravariants. En conséquence, la comparaison d'un nombre suffisant de ces vecteurs avant et après que la transformation soit appliquée permet de retrouver cette dernière. Dans [FS88], ces combinaisons usent implicitement du fait que la transformation recherchée est un étirement et non une transformation linéaire quelconque.

Une approche un peu différente utilisant les moments complexes [LD89] a permis d'exhiber dans le cas 3D, trois vecteurs covariants qui sont des combinaisons des seuls moments d'ordre trois. Cet algorithme est plus intéressant que le précédent car il ne fait pas appel aux moments d'ordre 4 et 5 qui sont très sensibles au bruit sur les données. Il se comporte donc mieux numériquement.

Néanmoins, on peut reprocher à ces algorithmes de ne pas explicitement obtenir un étirement, mais de trouver une transformation linéaire par un algorithme qui suppose qu'elle est un étirement. Le fait que l'information d'étirement soit utilisée implicitement, et non explicitement, indique que ces algorithmes doivent être peu robustes au bruit sur les données.

1.4.5 Affinité/Transformation linéaire

L'approche de Cyganski, évoquée dans la section précédente, consiste à retrouver la transformation linéaire entre les deux ensembles de points par comparaison de tenseurs d'ordre 1

(vecteurs covariants ou contravariants) qui sont des combinaisons des tenseurs géométriques des deux ensembles de points. Puisque chaque tenseur d'ordre 1 fournit une seule équation, suivant la dimension de l'espace de travail, le nombre nécessaire de ces vecteurs change. Suivant le nombre de vecteurs, il faudra contracter des tenseurs d'ordre plus ou moins élevé. Dans le cas 2D, Cyganski [CO85] a montré que l'utilisation des tenseurs d'ordre 2, 3 et 4 suffit pour calculer la transformation linéaire 2D subie par l'ensemble. Il a ensuite étendu sa démarche au cas 3D, où il a proposé une solution qui utilise les tenseurs d'ordre 2, 3, 4 et 5 [PCO85].

L'avantage de ces algorithmes réside dans la simplicité relative des calculs. Cependant, ils conduisent à utiliser des moments d'ordre supérieur à 3, ce qui n'est pas mathématiquement nécessaire, puisque les tenseurs des moments d'ordre 3, seuls, suffisent à caractériser une transformation linéaire.

Dans le cas 2D, l'algorithme dit de normalisation, souvent utilisé en traitement d'image pour recalculer deux images transformées l'une de l'autre par une affinité, résout le même problème en utilisant les moments d'ordre 1, 2 et 3 mais est spécifique à ce nombre de dimensions.

Cadre mathématique

Il nous semble que le cadre mathématique adéquat de ce problème est celui de l'algèbre des formes trilinéaires symétriques dans les espaces euclidiens, éventuellement associé aux notations tensorielles. Dans ce cadre, nous allons décrire une méthode originale de recalage affine valide dans le cas 2D, qui utilise uniquement le tenseur d'ordre 3. Des algorithmes plus simples peuvent être utilisés. Mais nous présentons cet algorithme car il nous semble avoir des chances d'être généralisable à une dimension quelconque.

De même qu'une forme quadratique est associée au tenseur d'inertie, il correspond, au tenseur d'ordre 3 des moments O , une forme triadique o . L'approche consiste donc à s'inspirer au maximum de la démarche utilisée dans la réduction d'une matrice symétrique pour, en quelque sorte, diagonaliser le tenseur d'ordre 3 des moments. Une fois déterminées les bases qui réduisent les deux tenseurs d'ordre 3 associés aux deux ensembles de points, la matrice de passage entre ces deux bases donne la transformation linéaire recherchée.

Un tenseur O d'ordre 3 en dimension n est constitué de l'ensemble $\{O_i\}_{1,\dots,n}$ de n matrices symétriques $n \times n$. En 2D, nous notons un tenseur symétrique de la façon suivante :

$$O = \left(\begin{array}{cc|cc} a & b & b & c \\ b & c & c & d \end{array} \right) = (O_1|O_2)$$

La forme triadique o , que O représente dans une base de deux vecteurs $\{E_i\}_{1,2}$ de \mathbb{R}^2 , est donc :

$$o(x, y) = ax^3 + 3bx^2y + 3cxy^2 + dy^3 \quad (1.23)$$

La forme polaire associée à cette forme triadique est la forme trilinéaire symétrique suivante :

$$\omega(X_1, X_2, X_3) = O \odot X_1 \odot X_2 \odot X_3$$

On remarque que $o(x, y)$ est un polynôme homogène en x et y de degré trois, dont on montre, par des calculs assez fastidieux en utilisant les formules de Cardan, que le discriminant est :

$$\delta = (ad - bc)^2 - 4(ac - b^2)(bd - c^2) \quad (1.24)$$

Ce discriminant permet, suivant son signe, de calculer le nombre de vecteurs isotropes normalisés de la forme triadique. Comme dans le cas des formes bilinéaires, un vecteur est dit isotrope s'il annule la forme triadique. Puisque la forme est homogène, tous les vecteurs proportionnels à un vecteur isotrope sont aussi isotropes. On parle alors d'axe ou de droite isotrope. Si $\delta > 0$, il y a une unique droite isotrope. Quand $\delta = 0$, la forme est dégénérée et possède deux vecteurs isotropes normalisés. Si $\delta < 0$, les droites isotropes sont au nombre de trois.

Similairement, en 3D, on a :

$$O = \left(\begin{array}{ccc|ccc|ccc} o_{111} & o_{121} & o_{131} & o_{121} & o_{221} & o_{231} & o_{131} & o_{231} & o_{331} \\ o_{121} & o_{221} & o_{231} & o_{221} & o_{222} & o_{232} & o_{231} & o_{232} & o_{332} \\ o_{131} & o_{231} & o_{331} & o_{231} & o_{232} & o_{332} & o_{331} & o_{332} & o_{333} \end{array} \right)$$

Classification des tenseurs d'ordre 3 en 2D

Supposons qu'il existe une base (E_1, E_2) de vecteurs où la forme triadique O est réduite à $x^3 + y^3$. Cette hypothèse implique que $\omega(E_1, E_2, E_1)$ et $\omega(E_1, E_2, E_2)$ sont nuls. En conséquence, puisque pour deux vecteurs X et Y , on a $X \odot Y = \text{trace}(XY^t) = X^t Y$, les produits scalaires suivants s'annulent :

$$(O \odot E_1 \odot E_2)^t E_1 = 0$$

$$(O \odot E_1 \odot E_2)^t E_2 = 0$$

On en déduit que les deux composantes du vecteur $O \odot E_1 \odot E_2$ sont nulles. Nous introduisons maintenant les deux matrices O_1 et O_2 qui composent O , et sont définies par :

$$O_1 = \begin{pmatrix} a & b \\ b & c \end{pmatrix}$$

$$O_2 = \begin{pmatrix} b & c \\ c & d \end{pmatrix}$$

Alors, le fait que les deux composantes de $O \odot E_1 \odot E_2$ soient nulles, se réécrit sous la forme :

$$(O_1 E_1)^t E_2 = 0$$

$$(O_2 E_1)^t E_2 = 0$$

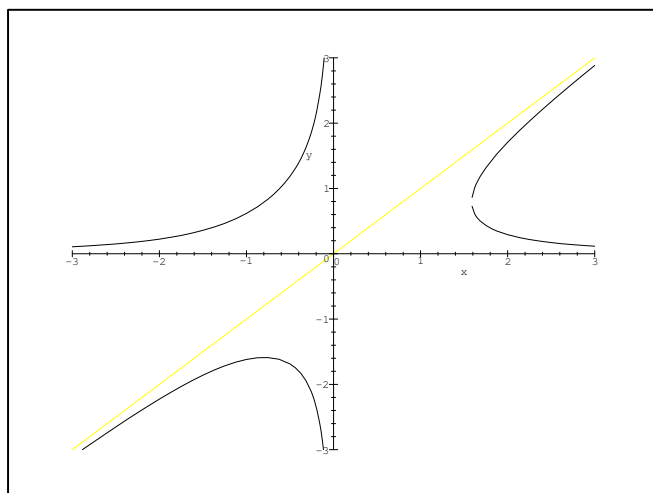
Il vient donc que $O_1 E_2$ et $O_2 E_1$ sont des vecteurs colinéaires. Or les vecteurs E_1 et E_2 jouent tous deux un rôle symétrique. En conséquence, une condition nécessaire pour que E_1 et E_2 existent est qu'ils soient des vecteurs solutions du problème propre généralisé suivant :

$$\exists \eta \text{ et } E \text{ non nuls} \quad \text{tels que} \quad O_1 E = \eta O_2 E$$

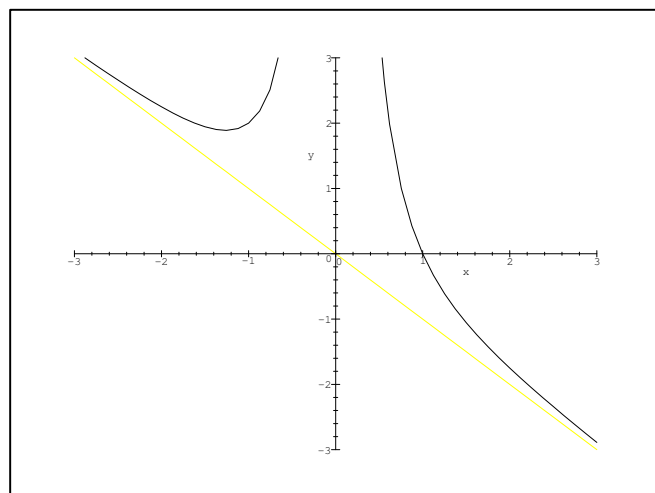
Le polynôme caractéristique de ce problème propre est donc :

$$\det(O_1 - \eta O_2) = (ac - b^2) - \eta(ad - bc) + \eta^2(bd - c^2)$$

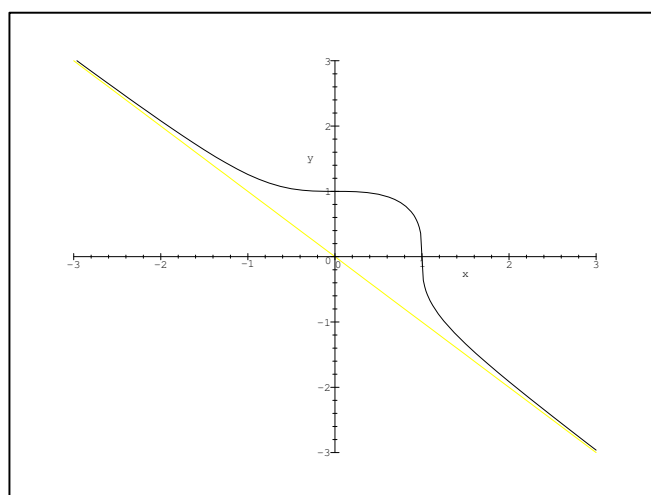
Le discriminant de ce polynôme de degré deux n'est autre que δ , celui du polynôme de degré trois (1.24). Le signe de ce discriminant donne donc le nombre possible de vecteurs propres normalisés, ou d'axes propres.



(a)



(b)



(c)

FIG. 1.2 - En (a) est tracée la fonction $x^2y - xy^2 = 1$ qui a trois branches. Les trois axes isotropes de la forme $x^2y - xy^2$ sont les trois asymptotes. En (c), la fonction $x^3 + y^3 = 1$ est visible. Elle est associée à une forme triadique de dos d'âne. Les deux axes propres de la forme $x^3 + y^3$ sont les deux axes (Ox) et (Oy). L'axe isotrope est l'asymptote. En (b) est tracé: $x^3 + x^2y = 1$, qui a deux asymptotes et qui est associé à une forme triadique dégénérée qui possède deux axes isotropes.

On peut donc établir la classification suivante des formes triadiques en 2D :

- Forme à trois branches de la figure 1.2(a) : $\delta < 0$, alors la forme triadique a trois axes isotropes et aucun vecteur propre.
- Forme dégénérée de la figure 1.2(b) : puisque $\delta = 0$, il a deux axes isotropes et un seul vecteur propre qui est aussi un vecteur isotrope ($(bd - c^2) \neq 0$) même si la valeur propre associée n'est pas nulle.
- Forme en dos d'âne de la figure 1.2(c) : comme $\delta > 0$, il a un axe isotrope et deux vecteurs propres normés distincts.

Réduction d'un tenseur d'ordre 3 en 2D

Cherchons à présent une base où le tenseur a une expression réduite.

Nous avons montré dans la section précédente que les formes triadiques à dos d'âne ont deux vecteurs propres distincts. On choisit la longueur et l'orientation de ceux-ci pour que $o(E_1) = o(E_2) = 1$. Cela est réalisable très simplement en divisant le vecteur E_i par $\sqrt[3]{o(E_i)}$. Cette base (E_1, E_2) réduit la forme triadique à $x^3 + y^3$. Si l'on impose en plus qu'elle soit aiguë et directe, la base réductrice est unique (voir figure 1.2(c)).

Ce résultat n'est valable que pour les formes triadiques à discriminant positif. Si celui-ci est négatif, on sait qu'il existe 3 vecteurs isotropes normés différents. Deux d'entre eux peuvent être utilisés pour former une base. On adapte alors les normes, de sorte que la forme soit réduite à $xy(x - y)$ dans cette base. Il est donc possible de réduire une forme triadique de discriminant négatif sous la forme $x^2y - xy^2$. Mais, même en imposant que la base aiguë soit directe, il y a dans ce cas 3 bases possibles (voir figure 1.2(a)).

Cette différence dans la réductibilité indique juste qu'il n'est pas possible de trouver une transformation linéaire réelle entre deux ensembles de points qui ont des tenseurs d'ordre 3 de types différents. En revanche, si les deux tenseurs sont de même type, c'est-à-dire s'ils ont tous les deux un discriminant négatif ou positif, la transformation linéaire L qui recale l'ensemble des X_i sur l'ensemble de X'_i est :

$$L = (E'_1 E'_2) C_k (E_1 E_2)^{-1}$$

On note $\{E_i\}$ la base qui réduit le tenseur avant transformation, et $\{E'_i\}$ celle après transformation. Comme dans la section 1.4.2, il faut introduire les matrices d'appariement de deux bases : I_d et $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$. Il y a donc deux transformations linéaires possibles si $\delta > 0$, et six si $\delta < 0$. Dans la section 1.4.2, les solutions obtenues avec les matrices d'inertie sont valides si les tenseurs d'ordre 3 des deux ensembles de points se transforment aussi par la solution. Inversement, ici, les solutions L obtenues par la réduction des tenseurs sont validées en regardant si la matrice d'inertie I se transforme bien en I' par L , soit $I' = LIL^t$.

On peut donc résumer l'algorithme d'obtention de la transformation affine optimale en 2D, entre deux ensembles compatibles (δ et δ' de même signe) :

- Calculer les tenseurs des moments d'ordre 3 des deux ensembles de points, dans les repères centrés sur les barycentres $O = \overline{Y_i \odot Y_i \odot Y_i} = (O_1|O_2)$ et $O' = \overline{Y'_i \odot Y'_i \odot Y'_i} = (O'_1|O'_2)$. Calculer les discriminants δ et δ' avec (1.24).

- Si δ et δ' sont positifs, trouver la base de valeurs propres (E_1, E_2) de $O_1^{-1}O_2$ et (E'_1, E'_2) de $O_1'^{-1}O_2'$. Diviser E_i par $O(E_i)^{1/3}$ et E'_i par $O'(E'_i)^{1/3}$.
- Si δ et δ' sont négatifs, choisir la base de vecteurs isotropes normés (E_1, E_2) de O et (E'_1, E'_2) de O' . Diviser E_1 par $-\frac{\omega(E_1, E_1, E_2)^{2/3}}{\omega(E_1, E_2, E_2)}$, E_2 par $\frac{\omega(E_1, E_2, E_2)^{2/3}}{\omega(E_1, E_1, E_2)}$, E'_1 par $-\frac{\omega(E'_1, E'_1, E'_2)^{2/3}}{\omega(E'_1, E'_2, E'_2)}$, et E'_2 par $\frac{\omega(E'_1, E'_2, E'_2)^{2/3}}{\omega(E'_1, E'_1, E'_2)}$.
- Les deux ou six transformations linéaires possibles sont $L = (E'_1 E'_2)C_k(E_1 E_2)^{-1}$, avec $C_k = I_d$ ou $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$.
- La meilleure translation est $T = \overline{X}'_i - L\overline{X}_i$

1.4.6 Homographie

Nous avons cherché à estimer une transformation homographique, dans le cas sans correspondance, en utilisant la propriété selon laquelle une homographie est une transformation linéaire en coordonnées homogènes. Alors, estimer une homographie de \mathbb{R}^p revient à trouver une transformation linéaire de norme 1 dans \mathbb{R}^{p+1} . En effet :

$$\mathcal{X}'_i \equiv \mathcal{H}\mathcal{X}_i \quad \text{ou} \quad \mathcal{X}'_i s_i = \mathcal{H}\mathcal{X}_i$$

Soit, en divisant par la norme terme à terme :

$$\frac{\mathcal{X}'_i}{\|\mathcal{X}'_i\|} = \frac{\mathcal{H}\mathcal{X}_i}{\|\mathcal{H}\mathcal{X}_i\|}$$

Maintenant, nous imposons à \mathcal{X}'_i et \mathcal{X}_i d'être de norme 1, au lieu d'imposer que la dernière composante est 1. Avec ces nouvelles contraintes, le problème revient à trouver L telle que :

$$\mathcal{X}'_i = \frac{\mathcal{H}\mathcal{X}_i}{\|\mathcal{H}\mathcal{X}_i\|}$$

Cette équation peut se résoudre itérativement :

- Normaliser \mathcal{X}'_i et \mathcal{X}_i à 1. Calculer \mathcal{H}_1 qui superpose \mathcal{X}'_i et \mathcal{X}_i .
- Calculer \mathcal{H}_{j+1} qui superpose \mathcal{X}'_i et $\frac{\mathcal{X}_i}{\|\mathcal{H}_j \mathcal{X}_i\|}$.
- Si, \mathcal{H}_{j+1} est stable s'arrêter, sinon retourner à l'étape précédente.

Cet algorithme n'est stable que si l'homographie s'approche bien par une affinité dans le domaine de l'espace où sont les points, ce qui restreint de façon importante ses potentialités.

1.4.7 Projection perspective

On peut s'interroger sur la possibilité de faire l'estimation de la transformation sans correspondance dans le cas des transformations linéaires lorsqu'elles sont composées avec une projection centrale, ou avec une transformation quelconque. Puisqu'aucune combinaison des moments géométriques n'est invariante par la transformation homographique, la limite d'utilisation des moments semble atteinte avec l'affinité pour obtenir une solution explicite, lorsque l'on ne dispose pas de la correspondance.

Une approche alternative consiste à utiliser toujours les moments d'inertie, mais en minimisant l'écart entre les moments de $\{X_i\}_{1,\dots,n}$ transformés par F , et les moments de $\{X'_i\}_{1,\dots,n}$. Pour que le problème de minimisation soit bien posé, il faut au moins que le nombre de moments indépendants utilisés soit supérieur au nombre de degrés de liberté de la transformation F . L'ordre des moments à considérer est donc directement lié au nombre de degrés de liberté de la transformation. Les moments d'ordre 1, soit le barycentre, fournissent p contraintes, puisque l'on est dans \mathbb{R}^p . Les moments d'inertie d'ordre deux, donnent $\frac{p(p+1)}{2}$ contraintes. À l'ordre 3, ils en fournissent $\frac{p(p+1)(p+2)}{6}$. Dans le cas général, l'ordre m fournit C_{p+m-1}^m contraintes.

Alors, étant donnés deux ensembles de points, estimer au mieux la transformation entre les deux ensembles bruités sans appariement revient à minimiser la fonctionnelle suivante :

$$e(F) = \sum_{m=1}^{n'} \|M^m(F(X_i)) - M^m(X'_i)\|^2 \quad (1.25)$$

où les tenseurs des moments sont définis par (1.17) et où le nombre n' de tenseurs à utiliser est une fonction du nombre des paramètres de F . Minimiser cette fonction (1.25) amène à des algorithmes itératifs qui ne sont pas stables dans la pratique (voir section 1.5.5). La limite d'utilisation des moments géométriques est donc bien atteinte avec l'affinité.

1.4.8 Conclusion

Il semble être possible d'estimer, de façon robuste, uniquement des transformations intrinsèquement affines entre deux ensembles de points sans avoir les correspondances point-à-point. Dans le cadre de l'estimation géométrique sans appariement, le principal problème qui reste sans réponse très satisfaisante est celui de l'estimation de la transformation affine en dimension quelconque supérieure à 2. L'aide de la géométrie algébrique devrait aider à résoudre cette question.

1.5 Robustesse dans la pratique

1.5.1 Influence du bruit sur l'estimation du déplacement avec ou sans appariement

Nous avons montré dans la section 1.4.2, que mathématiquement, les solutions du problème avec et sans appariement coïncident. On a donc, deux algorithmes qui doivent donner

la même solution. Or, on sait que l'égalité mathématique n'est pas l'équivalence algorithmique, en particulier en ce qui concerne le comportement vis-à-vis d'un bruit gaussien centré. Ainsi, même si l'erreur d'estimation de l'angle de rotation est expérimentalement proportionnelle à l'écart type de ce bruit dans les deux cas, l'algorithme des bases propres se révèle, sur les mêmes données, un peu moins précis que celui fondé sur la matrice de corrélation croisée (voir figure 1.3). Il n'est pas étonnant, que l'algorithme qui utilise l'information d'appariement soit numériquement plus précis que celui qui n'en tient pas compte.

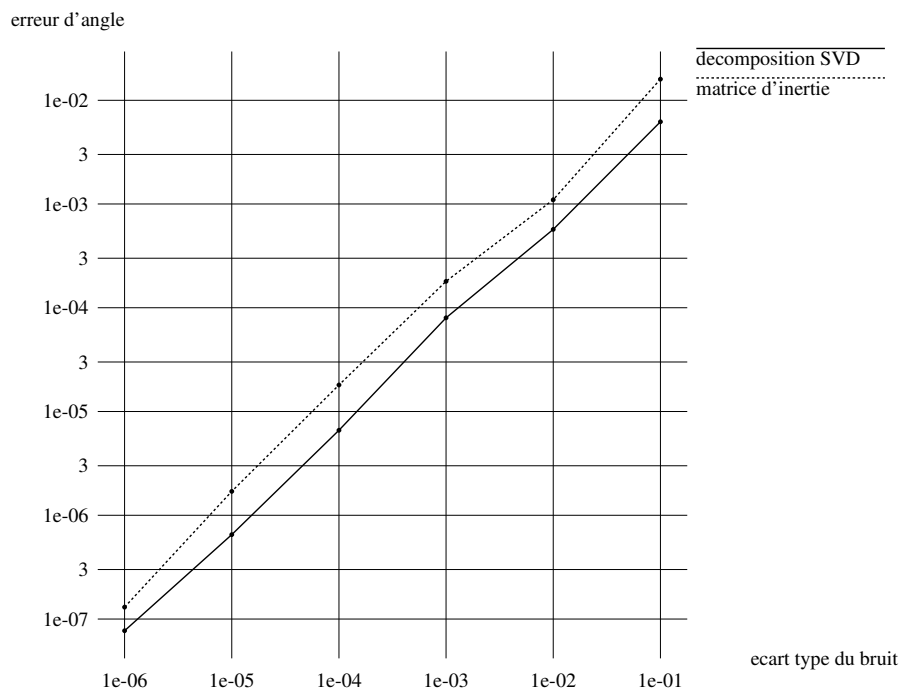


FIG. 1.3 - Comparaison de l'erreur d'estimation de l'angle de la rotation entre l'algorithme qui utilise la décomposition SVD et nécessite de connaître les appariements et celui qui utilise les matrices d'inertie où l'appariement est inconnu.

Les mesures de la figure 1.3 sont obtenues avec un ensemble X_i de 50 points 2D aléatoires distribués selon une gaussienne d'écart type 1.0. Le point X'_i est obtenu par l'application d'une rotation de 90° sur X_i , additionné d'un bruit gaussien centré.

1.5.2 Influence du bruit sur l'estimation de la similitude avec ou sans appariement

Nous avons fait des mesures similaires à la section précédente sur l'erreur d'estimation de l'échelle. Les résultats de la figure 1.4 sont obtenus avec 50 points 2D X_i aléatoirement distribués selon une gaussienne d'écart type 1.0 sur le premier axe et 2.0 sur le deuxième. Les points X'_i sont obtenus par l'application d'une rotation de 90° sur les X_i , additionnés

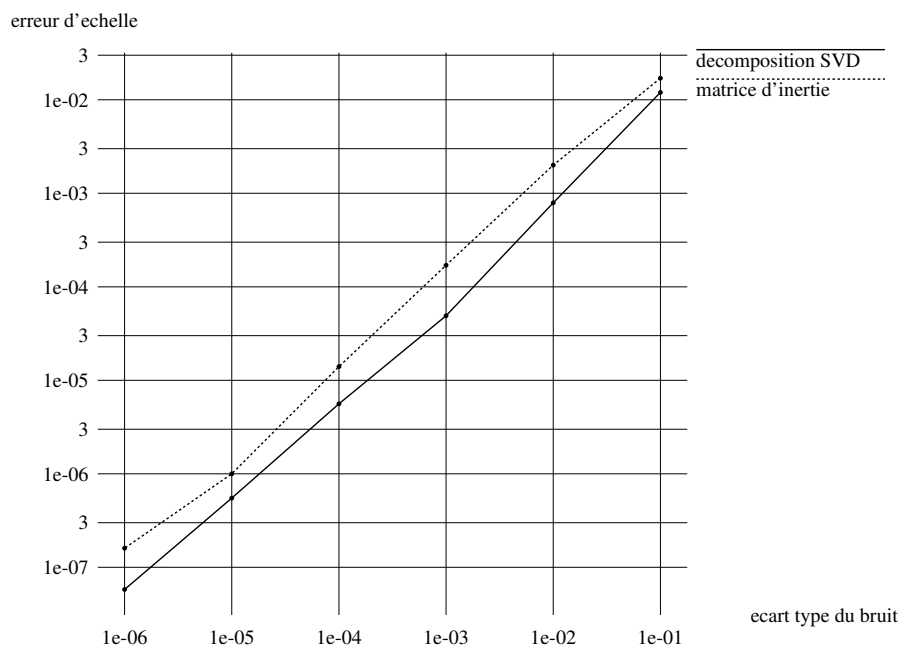


FIG. 1.4 - Comparaison de l'erreur d'estimation de l'échelle entre l'algorithme qui utilise la décomposition SVD et celui qui utilise les matrices d'inertie.

d'un bruit gaussien centré. L'algorithme qui utilise la décomposition SVD est deux fois plus précis mais les résultats avec ou sans appariement restent proches.

L'erreur d'estimation du facteur d'échelle est du même ordre de grandeur que celle sur l'angle de la rotation. Comme pour l'angle de rotation, l'erreur d'estimation de l'échelle est proportionnelle à l'écart type du bruit sur les données.

1.5.3 Influence du bruit sur l'estimation d'une transformation linéaire 2D sans appariement

Nous avons aussi comparé l'algorithme de Cyganski avec le nôtre qui permet d'estimer une transformée affine en 2D sans correspondance avec uniquement le tenseur d'ordre 3. Le cadre expérimental de ces résultats est identique à celui de la section précédente. Les coefficients de la matrice linéaire L sont de l'ordre de 1. Sur ce jeu de données, on constate que l'algorithme de Cygansky qui utilise les moments d'ordre 4 est 5 fois moins précis que notre algorithme. Pour d'autres ensembles de points 2D, l'écart entre les deux méthodes peut se creuser et atteindre un rapport 70 en faveur de notre méthode.

Nous avons vérifié avec des tests intensifs que l'erreur d'estimation des coefficients de l'affinité reste proportionnelle à l'écart type du bruit, malgré l'utilisation de tenseurs d'ordre 3 ou 4. Par contre, l'influence d'un même bruit produit une erreur différente d'un ou deux ordres de grandeur, entre les méthodes sans et avec correspondance. La différence de comportement vis-à-vis du bruit entre les méthodes avec et sans appariement apparaît donc dès que la

transformation est supposée plus riche qu'une similitude.

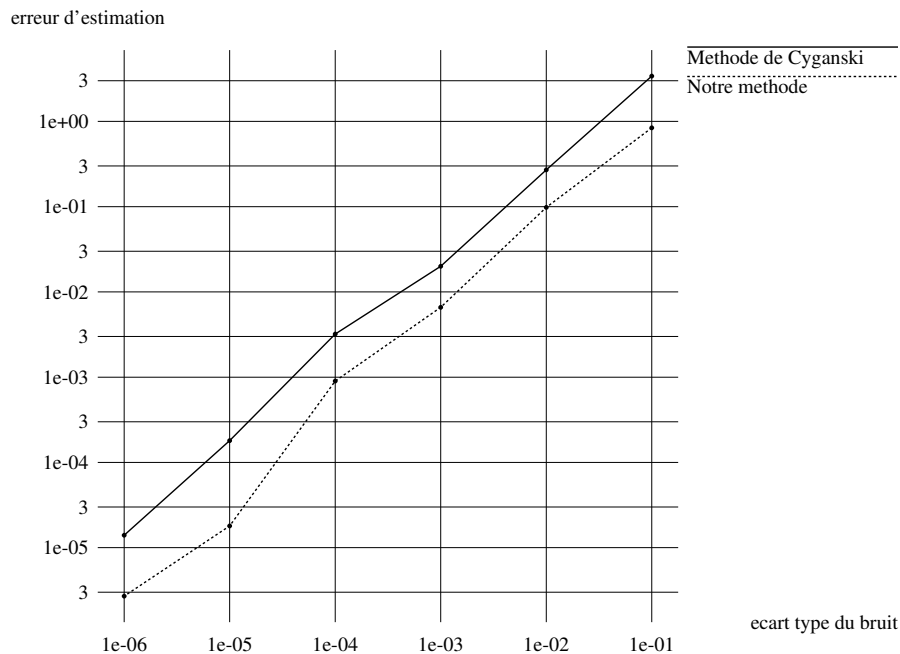


FIG. 1.5 - Comparaison de l'erreur d'estimation de la transformée linéaire entre notre algorithme qui utilise les tenseurs d'ordre 3 et celui de Cyganski qui utilise les tenseurs jusqu'à l'ordre 4.

1.5.4 Influence du bruit sur l'estimation d'une transformation homographique 1D sans appariement

Nous avons implanté la méthode d'estimation d'une transformation homographique lorsque l'on ne dispose pas des appariements, décrite en section 1.4.6, dans le cas à une dimension. Seul le cas 1D a pu être programmé puisque l'on ne dispose que d'un algorithme 2D pour estimer une affinité sans les correspondances (section 1.4.5).

La robustesse par rapport au bruit a été testée. Comme on peut s'y attendre, la résistance au bruit de cette méthode est assez faible comme en témoigne la figure 1.6. L'algorithme étant itératif, il convient d'observer sa robustesse par rapport à l'éloignement du point de départ. Dans ces expérimentations, la transformation de départ avait des coefficients différents de pas plus de 50%, l'algorithme ne pouvant pas supporter un écart beaucoup plus grand.

En conséquence, dans la pratique cet algorithme se révèle pas assez robuste pour avoir une utilité autre que celle de faire un raffinement homographique.

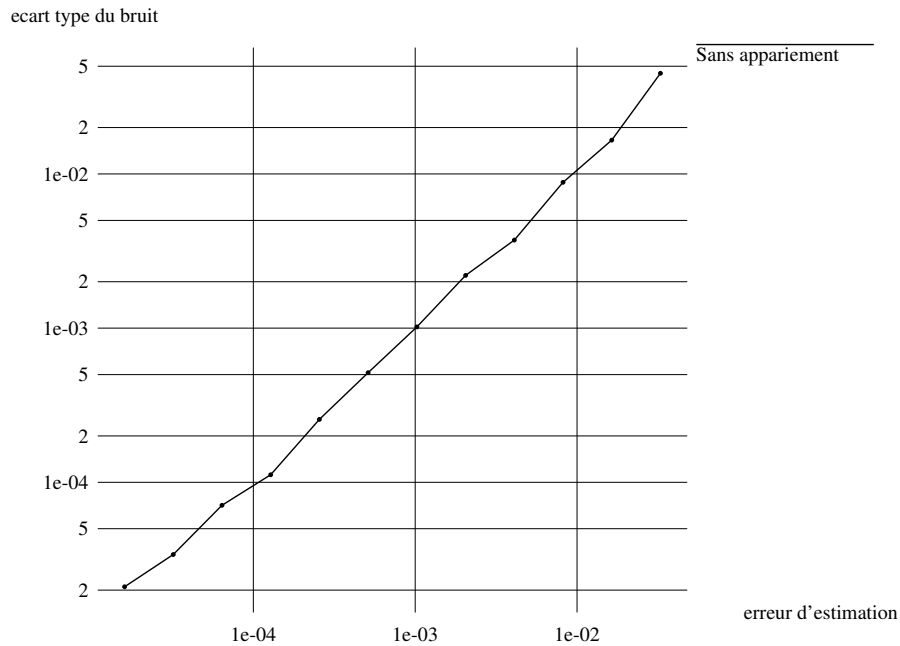


FIG. 1.6 - Erreur d'estimation de la transformée homographique 1D sans appariement.

1.5.5 Estimation d'une transformation perspective avec ou sans appariement

Nous avons proposé, dans la section 1.4.7, une méthode pour estimer une homographie sans appariement en minimisant la différence des moments. Malheureusement, cette technique trop sujette à être piégée dans des minima locaux se révèle inutilisable dans la pratique. De plus, elle possède parfois plusieurs minima numériques absolus. Par exemple, dans le cas d'une projection perspective 2D/1D, le critère choisi est la différence des moments de l'ensemble des points 1D 'image' et des projections u , jusqu'à l'ordre 8. Or, il est apparu sur des exemples numériques particuliers que les deux transformations suivantes minimisent le même critère. La transformation originale est :

$$u = \frac{x + 0.1y - 1}{0.1x + y + 5}$$

La transformation perspective qui minimise numériquement aussi le critère est :

$$u = \frac{x + 0.145256y - 1.041142}{0.145259x + y + 5.205709}$$

Puisqu'obtenir une transformation perspective est important pour la calibration de caméra en particulier, nous avons aussi testé la stabilité de l'estimation avec appariement en présence de bruit. Dans ces tests (voir figure 1.7), nous avons constaté que l'estimation est

plus délicate que dans le cas d'une transformation linéaire (déplacement, similitude ou affinité) car elle est d'autant plus sensible au bruit sur les points projetés que l'ensemble de points à projeter est éloigné du plan de projection.

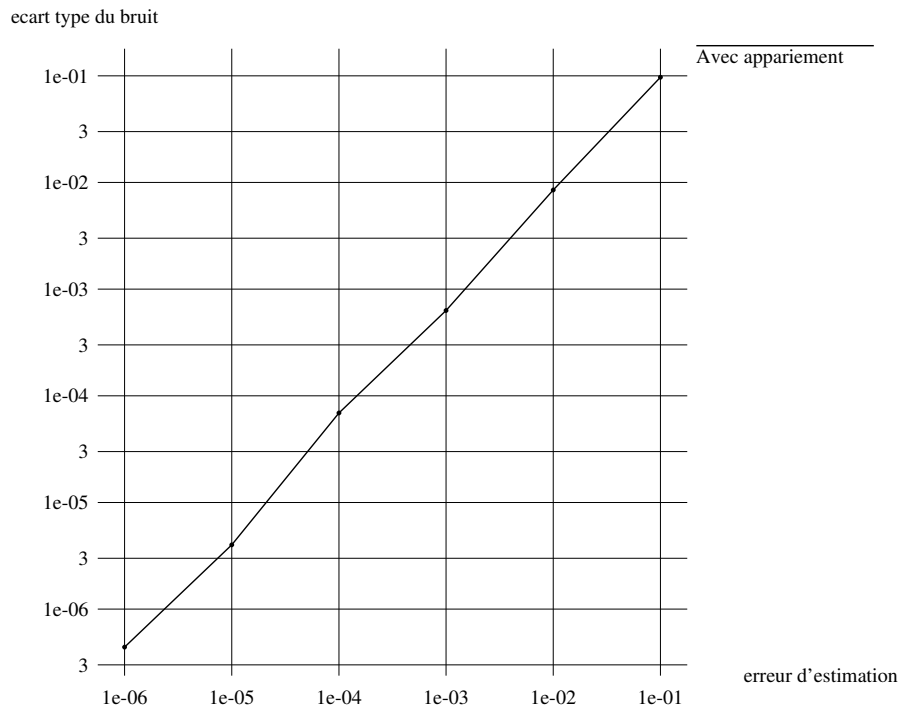


FIG. 1.7 - Erreur d'estimation de la transformée perspective avec appariement.

1.6 Appariement et estimation couplés

Nous venons de décrire les algorithmes qui permettent de calculer une transformation géométrique, quand elle est de type usuel, entre deux ensembles de points donnés. À chaque fois, l'estimation de la transformation est séparée de la recherche des correspondances entre les points. Sans appariement, le cas de deux ensembles transformés par une application non-linéaire ne peut être traité de manière satisfaisante par les approches exposées précédemment.

Une autre configuration importante n'a pas aussi été traitée. C'est le cas où uniquement une partie des points $\{X'_i\}$ peut être mise en correspondance par F . Il faut alors déterminer la partie concernée en même temps que l'on estime F . On fait alors appel aux techniques d'estimation robustes [HJL⁺89].

Si l'un des ensembles de points est incomplet sans contenir de points aberrants, cela se traduit par l'inclusion du plus petit ensemble dans le plus grand. Nommons le plus grand de ces ensembles, l'ensemble modèle. On dispose alors d'une caractérisation du problème qui peut se résoudre, par exemple, dans le cadre *hypothèse-vérification* [Aya88]. Nous avons

utilisé une autre approche fondée sur l'utilisation de la fonction de plus proche voisin proposée en particulier pour faire le recalage rigide (algorithme ICP). Cette dernière approche est en effet plus naturelle pour introduire les méthodes de classification qui sont un cadre utile pour étendre les algorithmes décrits dans ce chapitre au cas où il y a plusieurs transformations F à obtenir en même temps sur une partition des données.

1.6.1 Méthodes ICP

Déplacements

Besl [BM92] a introduit un algorithme pour effectuer du recalage rigide en 3D, nommé 'Iterative Closest Point' (ICP). Cet algorithme permet de recalibrer un objet modèle défini par un ensemble de points, de lignes, de courbes implicites, de courbes paramétriques, de surfaces maillées, de surfaces implicites, ou de surfaces paramétriques. Soit S_{ref} , ce modèle de référence. Pour rester cohérent avec le début du chapitre, S_{ref} est ici égal à $\{X_i''\}_{1,\dots,n}$.

L'algorithme minimise itérativement la distance entre le modèle et les données sous la forme de l'erreur aux moindres carrés suivante :

$$e_{ICP}(R, T, X_i'') = \sum_{i=1}^n d^2(RX_i + T, X_i'') \quad \text{avec} \quad X_i'' \in S_{ref} \quad (1.26)$$

où X_i'' est le point le plus proche de X_i qui est contenu dans S_{ref} .

La méthode de résolution est alors d'itérer une étape d'appariement suivie d'une étape d'estimation du déplacement. Dans la première phase de chaque itération, la mise en correspondance d'un point des données X_i est réalisée avec le plus proche point X_i'' sur le modèle S_{ref} . Dans la deuxième phase, le meilleur déplacement (R, T) entre les données $\{X_i\}_{1,\dots,n}$ et l'ensemble de leurs correspondants $\{X_i''\}_{1,\dots,n}$ est calculé en utilisant une des méthodes de la section 1.3.2. La convergence de l'algorithme ICP vers un minimum local est démontrée, lorsque les données sont un sous-ensemble du modèle transformé par un déplacement rigide.

L'algorithme ICP est le suivant :

étape 0 : Fixer le modèle de référence S_{ref} .

étape i : Calculer le point X_i'' appartenant au modèle S_{ref} qui est le plus proche du point X_i , pour chaque i .

étape i' : Calculer le déplacement (R, T) qui minimise l'erreur aux moindres carrés entre les points X_i'' et X_i et l'appliquer sur $\{X_i\}_{1,\dots,n}$.

étape i'' : Si l'erreur aux moindres carrés est réduite d'une valeur inférieure à un seuil donné a priori qui spécifie la précision, alors il faut s'arrêter ou sinon retourner à l'**étape i**.

Transformation non-rigide

La généralisation de la méthode ICP à d'autres types de transformations est possible, tout en gardant exactement le même schéma d'algorithme.

Nous avons développé plus particulièrement, dans le dernier chapitre, en se fondant sur les algorithmes d'estimation de la section 1.3, l'estimation d'une transformation homographique, d'une affinité, d'un étirement ou d'une similitude, entre des données partielles et un modèle de l'objet à recaler, lorsque la correspondance est inconnue.

La sous-classe des algorithmes avec appariement vers le plus proche voisin, décrite dans le dernier chapitre, semble particulièrement intéressante et relativement efficace. Pour l'algorithme GICP, généralisation de l'algorithme 'Iterative Closest Point', le couplage entre la recherche des appariements et l'estimation est résolu en posant la totalité du problème comme une estimation. En effet, pour un modèle unique constitué des points $S_{ref} = \{X_i\}_{1,\dots,n}$, l'erreur aux moindres carrés minimisée est :

$$e_{GICP}(F, X_i'') = \sum_{i=1}^n d^2(F(X_i), X_i'') \quad \text{avec} \quad X_i'' \in \{X_i'\}_{1,\dots,n} \quad (1.27)$$

La partie $\{X_i''\}_{1,\dots,n}$, de l'ensemble $\{X_i'\}_{1,\dots,n}$, qui peut être appariée, est donc obtenue en même temps que la transformation optimale F comme la solution d'un problème d'estimation.

Néanmoins, cette méthode est itérative, elle présente donc le désavantage de devoir être initialisée. Heureusement, pour une application donnée, il existe souvent une méthode d'initialisation qui donne une bonne transformation de départ. Cette initialisation est toujours très dépendante de l'application visée. Un exemple de ce type de coopération entre une méthode d'initialisation peu précise et un raffinement ICP est présenté dans le chapitre 4 pour faire du recalage rigide.

Il est important de noter que ce qui fait la robustesse de l'ICP est de supposer l'ensemble des données $\{X_i\}$ incluse dans le modèle $\{X_i'\}$. Cela permet aussi, à son extension à d'autres types de transformations, d'être robuste. D'autres types d'extensions de l'ICP sont possibles, comme les déformations affines locales et projections perspectives 3D/2D [Fel95].

Points parasites

Les deux algorithmes précédents supposent que les données sont incluses dans la forme modèle. Or, si ce n'est pas le cas, la présence de points parasites dans les données qui ne peuvent pas être recouverts par la forme modèle recalée, biaise les résultats de manière significative. Il faut donc trouver de façon robuste la partie des données qu'il faut recaler. Cela nécessite d'être capable de différencier les données potentiellement intéressantes des autres. Il faut donc avoir une caractérisation de ce que sont les données aberrantes. Divers travaux se sont attachés, dans le cadre ICP, à développer des versions robustes aux points parasites, en utilisant différents critères statistiques pour les éliminer [Zha94, KVB93, FA94]

Mais, avec ces méthodes améliorées, comme cela l'a été remarqué [FA94], moins la transformation est contrainte, plus l'algorithme GICP va pouvoir converger vers une forme dégénérée incluse dans un sous espace. En conséquence, il est nécessaire de contraindre la méthode GICP à expliquer toutes les données, ou alors caractériser de façon plus précise un point aberrant. L'approche floue de la classification permet de faire dans un cadre cohérent.

1.6.2 Classification floue

D'après [Sto90], le cadre de la classification pour faire du recalage est plus satisfaisant que celui fourni par le paradigme hypothèse-vérification pour faire du recalage. Nous pouvons ajouter à ces arguments que le cadre de la classification est intéressant car il permet d'étendre le recalage dans le cas où plusieurs transformations se partagent les données. De plus, la classification a introduit depuis longtemps les méthodes d'estimation robuste par la définition d'une classe de rejet. En particulier, opérer avec deux classes, dont une de rejet, est une méthode d'estimation robuste du type M-estimateur.

Il existe de nombreuses méthodes de classification, mais nous avons choisi l'approche floue car elle est réputée pour ses qualités quant à sa convergence, et comme l'a montré Krishnapuram [KF92], cette méthode permet de traiter les cas où les classes se recoupent, plus efficacement. Plus profondément, l'approche floue transforme un problème binaire en un problème continu, donc permet de transformer une recherche combinatoire en une estimation par minimisation d'une fonctionnelle.

L'approche floue

La modélisation des problèmes de vision avec des sous-ensembles flous [Bou93] permet de reporter les prises de décision en phase ultime d'un processus de traitement, minimisant ainsi la propagation des erreurs décisionnelles dans des phases intermédiaires. Des approches locales ou globales peuvent être utilisées pour définir les fonctions d'appartenance floues caractéristiques des propriétés des données traitées en vision. Le premier type d'approche, proposé par Pal et Rosenfeld, généralise les concepts de topologie et de géométrie (connexité, adjacence, compacité) dans le contexte des sous-ensembles flous. Les approches globales sont fondées sur les techniques de classification adaptative floue introduites initialement par Ruspini, Dunn et Bezdek, qui minimisent une famille de fonctionnelles de base : 'Fuzzy c-Means' (FCM) [Bez81]. Le principe de la classification floue est d'autoriser une appartenance partielle ou distribuée de chaque donnée à l'ensemble des classes de l'espace des caractéristiques.

Nous allons présenter deux algorithmes de classification et leurs avantages respectifs : FCM et l'extension en présence de points parasites de Davé.

La classification floue

L'algorithme FCM et ses dérivés ont été utilisés avec succès dans de nombreuses applications en reconnaissance des formes et en segmentation d'images. La classification floue est une généralisation des méthodes de partitions où les classes de l'ensemble des n points dans \mathbb{R}^p sont des sous-ensembles non pas classiques mais flous. Alors, chaque point X_i a un degré d'appartenance u_{ik} , à la classe k , entre 0 et 1. La somme des degrés d'appartenance u_{ik} d'un point sur l'ensemble des classes doit valoir 1. Cette contrainte $\sum_{k=1}^c u_{ik} = 1$ oblige les classes à recouvrir toutes les données par les c sous-ensembles flous.

L'algorithme de classification minimise la fonctionnelle suivante :

$$e_{FCM}(T_k, u_{ik}) = \sum_{i=1}^n \sum_{k=1}^c u_{ik}^m d^2(X_i, T_k) \quad \text{avec} \quad \sum_{k=1}^c u_{ik} = 1 \quad (1.28)$$

L'algorithme FCM se résume ainsi (la convergence vers un minimum local est démontrée dans [Bez81] quand $m > 1$):

étape 0 : Donner le nombre de classes c , et le facteur flou m (généralement choisi égal à 1.5). Initialiser aléatoirement la position T_k des centres des classes.

étape f : Générer une nouvelle partition en utilisant l'équation suivante du degré d'appartenance floue :

$$u_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d(X_i, T_k)}{d(X_i, T_j)} \right)^{\frac{2}{m-1}}} \quad (1.29)$$

étape f' : Calculer les nouveaux centres de classes T_k en utilisant l'équation suivante :

$$T_k = \frac{\sum_{i=1}^n u_{ik}^m X_i}{\sum_{i=1}^n u_{ik}^m} \quad (1.30)$$

étape f'' : Si le centre des classes est stable, il faut s'arrêter, sinon retourner à l'**étape f**.

C'est le centre de gravité T_k , qui est le prototype de la classe k .

La complexité de chaque étape est de l'ordre de $O(cnp)$, ce qui peut conduire à un coût de calcul important avec un grand nombre de classes.

Classification floue avec des points parasites

Les résultats de l'algorithme précédent peuvent beaucoup changer en présence de points parasites. Davé [Dav91] a étendu le FCM pour prendre en compte ces points aberrants et il a introduit, dans les approches floues, le concept de classe de rejet pour les récolter. Par définition, tous les points sont à la même distance du prototype T_c de la classe de rejet. Pour contrôler cet algorithme, un nouveau paramètre δ est nécessaire, qui permet d'indiquer qu'elle est la distance minimale à partir de laquelle un point est considéré comme trop éloigné du centre d'une classe pour pouvoir lui appartenir.

Une connaissance supplémentaire est donc introduite : le paramètre δ . Ce paramètre spécifie l'épaisseur de la zone d'influence de chaque prototype.

L'algorithme FCM modifié avec la classe de rejet est le suivant :

étape 0 : Fixer le nombre c de classes, et le facteur flou m (généralement $m = 1.5$). Calculer la position initiale des centres des classes T_k en utilisant la méthode FCM. Spécifier la distance δ qui définit la zone d'influence géométrique de chaque prototype.

étape f : Générer une nouvelle partition en utilisant l'équation (1.29) du degré d'appartenance flou u_{ik} .

étape f' : Calculer le nouveau centre des classes T_k en utilisant l'équation (1.30) seulement pour k entre 1 et $c - 1$.

étape f'' : Si le centre des classes est stable, il faut s'arrêter, sinon retourner à l'**étape f**.

Dans cette description, X_i est un des n éléments de l'ensemble des données contenues dans \mathbb{R}^p . $d(X_i, T_k)$ est défini comme la distance euclidienne entre le point X_i et le centre de la classe T_k pour k entre 1 et $c - 1$, et pour la classe de rejet, la distance est $d(X_i, T_c) = \delta$. La valeur de δ doit être choisie avec attention (voir deux exemples dans les sections 4.2.2 et 5.5.2).

La convergence de cet algorithme est assurée. La classe de rejet permet d'avoir une méthode de classification qui intègre un estimateur robuste de type M-estimateur. Un M-estimateur limite l'influence des points aberrants en les pondérant itérativement par une valeur qui décroît avec la distance. La classification fait exactement de même par le biais du degré d'appartenance u_{ik} qui est une fonction inverse de la distance du point X_i au prototype T_k relativement à tous les prototypes (voir équation (1.29)).

Extensions

L'extension du recalage à plusieurs transformations doit faire appel à la plupart des algorithmes décrits ou introduits dans ce chapitre. En particulier, nous avons étudié une sous-catégorie des problèmes d'estimation couplés à l'appariement que nous appelons les problèmes de recouvrement géométrique. Dans ce type de problème, les données doivent être segmentées pour que, sur chaque partie, une transformation géométrique de même type puisse être estimée. Dans le chapitre 5, nous présentons donc cette extension de la classe des algorithmes avec appariement vers le plus proche voisin, au cas où différentes instances de l'ensemble à recalcer apparaissent et recouvrent la totalité des données. Un exemple typique de cette classe de problèmes est la segmentation d'images en régions qui présentent le même type de distribution d'intensités.

Par cette approche, les problèmes de recouvrement, même avec plusieurs transformations à estimer en parallèle, sont donc réduits à la minimisation d'une fonction qui n'est pas analytiquement explicite, mais qui peut être minimisée sur un support continu. Cette propriété est importante puisqu'elle évite d'avoir à recourir aux méthodes combinatoires. Ce type de problèmes constitue la troisième catégorie de problèmes que nous avons défini dans ce chapitre.

La dernière catégorie de problèmes correspond à ceux qui restent très difficiles par nature. En effet, comme le montre la figure 1.8, les problèmes d'estimation pure ou globale sans mise en correspondance supposent que l'appariement entre les deux ensembles est bijectif. Les problèmes qui se réduisent à une minimisation sur un support continu font l'hypothèse d'inclusion d'un ensemble dans l'autre. Par contre, la dernière catégorie doit résoudre des problèmes où les appariements n'ont aucune des précédentes contraintes simplificatrices. La classification floue avec une classe de rejet donne uniquement un début de réponse pour traiter de manière robuste ces problèmes.

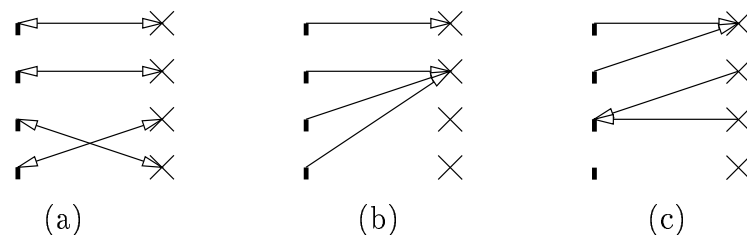


FIG. 1.8 - *Différents types d'appariements possibles entre deux ensembles de quatre points. La figure (a), où l'appariement est bijectif, correspond à celui qui est supposé dans les deux premières catégories. La figure (b) montre le type d'appariement qui apparaît dans les problèmes de la troisième catégorie. Enfin, la figure (c) montre les types d'appariements qui doivent être traités pour résoudre un problème de la dernière catégorie.*

1.7 Conclusion

Dans ce chapitre, ont été présentés les outils et la méthodologie que nous avons utilisés pour traiter des problèmes qui se sont posés à nous lors de la construction d'une chaîne de traitement pour l'analyse 3D par stéréoscopie : calibration, segmentation, reconstruction, recalage et modélisation 3D.

Dans cette méthodologie, le problème doit être analysé sous les aspects duaux de l'estimation et de l'appariement, où réside souvent la difficulté principale des problèmes de vision par ordinateur. Le problème peut alors être classé, soit comme un problème d'estimation simple avec appariement, d'estimation globale sans appariement, ou d'estimation couplée avec la mise en correspondance.

Chaque problème a ses particularités, notamment le cadre d'application, le type des données, la complexité et la robustesse recherchée, qui conduisent souvent à développer des réponses spécifiques. Néanmoins, nous avons traité selon notre approche les sujets énumérés précédemment, et nous allons donc présenter, dans les chapitres suivants, un ensemble de méthodes, qui mises bout à bout, forment, avec l'aide d'autres algorithmes, un processus de modélisation 3D semi-automatique par analyse de paires d'images stéréoscopiques.

Chapitre 2

Calibration précise de caméra par estimation des moments ‘image’

2.1 Introduction

Dans un processus de vision, il est utile de connaître la physique de l’instrument qui fournit les données de base : les images. Cette connaissance permet en particulier de calculer la position exacte des éléments de la scène à partir d’images. La calibration consiste à obtenir les paramètres du modèle de la caméra. En toute généralité, les caractéristiques mécaniques, optiques, photométriques et électroniques peuvent être modélisées, tout dépend de l’utilisation. Par exemple, pour certaines tâches de positionnement relatif, l’auto-calibration suffit [Fau92, Moh93]. Puisque l’application envisagée est la modélisation 3D avec son aspect photométrique, la géométrie et la radiométrie de notre caméra sont calibrées. On se restreint, dans ce chapitre, aux aspects géométriques de la calibration : digitalisation de l’image, distorsions géométriques dues à l’optique de l’objectif, localisation dans l’espace de la caméra. La calibration radiométrique est traitée en annexe A.

Les modèles considérés se fondent alors sur le modèle sténopé amélioré en tenant compte de différents types de distorsions. Le modèle permet d’explicitier la transformation entre un objet de l’espace et son image. Comme les paramètres intrinsèques du modèle de caméra sont interdépendants (par exemple, le centre se déplace quand la focale change [LT88]), il est préférable d’estimer en même temps tous les paramètres du modèle sur une vue, comme dans la méthode proposée dans [FT86].

Nous avons constaté dans le chapitre précédent, combien l’estimation d’une transformation perspective est délicate lorsque l’on ne dispose pas des appariements entre un élément 3D et son image. Nous utilisons donc une mire qui par un prétraitement s’appuyant sur ses caractéristiques propres permet d’obtenir la mise en correspondance. Une fois le modèle choisi, il reste alors à déterminer le type de données à utiliser et le problème se réduit à une estimation selon un critère d’adéquation entre ce modèle et ces données.

Les données sont, le plus souvent, des points. Quelques méthodes utilisent d’autres types d’objets géométriques comme les lignes [CT90, Vai90], ou comme les ellipses [HR92, RZMF92, Ma93b]. Avec les points, selon la modélisation et le critère d’adéquation, le problème peut être formulé linéairement ou non, avec ou sans contraintes. Ceci conduit à des

méthodes analytiques (Annexe B) [Tsa86, FT86, PHYT95], ou à des algorithmes itératifs [CR89].

Pour avoir une estimation précise des paramètres, il faut contrôler un minimum l’environnement de la phase de calibration. Ainsi, une mire contrôlée géométriquement permet de disposer de données tridimensionnelles très précises. Les conditions d’éclairage doivent être satisfaisantes pour que les données extraites de l’image soient suffisamment bonnes. Un grand nombre de données est généralement considéré pour améliorer l’estimation; néanmoins, la précision de celles-ci est directement liée à celle des mesures. Comme la précision de la méthode d’extraction des données est aussi importante que celle de l’algorithme de calibration, une bonne approche du problème nécessite donc de concevoir la technique de mesure des données avec celle d’estimation du modèle.

Or nous avons vu que l’utilisation des moments d’une forme permet d’obtenir des informations précises sur celle-ci. Ainsi nous montrons que les moments géométriques d’une forme dans une image sont calculables avec précision, robustesse et simplicité, indépendamment de l’éclairage, si l’on utilise l’information d’intensité [Bey92, AATRM89]. Après les quelques rappels et notations de la partie 2.2, la précision théorique de notre méthode de mesure des moments, fondée sur les niveaux de gris, est obtenue, de même que les conditions dans lesquelles ces mesures doivent être faites (partie 2.3). Nous avons ensuite, dans la partie 2.4, relié la description d’un disque elliptique de l’espace aux moments géométriques de son image. Ces équations permettent la conception d’une méthode de calibration fondée non pas sur la correspondance entre points mais sur l’utilisation des moments d’une ellipse. La méthode de résolution itérative employée est décrite dans la partie 2.5. Enfin, nous présentons la validation de la méthode sur des images réelles et de synthèse (partie 2.6).

2.2 Modélisation de la caméra

Une caméra réalise une transformation des points de l’espace réel pour produire une image. On suppose généralement que la transformation réalisée est une simple projection centrale par rapport au centre de la caméra : c’est le modèle sténopé. Il peut être complété par la modélisation des diverses distorsions qui entachent généralement l’optique de la caméra. Voici un récapitulatif des modèles de caméra les plus utilisés.

2.2.1 Le modèle sténopé

Dans le modèle sténopé, l’optique est modélisée par une projection centrale sur le plan rétinien, de centre C (figure 2.1). Un point X de l’espace est repéré par rapport au référentiel absolu \mathcal{R}_{absolu} . Les paramètres extrinsèques de la caméra sont la rotation $R(p_1, p_2, p_3)$ et la translation $T(t_x, t_y, t_z)$ qui correspondent au déplacement entre les repères \mathcal{R}_{absolu} et $\mathcal{R}_{caméra}$.

La discrétisation est supposée s’effectuer sur une rétine rectangulaire avec un pas d’échantillonnage vertical $\frac{1}{a_v}$ et un pas horizontal $\frac{1}{a_u}$. Les paramètres intrinsèques sont les coordonnées (u_0, v_0) et l’inverse a_u et a_v des pas d’échantillonnage. (u_0, v_0) est la position dans l’image de l’intersection axe de visée-rétine. La relation entre les coordonnées (x_{ret}, y_{ret}) d’un point

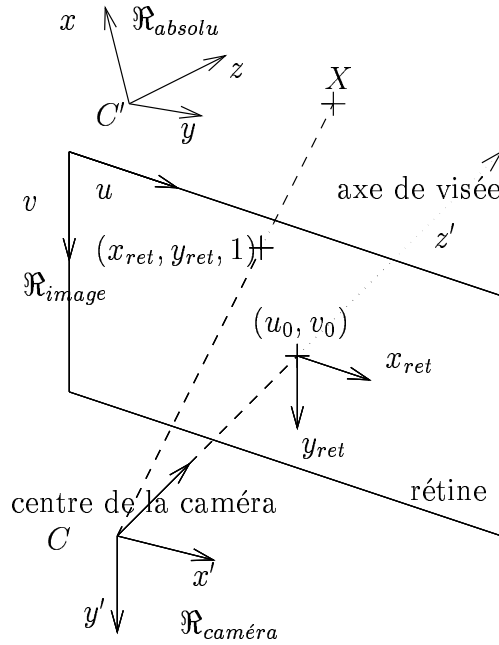


FIG. 2.1 - Le modèle sténopé de la caméra.

de la rétine et celles (u, v) dans \mathfrak{R}_{image} est donc :

$$\left. \begin{aligned} u &= u_0 + a_u x_{ret} \\ v &= v_0 + a_v y_{ret} \end{aligned} \right\} \quad (2.1)$$

En composant dans l'ordre - le déplacement rigide, la projection centrale et la numérisation (2.1) -, on obtient la relation entre un point de l'espace et son image. Cette relation s'exprime simplement en utilisant les coordonnées homogènes :

$$\left. \begin{aligned} u &= \frac{L_1 \mathcal{X}}{L_3 \mathcal{X}} \\ v &= \frac{L_2 \mathcal{X}}{L_3 \mathcal{X}} \end{aligned} \right\} \quad \text{ou} \quad U = \begin{pmatrix} su \\ sv \\ s \end{pmatrix} = \begin{pmatrix} L_1 \\ L_2 \\ L_3 \end{pmatrix} \mathcal{X} \quad (2.2)$$

La transformation complète peut donc être représentée par une matrice perspective 3x4, notée \mathcal{L} .

$$\mathcal{L} = \begin{pmatrix} L_1 \\ L_2 \\ L_3 \end{pmatrix} = \begin{pmatrix} a_u & 0 & u_0 & 0 \\ 0 & a_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} R & T \\ 0 & 1 \end{pmatrix} \quad (2.3)$$

On remarque qu'une matrice \mathcal{L} doit vérifier deux contraintes pour être compatible avec son expression en fonction des 10 paramètres indépendants R, T, u_0, v_0, a_u, a_v (2.3). Si on les oublie, la matrice est définie à un facteur d'échelle près. Dans ce cas, le modèle sténopé ne respecte plus l'orthogonalité des axes de la rétine.

2.2.2 Modèles avec distorsions

La modélisation précédente suppose que l'optique est parfaite. Les aberrations géométriques décrivent l'écart entre les positions observées et idéales de l'image d'un point de

l'espace [Pér93]. Or la calibration peut estimer uniquement les distorsions puisque celles-ci provoquent des déformations indépendamment de la direction du rayon lumineux 'source'. Seuls les premiers termes du développement limité de cet écart sont retenus. Trois types de distorsions sont plus particulièrement considérés [Bey92, WCH90]:

- La distorsion radiale, de paramètre α , est généralement la plus importante [Bey92]:

$$\left. \begin{aligned} x_d &= x_{ret} + \alpha x_{ret}(x_{ret}^2 + y_{ret}^2) \\ y_d &= y_{ret} + \alpha y_{ret}(x_{ret}^2 + y_{ret}^2) \end{aligned} \right\} \quad (2.4)$$

- Le décentrage des lentilles par rapport à l'axe provoque une distorsion décrite par la relation suivante [WCH90]:

$$\left. \begin{aligned} x_d &= x_{ret} + \beta(3x_{ret}^2 + y_{ret}^2) + 2\gamma x_{ret}y_{ret} \\ y_d &= y_{ret} + \gamma(x_{ret}^2 + 3y_{ret}^2) + 2\beta x_{ret}y_{ret} \end{aligned} \right\} \quad (2.5)$$

- Enfin, il résulte une distorsion des petites erreurs de parallélisme entre lentilles. La relation qui la décrit est [WCH90]:

$$\left. \begin{aligned} x_d &= x_{ret} + \delta(x_{ret}^2 + y_{ret}^2) \\ y_d &= y_{ret} + \delta(x_{ret}^2 + y_{ret}^2) \end{aligned} \right\} \quad (2.6)$$

La prise en compte des distorsions conduit à une relation non linéaire entre un point de l'espace et son image. En effet, pour obtenir la relation qui lie un point à son image, il faut composer dans l'ordre le déplacement rigide, la projection centrale, les distorsions (2.4), (2.5) ou (2.6), et la numérisation (2.1).

2.3 Mesures dans l'image

Pour faire une bonne estimation des paramètres du modèle de caméra, du centre de l'image en particulier, il est nécessaire d'avoir des données précises, notamment sur l'image, où la localisation doit être faite à au moins $1/10^{\text{ème}}$ de pixel près. Les caractéristiques géométriques d'une région de l'image comme l'aire, le barycentre et les moments d'inertie, sont faciles à mesurer avec une grande précision si on utilise les niveaux de gris. Sous certaines hypothèses, nous avons établi la précision théorique atteinte par notre méthode.

2.3.1 Principe

Un capteur CCD a une remarquable linéarité (les défauts de linéarité relatifs sont inférieurs à 10^{-3}) sur une large plage de fonctionnement. La réponse d'une cellule s'avère donc proportionnelle à l'éclairement reçu par sa surface [Rag89]. Ainsi, si la caméra est pointée, avec un éclairage ambiant, vers un plan noir sur lequel une forme blanche est peinte, la valeur de chaque pixel de l'image obtenue est proportionnelle à l'aire de la surface blanche vue par le pixel. Or, connaître l'aire de l'intersection entre chaque pixel et l'image de la surface blanche est utile pour calculer une valeur approchée de ses caractéristiques géométriques.

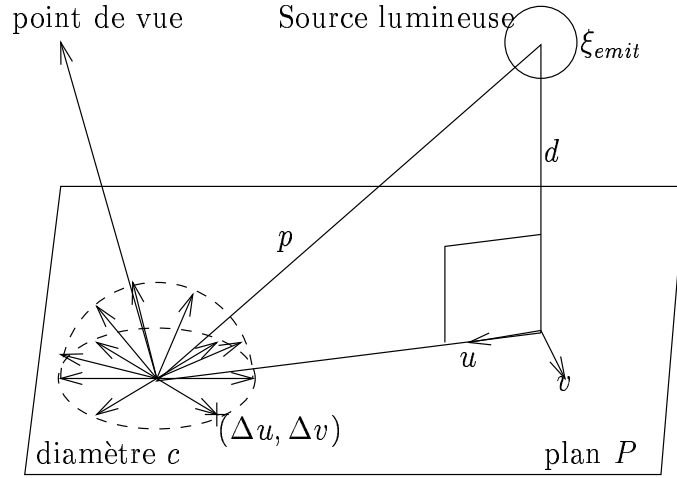


FIG. 2.2 - Réflexion diffuse d'une surface lambertienne.

En particulier, l'utilisation des niveaux de gris permet d'obtenir le barycentre de la surface avec une précision supérieure au pixel.

Cette situation est idéale, mais il est possible de s'en approcher par un traitement approprié de l'image, si l'éclairage et la mire sont bien conditionnés. En effet, se placer dans des conditions telles que les gradients d'intensité soient linéaires n'est pas difficile. La surface de la mire doit être peu spéculaire pour être supposée à photométrie lambertienne, dans de bonnes conditions d'illumination. On suppose en particulier que l'éclairage est ambiant ou qu'il provient de sources éloignées. Dans ces conditions, les pentes des gradients lumineux peuvent être corrigées sur l'image de la mire, pour se ramener à la situation idéale.

2.3.2 Correction des gradients d'intensité

Dans l'hypothèse lambertienne, une surface rayonne de façon diffuse une fraction de la lumière reçue. En chaque point du plan P (figure 2.2), l'intensité émise dans une direction quelconque est :

$$I = \frac{\rho_{reflex} \xi_{emit} d}{p^2} \frac{d}{p} = \frac{\rho_{reflex} \xi_{emit} d}{p^3} = \frac{\rho_{reflex} \xi_{emit} d}{(d^2 + u^2 + v^2)^{\frac{3}{2}}}$$

où ρ_{reflex} est le coefficient de réflexion diffuse, et ξ_{emit} l'émissance de la source.

Si l'on approxime par un plan l'intensité réémise $I(u, v)$, au voisinage d'un point (u', v') , la fonction d'erreur relative au second ordre est :

$$e_{photo}(\Delta u, \Delta v) \simeq \frac{\Delta^2 I(u, v)}{I(u, v)} = \frac{\partial^2 I(u, v)}{I(u, v) \partial u^2} \Delta^2 u + \frac{\partial^2 I(u, v)}{I(u, v) \partial v^2} \Delta^2 v + \frac{\partial^2 I(u, v)}{I(u, v) \partial u \partial v} \Delta u \Delta v$$

avec :

$$\frac{\partial^2 I}{\partial u \partial v} = \frac{15uv \rho_{reflex} \xi_{emit} d}{(d^2 + u^2 + v^2)^{\frac{7}{2}}}$$

$$\frac{\partial^2 I}{\partial u^2} = \frac{3(4u^2 - v^2 - d^2)\rho_{reflex}\xi_{emil}d}{(d^2 + u^2 + v^2)^{\frac{7}{2}}}$$

$$\frac{\partial^2 I}{\partial v^2} = \frac{3(-u^2 + 4v^2 - d^2)\rho_{reflex}\xi_{emil}d}{(d^2 + u^2 + v^2)^{\frac{7}{2}}}$$

Du fait de la symétrie circulaire, on peut se placer dans le repère de la facette qui annule v . Après développement :

$$e_{photo}(\Delta u, \Delta v) = \frac{3}{p^4}((4p^2 - 5d^2)\Delta^2 v - p^2 \Delta^2 u)$$

L'erreur commise sur un disque de diamètre c se majore, à partir de la formule précédente, par $12(\frac{c}{p})^2 = 3(\frac{c}{p})^2$, où p est l'éloignement du centre de la facette à la source lumineuse. Pour que l'erreur soit inférieure à un niveau de gris sur une image 8 bits, il faut que c soit 30 fois plus petit que p . L'approximation au premier ordre des gradients lumineux est donc justifiée dans nos expériences, car les facettes dessinées sur la mire ont un côté de 5 cm avec une source placée à 1.5 m.

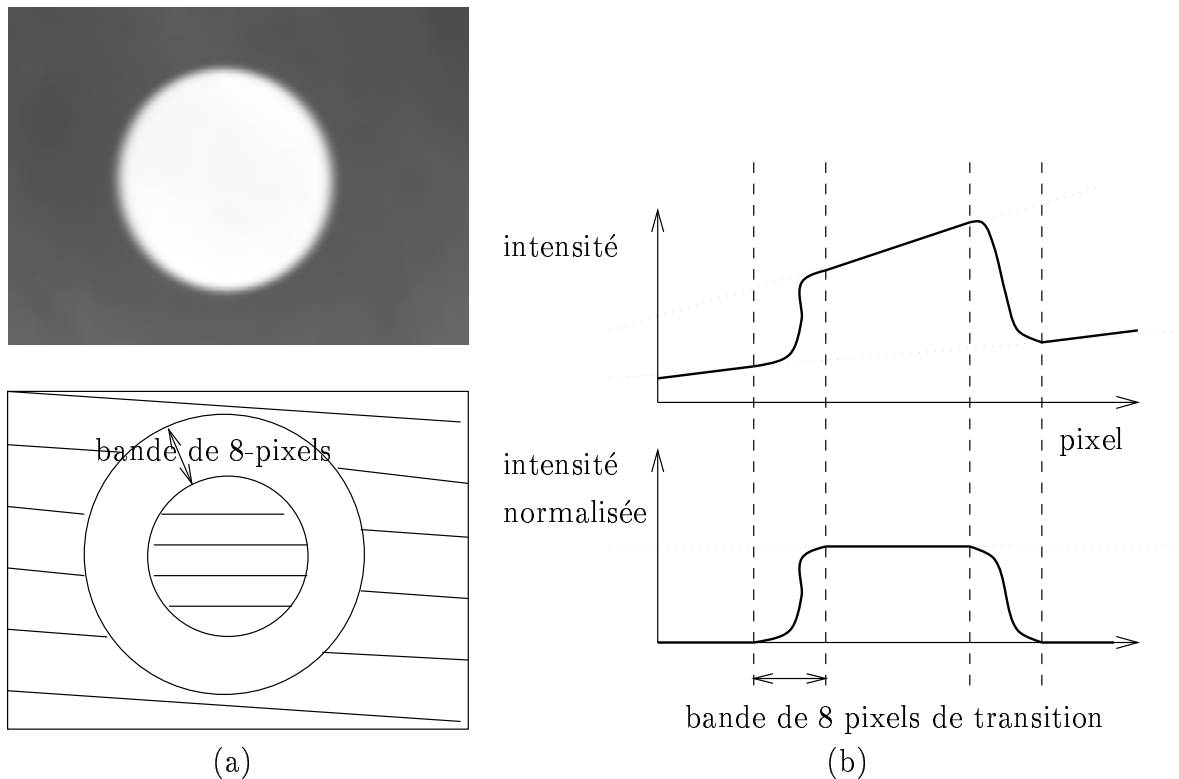


FIG. 2.3 - (a) limite des zones claire et sombre, (b) correction de la variation linéaire de l'intensité pour obtenir l'image normalisée.

En conséquence, dans nos conditions expérimentales, l'intensité de la région 'image' d'une surface lambertienne varie linéairement en fonction des coordonnées des pixels. Le plan qui

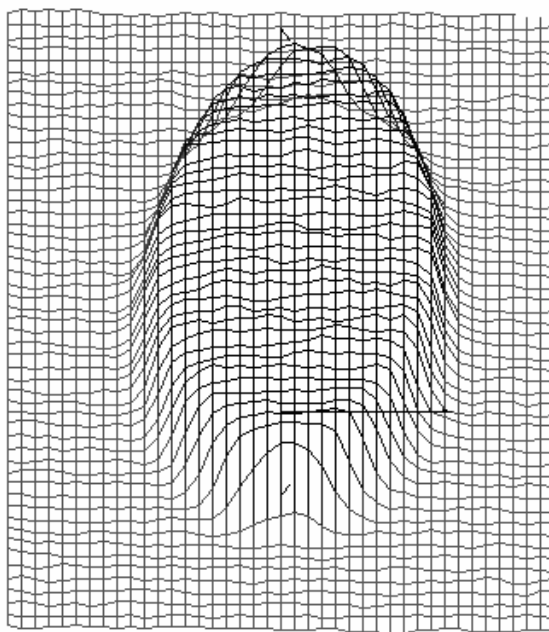


FIG. 2.4 - *Distribution réelle de l'intensité d'un disque blanc sur fond noir.*

approxime le mieux la distribution d'intensité de ces régions est calculé par une méthode aux moindres carrés. Dans le cas où une surface claire est observée sur un fond sombre, le calcul est réalisé sur le fond $I_{fond}(u, v)$ et sur la région claire $I_{forme}(u, v)$ qui sont définis dans la figure 2.3. Ces deux équations suffisent pour normaliser l'image entre 0 et 1 pour chaque pixel (u, v) :

$$\Xi(u, v) = \frac{I(u, v) - I_{fond}(u, v)}{I_{forme}(u, v) - I_{fond}(u, v)}$$

La valeur 0 indique que le pixel de l'image normalisée ne voit pas la surface claire, alors que la valeur 1 indique au contraire qu'il s'intersekte en totalité avec l'image de la surface claire. On obtient une image dont la valeur d'un pixel donne le pourcentage de son aire couverte par la projection de la surface claire. L'image normalisée est donc celle qui serait observée si l'éclairage était à l'infini.

Sur la figure 2.4, la rupture d'intensité au niveau du contour n'est pas brutale, mais au contraire assez douce. Ce lissage est provoqué par différents phénomènes physiques : la réponse électronique du capteur, l'effet de profondeur, les aberrations et la diffusion de l'optique, les interactions électroniques entre pixels. En particulier, avec notre caméra, un étalement de huit pixels du contour se produit pour une mise au point approximative sur l'objet.

2.3.3 Mesures des moments géométriques

La réponse d'une cellule CCD est proportionnelle à l'intensité reçue sur sa surface. En conséquence, pour une distribution d'intensité constante sur la surface du pixel (u, v) , où u et v sont les coordonnées de son coin supérieur gauche, de valeur $\Xi(u, v)$, la contribution au moment d'ordre (p, q) est :

$$M_{p,q}^{pixel} = \Xi(u, v) \int_{aire\ pixel(u,v)} x^p y^q dx dy = \Xi(u, v) \left(\frac{(u+1)^{p+1} - u^{p+1}}{p+1} \right) \left(\frac{(v+1)^{q+1} - v^{q+1}}{q+1} \right)$$

En pondérant le moment de chaque pixel par la valeur $\Xi(u, v)$ de l'image normalisée, les caractéristiques géométriques de la forme se calculent avec précision. Le moment d'ordre (p, q) est alors donné par l'équation suivante :

$$M_{p,q} = \sum_{u,v} \Xi(u, v) \left(\frac{(u+1)^{p+1} - u^{p+1}}{p+1} \right) \left(\frac{(v+1)^{q+1} - v^{q+1}}{q+1} \right) \quad (2.7)$$

L'aire A , le barycentre G et la matrice inertie surfacique I de la région 'image' sont alors respectivement donnés par :

$$\left. \begin{aligned} A &= \sum_{u,v} \Xi(u, v) \\ G &= \frac{1}{A} \sum_{u,v} \Xi(u, v) \begin{pmatrix} u + \frac{1}{2} \\ v + \frac{1}{2} \end{pmatrix} \\ I &= \frac{1}{A} \sum_{u,v} \Xi(u, v) \begin{pmatrix} u^2 + u + \frac{1}{3} & (u + \frac{1}{2})(v + \frac{1}{2}) \\ (u + \frac{1}{2})(v + \frac{1}{2}) & v^2 + v + \frac{1}{3} \end{pmatrix} \end{aligned} \right\} \quad (2.8)$$

Pour estimer les caractéristiques géométriques, il suffit de calculer une somme pondérée avec les niveaux de gris normalisés. L'obtention de ceux-ci ne présente pas non plus de difficulté sur une image de mire. En effet, le calcul et la correction des gradients d'intensité sont simples quand on a pu délimiter avec certitude, grâce à un détecteur de contours [Der87], une zone 'intérieure' à chaque région et une 'extérieure' voisine. Avec notre caméra, une bande de 3 pixels de chaque côté du contour est exclue.

2.3.4 Précision du calcul des moments

L'analyse des erreurs sur les moments doit tenir compte de quatre sources d'erreur dues :

- : au bruit radiométrique d'une cellule CCD : e_{bruit} ,
- : à l'approximation photométrique : e_{photo} (voir section 2.3.2),
- : à la discrétisation de l'intensité d'un pixel de l'image : e_{digit} ,
- : à la discrétisation en pixels de l'image : e_{local} . La figure 2.5 montre des exemples de surfaces qui ne peuvent pas être différenciées lorsqu'elles sont intégrées sur un pixel.

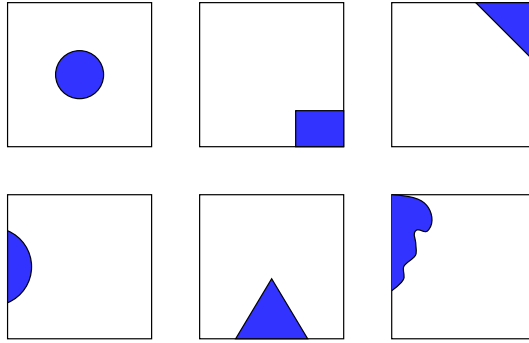


FIG. 2.5 - Pour chaque pixel, l'aire de la forme frisée est le même, mais leurs moments géométriques d'ordre 1 et 2 sont différents. Cette indétermination au niveau sous-pixel induit l'erreur e_{local} .

Ces sources d'erreurs sont supposées indépendantes. Nous allons maintenant trouver une borne de l'écart type de chacune de ces erreurs :

e_{bruit} : Le bruit en intensité présent sur chaque pixel est supposé gaussien centré de variance σ^2 . Alors, l'écart type de l'erreur produite par le bruit sur l'aire A est $\sqrt{n_{région}}\sigma$, où $n_{région}$ est le nombre de pixels de la région. Pour les composantes du barycentre G , l'écart type est approximativement $\sigma\sqrt{\frac{M_{2,0}}{A^2}}$ et $\sigma\sqrt{\frac{M_{0,2}}{A^2}}$, car d'après l'équation (2.8) :

$$var(G) \simeq \sum_{u,v} \frac{1}{A^2} \left(\begin{matrix} (u + \frac{1}{2})^2 \\ (v + \frac{1}{2})^2 \end{matrix} \right) var(\Xi(u, v))$$

L'écart type, sur les éléments I_{uu} , I_{uv} et I_{vv} de la matrice d'inertie, a une forme analogue en remplaçant les moments par $M_{4,0}$, $M_{2,2}$ et $M_{0,4}$ respectivement. Une forme aplatie provoque donc des erreurs non équivalentes sur chaque composante. Avec un disque de rayon r , l'écart type sur le barycentre est inférieur à σ , alors que, pour l'inertie, il est majorée par σr . En conséquence, un bruit dont l'écart type est de quelques niveaux de gris produit une erreur inférieure à 0.01 pixel sur le barycentre, pour des images 8 bits très contrastées.

e_{photo} : Dans nos expériences, comme cela est décrit précédemment dans la section 2.3.2, l'erreur induite par l'approximation photométrique est inférieure à un niveau de gris pour chaque pixel, sur des images contrastées. Elle est beaucoup plus petite que celle provoquée par le bruit e_{bruit} .

e_{digit} : L'erreur de quantification de l'intensité peut être décrite comme une variable aléatoire uniforme centrée de variance $\sigma^2 = \frac{1}{12I_{max}^2}$ sur l'intensité de chaque pixel, où I_{max} est la hauteur du saut d'intensité entre la région claire et la région sombre [Der87]. Pour des images 8 bits, avec un contraste maximum ($I_{max} = 256$), l'écart type est de l'ordre de 10^{-3} . Comme pour e_{photo} , e_{digit} est petite.

e_{local} : L'erreur provoquée par la discrétisation spatiale est nulle sur A , puisque l'aire n'est pas localisée. Dans la partie 2.8, en dimension un, nous montrons que, indépendamment de la réponse de la caméra, une borne de l'erreur de localisation est de $\frac{1}{8A}$ sur la position du barycentre, et de $\frac{1}{4}$ sur I . À partir de ces résultats, un passage à la dimension deux peut s'effectuer. Soit $n_{contour}$ le nombre de pixels du contour de la région considérée, la moitié des pixels du contour peuvent biaiser la mesure dans le même sens, donc l'erreur e_{local} est majorée par $\frac{n_{contour}}{2} \frac{1}{8A} = \frac{n_{contour}}{16A}$. On sait que la forme qui minimise le rapport de la longueur du contour avec l'aire est un disque. Soit r son rayon, la borne de l'erreur e_{local} est alors majorée par $\frac{1}{8r}$. Il suffit donc d'un cercle de rayon 13 pixels pour avoir une précision de l'ordre du centième de pixel sur la position du barycentre.

L'écart type théorique total obtenu est donc, pour un cercle de rayon r plus grand que 13 pixels, inférieur à $5/100^{\text{ème}}$ de pixel sur le barycentre, si le bruit est inférieur à 10 niveaux de gris. Une aussi grande précision ne peut pas être obtenue uniquement par la moyenne du contour de la région (voir section 2.6.1). En pratique, il est donc possible d'atteindre une erreur inférieure à $1/10^{\text{ème}}$ de pixel, sur des images réelles de formes arrondies et compactes, si l'environnement est bien conditionné.

2.3.5 Conclusion

Nous avons prouvé qu'il est possible de calculer les caractéristiques géométriques d'une forme plane vue dans une image avec une bonne précision, si elle est la projection d'une facette lambertienne de taille petite par rapport à la distance de la source lumineuse, et si sa taille est supérieure à une vingtaine de pixels. Une forme qui optimise bien l'erreur de localisation sur le barycentre est une forme arrondie et compacte.

La méthode d'estimation proposée a deux propriétés remarquables : elle est assez robuste au bruit photonique et au lissage par un filtre symétrique quelconque de l'image (en particulier pour les coordonnées du barycentre). Cette propriété donne un avantage important dans la pratique, puisque la mise au point est rarement réalisée sur la mire. Enfin, la méthode est relativement simple à mettre en place.

2.4 Transformation perspective d'une ellipse

Les caractéristiques géométriques d'une région de l'image peuvent être mesurées avec précision. Mais pour pouvoir utiliser ces données dans la calibration, il faut être capable de les relier à la description 3D de la surface observée. L'ellipse présente un grand intérêt de part ses nombreuses propriétés vis-à-vis de la transformation perspective (ou transformation conique) [Her92]. Cependant, comment une ellipse se transforme-t-elle par le modèle sténopé? La caméra doit voir l'ellipse entièrement et son image rester elliptique. De fait, les conditions de la calibration permettent ces restrictions, mais la généralisation de ce qui suit aux coniques est aisée. À partir des différentes représentations d'une ellipse dans le plan, le lien entre l'équation d'une ellipse et ses moments géométriques d'ordre inférieur ou égal à deux sera établi. Ces équations permettent de faire la calibration, à partir du modèle de la mire et des données 2D extraites des images.

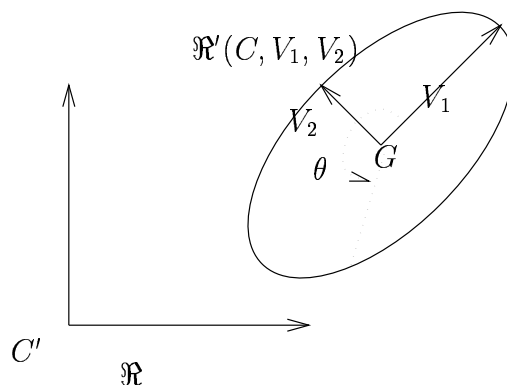


FIG. 2.6 - Repère absolu \mathfrak{R} et canonique $\mathfrak{R}'(G, V_1, V_2)$ d'une ellipse.

2.4.1 Caractérisations d'une ellipse

Une ellipse peut être caractérisée dans le plan par son centre G et les vecteurs demi-grand axe V_1 et demi-petit axe V_2 donnés dans le repère orthonormal \mathfrak{R} (figure 2.6). Les points de l'ellipse sont alors décrits par :

$$X = G + \cos(\theta)V_1 + \sin(\theta)V_2 \quad \theta \in [0, 2\pi]$$

Cette équation se traduit en coordonnées homogènes par la relation suivante :

$$\mathcal{X} = \mathcal{G} + \cos(\theta)\mathcal{V}_1 + \sin(\theta)\mathcal{V}_2 \quad \theta \in [0, 2\pi]$$

où $\mathcal{G} = \begin{pmatrix} G \\ 1 \end{pmatrix}$, $\mathcal{V}_1 = \begin{pmatrix} V_1 \\ 0 \end{pmatrix}$ et $\mathcal{V}_2 = \begin{pmatrix} V_2 \\ 0 \end{pmatrix}$. Soit \mathcal{B} la matrice 3×3 constituée, dans l'ordre, des vecteurs \mathcal{V}_1 , \mathcal{V}_2 et \mathcal{G} . Cette matrice est une première caractérisation de l'ellipse dans le repère \mathfrak{R} .

L'équation dans un repère donné représente autrement l'ellipse dans le plan. L'équation implicite de celle-ci s'écrit en coordonnées homogènes sous la forme suivante :

$$\mathcal{X}'^t \mathcal{E} \mathcal{X}' = 0$$

\mathcal{E} est une matrice symétrique 3×3 de signature $(\epsilon, \epsilon, -\epsilon)$ avec $\epsilon = \pm 1$. La matrice \mathcal{E} dans le repère \mathfrak{R} est une autre caractérisation de l'ellipse dans le plan. Cette matrice est définie à un facteur près.

Il existe un lien entre la représentation par \mathcal{B} et celle par \mathcal{E} . Pour l'expliciter, on se place dans le repère orthogonal \mathfrak{R}' d'origine G et de vecteurs V_1 et V_2 (figure 2.6). Dans ce repère, l'équation implicite est réduite à son expression canonique :

$$\mathcal{X}'^t \mathcal{E}' \mathcal{X}' = \mathcal{X}'^t \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathcal{X}' = 0 \quad (2.9)$$

La matrice \mathcal{E}' associée est alors particulièrement simple.

La transformation entre les deux repères \mathfrak{R} et \mathfrak{R}' s'explique en fonction de la matrice \mathcal{B} . Ainsi, un point de coordonnées \mathcal{X}' dans le repère \mathfrak{R}' a pour nouvelles coordonnées $\mathcal{X} = \mathcal{B}\mathcal{X}'$ dans le repère \mathfrak{R} . En réalisant le changement de repère dans l'équation implicite réduite (2.9), la relation entre la matrice \mathcal{E} et \mathcal{B} se déduit, par identification :

$$\mathcal{E} = \mathcal{B}^{t-1} \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathcal{B}^{-1}$$

Ce qui peut s'écrire, après inversion, puisque \mathcal{E} est de rang 3 :

$$\mathcal{E}^{-1} = \mathcal{B} \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathcal{B}^t \quad (2.10)$$

Alors, l'inverse de la matrice de la forme quadratique se décompose sous la forme suivante :

$$\mathcal{E}^{-1} = \begin{pmatrix} GG^t - V_1V_1^t - V_2V_2^t & G \\ G^t & 1 \end{pmatrix} \quad (2.11)$$

On remarque que la dernière colonne de \mathcal{E}^{-1} (ou la dernière ligne), quelque soit le repère choisi, est égale, à un facteur près, au vecteur homogène \mathcal{G} qui donne la position du centre de l'ellipse dans ce repère.

2.4.2 Matrice d'inertie homogène

La matrice d'inertie homogène du disque elliptique est définie par :

$$\mathcal{I} = \int_{\text{disque elliptique}} \mathcal{X}\mathcal{X}^t dS = A \begin{pmatrix} I & G \\ G^t & 1 \end{pmatrix} \quad (2.12)$$

où I , G et A sont la matrice d'inertie, le barycentre et l'aire du disque respectivement. Pour un disque de rayon unité dans le repère barycentrique, le calcul donne :

$$\mathcal{I}' = \begin{pmatrix} \frac{\pi}{4} & 0 & 0 \\ 0 & \frac{\pi}{4} & 0 \\ 0 & 0 & \pi \end{pmatrix}$$

Comme dans la section précédente, en passant de \mathfrak{R}' au repère \mathfrak{R} ($\mathcal{X} = \mathcal{B}\mathcal{X}'$), on en déduit par substitution dans (2.12) que la matrice d'inertie homogène de l'ellipse, dans le cas général, s'écrit :

$$\mathcal{I} = \det(\mathcal{B})\mathcal{B}\mathcal{I}'\mathcal{B}^t$$

Cette relation est vraie uniquement si la transformation \mathcal{B} est affine. Après un développement semblable à celui de l'équation (2.11), on obtient l'expression suivante en introduisant \mathcal{I}' calculé pour le disque unité :

$$\mathcal{I} = \pi \det(V_1 \ V_2) \begin{pmatrix} \frac{1}{4}(V_1V_1^t + V_2V_2^t) + GG^t & G \\ G^t & 1 \end{pmatrix}$$

L'aire de l'ellipse A s'avère bien être $\pi \det(V_1 \ V_2) = \pi \|V_1\| \|V_2\|$. La relation entre la matrice d'inertie homogène \mathcal{I} et \mathcal{E} se déduit, par comparaison avec (2.11) :

$$\frac{\mathcal{I}}{A} = \frac{1}{4} (5\mathcal{G}\mathcal{G}^t - \mathcal{E}^{-1}) \quad (2.13)$$

Pour cette relation, \mathcal{E} est supposée normalisée, c'est-à-dire que le dernier élément de la matrice est 1.

Comme nous l'avons vu dans la section précédente, la dernière colonne de \mathcal{E}^{-1} est égale à \mathcal{G} , en conséquence, il suffit d'inverser la matrice de l'équation implicite homogène pour connaître la matrice d'inertie homogène de cette ellipse. Réciproquement, la seule connaissance des caractéristiques géométriques G et I permet de déterminer complètement l'ellipse. Une ellipse est donc caractérisée par ses seuls moments d'ordre inférieur ou égal à deux.

2.4.3 projection d'une ellipse

Les caractéristiques d'une ellipse E_{image} , qui est la projection perspective d'une ellipse E_{espace} , peuvent être obtenues à partir des formules précédentes. Soient \mathcal{V}_1 , \mathcal{V}_2 et \mathcal{G} les trois vecteurs (demi-grand axe, demi-petit axe et centre) qui caractérisent l'ellipse E_{espace} dans le repère absolu de l'espace 3D. Soit $P_{ellipse}$ le plan qui contient l'ellipse, et X un point de $P_{ellipse}$ de coordonnées $(x, y, 1)$ dans le repère orthogonal où l'équation de l'ellipse E_{espace} est canonique (2.9). Les coordonnées de homogènes de X dans le repère de référence de l'espace sont :

$$(\mathcal{V}_1 \ \mathcal{V}_2 \ \mathcal{G}) \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

L'image de ce point est donc, d'après (2.2) :

$$U = \mathcal{L} (\mathcal{V}_1 \ \mathcal{V}_2 \ \mathcal{G}) \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \mathcal{L}\mathcal{B} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Nous introduisons donc une matrice \mathcal{B} comme précédemment, sauf que à présent, c'est une matrice 4×3 au lieu de 3×3 . La relation entre le plan de l'ellipse $P_{ellipse}$ et le plan 'image' est donc une transformation homogène linéaire de la matrice 3×3 $\mathcal{L}\mathcal{B}$. En conséquence, puisque l'équation (2.10) reste vraie pour une transformation homogène quelconque, l'équation devient pour une ellipse 3D définie par \mathcal{B} qui se projette en l'ellipse E_{image} de forme quadratique \mathcal{E} :

$$\mathcal{E}^{-1} = \mathcal{L}\mathcal{B} \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix} (\mathcal{L}\mathcal{B})^t \quad (2.14)$$

À partir de cette équation (2.14), grâce à la remarque faite sur (2.11), nous pouvons déduire le centre de l'ellipse E_{image} en fonction des éléments de la matrice $\mathcal{L}\mathcal{B}$, c'est-à-dire selon les caractéristiques $(\mathcal{V}_1, \mathcal{V}_2, \mathcal{G})$ de l'ellipse E_{espace} et selon les paramètres (L_i) de la transformation

perspective \mathcal{L} . Les coordonnées homogènes $U_g = (s_g u_g \ s_g v_g \ s_g)^t$ du barycentre de l'ellipse image sont donc données, après calcul, par :

$$\left. \begin{aligned} s_g u_g &= (L_1 \mathcal{V}_1) (L_3 \mathcal{V}_1) + (L_1 \mathcal{V}_2) (L_3 \mathcal{V}_2) - (L_1 \mathcal{G}) (L_3 \mathcal{G}) \\ s_g v_g &= (L_2 \mathcal{V}_1) (L_3 \mathcal{V}_1) + (L_2 \mathcal{V}_2) (L_3 \mathcal{V}_2) - (L_2 \mathcal{G}) (L_3 \mathcal{G}) \\ s_g &= (L_3 \mathcal{V}_1)^2 + (L_3 \mathcal{V}_2)^2 - (L_3 \mathcal{G})^2 \end{aligned} \right\} \quad (2.15)$$

Une fois que le centre $\frac{U_g}{s_g} = (u_g \ v_g \ 1)^t$ de l'ellipse 'image' est connu, la matrice d'inertie I s'obtient en fonction de \mathcal{L} et \mathcal{B} grâce au bloc supérieur gauche de \mathcal{E}^{-1} dans l'équation (2.13) :

$$I = \frac{1}{4} \left(5 \frac{1}{s_g^2} U_g U_g^t - \frac{1}{s_g} \begin{pmatrix} e_{uu} & e_{uv} \\ e_{uv} & e_{vv} \end{pmatrix} \right) \quad (2.16)$$

avec

$$\begin{aligned} e_{uu} &= (L_1 \mathcal{V}_1)^2 + (L_1 \mathcal{V}_2)^2 - (L_1 \mathcal{G})^2 \\ e_{vu} &= L_1 \mathcal{V}_1 \ L_2 \mathcal{V}_1 + L_1 \mathcal{V}_2 \ L_2 \mathcal{V}_2 - L_1 \mathcal{G} \ L_2 \mathcal{G} \\ e_{vv} &= (L_2 \mathcal{V}_1)^2 + (L_2 \mathcal{V}_2)^2 - (L_2 \mathcal{G})^2 \end{aligned}$$

2.4.4 Conclusion

Une ellipse est caractérisable par ses moments géométriques d'ordre inférieur ou égal à deux, et le lien entre sa description dans le plan et ses moments est relativement simple. Par l'utilisation des coordonnées homogènes, cette relation a été généralisée à la transformation perspective d'une ellipse de l'espace. Alors, le barycentre et les termes d'inertie s'expriment comme le rapport de polynômes de degré deux en fonction des caractéristiques de l'ellipse 3D et de la transformation perspective.

2.5 Résolution

Une méthode précise de calcul des moments d'une forme dans une image a été élaborée. Quand la forme est un disque elliptique, les relations entre ces moments et l'ellipse 3D ont été explicitées. Ces relations qui lient les données mesurées sur l'image et les données tridimensionnelles de la mire permettent l'estimation des paramètres de la calibration, puisque la mire est supposée de géométrie parfaitement connue.

La mire utilisée est constituée de deux plans métalliques où sont fixés un quadrillage régulier de disques de couleur unie, comme c'est visible dans la figure 2.8. La régularité du quadrillage et le fait que la mire soit toujours dans une position approximativement verticale permet d'ordonner les images des ellipses, puis d'associer chaque ellipse de l'image au disque à surface de la mire, dont elle est la projection.

De par la structure de la mire, l'appariement entre les disques 3D et les ellipses 'images' est donc simple à réaliser.

2.5.1 Critère d'adéquation

Le critère d'adéquation du modèle avec les données est choisi du type moindres carrés. Si les données sont des points U_i^r dans l'image, le critère est basé sur la distance euclidienne.

Ainsi, si l'on désire minimiser la distance en pixel, le critère à choisir est :

$$\min_{\text{paramètres}} \sum_i (u_i^r - u_i)^2 + (v_i^r - v_i)^2$$

Par contre, pour l'erreur réelle sur la rétine, par exemple en millimètres, le critère à minimiser est :

$$\min_{\text{paramètres}} \sum_i \left(\frac{u_i^r - u_i}{a_u} \right)^2 + \left(\frac{v_i^r - v_i}{a_v} \right)^2$$

Ce critère a donné de meilleurs résultats dans nos expérimentations. U_i s'exprime en fonction des paramètres de la calibration et des caractéristiques de la mire. Dans le cas de la correspondance point à point, U_i est donné par l'équation (2.2). Si les données mesurées sont les centres des ellipses images, l'utilisation des équations (2.15) s'avère alors nécessaire.

Il est possible de généraliser ce qui précède à un modèle de caméra autre que sténopé. En effet, du fait de la faiblesse des distorsions géométriques, l'image de l'ellipse peut être considérée comme uniformément distordue. Il suffit alors, pour avoir le lien entre une ellipse 3D et le centre de son image, de composer les formules de distorsions (2.4) (2.5) ou (2.6) avec (2.15).

Pour les termes de la matrice d'inertie, le critère d'adéquation est la distance des trois termes indépendants de la matrice. Avec le modèle sténopé les termes d'inertie fournissent trois contraintes supplémentaires par ellipse (2.16). Mais il semble difficile de les généraliser au cas avec des distorsions.

2.5.2 Méthode itérative

Le problème de la calibration d'une caméra peut s'exprimer, sous forme minimale, comme un problème d'estimation non linéaire de moindres carrés sans contrainte. Pour cela, la rotation doit être représentée sous une forme minimale. Les paramètres (p_1, p_2, p_3) choisis sont ceux d'Euler-Rodrigues, car ils permettent d'écrire les coordonnées de la projection d'un point comme le rapport de deux polynômes de degré trois. On a alors $(p_1, p_2, p_3)^t = \tan(\frac{\theta}{2})N$ où θ est l'angle de la rotation et N la direction normée de son axe.

L'algorithme utilisé pour minimiser le critère est dit de Gauss-Newton [DS83]. Il nécessite le calcul du vecteur des dérivées partielles du critère par rapport aux paramètres de la calibration, ce qui peut être facilement fait avec un calculateur formel. Les valeurs des paramètres initiaux sont obtenues par la méthode de Faugeras et Toscani [FT86]. Ceci assure une convergence rapide de l'algorithme.

2.6 Résultats

La vérification expérimentale d'une méthode de calibration est difficile, car elle demande de mesurer des distances et des orientations avec une précision importante. Une première alternative à cette difficulté est de créer des images de synthèse d'une scène de calibration artificielle où la géométrie et la photométrie sont reproduites. Une autre possibilité est de vérifier la stabilité de l'estimation des paramètres intrinsèques.

2.6.1 Images de synthèse

Précision sur les centres des ellipses

Avec une scène de synthèse, la maîtrise du processus permet de connaître les erreurs faites, notamment sur la mesure des données dans l'image. Pour cela, les mesures sont comparées avec les données obtenues par calcul. Dans le tableau suivant, les erreurs de mesures, sur les barycentres de 280 ellipses de diamètre 40 pixels environ, sont calculées :

erreur	contours	intensité
maximum	1.2 pixel	0.07 pixel
moyenne	0.3 pixel	0.03 pixel
écart type	0.3 pixel	0.02 pixel

TAB. 2.1 - *Erreur d'estimation du barycentre d'une ellipse dans une image suivant que l'on utilise ou pas les niveaux de gris.*

Le tableau 2.1 montre la supériorité de la méthode fondée sur l'intensité des pixels, sur la technique qui utilise les contours (rapport dix en précision). L'ordre de grandeur des erreurs ($5/100^{\text{ème}}$) est respecté par rapport au calcul théorique d'erreur (partie 2.3). Une alternative à notre méthode est d'utiliser un détecteur de contours de précision sous-pixel [OS91, BD94, Dev95].

La méthode de calcul du barycentre fait une moyenne sur un grand nombre de pixels. Elle doit donc être assez résistante au bruit de l'image. La robustesse a été testée et vérifiée sur des images de disques de différents diamètres, bruitées avec un bruit gaussien centré de variance σ^2 . Le tableau 2.2 suivant donne l'écart type de l'erreur provoquée uniquement par le bruit, pour deux diamètres sur une centaine de disques.

diamètre	$\sigma = 2$	$\sigma = 4$	$\sigma = 8$	$\sigma = 16$
10 pixels	0.004	0.01	0.02	0.03
20 pixels	0.0006	0.001	0.003	0.005

TAB. 2.2 - *Erreur induite par un bruit gaussien centré de variance σ sur l'estimation du centre d'une ellipse, suivant diamètre en pixels.*

L'erreur sur le barycentre, due au bruit, diminue en fonction du diamètre du disque. Elle est très inférieure à la majoration prévue (partie 2.3), dès que le rayon est supérieur à une dizaine de pixels.

Corrélation des estimations

Dans la matrice de régression, un coefficient proche de un indique qu'il existe un lien entre deux des paramètres de la calibration. Or, on observe expérimentalement un lien entre

les paramètres t_z , a_u et a_v , de même que pour les groupes de paramètres t_x , u_0 , p_2 et t_y , v_0 , a_v , p_1 , p_3 . Ces liaisons s'expliquent quand on examine les modifications produites sur l'image de la mire par de petites variations des paramètres de la calibration. Certaines combinaisons de ces variations produisent en effet seulement de très petites modifications de l'image. Par exemple, une faible variation de profondeur peut être approximativement compensée par une dilatation de la rétine, d'où la liaison entre les paramètres du groupe t_z , a_u et a_v . Les deux autres groupes de liens s'expliquent par le fait qu'une petite translation de la mire ou une petite rotation (proche d'une translation) peut être compensée par un déplacement du centre de l'image.

L'estimation du centre est souvent délicate, pourquoi? Un développement limité, supposant une variation uniquement sur u_0 et t_x , montre que le critère minimisé ne croit pas quand l'épaisseur de la mire est petite relativement à sa profondeur z_m , si les variations vérifient la relation suivante :

$$\Delta t_x = -\frac{z_m \Delta u_0}{a_u} \quad (2.17)$$

Ceci se traduit aussi géométriquement par l'existence de directions privilégiées où le critère varie peu, au voisinage de son minimum. En conséquence, la qualité de l'estimation est directement fonction du rapport de la profondeur de la mire par sa distance à la caméra. Ce rapport doit être le plus grand possible.

Précision de la calibration

Quand la caméra de synthèse est un sténopé parfait, les paramètres sont tous retrouvés avec une erreur relative de moins de 0.07% (voir partie 2.6.1). Pour initialiser l'algorithme d'estimation, la méthode de Faugeras-Toscani est utilisée. Elle est basée sur la correspondance de points (voir l'annexe B et plus particulièrement la section B.5 pour avoir plus de détails). Avec le type de mire utilisé (figure 2.8), les points mis en correspondance sont le centre des ellipses 3D et 'image'. Malgré cette approximation, la méthode de Faugeras-Toscani donne une estimation avec une erreur relative maximum de 0.7%. En conséquence, si le maximum de précision n'est pas nécessaire, la correspondance des barycentres peut être supposée. Ces résultats ont été obtenus, en calculant l'écart type sur 6 positions assez différentes de la caméra (figure 2.7).

Pour se rapprocher de la réalité, une importante distorsion radiale a été ajoutée à la caméra synthétique ($\alpha = 1.0$). Si la calibration est réalisée avec l'hypothèse du modèle sténopé, les résultats sont fortement perturbés. Et le biais augmente avec l'importance des distorsions. Sur 6 positions de la caméra, le centre de l'image (u_0, v_0) est biaisé en moyenne d'une vingtaine de pixels, comme le montre le tableau suivant :

Le tableau 2.3 donne les biais observés, sur les différents paramètres estimés, pour trois hypothèses de modèles : sténopé, avec décentrage et distorsions radiales (équations (2.4 et 2.5)), avec distorsions radiales (équation (2.5)). Le modèle pertinent est le dernier. On note que l'introduction de paramètres non pertinents (paramètres β et γ du décentrage) induit une moins bonne estimation des autres paramètres. De plus une dispersion 10 fois plus grande des résultats est observée.

	sténopé	décentré+radial	radial
u_0	24 pixel	-0.3 pixel	0.1 pixel
v_0	27 pixel	0.2 pixel	0.01 pixel
a_u	390 p/mm	1 p/mm	0.46 p/mm
a_v	380 p/mm	0.7 p/mm	0.28 p/mm
t_x	-23 mm	0.4 mm	-0.1 mm
t_y	-30 mm	-0.2 mm	-0.02 mm
t_z	491 mm	1.9 mm	1.3 mm
θ	1.6°	0.01°	0.008°
α		-0.002	-0.0008

TAB. 2.3 - *Écarts d'estimation par rapport aux valeurs exactes avec 3 méthodes sur une mire de synthèse et avec une caméra qui présente uniquement des distorsions radiales.*

De même, la méthode point à point avec distorsions donne des résultats biaisés. Le biais est particulièrement sensible sur t_z (voir aussi le paragraphe 2.6.2).

Même pour un objet pris dans de bonnes conditions à une distance de 1 mètre, une erreur d'estimation de 25 pixels du centre de la caméra, comme celle observée en faisant l'hypothèse du modèle sténopé, produit d'après la formule (2.17) un biais de 2.5 cm dans la connaissance de la position absolue de l'objet. Placé à une distance de 10 mètres le biais produit atteint 25 cm. De plus, le tableau précédent confirme expérimentalement l'équation (2.17), puisque les biais sur u_0 et t_x (ou v_0 et t_y) sont proportionnels.

L'erreur induite par la mauvaise estimation du centre produit donc des écarts de localisation par rapport à la caméra lors de la reconstruction qui augmentent avec la distance à la caméra. Par contre comme cela a été remarqué [MM91], les erreurs de positionnement relatif, par rapport à un repère sur la mire, vont être beaucoup plus petites pour des points reconstruits dans le voisinage de la mire, puisque la mire peut être considérée dans un intervalle faible de distance à la caméra. Ceci implique, que si l'on désire faire uniquement de la reconstruction relative, une imprécision importante sur l'estimation des paramètres intrinsèques n'est pas gênante.

Avec des moments d'inertie

Dans le cas où le modèle sténopé est valide, trois contraintes supplémentaires sont données par les trois moments d'inertie de l'image (2.16). L'apport éventuel de l'utilisation de ces contraintes a été testé sur 6 images de synthèse de la mire.

Le tableau 2.4 précédent montre l'erreur relative faite dans l'estimation des paramètres, avec ou sans l'utilisation des termes d'inertie. Sur certains paramètres, l'estimation est améliorée d'un facteur 2 à 3 avec les moments d'inertie. L'apport n'est pas très important, mais il faut remarquer que les moments d'ordre 2 sont plus sensibles aux perturbations que ceux d'ordre inférieur.

	sans inertie	avec inertie
u_0	0.04%	0.04%
v_0	0.04%	0.01%
a_u	0.03%	0.02%
a_v	0.02%	0.02%
t_x	0.07%	0.08%
t_y	0.07%	0.03%
t_z	0.03%	0.02%
θ	0.03%	0.03%

TAB. 2.4 - *Erreur relative d'estimation des paramètres de la caméra lors de la calibration suivant que l'on utilise ou pas les termes d'inertie.*

2.6.2 Images réelles

La mire utilisée est constituée de deux plans métalliques où sont fixés des disques de couleur unie (figure 2.8). La surface est traitée pour être faiblement spéculaire.

Comparaison des méthodes

Les spécifications d'une caméra réelle ne permettent généralement qu'un calcul approximatif des paramètres exacts. Sans référence, il est difficile de tirer beaucoup d'informations de la comparaison des résultats des différentes méthodes de calibration car elles peuvent être biaisées de la même manière. Dans le tableau 2.5 suivant, se trouvent les résultats d'une calibration par trois algorithmes différents: l'algorithme point à point sans distorsion [FT86], l'algorithme itératif point à point avec distorsions radiales [LT88], et notre algorithme avec des ellipses et distorsions radiales. Chaque algorithme travail sur les mêmes données précises à moins de $1/10^{\text{ème}}$ de pixel.

Le résidu moyen, entre les mesures 'images' et leur estimée après minimisation, est aussi indiqué. Ce résidu est pour les images réelles d'environ $1/20^{\text{ème}}$ de pixel avec les points comme avec les ellipses. L'ordre de grandeur du résidu est cohérent avec les estimations d'erreurs précédentes (voir 2.6.1). En effet, si le modèle de caméra est adéquat, ce résidu doit être inférieur ou égal à l'erreur faite sur les barycentres, suivant l'importance du biais présent dans les mesures.

Dans cet exemple, l'erreur produite en supposant la mise en correspondance des barycentres d'une ellipse et de son image est du même ordre de grandeur que l'erreur de mesure sur les barycentres 2D. Pourtant, les méthodes avec ou sans l'utilisation de cette approximation donnent des résultats proches mais qui peuvent être distingués (différence de 5 mm en t_z par exemple). En effet, si la plus grande précision est désirée, on peut faire l'hypothèse de la mise en correspondance des barycentres seulement quand elle produit une erreur négligeable par rapport à l'erreur de mesure.

Sur les images de synthèse, le résidu moyen observé est encore plus petit. Ceci indique que sur les images de synthèse, la méthode de mesure des barycentres donne des résultats

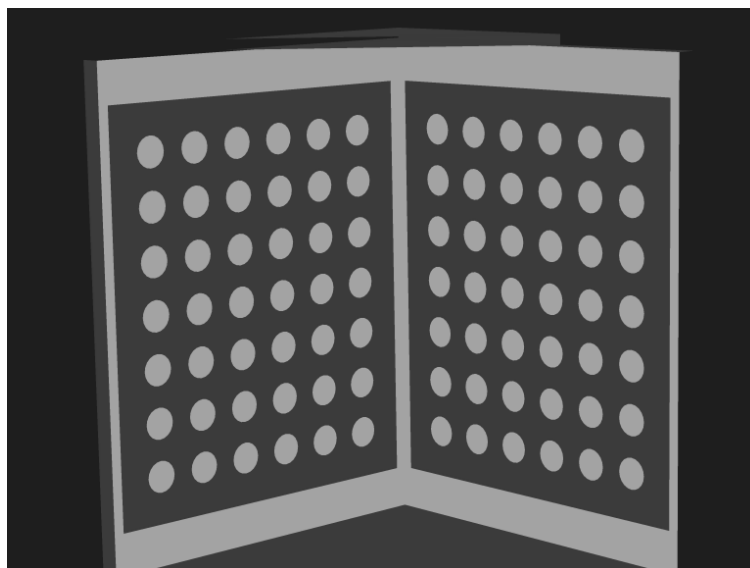


FIG. 2.7 - *Image synthétique de la mire.*

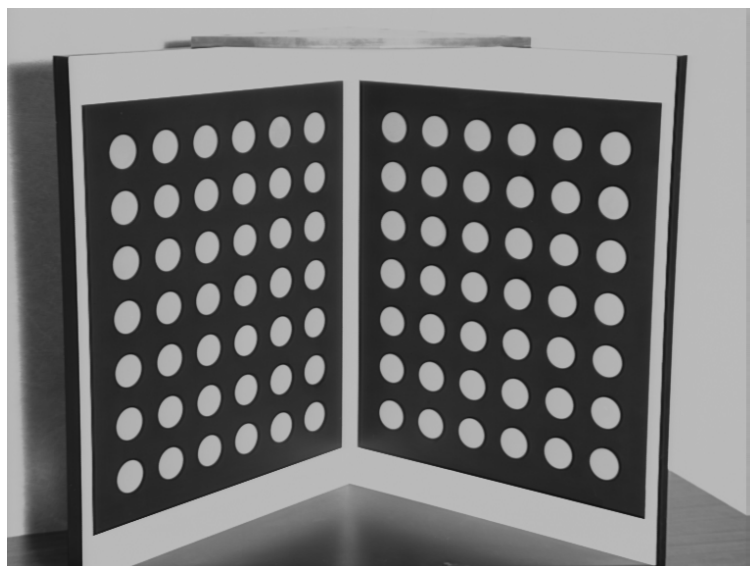


FIG. 2.8 - *Image de la mire métallique.*

	Toscani	points	ellipses
u_0	376.1 pixel	380.2 pixel	379.6 pixel
v_0	261.3 pixel	288.6 pixel	288.0 pixel
a_u	1698.5 p/mm	1810.1 p/mm	1804.1 p/mm
a_v	1648.9 p/mm	1755.4 p/mm	1750.4 p/mm
t_x	-216.5 mm	-219.9 mm	-219.5 mm
t_y	-98.2 mm	-121.9 mm	-121.4 mm
t_z	1428.6 mm	1535.2 mm	1530.6 mm
θ	51.00°	51.13°	51.11°
α		-0.351	-0.348
résidu	0.23 pixel	0.058 pixel	0.056 pixel

TAB. 2.5 - *Estimations obtenues avec trois méthodes différentes sur les mêmes points 2D.*

avec une précision inférieure à 1/10^{ème} de pixel, et explique donc l'obtention de meilleurs calibrations que dans le cas réel. En particulier, sur les images de synthèse, notre méthode augmente la précision beaucoup plus nettement.

Stabilité des paramètres intrinsèques

Il est possible de pallier partiellement au manque de référence sur les paramètres réels. En effet, si l'estimation des paramètres intrinsèques n'est pas stable, le modèle de caméra considéré n'est pas pertinent. En effet, quand la mire est déplacée dans le champ de vision de la caméra, sans toucher aux réglages de cette dernière, seuls les paramètres extrinsèques doivent changer d'une estimation à l'autre. Plusieurs modèles ont été testés, suivant le choix et la combinaison des divers types de distorsions (partie 2.2). Les estimations, avec la modélisation des distorsions radiales, se sont avérées les plus stables. Par exemple, sur une série de 12 images de la mire dans différentes positions, les résultats suivants ont été obtenus :

sténopé	u_0	v_0	a_u	a_v	
moyenne	248.8	250.8	1189	1690	
écart type	7.6	9.8	5.4	7.5	
écart en %	3.1%	3.9%	0.46%	0.45%	
radial	u_0	v_0	a_u	a_v	α
moyenne	252.4	253.5	1276	1812	-0.38
écart type	0.25	0.45	1.9	2.5	0.008
écart en %	0.1%	0.18%	0.15%	0.14%	2.1%

TAB. 2.6 - *Stabilité de l'estimation des paramètres intrinsèques suivant le modèle utilisé.*

Le choix du modèle est très important. Car comme le montre le tableau 2.6, si le modèle

ne possède pas suffisamment de degrés de liberté (sur-contraint), l'estimation risque d'être biaisée. À l'opposé, si le modèle est trop général (sous-contraint), l'estimation des paramètres, pertinents ou non, peut être instable. Ainsi, quand les distorsions radiales et de décentrage sont modélisées, l'estimation du coefficient de décentrage β varie de 35% sur la série des 12 images. Avec notre caméra CCD, une Sony couleur XC-007 et un objectif Canon J15x9.5B, le modèle le plus adéquat parmi la dizaine de modèles testés est celui qui a des distorsions radiales du premier ordre.

2.7 Conclusion

Par l'utilisation d'une mire, la calibration se réduit à faire de l'estimation sur les paramètres du modèle de la caméra.

La méthode de calibration présentée, dans ce chapitre, utilise non pas la correspondance entre points mais entre des surfaces caractérisées par leurs moments car cela permet de gagner en précision. Une technique de calcul des moments, fondée sur la photométrie des surfaces, a été développée qui prouve qu'il est possible de calculer des moments avec une précision sous-pixel dans les images malgré la discrétisation. Cette technique est de plus robuste au bruit photonique de l'image, à une mauvaise mise au point, et permet donc de fournir à l'algorithme de calibration des données précises (moins $1/10^{\text{ème}}$ de pixel sur le barycentre).

Pour profiter de toute la précision des données extraites de l'image, nous avons explicité les équations, exactes, qui lient les moments d'ordre inférieur ou égal à deux de l'image d'une ellipse à sa description 3D. D'après nos résultats, quelque soit le modèle de caméra, le barycentre des ellipses 'images' peut être utilisé comme donnée pour l'estimation. En revanche, les termes d'inertie fournissent trois contraintes supplémentaires dans le cas du modèle sténopé qui améliorent peu la précision.

Quand le modèle est bien choisi, la méthode de calibration proposée permet une bonne estimation des paramètres de la caméra (la précision est supérieure à 0.1% sur les images de synthèse). Suite à des tests avec notre caméra, le modèle avec distorsions radiales est le plus adéquat. Il permet d'estimer les paramètres de la caméra avec une stabilité de mesure d'environ 0.2%.

2.8 Calcul de l'erreur sur les moments

Nous présentons ici, dans le cas à une dimension, le calcul de l'erreur de localisation provoquée par la discrétisation en pixels de l'image. Ce calcul permet de justifier pourquoi notre technique d'extraction des barycentres arrive à atteindre une précision bien inférieure au pixel, même en présence d'un lissage symétrique important de l'image. La réponse $R(u)$ de la caméra est en effet prise en compte dans ce calcul (figure 2.9).

La distribution $CS(u)$ observée sur l'image est le résultat de la convolution de l'échelon $S(u)$ par la réponse $R(u)$ de la caméra. Soit :

$$CS(u) = \int_{-\infty}^{+\infty} R(u-x)S(x) dx$$

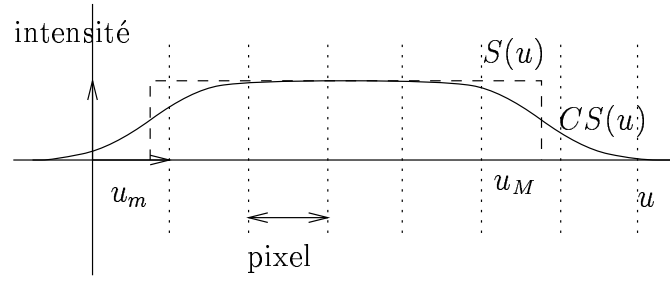


FIG. 2.9 - *Discretisation de la courbe d'intensité d'un échelon parfait (u_m, u_M) avant $S(u)$ et après $CS(u)$ convolution par la réponse de la caméra.*

$$\begin{aligned} &= \int_{-\infty}^{+\infty} R(t)S(u-t) dt \\ &= \int_{u-u_M}^{u-u_m} R(t) dt \end{aligned}$$

En conséquence, la dérivée première de $CS(u)$ est :

$$CS'(u) = R(u - u_m) - R(u - u_M) \quad (2.18)$$

La réponse est supposée normalisée. On a donc :

$$\int_{-\infty}^{+\infty} R(u) du = 1 \quad (2.19)$$

Les moments de $CS(u)$ et $S(u)$ peuvent être calculés exactement par une intégration par parties. Alors :

$$\begin{aligned} M_p(S) &= \frac{1}{p+1} (u_M^{p+1} - u_m^{p+1}) \\ M_p(CS) &= - \int_{-\infty}^{+\infty} \frac{u^{p+1}}{p+1} CS'(u) du \end{aligned}$$

p étant l'ordre du moment. En utilisant l'équation (2.18), on déduit :

$$M_p(CS) = \int_{-\infty}^{+\infty} \frac{(u + u_M)^{p+1} - (u + u_m)^{p+1}}{p+1} R(u) du$$

À partir de ces formules, on peut examiner le biais que la réponse de la caméra introduit dans le calcul des moments. Ainsi les moments d'ordre zéro de $CS(u)$ et $S(u)$ sont égaux à $u_M - u_m = M_0$. La différence entre le moment du premier ordre observé $M_1(CS)$ et celui réel $M_1(S)$ est $M_0 \int_{-\infty}^{+\infty} uR(u) du$. En conséquence, les deux moments du premier ordre sont égaux si $R(u)$ vérifie :

$$\int_{-\infty}^{+\infty} uR(u) du = 0$$

Ceci est vrai dès que $R(u)$ est symétrique. Pour les moments du second ordre, même si $R(u)$ est symétrique, il est nécessaire de corriger le moment observé de $CS(u)$ du terme $M_0 \int_{-\infty}^{+\infty} u^2 R(u) du$. Une réponse symétrique de la caméra introduit donc un biais uniquement pour le calcul des moments d'ordre pair supérieur à deux.

Il faut aussi tenir compte de la numérisation de l'image. L'erreur de localisation due au manque de résolution est la différence entre le moment observé numérisé (2.7) et celui non numérisé :

$$e_{local}(M_p) = \sum_{n=-\infty}^{+\infty} \int_n^{n+1} \left(\frac{(n+1)^{p+1} - n^{p+1}}{p+1} - u^p \right) CS(u) du$$

L'erreur au premier ordre est :

$$e_{local}(M_1) = \sum_{n=-\infty}^{+\infty} \int_n^{n+1} \left(n + \frac{1}{2} - u \right) CS(u) du$$

Soit, après une intégration par parties et quelques simplifications :

$$e_{local}(M_1) = \sum_{n=-\infty}^{+\infty} - \int_n^{n+1} \frac{1}{2} (n + \frac{1}{2} - u)^2 CS'(u) du$$

Puisque $\frac{1}{2}(n + \frac{1}{2} - u)^2$ varie entre $[0, \frac{1}{8}]$, en introduisant l'équation (2.18), on déduit l'inégalité suivante :

$$-\frac{1}{8} \sum_{n=-\infty}^{+\infty} \int_n^{n+1} R(u - u_m) du \leq e_{local}(M_1) \leq \frac{1}{8} \sum_{n=-\infty}^{+\infty} \int_n^{n+1} R(u - u_M) du$$

Donc d'après l'équation (2.19) :

$$| e_{local}(M_1) | \leq \frac{1}{8}$$

L'erreur sur le moment d'ordre un est donc bornée. Ainsi pour le barycentre $G = \frac{M_1}{M_0}$, l'erreur est majorée par $\frac{1}{8M_0}$. De façon similaire, l'erreur de localisation du moment du second ordre est :

$$e_{local}(M_2) = \sum_{n=-\infty}^{+\infty} \int_n^{n+1} \left(n^2 + n + \frac{1}{3} - u^2 \right) CS(u) du$$

D'où, après intégration par parties :

$$e_{local}(M_2) = \int_{-\infty}^{+\infty} \left(n^2 + \frac{2n^3}{3} + \frac{n}{3} + \frac{u^3}{3} - (n^2 + n + \frac{1}{3})u \right) CS'(u) du$$

Après approximation :

$$| e_{local}(M_2) | \leq \frac{M_0}{4}$$

En conséquence, l'erreur sur l'inertie surfacique $I = \frac{M_2}{M_0}$ est bornée par $\frac{1}{4}$ et le biais sur I est $\int_{-\infty}^{+\infty} u^2 R(u) du$. Si la réponse de la caméra est une gaussienne, le biais est égal à la variance de cette dernière.

Chapitre 3

Reconstruction globale de facettes 3D à partir d'images segmentées

3.1 Introduction

La reconstruction consiste à obtenir des informations 3D sur la scène observée, à partir de données extraites dans les images. Dans notre application de modélisation métrique, l'objectif de la reconstruction est d'extraire le maximum d'informations géométriques correctes du point de vue de la paire d'images traitées. De fait, elle nécessite des étapes d'analyse préliminaires, dont le nombre et la complexité dépendent du type de primitives reconstruites. Généralement, trois types de processus sont effectués sur la paire stéréoscopique :

- La calibration de la paire corrige d'une part les distorsions de la caméra pour se placer dans le modèle sténopé parfait, et d'autre part permet le calcul du déplacement entre les deux points de vue,
- Les primitives contenues dans chaque image sont extraites de celles-ci. Pour notre système de vision, les informations extraites sont des régions. L'algorithme d'extraction utilisé est alors une méthode de segmentation, où l'image est partitionnée en régions supposées être la projection d'une même facette d'un objet de l'espace 3D [Vin91, Ran92, AMG93, Bou93]. Nous proposons dans le dernier chapitre, section 5.4, une approche de résolution différente de la segmentation des images en éléments où l'intensité varie linéairement ou quadratiquement.
- Enfin, les primitives extraites dans l'image gauche et droite sont mises en correspondance aisément deux à deux [SVCG89, Ran92], en s'appuyant sur les nombreuses caractéristiques propres des régions.

La primitive géométrique la plus simple pour faire de la photométrie est la facette 3D. Comme nous désirons poursuivre la reconstruction géométrique par une reconstruction photométrique, la région est choisie comme primitive de base dans l'image. Il existe de nombreux systèmes de vision qui effectuent un autre choix de primitive tels les points [MF92, BMV93], les segments [Aya88], et les courbes [RZMF92, Ma93b, SB95]. Mais, les primitives de plus

bas niveau, comme les points ou les segments, bien que plus faciles à extraire de l'image, nécessitent trop souvent une étape supplémentaire, complexe, pour les réunir sous forme de facettes en 3D et d'objets.

Le choix des régions comme primitives extraites des images pose le problème de reconstruire des facettes 3D à partir d'une paire de régions (figure 3.1). Chacune de ces dernières est la projection d'une face plane de l'espace 3D. Le problème est donc, avec une paire de régions de l'image gauche et droite, de retrouver la position du support de la face 3D.

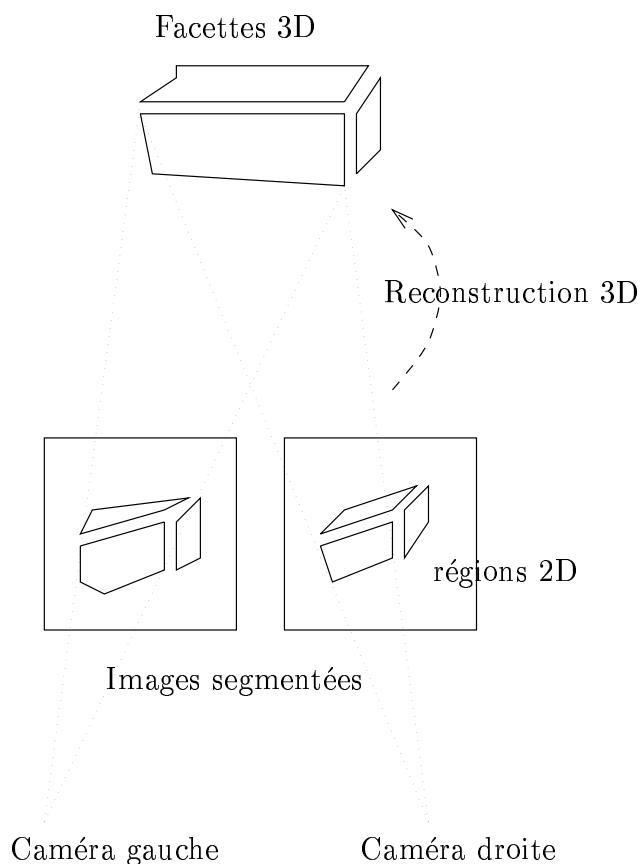


FIG. 3.1 - *Reconstruction d'un objet 3D à partir d'une paire segmentée en régions.*

Citons les méthodes de reconstruction de facettes 3D. D'abord, il est possible de travailler en cherchant des appariements au niveau local, sur des primitives de plus bas niveau que les régions, tels les sommets du contour. Par exemple, une fois appariés, grâce à la contrainte épipolaire, les sommets de la facette peuvent être reconstruits par triangulation. Comme l'information sur l'origine des points dans les régions est disponible, le contour de la facette se déduit. Il est possible d'utiliser de façon analogue les segments qui forment l'approximation polygonale du contour [CM94, JF92].

Une deuxième approche, plus originale, utilise les informations globales sur la paire de régions, en posant la reconstruction de facette comme un problème d'estimation sans nécessiter des appariements. Si la transformation perspective réalisée par la caméra est approximée par

une transformation orthographique ou para-perspective [Alo90] qui sont linéaires, la position de la facette s'exprime en fonction des caractéristiques géométriques des régions gauche et droite (voir le travail de Jean-Marc Vézien [Véz95]). Un raffinement itératif par minimisation des erreurs introduites dans l'approximation est aussi réalisable [LVG92]. Néanmoins, l'aspect itératif de cette méthode présente le défaut de se faire facilement piéger dans un minimum local.

L'avantage des méthodes avec appariement sur celles sans appariement, ou globales, est de pouvoir traiter éventuellement les occultations de la facette. En revanche, les méthodes locales sont généralement peu robustes aux inévitables petite erreurs de segmentation et peuvent présenter de problèmes de fermeture des faces.

La méthode exposée dans ce chapitre, qui évite toute approximation de la transformation perspective dans le but d'avoir la plus grande précision possible, s'inscrit dans la classe des méthodes d'estimation sans appariement fondée sur les moments géométriques. Ce qui lui permet d'être robuste aux petites erreurs de segmentation. Bien que, dans toute sa généralité, la reconstruction de facettes 3D ne peut pas être ramenée à un problème linéaire, nous avons cherché une méthode directe linéaire. Ainsi, nous proposons de rectifier les images. En effet, à cette seule condition, les équations, qui lient les caractéristiques géométriques des régions et le plan support de la face tridimensionnelle, peuvent être traitées avec des moments de manière exacte. Cette étape préliminaire de rectification est réalisable juste après la calibration, et elle rend les épipolaires parallèles entre elles sur les deux images. Alors, en se restreignant aux images rectifiées, deux méthodes de reconstruction globale de faces planes, sans appariement au niveau du pixel, sont proposées, l'une géométrique, l'autre photométrique.

Lors de cette étude, il nous a semblé utile de travailler dans un espace intermédiaire dénommé *espace des disparités*. En effet, les équations qui lient le 3D aux projections 2D y sont plus simples que celles écrites dans l'espace réel. Il est donc plus facile de raisonner dans l'*espace des disparités*. Le retour à l'espace 3D métrique peut alors être réalisé si l'on dispose de la calibration de la paire. Si cette dernière n'est pas connue, comme dans certaines approches [MF92, BMV93], on peut exploiter cet espace non métrique qui est isomorphe au réel à une transformation homographique près.

Après avoir récapitulé les relations entre un point et ses projections suivant qu'il est libre ou contraint d'appartenir à un plan fixe, nous introduisons, dans la section 3.2, l'espace intermédiaire de travail dit *espace des disparités*, qui permet de simplifier de beaucoup la formulation des équations. L'existence de cet espace nécessite une rectification de la géométrie des caméras. Dans la section 3.3, nous mettons à profit l'*espace des disparités* pour résoudre la reconstruction photométrique comme géométrie de facettes 3D. Cette dernière est robuste et précise mais nécessite des faces non occultées. Or la présence d'une occultation peut être partiellement testée avec des invariants globaux. Nous décrivons donc de quelle manière les faces non occultées sont sélectionnées. De façon complémentaire, la reconstruction photométrique est robuste aux occultations, mais nécessite la présence de dégradés réguliers d'éclairage sur les régions. Dans la section 3.4, on présente les résultats produits par notre méthode qui illustrent sa robustesse et sa capacité à traiter les occultations. Enfin dans la section 3.5, la précision de la méthode est discutée et comparée avec celle d'autres approches grâce à l'utilisation d'un outil pour mesurer la similarité entre deux ensembles 3D.

3.2 L'espace des disparités

3.2.1 Projection stéréoscopique d'un point

En toute généralité, lors de l'acquisition d'une paire stéréoscopique, la position des caméras est quelconque. Le lien qui unit les coordonnées X d'un point de l'espace 3D, à son image gauche $U_g = (u_g, v_g)$ et droite $U_d = (u_d, v_d)$ est décrit à partir des relations suivantes :

$$\left. \begin{aligned} u_g &= \frac{L_{1g}}{L_{3g}}\mathcal{X} & u_d &= \frac{L_{1d}\mathcal{X}}{L_{3d}\mathcal{X}} \\ v_g &= \frac{L_{2g}}{L_{3g}}\mathcal{X} & v_d &= \frac{L_{2d}\mathcal{X}}{L_{3d}\mathcal{X}} \end{aligned} \right\} \quad (3.1)$$

La matrice de transformation perspective \mathcal{L}_i de la caméra i est constituée des lignes à quatre éléments L_{1i} , L_{2i} et L_{3i} . En réunissant ces équations, on obtient la liaison entre les trois coordonnées X d'un point de l'espace et les quatre coordonnées (u_g, v_g, u_d, v_d) de son image dans la paire.

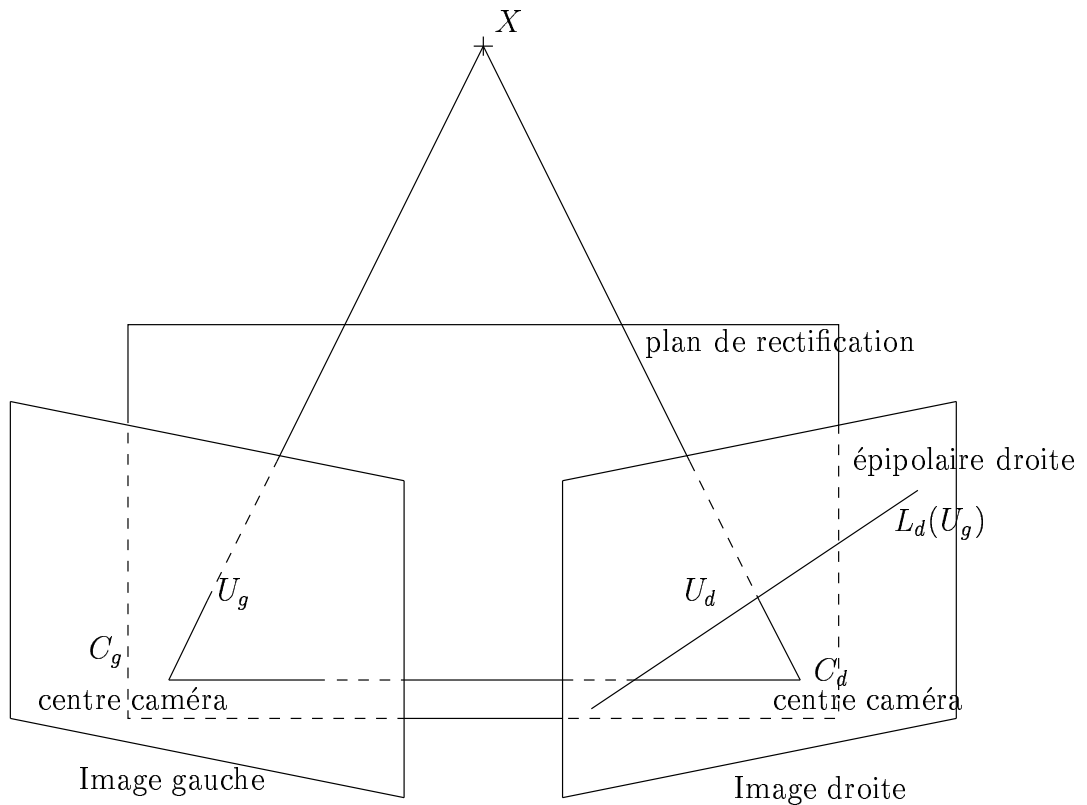


FIG. 3.2 - Le point U_g est l'image d'un point sur l'axe qui passe par C_g et X . Cette droite se projette sur l'image droite et définit la ligne épipolaire du point U_g . Le point U_d appartient donc nécessairement à la ligne épipolaire $L_d(U_g)$.

Un point est reconstruit par triangulation de sa position 3D à partir des coordonnées de ses deux projections 2D. La donnée de trois des coordonnées, (u_g, v_g, u_d) par exemple

suffit puisque les coordonnées 'image' sont liées par la contrainte épipolaire (figure 3.2). Plus généralement, on peut établir un lien univoque entre les points 3D devant les deux rétines des caméras et les points de coordonnées (u_g, v_g, u_d) s'ils sont définis. Nous nommons *espace des disparités*, cet espace constitué des coordonnées (u_g, v_g, u_d) en bijection avec les points visibles par les caméras.

3.2.2 Projection stéréoscopique d'un point dans un plan

Dans le cas d'une géométrie des caméras quelconque, il est important d'étudier la transformation subie par une surface plane de l'espace réel, par passage dans l'espace des disparités. Comme la relation entre l'espace réel et des disparités n'est pas linéaire, il est difficile de reconstruire des facettes 3D planes dans le cas général.

Soit un plan réel caractérisé par un point X_0 et deux axes V_1 et V_2 . Avec les notations homogènes, un point X appartient au plan précédemment défini à la condition qu'il existe λ et μ tels que :

$$\mathcal{X} = \begin{pmatrix} V_1 & V_2 & X_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \lambda \\ \mu \\ 1 \end{pmatrix} = \mathcal{P} \begin{pmatrix} \lambda \\ \mu \\ 1 \end{pmatrix}$$

La matrice 4×3 , \mathcal{P} , ainsi introduite, caractérise le plan réel avec les notations homogènes. Ainsi, un point du plan, de coordonnées (λ, μ) , a pour projection le point suivant, sur chaque image i :

$$\mathcal{U}_i \equiv \mathcal{L}_i \mathcal{P} \begin{pmatrix} \lambda \\ \mu \\ 1 \end{pmatrix}$$

Le signe \equiv indique l'égalité à un facteur près. On retrouve l'énoncé comme quoi le lien entre les deux projections d'un point inclus dans un plan est une homographie [Lus87] :

$$\mathcal{U}_d \equiv (\mathcal{L}_d \mathcal{P})(\mathcal{L}_g \mathcal{P})^{-1} \mathcal{U}_g \equiv \mathcal{H}_{d/g} \mathcal{U}_g \quad (3.2)$$

La matrice 3×3 de passage entre les points gauches et droits $\mathcal{H}_{d/g}$ est homogène. À partir de la première ligne de cette équation vectorielle, nous déduisons qu'un plan 3D réel se transforme, par passage dans l'espace des disparités, en une surface 2D d'équation :

$$u_d H_{3d/g} \mathcal{U}_g - H_{1d/g} \mathcal{U}_g = 0 \quad (3.3)$$

où $H_{1d/g}$, $H_{2d/g}$ et $H_{3d/g}$ sont les lignes de la matrice $\mathcal{H}_{d/g}$. Les termes de plus haut degré de la quadrique précédente sont $u_d u_g$ et $u_d v_g$. La deuxième ligne de l'équation homogène ne présente pas ici d'intérêt car elle contient la variable v_d qui n'appartient pas à l'espace des disparité tel que nous l'avons défini. Cette équation est la contrainte épipolaire.

La mise en œuvre d'une méthode de reconstruction 3D non fondée sur des appariements explicites nécessite la manipulation d'attributs globaux sur des ensembles de points 'image'. Les attributs géométriques les plus souvent utilisés sont les moments géométriques d'ordre petit, comme le barycentre et les moments d'inertie, puisqu'ils sont plus robustes au bruit

que ceux d'ordre élevé. Toutefois, un plan de l'espace 3D est transformé en une surface complexe de l'espace des disparités qu'il est difficile de manipuler. Par là même, dans le cas d'une géométrie des caméras quelconque, il n'est pas possible de relier explicitement les caractéristiques géométriques des régions gauche et droite en fonction des caractéristiques de la facette source.

Ainsi, pour reconstruire en 3D les faces de manière exacte grâce aux moments, il faut particulariser la géométrie des caméras gauche et droite. L'idée est donc d'imposer qu'un plan réel se transforme en un autre plan dans l'espace des disparités. Or, pour que l'équation de la quadrique (3.3) soit réduite à un plan, cela implique la condition suivante :

$$H_{3d/g} \equiv (0 \quad 0 \quad 1)$$

En utilisant l'équation (3.2), cette condition induit :

$$\mathcal{L}_d \mathcal{P} \equiv \begin{pmatrix} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ 0 & 0 & 1 \end{pmatrix} \mathcal{L}_g \mathcal{P}$$

De cette équation valable quelque soit \mathcal{P} , on en déduit que $L_{3g} \equiv L_{3d}$. Or d'après l'équation 2.3, L_3 est égal à $(R_3 t_z)$, avec R_3 la troisième ligne de la matrice de rotation et t_z la dernière composante de la position du centre de la caméra. Puisque $\|R_{3g}\|$ et $\|R_{3d}\|$ sont de norme 1, L_{3d} et L_{3g} sont égaux et non seulement proportionnels. En conséquence, ceci se traduit par le fait que si les focales des deux caméras sont identiques, les plans rétiniens des deux caméras doivent être confondus. Que les focales soient ou non égales, la contrainte $L_{3d} \equiv L_{3g}$ se traduit par la condition nécessaire et suffisante suivante : les deux plans images sont parallèles et parallèle à la ligne reliant les centres de projections. Ceci implique que les épipolaires doivent être parallèles sur l'ensemble des deux vues. C'est la condition pour que la reconstruction des faces soit réalisée de manière exacte avec les moments. Cette configuration géométrique des caméras est un peu plus générale que la structure dites rectifiée, mais puisque les focales du couple stéréoscopique sont identiques dans la pratique, nous nous restreignons au cas rectifié.

3.2.3 De l'espace des disparités vers celui réel

Nous allons à présent expliciter le lien entre un point réel et son correspondant dans l'espace des disparités.

Vue la contrainte $L_{3g} = L_{3d} = L_3$, le système d'équations (3.1) se simplifie et permet d'exprimer les coordonnées X d'un point réel en fonction des coordonnées 'image' (u_g, v_g, u_d) de manière linéaire :

$$\left. \begin{aligned} (u_g L_3 - L_{1g}) \mathcal{X} &= 0 \\ (v_g L_3 - L_{2g}) \mathcal{X} &= 0 \\ (u_d L_3 - L_{1d}) \mathcal{X} &= 0 \end{aligned} \right\} \quad (3.4)$$

Détaillons cette relation. Pour alléger les notations, nous choisissons l'origine des deux repères 'image' au centre des images (intersection de l'axe de visée de la caméra avec l'image), l'axe des x aligné avec les deux centres des caméras, et la longueur des axes égale à l'unité

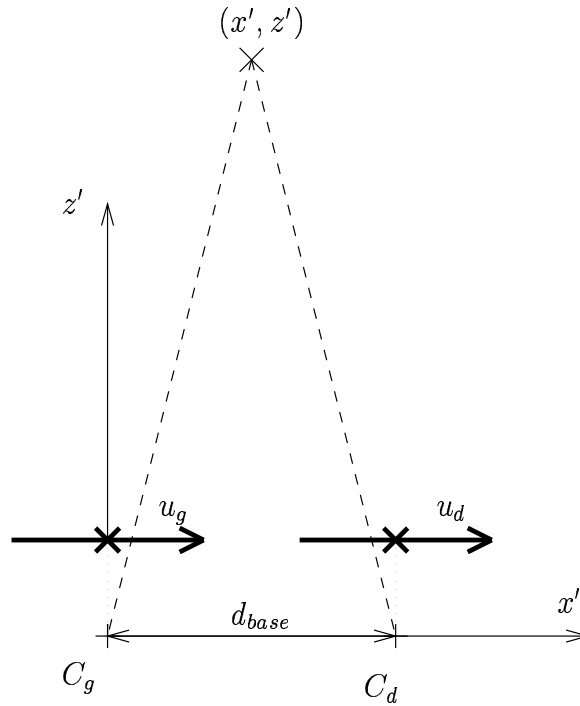


FIG. 3.3 - Géométrie des caméras rectifiées sur une coupe 2D.

de l'espace 3D et non à la taille d'un pixel. Alors, sous ses hypothèses, on a $v_g = v_d = v$ et les coordonnées horizontale gauche u_g , horizontale droite u_d et verticale v d'un point de coordonnées X' dans le repère de la caméra gauche sont :

$$\left. \begin{aligned} u_g &= \frac{x'}{z'} \\ u_d &= \frac{x' - d_{base}}{z'} \\ v &= \frac{y'}{z'} \end{aligned} \right\} \quad (3.5)$$

La distance entre les deux caméras, suivant l'axe des x' , est donnée par d_{base} (voir figure 3.3). La transformation inverse est donc :

$$\left. \begin{aligned} x' &= \frac{u_g d_{base}}{u_g - u_d} \\ y' &= \frac{v d_{base}}{u_g - u_d} \\ z' &= \frac{d_{base}}{u_g - u_d} \end{aligned} \right\} \quad (3.6)$$

L'inverse de la disparité $u_d - u_g$ intervient directement dans cette transformation inverse, d'où la dénomination d'*espace des disparités*. La relation entre l'espace réel et des disparités est homographique d'après les équations (3.5) et (3.6). Cet espace est parfois introduit dans les méthodes de reconstruction par corrélation des pixels sans être véritablement dénommé : '*search space for intra-scanline search*' [OK85], '*cross correlation volume*' [CM85]. L'introduction de l'espace des disparités facilite grandement la manipulation des facettes planes dans le cadre de la stéréovision. Effectivement, l'idée directrice est alors de travailler dans

cet espace pour effectuer la reconstruction puis, grâce à la calibration du couple, retourner vers l'espace 3D réel. Travailler uniquement dans l'espace des disparités, sans faire le retour dans l'espace réel, se rapproche des méthodes de reconstruction utilisées dans le cadre de la recherche sur l'auto-calibration, puisque la reconstruction est alors obtenue à une transformée perspective près. De fait, si l'on dispose d'une méthode robuste pour recalculer de façon homographique le modèle euclidien d'un objet de la scène sur la reconstruction (voir chapitre 5), la transformation homographique 3D obtenue permet de retrouver la calibration de chaque vue.

3.2.4 Relation affine entre les images

Un des avantages de travailler avec l'espace des disparités réside dans la simplicité à expliciter le lien entre la projection gauche et droite d'un point contenu dans un plan fixé. Dans l'espace rectifié, l'équation du plan support de la face est supposée de la forme suivante :

$$au_g + bu_d + cv + d = 0 \quad (3.7)$$

En conséquence, un point de l'image droite (u_g, v) est aussi visible sur l'image gauche en les coordonnées suivantes $(\frac{au_g + cv + d}{b}, v)$. La relation est linéaire et la dépendance avec les caractéristiques du plan est particulièrement simple :

$$\begin{pmatrix} u_d \\ v \end{pmatrix} = \begin{pmatrix} \frac{a}{b} & \frac{c}{b} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} u_g \\ v \end{pmatrix} + \begin{pmatrix} \frac{d}{b} \\ 0 \end{pmatrix}$$

Cette relation est utilisée comme équation de base dans nos méthodes de reconstruction géométrique et photométrique présentées dans les sections suivantes, mais elle peut aussi servir à faire de la reconstruction mixte géométrique et photométrique en utilisant l'information de texture des régions [TV95].

3.2.5 Interprétation géométrique

Il est possible d'interpréter la géométrie rectifiée de deux façon différentes. Elle peut être vue comme une projection perspective sur deux images dans un même plan (voir figure 3.4(a)), ou bien comme une projection orthographique sur chaque image, si elles sont placées perpendiculairement l'une à l'autre (figure 3.4(b)). Dans le deuxième cas, l'espace entre les images est celui des disparités. Cette double interprétation est la traduction géométrique de l'isomorphisme entre l'espace réel des points visibles de la scène et l'espace des disparités.

On peut remarquer que la déformation qui fait passer de l'espace réel à l'espace des disparités consiste à rejeter les foyers des caméras à l'infini. Il est important de savoir que si l'on travaille avec les coordonnées (x', y', z') , la projection est perspective, alors que si l'on travaille avec les coordonnées isomorphes (u_g, u_d, v) , elle est orthographique. L'intérêt de la projection orthographique est que les moments géométriques des images d'une face sont liés aux paramètres du plan support de la facette. Alors, la reconstruction des facettes planes est faite dans l'espace transformé, pour déduire, après transformation inverse (3.6), l'équation réelle de la facette.

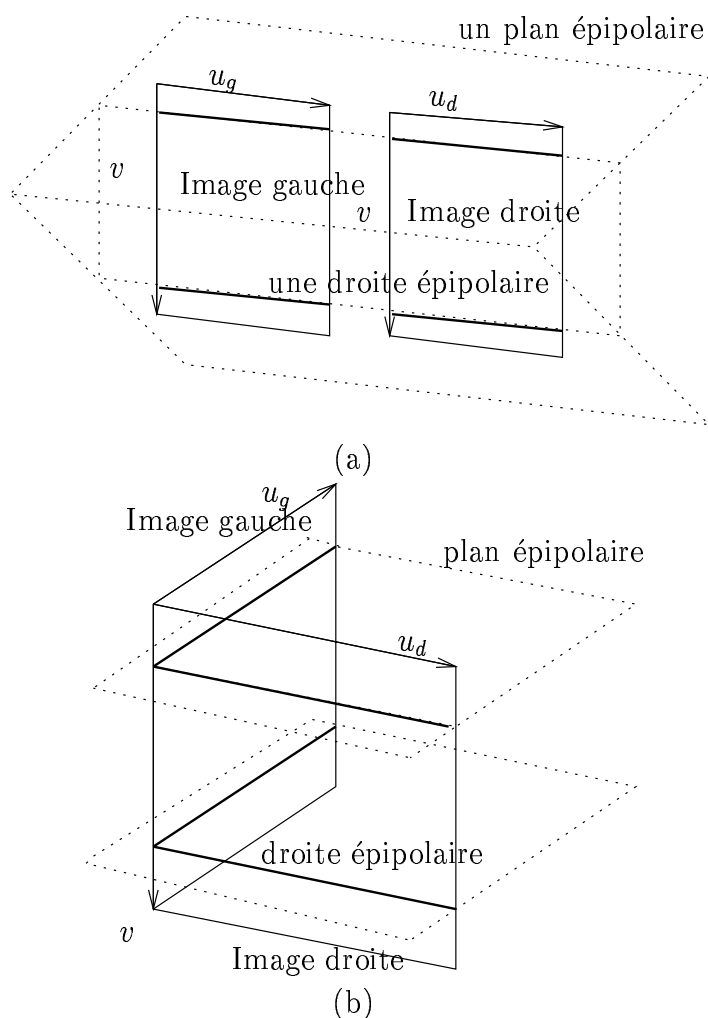


FIG. 3.4 - (a) Géométrie des caméras dans le cas rectifié. Les épipolaires sont parallèles. (b) Géométrie cohérente à la géométrie du cas (a) après rejet des centres des caméras à l'infini.

3.2.6 Images intermédiaires

Un des autres avantages de l'utilisation de l'espace des disparités est la facilité qu'il présente pour utiliser des images intermédiaires entre les deux vues. Soit d_{inter} la distance qui repère comme d_{base} la position d'une caméra intermédiaire à partir de la position de gauche (figure 3.3). Les coordonnées (u', v) de la projection d'un point sur cette vue intermédiaire sont donc :

$$\left. \begin{aligned} u' &= \frac{x' - d_{inter}}{z'} \\ v &= \frac{y'}{z'} \end{aligned} \right\}$$

L'équation (3.6) permet d'éliminer X' et donc d'avoir une équation uniquement dans l'espace des disparités :

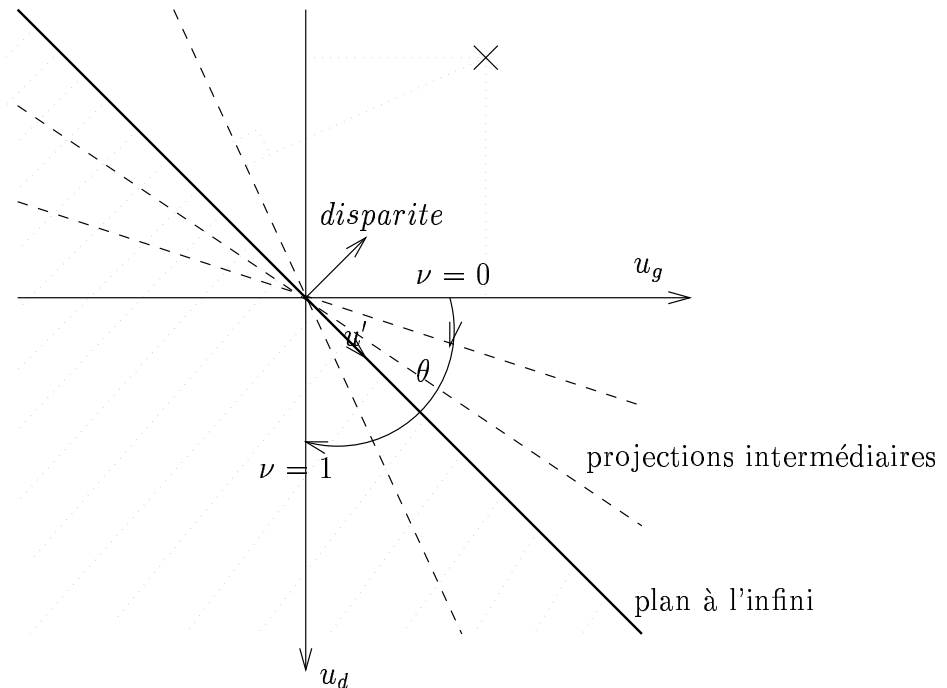


FIG. 3.5 - Coupe 2D de l'espace des disparités suivant un plan épipolaire (voir figure 3.4). Pour un point à l'infini $u_g = u_d$. Cette équation définit donc le plan à l'infini. Ainsi, la partie hachurée n'est pas visible par la caméra.

$$u' = u_g(1 - \nu) + u_d\nu \quad \text{avec} \quad \nu = \frac{d_{inter}}{d_{base}}$$

du plan de projection associé à l'image intermédiaire

La coordonnée u' est donc une simple interpolation des coordonnées u_g et u_d . Dans l'espace des disparités (u_g, u_d, ν) , l'image dans la vue intermédiaire d'un point est donc la projection sur un plan passant par l'origine dont la normale est uniquement liée à ν , comme le montre la figure 3.5 (l'angle θ est égal à $\arctan \frac{\nu}{1-\nu}$).

En choisissant comme vue intermédiaire le milieu des vues gauche et droite, le système de coordonnées (u', d) défini sur la figure 3.5 fournit le repère l'on représente les cartes de disparité. chaque point (u', ν) , la troisième coordonnée d est proportionnelle à sa disparité.

En conséquence, l'espace des disparités permet d'interpréter une séquence d'image rectifiée comme des projections orthographiques sur un plan en rotation autour de l'axe des ν . Cette propriété est très intéressante puisqu'elle permet d'appliquer à des séquences d'images rectifiées, une méthode de reconstruction volumique qui utilise directement les méthodes de reconstruction d'images scanner développées en imagerie médicale, qui disposent d'architectures spéciales.

3.2.7 Rectification de paires stéréoscopiques

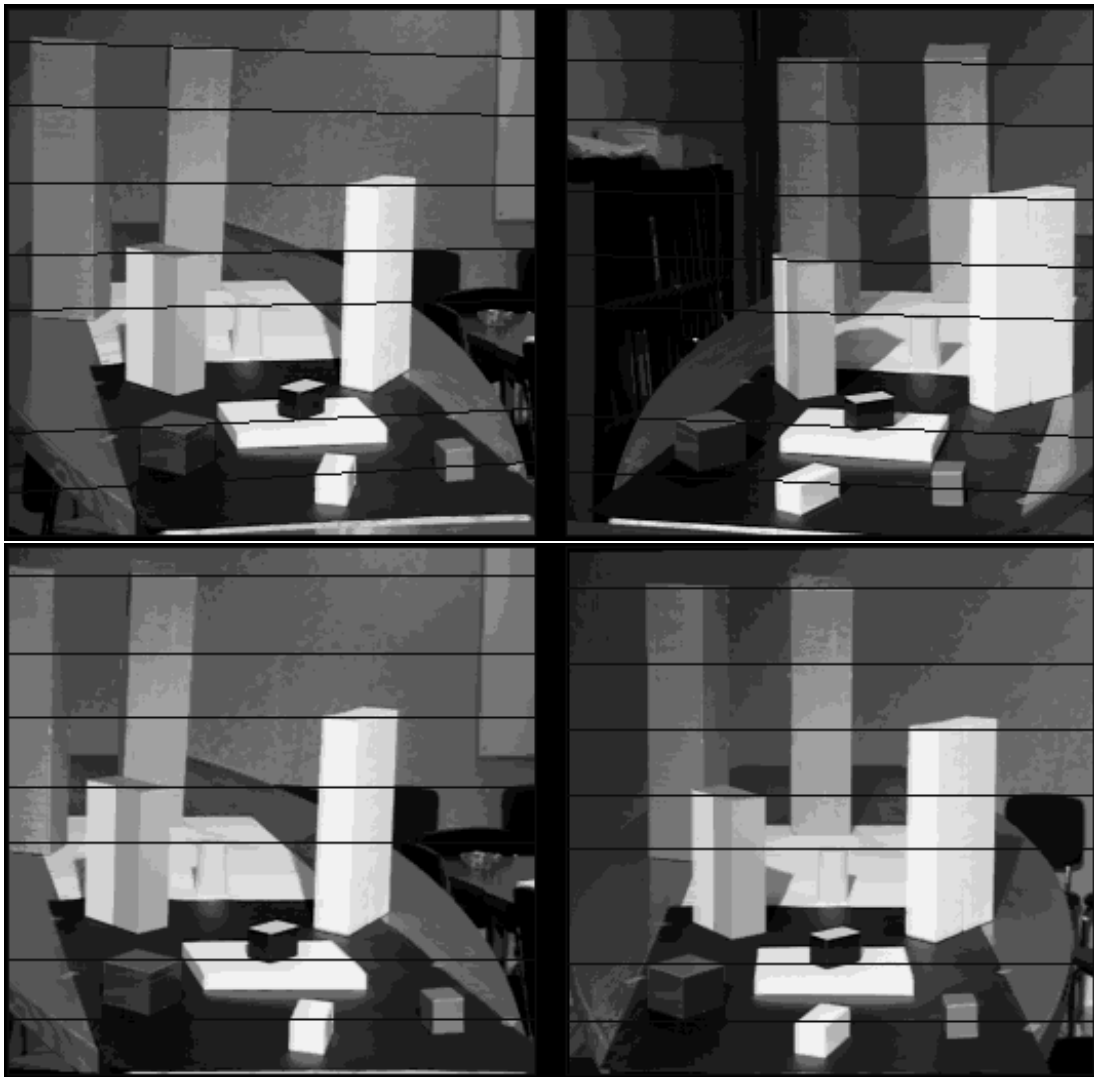


FIG. 3.6 - *Paire stéréoscopique avant et après rectification. Les épipolaires sont devenues horizontales.*

En général, les droites épipolaires ne sont pas parallèles (figure 3.6). Dans notre méthode de reconstruction, la paire stéréoscopique doit être rectifiée. Il est donc nécessaire d'avoir un algorithme de rectification pour calculer les nouvelles orientations des caméras, et de recalculer les images qui correspondent à la géométrie épipolaire redressée (figure 3.6).

Les épipolaires sont parallèles, si et seulement si les deux plans où se forment les images sont confondus, quand les caméras gauche et droite ont la même focale. La rectification consiste donc à choisir un plan 3D comme nouveau plan 'image' commun aux deux caméras. Ce plan est nécessairement parallèle à la droite définie par les deux centres de caméra C_g et C_d (figure 3.2). Il existe une infinité de plans qui conviennent. Ces plans sont tous tangents au cylindre d'axe (C_g, C_d) et de rayon la focale. Il suffit alors de choisir un plan qui ne produit pas trop de déformations sur les images : celui qui a l'orientation la plus proche de la normale moyenne des deux plans 'image' par exemple [AH88]. Dans le cas de l'utilisation de trois vues, il convient de se reporter à [Aya88]. Ensuite, l'image rectifiée est calculée par reprojection de l'image originale sur le nouveau plan rétinien. Enfin, l'axe de profondeur de la caméra devant être, par convention, confondu avec la normale au plan rétinien, les orientations des deux points de vue sont remises à jour. Les repères sur les images sont choisis alignés avec les épipolaires.

Pour générer des images rectifiées de qualité à partir des images originales, il est nécessaire d'interpoler l'image originale entre ses pixels. La méthode utilisée est une interpolation bilinéaire à l'intérieur du pixel, les valeurs des quatre sommets étant connues. Cette méthode donne de bons résultats même quand une région de l'image originale est dilatée sur l'image rectifiée.

Le processus de rectification donne des résultats facilement exploitables sur des paires dont la configuration n'est pas trop éloignée de la géométrie d'un couple rectifié (figure 3.4). Dans le cas contraire, la rectification peut produire des distorsions importantes difficiles à gérer en utilisant de simples images rectangulaires. Néanmoins, à partir de la seule connaissance de l'image originale et de la déformation induite par la rectification, il est possible de faire le calcul des moments des régions sans qu'il soit nécessaire de renumériser l'image rectifiée. Cette dernière approche, même si elle rend le calcul des moments plus compliqué, a l'avantage de ne pas tronquer les informations et peut s'appliquer à toutes les configurations géométriques de caméras.

3.3 La reconstruction de facettes 3D

3.3.1 La reconstruction géométrique

La linéarité de la projection orthographique permet d'explicitier le lien entre les caractéristiques géométriques d'une facette 3D et de son image en fonction de son positionnement. Soit l'équation (3.7) du plan support de la face :

$$au_g + bu_d + cv + d = 0$$

Sur chaque projection de la face, c'est-à-dire sur la région gauche ou droite de la paire d'images, on calcule les coordonnées du barycentre (\bar{u}_g, \bar{v}) ou (\bar{u}_d, \bar{v}) , ainsi que les trois

moments d'inertie $(\overline{u_g u_g}, \overline{v u_g}, \overline{v v})$, et $(\overline{u_d u_d}, \overline{v u_d}, \overline{v v})$. Ceux-ci sont calculés dans le repère barycentrique de la région. L'utilisation de ces seules données $(\overline{u_g}, \overline{u_d}, \overline{v})$ et $(\overline{u_g u_g}, \overline{v u_g}, \overline{u_d u_d}, \overline{v u_d}, \overline{v v})$ s'avère suffisante pour retrouver directement l'équation de la facette. Par intégration simple de (3.7) sur la face, on obtient une première équation :

$$a\overline{u_g} + b\overline{u_d} + c\overline{v} + d = 0 \quad (3.8)$$

Le coefficient d s'exprime donc, grâce à la relation (3.8), en fonction de la normale (a, b, c) au plan. On produit trois autres relations, une pour chaque coordonnée, par intégration double de l'équation (3.7), après soustraction de (3.8) :

$$a\overline{v u_g} + b\overline{v u_d} + c\overline{v v} = 0 \quad (3.9)$$

$$a\overline{u_g u_g} + b\overline{u_g u_d} + c\overline{u_g v} = 0 \quad (3.10)$$

$$a\overline{u_d u_g} + b\overline{u_d u_d} + c\overline{u_d v} = 0 \quad (3.11)$$

Le coefficient d n'apparaît pas dans ces dernières équations car les moments d'inertie sont centrés. Les deux dernières expressions (3.10) et (3.11) ne sont pas directement utilisables puisqu'il y apparaît le moment $\overline{u_g u_d}$ qui n'est pas calculable à partir des images sans avoir la correspondance pixel à pixel entre les deux régions. Toutefois, l'élimination de ce terme entre les deux équations aboutit à l'expression suivante :

$$c = \frac{a^2 \overline{u_g u_g} - b^2 \overline{u_d u_d}}{a \overline{v u_g} - b \overline{v u_d}} \quad (3.12)$$

De fait, par substitution dans (3.9), on en déduit le rapport :

$$\frac{a}{b} = \sqrt{\frac{\overline{v u_d}^2 + \overline{u_d u_d} \overline{v v}}{\overline{v u_g}^2 + \overline{u_g u_g} \overline{v v}}} \quad (3.13)$$

Il existe deux solutions pour ce rapport, l'une positive, l'autre négative. Dans l'absolu, on ne peut pas départager ces deux solutions sans avoir une information annexe sur ce que l'on observe. Néanmoins, la solution négative correspond à une face dont chaque côté est vu par une caméra différente. Orm cette configuration est très peu probable dans les scènes courantes, puisqu'il faudrait approximativement regarder une feuille selon sa tranche. Enfinm ce genre de face est généralement très mal segmentée. Ainis, la solution positive est toujours préférée. Comme le vecteur (a, b, c) est normalisé, il est complètement déterminé grâce aux équations (3.13) et (3.12). Le plan est donc complètement localisé en y ajoutant l'équation (3.8). Il est ensuite aisé de faire le passage à la facette reconstruite dans l'espace 3D ordinaire en utilisant (3.5). L'équation dans l'espace réel de la facette définie par (3.7) dans l'espace rectifié est en effet d'après (3.5) :

$$x(a + b) + yc + dz - bd_{base} = 0$$

Pour que la facette soit reconstruite de manière satisfaisante par cette méthode, elle ne doit pas être occultée. De plus, la reconstruction géométrique donne de mauvais résultats pour des formes de régions dégénérées, les régions très fines par exemple. Mais en dehors de

ces deux situations, la reconstruction réalisée est à la fois précise et robuste au bruit (voir section 3.4 et 3.5).

3.3.2 Cohérence de la reconstruction géométrique

Quant au cas où la facette à reconstruire est occultée, la méthode précédente va produire une reconstruction plus ou moins biaisée suivant l'ampleur de l'occultation (voir [VT95] pour l'étude de ce biais d'occultation). Mais il est possible d'éviter de reconstruire les faces défectueuses en sélectionnant les paires de régions pertinentes. En effet, les moments d'ordre trois et deux peuvent être utilisés pour vérifier la compatibilité avec l'hypothèse d'une face non occultée. Nous avons montré dans la section 3.2.4 que la région gauche est la transformée affine de la région droite associée. S'il n'y a pas d'occultation, il existe donc les invariants indépendants suivants [FS93] des moments M_{ij} de chaque région :

$$Inv_1 = (M_{20}M_{02} - M_{11}^2)/M_{00}^4$$

$$Inv_2 = (M_{30}^2M_{03}^2 - 6M_{30}M_{21}M_{12}M_{03} + 4M_{30}M_{12}^3 + 4M_{21}^3M_{03} - 3M_{21}^2M_{12}^2)/M_{00}^{10}$$

$$Inv_3 = (M_{20}(M_{21}M_{03} - M_{12}^2) - M_{11}(M_{30}M_{03} - M_{21}M_{12}) + M_{02}(M_{30}M_{12} - M_{21}^2))/M_{00}^7$$

Dans le cas où les invariants Inv_1 , Inv_2 ou Inv_3 sont différents pour les régions gauche et droite, la facette est à rejeter car c'est une facette occultée ou mal segmentée dans la paire. Cette étape de sélection des facettes permet de ne retenir que les facettes qui seront bien reconstruites par la méthode géométrique et de fournir les régions rejetées à un autre algorithme robuste aux occlusions, comme la reconstruction photométrique de la section 3.3.3, par exemple.

Les moments géométriques peuvent être définis par :

$$M_{ij} = \sum_{(u,v) \in \text{région}} (u - \bar{u})^i (v - \bar{v})^j$$

Mais, il est plus avantageux d'utiliser les moments pondérés par les intensités dans l'image $I(u, v)$ (éventuellement les intensités sont normalisées comme dans le chapitre 2) :

$$M_{ij} = \sum_{(u,v) \in \text{région}} I(u, v) (u - \bar{u})^i (v - \bar{v})^j$$

Nous avons étudié sur une scène de synthèse test la robustesse des trois invariants Inv_1 , Inv_2 et Inv_3 . On observe (figure 3.7) que les invariants Inv_2 et Inv_3 ne sont pas suffisamment stables pour fournir un bon critère. Par contre, l'invariant d'ordre 2, Inv_1 n'est pas trop sensible aux petites erreurs et néanmoins, il est discriminant (figure 3.8). Pour éliminer les régions qui ont une occultation supérieure à 10%, il suffit, d'après la figure 3.8, de sélectionner uniquement les paires de régions dont le rapport des invariants Inv_1 gauche et droit est entre 0.96 et 1.04.

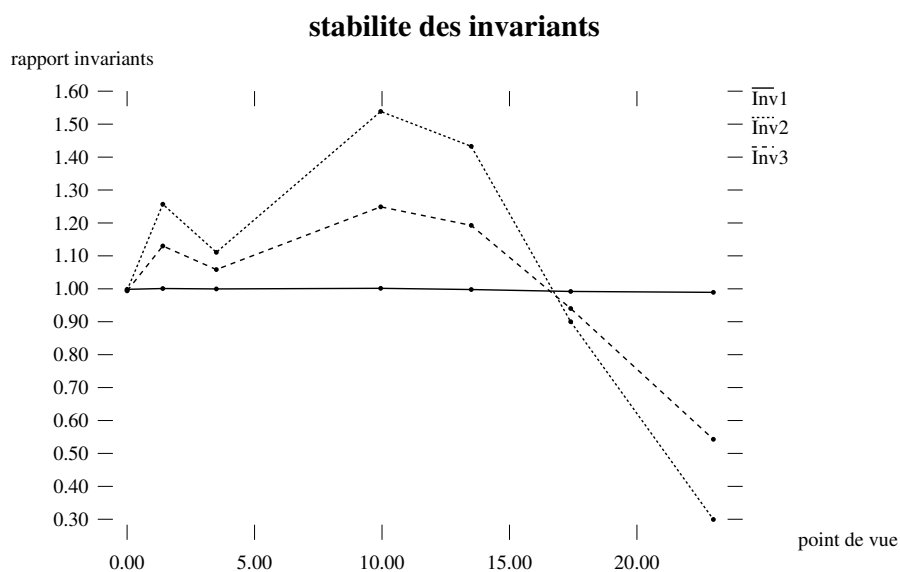


FIG. 3.7 - *Stabilité de l'invariant Inv_1 et instabilité des invariants Inv_2 et Inv_3 en fonction des changements d'orientation d'une face non occultée.*

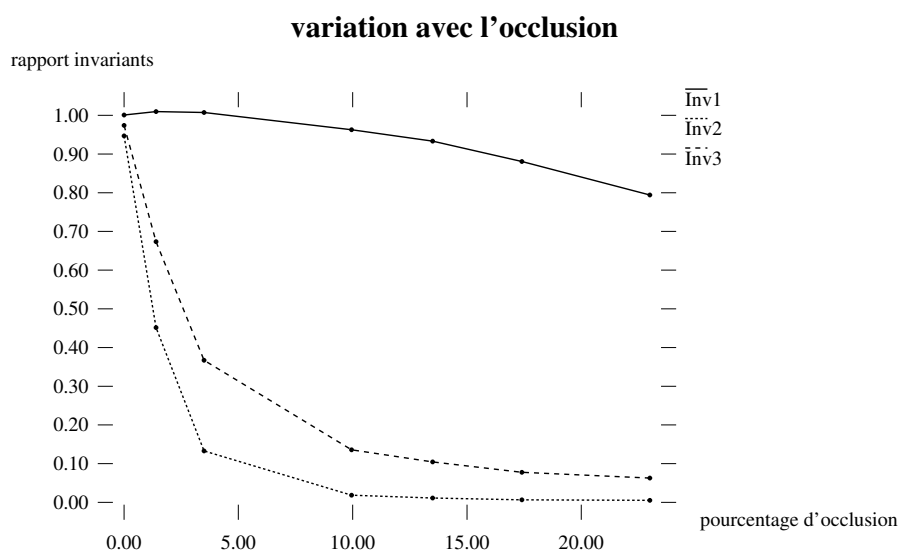


FIG. 3.8 - *Efficacité des invariants Inv_1 , Inv_2 et Inv_3 en fonction du degré d'occultation.*

3.3.3 La reconstruction photométrique

La reconstruction géométrique utilise la forme des régions gauche et droite. Mais celles-ci ont aussi comme attribut un gradient d'intensité en leur intérieur. Cette information peut servir pour la reconstruction. Il faut remarquer que l'information de gradient a l'avantage de

ne pas être perturbée par les occultations.

Soit $I_g(u_g, v)$ et $I_d(u_d, v)$ les distributions d'intensité des régions gauche et droite. On suppose que la distribution est planaire. Les surfaces doivent donc être lambertiennes et les sources lumineuses éloignées.

$$I_g(u_g, v) = \alpha_g u_g + \beta_g v + \gamma_g \quad (3.14)$$

$$I_d(u_d, v) = \alpha_d u_d + \beta_d v + \gamma_d \quad (3.15)$$

On fait alors l'hypothèse habituelle de la reconstruction photométrique, utilisée dans les méthodes créant des cartes de disparités par corrélation [OK85, CM85], où l'intensité de l'image d'un point est indépendante du point de vue. Cette hypothèse ne prend en compte, ni les variations d'angle solide suivant le point de vue, ni les éventuelles spéularités des surfaces.

L'égalisation des intensités gauche (3.14) et droite (3.15) donne directement l'équation de la facette 3D dans l'espace des disparités, ce qui illustre un des autres avantages à travailler dans cet espace :

$$\alpha_g u_g - \alpha_d u_d + (\beta_g - \beta_d)v + \gamma_g - \gamma_d = 0 \quad (3.16)$$

À partir de (3.16), il suffit, comme dans le cas géométrique, d'utiliser la transformation inverse (3.5) pour trouver l'équation de la face dans l'espace réel.

Évidemment, la reconstruction photométrique ne peut pas fonctionner si les gradients d'intensité sont nuls ou trop faibles.

3.3.4 Cohérence de la reconstruction photométrique

Un critère, pour éliminer les régions probablement incorrectes quant à leur photométrie, est disponible comme dans le cas de la méthode géométrique. Ce critère est fondé sur la validité de l'approximation planaire de l'intensité de l'image $I(u, v)$. Soit l'erreur relative d'approximation planaire :

$$e_{approx} = \frac{\sum_{(u,v) \in region} (I(u, v) - \alpha u - \beta v - \gamma)^2}{\alpha^2 + \beta^2}$$

Alors, si ce critère est trop grand, la région n'est pas fiable et sa reconstruction risque d'être erronée.

3.4 Résultats de reconstructions

Les algorithmes de reconstruction ont été testés sur différentes scènes réelles et de synthèse, plus ou moins compliquées. Nous présentons une comparaison des algorithmes géométrique et photométriques ainsi que de leur cadre de validité.

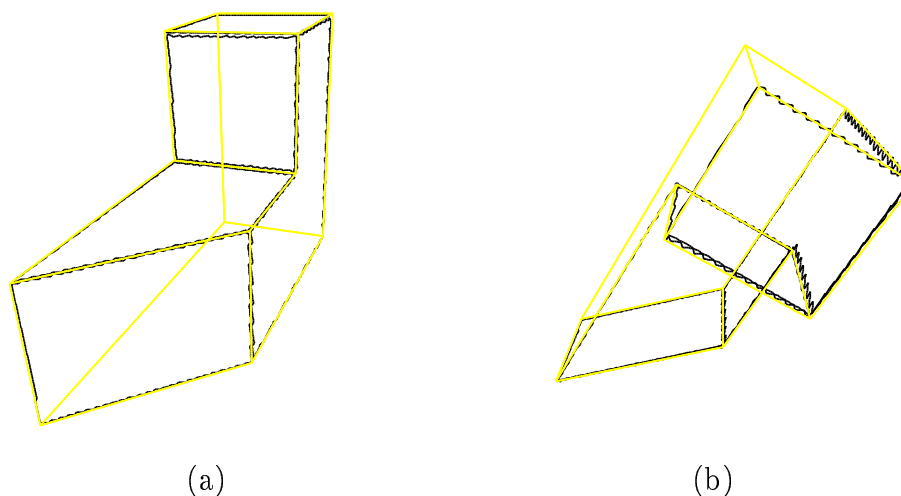


FIG. 3.9 - Vue de face (a) et de côté (b) de la reconstruction (traits en zigzag) d'un objet de synthèse et du modèle exact associé.

3.4.1 Images de synthèse

Les premiers tests ont été réalisés sur des images synthétiques. Les scènes sont uniquement constituées d'un unique objet. Les images sont segmentées [AMG93] puis mises en correspondance. Ensuite, la reconstruction est comparée au modèle exact utilisé pour générer la scène comme on peut le voir dans la figure 3.9. On voit sur la vue de côté (b) que la reconstruction (traits en zigzag sur la figure 3.9) est à la bonne position et à la bonne taille. De même, la géométrie de l'objet est bien respectée, en particulier les angles droits. Le tableau 3.1 présente les résultats numériques de la comparaison.

Couple de facette	Angle réel	reconstruit
0-1	135.0	135.45
0-2	90.0	89.61
1-2	90.0	91.24
1-3	90.0	87.79
1-4	90.0	88.98
2-3	90.0	94.33
2-4	0.0	2.45
3-4	90.0	95.20

TAB. 3.1 - Cohérence des normales reconstruites sur l'objet de synthèse de la figure 3.9

3.4.2 Méthodes géométrique et photométrique

La reconstruction a ensuite été testée sur un cube réel. La figure 3.11 est la paire obtenue suite aux étapes de calibration et de rectification.

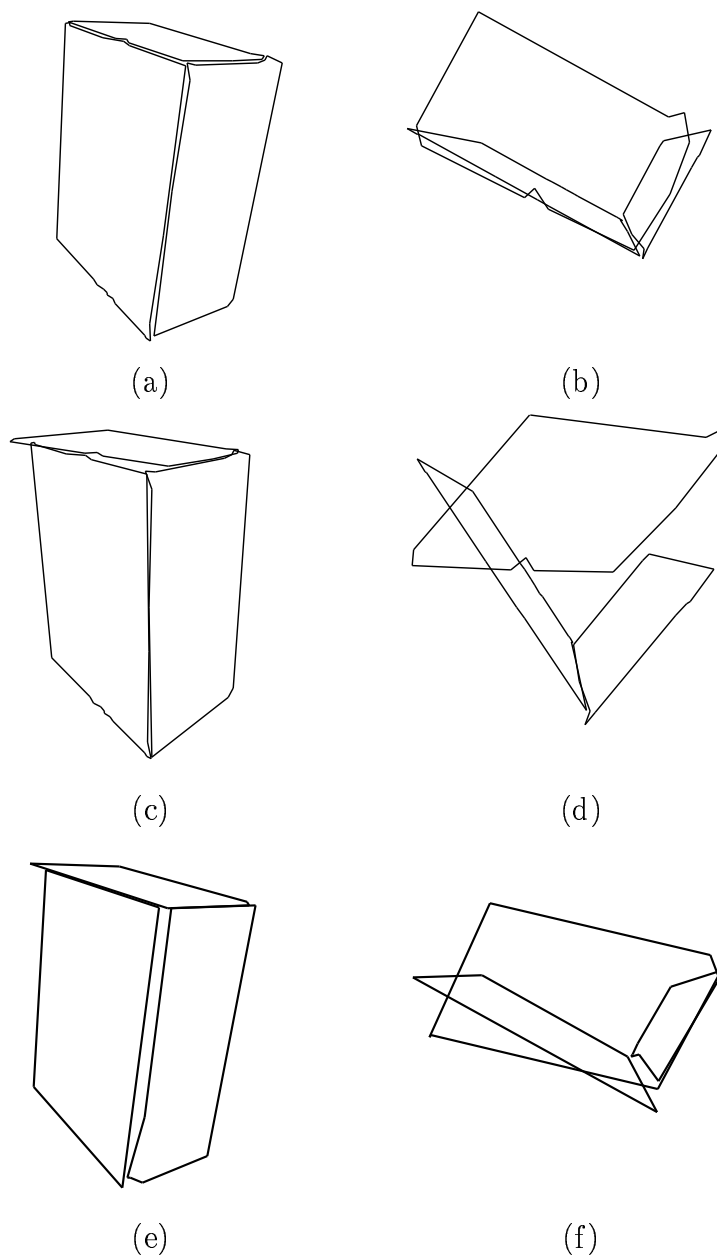


FIG. 3.10 - Vue de face (a) et de dessus (b) de la reconstruction du cube de la scène, à partir de la géométrie des régions. Vue de face (c) et de dessus (d) de la reconstruction photométrique du cube de la scène. Vue de face (e) et de dessus (f) de la reconstruction photométrique, après correction radiométrique des images. Les images originales sont dans la figure 3.11

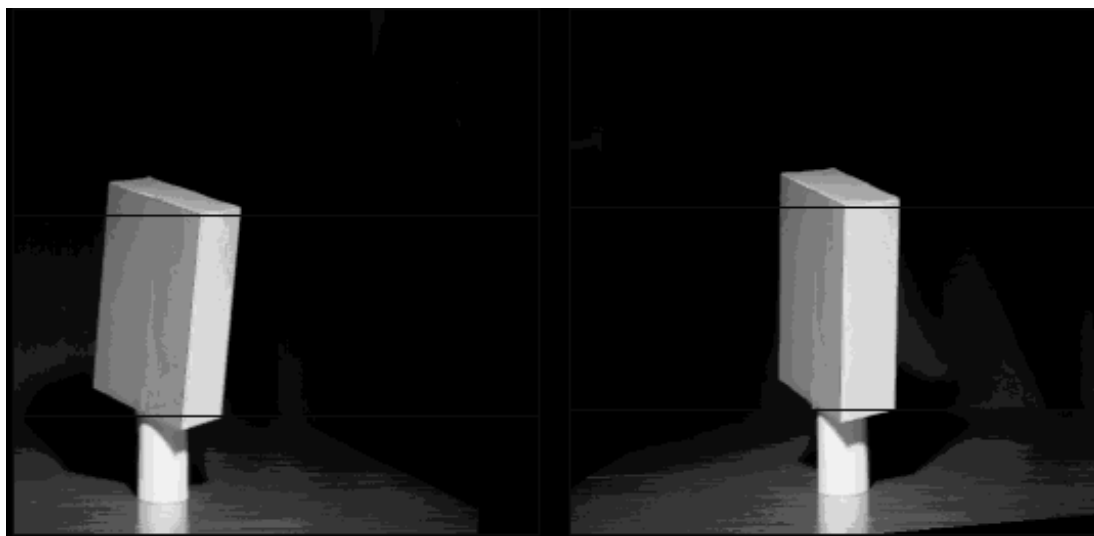


FIG. 3.11 - *Paire stéréoscopique réelle d'un cube après rectification.*

Les images de la paire ont été segmentées par un algorithme de croissance de régions [Vin91] puis les régions obtenues ont été mises en correspondance [SVCG89] entre l'image gauche et droite. Les deux méthodes de reconstruction géométrique et photométrique ont été testées. La méthode de reconstruction géométrique (figure 3.10) donne d'assez bons résultats quant à l'orthogonalité des faces et à leur positionnement relatif.

La méthode de reconstruction photométrique (figure 3.10) donne de moins bons résultats car les gradients sur chaque face sont difficiles à estimer précisément. Les raisons de cette difficulté sont de deux types :

- La caméra n'a pas été calibrée radiométriquement. Or, notre caméra atténue les bords de l'image d'une vingtaine de niveaux de gris. Ceci biaise beaucoup la reconstruction photométrique. Au cours de ces tests, il est apparu combien la calibration radiométrique de la caméra est importante. En effet, elle est nécessaire dès que l'on veut utiliser l'information photométrique de la scène (voir Annexe A). On peut voir sur le bas de la figure 3.10, l'amélioration produite par la calibration radiométrique de la paire d'images.
- Certaines facettes, en particulier le dessus du cube, ont un gradient très faible qui est mal estimé à cause de la digitalisation. La reconstruction est alors biaisée. La texture et les variations de pigment peuvent aussi produire des erreurs importantes dans l'estimation du gradient.

3.4.3 Sélection des faces

Nous présentons maintenant des résultats sur une scène réelle plus compliquée (figure 3.6) formée de polyèdres. On voit dans la figure 3.12 plusieurs parallélépipèdes bien reconstruits.

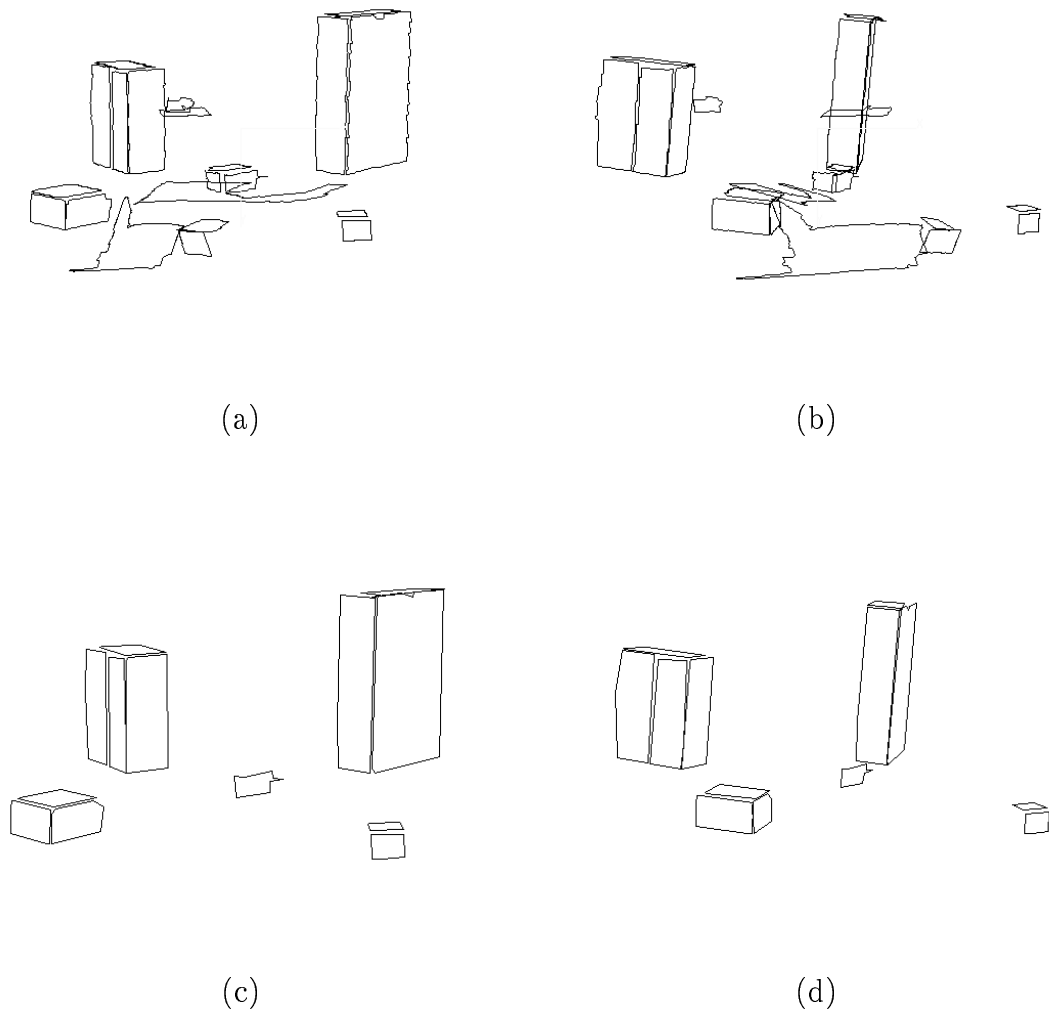


FIG. 3.12 - Vue de face (a) et de côté (b) de la reconstruction de la scène visible dans la figure 3.6. Vue de face (c) et de côté (d) de la reconstruction de la même scène après élimination automatique des faces mal segmentées.

Mais, certaines facettes solitaires semblent flotter dans l'espace sans correspondre à des objets physiques. Les régions utilisées pour les reconstruire sont donc mal segmentées. Ces faces défectueuses sont éliminées en utilisant le test de cohérence fondé sur l'invariant Inv_1 , comme on le voit sur le bas de la figure 3.12.

3.4.4 Des objets non polygonaux

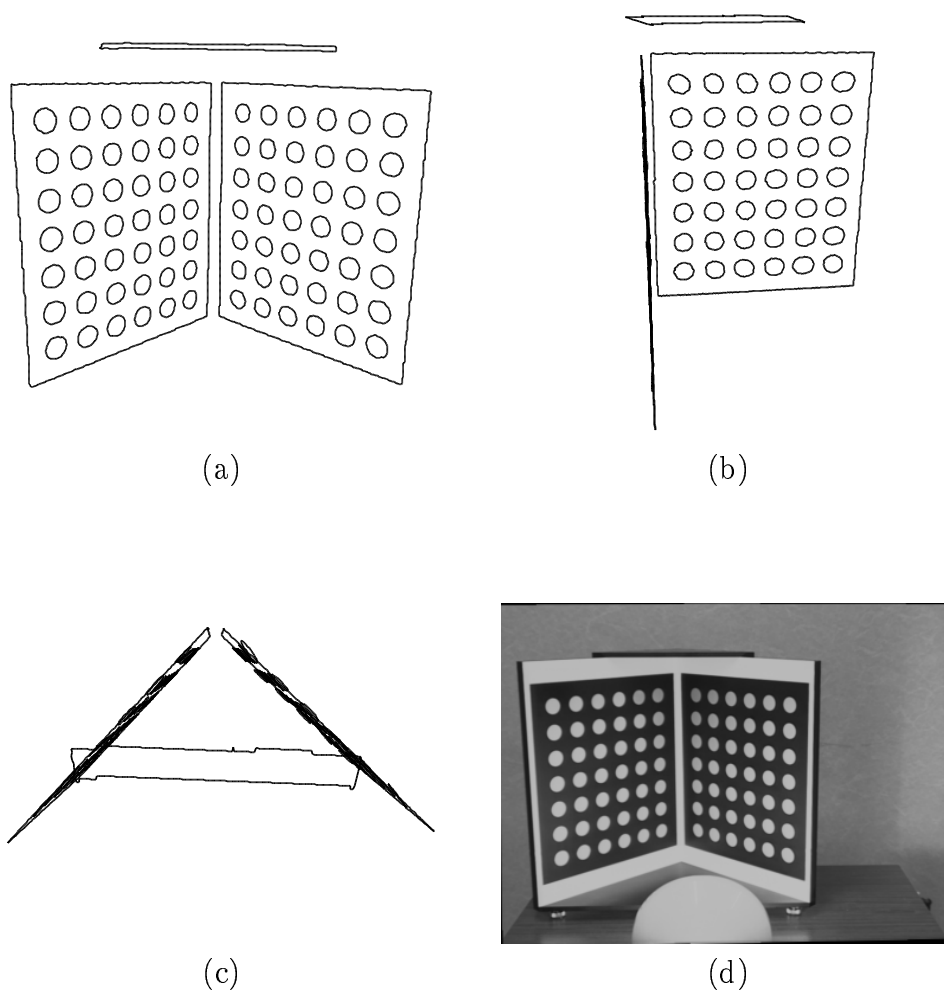


FIG. 3.13 - Vue de face (a), de côté (b) et de dessus (c) de la reconstruction de la mire de calibration obtenue à partir de l'image (d).

La contrainte sur les objets, pour pouvoir les reconstruire, est d'avoir des facettes à contour plan. Ainsi, à partir d'une paire rectifiée (figure 3.13), segmentée avec un algorithme flou [Bou93], les cercles disposés sur la mire sont reconstruits avec une bonne précision dans un même plan (figure 3.13). Du fait de la présence d'ellipses, il est difficile de faire la reconstruction avec les techniques classiques de reconstruction qui se fondent sur la mise en correspondance de points ou de segments sur les contours.

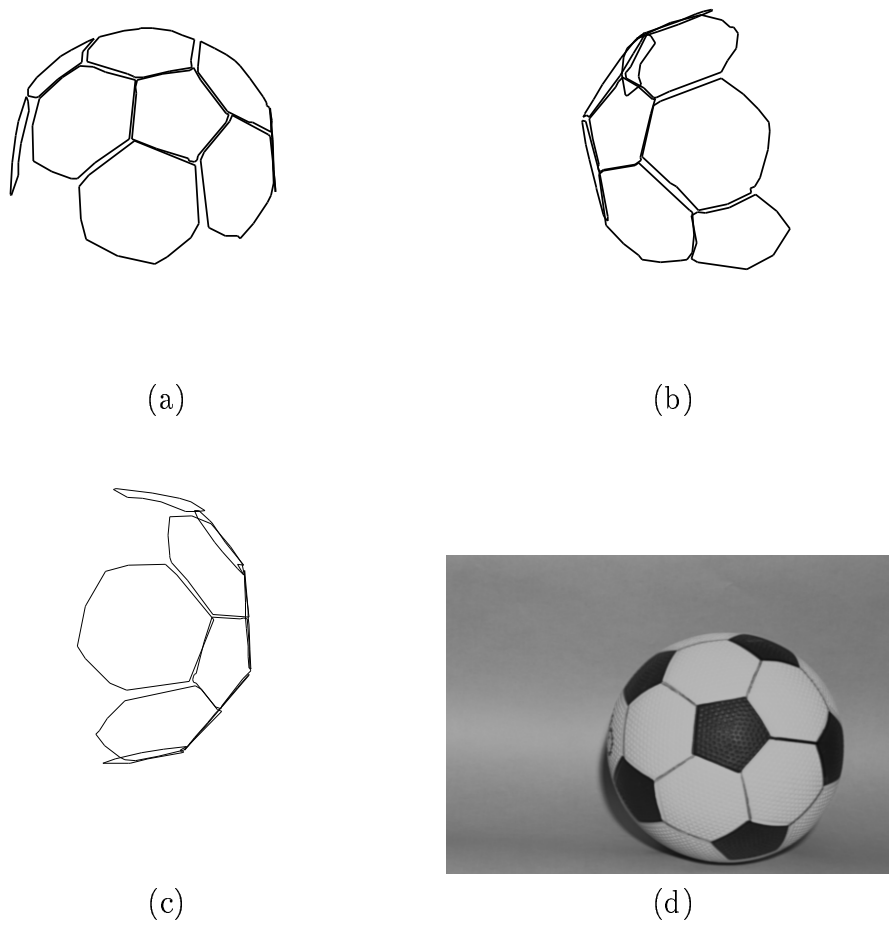


FIG. 3.14 - Vue de face (a), de côté (b) et de dessus (c) du ballon reconstruit à partir de l'image originale (d).

Notre méthode peut aussi reconstruire une approximation polygonale des objets courbes à la condition que le contour des faces courbes soit plan. En effet, aucune hypothèse n'est faite sur la forme des faces pour la méthode purement géométrique. Dans la figure 3.14, un ballon, objet sphérique, est reconstruit.

3.4.5 Avec ou sans rectification intermédiaire

La rectification permet de faire la reconstruction en facettes de manière exacte. Ceci fait gagner en précision la reconstruction par rapport à la technique proposée par Jean-Marc Vézien [Véz95] qui approxime la projection perspective par une transformation orthographique ou perspective pour obtenir une correspondance affine approchée entre les deux images. Pour illustrer l'avantage de procéder à la rectification, nous avons comparé les résultats de ces deux algorithmes sur la même paire d'images.

Les conditions de prise de vue sont tels que l'approximation reste valide (cf. [Véz95]) : 10° de vergence des caméras, 40 cm de base stéréoscopique pour une scène éloignée de 1.6 m. Le couple d'images obtenu est visible sur la figure 3.15(a) et (b). Pour l'essentiel, cette scène est constituée de morceaux de papier de différentes couleurs disposés sur deux plans inclinés. Le résultat de la méthode avec approximation mais sans rectification est visible en (c) et (d). On remarque une certaine dislocation des facettes reconstruites. La reconstruction obtenue en (e) et (f) avec notre méthode, après rectification des images, présente des faces plus jointives. Par ces images, on voit visuellement l'intérêt de faire la rectification et le gain en précision obtenue.

On remarque que l'ensemble des facettes planes de la scène est bien reconstruit, sauf l'éponge, le stylo Bic, les ciseaux et le fond. En effet, l'éponge ne répond pas à l'hypothèse d'uniformité radiométrique de la surface à cause de ses nombreuses cavités. Les ciseaux, comme le stylo, sont eux beaucoup trop spéculaires pour être bien segmentés. Quant au fond, il ne peut pas être reconstruit car il est occulté par le cadre de l'image.

3.5 Comparaison avec d'autres méthodes

Suite aux résultats obtenus, il se révèle indispensable d'avoir un outil de comparaison de données 3D. Cet outil est en effet très utile pour valider notre méthode de reconstruction 3D en calculant l'erreur moyenne de la reconstruction par rapport au modèle de référence s'il est disponible, ou par rapport au résultat d'autres reconstructions.

3.5.1 Présentation de l'outil de mesure

La définition de cet outil pose le problème suivant : en présence de deux ensembles de données 3D, par quels critères estimer la ressemblance de deux jeux de données ? Comme les deux ensembles de données 3D que nous utilisons ne peuvent se recouvrir que partiellement, il est nécessaire de sélectionner les parties des deux ensembles qui peuvent être mises en correspondance. En conséquence, chaque ensemble 3D est divisé en deux parties dont l'une ne peut pas être mise en correspondance. Sur la portion des données qui ne recouvre pas le modèle, la seule mesure intéressante est sa taille relative. Par contre, sur la partie commune,

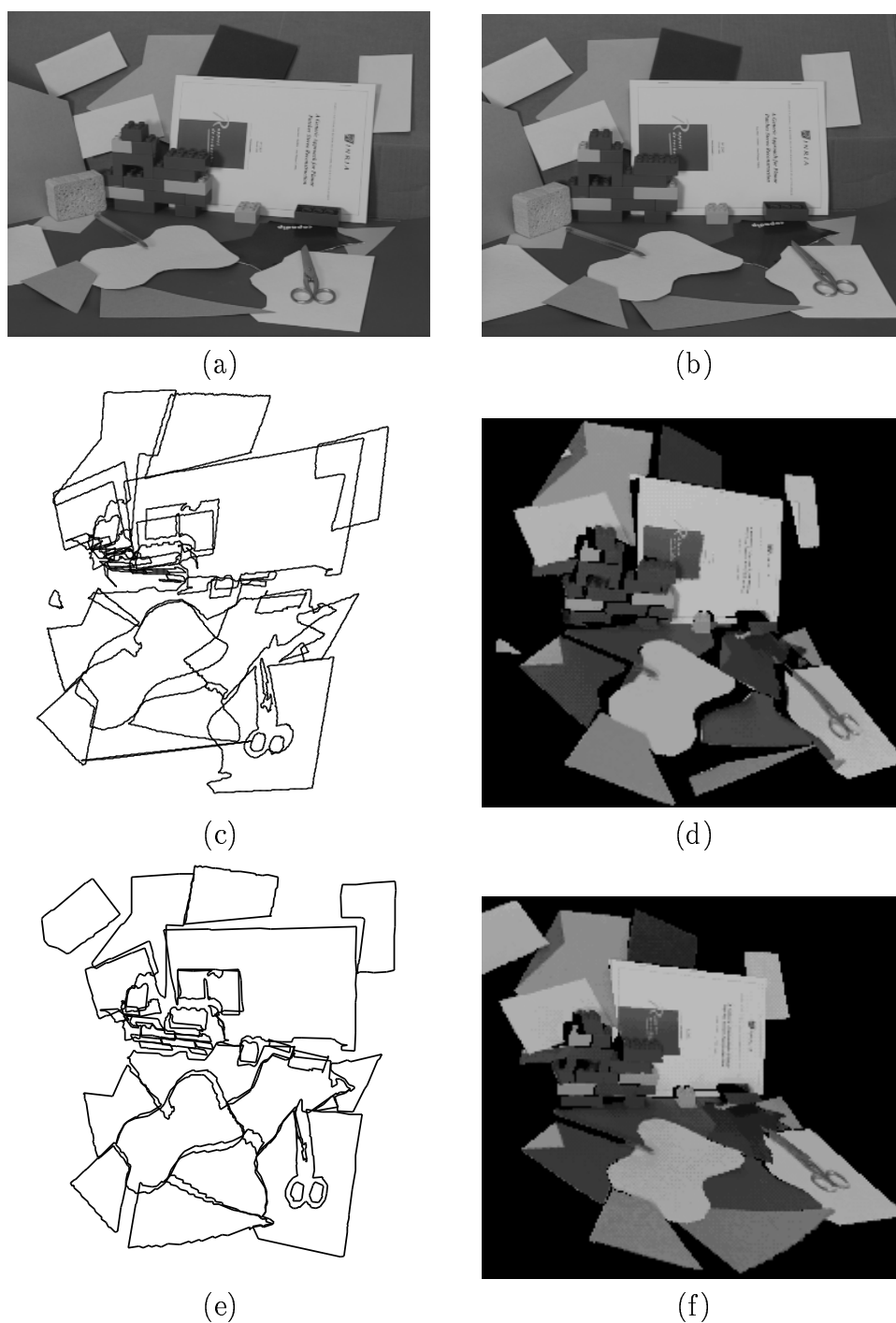


FIG. 3.15 - Vues originales gauche (a) et droite (b) d'une scène qui forme deux plans. Vues légèrement décalées vers le haut des contours (c) et en faces texturées (d) de la reconstruction sans rectification intermédiaire et donc avec approximation de la projection perspective (Méthode de Jean-Marc Vézien). Vues légèrement de dessus des contours (e) et en faces texturées (f) de la reconstruction avec rectification et notre méthode sans approximation.

les deux mesures importantes sont le biais et l'écart type des déplacements entre les données mises en correspondance.

On suppose que les ensembles 3D sont constitués de points, puisqu'il est toujours possible d'échantillonner avec des points des géométries linéiques et surfaciques. Nous avons choisi, pour pouvoir estimer les indices de similarité, de faire la mise en correspondance de chaque point 3D du premier ensemble dit de référence avec le point le plus proche en distance de l'autre ensemble.

Une fois la mise en correspondance effectuée, une classification est réalisée sur les déplacements des points en correspondance entre les deux ensembles. Un algorithme de segmentation permet de séparer, de manière adaptative, les données en recouvrement de celles qui sont en fausse correspondance. L'algorithme de classification utilisé est celui de Davé (voir section 1.6.2) avec deux classes dont la première récolte les déplacements faibles et la seconde, classe bruit, le reste des déplacements 3D. Il est alors aisé de calculer sur la première classe des critères statistiques sur les déplacements, comme le biais et l'écart type moyen des erreurs de reconstruction sur la partie en correspondance.

Cet outil permet l'évaluation objective de nos résultats par rapport à ceux qui furent obtenus par d'autres approches.

3.5.2 Performance de la reconstruction

Voici à présent un exemple des mesures que l'on peut obtenir grâce à notre méthode qui évalue la similarité de deux ensembles 3D. Nous disposons du modèle géométrique exact de la mire que nous utilisons (voir vue (a) de la figure 3.16). En effet, la mire est l'objet qui a permis de faire la calibration des vues, c'est-à-dire l'estimation de la position de chaque vue. En conséquence, on dispose, avec une bonne précision (chapitre 2) d'une position de référence. Celle-ci est présente dans une scène complexe composée de plusieurs objets dont un ballon. La reconstruction 3D obtenue par notre méthode est visible dans la vue (b) de la figure 3.16, ainsi que dans la figure 0.1.

L'espace des déplacements est visible dans la figure 3.17. Le cœur central de cet ensemble est séparé du reste par une méthode de classification. En effet, les points associés à des déplacements faibles sont selon toute vraisemblance des points communs aux deux ensembles 3D, au bruit et erreurs près.

Une fois la classification réalisée, on visualise sur la figure 3.18(a) la partie du modèle géométrique de la mire qui a été effectivement reconstruit. Une analyse statistique très simple des erreurs de reconstruction est alors possible. Ces tests nous ont montré que notre reconstruction avait une précision de l'ordre de 4 mm à une distance de 130 cm pour une base de 34 cm (voir tableau 3.2). Cette valeur correspond approximativement à la demi-largeur à mi hauteur du pic de la figure 3.20.

On montre que l'erreur de localisation stéréoscopique en profondeur d'un objet à une distance p_{objet} avec une base d_{base} est donnée par la formule suivante dans le cas rectifié (voir figure 3.19) :

$$e_{prof} = \frac{2p_{objet}^2}{a_u d_{base}}$$

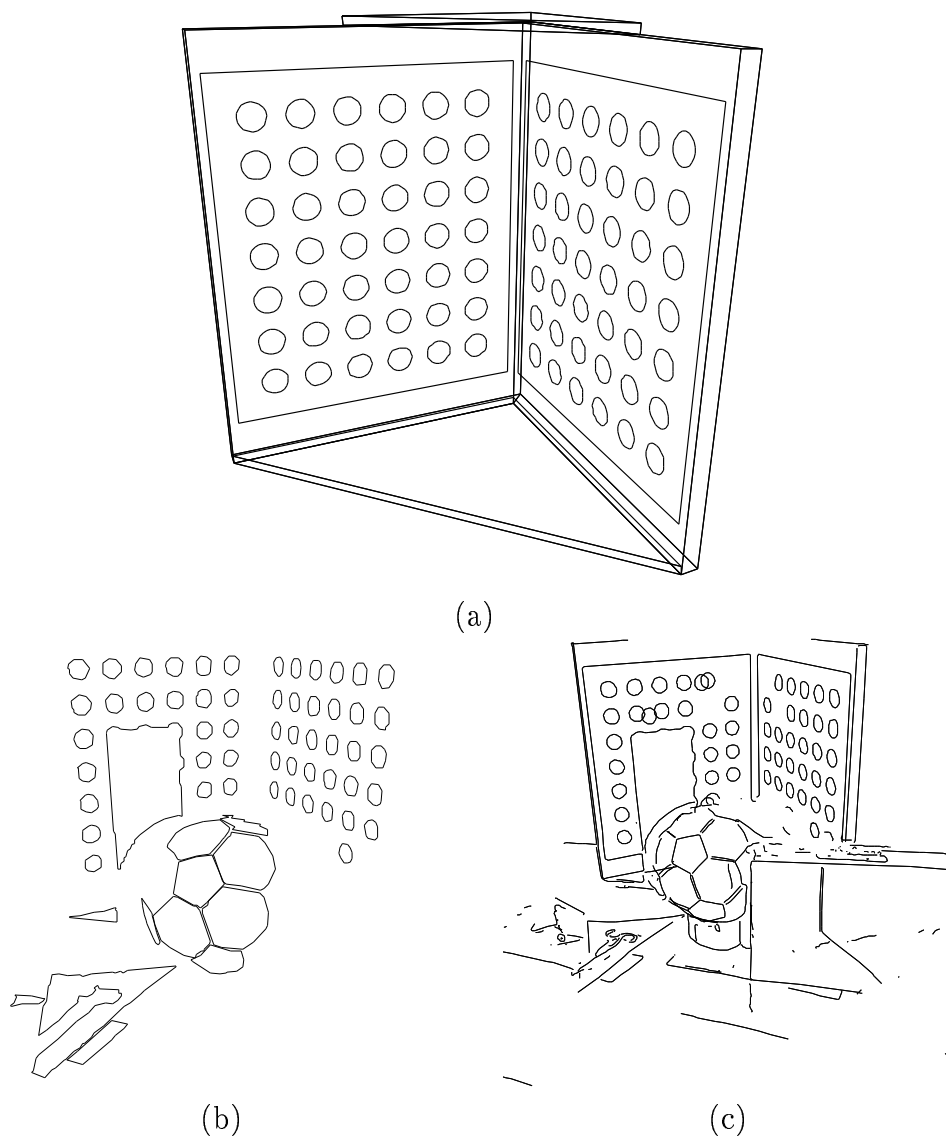


FIG. 3.16 - Le modèle de référence (a), notre reconstruction (b) et celle gracieusement fournie de B. Serra (c). B. Serra utilise pour faire la reconstruction un algorithme de mise en correspondance sous-pixel des contours des deux images. Ces contours sont représentés par des chaînes ouvertes de segments. Cette méthode permet donc d'obtenir une bonne précision en même temps d'une reconstruction d'une grande partie des contours.

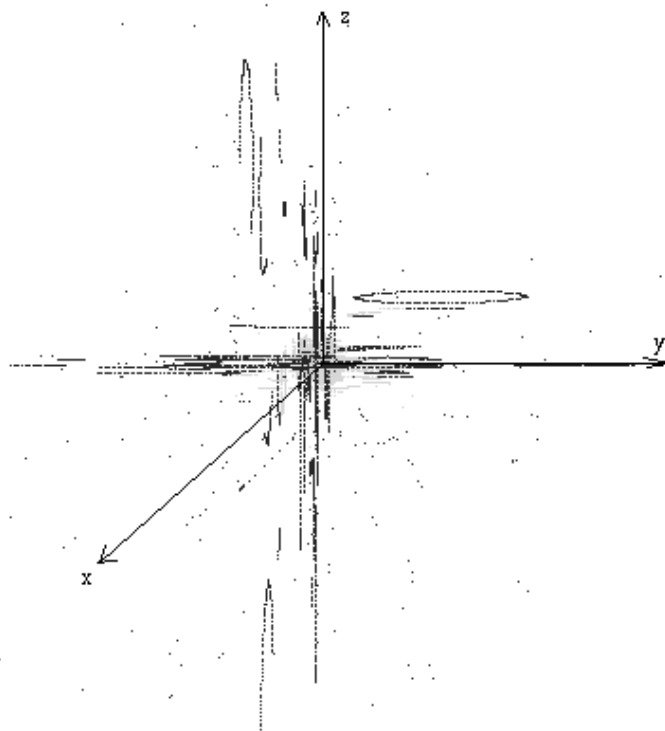


FIG. 3.17 - Chaque point représente le sommet d'un déplacement 3D entre un point du modèle et de la reconstruction. La partie centrale, proche de l'origine, est donc associée à la partie commune entre la reconstruction et le modèle de la mire. La partie en noir correspond aux appariements aberrants.

où a_u est l'inverse de la taille d'un pixel de la rétine des caméras. L'erreur théorique est donc de 9 mm en profondeur et 1.2 mm de face. La précision obtenue par notre reconstruction est inférieure à l'erreur induite par la discrétisation de l'image sans explicitement faire une détection sous-pixel des contours des régions. L'observation du tableau 3.2 montre que l'écart type des erreurs est proche de la limite théorique de 1.2 mm.

3.5.3 Comparaison des méthodes de reconstruction

Notre outil permet en outre de comparer objectivement différentes méthodes de reconstruction (voir figure 3.16). Ainsi, la méthode de B. Serra [SB95], de même que la notre, fournissent des résultats sous-pixel en utilisant des primitives de base d'assez haut niveau. En revanche, nous n'utilisons pas comme B. Serra des courbes quelconques mais des contours fermés planaires.

Cette restriction permet d'obtenir une reconstruction, sur cette scène particulière, légèrement plus précise. Ceci se voit lorsque l'on compare les histogrammes des déplacements entre le modèle géométrique de référence et chaque reconstruction de la mire (voir histo-

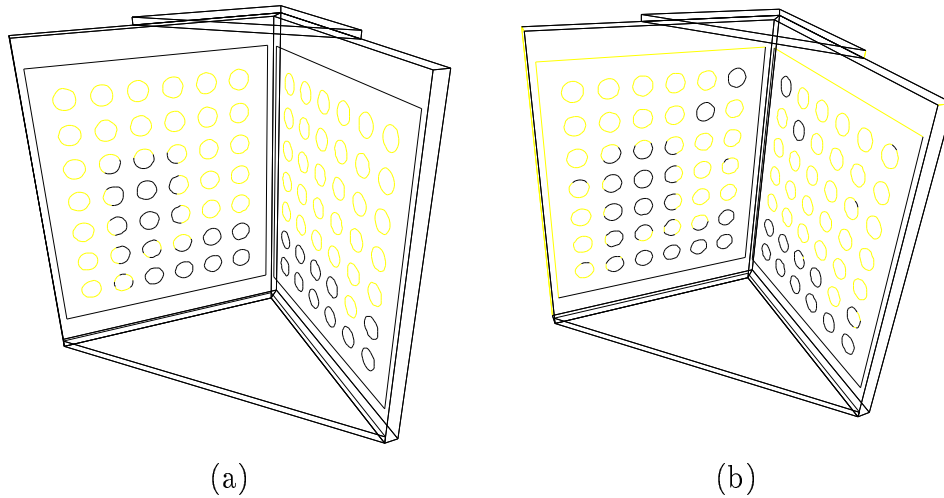


FIG. 3.18 - (a) Séparation du modèle parfait de la mire en la partie reconstruite et non reconstruite avec notre méthode (aux artéfacts près). (b) Même résultat mais avec la reconstruction de B. Serra.

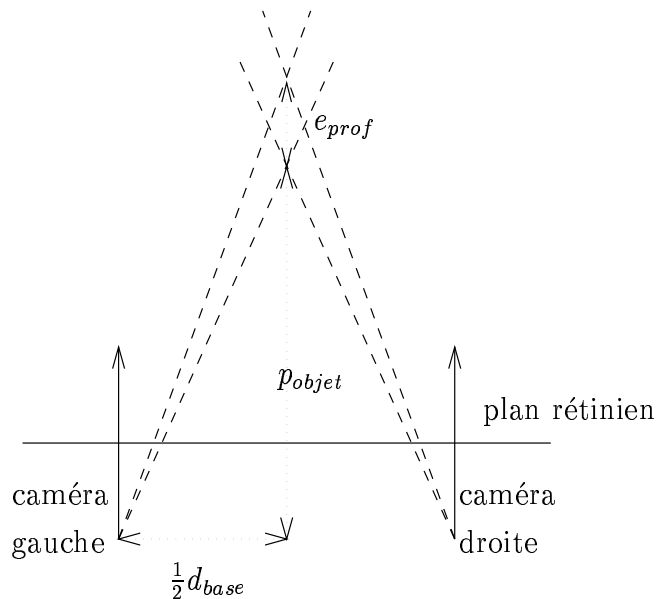


FIG. 3.19 - Calcul de l'erreur de localisation stéréoscopique d'un objet à une distance p_{objet} avec une base d_{base} .

gramme 3.20). Cette observation est confirmée par le tableau 3.2.

Par contre, on remarque sur ce même tableau 3.2 que notre méthode reconstruit un pourcentage légèrement plus faible de la mire. Cela s'explique par le fait que nous sommes restreints à des courbes fermées planaires comme on peut l'observer en comparant

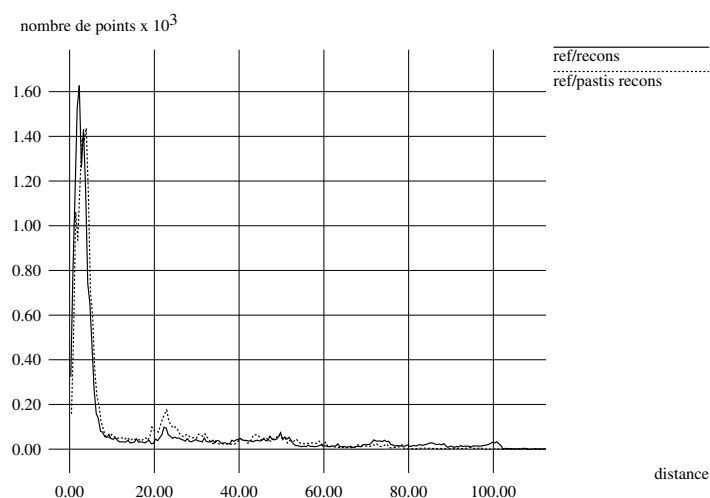


FIG. 3.20 - *Histogrammes des déplacements entre le modèle et les reconstructions obtenues avec notre méthode et celle de Serra.*

	notre méthode	méthode pastis
taille de la partie sans correspondance	32.5%	38.0%
biais moyen sur partie en correspondance	2.93 mm	3.26 mm
écart type sur partie en correspondance	1.55 mm	1.36 mm

TAB. 3.2 - *Biais et erreurs moyennes sur la partie en correspondance et taille de la partie sans correspondance entre la mire de référence obtenue par calibration et la reconstruction 3D de la mire.*

les figures 3.18.

3.6 conclusion

Notre méthode de reconstruction de facettes 3D est globale et ne nécessite pas de faire un appariement au niveau des pixels ou des contours. Cherchant la plus grande précision possible, nous avons prouvé qu'une méthode sans appariement fondée sur les moments géométriques des régions ne peut être exacte que dans le cas où les caméras ont une géométrie rectifiée.

Le fait que la méthode utilise des images rectifiées n'altère en rien ces performances puisque à partir d'un couple quelconque d'images, il est possible de recalculer une paire rectifiée, quand les deux caméras sont calibrées. Nous avons en particulier montré qu'il était plus simple et plus précis de rectifier l'image, puis d'appliquer notre reconstruction, plutôt que

d'approximer la transformation perspective par une projection para- ou ortho perspective.

Les méthodes proposées, fondées sur la géométrie ou sur la photométrie, ont été testées avec succès sur différentes scènes réelles. Elles sont robustes aux erreurs de segmentation et précises puisqu'elles permettent d'atteindre une précision équivalente aux résultats d'une méthode sous-pixel sans explicitement y avoir recours. De plus, leur combinaison permet de traiter les cas de facettes occultées.

De surcroît, pour chaque méthode, nous avons tiré profit de la simplicité à formuler les problèmes de reconstruction de facettes, lorsqu'on travaille dans l'*espace des disparités* explicité dans ce chapitre.

Notre reconstruction utilise la primitive face 3D. Mais, chaque primitive ayant ses avantages et ses limites, une perspective intéressante à l'avenir, nous semble de combiner les facettes planes avec les contours pour obtenir la reconstruction la plus complète possible.

Chapitre 4

Méthodes Floue et ICP pour un recalage 3D plus précis et robuste

4.1 Introduction

Le recalage est un sujet qui suscite un intérêt renouvelé de par ses applications dans de nombreux domaines aussi différents que l'imagerie médicale tridimensionnelle [Aya93], l'imagerie aérienne [HH94] ou la cartographie [KPLR91], et plus généralement en modélisation automatique 3D. Nous ne faisons pas dans ce chapitre de différences entre le recalage à l'aide d'un modèle géométrique des données 3D et la concaténation de reconstructions 3D obtenues sous différents angles de vue. En effet, nous entendons, ici, par modèle géométrique, une description complète géométriquement de l'objet observé. La reconstruction, et le modèle géométrique qui est recalé, sont donc représentés par les mêmes types de structures. La description géométrique étant supposée connue a priori, on peut parler de manière générale de deux ensembles de données à recaler.

La difficulté du recalage est que les deux ensembles sont incomplets. La reconstruction d'un objet est toujours partielle, à cause du point de vue, des occultations. Inversement, les erreurs d'analyse préliminaires, la présence d'un fond ou d'autres objets reconstruits font que le modèle est incomplet. Le recalage est donc un problème de reconnaissance où l'estimation d'un déplacement rigide est couplée à une recherche d'appariement entre les deux ensembles de données.

Les méthodes de recalage peuvent se classer en deux catégories principales : suivant que la recherche est faite par accumulation avec une sélection parmi tous les appariements dans une première étape qui est suivie de l'obtention du déplacement optimum, ou que la méthode procède itérativement en combinant des étapes de mise en correspondance et des étapes d'estimation.

Le premier type de méthode s'appuie sur la géométrie locale pour construire l'ensemble des hypothèses valides de mises en correspondance. La difficulté est alors de sélectionner le déplacement qui valide le plus d'hypothèses d'appariements. Celle-ci correspond à un déplacement qui recale une partie commune aux deux ensembles de données 3D. On peut classer dans cette catégorie les méthodes de recalage fondées sur la transformation de Hough généralisée [BB82] avec classification [DK86, Sto87], et les tables de hachage [RH91, GA92].

Or celles-ci ont souvent le désavantage d'être facilement perturbées par de faux appariements ou des occultations importantes, lors de l'estimation du déplacement. Ces méthodes sont donc souvent peu précises.

La deuxième classe de méthodes réalise l'étape de mise en correspondance et de recalage de manière itérative, ce qui permet de remettre l'appariement en cause à chaque étape. Il n'est donc pas gênant que la mise en correspondance ne soit qu'approximative dans les premières étapes de l'algorithme. On peut citer dans cette catégorie les méthodes fondées sur le schéma hypothèse vérification comme les arbres d'interprétation [FH83, HB84, GLP87, Aya88, WI95], et les approches 'Iterative Closest Point' robustes décrites dans la section 1.6. En effet, la fonction 'plus proche voisin' convient bien pour faire les appariements. Néanmoins, avec cette méthode, l'appariement initial ne doit pas être trop perturbé par les faux appariements, au risque de converger vers une solution absurde. Ces méthodes nécessitent donc que le déplacement entre les deux ensembles de données à recaler ne soit pas trop important. Cette contrainte limite la généralité de ce type de méthodes, mais les recalages obtenus sont généralement très précis.

Dans ce chapitre, nous proposons de combiner les deux types de méthodes précédemment décrits pour tirer profit de leurs avantages complémentaires : la possibilité de grands déplacements avec une méthode d'accumulation et la précision avec une méthode itérative de type ICP robuste.

De manière générale, nous pouvons distinguer trois critères de robustesse pour une méthode qui procède par minimisation :

- la robustesse au bruit sur les données, le bruit étant souvent modélisé par une variable aléatoire Gaussienne centrée,
- la robustesse aux points aberrants qui sont des données parasites non significatives,
- et la robustesse à l'initialisation ou robustesse à la diversité des jeux de données.

Dans la section 2, nous présentons un schéma en trois étapes d'une méthode robuste au bruit, aux points aberrants et à l'initialisation, pour faire le recalage 3D par accumulation. La première étape consiste à créer l'ensemble des déplacements qui peuvent superposer une partie des deux ensembles 3D (section 4.2.1). Nous introduisons un premier filtrage des appariements à l'aide d'un coefficient de similarité associé à chacun pour que seules des primitives assez semblables soient considérées. La deuxième étape consiste à déterminer les déplacements significatifs dans cet ensemble (section 4.2.2). Nous avons, pour cela, développé un algorithme de classification floue fondée sur la méthode de Davé ayant la particularité de gérer les données bruitées et les points parasites. Ceci nous a permis d'approcher de manière robuste les solutions optimales. A ce stade, une dernière étape est nécessaire pour réaliser un recalage fin et atteindre un résultat plus précis (section 4.2.3). Dans la section 4.3, les possibilités de la méthode sont illustrées sur quelques exemples de jeux de données réelles. Enfin, dans cette dernière section, l'outil de mesure de la similarité de deux ensembles de données 3D du chapitre précédent est utilisé pour valider nos résultats de recalage.

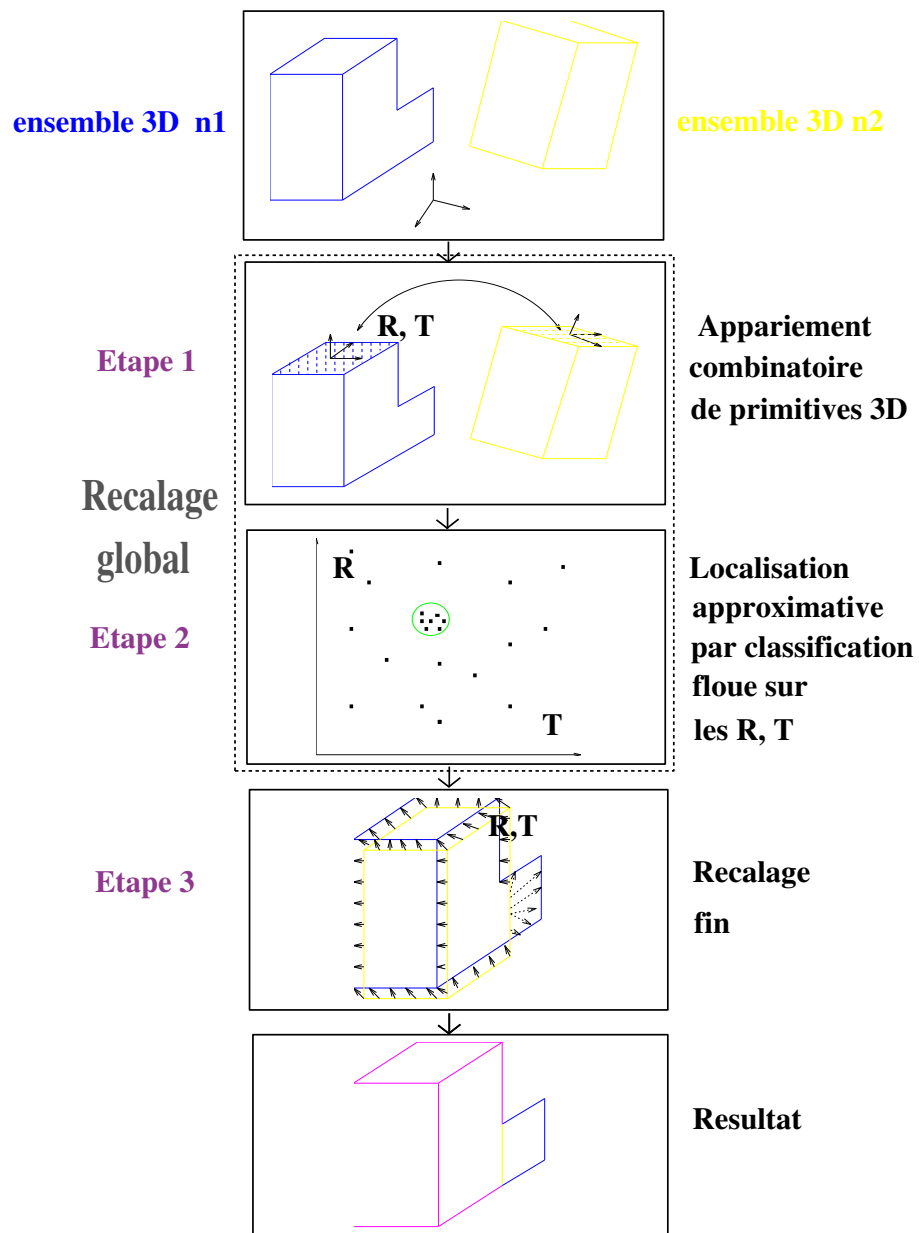


FIG. 4.1 - Approche générale du recalage 3D mixte. (R, T) est un déplacement. Un coefficient de confiance φ est accordé à chaque déplacement.

4.2 Schéma général du recalage 3D avec une approche floue

Dans ce chapitre, nous avons utilisé des facettes puisque la reconstruction est obtenue sous la forme de facettes, comme d'ailleurs les modèles CAO de nos objets. Mais de nombreuses autres primitives sont aussi utilisables, selon le type des données manipulées : base 3D de 4 points [RH91], couple de segments, ou repère local de Frénet sur des courbes [GA92].

La méthode réalise les trois étapes suivantes (figure 4.1) : un recalage approché mais robuste par une approche floue après une phase d'appariement combinatoire, puis un raffinement itératif du recalage par une méthode de mise en correspondance vers le plus proche voisin.

4.2.1 Mise en correspondance de primitives 3D

La première étape de la méthode consiste à faire la mise en correspondance 3D de manière à explorer toutes les possibilités. Un ensemble 3D est constitué d'entités géométriques comme des points, segments, courbes ou facettes. Il faut choisir la primitive de base à apparier, de façon à permettre les deux estimations suivantes entre deux primitives quelconques :

- calcul des déplacements qui les superposent,
- calcul d'une valeur de ressemblance entre les deux primitives.

Une mesure importante de la combinatoire du problème d'appariement est le rapport SNR du nombre de mises en correspondance justes sur leur nombre total. Par exemple, si l'on suppose que le nombre de primitives dans l'ensemble à recaler est n_{prim} et que l'autre ensemble lui est identique à un déplacement près, le nombre total d'appariements possibles est $n_{prim}^2 n_{depla}$, alors qu'il n'y a que n_{prim} appariements justes. Le rapport SNR est donc dans ce cas égal à :

$$SNR = \frac{1}{n_{prim} n_{depla}} \quad (4.1)$$

Or, ce rapport décroît rapidement avec le nombre de primitives n_{prim} . Il est donc préférable de choisir des primitives relativement abstraites, qui sont en petit nombre comme les facettes, pour limiter la combinatoire.

Pour éviter de faire l'appariement des deux faces au niveau des points de contours, avec tous les difficultés que cela implique, nous avons choisi de le faire par une estimation globale qui se passe des appariements. Détaillons à présent notre choix pour faire le calcul des déplacements et du facteur de ressemblance dans le cas d'une paire de faces. Une facette peut être partiellement caractérisée par son barycentre et ses axes principaux d'inertie. Ce point et les axes normalisés fournissent un repère propre à la face. Il devient alors possible de calculer les déplacements qui superposent deux faces quelconques à partir de ces repères propres (figure 4.2). Il y a plusieurs déplacements possibles pour un même couple de facettes, mais ce nombre est fini dans les cas non dégénérés. Soit n_{depla} ce nombre. Par exemple (voir la section 1.4.2), pour des rectangles, $n_{depla} = 4$, alors que pour des carrés $n_{depla} = 8$. Le

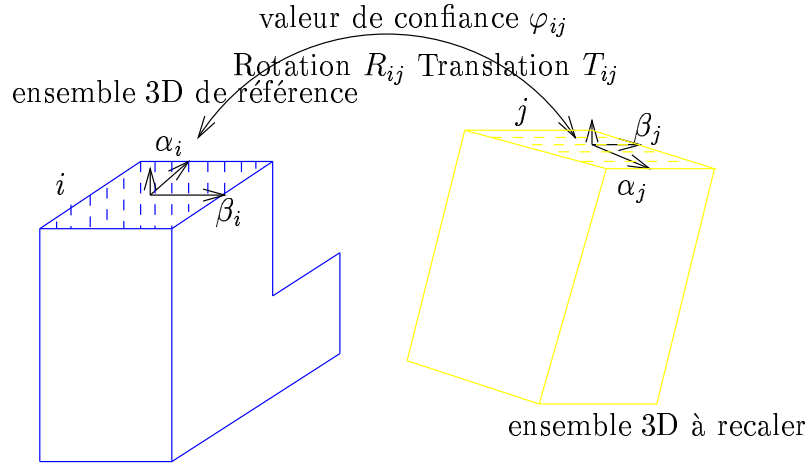


FIG. 4.2 - Appariement des faces i et j de deux ensembles de faces. Pour une paire (i, j) de rectangles. Quatre déplacements (R_{ij}, T_{ij}) sont au moins possibles, comme le montre la figure 1.1.

cas dégénéré est celui où la face est circulaire, puisque les axes propres sont connus à une rotation près.

De plus, on obtient par ce calcul deux valeurs intrinsèques à la face i qui sont la longueur des deux plus longs axes principaux d'inertie α_i et β_i . À partir de ces données invariantes par déplacement rigide, il est possible d'affecter à chaque appariement une valeur de confiance qui décrit la qualité de cet appariement. Dans notre exemple, nous avons choisi de combiner les valeurs des axes principaux d'inertie, pour un couple de faces (i, j) , de la manière suivante :

$$\varphi_{ij} = \min\left(1, \frac{e_{recons}}{|\alpha_i - \alpha_j|}\right) \min\left(1, \frac{e_{recons}}{|\beta_i - \beta_j|}\right) \quad (4.2)$$

pour ainsi avoir une valeur entre 0 et 1 qui mesure leur ressemblance. Bien sûr, d'autres options peuvent être choisies. Le paramètre e_{recons} est l'erreur moyenne des données 3D (section 4.3.6). Plus les deux facettes sont de géométries différentes, et plus la valeur du critère est proche de 0.

La rapidité de la décroissance avec l'écart des longueurs des axes est réglée par le coefficient e_{recons} (voir figure 4.3). Les appariements qui ont une valeur de similarité très faible (entre 0 et 0.1) sont ignorés dans la suite car ils ne présentent pas d'intérêt et cela évite l'explosion combinatoire.

En conséquence, après l'étape de mise en correspondance, on dispose d'un ensemble d'appariements de facettes, où sont associés à chaque paire un facteur de ressemblance et des déplacements.

4.2.2 Localisation 3D approchée

Dans la première étape, un ensemble d'appariements a été obtenu, où chaque élément est associé à des déplacements et un facteur de confiance. Un déplacement 3D peut être

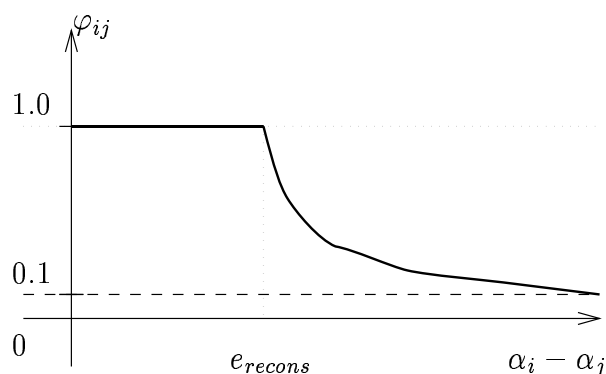


FIG. 4.3 - Variations de la valeur de confiance φ_{ij} avec la différence de la longueur du premier axe propre $\alpha_i - \alpha_j$. e_{recons} est l'erreur moyenne sur les données 3D.

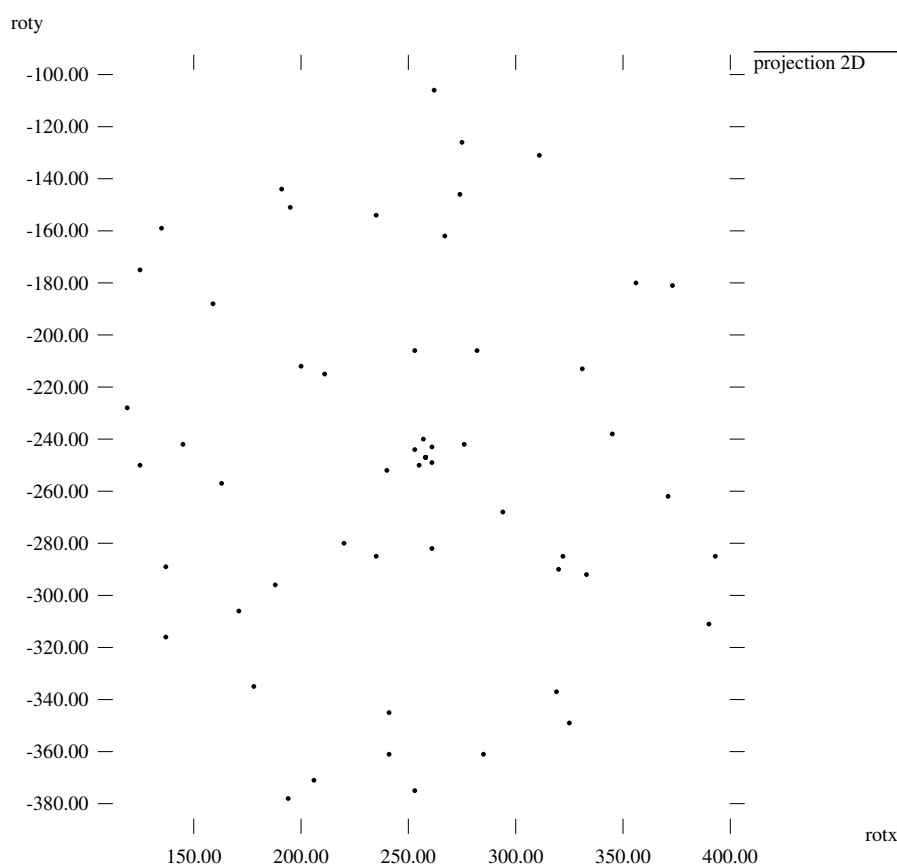


FIG. 4.4 - Projection 2D de l'ensemble des rotations 3D à classifier pour obtenir le point d'accumulation central correspondant à une hypothèse intéressante pour le recalage.

représenté de diverses manières comme un point X de \mathbb{R}^6 , avec trois termes de translation et les trois angles d'Euler, par exemple. Nous extrayons alors l'ensemble des déplacements 3D qui superposent au moins une facette. On réalise donc le changement d'indice suivant :

$$(R_{ij}, T_{ij}) \rightarrow X_{k(i,j)} \quad (4.3)$$

$$\varphi_{ij} \rightarrow \varphi_{k(i,j)}$$

Dans cet ensemble $\{X_k\}_k$ de déplacements, *un point d'accumulation correspond à une hypothèse intéressante de recalage* entre l'une des parties commune aux deux ensembles 3D (figure 4.4). C'est le principe d'accumulation propre aux méthodes de type transformée de Hough généralisée. L'avantage d'utiliser une méthode de classification est de ne pas être pénalisé par la discrétisation de l'espace d'accumulation comme dans la transformée de Hough (section 4.5). Un deuxième intérêt d'utiliser une classification est qu'elle manipule en même temps plusieurs classes. Elle gère donc plusieurs hypothèses de déplacement en parallèle, en essayant de recouvrir au maximum l'ensemble des déplacements possibles.

Comme les méthodes de classification floue sont particulièrement robustes [KF92, KN95], elles sont appropriées pour sélectionner ces points d'accumulation. Alors, chaque point d'accumulation correspond au prototype d'une classe.

Convergence améliorée

Nous présentons ici un exemple simple en dimension un, qui montre que la méthode de classification floue converge en effet plus facilement que la méthode classique. Cette propriété provient du fait que la fonction à minimiser est plus lisse dans le cas flou. Pour l'illustrer, nous avons tracé la contribution au critère d'erreur, d'un point placé à l'origine, dans les deux méthodes. Soit $x_1 = 0$ ce point, et soit t_1 et t_2 la position des prototypes des deux classes. Dans le cas classique, la contribution à l'erreur du point x_1 est :

$$e_{classique} = \min(d^2(x_1, t_1), d^2(x_1, t_2)) = \min(t_1^2, t_2^2)$$

Dans le cas flou, d'après l'équation (1.29), elle est :

$$e_{FCM} = u_{11}^m d^2(x_1, t_1) + u_{12}^m d^2(x_1, t_2) = u_{11}^m t_1^2 + u_{12}^m t_2^2$$

avec :

$$u_{11} = \frac{1}{1 + \left(\frac{t_1}{t_2}\right)^{\frac{2}{m-1}}} \quad \text{et} \quad u_{12} = \frac{1}{1 + \left(\frac{t_2}{t_1}\right)^{\frac{2}{m-1}}}$$

La figure 4.5 montre ces deux erreurs lorsque la deuxième classe est de position figée ($t_2 = 2$). On remarque que, au contraire du cas flou, la fonction à minimiser présente des paliers et des ruptures de pente. Le facteur flou m s'interprète donc comme le facteur de lissage de la fonction d'erreur à minimiser.

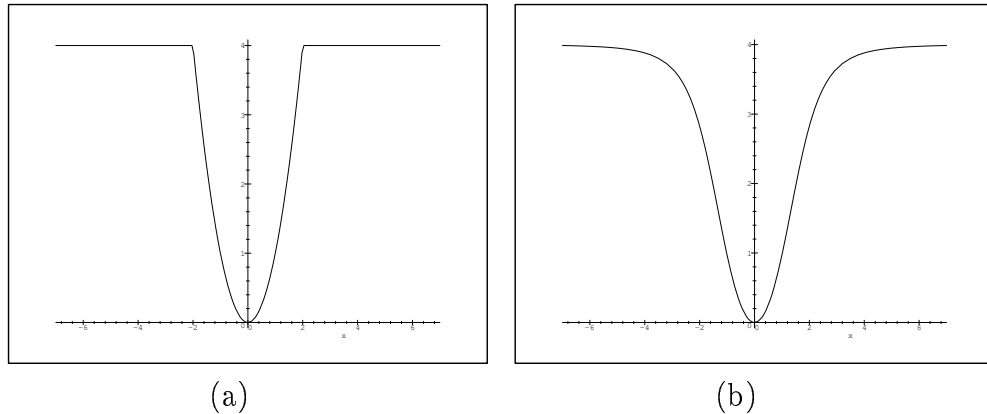


FIG. 4.5 - (a) variations de $e_{classique}(t_1)$ avec $t_2 = 2$. (b) variations de $e_{FCM}(t_1)$ avec $t_2 = 2$ et $m = 1.5$. On remarque que la le critère flou est plus lisse que celui de la classification classique.

Points parasites

Dans le cadre du recalage 3D, l'approche floue introduite à la fin du chapitre 1 consiste à considérer l'ensemble des appariements possibles comme l'ensemble d'entrée de la classification, et l'espace des déplacements comme l'espace de travail. Dans ce contexte, les classes significatives sont lourdement perturbées par un nuage de points parasites. C'est la raison qui nous amène à tenir compte davantage, dans notre démarche, de l'aspect modélisation du bruit.

De fait, nous avons utilisé la méthode de classification robuste proposée par Davé (cf. section 1.6.2) où une classe supplémentaire nommée 'classe bruit' est introduite pour récolter tous les points parasites qui perturbent l'estimation des classes. La 'classe bruit', ou classe de rejet, est définie par un prototype qui est à égale mesure de similarité de tous les points de l'espace des caractéristiques. Cette artifice permet de fixer l'épaisseur de la zone d'influence de chaque prototype. L'algorithme est contrôlé avec un paramètre $p_{pertinent}$ qui indique l'importance numérique des données parasites. Nous avons testé la méthode, qui donne d'excellents résultats quand le rapport SNR du nombre de points à ne pas rejeter sur le nombre total est supérieur à 10%. Malheureusement, dans le cadre du recalage 3D, le nombre de points parasites est généralement beaucoup plus important (voir l'équation 4.1). Sans l'ajout d'informations supplémentaires, l'algorithme est incapable de départager les données pertinentes du nuage de points aberrants, sauf dans les situations les plus simples.

Intégration de connaissances dans la classification floue

Comme la proportion de déplacements non valides est très importante, il faut augmenter l'efficacité de la classification. Pour cela, nous proposons l'introduction de connaissances géométriques sur les facettes traduites par un facteur de confiance φ_i associé à chaque point de l'espace des transformations (équation (4.2)). L'idée est de pondérer la participation de chaque point de l'espace de classification par ce coefficient. Nous minimisons donc la

fonctionnelle suivante :

$$e_{WFCM} = \sum_{k=1}^c \sum_{i=1}^n \varphi_i u_{ik}^m d^2(X_i, T_k) \quad (4.4)$$

où $d(X_i, T_k)$ est la distance entre le point X_i et le prototype T_k de la classe k . u_{ik} est le degré d'appartenance du point k à la classe i . u_{ik} est toujours positif et sa somme sur toutes les classes est égale à 1. Cette contrainte $\sum_{k=1}^c u_{ik} = 1$ force la méthode de classification à expliquer la totalité des données par c sous-ensembles flous (voir [Bez81] pour des explications plus complètes sur cet aspect des méthodes de classification floues).

L'algorithme de classification se déduit de la fonctionnelle (4.4) qui peut être résumé de la manière suivante :

- **étape 0** : Fixer le nombre de classes c et le facteur flou m . Calculer la position initiale des centres T_k des classes en utilisant l'algorithme des FCM, par exemple. Donner le paramètre $p_{pertinent}$ qui spécifie l'étendue du bruit.
- **étape w** : Générer une partition en utilisant l'équation suivante du degré d'appartenance :

$$u_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d(X_i, T_k)}{d(X_i, T_j)} \right)^{\frac{2}{m-1}}}$$

- **étape w'** : Calculer les nouveaux centres des classes en utilisant l'équation suivante (pour $k = 1$ à $c - 1$) :

$$T_k = \frac{\sum_{i=1}^n \varphi_i u_{ik}^m X_i}{\sum_{i=1}^n \varphi_i u_{ik}^m}$$

- **étape w''** : S'arrêter quand la partition est stable, sinon retourner à l'**étape w**.

X_i est un élément de l'ensemble de n points dans \mathbb{R}^p ($p = 6$ dans notre cas) et le facteur de confiance associé est φ_i . m est le facteur flou généralement fixé à 1.5 dans nos expériences. $d(X_i, T_k)$ est défini comme la distance euclidienne entre le point X_i de l'ensemble des données et le centre T_k de la classe k , pour $k = 1$ à $c - 1$. Pour la classe de bruit, la définition de la distance est (on peut aussi la choisir fixe égale à δ comme dans le chapitre 5, mais alors la méthode est moins adaptative car il n'y a plus d'invariance par rapport à l'échelle de la taille des points d'accumulation) :

$$d(T_c, X_i)^2 = \frac{p_{pertinent}}{n(c-1)} \sum_{k=1}^{c-1} \sum_{j=1}^n d^2(X_i, T_j) \quad (4.5)$$

Démonstration de la convergence

La convergence de la méthode de classification vers un minimum local, dans ce cadre étendu, est conservée. Nous présentons dans cette section uniquement la ligne directrice de la démonstration de la convergence vers un minimum local de l'algorithme FCM, lorsque que la valeur de confiance φ_k est introduite. Cette démonstration est une adaptation de la convergence de FCM [Bez81, BHST87] avec la même condition sur le facteur flou : $m > 1$. Une extension similaire peut être faite à partir de la preuve de l'algorithme de Davé.

Soit la matrice $\underline{U} = (u_{ik})$ et le vecteur $\underline{T} = (T_k)$. Nous voulons minimiser la fonctionnelle suivante :

$$e_{WFCM}(\underline{U}, \underline{T}) = \sum_{k=1}^c \sum_{i=1}^n \varphi_i u_{ik}^m d^2(X_i, T_k) \quad \text{avec} \quad \sum_{k=1}^c u_{ik} = 1 \quad \text{et} \quad u_{ik} \geq 0 \quad (4.6)$$

où $d(X, Y)$ est la distance Euclidienne entre X et Y . La démonstration de la convergence théorique est fondée sur l'application du théorème de Zangwill qui assure la convergence après un nombre infini d'itérations comme dans une méthode de descente de gradient ou de Newton.

Premièrement, nous fixons \underline{T} . Puisque les colonnes de la matrice \underline{U} sont indépendantes, nous déduisons que :

$$\min_{\underline{U}}(e_{WFCM}) = \sum_{i=1}^n \varphi_i \min_{\underline{U}} \left(\sum_{k=1}^c u_{ik}^m d^2(X_i, T_k) \right)$$

Alors, pour chaque terme, il faut faire une minimisation sous les contraintes $\sum_{k=1}^c u_{ik} = 1$ et $u_{ik} \geq 0$. En appliquant les multiplicateurs de Lagrange, on obtient le minimum sous la forme suivante, après calcul :

$$u_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d(X_i, T_k)}{d(X_i, T_j)} \right)^{\frac{2}{m-1}}} \quad (4.7)$$

Deuxièmement, nous fixons \underline{U} . Alors, il faut minimiser le terme suivant :

$$e_{WFCM}(\underline{T}) = \sum_{k=1}^c \sum_{i=1}^n \varphi_i u_{ik}^m (X_i - T_k)^t (X_i - T_k)$$

Pour chaque k , il est nécessaire que les dérivées $e'_{WFCM}(T_k, V)$ suivant chaque vecteur unité $V \in \mathbb{R}^c$ soient nulles :

$$\begin{aligned} e'_{WFCM}(T_k, V) &= -2 \sum_{i=1}^n \varphi_i u_{ik}^m (X_i - T_k)^t V = 0 \quad \forall V \\ &\Leftrightarrow \sum_{i=1}^n \varphi_i u_{ik}^m (X_i - T_k) = 0 \end{aligned}$$

D'où, nous déduisons que :

$$T_k = \frac{\sum_{i=1}^n \varphi_i u_{ik}^m X_i}{\sum_{i=1}^n \varphi_i u_{ik}^m} \quad (4.8)$$

Le prototype T_k peut être interprété comme le centre de gravité de l'ensemble $\{X_i\}_{1,\dots,n}$, où chaque point est pondéré par $\varphi_i u_{ik}^m$. Il faut noter que les équations (4.7) et (4.8) permettent d'écrire les étapes 2 et 3 qui sont au coeur de l'algorithme de classification décrit dans la section 4.2.2.

Une interprétation duale de la fonctionnelle (4.6) est la suivante :

$$e_{WFCM}(\underline{U}, \underline{T}) = \sum_{k=1}^c \sum_{i=1}^n w_{ik}^m d^2(X_i, T_k)$$

où $w_{ik} = \sqrt{\varphi_i} u_{ik}$. Cette formule est la même dans l'algorithme FCM, où les contraintes sur w_{ik} ne sont pas $\sum_{k=1}^c w_{ik} = 1$ mais $\sum_{k=1}^c w_{ik} = \sqrt{\varphi_i}$. Ainsi, nous pouvons interpréter notre généralisation de l'algorithme de classification floue introduisant la pondération par la confiance φ_i sur chaque point i , comme une méthode de classification avec une contrainte non homogène sur les degrés d'appartenance. Ceci révèle un lien intéressant avec les approches possibilistes de la classification [KK93].

En résumé, l'introduction du coefficient de confiance permet de préserver la proportion des points valides parmi les points parasites et d'augmenter ainsi la qualité de la classification. Plus généralement, il est possible grâce à cette mesure de représenter des connaissances supplémentaires associées à chaque point de l'espace de classification, quelque soit la méthode utilisée.

4.2.3 Raffinement itératif du recalage

A ce stade, on dispose de $c - 1$ hypothèses intéressantes de déplacements qui recalent des parties des deux jeux de données 3D. La première étape utilise des facettes qu'elles supposent non occultées pour faire l'appariement. Les déplacements obtenus par la deuxième étape peuvent donc être légèrement biaisés par des appariements de faces occultées. Il est donc nécessaire d'avoir une étape de recalage plus fin au niveau des points pour traiter les cas d'occultation des faces. Dans ce contexte, une méthode itérative de recalage rigide avec mise en correspondance vers le plus proche voisin est bien adaptée. Lors de cette étape, l'algorithme travaille avec les points de contour des faces, pour avoir même en présence de faces occultées une bonne précision du recalage.

L'algorithme itératif de recalage ICP brut, qui est décrit dans la section 1.6.1, n'est pas adapté à notre problème, car il nécessite des données qui se recouvrent complètement. Pour remédier à cela, les mises en correspondance sont analysées statistiquement à chaque étape de l'algorithme pour ne retenir que les plus pertinentes au sens de la longueur de l'appariement. Les faux appariements sont alors en grande partie ignorés, ce qui permet le recalage avec des ensembles 3D se recouvrant partiellement (figure 4.6). Lors de cette analyse statistique, un paramètre $p_{recouvr}$ est introduit, qui indique un ordre de grandeur du pourcentage de points

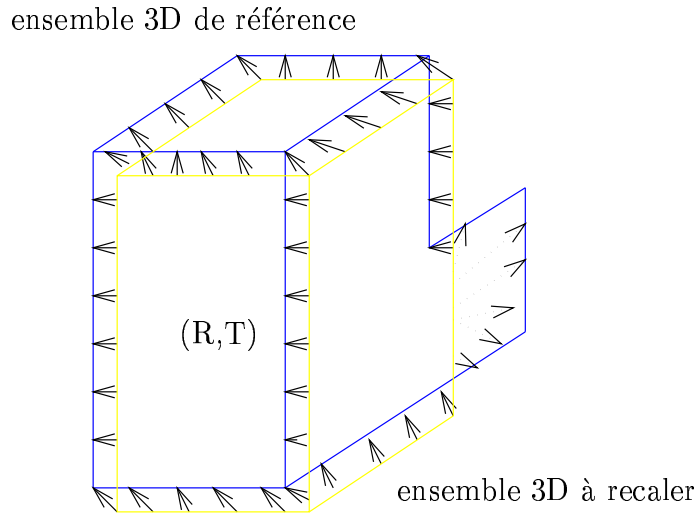


FIG. 4.6 - *L'appariement est réalisé avec le plus proche voisin, et suivant la longueur du vecteur déplacement, la correspondance est prise ou n'est pas prise en compte.*

appariables (% du recouvrement). Nous avons utilisé la version robuste de l'ICP développée par J.-M. Vézien [Véz95] qui est fondée sur un filtre médian, car elle est plus économique en calculs que la version robuste par classification floue présentée dans la section 1.6.2, pour un résultat équivalent.

L'étape de recalage fin est appliquée sur chaque hypothèse de déplacement fourni par la classification. Aussi, la comparaison des résidus après minimisation permet de trier les recalages les plus intéressants. Bien que les méthodes de type ICP soient généralement assez coûteuses en temps de calcul, le résultat souhaité s'obtient dans notre cas assez rapidement car la position initiale est suffisamment proche de la solution exacte.

4.2.4 Résumé

La mise en correspondance, le recalage grossier puis fin, forment les trois étapes de la procédure de recalage robuste. Notre méthode suppose la connaissance d'un certain nombre de paramètres qui sont les suivants :

e_{recons} : écart non significatif pour comparer la longueur des axes d'inertie.

$p_{pertinent}$: un paramètre lié au rapport du nombre d'appariements justes de facettes sur leur nombre total. C'est le seul paramètre réellement critique du point de vue de la qualité des résultats.

c : le nombre de classes, ou de déplacements potentiels à estimer.

$p_{recouvr}$: la proportion de points non appariables lors du recalage local fin.

Souvent, les reconstructions sont très partielles à cause du point de vue, ou des défauts du processus d'analyse d'image, par exemple. Pourtant, la méthode proposée a l'avantage de permettre le recalage 3D d'un modèle d'objet isolé sur ce type de reconstructions partielles. Ainsi, nous avons pu obtenir des résultats sur des ensembles qui possèdent une partie commune inférieure à 20% de la taille du plus grand ensemble à recalcer.

4.3 Résultats

Notre méthode de recalage a été validée complètement sur les objets synthétiques où le déplacement à obtenir est parfaitement connu. Les tests sur les données réelles ont aussi permis de valider la méthode sur ce type de données, dans la mesure où le déplacement exact est inconnu.

4.3.1 Réglage des paramètres

Dans la pratique, comment choisir les paramètres de l'algorithme :

e_{recons} : e_{recons} indique la tolérance observable sur les données d'entrée du recalage. Ainsi, l'examen de la précision obtenue par les algorithmes ou capteurs qui produisent ces données 3D permet de régler e_{recons} une fois pour toutes.

$p_{pertinent}$: Expérimentalement, nous avons constaté que la valeur de SNR permet d'avoir un bon ordre de grandeur de $p_{pertinent}$ (à un facteur constant près).

c : Le nombre de classes c doit être choisi après l'analyse des symétries de l'ensemble 3D à recalcer qui peuvent conduire à l'existence de plusieurs solutions également équivalentes (voir 4.3.2). Il faut aussi tenir compte d'éventuelles apparitions multiples de l'objet à recalcer dans l'ensemble à interpréter géométriquement. Néanmoins, ce paramètre peut être largement sous évalué sans difficulté : seul un sous-ensemble des solutions possibles est alors obtenu.

$p_{recouvr}$: Ce paramètre est relié à $p_{pertinent}$. Ce pourcentage $p_{recouvr}$ peut être largement sous évalué. Il correspond en fait à une borne supérieure du pourcentage de mises en correspondance de points qui peuvent s'avérer fausses, dans l'étape de recalage fin.

En conséquence, la méthode de recalage est simple d'utilisation puisque contrôlée par le paramètre $p_{pertinent}$ et le nombre $c - 1$ de solutions que l'on désire obtenir. Ceci permet d'envisager une automatisation complète du processus de recalage.

4.3.2 Les objets symétriques

Les situations où l'ensemble à recalcer présente des symétries ou apparaît plusieurs fois sont généralement difficiles à gérer avec les méthodes classiques. Notre méthode présente l'avantage de manipuler l'ensemble des solutions possibles de manière unifiée. En effet, il permet de choisir le nombre de classes c en fonction du nombre de recalages que l'on désire

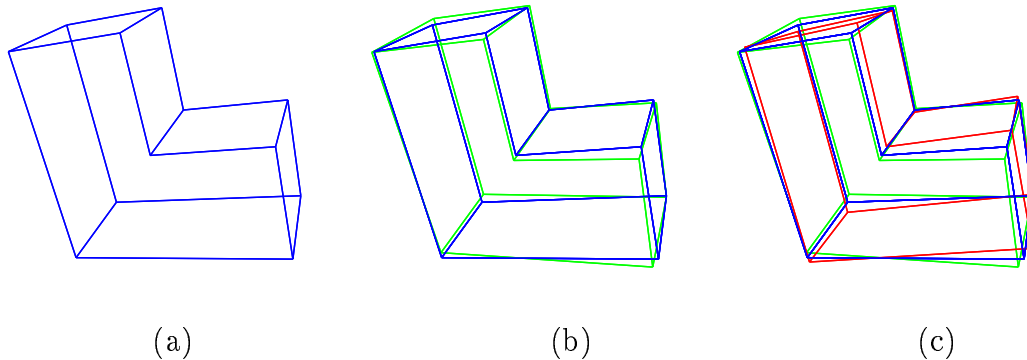


FIG. 4.7 - (a) Objet 3D de synthèse. (b) Une solution de la phase du recalage approché qui possède deux solutions également possibles du fait la symétrie de l'objet par rapport à son axe médian (c).

	p_1	p_2	p_3	t_x	t_y	t_z
valeurs classe 1	0.005	0.882	0.017	0.017	-0.002	0.007
valeurs exactes 1	0.000	1.000	0.000	0.000	0.000	0.000
valeurs classe 2	2.919	-0.012	0.854	0.006	0.016	-0.016
valeurs exactes 2	3.014	0.000	0.884	0.000	0.000	0.000

TAB. 4.1 - Déplacements obtenus par la phase de recalage grossier et les valeurs effectives de la figure 4.7. La rotation est représentée par (p_1, p_2, p_3) qui sont les composantes de l'axe normalisé multiplié par l'angle en radian.

obtenir. La classification floue permet d'éviter la redondance des solutions en contraignant les classes à recouvrir au mieux l'ensemble des déplacements possibles.

Par exemple, dans la figure 4.7, l'objet synthétique à recaler est identique à l'objet de référence, à un déplacement rigide près. Du fait des symétries de cet objet, deux recalages sont possibles qui sont effectivement bien retrouvés. Le tableau 4.1, où sont comparés les déplacements exacts et estimés par la phase de recalage global ($c = 3$, $p_{pertinent} = 0.035$), en témoigne. Le résultat du recalage fin aboutit à une superposition parfaite avec l'objet de la figure 4.7(a).

Plus généralement, la distance moyenne entre les parties communes des deux modèles après le recalage fin, fournie par cette dernière étape, donne une valeur pour trier les solutions suivant leur qualité.

4.3.3 Utilisation de la couleur

Une facette ne possède pas que des caractéristiques propres de nature géométrique. Elle

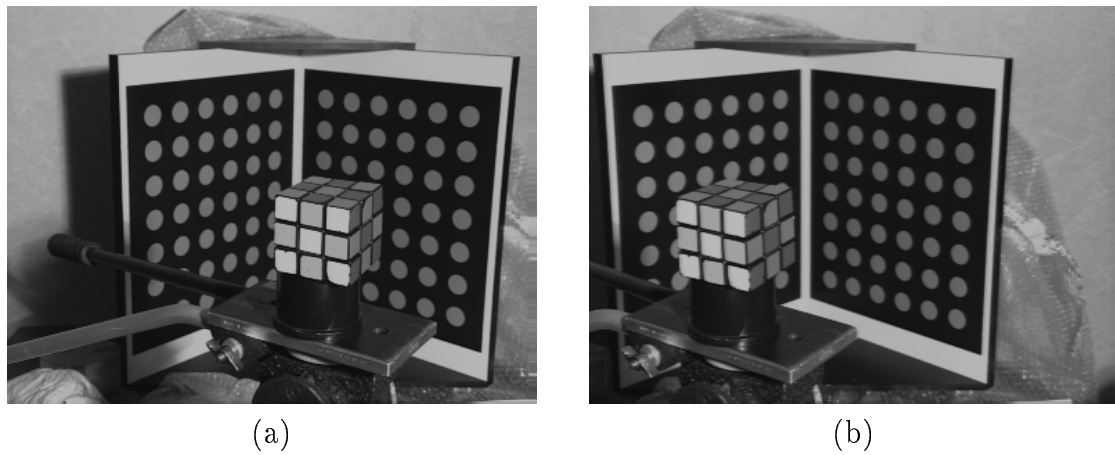


FIG. 4.8 - (a)(b) paire stéréoscopique traitée pour obtenir la reconstruction de figure 4.9(a).

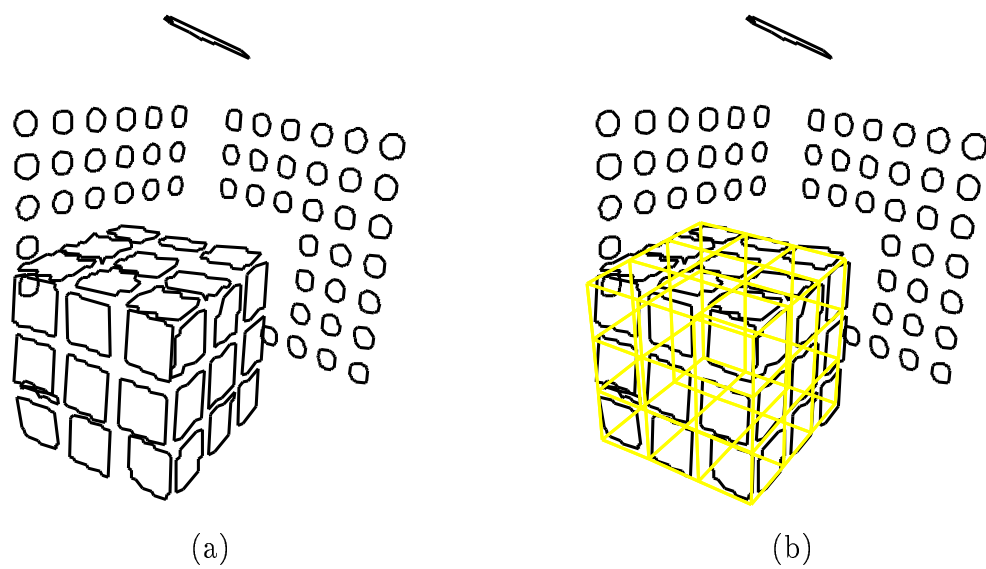


FIG. 4.9 - (a) Vue de la reconstruction 3D obtenue par analyse sur la paire de la figure 4.8. (b) Le modèle parfait du rubik cube est recalé sur la reconstruction.

peut aussi posséder une couleur moyenne. Cette information de couleur, décrite par les composantes de couleurs (I_i^{rouge} , I_i^{vert} , I_i^{bleu}) normalisées de la facette i , peut être intégrée au facteur de confiance (équation 4.2). Nous l'avons fait de la manière suivante :

$$\varphi_{ij} = \min\left(1, \frac{e_{recons}}{|\alpha_i - \alpha_j|}\right) \min\left(1, \frac{e_{recons}}{|\beta_i - \beta_j|}\right) \\ \min\left(1, \frac{e_{couleur}}{|I_i^{rouge} - I_j^{rouge}|}\right) \min\left(1, \frac{e_{couleur}}{|I_i^{vert} - I_j^{vert}|}\right)$$

où $e_{couleur}$ est l'écart à partir du quel on distingue deux couleurs. $e_{couleur}$ est déterminé à partir du pouvoir séparateur des couleurs du système d'analyse.

4.3.4 Recalage avec plusieurs objets

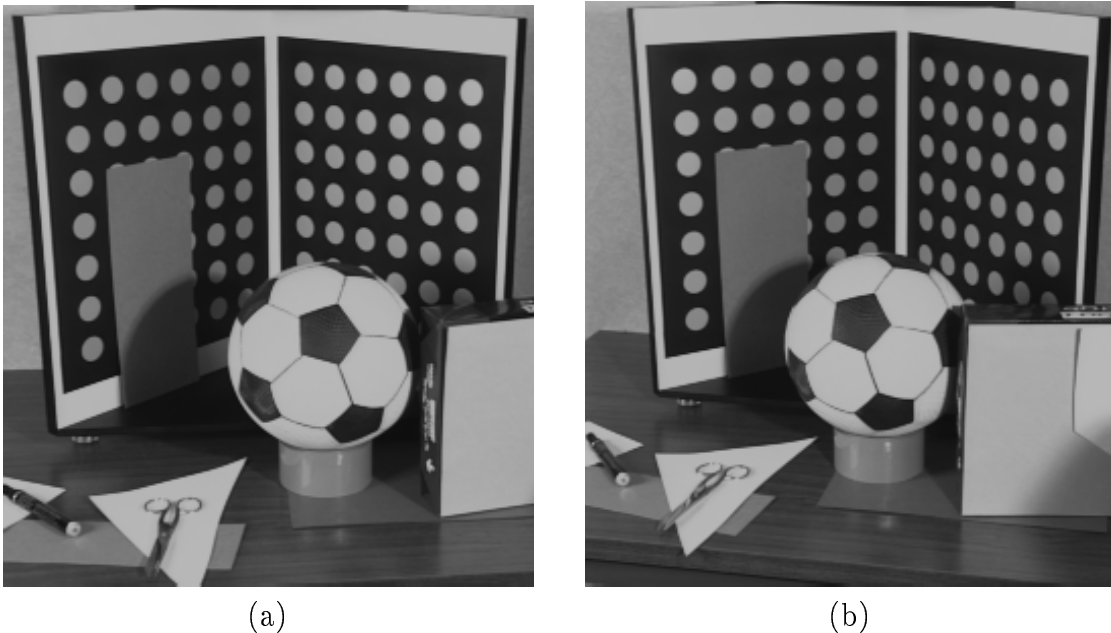


FIG. 4.10 - (a)(b) Images originales de la scène. Les vues des résultats de reconstruction 3D obtenus avec nos algorithmes de stéréovision sont présentées dans la figure 4.11(a)(b).

La méthode décrite plus haut, vue sa robustesse, permet de faire du recalage avec plusieurs objets. A fin d'illustration, le résultat du processus d'extraction de facettes 3D sur les images originales (a) et (b) de la figure 4.10 est visible dans la figure 4.11(a)(b). Cette scène contient deux objets principaux : une mire et un ballon, dont nous avons le modèle 3D complet. Ces deux modèles ont donc été recalés avec succès par notre méthode de recalage floue.

Dans ce cas là, les résultats obtenus en 3D sont suffisamment précis pour que le modèle recalé se reprojette bien sur la paire stéréoscopique. Il n'est donc pas nécessaire d'affiner le recalage par une méthode de 3D/2D qui minimise l'erreur directement entre le modèle et les images.

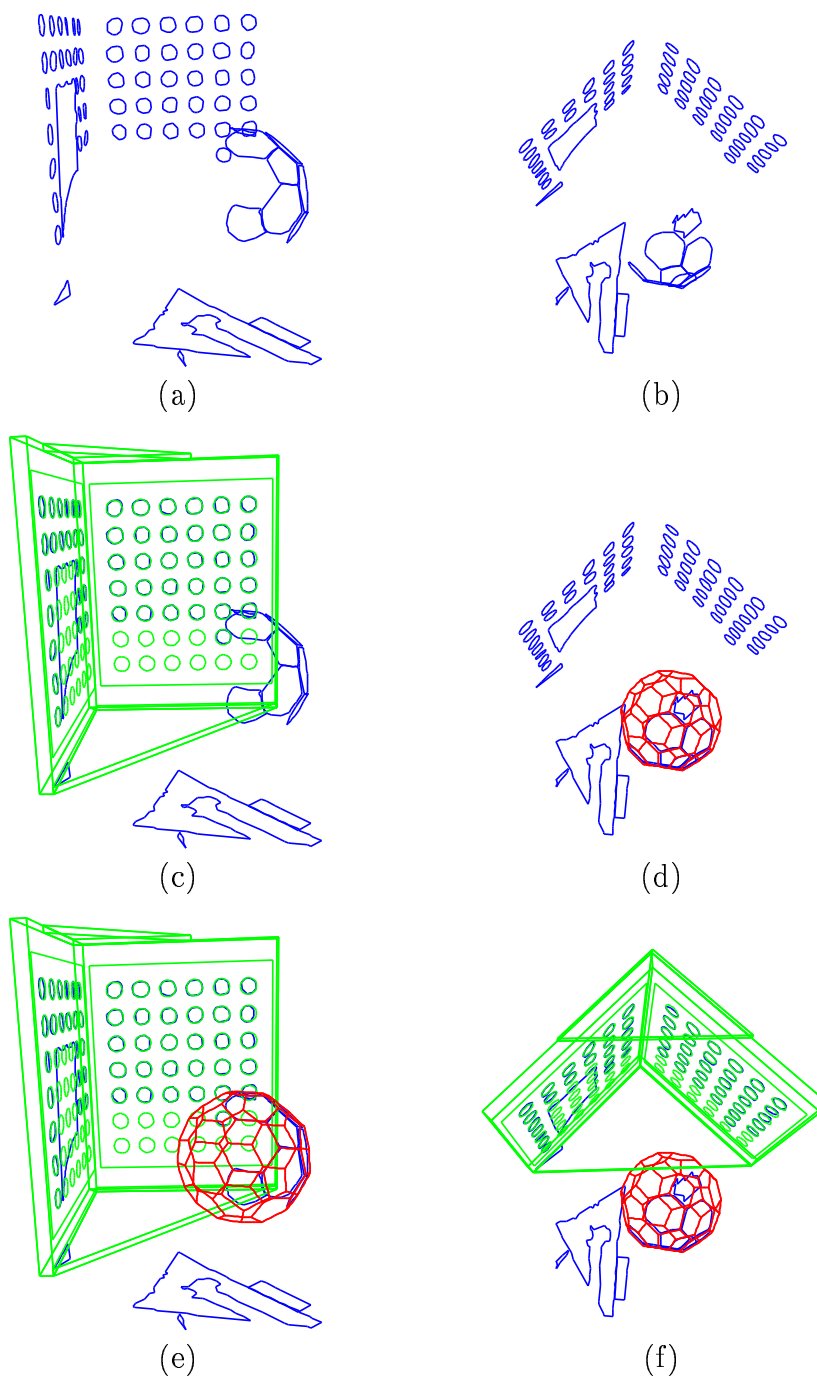


FIG. 4.11 - (a) et (b) sont deux points de vue de la reconstruction 3D obtenue à partir de la paire de la figure 4.10. Dans (c), la mire est recalée sur la reconstruction. Dans (d), seul le ballon est recalé, alors que (e) et (f) montrent les deux objets recalés. Ces résultats ont été obtenus par l'application complète des trois étapes de la méthode pour avoir un recalage très précis.

4.3.5 Validation par rétroprojection

Une fois les principaux objets de la scène recalés, il est possible de reprojeter une des images originales de la paire stéréoscopique sur les modèles 3D recalés pour vérifier visuellement l'exactitude du recalage. Nous avons utilisé l'image comme une diapositive dans un programme de rendu d'images par lancer de rayons. Ce programme nommé Rayshade [FvDFH90] suit les rayons lumineux dans le sens inverse, en partant du point de vue, pour trouver la couleur de chaque pixel de l'image à générer. Pour projeter l'image originale sur les différents modèles 3D recalés qui sont a priori uniformément blancs, nous avons placé une source lumineuse au centre de la caméra originale, et nous avons créé une facette transparente de la taille de la rétine sur laquelle l'image originale est plaquée. Quelques images de synthèse générées sont visibles sur la figure 4.12.

4.3.6 Écart de localisation

Il est important de valider nos algorithmes de recalage 3D en calculant l'écart moyenne entre la reconstruction et le modèle recalé. Nous utilisons pour cela l'outil de comparaison de deux ensembles 3D qui est défini au chapitre précédent.

Dans cette méthode, une classification est réalisée sur les déplacements des points en correspondance entre les deux ensembles. L'algorithme de segmentation floue permet de séparer de manière adaptative, sans difficultés, comme en témoigne l'ampleur du pic de la figure 4.13, la part des données en recouvrement de celle qui est en fausse correspondance.

Une fois l'ensemble 3D segmenté comme dans la figure 4.14, il est aisé de calculer sur la première classe des critères statistiques sur les déplacements. Ainsi, le centre de cette classe s'interprète comme le biais moyen du déplacement entre la partie commune des deux ensembles 3D. Le rayon moyen de la classe est la distance moyenne entre les parties en correspondance. Enfin, la taille relative de la partie sans mise en correspondance est celle de la classe rejet.

taille de la partie sans correspondance	32.1%
biais moyen sur la partie en correspondance	1.78 mm
écart type sur la partie en correspondance	1.11 mm

TAB. 4.2 - *Biais et erreur sur la partie en recouvrement ainsi que la taille de la partie sans correspondance des points de la mire de la figure 4.11(d)(e)(f).*

D'après le tableau 4.2, nous avons obtenu entre la reconstruction 3D de la figure 4.11(a)(b) et la mire recalée, une erreur totale de 2.9 mm. Cette scène présente une profondeur de 1200 mm avec une base stéréoscopique de 340 mm. Théoriquement, l'erreur de reconstruction est donc de 1.2 mm de face et 9 mm selon l'axe de la caméra (voir section 3.5.2 pour avoir des détails sur le calcul). En conséquence, le paramètre e_{recons} du critère (4.2) est fixé autour de 3 mm.

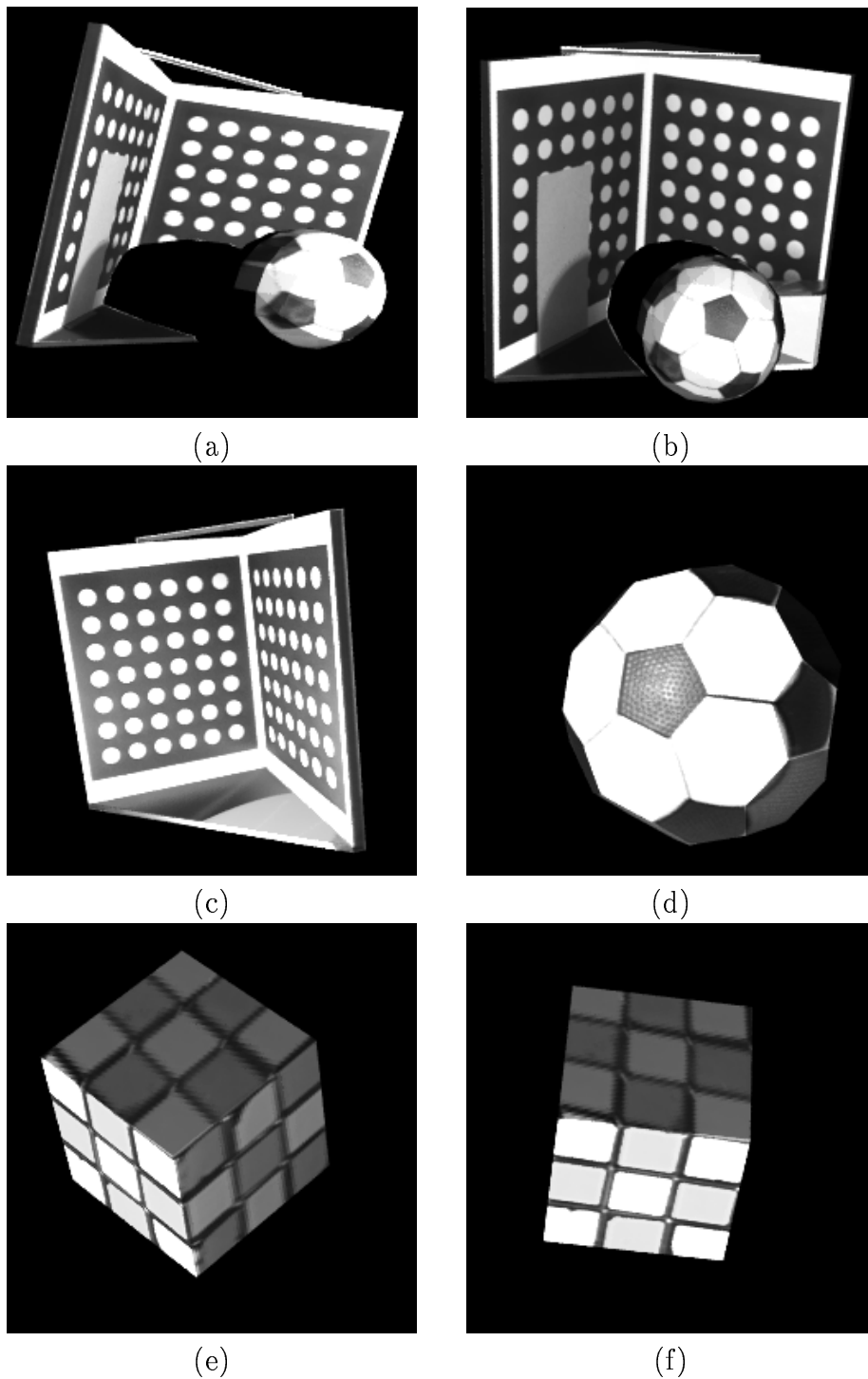


FIG. 4.12 - (a)(b) Point de vue de synthèse des modèles recalés (voir figure 4.11(e)(f)) sur lesquels l'image originale 4.10(a) est projetée. (c) Point de vue synthétique de la mire de calibration seule. (d) Point de vue synthétique du ballon obtenu après l'analyse de la paire de la figure 0.2. (e)(f) Point de vue de synthèse sur la scène du rubik cube de la figure 4.9(b) sur lequel est projetée l'image de la figure 4.8(a).

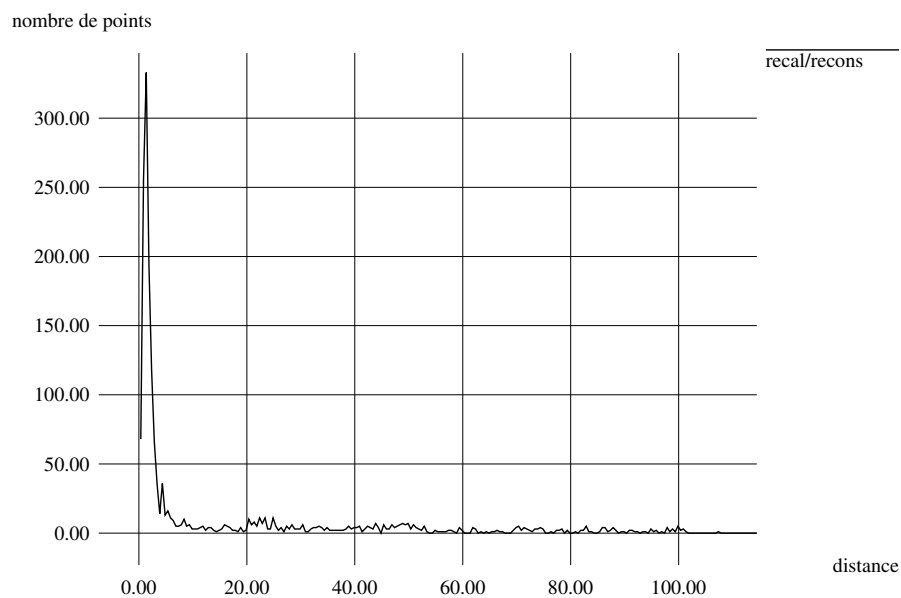


FIG. 4.13 - *Histogramme des longueurs en mm des déplacements entre les points appariés entre les deux ensembles 3D. Dans cette exemple, l'objet de référence est celui de la figure 4.14; le deuxième ensemble est la reconstruction 3D visible dans la figure 4.11(a)(b).*

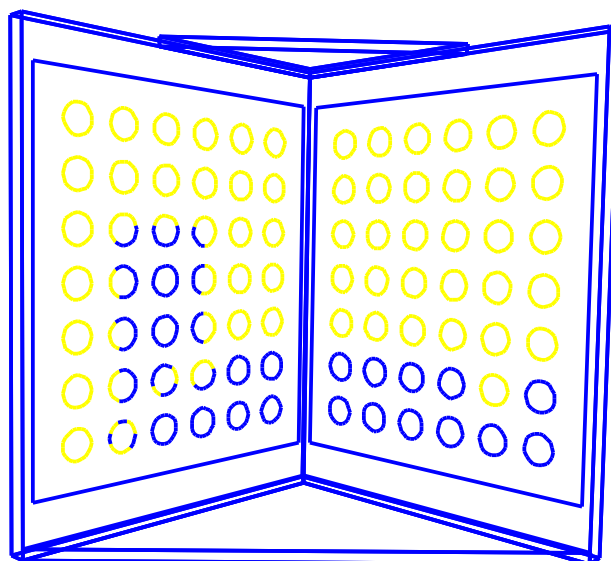


FIG. 4.14 - *Segmentation de l'ensemble de référence en sa partie commune avec la reconstruction (en gris) et sa partie non appariable. La reconstruction 3D associée est visible sur la figure 4.11(a)(b). Comme le recalage de la mire est très proche de la position obtenue par la calibration, ce résultat est très proche de celui de la figure 3.18(b)*

4.4 Conclusion

Nous proposons dans ce chapitre une méthode de recalage de données 3D constituée de trois étapes qui combine aussi bien les avantages complémentaires des méthodes à appariement combinatoire (grands déplacements), et à appariement itéré (précision et occlusions au niveau du point). Les deux premières étapes constituent une méthode par appariement combinatoire robuste qui est fondée sur un algorithme de classification floue. Les résultats obtenus mettent en évidence l'intérêt des approches par classification floue pour la résolution des problèmes d'estimation multiple en vision par ordinateur.

Notre méthode permet d'obtenir un recalage suffisamment précis pour faire de la concaténation de reconstructions 3D à partir de points de vue différents. Les ensembles sont recalés de manière rigide, mais une simple extension de notre approche permettra de combiner une transformation rigide avec un changement d'échelle.

Enfin, le résidu obtenu après recalage fin, sur les différents modèles géométriques d'objets qui peuvent se trouver dans la scène, permet d'envisager l'utilisation de notre méthode non seulement pour la reconnaissance 3D d'un modèle dans la reconstruction, mais aussi pour la reconnaissance à partir de bibliothèques d'objets.

4.5 Transformée de Hough généralisée aux déplacements rigides

Nous présentons ici, comme rappel, une autre méthode combinatoire basée sur une transformée de Hough qui permet de retrouver des déplacements initiaux approchés. Cette méthode a été implantée puis abandonnée. En effet, elle présente le défaut propre à ce type d'approche lorsqu'on l'applique à un espace de recherche continu qui consiste en la difficulté à régler le pas de discrétisation de l'espace de recherche pour avoir des résultats stables. Des arguments complémentaires sont mis en avant par Stockman [Sto87].

Comme précédemment, chaque paire de facettes peut être mise en correspondance dans l'espace 3D par un nombre fini de couple (R, T) . L'appariement vote donc, en pondérant par la confiance associée φ , pour au moins 4 déplacements. Le vote est fait dans un accumulateur de Hough de dimension 6 (les 3 angles de la rotation et les 3 composantes de la translation). Habituellement, l'accumulateur de Hough est discrétisé et il est représenté sous forme d'un tableau. Dans le cas de 6 dimensions, ce procédé est coûteux car une discrétisation fine nécessite beaucoup de mémoire. Dans notre cas, il est préférable d'utiliser une représentation de l'accumulateur sous la forme d'un arbre de profondeur 6, où une feuille est un petit volume de l'accumulateur. En effet, le nombre de votants est faible par rapport au nombre de cellules de l'espace de vote; beaucoup sont donc vides. Ainsi, au lieu de les créer toutes a priori en discrétisant cet espace, il faut les construire seulement si elles contiennent au moins un vote [BB82]. Ces cellules sont donc les feuilles d'un arbre dynamiquement créé, où chaque niveau est associé à un paramètre du déplacement rigide (figure 4.15). La structure en arbre permet un accès rapide aux cases et elle est économique en mémoire. La valeur du vote d'un appariement est sa confiance. Elle est ajoutée à la valeur courante de la cellule.

Lorsque toutes les paires ont voté, l'algorithme cherche la ou les cellules qui ont recueilli le plus de votes. Le nombre minimum de couples de faces qui doit être obtenu pour donner un résultat est de 2. Le chemin qui conduit à la meilleur cellule donne une valeur approximative du déplacement entre les facettes appariées.

Si aucune case ne se dégage, la taille des cellules est trop petite par rapport aux erreurs de reconstruction. Deux paramètres permettent alors de régler la discrétisation en espace et en angle. Ils sont des fonctions des erreurs de reconstruction observées, mais leur réglage est beaucoup plus sensible que celui de notre paramètre $p_{pertinent}$.

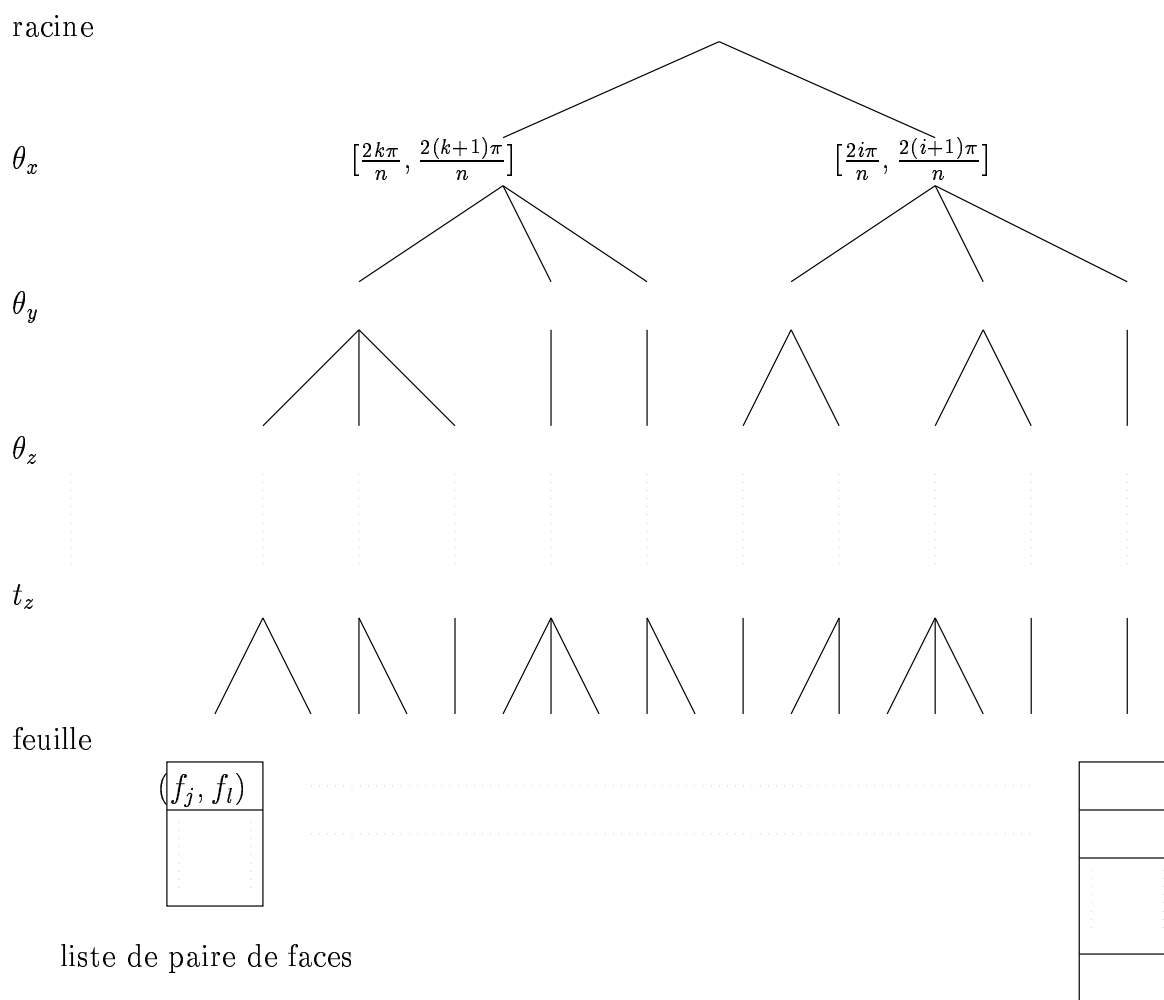


FIG. 4.15 - Exemple d'accumulateur de vote de Hough sur l'espace des déplacements représenté par $(\theta_x, \theta_y, \theta_z, t_x, t_y, t_z)$. Le nombre possible de nœuds de chaque niveau est n . À chaque feuille est associé la valeur du vote et la liste des couples votants.

Chapitre 5

Modélisation par des prototypes recouvrants

5.1 Introduction

Accumuler, par des reconstructions suivant différents points de vue, des informations géométriques sur un même objet n'est pas suffisant pour obtenir un modèle de celui-ci. Les différentes reconstructions recalées entre elles contiennent en effet souvent des parties redondantes qu'il faut fusionner et des parties manquantes qu'il faut interpoler. Poursuivre la modélisation consiste donc à construire une représentation plus abstraite de ces données 3D ou 2D.

Une approche pour traiter ce problème qui combine la mise en correspondance des données et le recalage est de chercher à recouvrir toutes les données par un ensemble d'instance d'un même prototype paramétré. Comme notre algorithme de reconstruction produit des facettes, il est pertinent d'obtenir une modélisation sous la forme d'un ensemble de prototypes qui sont des parallélépipèdes.

Dans le cas 3D, nous avons distingué deux classes d'algorithmes pour résoudre le problème du recalage avec appariement partiel : ceux qui remettent itérativement en cause les appariements et ceux qui procèdent par accumulation. Mais, même si la description des objets peut être géométriquement complexe, un seul modèle ou prototype peut être utilisé à la fois, par ces différents types de méthodes. En particulier, c'est le cas de la méthode que nous avons décrite dans le chapitre précédent.

Contrairement au cas 3D, en 2D, une approche populaire est fondée sur l'utilisation des méthodes de classification. Ainsi, les prototypes utilisés sont très simples : ce sont des lignes ou des ellipses. Mais en revanche, plusieurs instances du prototype sont utilisées en même temps. En particulier, les méthodes de classification floue ont montré leurs capacités à réaliser des classifications géométriques [KF92, DB92, KNF92].

Ces deux démarches, n'ont pas de raison d'être spécifiques au 2D ou au 3D et sont fortement complémentaires. Mais comment partitionner au mieux les données pour que sur chaque partie, le prototype déformable se superpose bien aux données ? Des travaux dans ce sens utilisent comme prototypes les super-quadriques [GB93, FLW89, Pen87], des surfaces algébriques [Tau91] ou plus simplement les quadriques [FS92].

S'il est impossible de séparer partitionnement et estimation dans ce type de problème [BSG90], alors, la classification floue fournit un cadre intéressant pour le poser. De fait, nous avons formalisé en dimension quelconque ce problème, dans une description unificatrice qui intègre non seulement les avantages des méthodes de recalage développées dans le contexte de la vision par ordinateur 3D mais aussi ceux complémentaires des algorithmes de reconnaissance des formes 2D.

L'unification des algorithmes FCM et ICP est très puissante et permet de produire une classe d'algorithmes très générique. En effet, cette approche, que nous présentons dans la section 5.2, peut être utilisée pour générer de nombreux algorithmes propres à un type de modèle dans l'espace de dimension p de son choix. Nous décrivons dans les sections suivantes trois applications de cette approche. Dans la section 5.3, la généralisation de l'algorithme ICP (GICP) à des transformations autres que le déplacement rigide comme les similitudes, les homographies, nous a permis de concevoir un algorithme de recalage à plusieurs prototypes. Dans la section suivante 5.4, nous avons appliqué notre approche à la segmentation d'images de niveaux de gris ou en couleurs, comme aux cartes de profondeur. Enfin, dans la dernière section nous présentons, toujours dans le même esprit, un algorithme pour recalculer finement et de façon coordonnée plusieurs contours actifs.

5.2 L'approche Modélisation Multi-Prototypes (MMP)

Pour décrire plus en détails notre approche, nous avons choisi de décrire, dans la prochaine section, un algorithme de recouvrement par des formes géométriques simples. C'est ce problème initial qui a motivé cette étude. Mais, nous voulons insister sur le fait que cet algorithme n'est que l'application d'un principe d'algorithme plus général. Ce principe consiste à formuler l'appariement, problème de nature combinatoire a priori, comme un problème d'estimation par minimisation sur un support continu, grâce à l'utilisation des degrés d'appartenance introduits dans le cadre des ensembles flous.

On peut résumer le principe de cette méthode par :

- Une étape **d'initialisation** qui doit être astucieuse pour faciliter au plus le travail des deux étapes suivantes. Cette étape est spécifique de l'objectif de chaque problème et du type de données.
- Une étape **d'appariement** où on recherche, pour chaque donnée X_i ($i = 1, \dots, n$), le plus proche voisin X'_{ik} dans le prototype k (avec éventuellement un prototype qui définit une classe de rejet). Ceci permet d'associer à chaque donnée un correspondant potentiel fonction de l'état courant du prototype. Ensuite, la formule suivante permet de déterminer le degré d'appartenance de chaque donnée à chaque prototype en se fondant sur leur proximité :

$$u_{ik} = \frac{1}{\sum_{j=1}^m \left(\frac{d(X_i, X'_{ik})}{d(X_i, X'_{ij})} \right)^{\frac{2}{m-1}}}$$

- Une étape de **recalage** de chaque prototype k sur les données. Chaque donnée i est pondérée par son degré d'appartenance u_{ik} associée à la classe. Ce prototype peut être une forme géométrique simple (section 5.3.4), une fonction sur \mathbb{R}^2 (section 5.4), ou une structure géométrique déformable plus complexe (section 5.5).
- Une étape **terminale**, qui par un test sur la stabilité des caractéristiques des prototypes obtenus, permet de savoir s'il faut itérer ou non les deux étapes précédentes.

5.3 Application au recouvrement par des formes géométriques simples

Dans cette partie, nous présentons une version de l'algorithme MMP, quand on choisit pour prototype une forme géométrique de référence simple qui peut se déformer suivant une transformation géométrique usuelle. La méthode explique un ensemble donné de points comme le recouvrement de c primitives géométriques qui se déforment dans un ensemble de transformations spécifiées. Mais, commençons par résoudre ce problème dans le cas d'une unique primitive.

5.3.1 Généralisation de la méthode ICP (GICP)

Nous avons évoqué dans la section 1.6.1 un algorithme 'Iterative Closest Point' généralisé à d'autres transformations. Une généralisation naturelle de l'algorithme ICP décrit dans la section 1.6.1 est de remplacer le déplacement rigide par un autre type de transformation F , notamment une similitude, une affinité ou une homographie.

Étant donné un ensemble de points, nous supposons que nous connaissons le prototype de cet ensemble et le type de transformation qui peut exister entre la forme modèle et les données. Par exemple, si l'ensemble échantillonne un parallélogramme, on peut choisir comme forme modèle, le carré unité dans son repère propre, et comme ensemble de déformations, les transformations affines, pour modéliser cet ensemble de points. Le but du GICP est d'obtenir la meilleure transformation au sens des moindres carrés qui superpose la forme de référence avec le jeu de données.

Dans cette section, nous allons nous intéresser plus particulièrement à des primitives géométriques simples déformées par 6 types de transformations F :

- translation $F(X) = X + T$,
- déplacement rigide $F(X) = RX + T$,
- similitude $F(X) = sRX + T$: un déplacement rigide plus un facteur d'échelle s ,
- étirement $F(X) = RDX + T$: un déplacement rigide plus des facteurs d'échelle sur chaque axe (D est une matrice diagonale),
- une transformation affine $F(X) = LX + T$,
- et une transformation homographique \mathcal{H} .




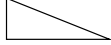
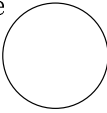


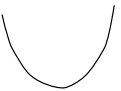

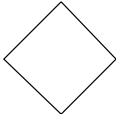

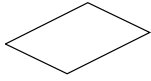
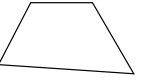
	Translation	Déplacement	Similitude	Affinité	Homographie
triangle unité centré 	triangles unités orientés et isocèles	triangles unités isocèles 	triangles isocèles 	triangles 	triangles
cercle unité centré 	cercles unité	cercles unités	cercles 	ellipses 	coniques 
carré unité centré 	carrés orientés unités	carrés unités 	carrés 	parallélogrammes 	quadrilatères 

FIG. 5.1 - Différentes formes engendrées par 5 types de transformations géométriques appliquées à 3 formes de référence particulières : le cercle, le carré et le triangle.

Pour les primitives particulières suivantes, un carré unité, un cercle unité et un triangle isocèle unité dans leur système de coordonnées propres, il est intéressant de garder à l'esprit la classe des formes engendrées par l'application des transformations citées plus haut, pour entrevoir l'utilisation possible de l'algorithme GICP. Par exemple, avec le carré unité, l'ensemble des carrés, des rectangles, des parallélogrammes, et des quadrilatères peut être engendré en choisissant le type de transformation adéquat. Avec le cercle unité, c'est l'ensemble des cercles, des ellipses, et des coniques qui sont engendrés. Le tableau 5.1 montre en détails toutes les possibilités.

Plus généralement, comme pour l'ICP, le GICP peut travailler avec des modèles définis par des ensembles de points, de lignes, avec des courbes implicites ou paramétriques, et avec des surfaces maillées, implicites ou paramétriques.

L'algorithme GICP

L'algorithme GICP, où l'ensemble des n points $\{X_i\}_{1,\dots,n}$, est connu, est le suivant :

- étape 0** : Fixer la forme du modèle de référence S_{ref} . F est initialisée.
- étape g** : Calculer le nouveau point X'_i appartenant au modèle S_{ref} qui est le plus proche du point $F(X_i)$, pour chaque i .
- étape g'** : Calculer la nouvelle transformation F qui minimise l'erreur aux moindres carrés entre les points X'_i et $F(X_i)$.
- étape g''** : Si l'erreur aux moindres carrés est réduite d'une valeur inférieure à un seuil donné a priori qui spécifie la précision, alors il faut s'arrêter ou sinon retourner à l'**étape g**.

L'erreur aux moindres carrés minimisée par l'algorithme GICP est :

$$e_{GICP}(F, X'_i) = \sum_{i=1}^n d^2(F(X_i), X'_i) \quad \text{avec} \quad X'_i \in S_{ref} \quad (5.1)$$

où $d(X, X')$ est la distance euclidienne entre deux points X et X' .

Pour pouvoir généraliser l'algorithme ICP, il est donc nécessaire d'être capable d'exprimer analytiquement quelle est la transformation optimale qui minimise l'erreur aux moindres carrés entre deux ensembles de n points appariés (**étape g'**). Lorsque $\{X'_i\}_{1,\dots,n}$ est connu, l'expression de la transformation optimale qui minimise $e_{GICP}(F)$ dans l'équation (5.1) pour différents types de transformations est donnée dans la section 1.3.

Convergence

L'algorithme GICP converge toujours de façon monotone vers un minimum qui minimise localement le critère aux moindres carrés (5.1).

La preuve de la convergence de l'algorithme GICP est exactement la même que celle qui permet de démontrer la convergence de l'algorithme ICP, mais elle est appliquée dans un contexte plus général, où la transformation n'est pas seulement une transformation rigide. En effet, la démonstration ne nécessite pas d'avoir une transformation rigide, mais elle réclame que la transformation qui minimise l'équation (5.1) existe pour toutes les itérations de l'**étape g'**. Or nous pouvons assurer que ces conditions sont vérifiées pour les transformations que nous avons considérées (voir section 1.3).

En résumé, les raisons de la convergence de l'algorithme GICP tiennent à ce que l'**étape g** et l'**étape g'** réduisent chacune, à chaque itération, la distance moyenne (5.1) entre le modèle et les données. L'**étape g**, en choisissant le point le plus proche X'_i de $F(X_i)$, réduit individuellement un terme de la somme (5.1), et par là-même la distance moyenne. Quant à l'**étape g'**, elle réduit par définition la distance moyenne, puisqu'elle consiste à trouver la transformation qui minimise cette erreur. Alors, comme e_{GICP} est toujours positive, l'algorithme converge de manière monotone vers un minimum.

La figure 5.2 montre les résultats obtenus après convergence de l'algorithme GICP, sur les mêmes données pour les transformations présentées précédemment. La forme de référence utilisée est le carré unité dans son repère propre.

Points parasites

L'algorithme ICP suppose que les données sont incluses dans la forme modèle. En conséquence, la présence de points parasites dans les données qui ne peuvent pas être recouverts par la forme modèle recalée, biaise les résultats de manière significative. La robustesse à ces points parasites est pourtant souvent utile pour que l'algorithme ICP soit efficace dans la pratique.

L'introduction d'un estimateur robuste, comme le permet aisément les méthodes de classification, permet de résoudre ce problème, mais néanmoins fragilise l'algorithme GICP en le rendant moins stable. En conséquence, il est nécessaire de contraindre la méthode GICP à expliquer toutes les données.

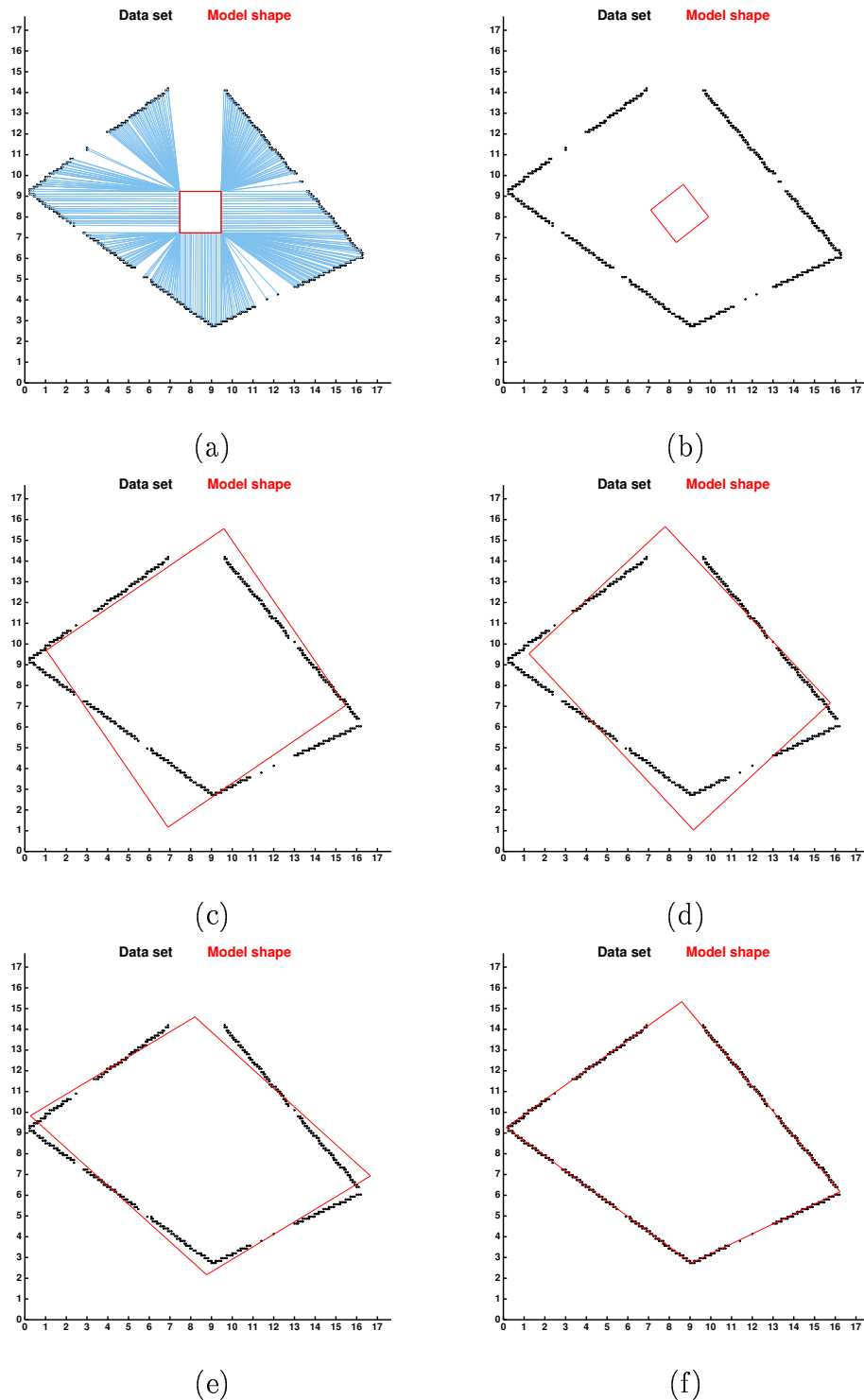


FIG. 5.2 - Résultat produit par l'algorithme GICP où la forme de référence utilisée est le carré unité dans son repère propre et la transformation (a) une translation, (b) un déplacement rigide, (c) une similitude, (d) un étirement, (e) une affinité et (f) une homographie. On remarque que les données peuvent être tronquées de façon importante sans poser de problèmes à l'algorithme. En (a), les appariements entre la forme de référence et les données sont aussi montrés.

5.3.2 Méthodes de classification

L'approche GICP est limitée du fait qu'une seule forme modèle peut être utilisée en même temps. Il s'agit donc d'ajouter le cadre algorithmique de la classification à l'algorithme GICP pour obtenir une méthode de modélisation géométrique multi-modèles. Nous avons décrit dans le chapitre 1, deux algorithmes de classification et leurs avantages respectifs: 'Fuzzy c-Means' (FCM) et son extension en présence de points parasites.

Il faut rappeler que l'algorithme de classification floue utilise une contrainte: la somme des degrés d'appartenance sur les classes est égale à 1. Ainsi, l'algorithme peut traiter les points ambigus, comme le montre la figure 5.3, ce que ne peut pas faire une méthode de classification classique. En particulier, on observe que les centres des classes sont mieux localisés par la méthode floue en présence de points ambigus. Dans ce chapitre, cette propriété se révèle très utile pour réduire le nombre de minima locaux où la solution est réduite à une forme dégénérée.

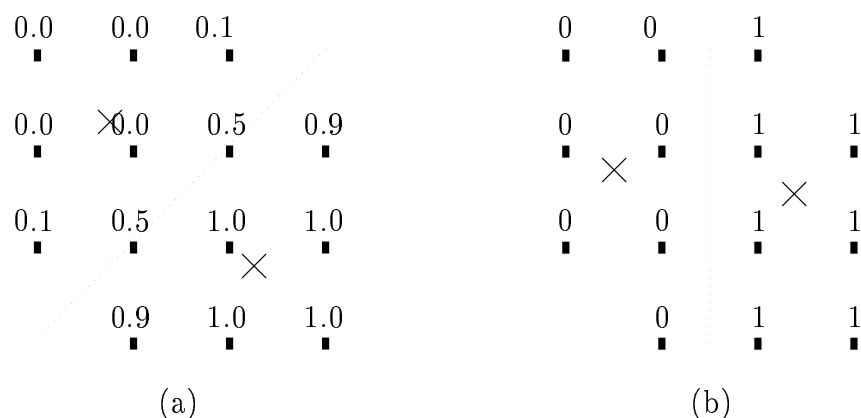


FIG. 5.3 - Classification obtenue sur les mêmes données avec la méthode de classification floue (a) qui traite mieux les points ambigus, et une méthode classique (b). On remarque en particulier que les centres des deux classes, marqués par des croix, sont mieux localisés par la méthode floue. Le chiffre au dessus point indique son degré d'appartenance à la classe du bas.

Il est aussi possible de traiter les points aberrants grâce à la classification floue en introduisant une classe de rejet comme cela à déjà été introduit dans le chapitre précédent. Dans le but d'éviter l'introduction d'une connaissance supplémentaire, Davé a proposé une heuristique pour choisir le paramètre δ dans la version robuste, que nous avons déjà utilisée. Mais, ici, nous n'avons pas fait ce choix, car contrairement au précédent chapitre (section 4.2.2) où les classes sont bien séparées géométriquement, ce critère ne nous a pas donné totale satisfaction sur les tests présentés dans la section 5.3.4 où les classes sont entrelacées.

Dans la méthode FCM et son extension, le prototype d'une classe est le centre de gravité. Il paraît très intéressant en vision par ordinateur comme en reconnaissance des formes, de disposer d'un prototype plus générique ayant une forme géométrique, soit ligne, triangle, rectangle, ou ellipse. Des dérivés de FCM existent où la définition du prototype est modifiée pour

qu'il soit plus riche géométriquement qu'un point. On trouve des méthodes qui utilisent des lignes et des hyper-plans [KF92], des hyper-sphères [KNF92], et des hyper-ellipsoïdes [DB92]. Néanmoins, ces méthodes sont spécifiques à une forme donnée. En utilisant, l'approche ICP, il est possible de dépasser cette limitation et nous proposons dans la section suivante un algorithme générique pour classer des formes géométriques d'un même type, qui peut être utilisé avec un grand nombre de formes, une fois conçue la fonction de plus proche voisin particulière adaptée à chaque forme de prototype.

5.3.3 Unification des algorithmes GICP et FCM

Il est possible de faire plusieurs rapprochements entre les algorithmes GICP et FCM. D'abord, les deux méthodes minimisent itérativement en deux étapes une fonction de coût aux moindres carrés (5.1) ou (1.28) entre un prototype et des données. Par exemple, on voit que l'**étape f'** et l'**étape g'** consistent à faire le même calcul : calculer la transformation géométrique qui minimise l'écart entre la forme prototype ou modèle et les données. Plus particulièrement, le calcul est exactement identique dans ces deux étapes quand le GICP est utilisé avec une translation et une forme modèle qui est réduite à un point, et quand le FCM a une unique classe. Il semble donc naturel d'unifier ces deux approches pour obtenir un algorithme puissant de Modélisation Multi-Prototypes (MMP) dont l'algorithme GICP et FCM ne sont que des cas particuliers. Ceci est réalisable très simplement en ajoutant au GICP le contexte des ensembles flous du FCM.

Description de l'algorithme MMP

L'algorithme de Modélisation Multi-Prototypes sans point parasite (*respectivement avec une classe de rejet F_c*) consiste en :

étape 0 : Fixer le nombre de classes c et la forme du prototype S_{ref} . Initialiser les transformations F_k (voir la section 5.3.3). (*Donner la distance δ*).

étape m : Calculer le plus proche voisin X'_{ik} appartenant au prototype S_{ref} de chaque point $F_k(X_i)$. (*seulement pour k entre 1 et $c - 1$*).

étape m' : Générer une nouvelle partition en utilisant l'équation du degré d'appartenance flou suivante :

$$u_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d(F_k(X_i), X'_{ik})}{d(F_j(X_i), X'_{ij})} \right)^{\frac{2}{m-1}}} \quad (5.2)$$

(où $d(F_c(X_i), X'_{ic}) = \delta$).

étape m'' : Calculer les nouvelles transformations F_k qui minimisent l'erreur pondérée aux moindres carrés entre les points $F_k(X_i)$ et X'_{ik} :

$$\sum_{i=1}^n (u_{ik})^m d^2(F_k(X_i), X'_{ik})$$

(*seulement pour k entre 1 et $c - 1$*).

étape m'' : Si la partition est stable, il faut s'arrêter, sinon retourner à l'étape m .

L'algorithme MMP minimise la fonction de coût suivante :

$$e_{MMP}(F_k, X'_{ik}, u_{ik}) = \sum_{k=1}^c \sum_{i=1}^n (u_{ik})^m d^2(F_k(X_i), X'_{ik}) \quad \text{avec} \quad \sum_{k=1}^c u_{ik} = 1 \quad \text{et} \quad X'_{ik} \in S_{ref} \quad (5.3)$$

L'algorithme GICP n'est qu'un cas particulier de l'algorithme MMP, où le nombre de classe c est égal à 1 (on a donc $u_{i1} = 1$). On voit aussi que l'algorithme FCM est un autre cas particulier, où le prototype S_{ref} est réduit à un point et où la transformation F_k est une translation.

Convergence

L'algorithme de modélisation Multi-Prototypes converge toujours de façon monotone vers un minimum local de l'erreur aux moindres carrés (5.3).

Les grandes lignes de la démonstration sont analogues à celles de l'algorithme GICP et FCM (pour la démonstration de la convergence de FCM, on peut se reporter à la section 4.2.2, cas où l'algorithme FCM est étendu). Le principe est que chaque étape réduit le critère de l'écart aux moindres carrés (5.3). Certes, cette convergence est uniquement vers un minimum local. Alors, comment initialiser les prototypes pour obtenir une solution intéressante?

Initialisation

Avec l'algorithme itératif ICP, la position de départ est souvent supposée connue. En effet, la solution que l'on obtient dépend de manière critique de la position initiale. L'algorithme FCM, ne prenant pas en compte les points parasites, est plus robuste à l'initialisation qui peut souvent alors être aléatoire.

L'algorithme MMP est, lui aussi, assez sensible à l'initialisation. Nous proposons donc le schéma suivant pour initialiser des transformations F_k :

- Trouver la partie de F_k qui est une translation T_k par identification de la translation avec le centre des classes qui sont solutions de l'algorithme FCM.
- Utiliser les matrices d'inertie des sous-ensembles flous obtenus précédemment pour calculer, par diagonalisation, la rotation initiale, ou l'étirement initial de chaque prototype, suivant le besoin.

Naturellement, ce schéma est aussi étendu pour former un schéma d'initialisation du plus approximatif au plus fin : on trouve la meilleure translation, le meilleur déplacement, le meilleur étirement, la meilleure affinité, et enfin la meilleure homographie (voir la figure 5.2). Ces étapes peuvent mélanger, suivant le problème, la version avec ou sans la classe de rejet.

Ces schémas nous ont donné satisfaction dans certains de nos tests (voir la figure 5.5(c) où les translations apparaissent comme des croix). Mais, comme cela est indiqué dans la section suivante, certains cas particuliers, où les formes sont très fortement entrelacées, mettent en échec ce schéma d'initialisation. Il faut alors, pour chaque type de configurations à problème, définir une façon adaptée d'initialiser l'algorithme MMP. Trouver une initialisation astucieuse, mais spécifique à un type de données, est aussi utile pour accélérer l'algorithme.

5.3.4 Résultats

Convergence rapide dans les premières étapes

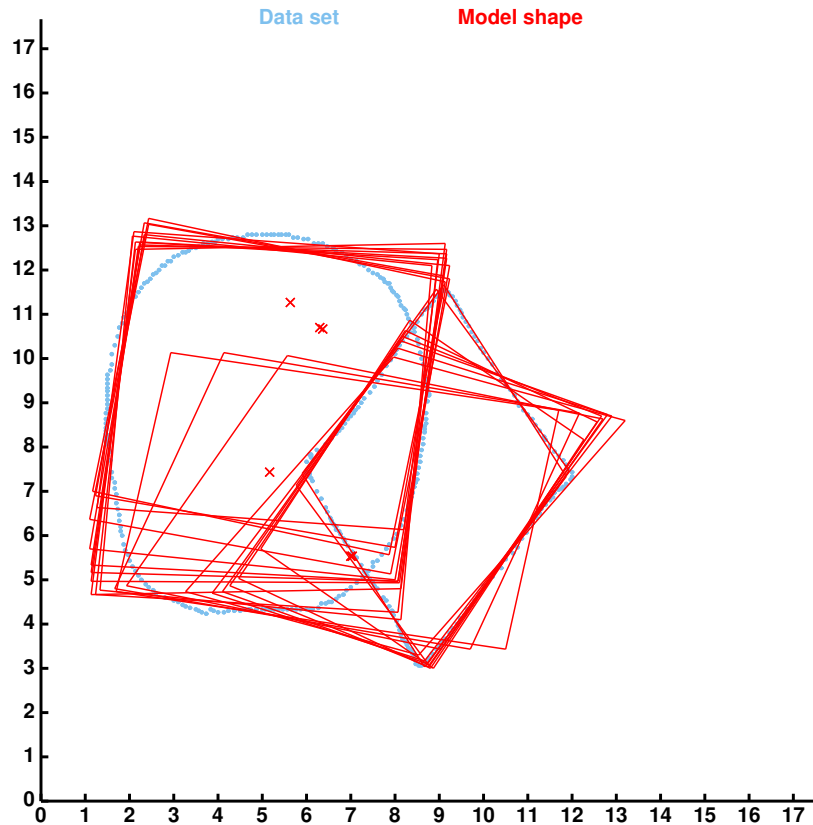


FIG. 5.4 - Quelques itérations de notre algorithme. Le résultat final est visible sur la figure 5.14(a).

Dans la figure 5.4, les points échantillonnent approximativement et partiellement un cercle et un parallélogramme. Ces points sont modélisés comme 2 parallélogrammes (figure 5.14(a)).

La figure 5.4, où est visible une itération toutes les 10 itérations, donne une idée de la recherche effectuée par l'algorithme. Le nombre d'itérations nécessaires pour arriver au résultat est de 156 (le résidu e_{MMP} avait alors une variation inférieure 0.001 à chaque étape).

Quant aux prototypes initiaux, ils sont obtenus par le schéma d'initialisation décrit dans la section 5.3.3. Cette figure présente un cas relativement difficile pour l'algorithme car l'initialisation est assez éloignée de la solution à trouver, puisque les deux formes sont entrelacées.

Mais, nous avons observé, en général, une convergence très rapide dans les premières étapes lorsque l'initialisation est proche de la solution recherchée. Les étapes supplémentaires de stabilisation peuvent en revanche être assez nombreuses (de l'ordre d'une centaine). Néanmoins, le nombre d'étapes ne peut être tronqué trop brutalement si on désire conserver

la robustesse de l'algorithme quand la solution initiale est assez éloignée de la transformation optimale.

Données entrelacées

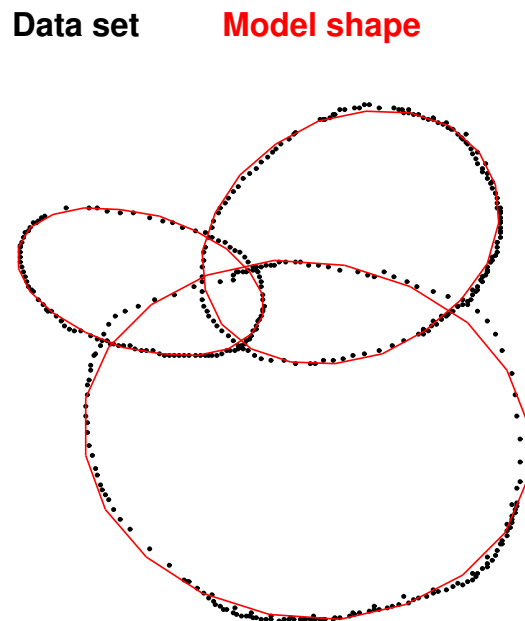


FIG. 5.5 - *L'algorithme MMP interprète bien l'ensemble comme trois ellipses qui s'intersectent.*

Dans la figure 5.5, l'ensemble des points est synthétique, bruité et incomplet. Il est modélisé avec justesse par l'algorithme MMP comme trois ellipses qui s'intersectent. Les prototypes initiaux sont obtenus par le schéma d'initialisation décrit dans la section 5.3.3.

Néanmoins, dans des configurations de ce type, où les objets à modéliser sont fortement entrelacés, le schéma d'initialisation de l'algorithme MMP ne donne généralement pas de bonnes formes de départ. En effet, notre schéma d'initialisation suppose que les parties à recaler par chaque prototype géométrique sont bien séparées. De façon analogue, lorsque les ellipses sont concentriques, le schéma d'initialisation proposé est mis en échec car la première étape de l'initialisation ne permet pas d'obtenir une classification pertinente de l'ensemble. Mais, dans certains cas particuliers, un schéma de l'algorithme MMP adapté peut être proposé. Par exemple, pour les ellipses concentriques, on peut supposer les centres

tous identiques et par une classification unidimensionnelle de l'histogramme des distances au centre, trouver les petits et grands axes initiaux.

En présence de points parasites

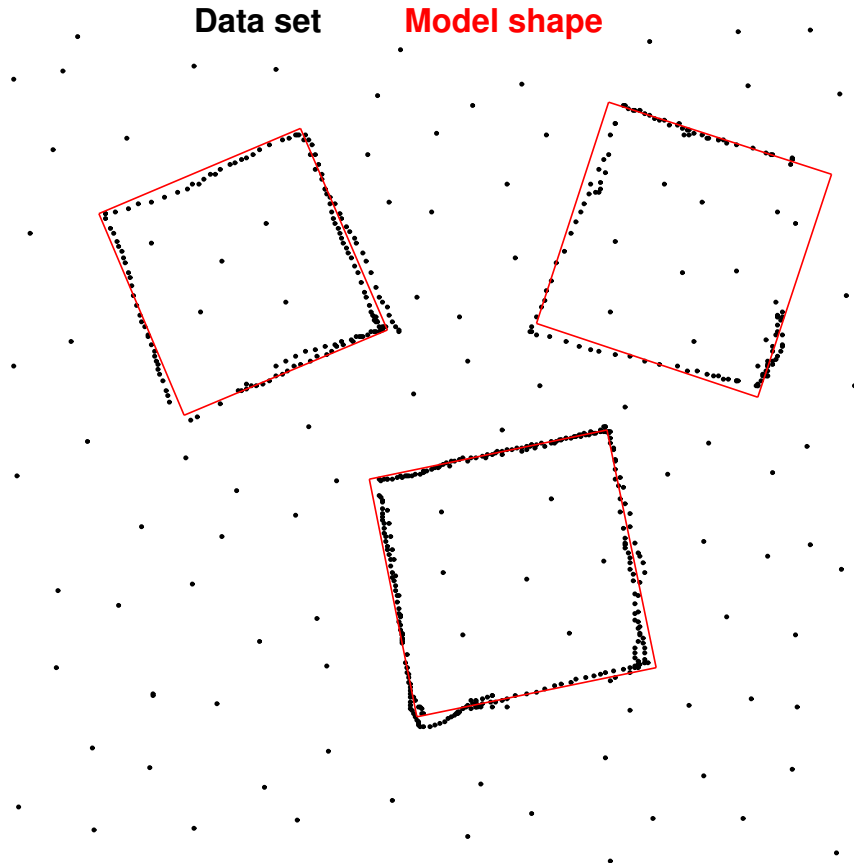


FIG. 5.6 - L'algorithme MMP interprète bien l'ensemble comme trois cubes malgré le bruit de fond représentant environ 20% des données.

La figure 5.6 montre un exemple 2D en présence de 20% de points aberrants. Dans ce cas précis, trois carrés sont présents, et sont retrouvés par notre algorithme en procédant en deux étapes :

- un recalage rigide avec 3 carrés unités sans introduire de classe bruit,
- un recalage par similitude des 3 carrés obtenus par l'étape précédente avec en plus une classe pour récolter les points parasites (δ distance des points parasites a été choisie égale à 20% du côté des cubes).

Lorsque les points parasites sont uniformément répartis, comme dans la figure 5.6, initialiser l'algorithme MMP par une recherche avec un déplacement rigide ou une similitude,

sans classe parasite est souvent suffisant. En effet, les points aberrants n'induisent alors pas de perturbation dans une direction privilégiée. Par contre, en présence d'un point d'accumulation de points parasites, la procédure précédente échoue. Il faut alors mieux considérer le point d'accumulation comme une nouvelle structure dégénérée, et faire une recherche avec une classe supplémentaire comme cela a été fait dans la figure 5.7.

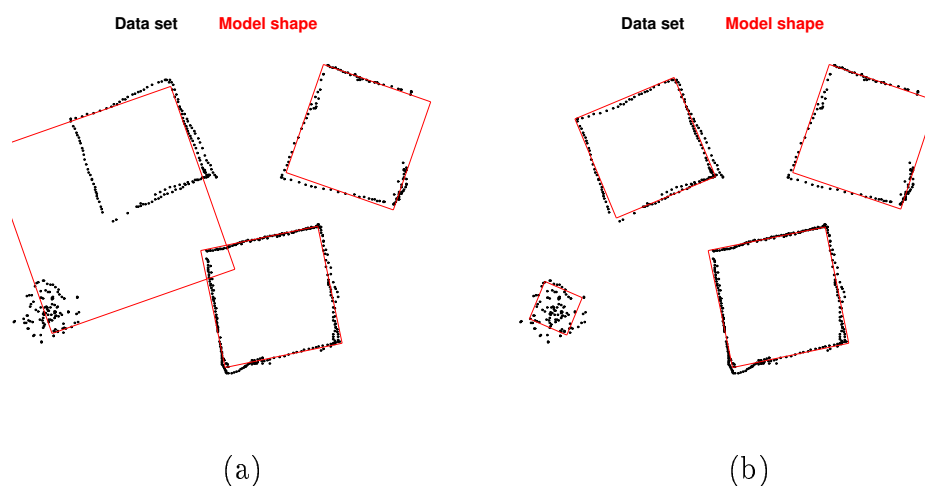


FIG. 5.7 - En (b), l'algorithme MMP interprète bien les trois cubes malgré car le point d'accumulation de points parasites est considéré comme une primitive dégénérée. Alors que si l'on utilise la version robuste, l'algorithme n'est pas capable de séparer les points parasites qui représentent 20% des données.

Sur des reconstructions 3D

Suivant le choix du type prototype, il est intéressant de modéliser des objets de forme polyédrique ou gauche.

Avec le système de modélisation décrit dans ce document, la modélisation par plusieurs parallélépipèdes est plus particulièrement intéressante. En effet, beaucoup d'objets de notre environnement sont composés d'un assemblage de ces formes, les bâtiments en particuliers.

Pour illustrer notre procédure de modélisation 3D à partir de paires stéréoscopiques, nous avons, à titre d'exemple, réalisé la modélisation complète d'un objet très simple constitué de pièces d'un jeu de Lego. Cet objet, visible sur la figure 5.8, est constitué de deux blocs principaux interconnectés. Nous avons pris quatre paires stéréoscopiques de cet objet selon les points de vue, Sud, Est, Nord et Ouest.

Une fois la calibration réalisée par la méthode décrite dans le deuxième chapitre, les images sont rectifiées. Puis les images rectifiées ont été segmentées par une méthode par 'split and merge' [Ran92] en utilisant au mieux l'information de couleur de chaque région. Les régions obtenues dans chaque image sont appariées selon des critères de compatibilité épipolaire et de couleurs et taille proches [Vin91].

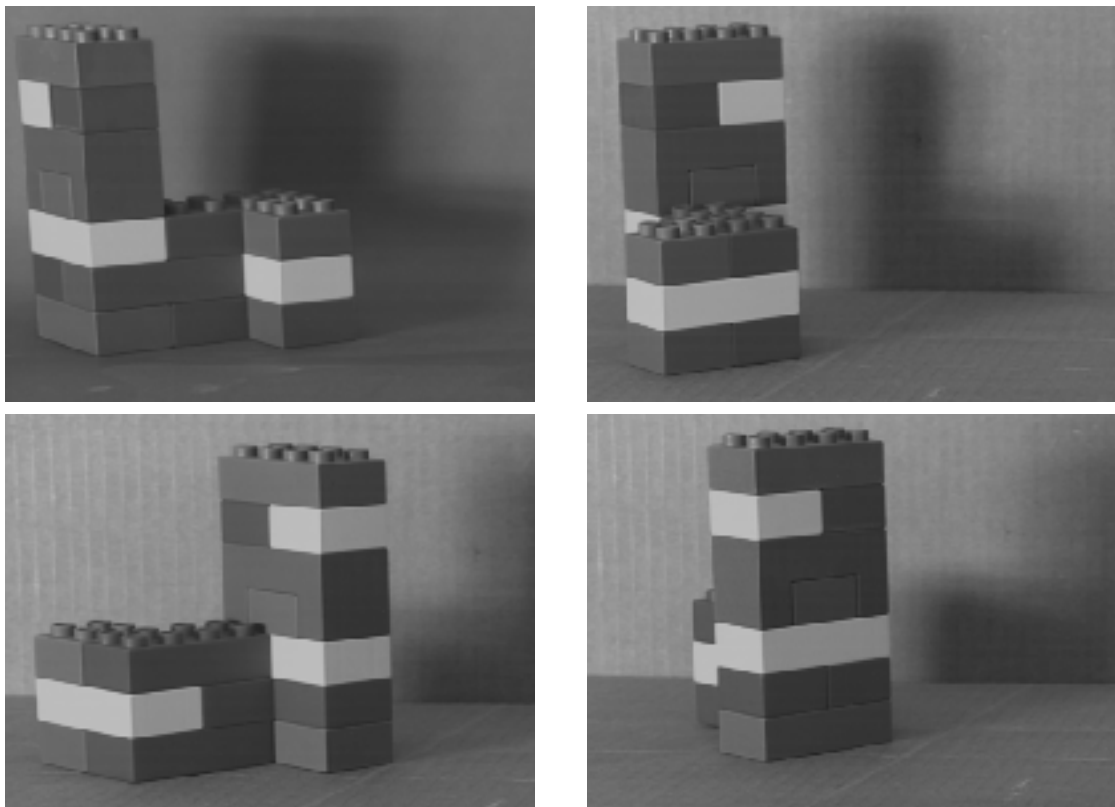


FIG. 5.8 - *Images gauches des quatre paires stéréoscopiques originales d'un jeu de Lego.*

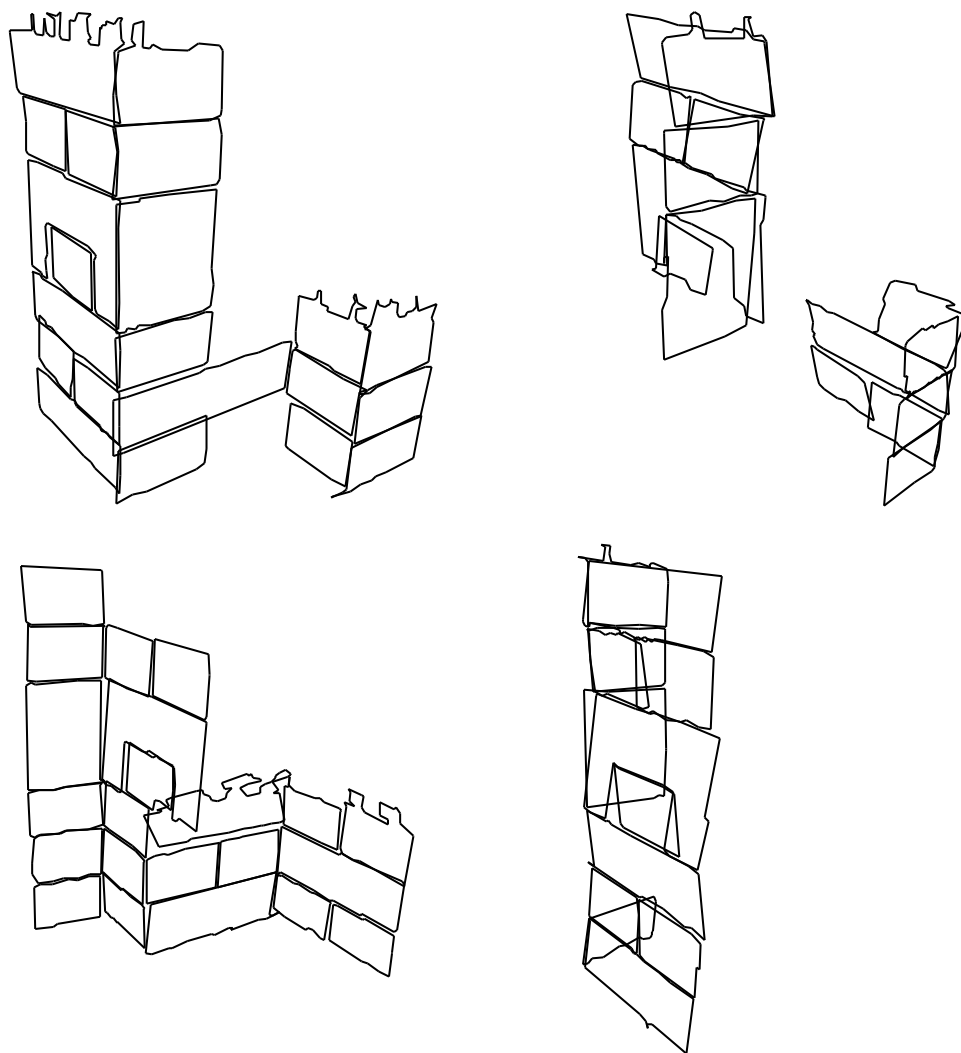


FIG. 5.9 - Les reconstructions du jeu de Lego obtenues à partir des points de vue de la figure 5.8. Les vues ont été choisies avec des orientations telles que les reconstructions puissent s'assembler pour constituer visuellement la reconstruction complète.

Alors, avec la méthode géométrique du chapitre 3, nous avons obtenu les reconstructions selon les quatre points de vue. Elles sont visibles dans la figure 5.9. Puisque les paires de sont pas calibrées par rapport à un même repère absolu, il est nécessaire de recalcr les reconstructions entre elles. Dans ce but, nous avons utilisé la méthode du chapitre 4 pour recalcr et ainsi fusionner la reconstruction ‘Sud’ avec celle ‘Est’ et la reconstruction ‘Nord’ avec celle ‘Ouest’. Par contre, pour recalcr les deux reconstructions ainsi produites, il a été nécessaire d’utiliser l’algorithme de recalcr fondé sur l’ICP avec une recherche exhaustive sur un échantillonnage des positions initiales possibles [Véz95]. En effet, du fait des symétries de l’objet, l’algorithme produit en priorité un recalcr où l’objet est sans épaisseur en plaquant les deux reconstructions l’une contre l’autre. La reconstruction complète est visible dans la figure 5.10.

On remarque combien cette reconstruction présente des défauts et des facettes redondantes. Il est donc nécessaire pour obtenir un modèle plus synthétique de l’ensemble de pièces de Lego, de chercher quelles sont les caractéristiques des primitives qui recouvrent cette reconstruction. Pour réaliser cela, nous avons interpolé les faces pour qu’elles présentent des points uniformément répartis en leur intérieur, puis nous avons utilisé l’algorithme MMP en procédant de manière à retrouver tout d’abord les deux blocs principaux. L’algorithme avec 2 parallélépipèdes a donc été directement appliqué, puis une fois la convergence atteinte, nous avons ajouté une classe de bruit pour affiner le recalcr sur les deux blocs principaux. Cette dernière étape permet aussi de récolter les points qui relient les deux grands blocs car ils se retrouvent dans la classe bruit. Alors, à partir de cette segmentation de la reconstruction en trois classes, nous avons pu initialiser l’algorithme avec trois parallélépipèdes, sans classe de rejet.

Cette façon de procéder, si elle semble un peu compliquée est générique, car elle permet de séparer récursivement les formes principales bien recouvertes par les primitives, des détails qui vont être recouverts à leur tour précisément en introduisant une nouvelle classe. Cette méthode trouve dans un même temps le recalcr à des échelles différentes et détermine automatiquement le nombre de classes utiles. Le résultat, superposé aux contours de la reconstruction est visible dans la figure 5.11.

Enfin, nous présentons dans la figure 5.12, le modèle résultat obtenu des pièces de Lego, sur lequel est partiellement retroprojetée la première image originale afin de faciliter la compréhension des vues.

Sur des reconstructions 3D avec normales

Notre algorithme peut aussi s’appliquer à des objets non polyédriques, mais courbes. La figure 5.13 montre le résultat de l’algorithme MMP quand le prototype est la sphère unité centrée. Les transformations F_k sont affines, et les données, gracieusement fournies par Pascal Fua, sont un modèle 3D de la surface d’une tête.

Pour ces données, comme on disposait en plus des normales à la surface $\{N_i\}_{1,\dots,n}$ en chaque point de l’ensemble $\{X_i\}_{1,\dots,n}$. Nous avons intégré cette information dans l’algorithme pour obtenir une méthode plus robuste. La normale sert uniquement à améliorer la mise en correspondance par le biais du calcul du degré d’appartenance u_{ik} . En effet, dans l’étape m' de l’algorithme MMP qui calcule le degré d’appartenance, nous ajoutons, à la distance entre

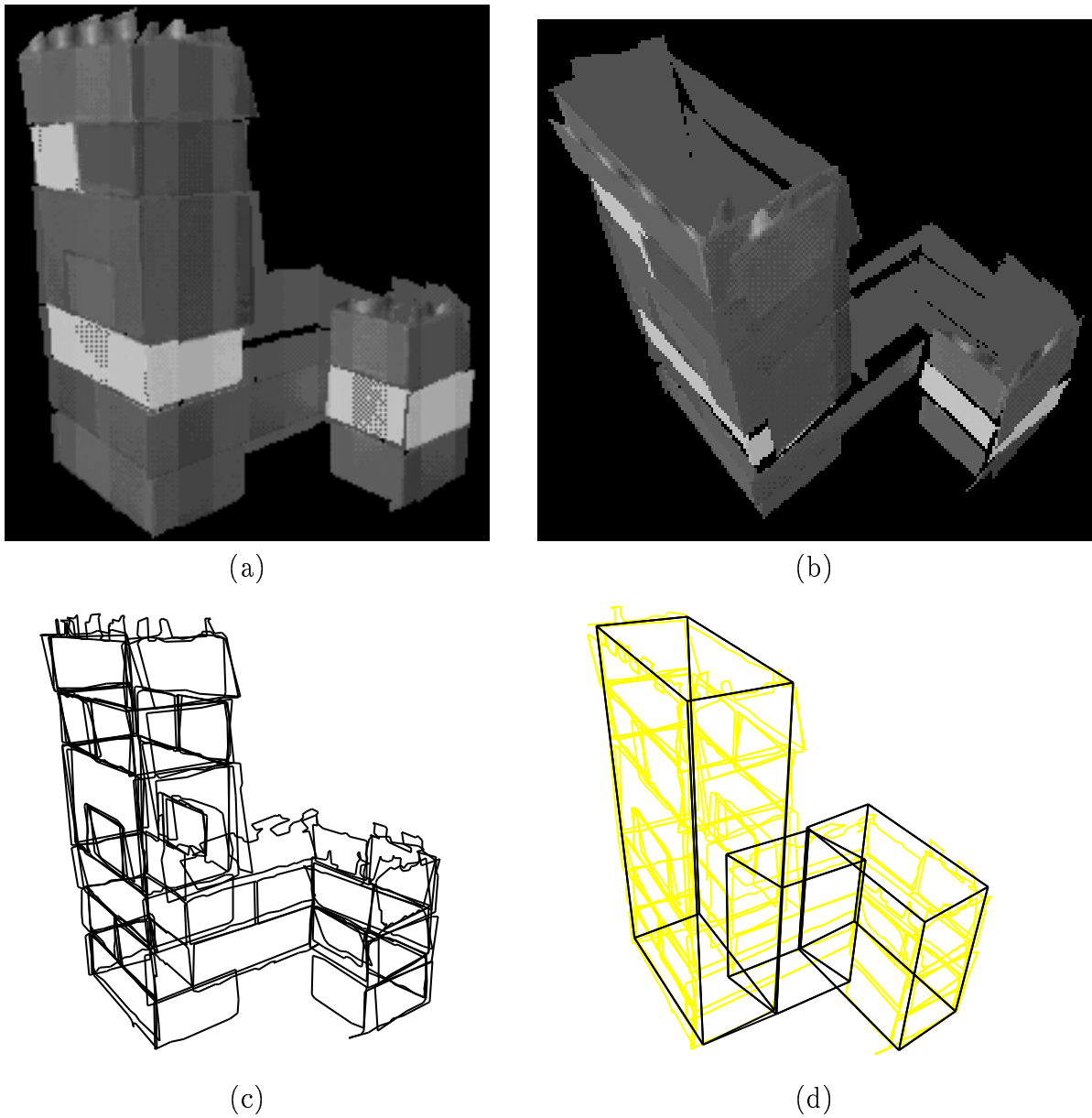


FIG. 5.10 - (a) et (b) sont deux vues de la reconstruction complète obtenue en recalant les reconstructions de la figure 5.9. La texture de la première image originale est retroprojetée sur cette reconstruction. (c) est une vue en fil de fer de cette même reconstruction. (d) montre les 3 parallélépipèdes qui recouvrent la reconstruction.

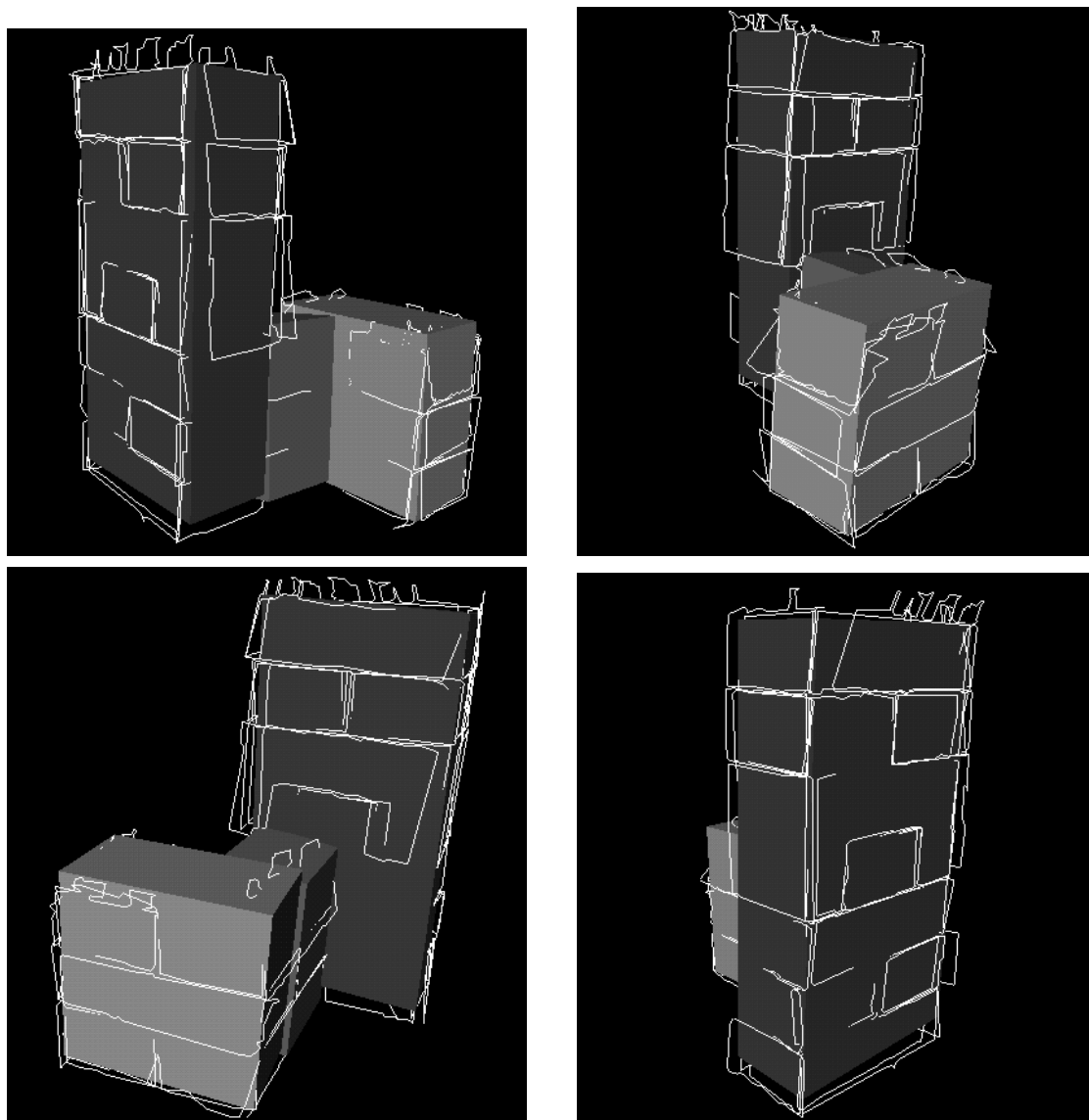


FIG. 5.11 - Les trois parallélépipèdes sont recalés avec la reconstruction complète du jeu de Lego, et sont vus approximativement selon les quatre lieux d'acquisition des images originales. La taille des primitives est légèrement réduite pour permettre une meilleure visualisation des contours reconstruits.

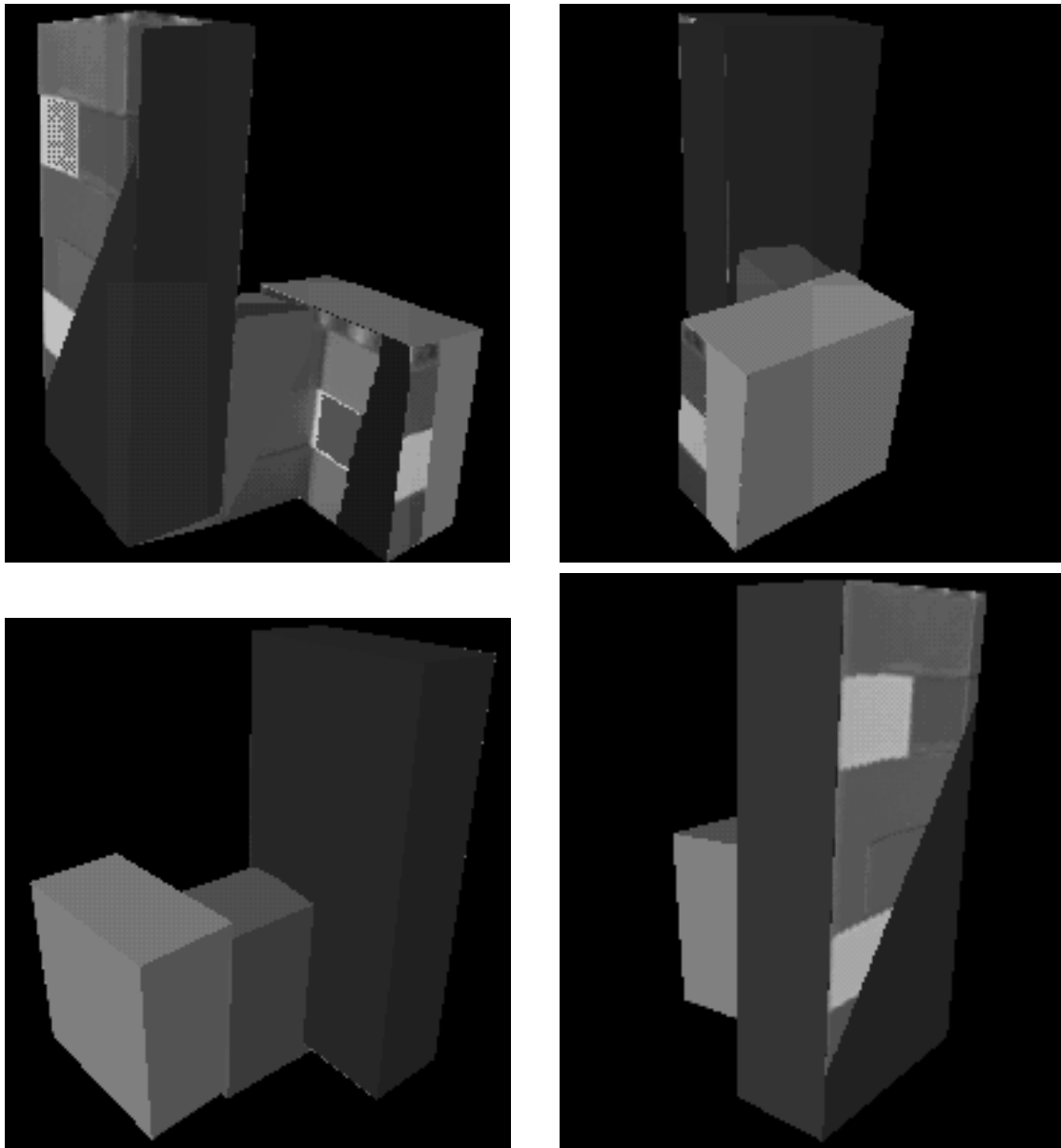


FIG. 5.12 - Les trois primitives de la figure 5.11 recalées sur la reconstruction sont vues selon les mêmes positions de prise de vue que celles de la figure 5.8. De plus, la texture de la première image originale est partiellement retroprojetée sur les primitives afin d'observer la qualité du recalage.

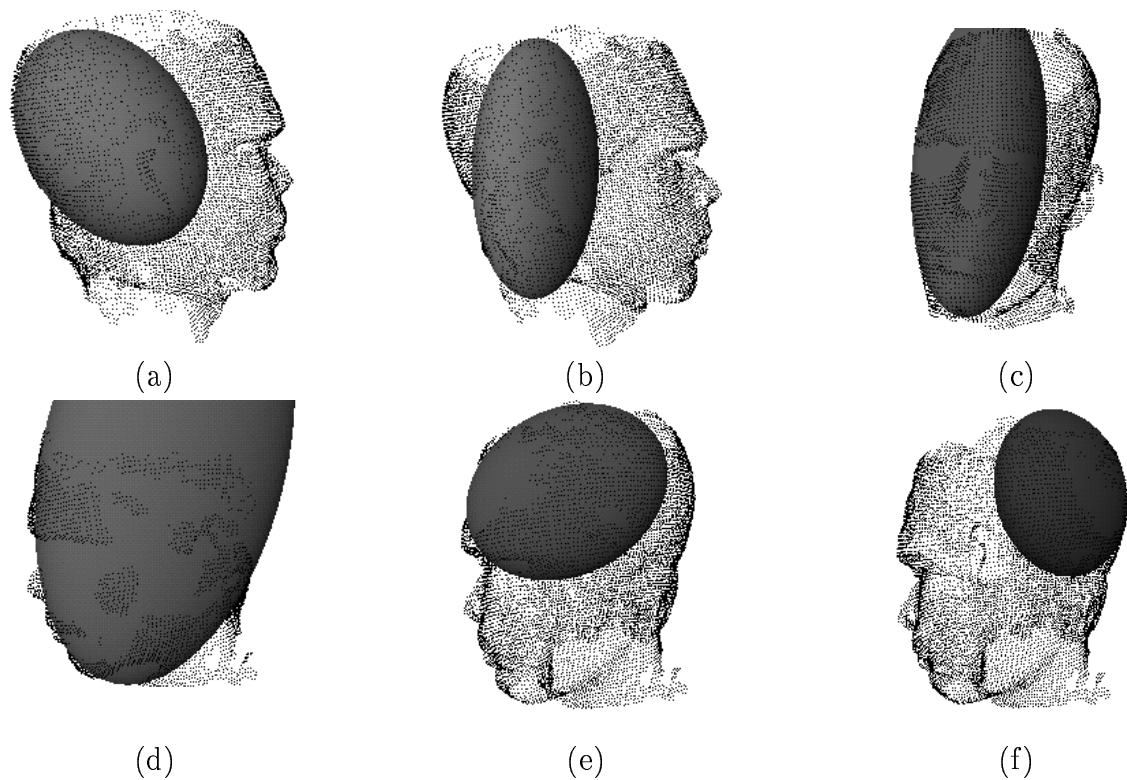


FIG. 5.13 - Les points et les normales du modèle 3D d'une tête, sont modélisés comme la superposition de 15 ellipsoïdes par l'algorithme MMP (les ellipsoïdes ne sont pas découpées). Ces ellipsoïdes recouvrent de façon satisfaisante le haut (a)(f) et le bas (b) de l'arrière du crâne, le front (d), et les cotés (e)(c) de la tête.

les points X'_{ik} et $F_k(X_i) = L_k X_i + T_k$, la distance entre les deux vecteurs normaux associés. Cette dernière distance doit être pondérée par un facteur μ , à régler, pour tenir compte de la différence de nature entre un point et un vecteur normé. Plus ce facteur est élevé, plus on tient compte des normales.

Calculer cette distance nécessite de connaître les deux vecteurs normaux. Premièrement, la normale N_i est transformée en $L_k^{-t}(N_i)$ par une l'affinité. Cette propriété provient du fait que la normale est définie comme orthogonale au plan tangent qui lui se transforme selon F_k . Deuxièmement, il faut calculer la normale à la surface du prototype au point le plus proche X'_{ik} . Comme le prototype est la sphère centrée de rayon unité, cette normale est identique au vecteur X'_{ik} .

Le nouvelle distance est donc :

$$d_{ik} = d(F_k(X_i), X'_{ik}) + \mu d(L_k^{-t}(N_i), X'_{ik})$$

La figure 5.13 montre que notre algorithme permet d'obtenir un modèle du crâne de la tête avec des éléments en forme d'ellipsoïdes. Par contre, la densité des points sur la face et une initialisation brutale n'ont pas permis d'obtenir sa modélisation.

Des prototypes hétérogènes

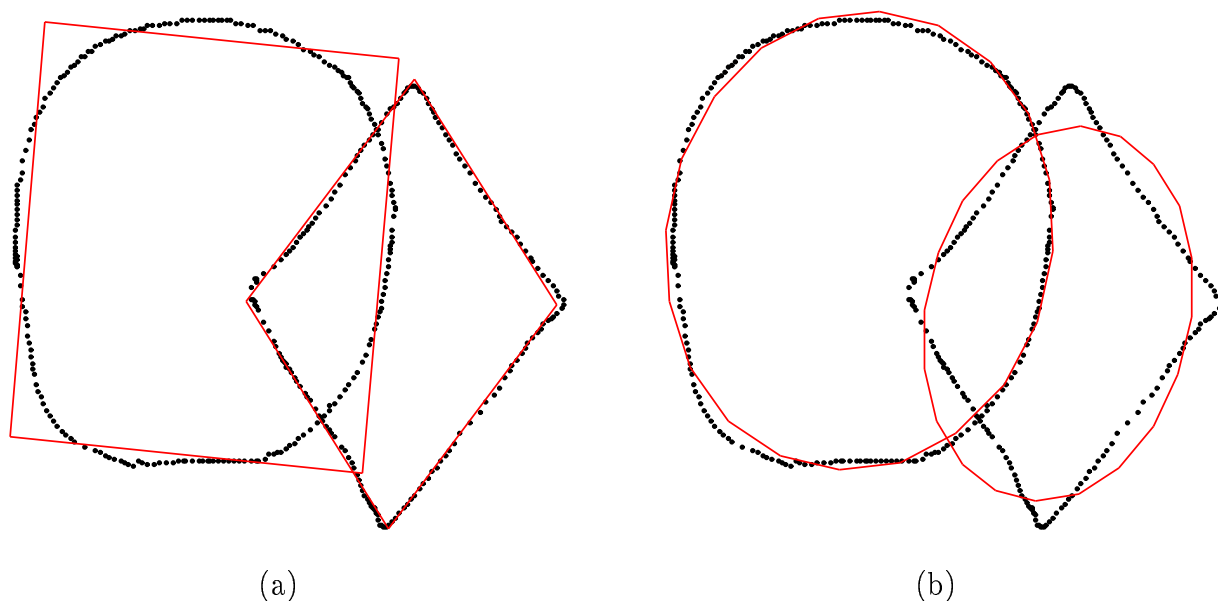


FIG. 5.14 - L'ensemble des points est synthétique, mais il est bruité et incomplet. Les points sont sur un cercle et sur un parallélogramme échantillonnés. Cet ensemble est modélisé par l'algorithme MMP comme 2 parallélogrammes en (a), et comme 2 ellipses en (b).

La figure 5.14 montre que l'algorithme MMP peut produire des solutions intéressantes même si le prototype ne correspond pas bien aux données. L'algorithme est assez robuste. Cette propriété permet d'envisager l'utilisation de cette méthode pour segmenter les données

avec des prototypes hétérogènes comme un parallélogramme et une ellipse, en choisissant la primitive qui minimise au mieux l'erreur moyenne entre les données et le prototype.

Sur l'exemple de la figure 5.14, le résidu entre l'ovale à gauche est de 0.05 mm avec le prototype carré, alors qu'il n'est que de 0.03 mm avec le prototype cercle. Inversement, le résidu du parallélogramme est de 0.08 mm avec le prototype elliptique, alors qu'il n'est que de 0.03 mm avec le prototype parallélogramme. La discrimination semble donc possible sur cet exemple, bien que les données soient assez déformées et perturbées. Cette technique demande à être testée plus profondément.

5.3.5 Discussion

Nous avons présenté un algorithme de Modélisation Multi-Prototypes qui permet de recouvrir un ensemble de points par plusieurs instances d'une même primitive paramétrée par une transformation géométrique. Cet algorithme permet de faire de la modélisation par un ensemble de rectangles, de triangles, d'ellipses, d'ellipsoïdes, de parallélépipèdes, ou de toutes autres primitives simples, dès que l'on peut écrire la fonction qui à tout point donne le plus proche voisin sur la primitive.

Une limitation importante de la classification floue, telle que nous l'avons introduite est de donner le nombre c de classes. Plusieurs travaux [KF92, XB91, ZM90] ont proposé des méthodes pour dépasser cette limitation qu'il serait intéressant d'adapter au cas de notre algorithme MMP.

5.4 Application à la segmentation des images

Il existe de nombreuses méthodes de segmentation des images [HS85], par classification de l'histogramme des niveaux de gris ou en couleurs [EYR80], par accroissement de régions [GM86, Vin91], par 'split and merge' [Ran92]. Beaucoup de ces méthodes ont trouvé un cadre unificateur par une formalisation variationnelle unifiée [KLM94, DF95].

Dans la pratique, ne pas avoir des régions qui sont constantes, mais des régions dont l'intensité varie linéairement est utile [Lec89, AMG93]. En effet, les gradients dus à l'éclairage peuvent souvent être approximés par des plans (section 2.3.2). En suivant le même raisonnement, on voit que la segmentation par des régions où l'intensité varie quadratiquement est aussi intéressante comme le montre son utilisation sur les images de profondeur [FHP83], et sur les images d'intensité [BC84].

Notre algorithme de modélisation MMP, dans sa version sans point aberrant, est aussi une méthode de segmentation de l'ensemble des données, où les points de chaque classe sont contraints à appartenir à un prototype. Notre approche nous a donc permis de développer une méthode de segmentation floue qui étend la classification sur l'histogramme [Bou93] pour permettre la segmentation avec des régions où l'intensité varie linéairement ou quadratiquement.

De plus, grâce à la classification robuste introduite dans le chapitre 1, nous pouvons faire de la segmentation robuste qui détermine automatiquement les pixels aberrants comme dans [JMB91].

5.4.1 Algorithme MMP adapté à la segmentation d'images

Nous avons développé une version de l'algorithme MMP spécialement adaptée à la segmentation des images. Une image de profondeur peut être vue comme un nuage de points 3D, par analogie, une image d'intensité aussi. Mais, le quadrillage régulier des images simplifie beaucoup le problème [MDC92], par rapport aux données purement 3D utilisées dans la section précédente. En effet, les données 2D 1/2 permettent de se ramener à une classification 2D et à des estimations géométriques 1D. Alors, chaque prototype est réduit à une fonction de deux variables sur l'image.

Nous avons testé deux versions de l'algorithme, l'une avec des fonctions linéaires, l'autre avec des fonctions quadratiques. Mais, l'ajout d'une simple procédure permet d'avoir un algorithme où les variations seront polynômiales. Il suffit de concevoir la procédure d'estimation adaptée, pour étendre l'algorithme au type désiré de variation de la fonction. Un prototype F_k est constitué par la liste des coefficients des fonctions d'intensité.

Voici la description de l'algorithme de segmentation d'une image $I(u, v)$:

étape 0 : Fixer le nombre de classes c et le type des fonctions $F_k(u, v)$ sur l'image (par exemple des polynômes). Initialiser ces fonctions $F_k(u, v)$ par une fonction constante avec une valeur aléatoire.

étape s : Générer une nouvelle partition de l'image en utilisant l'équation suivante du degré d'appartenance flou :

$$u_k(u, v) = \frac{1}{\sum_{j=1}^c \left(\frac{d(F_k(u, v), I(u, v))}{d(F_j(u, v), I(u, v))} \right)^{\frac{2}{m-1}}}$$

étape s' : Calculer les nouvelles fonctions sur l'image $F_k(u, v)$ qui minimisent l'erreur pondérée aux moindres carrés entre les points $(x, y, F_k(u, v))$ et $(x, y, I(u, v))$:

$$\sum_{(u,v) \in image} (u_k(u, v))^m d^2(F_k(u, v), I(u, v))$$

étape s'' : Si les fonctions sont stables, il faut s'arrêter, sinon retourner à l'étape s .

Lorsque l'algorithme a convergé, on dispose alors de c fonctions qui donnent la variation de l'intensité des images pour chaque classe. Par contre, la classification obtenue est floue, c'est-à-dire que l'on n'a pas une image de labels, mais c images de degrés d'appartenance $u_k(u, v)$. Pour néanmoins calculer l'image de label, on décide d'attribuer à chaque pixel la classe qui a le degré d'appartenance maximum.

5.4.2 Segmentation des cartes de profondeur

Dans la figure 5.15, le nombre de classes est volontairement petit, car sinon il est difficile de voir les différences entre l'image originale et l'image reconstituée à partir de la segmentation (un tel exemple est visible sur la figure 5.16). Cette image d'approximation est calculée en

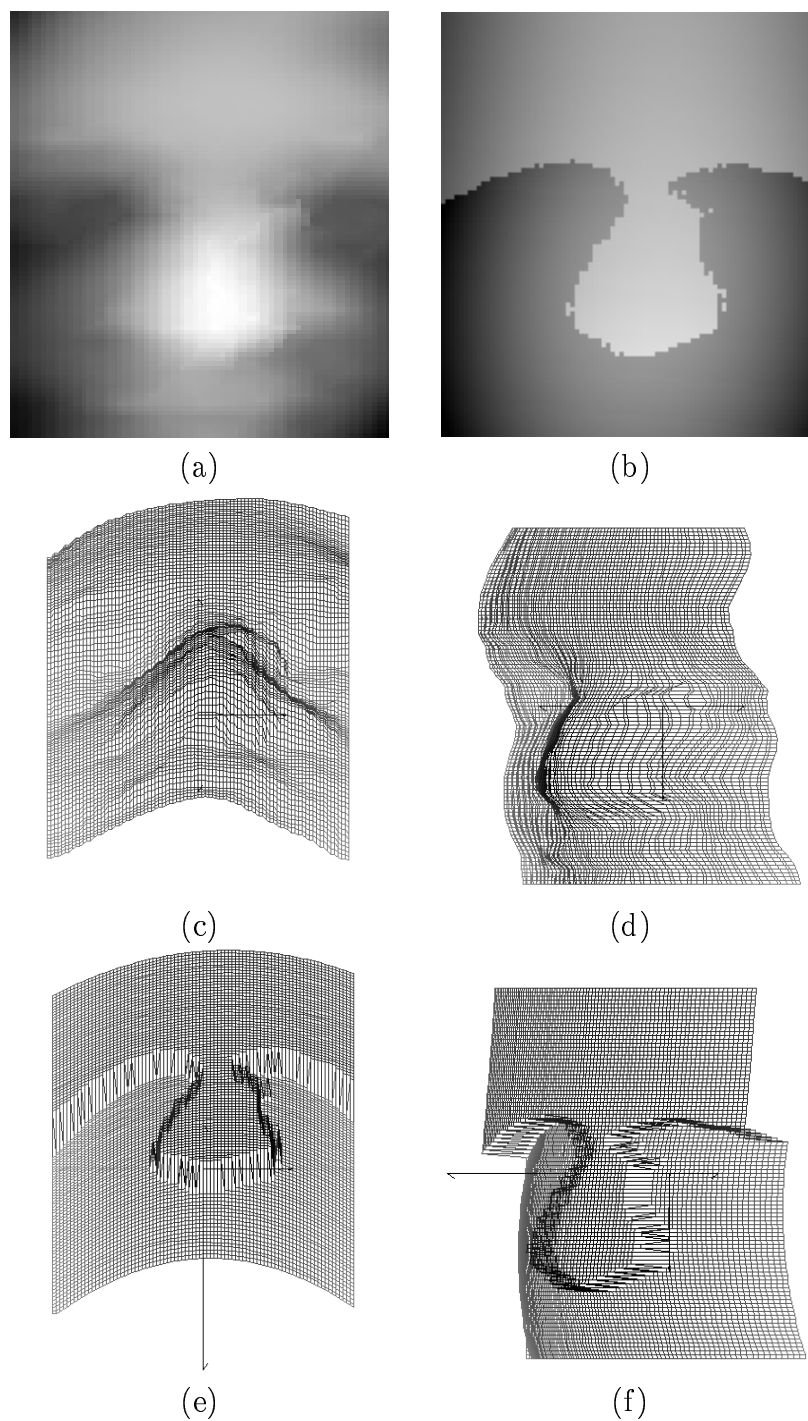


FIG. 5.15 - (a) est la carte de profondeur originale, dont on peut voir deux vues 3D en (c) et (d). (b) est l'approximation par deux fonctions quadratiques (observer les courbures inverses de la région supérieure : front et nez) obtenue par l'approche MMP. (e) et (f) ont les mêmes points de vue que (c) et (d), mais elles présentent l'approximation quadratique.

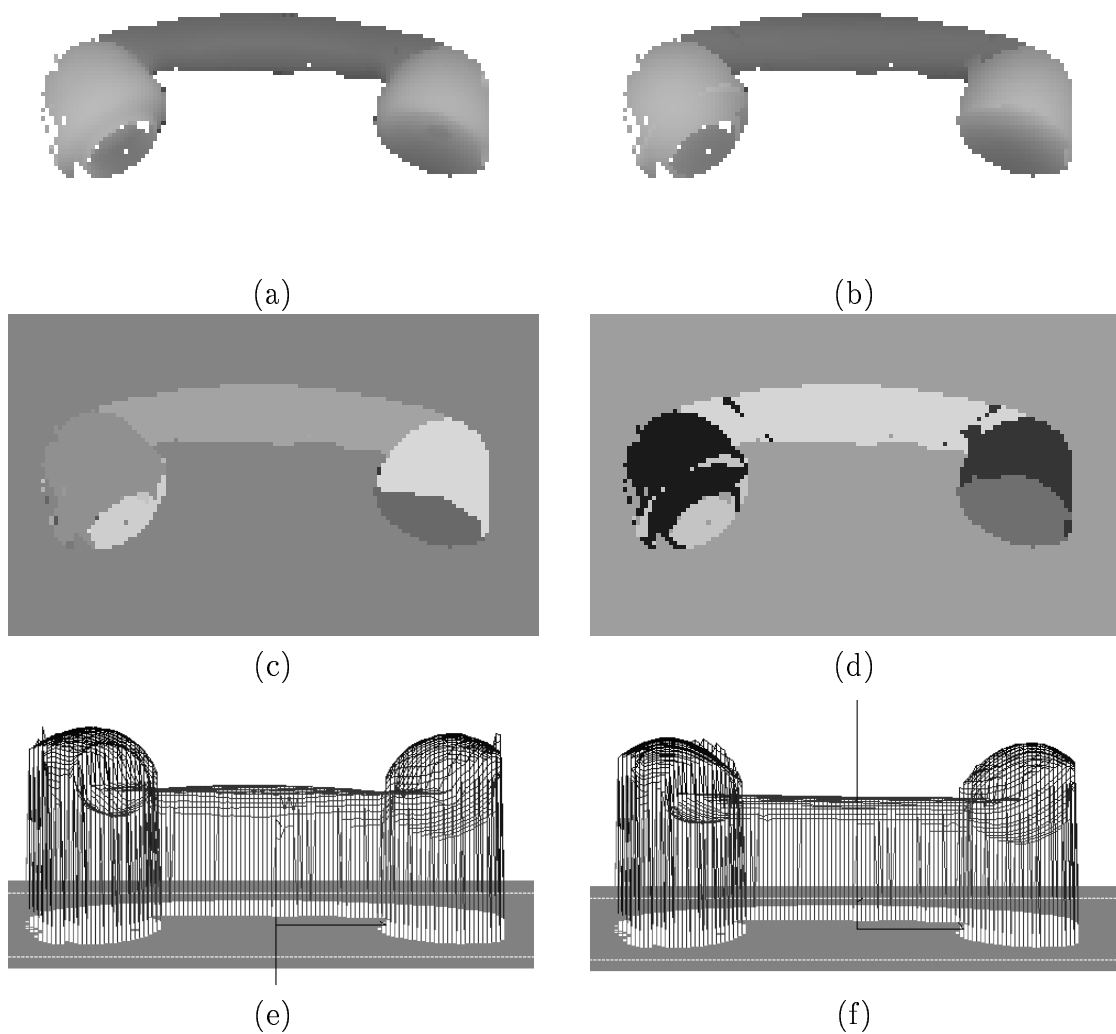


FIG. 5.16 - (a) est la carte de profondeur originale d'un combiné de téléphone, dont une vue 3D est en (e). (b) est l'approximation par cinq fonctions quadratiques obtenue par la méthode MMP dont une vue 3D est en (f). (d) est l'image de labels finale et (c) est l'image de labels initiale qui est obtenue par segmentation avec des régions de profondeur constante.

remplaçant le label par la valeur de la fonction quadratique associée au label, sur chaque pixel.

Pour la figure 5.15, les fonctions initiales sont constantes. La valeur de ces constantes est tirée aléatoirement dans l'intervalle de variation de la profondeur. Mais, une initialisation plus sophistiquée, du type [BM94, YL89], permettrait d'accélérer la convergence, et aurait l'avantage de fournir des résultats plus proches de nos attentes sur ce type de données, lorsque le nombre de classes est plus grand.

Un exemple d'initialisation plus sophistiqué est visible sur la figure 5.16. Une première segmentation est réalisée avec trois classes qui ont des fonctions d'intensité constante. Les régions de grande taille obtenues sont alors au nombre de six (figure 5.16(c)). Cette image de label permet d'initialiser l'algorithme de segmentation avec des régions à profil quadratique, pour aboutir à une approximation très proche de l'image originale (figure 5.16(b) et (f)).

5.4.3 Segmentation des images d'intensité

La figure 5.17 montre la segmentation de l'image d'un visage par des morceaux de fonctions quadratiques. On remarque, sur la série d'images avec des orientations différentes, que cette méthode permet de localiser et de suivre les yeux et la bouche sur cette séquence.

Une autre application de cette méthode consiste à utiliser l'image de labels et les équations des classes pour reconstituer une approximation de l'image. La bonne proximité entre les images d'approximation et originales permet d'envisager l'application de cette méthode à la compression des images en ne transmettant que les fonctions d'intensité et l'image de labels compressée par une technique standard.

5.4.4 Segmentation d'images couleur

La section précédente présente des résultats de segmentation avec des images en niveau de gris. Mais la prise en compte de la couleur est une extension simple. En effet, sur la même surface, quelque soit le canal, l'intensité varie de la même façon (b et c). Ce qui change pour chaque canal est donc juste le terme constant a de la fonction prototype qui approxime la variation de l'intensité. Ainsi dans le cas linéaire, les canaux Rouge, Vert et Bleu sont segmentés à l'aide de la fonction prototype suivante :

$$I_R(u, v) = a_R + bu + cv$$

$$I_V(u, v) = a_V + bu + cv$$

$$I_B(u, v) = a_B + bu + cv$$

Les caractéristiques d'un prototype affine sont donc (a_R, a_V, a_B, b, c) , en couleur, au lieu de (a, b, c) sans couleur.

La couleur permet d'obtenir des segmentations de meilleure qualité en traitant plus d'informations que les niveaux de gris. Comme l'illustre la figure 5.18, notre méthode MMP permet de segmenter correctement en facettes cette scène polyédrique.



FIG. 5.17 - Les images originales d'un visage avec différentes orientations sont sur la première ligne. Sur la deuxième ligne, ces images sont segmentées en 5 classes qui ont une variation quadratique de l'intensité. La troisième ligne présente les images de labels associées. Sur la quatrième ligne, la segmentation du visage vu de face, avec 5, 4, 3 et 2 classes est visible. Les images de label associés sont sur la dernière ligne.

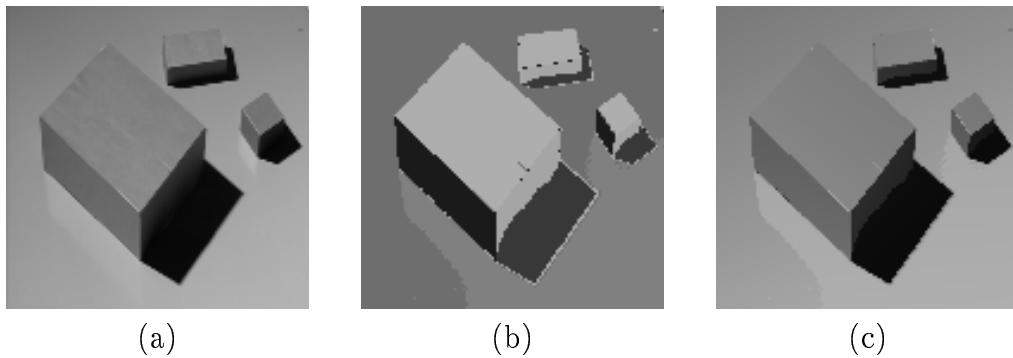


FIG. 5.18 - (a) image originale en couleur. (b) image de labels de la segmentation avec 6 classes. (c) image couleur approximée linéairement.

5.4.5 Discussion

Nous avons présenté l'application de notre approche à la segmentation. Cette méthode présente des propriétés intéressantes de convergence et s'adapte facilement à différents types d'images : images d'intensité en niveaux de gris ou en couleurs, cartes de profondeur.

La classification induit que les régions sont de tailles équivalentes. Ceci est heureux car cela évite l'apparition de régions très petites et non-pertinentes. Néanmoins, il n'y a aucune contrainte dans cette méthode qui oblige les régions à avoir un contour assez régulier. C'est un défaut qu'il serait utile de corriger.

5.5 Application à plusieurs modèles physiques déformables

Dans le cas où l'on désire modéliser un objet 2D ou 3D dont la forme est trop complexe pour bien être approximée par des primitives géométriques simples de base, on recourt à un modèle paramétré de référence plus général, libre de se déformer comme les 'snakes' ou les splines [CCA91, FL94, BD93]. Puisque ces modèles géométriques ont un grand nombre de degrés de liberté, ils ont de bonnes propriétés pour recouvrir les données. Mais quand ces dernières manquent, elles ont des difficultés pour les interpoler [Wei90, FB96]. Faire une bonne interpolation des données manquantes nécessite une contrainte globale sur la forme du modèle. Cette contrainte est souvent induite par l'utilisation d'une famille de surfaces ou courbes plus restreinte comme les super-quadriques et les super-ellipsoïdes [GB88, BV91], des sous-ensembles de courbes algébriques [Tau93], des modèles hybrides [CC94, BCA95], ou des modèles physiques déformables [KWT87, PS91, Nas94].

Nous présentons maintenant une dernière application de notre approche pour des modèles que nous nommons articulés, et qui permettent de faire du recalage fin de contours 2D polyédriques ou non, tout en étant contraints de façon globale sur leur forme.

5.5.1 Modèle déformable constitué de segments articulés

Ces modèles articulés, ou structures polygonales sont une variation des modèles physiques déformables. En effet, les modèles articulés sont aussi constitués d'un ensemble de points pouvant se déplacer, reliés par des segments.

Avec les modèles physiques déformables, les sommets du modèle sont attirés par le point des données le plus proche. Dans notre cas, c'est l'inverse puisque chaque point des données exerce sur les primitives articulées une force d'attraction qui est proportionnelle à l'éloignement. Cette inversion, a priori anodine, change la nature des problèmes qu'il est possible de traiter avec ces deux types de modèles. Cela a déjà été remarqué dans le cadre des 'snakes' sans être utilisé [LM94]. Par construction les modèles physiques, ou les 'snakes', peuvent se recaler finement sur une petite partie des contours à partir d'une position initiale bien choisie. Avec les modèles articulés, l'inclusion est dans l'autre sens. Les primitives articulées doivent donc recouvrir l'ensemble des contours (section 5.5.3). Cette contrainte est modulable en introduisant une classe de rejet qui recueille les points parasites. Un des avantages de modéliser toutes les données est de permettre une initialisation du modèle physique plus distante de la solution finale comme le montre la section 5.5.2.

Un autre avantage des primitives articulées sur les modèles physiques déformables est qu'ils permettent de s'affranchir d'une discrétisation fine du modèle, en pouvant aller jusqu'à une approximation polygonale d'un contour courbe. Les modèles physiques sont en effet réservés aux courbes, alors que les primitives articulées permettent de modéliser tous les types de contours, qu'ils soient polyédriques ou courbes.

Dans un modèle physique, chaque sommet du système subit une force vers le plus proche voisin du contour, qui est proportionnelle à sa distance. Dans notre cas, chaque point des contours de l'image induit une force sur la primitive déformable. Nous supposons donc qu'elle agit sur le plus proche voisin qui appartient à la primitive. Quand ce point est entre deux sommets de la primitive, la force est répartie sur les deux extrémités du segment en la pondérant par la distance au sommet. Par ce biais, nous nous ramenons à une action des données sur les sommets, comme pour les modèles physiques déformables. Cette modification était nécessaire pour pouvoir incorporer les modèles physiques dans le cadre général de l'algorithme MMP décrit dans la section 5.2.

De manière générale, le grand nombre de degrés de liberté d'un modèle de contour actif peut laisser la possibilité au contour de converger et de se déformer vers presque n'importe quoi sous l'influence des forces de proximité, en l'absence de contraintes supplémentaires. La contrainte sur un modèle déformable physique est induite par l'hypothèse que ses sommets sont reliés par des ressorts et que le système doit se comporter selon les lois de la physique, d'où son nom. Pour stabiliser les oscillations du système une force de frottement fluide est ajoutée. Un autre contrainte doit être utilisée pour faire de l'interpolation : la régularité de la forme.

Nous nous sommes inspirés de la démarche modèle déformable physique pour déplacer nos primitives articulées. Le but de ces contraintes est d'empêcher que la primitive initiale se déforme trop brutalement et converge alors vers des positions peu intéressantes. Pour autant, la primitive doit être très facilement déformable pour bien s'adapter aux contours. Or, ce type de comportement est caractéristique des liquides visqueux, qui sont infiniment

déformables au repos, mais présentent une résistance forte aux déformations plus la vitesse est grande. Un liquide visqueux subit une force de frottement interne proportionnelle à la différence de vitesses de chaque point par rapport à ses voisins. Cette force est caractérisée par un coefficient de viscosité $v_{elastic}$. De plus, globalement le liquide est soumis à une force d'amortissement fluide de facteur a_{fluide} . Nous avons donc contraint notre primitive à se déformer dynamiquement comme un liquide visqueux par l'équation suivante :

$$\mu_{sommets}\ddot{X}_i + a_{fluide}\dot{X}_i - v_{elastic}(\dot{X}_i - \dot{X}_{i+1}) - v_{elastic}(\dot{X}_i - \dot{X}_{i-1}) = \Phi_i \quad (5.4)$$

où Φ_i est la force qui s'exerce sur le sommet X_i . $\mu_{sommets}$ est la masse des sommets. Cette équation est résolue itérativement par un schéma d'intégration d'Euler.

Nous avons présenté, pour l'instant, qu'une variante des contours déformables physiques. Voici à présent comment on arrive à intégrer cette variante dans le cadre de la modélisation avec plusieurs prototypes, pour disposer d'une méthode de contours actifs multiples.

Il est clair que le calcul de la force est une autre façon de faire de l'appariement implicitement, comme avec le plus proche voisin. La condition pour que le schéma de l'algorithme MMP fonctionne est que les étapes d'appariement et d'estimation diminuent la distance entre les prototypes et les données. Par analogie, on peut considérer qu'une itération du schéma d'intégration d'Euler de l'équation dynamique (5.4) du contour actif diminue l'écart entre le modèle et les données. Cette itération va donc remplacer l'étape d'estimation géométrique.

Ainsi, une classe correspond à une primitive articulée. Cette dernière a une structure physique bien déterminée par la façon dont sont choisies les connections entre les sommets. Toutefois, chaque sommet de la primitive peut se déplacer. Ainsi, les coordonnées des sommets forment l'ensemble des degrés de liberté de la primitive. Les prototypes F_k sont donc constitués de la liste des coordonnées des sommets de la primitive articulée k (cf. figure 5.19).

L'algorithme de recalage multiple des contours articulés se résume donc à :

étape 0 : Fixer le nombre de classes c et la structure des primitives articulées. Initialiser les sommets des primitives F_k interactivement. (*Donner la distance δ .*)

étape a : Calculer le plus proche voisin X'_{ik} appartenant à la primitive F_k de chaque point X_i . Ajouter à chaque sommets du segment de la primitive F_k qui contient X'_{ik} , la force $\Phi = X_i - X'_{ik}$ pondérée respectivement par la distance de X'_{ik} aux sommets du segment. (*seulement pour k entre 1 et $c - 1$.*)

étape a' : Générer une nouvelle partition en utilisant l'équation suivante du degré d'appartenance flou :

$$u_{ik} = \frac{1}{\sum_{j=1}^c \left(\frac{d(X_i, X'_{ik})}{d(X_i, X'_{ij})} \right)^{\frac{2}{m-1}}}$$

(où $d(X_i, X'_{ic}) = \delta$).

étape a'' : Calculer les nouveaux sommets des primitives F_k en résolvant une itération de l'équation (5.4) sur l'ensemble des sommets. (*seulement pour k entre 1 et $c - 1$.*)

étape a''' : Si les sommets des contours actifs sont stables, il faut s'arrêter, sinon retourner à l'étape a.

5.5.2 Recalage fin sur des contours polyédriques

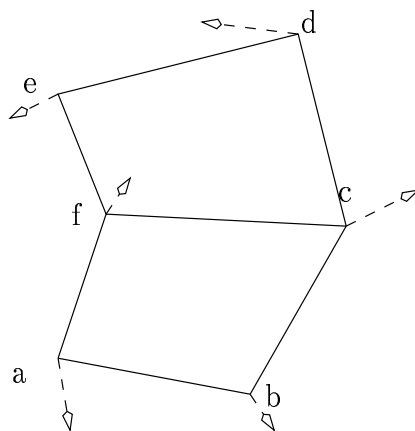


FIG. 5.19 - Structure en 8 digital décrite par ses six sommets (a, b, c, d, e, f) et ses sept arrêtes ($a - b$), ($b - c$), ($c - d$), ($d - e$), ($e - f$), ($f - a$) et ($c - f$). Chaque sommet peut se déplacer en $2D$, d'où 12 degrés de liberté.

Une des qualités des contours déformables, donc des primitives articulées, est qu'il est possible de choisir la structure du modèle en fonction de l'application. Ainsi, sur les images aériennes des figures 5.20 et 5.21, il est clair qu'un modèle convenable d'une toiture de maison à deux pentes est une structure en 8 digital (figure 5.19). La structure géométrique de cette forme de contour est simple à décrire par des listes d'association entre segments et sommets.

Pour obtenir les résultats des figures 5.20 et 5.21, nous avons initialisé le paramètre a_{fluide} d'amortissement à 0.1, la viscosité $v_{elastic}$ à 2.5, la masse $\mu_{sommets}$ à 1. Le pas de temps Δt dans la méthode d'Euler est 0.1. Ces paramètres sont nombreux, mais relativement fixes. Seul le pas Δt doit être diminué en cas de non convergence. Le paramètre δ qui détermine l'influence d'un point est plus délicate à régler, car il définit implicitement la taille de la classe des points rejetés (avec le facteur flou m toujours choisi égal à 1.5). La valeur de $\delta = 2$ pixels, obtenue par des essais, nous a donné satisfaction dans les deux cas.

Néanmoins, une initialisation assez proche du contour du toit est nécessaire, pour ne pas converger vers les contours parasites voisins des toits qui sont assez denses sur ces images.

5.5.3 Recouvrement par plusieurs contours

La figure 5.22 illustre la possibilité de faire du recalage fin avec plusieurs contours actifs en même temps. Avec les approches habituelles, qui font l'appariement des sommets du contour déformable avec le plus proche voisin ou utilisent un champ de force sur l'image dérivée du gradient, la définition du modèle initial pour obtenir le contour central est assez délicate dans cet exemple. En effet, l'initialisation doit être suffisamment proche de la paroi intérieure du crâne pour que le contour actif ne soit pas attiré par les autres contours. Par contre, puisque notre algorithme détermine à chaque étape l'appartenance d'un point du contour à un primitive articulée ou à une autre, l'initialisation peut être beaucoup moins précise et

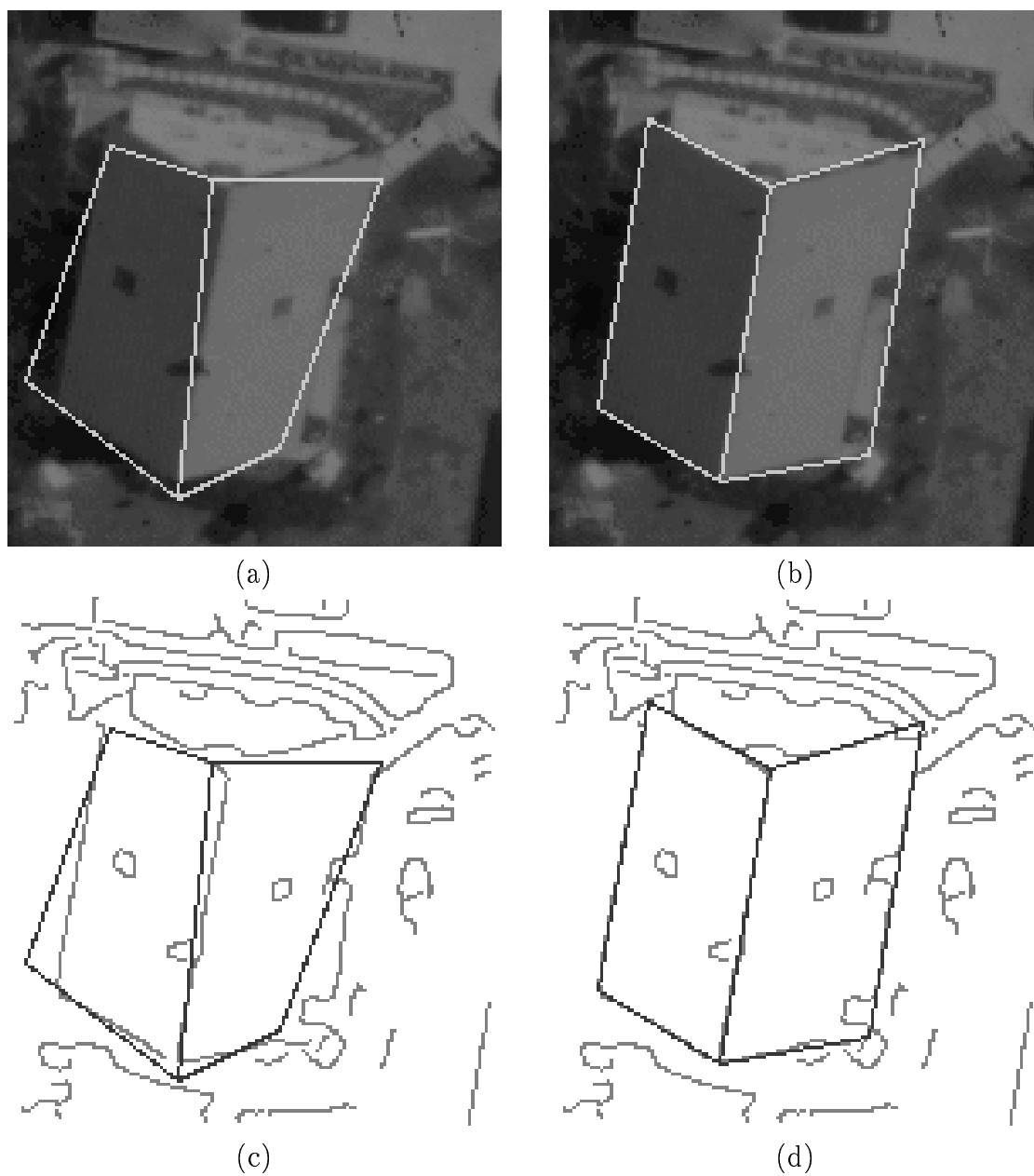


FIG. 5.20 - *Superposition, sur une image aérienne d'une résidence de la primitive articulée initiale (a) et finale (b). (c) et (d) montrent la même superposition sur les contours qui ont servi au recalage.*

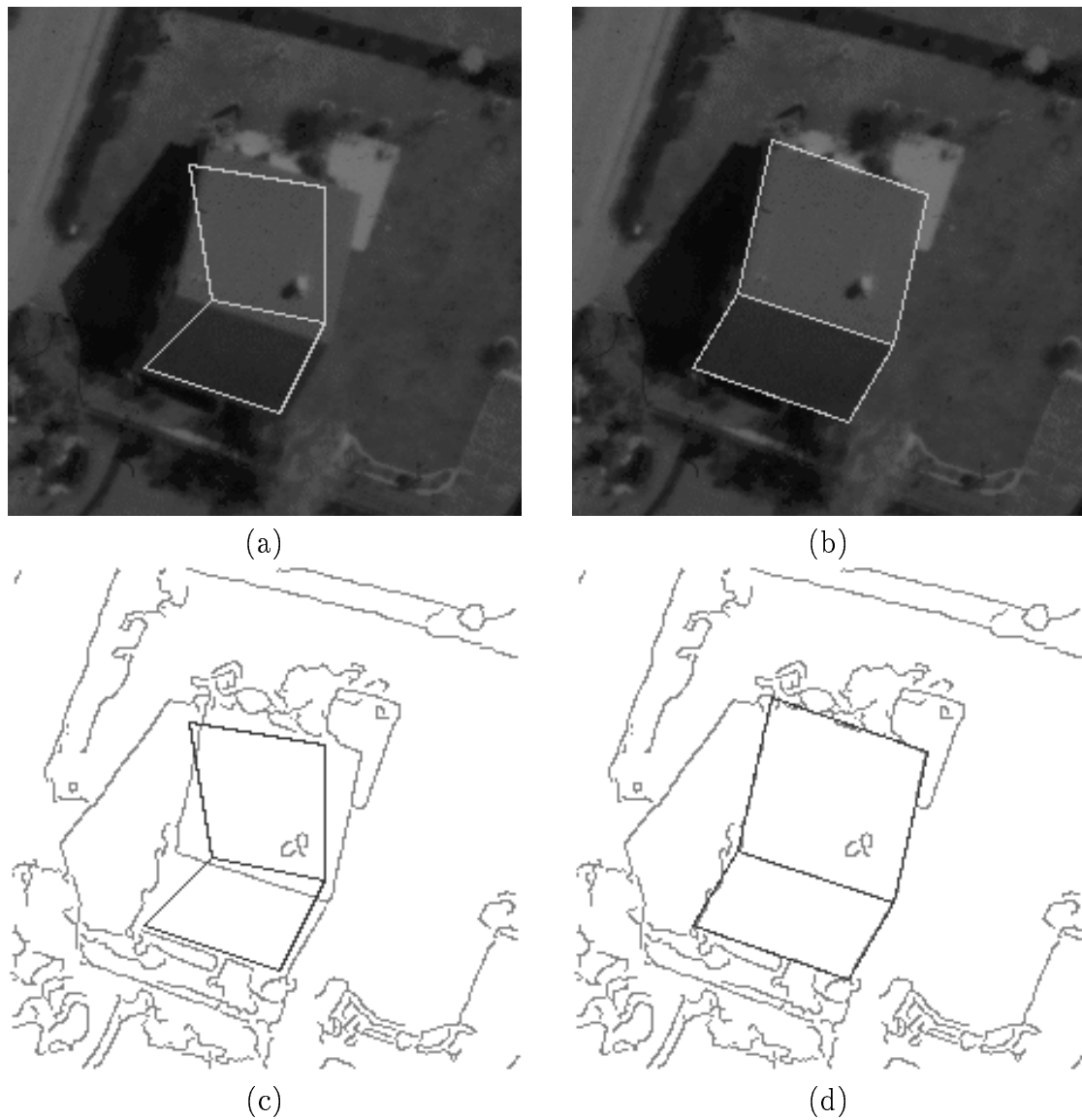


FIG. 5.21 - *Superposition, sur une image aérienne d'une résidence de la primitive articulée initiale (a) et finale (b). (c) et (d) montrent la même superposition sur les contours qui ont servi au recalage.*

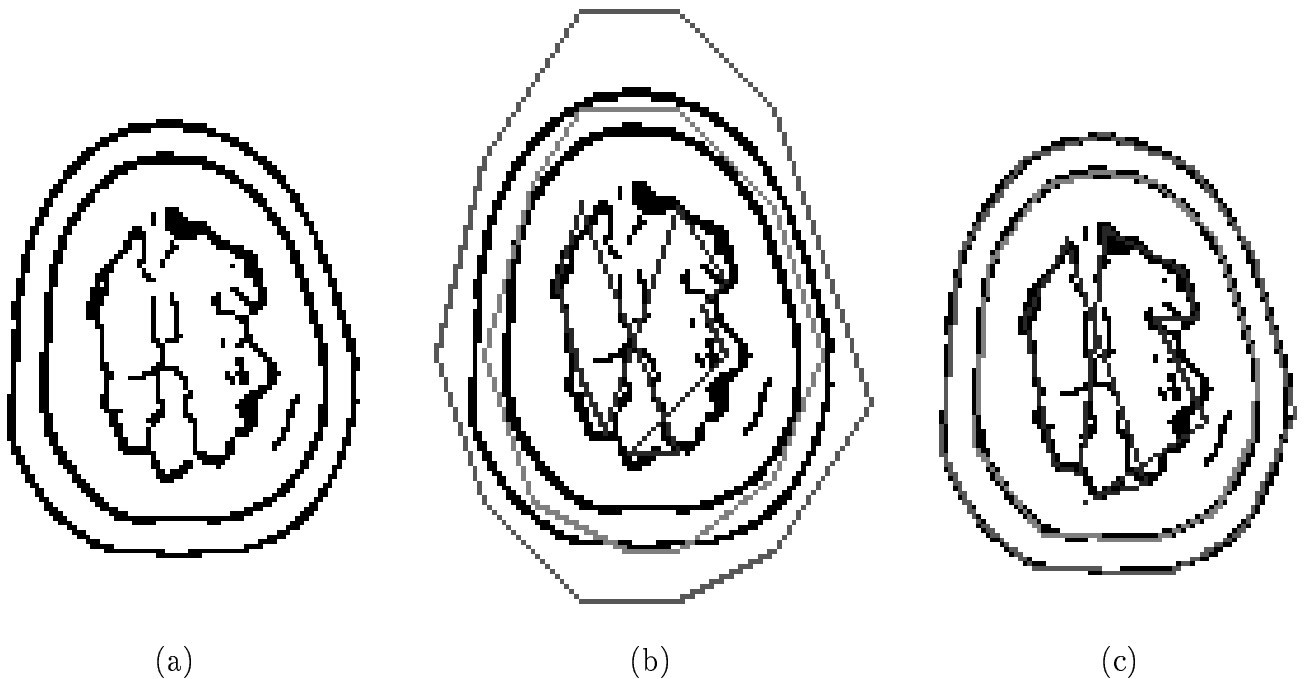


FIG. 5.22 - (a) est la coupe originale du crâne qui présente trois contours : l'extérieur, l'intérieur du crâne, et le contour du cerveau. (b) montre l'initialisation par trois chaînes de 10 segments qui est relativement approximative. (c) est le résultat après recalage fin réalisé en parallèle.

osciller entre les parties intérieure et extérieure du crâne comme dans la figure 5.22(b). Il n'est toutefois pas possible de s'écarter beaucoup plus sans risquer de dégrader les résultats.

La condition de cette plus grande robustesse est la possibilité de recouvrir l'ensemble des contours par des primitives articulées même si elles sont dégénérées.

5.5.4 Discussion

Nous avons donc présenté dans cette section, l'application de notre approche de modélisation multi-prototypes aux contours actifs dans le but de pouvoir modéliser des objets de forme plus compliquée. Cela nous a conduit à introduire un nouveau type de modèle : la primitive articulée qui est une structure formée de segments dont les sommets se comportant physiquement comme un fluide visqueux. Cette primitive présente l'avantage de permettre de modéliser aussi bien les contours courbes que ceux polyédriques. Enfin, l'introduction de l'approche MMP permet d'améliorer la robustesse des modèles physiques à l'initialisation.

5.6 Conclusion

Nous avons présenté une généralisation (GICP) de la méthode ICP à plusieurs importantes transformations géométriques comme les similitudes et les homographies. Mais, comme

cet algorithme ne peut manipuler qu'un unique prototype, nous avons proposé un nouvel algorithme pour faire la modélisation avec plusieurs prototypes. Par là, nous avons abouti à la conception d'une méthode générale de Modélisation Multi-Prototypes (MMP).

La MMP est une méthode pour résoudre le problème du recouvrement ou de modélisation d'un ensemble de données par des prototypes géométriques déformables. En effet, il s'inspire d'une méthode de classification avec ou sans points parasites, et de la version pondérée de l'algorithme GICP pour produire des résultats robustes aux points aberrants et aux positions bruitées des points. La description de l'algorithme MMP est très simple. Néanmoins, il est puissant et général, car il combine itérativement une étape d'appariement et une étape d'estimation. De plus, il est contrôlé par seulement deux paramètres importants : le nombre de classes c et la distance δ des points probablement parasites. Le nombre de classes est un paramètre de contrôle efficace. Mais, pour certaines applications, il est intéressant d'étudier, à l'avenir, comment on peut déterminer ce paramètre automatiquement.

Enfin, nous voulons insister sur le fait que notre approche qui consiste à mélanger classification et recalage avec appariement vers le plus proche voisin est une voie de recherche prometteuse. Nous avons montré que ce type d'approche peut être appliqué avec succès sur des fonctions pour faire de la segmentation d'images, et sur des structures articulées, variantes des modèles physiques déformables pour faire du recalage fin de contours 2D déformables. L'application aux splines semble aussi possible. Une autre application, à des surfaces maillées déformables, est aussi envisageable dans un esprit proche des travaux de [FL94].

Si la généralisation à des projections en dimension p de prototypes en dimension q pose des problèmes de stabilité (voir annexe C), c'est un sujet d'études intéressant par ces nombreuses applications, en particulier pour le suivi en parallèle de plusieurs objets dans une séquence d'images.

Conclusion

Contributions

Les deux contributions principales de cette thèse sont :

- d'ordre méthodologique pour traiter des problèmes de vision qui combinent estimation géométrique et appariement,
- de nature algorithmique, puisque nous proposons un algorithme général pour résoudre des problèmes où il faut trouver les caractéristiques géométriques des prototypes qui recouvrent au mieux l'ensemble des données.

Notre méthodologie pour poser les problèmes de modélisation automatique par ordinateur consiste à les analyser sous les problématiques duales de l'estimation des caractéristiques géométriques et de la sélection des données (ou appariement de données) pertinentes. On cherche alors à poser le problème de façon à découpler au mieux l'estimation géométrique de l'appariement. Par cette analyse, on tente de le classer dans l'une des quatre catégories suivantes de problèmes bien posés de difficulté croissante :

- estimation pure avec appariement connu,
- estimation indépendante de l'appariement,
- estimation avec recouvrement complet des données par plusieurs prototypes géométriques.
- estimation avec sélection de la partie des données à traiter,

L'avantage de suivre cette méthodologie est qu'elle conduit généralement à des algorithmes robustes vis-à-vis de la diversité des jeux de données et des bruits ou petites erreurs. Cette dernière propriété permet, si on y veille, d'obtenir des résultats précis.

Nous avons appliqué l'approche précédente pour traiter certaines des étapes d'un système de modélisation semi-automatique qui produit une description géométrique et photométrique.

Ainsi, notre cadre méthodologique nous a conduit à ne pas sortir de la façon classique de poser le problème de la calibration d'une caméra : estimer les paramètres intrinsèques et extrinsèques à l'aide d'une mire composée de cibles. Mais nous l'avons traité selon une approche qui utilise les moments géométriques des cibles. En effet, nous avons montré que

ces moments peuvent être calculés avec une précision sous-pixel dans une image, et sont donc une alternative intéressante à des mises en correspondance locales des contours des cibles.

L'application de notre approche à la reconstruction 3D de facettes à partir de régions, nous a fait aborder le problème de façon inhabituelle. Nous avons en effet montré que la configuration rectifiée du couple de caméras permet de reconstruire le support de la face, en utilisant seulement les moments des régions gauche et droite, sans qu'il soit donc nécessaire de faire des appariements locaux sur les contours.

Nous avons ensuite introduit la classification floue robuste pour obtenir une localisation, et un positionnement 3D approché, ce dernier étant ensuite raffiné. Cette approche, qui peut s'appliquer à d'autres problèmes de la vision 3D, illustre l'avantage qu'il y a à concevoir un algorithme en deux étapes, l'une imprécise géométriquement mais robuste dans toutes les configurations, et l'autre apportant la précision mais moins robuste par nature. En effet, les propriétés, précision et robustesse, sont souvent difficilement compatibles.

Enfin, nous avons proposé une approche originale et unifiée pour résoudre le problème peu traité de la modélisation géométrique 3D (ou 2D), de façon automatique, en recouvrant l'ensemble des données par des prototypes géométriques déformables qu'ils soient anguleux ou courbes, ce qui ouvre une voie intéressante dans l'interprétation générique d'objets. Cela nous a conduit à développer, selon le même principe, trois algorithmes qui respectivement :

- segmentent une image en régions où l'intensité varie linéairement, quadratiquement, ou polynômialement,
- combinent le recalage de plusieurs structures polygonales articulées et déformables de même type,
- recouvrent l'ensemble des données par des parallélépipèdes ou des ellipsoïdes.

Perspectives

La modélisation par recouvrement de prototypes trouve des applications bien au delà de la modélisation polyédrique, dans le domaine médical par exemple, entrevu dans le dernier chapitre. Le développement de cette méthode pour traiter d'autres types de données, en particulier l'étude des conditions de son application et les avantages qu'elle peut apporter, ouvre une voie de recherche originale et passionnante.

Le système de modélisation stéréoscopique développé à l'heure actuelle ne peut prétendre pouvoir modéliser avec succès des environnements complexes. Car même s'ils sont polyédriques, l'hypothèse d'albédo constant sur les facettes 3D est une contrainte trop sévère sur les objets. Néanmoins, les idées proposées lors de sa conception sont réutilisables pour développer un système encore plus ambitieux de modélisation 3D surfacique. Celui-ci, pour être plus efficace, devra intégrer l'approche par corrélation pour traiter les parties fortement texturées des images, éventuellement les contours pour appuyer la corrélation, et enfin l'approche région pour les parties à l'intensité approximativement constante ou linéaire.

En effet, l'accroissement de l'utilisation des simulations nécessite des outils de modélisation géométrique et photométrique de l'environnement ou d'objets, qui soient les plus automatiques possibles. Les capteurs actifs de type caméra laser de profondeur ont fait des

progrès importants ces dernières années. Ils arrivent à produire des données 3D très précises, et les accompagnent d'informations sur la couleur des objets. Pour autant, les capteurs passifs de type caméra vidéo que nous avons utilisés, restent potentiellement les seuls à pouvoir modéliser géométriquement des objets qui se déplacent ou se déforment. La résolution de ce problème, de par sa grande utilité et sa difficulté, donnera à la vision par ordinateur toutes ses lettres de noblesse.

Annexe A

Calibration Radiométrique de Caméra à focale variable

A.1 Introduction

Dans un système de vision, il est utile, pour faciliter l'analyse des images, de connaître la physique du capteur qui les fournit. En particulier, il est important d'avoir les caractéristiques géométriques et radiométriques de la caméra puisque l'on veut obtenir des informations de nature géométrique comme photométrie. La modélisation géométrique du capteur et les méthodes d'estimation de ses paramètres sont traités par la calibration de caméra. La calibration géométrique de caméra est un sujet bien exploré [Tsa86, Tos87, DRLR88, HCLL89, MM91, Bey92] (chapitre 2) et l'on dispose de nombreux modèles de caméra à focale fixe [Bey92, WCH90]. Par contre, si la calibration radiométrique est déjà bien traitée en astronomie, elle est encore que rarement utilisée dans la pratique en vision par ordinateur [HK94]. Elle permet pourtant de supposer que, après correction, chaque pixel a un comportement identique du point de vue photométrique. Or, de nombreuses méthodes d'analyse des images, comme les détecteurs de contours, les méthodes de reconstruction par corrélation et le 'shape-from-shading' reposent sur l'hypothèse que chaque pixel se comporte de la même façon.

Nous nous intéressons dans cette annexe uniquement à la partie radiométrique de la calibration. Il y a de nombreuses sources de perturbations radiométriques, comme les divers défauts de construction qui peuvent affecter la matrice CCD. Par exemple, une cellule CCD a une remarquable linéarité mais, suivant le pixel, sa réponse peut avoir une pente différente, d'où des écarts de sensibilité. L'objectif de la caméra perturbe lui aussi la radiométrie à cause du phénomène de 'vignettage' en particulier, qui produit un assombrissement sur les bords de l'image.

Ces perturbations sont rappelées succinctement dans la partie A.2. Comme nous avons pu le constater expérimentalement (partie A.4), les variations réelles de radiométrie d'un pixel à un autre sont difficiles à modéliser complètement. La correction passe donc par la prise d'images de référence permettant de calibrer indépendamment chacun des pixels (partie A.3). Dans la partie A.4, nous présentons des résultats sur l'estimation des corrections à apporter sur la caméra du laboratoire quand la focale varie aspect peu étudié, et les

observations expérimentales sont discutées. Enfin, dans la dernière partie, l'intérêt de la calibration radiométrique est illustré sur quelques exemples.

A.2 Défauts radiométriques

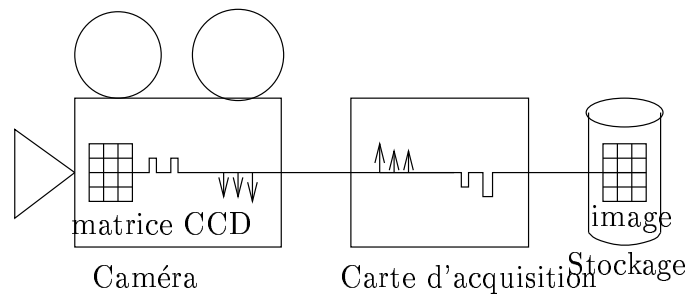


FIG. A.1 - *Système d'acquisition d'images.*

Les pixels sur une image peuvent avoir un comportement radiométrique différent du fait des défauts de construction de la matrice CCD [Bui89] et des écarts d'éclairement dus à l'optique de l'objectif [And71]. Il est aussi nécessaire de modéliser le bruit du capteur pour avoir un modèle complet du processus de formation des images [HK94]. On ne s'intéresse pas ici aux déformations de type géométrique : distorsions, synchronisation du balayage.

A.2.1 Écarts de sensibilité de la matrice CCD

Une cellule CCD, par effet photoélectrique, produit un nombre d'électrons proportionnel au nombre de photons incidents durant le temps d'intégration. Le rendement quantique est très bon. Comme les défauts de linéarité sont inférieurs à 0.1% sur une large plage de fonctionnement, la réponse d'une cellule CCD est donc proportionnelle à l'éclairement reçu par sa surface. Le gain de la réponse dépend de l'épaisseur de la couche de silicium de la cellule, dont la régularité sur toute la surface de la rétine est difficile à maîtriser parfaitement lors du processus de fabrication. Les variations de sensibilité peuvent facilement atteindre 1% sur l'ensemble de la rétine. De plus, différents défauts, dits cosmétiques, affectent localement la qualité de la surface sensible. Par exemple, il est fréquent de trouver des pixels insensibles dans une matrice CCD.

Les écarts de sensibilité entre pixels, de par la linéarité du capteur, peuvent donc être modélisés par un biais multiplicatif $g_{sensibilité}(u, v)$ propre à chaque pixel (u, v) (voir équation A.1).

A.2.2 Courant d'obscurité

Généralement, la matrice CCD fonctionne à température ambiante. En conséquence, un signal thermique s'ajoute à la réponse de chaque élément. C'est le courant d'obscurité. D'un pixel à l'autre, le courant d'obscurité peut varier de 10%. Nous avons observé une valeur de cet

ordre avec notre caméra (partie A.4.2). Le courant d'obscurité est bien sûr proportionnel au temps d'intégration. Par là-même, il reste faible pour des temps d'intégration très courts tels ceux que l'on rencontre habituellement en vidéo. Si un temps d'intégration long est utilisé comme dans les applications astronomiques, il est alors nécessaire de refroidir le capteur pour diminuer le bruit thermique [Bui89]. Ce biais est additif. Il peut donc être modélisé par une variable aléatoire $I_{obscurité}(u, v)$. A ce dernier s'ajoute le bruit $I_{transmission}(u, v)$ dû à la transmission entre la caméra et la carte d'acquisition. $I_{transmission}(u, v)$ est supposé centré et indépendant d'un pixel à l'autre. Avec notre système, ce bruit est assez important à cause du passage numérique/analogique en sortie de caméra puis analogique/numérique en entrée de la carte d'acquisition des images (figure A.1).

A.2.3 Variations d'éclairement dues à l'objectif

Un objectif ne produit pas un éclairage uniforme de la rétine même si un écran blanc uniforme est observé. Il y a un assombrissement aux bords qui varie avec la focale. Nous avons observé une variation maximum de 4% avec notre objectif (voir partie A.4.4). Comme pour l'écart de sensibilité des cellules CCD, la variation d'éclairement est modélisable par un biais multiplicatif $g_{objectif}(u, v)$. La modélisation de cette variation $g_{objectif}(u, v)$, en fonction de la position du pixel, est possible si l'optique de l'objectif est complètement connue. Malheureusement, ces informations ne sont que très rarement données par le constructeur. Néanmoins, de par la conception des objectifs, la variation d'éclairement doit présenter une symétrie circulaire autour du centre de la caméra. Nous allons donc décrire l'assombrissement observable pour des modèles très rudimentaires d'objectif.

Modèle sténopé

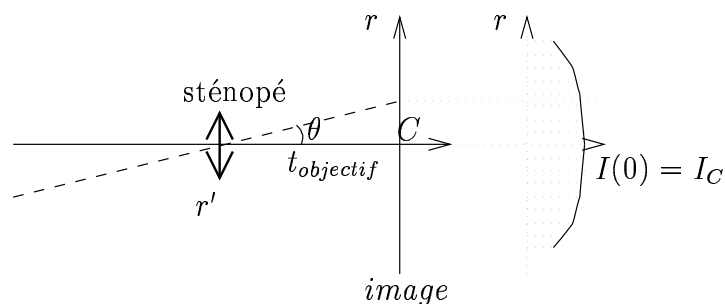


FIG. A.2 - Profil de l'assombrissement aux bords d'une lentille en trou d'épingle de tirage $t_{objectif}$.

Soumise à un éclairage uniforme (écran blanc infini), une lentille mince de petit diamètre éclaire le plan image avec une décroissance circulaire, fonction de la distance à son centre C (figure A.2). Dans ce cas, l'éclairage maximum $I(0)$ s'observe donc au centre C et l'assombrissement au bord est donné par la formule suivante [Hor86]:

$$\frac{I(\theta)}{I(0)} = \cos^4(\theta) = \frac{1}{\left(1 + \left(\frac{r}{t_{objectif}}\right)^2\right)^2}$$

où θ est l'angle selon lequel le pixel voit l'objet réel (figure A.2). Le paramètre $t_{objectif}$ est assimilé couramment à une focale, mais nous préférons le terme, plus exact en optique, de tirage.

Comme les rétines des caméras sont généralement petites, l'assombrissement est assez faible. Par exemple, une rétine de 10 mm, avec un objectif de 50 mm, est assombrie au bord d'au maximum 2%. Outre une variation globale de l'éclairage de la rétine (variation de $I(0)$), un changement de tirage induit une variation locale de l'assombrissement en un point donné de la rétine. Ainsi, le doublement du tirage de l'objectif précédent divise l'écart d'assombrissement en chaque point par quatre.

Cette description est valide pour un objectif à une seule lentille, mais la décroissance est encore plus forte pour un objectif réel à cause du phénomène dit de 'vignettage' des objectifs.

Objectif simplifié

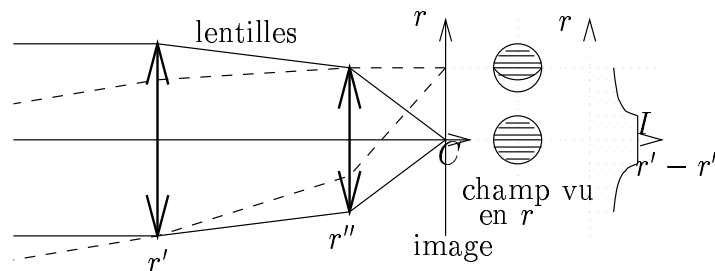


FIG. A.3 - *Phénomène de vignettage d'un objectif. Suivant sa position, un point r de l'image ne voit pas la même portion du champ de vue.*

L'épaisseur de l'objectif constitué d'une série de lentilles ne permet pas à tous les pixels de recevoir le même flux lumineux (figure A.3). Le vignettage est d'autant plus important que l'objectif est ouvert [And71]. La non-uniformité d'éclairage peut aisément atteindre 10% entre le centre et le bord du champ.

Dans le cas où l'objectif peut être réduit à deux diaphragmes de diamètre r' et r'' comme dans la figure A.3, la fonction d'assombrissement s'obtient par autocorrélation des deux disques [Bru59] (figure A.4) :

Objectif réel

L'effet de vignettage s'ajoute à la variation avec l'angle solide décrite dans les sections précédentes. Dans les objectifs réels, d'autres phénomènes interviennent telles la réflexion et les diffractions qui compliquent l'obtention de l'expression formelle de $g_{objectif}(u, v)$.

De nombreux objectifs zooms possèdent quatre groupes optiques : une lentille Frontale, un Variateur, un Compensateur et un Relai optique (groupes F, V, C, R dans la figure A.5).

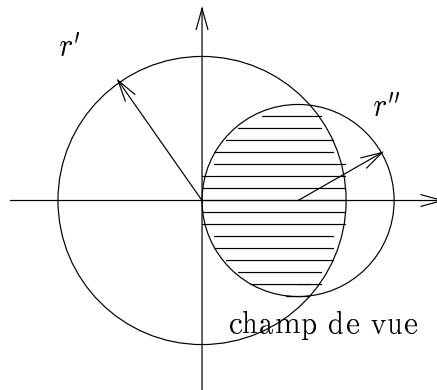


FIG. A.4 - Autocorrélation de deux disques pour calculer l'assombrissement dû au vignettage.

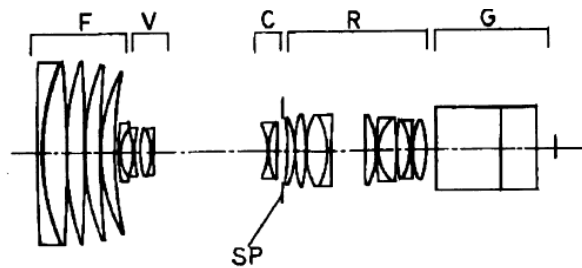


FIG. A.5 - Dispositif optique réel de notre objectif Canon J15x9.5B 1:1.8 à focale variable extrait du brevet associé [Usu94].

Chacun de ces groupes assure une fonction privilégiée. La lentille frontale F bouge seule pour assurer la mise au point. Le variateur V permet d'obtenir de grandes variations de la longueur focale. Le compensateur C est l'élément qui maintient constant le plan de focalisation. Enfin, le relai optique R assure la convergence des rayons lumineux dans le plan de l'image.

La variation de la focale introduit dans les objectifs réels des changements de l'assombrissement au bord, qui sont des fonctions du déplacement des lentilles. Ces variations sont relativement complexes à modéliser et calculer, si bien que les opticiens se contentent pour un objectif réel d'estimer les variations radiométriques par des techniques de lancer de rayons de type Monte-Carlo.

A.3 Correction radiométrique

D'après la description précédente, le modèle radiométrique de la caméra est pour chaque pixel de coordonnée (u, v) :

$$\begin{aligned}
I_{lue}(u, v) &= g_{sensibilité}(u, v)g_{objectif}(u, v)I_{exacte}(u, v) + I_{obscurité}(u, v) + I_{transmission}(u, v) \quad (\text{A.1}) \\
&= g(u, v)I_{exacte}(u, v) + I_0(u, v)
\end{aligned}$$

où I_{exacte} est l'intensité qui serait lue si la caméra était parfaite. Le modèle est de type linéaire. La variation d'illumination de la rétine produite par l'objectif s'avère le phénomène le plus important avec les écarts de sensibilité. Mais tandis que l'on peut modéliser les variations d'illumination par un polynôme à symétrie circulaire, il se révèle impossible d'avoir le modèle de la répartition des écarts de sensibilité des cellules CCD, d'après nos expérimentations effectuées sur la caméra couleur du laboratoire. Il est donc nécessaire de calibrer radiométriquement sans modèle a priori, pixel par pixel.

Pour chacun des pixels (u, v) , on cherche à retrouver son intensité exacte I_{exacte} à partir de l'intensité observée I_{lue} . Il faut donc obtenir $g(u, v)$ et $I_0(u, v)$. L'image des 'offsets' $I_0(u, v)$ est obtenue par la prise d'une image avec l'objectif fermé, à la température de travail. D'autre part, $g(u, v)$ s'obtient à un facteur près, en prenant l'image d'une surface de pigment uniforme éclairée uniformément. Cette image se nomme Plage de Lumière Uniforme (PLU) ou 'flat-field' en astronomie. La procédure de correction radiométrique est très simple au niveau calcul, puisqu'il suffit de soustraire l'image d'offsets à celle qui doit être calibrée, puis de diviser le résultat par la PLU normalisée. Par contre, l'obtention pratique d'images de référence de bonne qualité est plus difficile.

A.4 Expérimentations

Pour obtenir des images d'offsets et PLU de bonne qualité, il est important d'éliminer préalablement le bruit temporel de transmission. Nous avons donc réalisé des expérimentations pour étudier ce bruit avant de nous intéresser à l'estimation du courant d'obscurité et des écarts de sensibilité. Le système d'acquisition utilisé pour ces mesures est constitué d'une caméra CCD Sony couleur XC-007 avec un objectif Canon J15×9.5B 1:1.8 à focale variable et d'une carte d'acquisition S2200 de Datacell.

A.4.1 Bruit temporel

Le bruit temporel contenu dans les images est produit essentiellement par la transmission entre la caméra et la carte d'acquisition. Ce bruit est assez important. Il est supposé gaussien centré, comme en témoigne l'histogramme du bruit observé de la figure A.6. Celui-ci a été obtenu par différence d'images acquises à des instants différents.

Il est assez simple d'éliminer ce bruit temporel par la moyenne d'une série d'images de la même scène prise à des instants t différents. La figure A.7 montre la variation de la variance du bruit en fonction du nombre d'images moyennées par rapport à la même image de référence. La décroissance de la courbe est fonction de la racine de l'inverse du nombre d'images. La courbe théorique de la variance v est facile à obtenir à partir de l'équation (A.1):

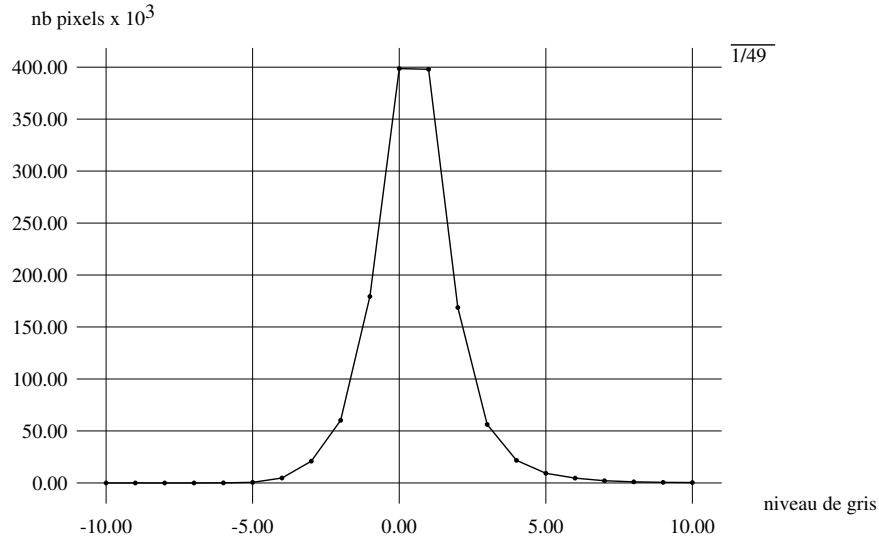


FIG. A.6 - Histogramme du bruit de transmission entre la caméra et la carte d'acquisition. L'allure est gaussienne avec un écart type de 1.5 niveau de gris.

$$\begin{aligned} \text{var}(I_{lue} - I_{lue}^{ref}) &= \text{var}(I_{obscurité} - I_{obscurité}^{ref} + \frac{1}{n} \sum_t I_{transmission}^t - \frac{1}{n_{ref}} \sum_t I_{transmission}^{t ref}) \\ &= (\frac{1}{n} + \frac{1}{n_{ref}}) \text{var}(I_{transmission}) + 2\text{var}(I_{obscurité}) \end{aligned}$$

où n est le nombre d'images moyennées et n_{ref} le nombre d'images moyennées dans l'image de référence. $\text{var}(I_{transmission})$ est alors la variance du bruit temporel de transmission entre la caméra et la carte, tandis que $\text{var}(I_{obscurité})$ est la variance du bruit thermique des pixels de la rétine.

Par régression linéaire avec la courbe $\frac{1}{n} + \frac{1}{n_{ref}}$, nous avons obtenu 1.52 niveau de gris comme estimation de l'écart type de $I_{transmission}$, et 0.27 niveau de gris comme estimation de l'écart type de $I_{obscurité}$. Dans la figure A.7 sont tracées deux courbes, l'une observée, l'autre théorique. Elles se superposent très bien, ce qui valide partiellement le modèle radiométrique utilisé.

En conséquence $I_{transmission}$ est, avec notre dispositif, une variable aléatoire gaussienne centrée de variance 1.5 niveau de gris. Ainsi, faire la moyenne de 25 images de la même scène permet d'éliminer presque entièrement le bruit temporel de transmission.

A.4.2 Courant d'obscurité

Le courant d'obscurité engendre un fond bruité sur toute l'image. Nous avons mesuré que ce fond a une valeur $I_{obscurité}(u, v)$ entre 10 et 11 niveaux de gris sur la plage de température

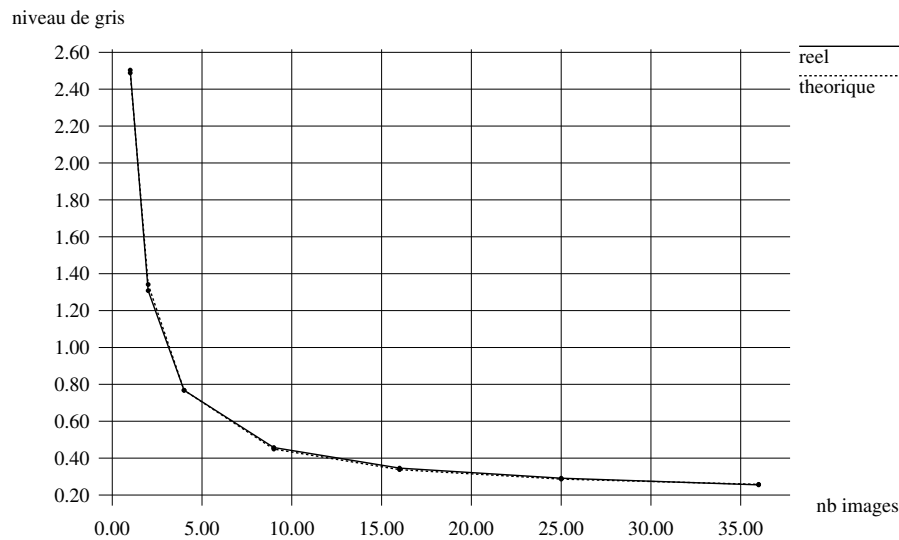


FIG. A.7 – Variation théorique et réelle de la variance du bruit en fonction du nombre d'images moyennées.

ambiante. L'écart type du courant d'obscurité qui a été obtenu dans la partie précédente permet de déduire que d'un pixel à un autre, le courant d'obscurité varie de 3% en moyenne. Néanmoins, la valeur 0.27 de l'écart type du courant d'obscurité est assez faible pour être négligée.

En conséquence, l'image d'offsets $I_{obscurité}$ peut donc être supposée constante sur toute la surface sensible, sans trop d'erreurs.

A.4.3 Écart de sensibilité

En retirant l'objectif et en le remplaçant par un papier diffusant, la surface sensible est alors éclairée uniformément et l'image obtenue donne à un facteur près le $g_{sensibilité}(u, v)$ de chaque pixel. Avec notre caméra, nous avons observé par comparaison des intensités des pixels de ce type d'image, une variance importante de 1.8 à 5.5 niveaux de gris suivant le canal de couleur (figure A.8).

Même après soustraction de la variance du bruit de transmission, on obtient des écarts type de $g_{sensibilité}(u, v)$ importants. Les écarts de sensibilité doivent donc être systématiquement corrigés. L'écart de sensibilité, estimé à 4% dans notre cas, peut produire jusqu'à une erreur d'une dizaine de niveaux de gris, à l'intensité maximum.

La figure A.9 montre en fausses couleurs l'image obtenue avec l'objectif Canon J15×9.5B, après soustraction de l'offset $\overline{I_{obscurité}}(u, v)$ pour la composante Rouge. D'après cette figure, les écarts de sensibilité présentent donc plutôt une symétrie rectangulaire. On retrouve le même aspect des variations pour les autres canaux de couleur.

Ainsi, dans la figure A.10, coupe de l'image de la figure A.9 après lissage, le profil des

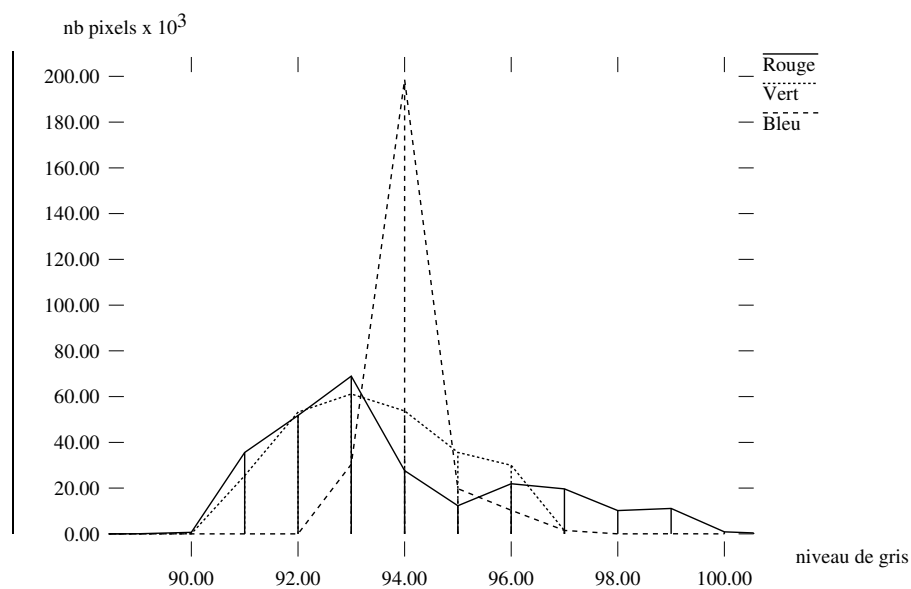


FIG. A.8 - *Histogramme pour les canaux Rouge, Vert et Bleu de l'écart de sensibilité de notre caméra CCD.*

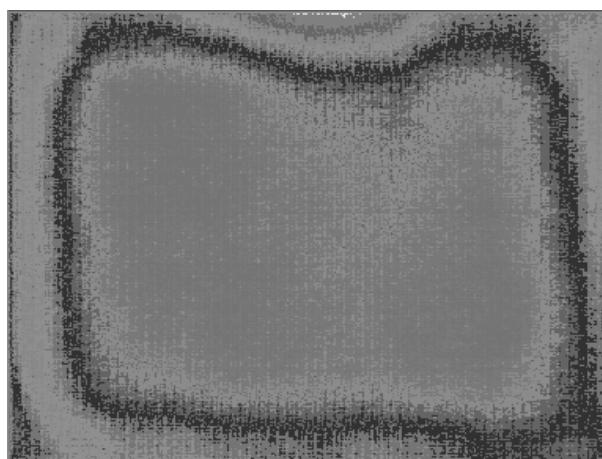


FIG. A.9 - *Image en fausse couleur des écarts de sensibilité $g_{sensibilité}(u, v)$ de la matrice CCD de notre caméra sur le canal Rouge.*

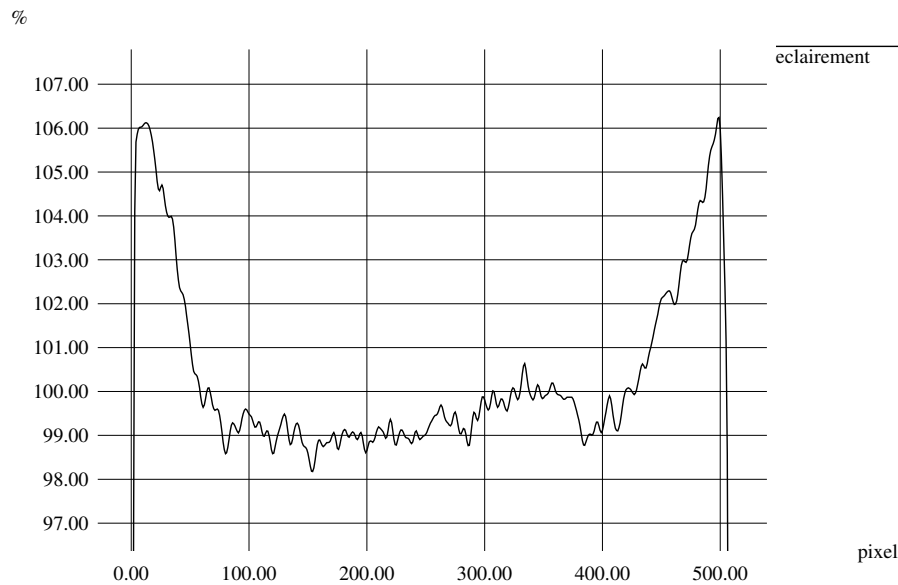


FIG. A.10 - Coupe de l'image de la figure A.9 après lissage.

écarts de sensibilité est en forme de trapèze. Néanmoins, la forme de la répartition sur la matrice CCD de ces écarts reste difficile à modéliser.

Les variations des écarts de sensibilité sont très liées à la technique de construction de la matrice CCD. L'image $g_{sensibilité}(u, v)$ obtenue ne peut donc pas être utilisée pour une autre caméra.

A.4.4 Variation d'éclairement

Pour obtenir les variations d'éclairement $g_{objectif}(u, v)$, il est nécessaire de photographier une surface à radiométrie constante, sous un éclairage uniforme comme le soleil. La figure A.11 montre en fausses couleurs l'image obtenue avec l'objectif Canon, après soustraction de l'offset $I_{obscurité}(u, v)$ et correction des écarts de sensibilité $g_{sensibilité}(u, v)$. Les variations d'éclairement présentent une symétrie circulaire. Cet effet s'explique par la construction à symétrie radiale de l'objectif. Sur la figure A.11, la symétrie n'est pas parfaitement circulaire car la carte $g_{sensibilité}(u, v)$ n'est pas très précise. En effet, obtenir un éclairage uniforme de la rétine n'est pas aisé dans la pratique.

Dans nos tests, les variations maximales de $g_{objectif}(u, v)$ sont de 4%, ce qui correspond à une erreur d'une dizaine de niveaux de gris pour l'intensité maximum enregistrable. L'objectif produit donc un des défauts radiométriques prépondérants du même ordre de grandeur que les écarts de sensibilité $g_{sensibilité}(u, v)$.

Nous avons réalisé le même type d'expérimentations avec un autre objectif, un 80mm 1:2.8 Tessar de Zeiss-Opton qui nous a fourni des résultats analogues.

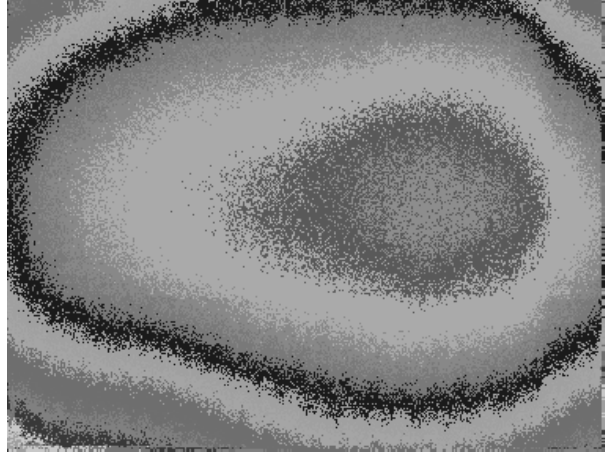


FIG. A.11 - Image en fausse couleur des variations d'éclairement $g_{\text{objectif}}(u, v)$ produit par notre objectif sur le canal Rouge.

A.4.5 Stabilité de la variation d'éclairement

En outre, nous avons étudié la variation d'éclairement lors de la modification des caractéristiques de l'objectif, du zoom, de la mise au point et du diaphragme. Nous avons constaté avec une certaine surprise que le changement de ces paramètres ne produit qu'une variation relativement faible de l'éclairement (figure A.12). Lors de nos expérimentations, nous avons observé un biais moyen de 1.5 niveaux de gris, malgré la présence de poussières sur le parcours lumineux. En effet, celles-ci produisent des variations locales de l'éclairement visibles à courte focale.

Ces expérimentations sont a priori difficiles à réaliser car il faut pouvoir acquérir à différentes focales (de 9.5 mm à 143 mm) l'image d'un fond uniforme. Toutefois, elles s'effectuent aisément en pointant l'objectif vers le ciel par temps de fort brouillard.

En conséquence, nous avons constaté expérimentalement que le profil radiométrique de la caméra ne change approximativement qu'à un facteur d'échelle près avec le zoom, la mise au point et le diaphragme.

A.4.6 Modèle expérimental simplifié

Suite à nos expérimentations, dans le cadre de notre application, il est possible de poser le modèle radiométrique simplifié suivant :

$$I_{\text{lue}}(u, v) = g(u, v)I_{\text{exacte}}(u, v) + \overline{I_{\text{obscurité}}(u, v)} + I_{\text{transmission}}(u, v) \quad (\text{A.2})$$

où $g(u, v) = g_{\text{sensibilité}}(u, v)g_{\text{objectif}}(u, v)$. Ce modèle reste approximativement valide avec une caméra équipée d'un objectif à focale variable.

La méthode de calibration simplifiée consiste donc à soustraire à l'image la valeur $\overline{I_{\text{obscurité}}(u, v)}$ puis à la diviser par l'image PLU de référence $g(u, v)$. Dans le cas d'une caméra couleur, ce calcul est réalisé indépendamment sur chaque canal.

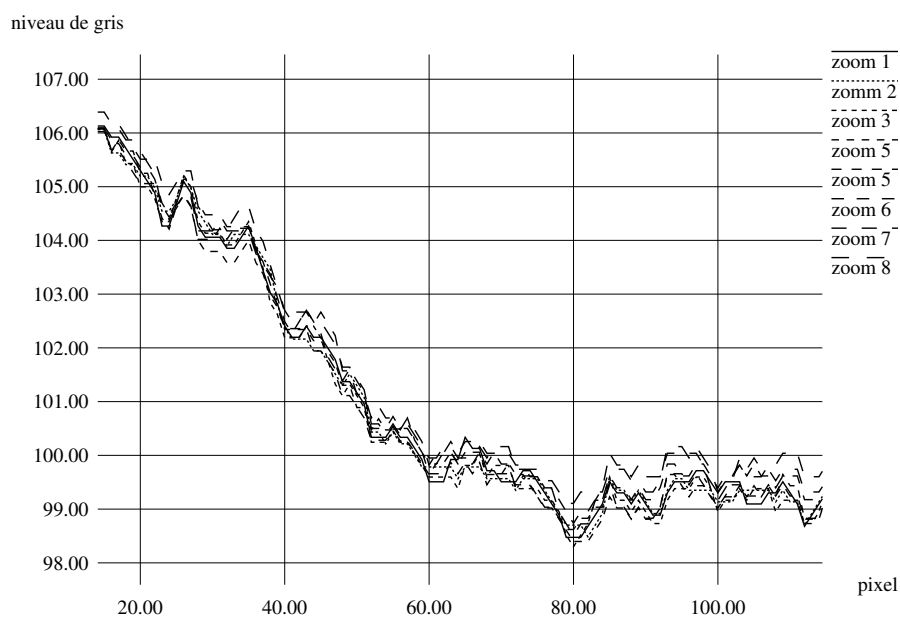


FIG. A.12 - *Variation de l'éclairement avec la focale.*

A.5 Exemples de corrections de la radiométrie

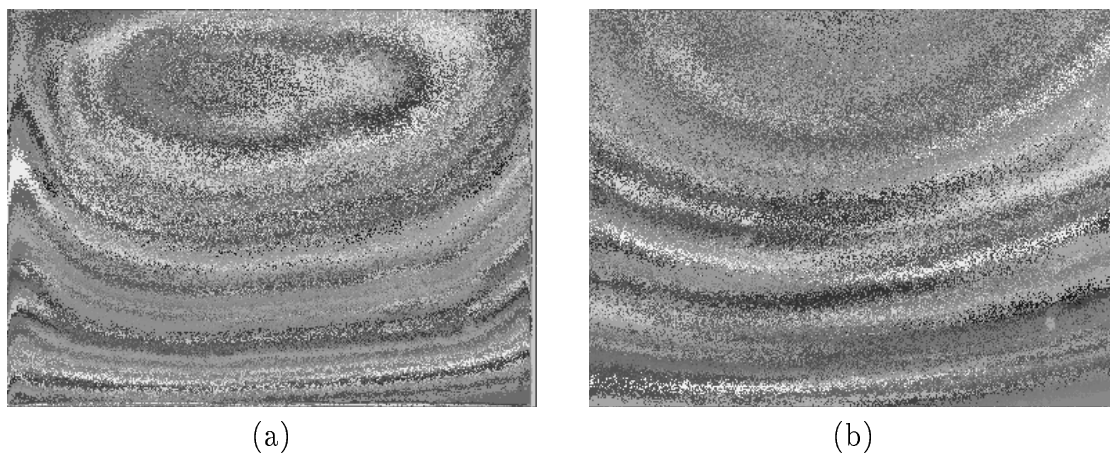


FIG. A.13 - *Vue avant (a) et après (b) correction radiométrique des gradients présents dans un ciel bleu près du zénith.*

Nous avons fait des essais de correction sur des images du ciel qui présentent l'avantage d'avoir un gradient lumineux reconnaissable. Par exemple, sur la figure en fausses couleurs A.13, la forme du gradient avant correction (a) paraît torturée. Cependant, après

correction radiométrique (b), la structure réelle du gradient apparaît. Cette image est prise avec un grand champ dans une direction proche du zénith.

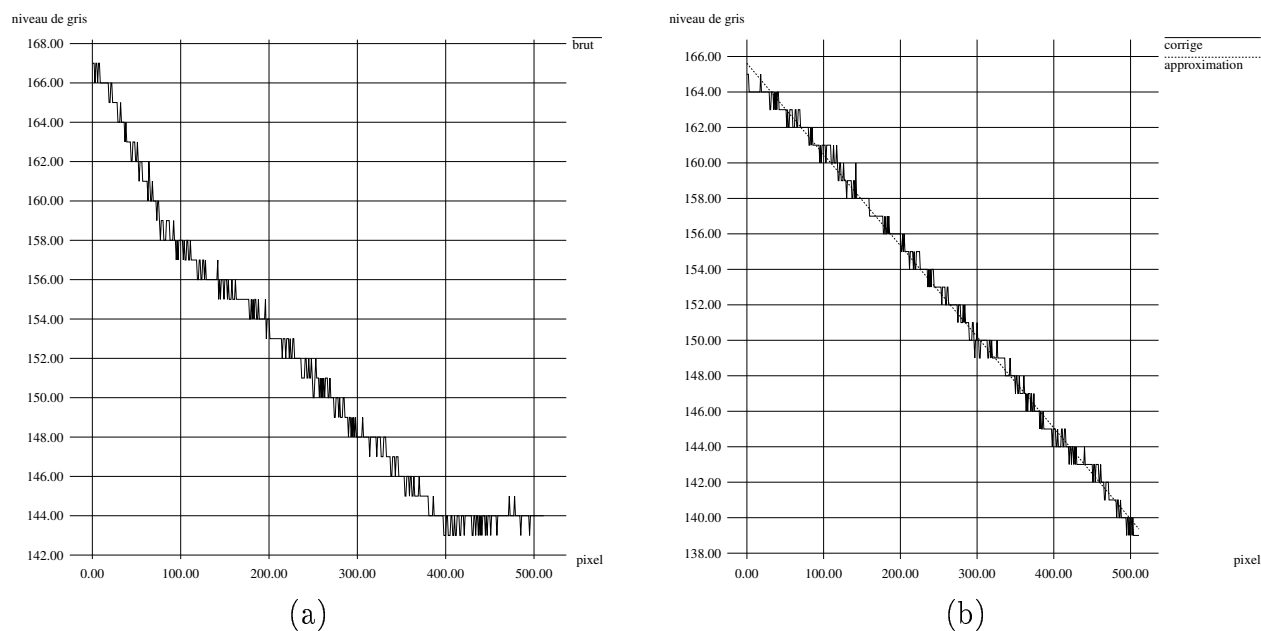


FIG. A.14 - Coupe d'une image du ciel avant (a) et après (b) correction radiométrique.

En rétrécissant le champ, le gradient du ciel apparaît comme linéaire avec une bonne approximation. Ainsi, on observe sur les coupes de la figure A.14 que la correction radiométrique rétablit la variation linéaire normale de l'intensité du ciel.

En rétablissant les gradients réels, la calibration radiométrique permet d'améliorer la segmentation par région des images et l'extraction des contours. Par exemple, sur un toit de matériau uniforme et lambertien, on retrouve grâce à la calibration radiométrique une distribution constante de l'intensité sur cette région (figure A.15).

L'image de la figure A.15 (a) est traitée avec un algorithme d'extraction des contours [Der87] avant et après correction. Les paramètres du filtre sont identiques. Néanmoins, des contours parasites sur le toit (figure A.16 (a)) de l'image avant correction sont absents sur la figure A.16 (b) qui est corrigée. On peut observer la même amélioration pour la segmentation en régions. Une méthode de segmentation par regroupement de régions [AMG93] a été utilisée sur la même image, avant et après correction radiométrique. On remarque sur le résultat (figure A.17) qu'une région parasite apparaît sur l'image avant correction.

Un exemple d'amélioration produite par la calibration radiométrique sur la reconstruction 3D de facettes par une méthode photométrique est aussi visible dans la figure 3.10, section 3.4.2.

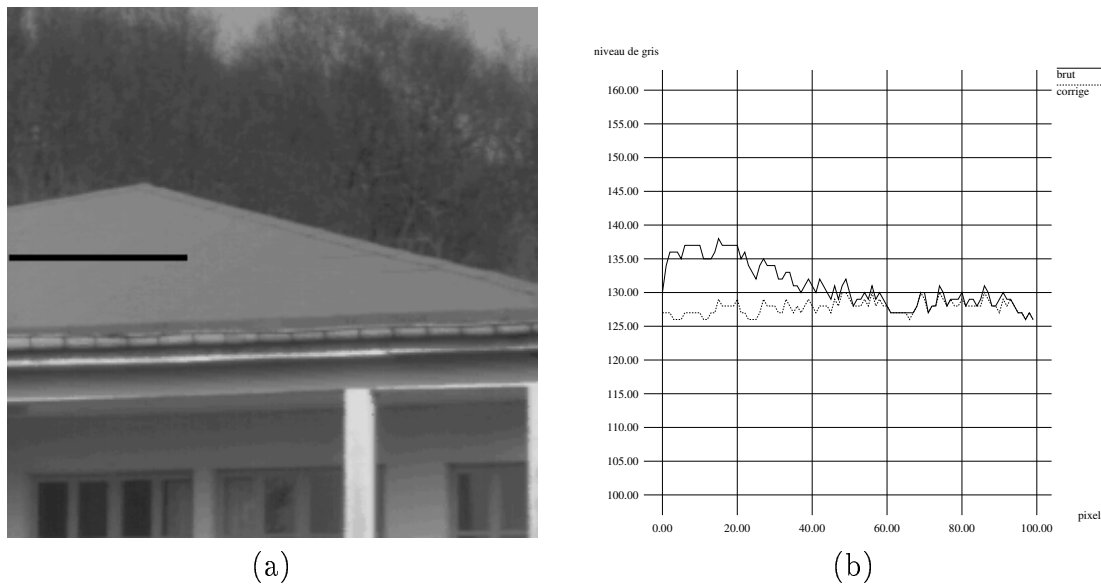


FIG. A.15 - (a) Détail de l'image d'un bâtiment. (b) coupe du toit de ce bâtiment avant et après correction radiométrique. Le lieu de la coupe est visible en noir sur le détail (a).

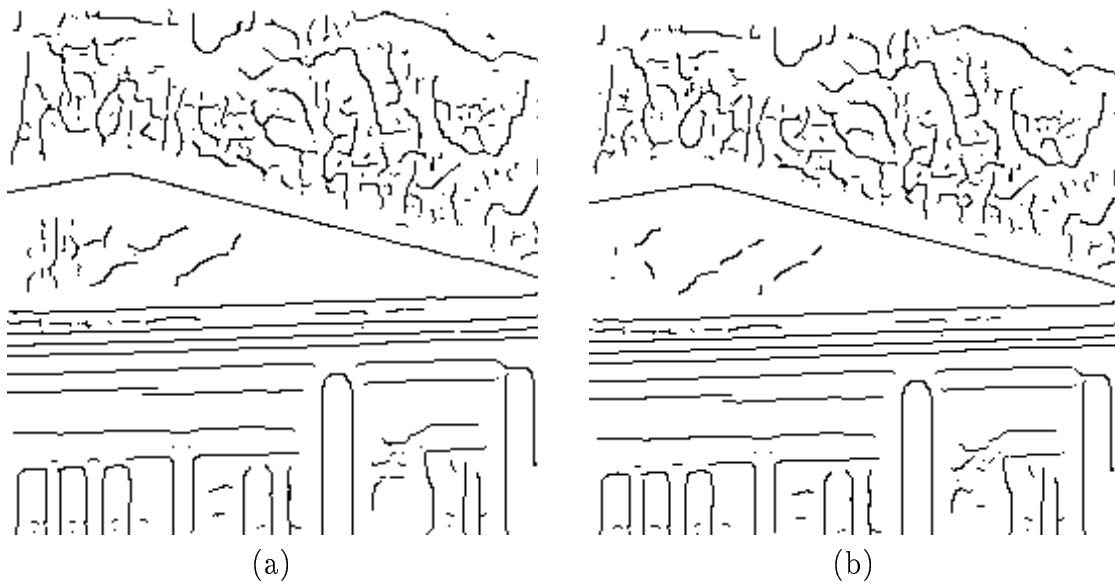


FIG. A.16 - Image des contours avant (a) et après correction radiométrique (b) du détails de la figure A.15(a). Des contours parasites sont éliminés grâce à la correction radiométrique.

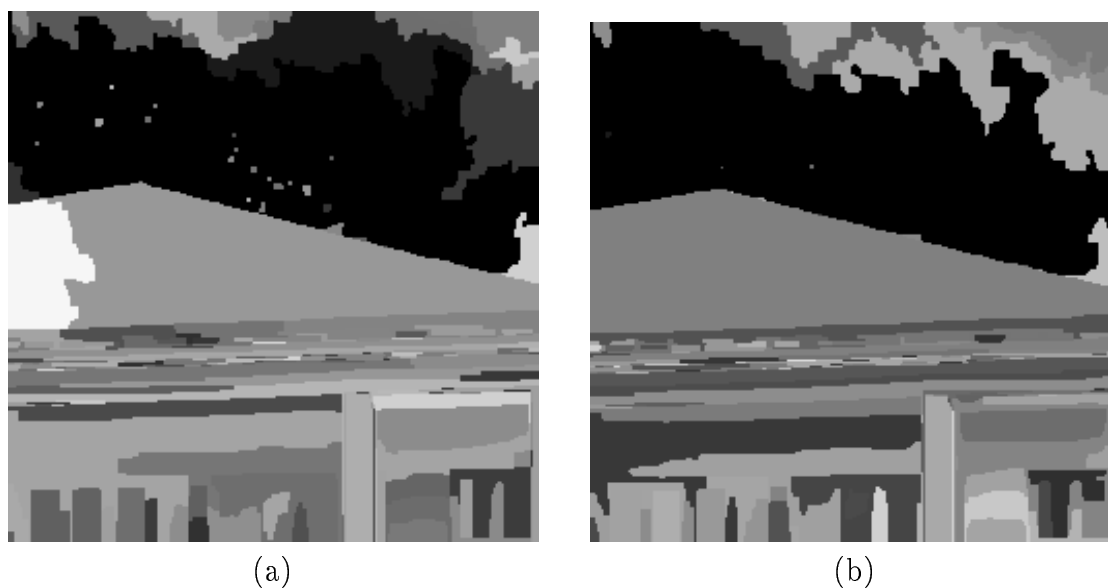


FIG. A.17 - *Image de segmentation avant (a) et après correction radiométrique (b) du détails de la figure A.15(a). Une région parasite apparaît sur l'image non corrigée.*

A.6 Conclusion

Des expérimentations avec un objectif à focale variable Canon J15×9.5B et Tessar 80mm ont permis d'observer que :

- la distribution des écarts de sensibilité radiométrique des cellules de la matrice CCD est difficile à modéliser de par sa forme,
- un changement du zoom, mise au point et diaphragme produit des variations faibles de l'illumination sur la rétine.

Ces expériences témoignent de la difficulté d'une modélisation théorique des corrections à apporter pour avoir une image bien calibrée photométriquement. Il est donc nécessaire de réaliser une calibration radiométrique à partir d'au moins une image de référence. Nous estimons que cette méthode permet de réduire l'écart de radiométrie entre les pixels de la rétine de 10% à environ 2% d'erreur. Les pixels les plus erronés dans l'image se trouvent sur le bord, dans une bande de 70 pixels pour le format vidéo 768×576.

Suite à nos expérimentations, dans le cadre de notre application, il est possible de poser le modèle radiométrique de la caméra simplifié donné par l'équation A.2. Ce modèle reste approximativement valide lors d'une variation de la focale.

Annexe B

Calibration pour la modélisation interactive

B.1 Introduction

Il existe de nombreux liens entre le problème de positionnement d'un objet par rapport à une caméra, celui du calcul du mouvement de la caméra et de sa calibration. Ces trois problèmes ne peuvent donc pas être considérés comme indépendants. Par exemple, la calibration résout le problème du positionnement de la caméra par rapport à un objet particulier (la mire) tout en déterminant les paramètres intrinsèques de celle-ci. Autre exemple : si l'on détermine le positionnement d'une caméra à différents instants par rapport au même objet, il est simple d'obtenir le déplacement de cette caméra. Ces problèmes font tous intervenir trois groupes de paramètres.

- Les *paramètres intrinsèques* qui modélisent les caractéristiques internes de la caméra. Pour le modèle sténopé, le plus simple, ces caractéristiques sont au nombre de 4 : la position du centre de l'image, et deux coefficients d'échantillonnage sur les deux axes de l'image.
- Les *paramètres extrinsèques* ; c'est-à-dire la position et l'orientation de la caméra par rapport à un référentiel fixe. Suivant le problème, ce référentiel peut être lié à un objet vu, ou à la caméra dans une autre position, ou enfin à une autre caméra.
- La *géométrie des objets* observés par la caméra.

Les paramètres extrinsèques sont à déterminer, dans les trois cas précisés, alors que les deux autres groupes de paramètres peuvent être supposés connus. Ainsi :

- Le problème de positionnement d'un objet par rapport à la caméra suppose que les paramètres intrinsèques soient connus, ainsi que tout ou partie de la géométrie de l'environnement perçu, suivant la méthode de résolution adoptée. Les méthodes à base de points ou problème perspective à n points [Har80, HCLL89, MM91, PHYT95] nécessitent au moins 3 points 3D connus. Des méthodes à base de lignes (au minimum 3

lignes 3D) ont aussi été proposées [DRLR88, LM89, CT90] ainsi que d'autres à base de primitives plus complexes comme les angles [Kan87], les trièdres [CT86] ou les ellipses [RZMF92].

- Les méthodes qui permettent de résoudre le problème du calcul du mouvement d'une caméra supposent que seuls les paramètres intrinsèques sont connus. Il existe des méthodes fondées sur les points, notamment la méthode des 8 points avec deux vues [LH81], ou basées sur les lignes [FLT87] ce qui nécessite trois vues et au moins 6 lignes.
- Pour la calibration d'une caméra, c'est la géométrie de l'objet qui est supposée connue. Généralement pour que cette connaissance soit précise, on utilise une mire. De nombreuses méthodes à base de points [Tos87] qui nécessite au moins six points, [Tsa86, LT88, Ma93a] ou de lignes [Vai90] ont été proposées avec différents modèles de caméra, suivant le type de distorsions modélisées.
- Enfin, l'auto-calibration [BMV93, MF92] ne nécessite théoriquement aucune connaissance a priori, si ce n'est une liste de points 2D mis en correspondance entre différentes vues. En contre partie, la reconstruction ne peut être réalisée que à une homographie de l'espace 3D près.

On remarque que des solutions à cette catégorie de problèmes ont d'abord été proposées avec des points puis étendues à des types de primitives géométriques de plus en plus complexes.

La modélisation automatique reste difficile pour les objets complexes, d'où la nécessité d'un outil interactif pour obtenir leur modèle. Mais, au cours du développement d'un système interactif de construction de modèles tridimensionnels, à partir de différentes vues d'une scène réelle, un problème pratique s'est posé à nous. Comment peut-on connaître les caractéristiques des différentes vues et obtenir la transformation qui fait passer du monde tridimensionnel à l'image? Nous avons donc besoin d'une méthode de calibration complète (paramètres intrinsèques et extrinsèques), avec des contraintes un peu particulières : la tâche du manipulateur doit être facilitée au maximum pour être accessible à un non spécialiste.

Le but de l'étude développée dans cette annexe n'est donc pas d'atteindre une grande précision dans la calibration. Cela demanderait d'avoir des informations redondantes, difficiles à obtenir sans avoir des mesures tridimensionnelles de la scène ou à partir d'images ne contenant pas de mire (voir chapitre 2). Nous avons alors cherché à retrouver la transformation qui fait passer du monde tridimensionnel à l'image, avec un nombre relativement faible d'informations 3D mesurées le plus simplement possible.

Il nous a été en pratique difficile de disposer de nombreuses mesures de points 3D, notamment lorsque les points sont à plusieurs mètres, voire quelques dizaines de mètres les uns des autres, ou lorsque l'on ne dispose pas de mire. Nous avons donc été amenés à étudier une calibration nécessitant des données plus faciles à mesurer. Pour ce faire, nous avons développé une méthode à base de primitives géométriques plus complexes que des points, comme des segments et des rectangles. Nous avons cherché à éviter d'avoir à connaître les positions, dans l'espace 3D, de ces primitives les unes par rapport aux autres.

Nous avons donc étudié une méthode de calibration à base de primitives qui ont un petit nombre de points (inférieur au nombre qui permettrait de faire directement une calibration à base de points, en considérant les sommets de cette primitive) comme les triangles, rectangles ou tétraèdres. Lorsque les *orientations* et les *tailles* des primitives de base sont connues dans le repère absolu, même si les *positions relatives* des primitives sont inconnues, il est possible d'effectuer la calibration complète des *paramètres intrinsèques et extrinsèques*. En choisissant des orientations particulières (des axes perpendiculaires), de simples mesures de longueurs suffisent, ce qui simplifie grandement la tâche de l'utilisateur du système.

Nous avons cherché à aboutir à des équations linéaires. Cette approche a été privilégiée, car ainsi nous avons pu réutiliser, dans un cadre plus général, l'algorithme proposé par Faugeras et Toscani [Tos87] qui présente comme avantage d'être analytique et permet la combinaison et la redondance des primitives. On peut alors réaliser de façon complète et simple la calibration pour chaque vue. Une fois la calibration effectuée, il est possible de calculer la position absolue de chaque primitive utilisée lors du processus de calibrage.

Après un rappel des équations de base liant un point de l'espace 3D à sa projection sur l'image (partie B.2), nous décrivons dans la partie suivante (partie B.3) une méthode de calibration à base de segments. A partir des équations obtenues, nous généralisons la méthode à des primitives plus complexes, constituées de segments jointifs (partie B.4). Nous rappelons ensuite l'algorithme utilisé pour résoudre le système d'équations obtenu avec les segments comme avec des primitives plus complexes (partie B.5), et discutons brièvement de l'existence d'une solution (partie B.6). Enfin, nous expliquons dans la partie B.7 comment, une fois la calibration réalisée, nous construisons le modèle de la scène. Des résultats de calibration et de construction sur des scènes réelles sont présentés à titre d'illustration.

B.2 Rappel des équations de base

Il s'agit de calculer la transformation qui fait passer de l'espace tridimensionnel à l'image. La caméra est modélisée comme un trou d'épingle (modèle sténopé). La transformation est

alors une projection perspective caractérisée par la matrice 3×4 , $\mathcal{L} = \begin{pmatrix} L_1 \\ L_2 \\ L_3 \end{pmatrix}$. L_1 , L_2 et

L_3 sont des vecteurs ligne de dimension quatre. On note Λ_i le vecteur constitué des trois premiers éléments de L_i . On pose ainsi $L_i = (\Lambda_i \quad l_{i,4})$.

Le point image (u, v) d'un point (x, y, z) est en coordonnées homogènes :

$$\begin{pmatrix} us \\ vs \\ s \end{pmatrix} = \mathcal{L} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \mathcal{L}\mathcal{X}$$

On en déduit deux équations qui lient un point de l'espace 3D, X , à son image, (u, v) , linéaires en fonction des éléments de la matrice perspective \mathcal{L} :

$$\left. \begin{aligned} (uL_3 - L_1)\mathcal{X} &= 0 \\ (vL_3 - L_2)\mathcal{X} &= 0 \end{aligned} \right\} \quad (\text{B.1})$$

En conséquence, si on dispose de la connaissance d'un nombre suffisant de points 3D X_i et des images associées (u_i, v_i) , il est possible d'estimer \mathcal{L} . Dans le modèle de caméra classique (avec une rétine rectangulaire) \mathcal{L} ne dépend que de dix paramètres : 6 paramètres extrinsèques (position et orientation de la caméra), 4 intrinsèques (le centre de l'image (u_0, v_0) et deux coefficients d'échantillonnage a_u et a_v). La matrice perspective doit donc vérifier deux contraintes déduites de la modélisation de la caméra. D'après [Tos87], en ne tenant compte que d'une seule de ces contraintes ($\|\Lambda_3\| = 1$), le problème a une solution analytique. Chaque point fournissant deux équations (B.1), il faut donc au minimum six points, dont les coordonnées dans l'espace 3D sont définies dans le repère absolu de la scène. Un nombre supérieur de points rendra l'estimation moins sensible aux erreurs des données. Cela est en pratique difficile à obtenir lorsque l'on ne dispose pas de mire. Nous avons donc été amenés à étudier une calibration avec des primitives géométriques plus complexes que des points de manière à ne pas avoir à connaître la position dans l'espace 3D de ces primitives les unes par rapport aux autres. Il n'est alors plus nécessaire de mesurer explicitement des coordonnées tridimensionnelles. Dans la partie suivante, la première primitive de base considérée sera le segment.

B.3 Segments

Un segment dans l'espace 3D est défini par ses deux extrémités X_0 et $X_0 + D$, ou \mathcal{X}_0 et $\mathcal{X}_0 + \mathcal{D}$ en coordonnées homogènes. Soit (u_1, v_1) et (u_2, v_2) , les images des deux sommets du segment. On a, d'après (B.1), les relations suivantes :

$$(u_1 L_3 - L_1) \mathcal{X}_0 = 0 \quad (\text{B.2})$$

$$(v_1 L_3 - L_2) \mathcal{X}_0 = 0 \quad (\text{B.3})$$

$$(u_2 L_3 - L_1) \mathcal{X}_0 + (u_2 L_3 - L_1) \mathcal{D} = 0 \quad (\text{B.4})$$

$$(v_2 L_3 - L_2) \mathcal{X}_0 + (v_2 L_3 - L_2) \mathcal{D} = 0 \quad (\text{B.5})$$

Combinons les équations de ce système sous une forme utile pour la suite :

$$(B.2) - (B.4) \quad (u_1 - u_2) L_3 \mathcal{X}_0 = (u_2 L_3 - L_1) \mathcal{D} \quad (\text{B.6})$$

$$(B.3) - (B.5) \quad (v_1 - v_2) L_3 \mathcal{X}_0 = (v_2 L_3 - L_2) \mathcal{D} \quad (\text{B.7})$$

$$u_2(B.2) - u_1(B.4) \quad (u_1 - u_2) L_1 \mathcal{X}_0 = u_1(u_1 L_3 - L_1) \mathcal{D} \quad (\text{B.8})$$

$$v_2(B.3) - v_1(B.5) \quad (v_1 - v_2) L_2 \mathcal{X}_0 = v_1(v_2 L_3 - L_2) \mathcal{D} \quad (\text{B.9})$$

On remarque alors que $(v_1 - v_2)(B.6) - (u_1 - u_2)(B.7)$ est une relation indépendante de la position 3D X_0 , du segment :

$$(v_1 - v_2) L_1 \mathcal{D} + (u_2 - u_1) L_2 \mathcal{D} + (u_1 v_2 - v_1 u_2) L_3 \mathcal{D} = 0$$

La dernière composante de \mathcal{D} , en coordonnées homogènes, est nulle : $\mathcal{D} = \begin{pmatrix} D \\ 0 \end{pmatrix}$. D caractérise le segment 3D en orientation et en longueur, indépendamment de sa position. En conséquence, avec les notations introduites $L_i = (\Lambda_i \quad l_{i,4})$, l'équation précédente s'écrit :

$$(v_1 - v_2)\Lambda_1 D + (u_2 - u_1)\Lambda_2 D + (u_1 v_2 - v_1 u_2)\Lambda_3 D = 0 \quad (\text{B.10})$$

On retrouve l'équation de base de [DRLR88]:

$$\mathcal{N} \cdot \mathcal{L}D = N \cdot \begin{pmatrix} \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \end{pmatrix} D = 0$$

Où $N = \begin{pmatrix} u_1 \\ v_1 \\ 1 \end{pmatrix} \wedge \begin{pmatrix} u_2 \\ v_2 \\ 1 \end{pmatrix}$ est la normale au plan d'interprétation du segment 3D, défini comme le plan passant par le point focal et le segment image [LM89].

En conséquence, *l'observation d'au moins 9 segments de position inconnue dans l'espace tridimensionnel, mais de direction connue dans un repère absolu, permet de calibrer la caméra (9 = 12 paramètres de la calibration - 3 paramètres de translation)*. La solution est toujours analytique, puisque (B.10) est linéaire en fonction des éléments de \mathcal{L} . Les éléments de la sous-matrice $\begin{pmatrix} \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \end{pmatrix}$ de \mathcal{L} sont la solution d'un système linéaire sous contrainte ($\|\Lambda_3\| = 1$).

Une fois ces derniers obtenus, les équations (B.7) (B.8) et (B.9) associées au segment j dans l'espace 3D, choisi comme référence ($X_0 = 0$) fournissent un système qui permet le calcul des $l_{i,4}$:

$$\begin{cases} (u_1^j - u_2^j) l_{1,4} = u_1^j (u_2^j \Lambda_3 - \Lambda_1) D \\ (v_1^j - v_2^j) l_{2,4} = v_1^j (v_2^j \Lambda_3 - \Lambda_2) D \\ (v_1^j - v_2^j) l_{3,4} = (v_2^j \Lambda_3 - \Lambda_2) D \end{cases} \quad (\text{B.11})$$

La matrice \mathcal{L} estimée, on peut alors extraire les paramètres de la calibration [Tos87]. De plus, on peut calculer la position de chaque segment 3D, dans le repère absolu, grâce aux équations (B.7) (B.8) et (B.9), par exemple.

Il est remarquable que dans l'équation (B.10) la longueur du segment 3D n'ait pas besoin d'être connue, seule sa direction compte. On n'a donc pas besoin des longueurs pour obtenir la partie $(\Lambda_1, \Lambda_2, \Lambda_3)$ de la matrice perspective \mathcal{L} . On établit ainsi le lien avec certaines approches à base de lignes, notamment le problème perspective à n lignes [LM89] et la calibration à base de lignes [Vai90]. Néanmoins, pour entièrement déterminer \mathcal{L} , il est nécessaire de connaître la longueur des segments 3D (B.11).

Afin d'estimer les paramètres intrinsèques de la caméra, ainsi que la position et l'orientation de la caméra, il est nécessaire, du fait de l'algorithme utilisé, de connaître l'orientation d'au moins 9 segments dans l'espace 3D, ainsi que leur longueur. La caméra est alors localisée par rapport au référentiel absolu utilisé pour repérer les orientations des différents segments. Son origine est un des sommets du segment j choisi comme référence. Le choix de primitives plus complexes, comme des assemblages de segments, permet de diminuer le nombre de primitives à utiliser au moment de la phase de calibration. Cela est abordé dans la partie suivante.

B.4 Segments jointifs

Lorsque deux segments dans l'espace ont un sommet en commun, on peut en déduire une relation supplémentaire. Supposons que chaque segment 3D i ($1 \leq i \leq n$) soit décrit par le bipoint $(X_0, X_0 + D_i)$ (tous ont en commun le premier sommet X_0). Pour le premier segment 3D ($i = 1$), on obtient les relations (B.6) (B.7) (B.8) et (B.9) où D est remplacé par D_1 . Pour un segment 3D différent, on a, en substituant (B.1) avec les équations précédentes :

$$(u_i - u_1)(u_2\Lambda_3 - \Lambda_1)D_1 + (u_1 - u_2)(u_i\Lambda_3 - \Lambda_1)D_i = 0 \quad (\text{B.12})$$

$$(v_i - v_1)(v_2\Lambda_3 - \Lambda_2)D_1 + (v_1 - v_2)(v_i\Lambda_3 - \Lambda_2)D_i = 0 \quad (\text{B.13})$$

Un ensemble de $(n + 1)$ points dans l'espace, décrit comme un ensemble de n segments 3D présentant un sommet en commun, conduit à $(2n - 1)$ équations. Les équations pour le cas du rectangle seront détaillées dans la partie 7.2. En conséquence, *avec des primitives tridimensionnelles à $n = 2$ (triangle), 3 (rectangle, tétraèdre), 4 et 5 segments, il faudra utiliser respectivement 3, 2, 2 et 1 primitives au minimum, afin d'obtenir un système d'équations cohérent.* Dans le dernier cas et pour des ensembles de segments plus importants, on retrouve la situation classique d'un ensemble de points 3D (au moins 6) connus dans un même repère. Pour chaque cas, nous avons obtenu des équations linéaires en fonction des éléments de la matrice perspective \mathcal{L} (B.10) (B.12) (B.13). On peut donc combiner les diverses primitives tridimensionnelles et utiliser dans une même phase de calibration des segments, des triangles, des rectangles et des tétraèdres, entre autres. Néanmoins, le choix des positions relatives des primitives tridimensionnelles doit être mené avec précaution, car certaines configurations spatiales conduisent à un système d'équations dégénéré. L'algorithme utilisé n'est alors plus applicable.

B.5 Algorithme de résolution

Nous rappelons dans cette partie, l'algorithme de résolution utilisé. Pour plus de détails se reporter à [Tos87], où l'algorithme est employé pour réaliser la calibration avec des points. Elle permet de résoudre un système d'équations linéaires au sens des moindres carrés, sous une contrainte de norme.

De par l'approche, le problème s'explique comme un système linéaire en $\begin{pmatrix} \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \end{pmatrix}$ à résoudre au sens des moindres carrés, sous contrainte :

$$\min_L \|AL\|$$

avec

$$L = \begin{pmatrix} \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \end{pmatrix}$$

et la contrainte

$$\|\Lambda_3\| = 1$$

où A est constituée de lignes formées à partir des équations linéaires du type de (B.10), (B.12) ou (B.13).

En utilisant les multiplicateurs de Lagrange, on déduit la solution analytique suivante :

$$L_{min} = \begin{pmatrix} -B_{6 \times 6}^{-1} C_{6 \times 3} \Lambda_{3min} \\ \Lambda_{3min} \end{pmatrix} \quad (\text{B.14})$$

où Λ_{3min} est le vecteur propre associé à la valeur propre minimum de la matrice $D_{3 \times 3} - C_{3 \times 6}^t B_{6 \times 6}^{-1} C_{3 \times 6}$, avec

$$A^t A = \begin{pmatrix} B_{6 \times 6} & C_{3 \times 6} \\ C_{3 \times 6}^t & D_{3 \times 3} \end{pmatrix}$$

Cette méthode de résolution présente deux avantages :

- la résolution est analytique, à partir d'équations linéaires,
- elle peut traiter les cas où l'information est redondante, ce qui permet d'améliorer la précision du résultat en utilisant plus de primitives que le nombre minimum nécessaire à l'algorithme.

En revanche, comme pour des raisons de simplicité on ne manipule que des équations linéaires, le nombre minimal d'informations pour l'algorithme est un peu supérieur au nombre minimal théorique. Par exemple, utilisé avec des segments, cet algorithme requiert 9 segments, alors que théoriquement, 7 ou 8 segments peuvent suffire, suivant les contraintes que l'on impose sur la matrice de transformation perspective de la caméra.

B.6 Existence d'une solution

Il existe des configurations spatiales qui rendent la résolution impossible par la méthode présentée. Il serait donc intéressant de les caractériser afin de les éviter. Ainsi il est bien connu que la calibration avec des points nécessite une distribution de points 3D non coplanaires. De même, pour la calibration à base de primitives tridimensionnelles, ces dernières, ramenées à une même origine ne doivent pas être coplanaires. Plus généralement, toute configuration qui fait de B une matrice non inversible provoque la dégénérescence du système. Malheureusement, ces configurations sont difficiles à caractériser ou à énumérer de façon exhaustive, et cela reste en soi un sujet à étudier.

Nous avons toutefois observé un autre type de configuration qui produit la dégénérescence du système, ici liée à la méthode de résolution. En particulier, cette situation apparaît lorsque les primitives à base de segments 3D, ramenés à une même origine, sont incluses dans un trièdre orthogonal. De façon analogue, la méthode fondée sur les points avec la contrainte $\|\Lambda_{3min}\| = 1$ dégénère avec l'algorithme de résolution de la partie précédente, si les points sont positionnés sur un trièdre. Ce type de dégénérescence est la conséquence du fait qu'une des contraintes sur la transformation perspective n'est pas utilisée : les éléments de la matrice perspective \mathcal{L} ne sont plus contraints. Pour pallier à cela, on peut employer une contrainte, posée a priori, qui agit sur l'ensemble de la matrice, par exemple $\|\mathcal{L}\| = 1$ comme dans la section 1.3.7.

Une fois la calibration réalisée sur plusieurs vues, à partir de quelques objets de la scène, il est possible de construire interactivement le modèle géométrique de l'ensemble de la scène par superposition sur chaque image.

B.7 Application à la construction interactive

B.7.1 Construction interactive

Il s'agit de construire de manière interactive un objet d'une scène à partir de différentes images de cette scène [JP92]. Après la détermination de la matrice de transformation perspective de la caméra par calibration, l'utilisation d'un modeleur comme outil de modélisation géométrique, à travers une vue identique à celle de la caméra, nous permet de superposer la projection de l'objet tridimensionnel à construire sur son image. Le modeleur est ainsi utilisé comme un calque sur l'image qui est placée dans une vue dite caméra.

Nous disposons, comme capteur d'images, d'une caméra DTC couleurs Sony. Nous utilisons comme modeleur géométrique, Action 3D, créé en collaboration avec la société Sogitec. Pour le moment, nous nous sommes intéressés à une scène de bureau, qui est un univers polyédrique. Ce modeleur permet, pour notre application, de créer des modèles en fil de fer avec des primitives à déformer (cube, sphère, cônes par exemple). On peut faire subir trois types de transformations à une primitive: translation, rotation et changement d'échelle sur les trois axes: x , y , z . L'utilisateur doit donc gérer interactivement 9 degrés de liberté. Pour éviter des erreurs de construction due à la profondeur (voir figure B.12), chaque nouvelle primitive insérée dans la scène possède au moins un point de contact avec un objet correctement positionné. En pratique, les objets ont au moins une face commune (une fonction a été implémentée afin de satisfaire cette contrainte), ce qui réduit le nombre de degrés de liberté (6: 2 pour la translation, 1 pour la rotation et 3 pour les facteurs d'échelle) et permet la construction incrémentale de ce qui est perçu par le capteur.

La représentation des modèles géométriques dans le modeleur est une représentation par bords. La structure est hiérarchique: les volumes sont définis comme des polyèdres, définis eux-mêmes par un ensemble de polygones orientés fermant un volume. Il est à noter que le modeleur peut créer des primitives plus complexes, ce qui nous permettrait de construire une scène plus complexe qu'une scène polyédrique.

La construction étant interactive, l'opérateur peut très facilement visualiser et donc vérifier que les primitives qu'il construit sont définies avec cohérence (pas d'interpénétration ou pas d'objet flottant par exemple). De plus, grâce aux fonctionnalités du modeleur tels que le zoom ou la multiplicité des vues, la construction interactive est rendue plus facile.

La précision de la modélisation est également laissée au choix de l'utilisateur, en fonction de l'application qu'il veut en faire. En effet, dans le cadre de la robotique, pour déplacer un robot qui doit éviter les obstacles, la connaissance du volume englobant et de la position de l'objet est suffisante. Il n'est pas nécessaire d'avoir plus de précision. Par contre, pour d'autres applications, comme l'architecture, il serait nécessaire, d'avoir une description plus détaillée.

B.7.2 Résultats de calibration

Nous avons testé la calibration avec les primitives suivantes : des points, des segments, des rectangles et une combinaison de segments et de rectangles. Dans un premier temps, ces tests ont été réalisés sur le minimum possible de primitives : 6 pour les points, 9 pour les segments, 2 pour les rectangles et pour le cas mixte : 1 rectangle et 4 segments. Nous avons pris soin d'éviter les configurations qui posent problème : les points tridimensionnels ne doivent pas être coplanaires et ne pas reposer sur les axes d'un trièdre orthogonal.

fichier 1	fichier 2
(582,274)	(583,273)
(513,301)	(512,301)
(183,384)	(182,385)
(184,268)	(181,267)
(502,451)	(501,450)
(509,306)	(508,306)

TAB. B.1 - *Points 2D exprimés en pixels pour la calibration avec 2 rectangles : sur chaque point une différence de quelques pixels cf. figures B.6 et B.7 et tableaux B.2 et B.3.*

Les coordonnées dans le repère de l'image sont obtenues en cliquant les points à l'écran tandis qu'il faut mesurer leur correspondant 3D dans le repère de la scène. La précision des mesures 2D est donc au pixel près, ce qui est assez faible et fait varier énormément les valeurs des paramètres intrinsèques obtenues (cf. tableaux B.1, B.2 et B.3, et figures B.6 et B.7).

Pour les rectangles, de la même manière que nous obtenons l'équation (B.10) pour un segment et à partir des équations (B.12) et (B.13), nous avons le système d'équations suivant :

$$\left\{ \begin{array}{l} \Lambda_1 D_1(v_1 - v_2) + \Lambda_2 D_1(u_2 - u_1) + \Lambda_3 D_1(u_1 v_2 - v_1 u_2) = 0 \quad (B.10) \\ \Lambda_1 [D_1(u_1 - u_3) + D_2(u_2 - u_1)] + \Lambda_3 [D_1(u_3 - u_1)u_2 + D_2(u_1 - u_2)u_3] = 0 \quad (B.12, i = 3) \\ \Lambda_1 D_1(v_1 - v_3) + \Lambda_2 D_2(u_2 - u_1) + \Lambda_3 [D_1(v_3 - v_1)u_2 + D_2(u_1 - u_2)v_3] = 0 \quad (B.13, i = 3) \\ \Lambda_1 [D_1(u_2 - u_4) + D_2(u_2 - u_1)] + \Lambda_3 [D_1(u_4 - u_2)u_1 + D_2(u_1 - u_2)u_4] = 0 \quad (B.12, i = 4) \\ \Lambda_1 D_1(v_1 - v_4) + \Lambda_2 (D_1 + D_2)(u_2 - u_1) + \Lambda_3 [D_1(u_1 v_4 - v_1 u_2) + D_2(u_1 - u_2)v_4] = 0 \quad (B.13, i = 4) \end{array} \right. \quad (B.15)$$

où D_1 et D_2 représentent les vecteurs largeur et longueur du rectangle considéré.

La calibration avec pour base le rectangle nécessite au moins 2 primitives non coplanaires, et dans nos tests, pour des raisons de simplification de mesures, les deux rectangles considérés forment un trièdre orthogonal. Le système d'équations de la calibration avec pour primitives le segment et le rectangle est composé de l'équation (B.10) (segment) et du système d'équations (B.15) (rectangle).

Pour pouvoir comparer les résultats de calibration des différentes primitives nous avons essayé de prendre autant que possible, les mêmes points sur une image. Nous avons ainsi choisi les parties visibles du bureau comme primitives (coins, arêtes, plateau etc. cf. figure B.2).

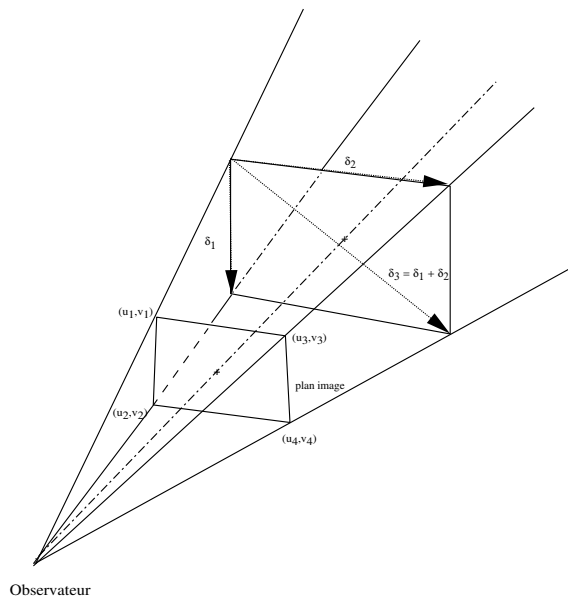


FIG. B.1 - Les données d'un rectangle nécessaires à la calibration

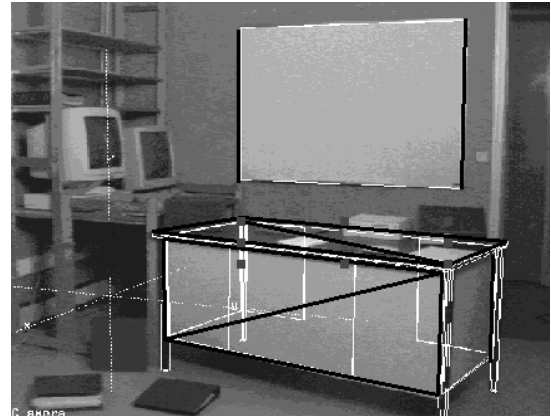


FIG. B.2 - En blanc le modèle reconstruit, en noir les primitives utilisées lors des différentes calibrations. Les primitives (points, segments et rectangles) sont essentiellement les parties visibles du bureau et les 2 arêtes verticales du tableau utilisées pour les calibrations mixte et avec segments).

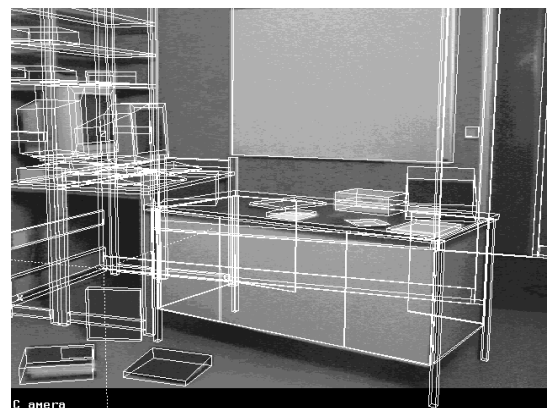


FIG. B.3 - Caméra calibrée avec 6 points. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.

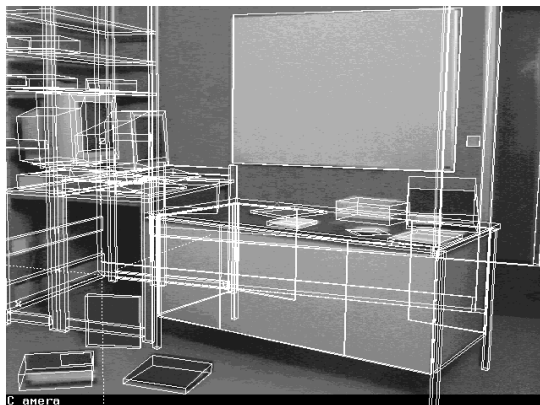


FIG. B.4 - *Caméra calibrée avec 8 points. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.*

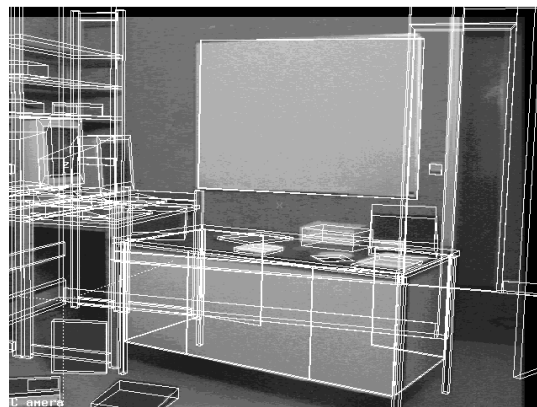


FIG. B.6 - *Caméra calibrée avec 2 rectangles : le plateau et la face avant du bureau (Premier fichier 2D). Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.*

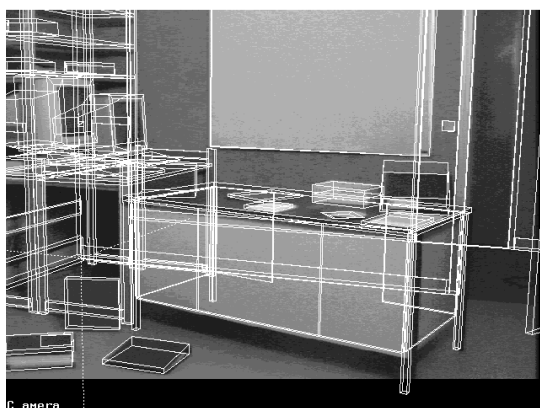


FIG. B.5 - *Caméra calibrée avec 9 segments. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.*

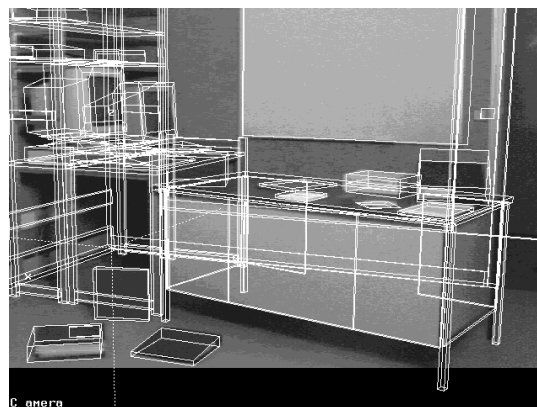


FIG. B.7 - *Caméra calibrée avec 2 rectangles avec un deuxième fichier 2D. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.*

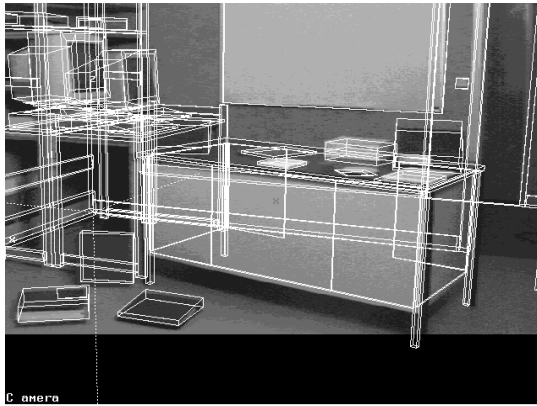


FIG. B.8 - *Caméra calibrée avec 1 rectangle (plateau) et 4 segments (appartenant à la face avant du bureau). Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.*

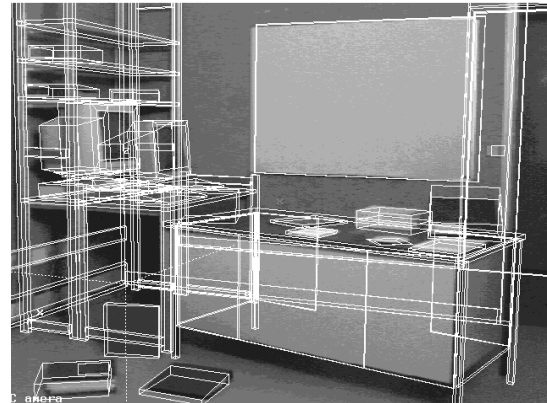


FIG. B.10 - *Caméra calibrée avec 1 rectangle et 5 segments dont 4 sont verticaux. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.*

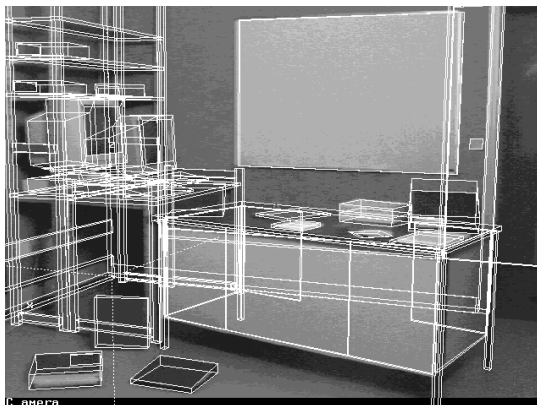


FIG. B.9 - *Caméra calibrée avec 1 rectangle (plateau) et 4 segments (dont 2 sont les arêtes verticales du tableau). Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.*

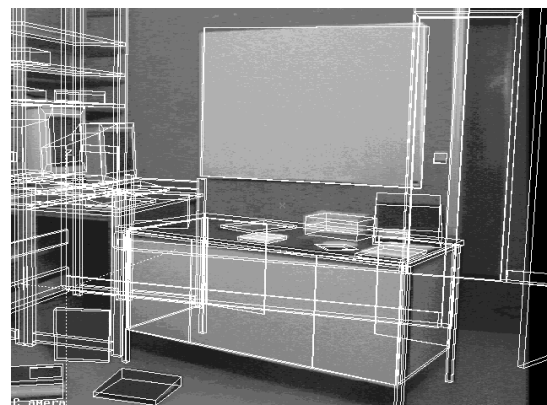


FIG. B.11 - *Caméra calibrée avec 2 rectangles et 1 segment vertical. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.*

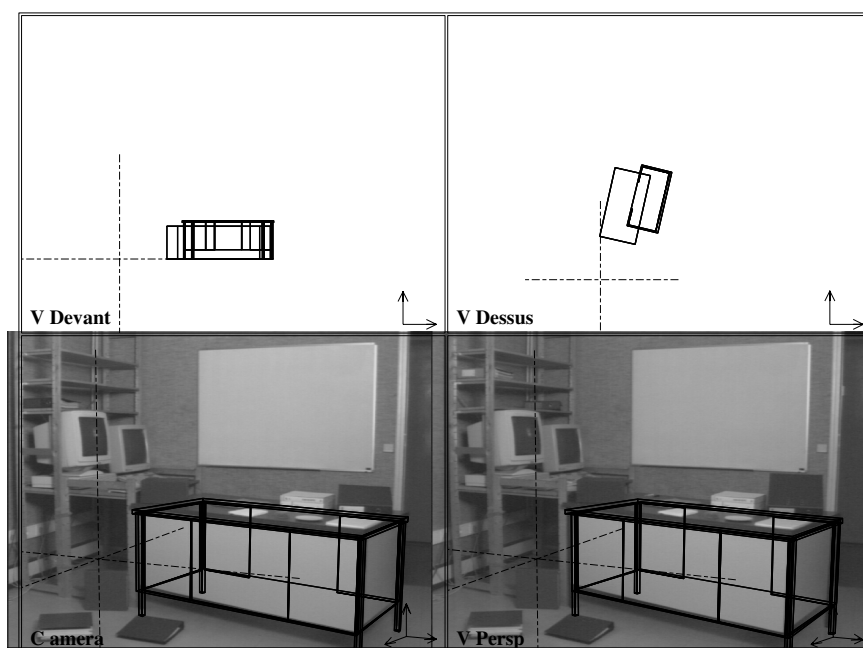


FIG. B.12 - *Phase de construction: erreur de construction en profondeur sur le corps du bureau: sa position et ses dimensions paraissent correctes sur les vue caméra et perspective alors qu'elles sont fausses.*

Les résultats obtenus avec différentes primitives sont relativement similaires (cf. tableaux B.2 et B.3). Le choix d'une primitive à la place d'une autre dépend uniquement des informations que l'on peut avoir de la primitive: ses points-image, ses mesures et son orientation tridimensionnelle. En effet, il est plus simple de mesurer la longueur d'un segment de direction particulière que de mesurer des points situés sur des objets dont les repères locaux sont différents. Cependant, il faut aussi s'assurer de l'existence de leur correspondant sur l'image. De plus, nous avons observé le fait suivant: si les primitives choisies ne sont pas bien réparties sur l'image, la méthode ne donne pas toujours de résultats. L'estimation est meilleure avec une bonne répartition sur l'image, des primitives (cf. figures B.8 et B.9). Par ailleurs, si nous augmentons leur nombre, les résultats s'améliorent (cf. figures B.4 et B.10). En repérant dans l'image deux rectangles formant un trièdre orthogonal dans l'espace 3D, nous choisirons de préférence la calibration mixte car elle offre la possibilité de rajouter des segments de même direction, par exemple verticaux, améliorant ainsi l'estimation des paramètres de la caméra.

B.8 Conclusion

Nous avons proposé une méthode de calibration adaptée au besoins d'un système interactif de construction de modèles tridimensionnels, à partir de différentes vues d'une scène réelle complexe.

Cette méthode ne manipule que des équations linéaires. L'algorithme utilisé permet d'obtenir la solution d'un système linéaire, au sens des moindres carrés, sous une contrainte de

paramètres	référence	6 points (figure B.3)	9 segments (figure B.5)	2 rectangles (fig.B.6, tab.B.1 Fichier 1)	1 rectangle 4 segments (fig.B.9)
(u_0, v_0) (pix)	(318,238)	(325,266)	(344,278)	(363,225)	(312,256)
(a_u, a_v)	(871,874)	(872,883)	(900,903)	(950,939)	(886, 877)
t_x (dm)	49.34	48.35	49.68	51.57	48.83
t_y (dm)	31.25	331.08	31.58	31.72	31.37
t_z (dm)	12.8	12.56	12.92	13.27	12.86
r_x (degrés)	-0.64	-0.56	-1.15	-1.57	-0.59
r_y (degrés)	6.31	7.99	8.93	5.7	7.36
r_z (degrés)	-160.55	-160.71	-161.94	-163.50	-156.71

TAB. B.2 - Paramètres de la caméra obtenus par calibration avec le nombre minimum de primitives: centre de l'image (u_0, v_0) , facteurs d'échelles (a_u, a_v) , translations (t_x, t_y, t_z) , rotation (r_x, r_y, r_z) .

param.	référence	8 points (fig. B.3)	2 rectangles (figure B.7) (tab.B.1 Fich.2)	1 rectangle 4 segments (fig. B.8)	1 rectangle 5 segments (fig. B.10)	2 rectangles 1 segment (fig. B.9)
(u_0, v_0)	(318,238)	(320,254)	(311,288)	(331,322)	(297,240)	(362,239)
(a_u, a_v)	(871,874)	(877,889)	(881,898)	(924, 909)	(850,856)	(949,940)
t_x (dm)	49.34	48.74	48.68	50.40	47.48	51.49
t_y (dm)	31.25	31.25	31.27	31.54	31.06	31.67
t_z (dm)	12.8	12.8	12.69	13.14	12.5	13.27
r_x (deg.)	-0.64	-0.71	-0.92	-1.47	-0.82	-1.72
r_y (deg.)	6.31	7.36	9.48	11.80	6.32	6.55
r_z (deg.)	-160.55	-160.55	-159.72	-161.42	-158.53	-163.49

TAB. B.3 - Paramètres de la caméra obtenus par calibration sur d'autres données.

norme. Cette approche a l'avantage de permettre l'utilisation combinée de différentes primitives polyédriques pour faire la calibration. Pour l'ensemble des primitives, il faut connaître la taille et l'orientation dans le même repère, des segments dont elles sont constituées. En revanche, il n'est pas nécessaire d'avoir les positions relatives de ces primitives. Dans la pratique, les orientations des segments sont choisies perpendiculaires les unes par rapport aux autres, ainsi seule leur longueur est à mesurer.

Nous avons observé que certaines configurations en orientation des primitives tridimensionnelles, conduisent à un système d'équations dégénéré. Ces configurations sont difficiles à énumérer de façon exhaustive, cela reste donc un sujet d'étude ouvert.

Pour avoir une modélisation correcte des objets, un minimum de connaissance concernant les dimensions des objets est nécessaire pour corriger les incertitudes des mesures 2D. En effet, les résultats sont essentiellement évalués par la qualité visuelle de la superposition des primitives utilisées sur l'image. Mais, cette évaluation reste grossière et très subjective. Si la calibration n'est pas correcte parce que l'on a pas utilisé suffisamment d'information géométrique, la modélisation ne peut être fiable.

Dans cette étude, nous nous sommes surtout intéressés à la modélisation géométrique, mais il est également tout à fait possible, de définir des informations autres propres à l'objet, telles sa couleur ou sa texture. En particulier, un système interactif qui permet de construire des modèles polyédriques sur lequel sont plaqués la texture provenant des images est un développement possible intéressant de ce travail.

Annexe C

Recalage 3D/2D fin

C.1 Introduction

La méthode présentée dans cette annexe consiste à raffiner l'estimation de la position du modèle 3D d'un objet directement sur la paire d'images et non pas sur la reconstruction uniquement comme dans le chapitre 4.

Avant de faire le recalage 3D/2D, les étapes précédentes de l'analyse ont extrait un grand nombre d'informations des images. Ainsi, la calibration corrige les images des défauts de la caméra pour se ramener à un modèle sténopé parfait et calcule le déplacement entre les deux points de vue. Puis, la segmentation extrait des régions qui correspondent à l'image d'une face plane de l'espace 3D. La reconstruction de ces faces est réalisée à partir des appariements de régions de l'image gauche et droite. Enfin, le modèle est recalé avec la reconstruction. Alors, si sa projection et son image ne sont pas parfaitement superposées, une étape de recalage fin est nécessaire.

On dispose donc, après ces traitements, d'un nombre important d'informations 3D et 2D en plus des images originales. On suppose l'objet à recalé connu à un déplacement rigide près, et que son modèle géométrique est composé de facettes planes. Les modèles de référence peuvent être construits, à partir de quelques vues de la scène, grâce à un outil interactif : un modeleur géométrique (voir Annexe B), ou tout autre système produisant des modélisations par faces.

L'accumulation des défauts des étapes antérieures de l'analyse des paires stéréoscopiques (calibration, segmentation, reconstruction, recalage) ne permet parfois d'avoir qu'une estimation imprécise de la position du modèle d'un objet (image (a) de la figure C.5). Il faut donc raffiner l'analyse par une étape qui recalé les facettes du modèle sur les images segmentées du couple stéréoscopique.

À partir d'une position approchée, nous proposons de recalé le modèle 3D par rapport aux images de manière itérative. Cet algorithme consiste à reprojeter (figure C.1) le modèle dans sa position courante, puis à remettre à jour l'appariement entre les régions et les faces par comparaison de la reprojektion et de la segmentation originale par un algorithme de recouvrement de régions (section C.2), et enfin à recalé le modèle un peu plus précisément (section C.3). La méthode proposée permet donc d'incrémenter pas à pas le nombre de faces 3D appariées avec des paires de régions gauche-droite. Par son approche, cette méthode est

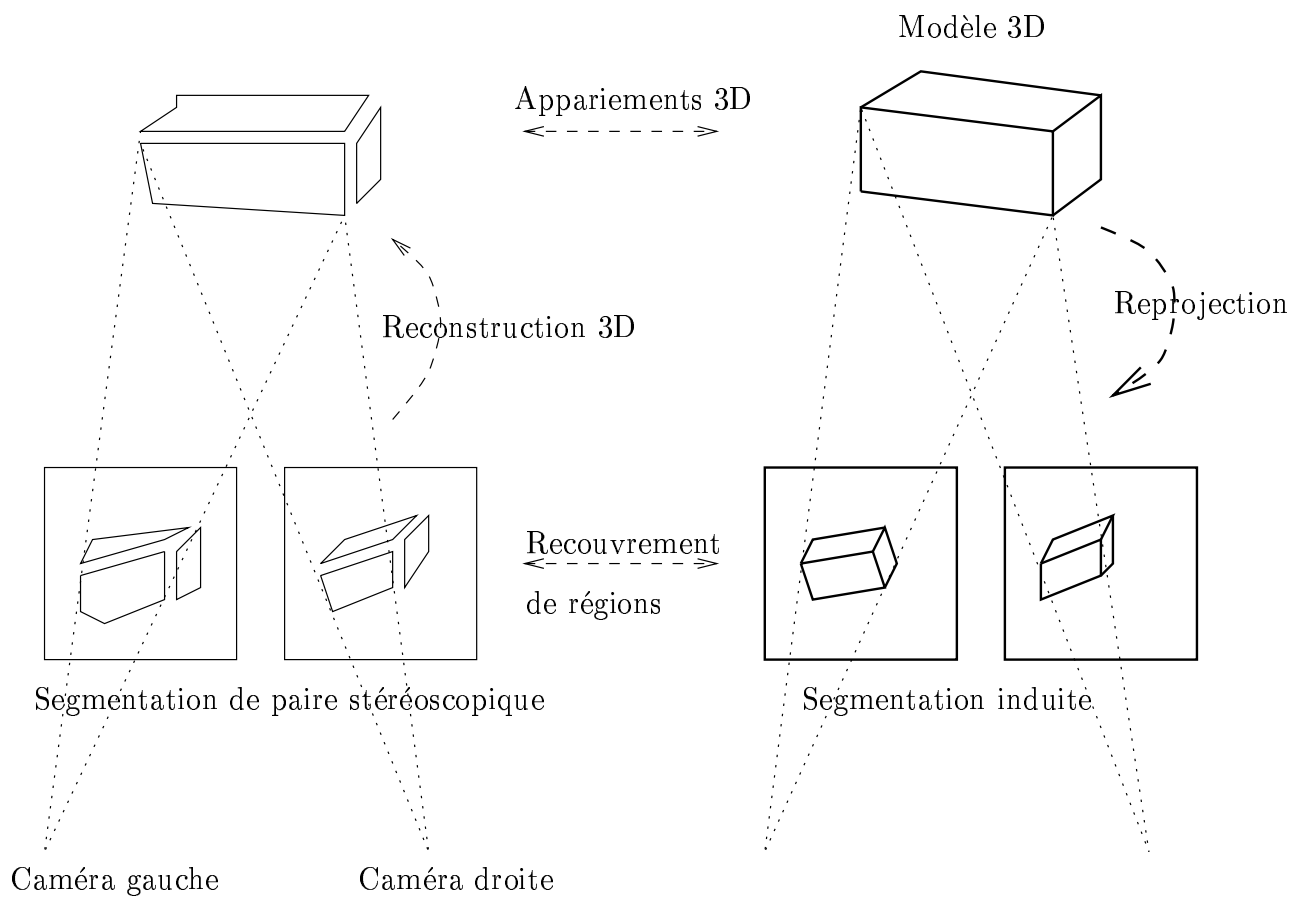


FIG. C.1 - Liens entre les objets 3D, les paires d'images et les algorithmes manipulés.

identique à celle développée dans la méthode ICP (voir le chapitre 5), sauf que le recalage est réalisé entre des données 3D et 2D. Une approche similaire à été proposée [Fel95] dans le cadre des applications médicales.

C.2 Mise en correspondance 3D/2D par reprojection

La première étape consiste à reprojeter l'objet dans sa position courante. Le modèle peut être reprojété en fil de fer sur l'image, sans ses parties cachées, comme dans la figure C.5. Puisque nous disposons de la segmentation des images, il est plus intéressant d'avoir la reprojektion en régions, où chaque région est associée à une surface 3D visible, comme dans l'image (b) de la figure C.2. En effet, elle peut alors être comparée avec l'image de segmentation (figure C.2).

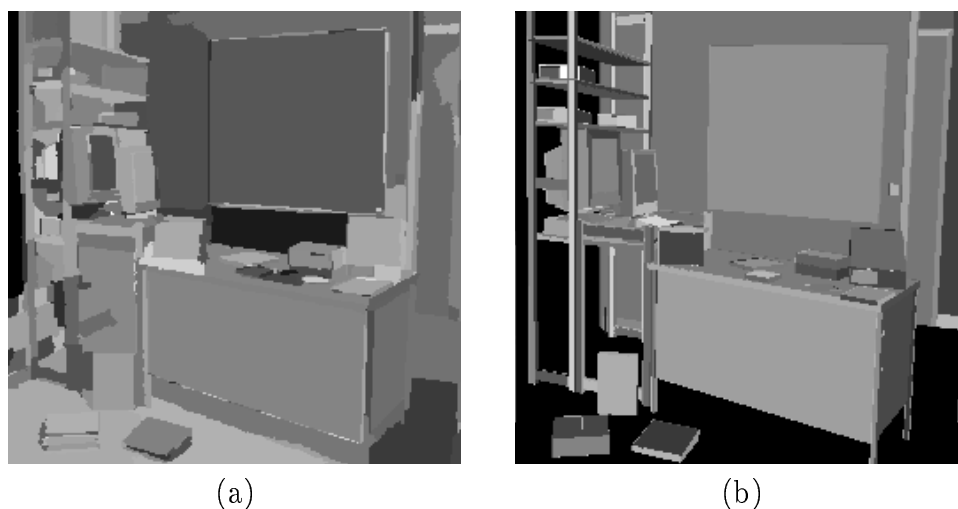


FIG. C.2 - Image de segmentation originale (a) et segmentation induite par le modèle (b).

La comparaison entre la segmentation induite par le modèle 3D et la segmentation obtenue à partir de la paire originale est réalisée avec un algorithme de recouvrement de régions. Celui-ci permet d'apparier les régions de la segmentation originale et induite, elles-mêmes reliées à des facettes 3D. Les paires de régions appariées ont des surfaces qui se recouvrent à plus de 80%. Cette contrainte forte permet de sélectionner uniquement des appariements justes. À la suite de la reprojection, un ensemble, plus riche, de facettes du modèle 3D de référence appariés avec des régions des segmentations droite et gauche est donc établi.

C.3 Estimation d'un meilleur recalage

C.3.1 À partir de sommets sur les contours

Le calcul du petit déplacement rigide, qui recale plus précisément le modèle parfait avec les deux régions de l'image, peut être réalisé à partir des sommets sur les contours, car il est facile de relier un point 3D à son image en fonction de ce déplacement. La distance entre la projection d'un des coins de la facette 3D du modèle et le coin de la région appariée à cette face, par le processus décrit précédemment, est minimisée en fonction du déplacement du modèle. Les sommets d'une région sont déterminés comme les points de l'approximation polygonale du contour qui sont proches de la projection d'un des coins de la face 3D (figure C.3). Cette condition est stricte pour éviter les appariements aberrants.

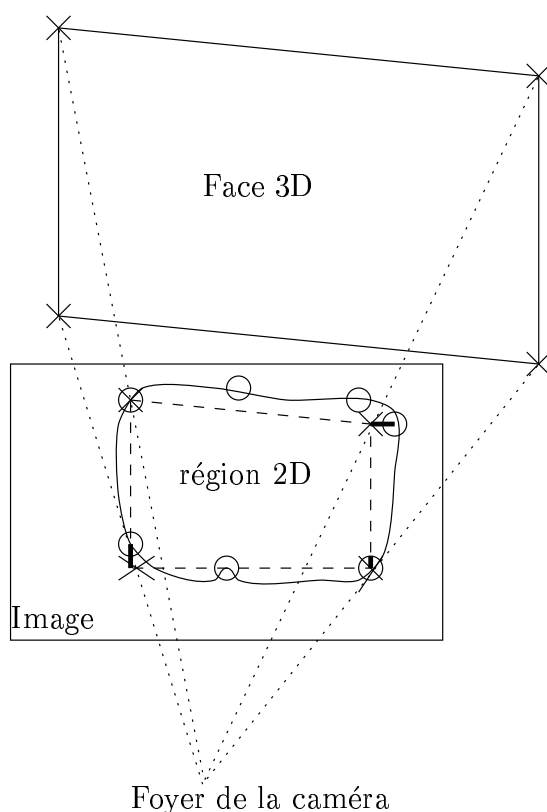


FIG. C.3 - Sélection des points sur le contour de la région en fonction de la projection des coins de la facette 3D.

Le problème de minimisation n'est pas linéaire. En effet, les équations qui lient un point X de l'espace à son image (u,v) sont d'après la section 2.2 :

$$u = u_0 + a_u \frac{R_1 X + t_x}{R_3 X + t_z}$$

$$v = v_0 + a_v \frac{R_2 X + t_y}{R_3 X + t_z}$$

où

- (u_0, v_0) est le centre de l'image,
- a_u et a_v les coefficients d'échantillonnage de la caméra,
- et $(R = \begin{pmatrix} R_1 \\ R_2 \\ R_3 \end{pmatrix}, T)$ le déplacement entre le repère de référence de l'objet et de la caméra. Le déplacement est le seul paramètre inconnu.

Le critère minimisé est la distance entre la projection d'un point et son image, soit (voir section 2.5 pour plus de détails):

$$\sum_i \left(\frac{u_i - u_0}{a_u} - \frac{R_1 X_i + t_x}{R_3 X_i + t_z} \right)^2 + \left(\frac{v_i - v_0}{a_v} - \frac{R_2 X_i + t_y}{R_3 X_i + t_z} \right)^2$$

L'optimisation est faite itérativement par la méthode de Gauss-Newton.

Une fois le recalage effectué, les segmentations originale et induite par la reprojection du modèle dans sa nouvelle position peuvent être comparées. Les correspondances entre les facettes 3D et les paires de régions gauche-droite sont mises à jour. Les appariements supplémentaires permettent alors d'améliorer à nouveau le recalage en retournant à la deuxième étape de l'algorithme. Cet algorithme est itéré jusqu'à la stabilisation du nombre d'appariements 3D/2D.

C.3.2 Par recouvrement des régions

Une méthode de recalage 3D/2D par recouvrement de région peut aussi être utilisée qui présente deux avantages par rapport à la méthode précédente: elle est plus robuste que la méthode basée sur les sommets des régions qui est fortement biaisée par des mauvais appariements 3D/2D, et elle est plus générale car non contrainte par la forme des régions. La méthode est aussi itérative et s'imbrique dans un processus de mise en correspondance facettes/régions. Le recalage du modèle 3D est donc réalisé sur la segmentation par maximisation du nombre de pixels dans l'intersection région/projection de la facette, par une méthode de type Powell sur le déplacement rigide.

D'autres types de recalage fin ont aussi été développés par Andrew Ackah-miezan et Monika Gorkani. Ces méthodes peuvent aussi être utilisées, avec un coût plus ou moins grand en temps de calcul, pour faire le suivi du modèle 3D d'un objet dans une séquence d'images.

- La méthode de recalage 3D/2D par minimisation d'une énergie de contour présente d'autres avantages: elle ne nécessite que le modèle 3D et l'image originale sur laquelle on calcule le potentiel. Il s'agit en effet de trouver la position du modèle minimisant le potentiel le long de son contour projeté. Avec cette approche, on n'impose pas de contraintes sur la forme des modèles à recalage, ni sur leur nombre. Enfin, la méthode n'utilise pas non plus de correspondances explicitées a priori.

- La minimisation sur la position du modèle, d'une énergie traduisant la différence entre l'image brute et l'image générée par la reprojection en tenant compte de la photométrie des facettes, est une autre approche du recalage 3D/2D. Alors, si l'on ne dispose pas d'un modèle de l'éclairage, les régions reprojetées sont d'intensité constante car les facettes 3D sont supposées lambertiennes et éclairées à l'infini.

C.4 Exemples

C.4.1 Une scène de bureau

La technique décrite a été testée sur une scène de bureau relativement complexe (figure C.4) et sur quelques scènes simples.

FIG. C.4 - *Modèle 3D utilisé pour la mise en correspondance de la scène de bureau.*

La figure C.5(a) présente la reprojection du modèle approximativement recalé avec la reconstruction avec la méthode décrite dans la section 4.5 qui est peu précise. Dans la figure C.5, on peut suivre la convergence de l'algorithme jusqu'à la stabilisation. Le nombre maximum d'appariements entre des facettes 3D et des régions n'est que de 8, vu la qualité de la segmentation.

C.4.2 Des parallélépipèdes

Dans le cas précédent, le recalage est réalisée avec le modèle 3D complet, considéré comme un unique objet rigide. Le recalage peut aussi être réalisé séparément sur différents objets dans une même image. Ainsi la figure C.6 montre le résultat des recalages indépendants de trois objets parallélépipédiques dans une scène simple. On voit que deux des parallélépipèdes ont été correctement positionnés dans la scène. Le plus grand est mal repositionné, ce qui illustre la grande sensibilité de cette méthode à la position initiale du recalage qui doit être

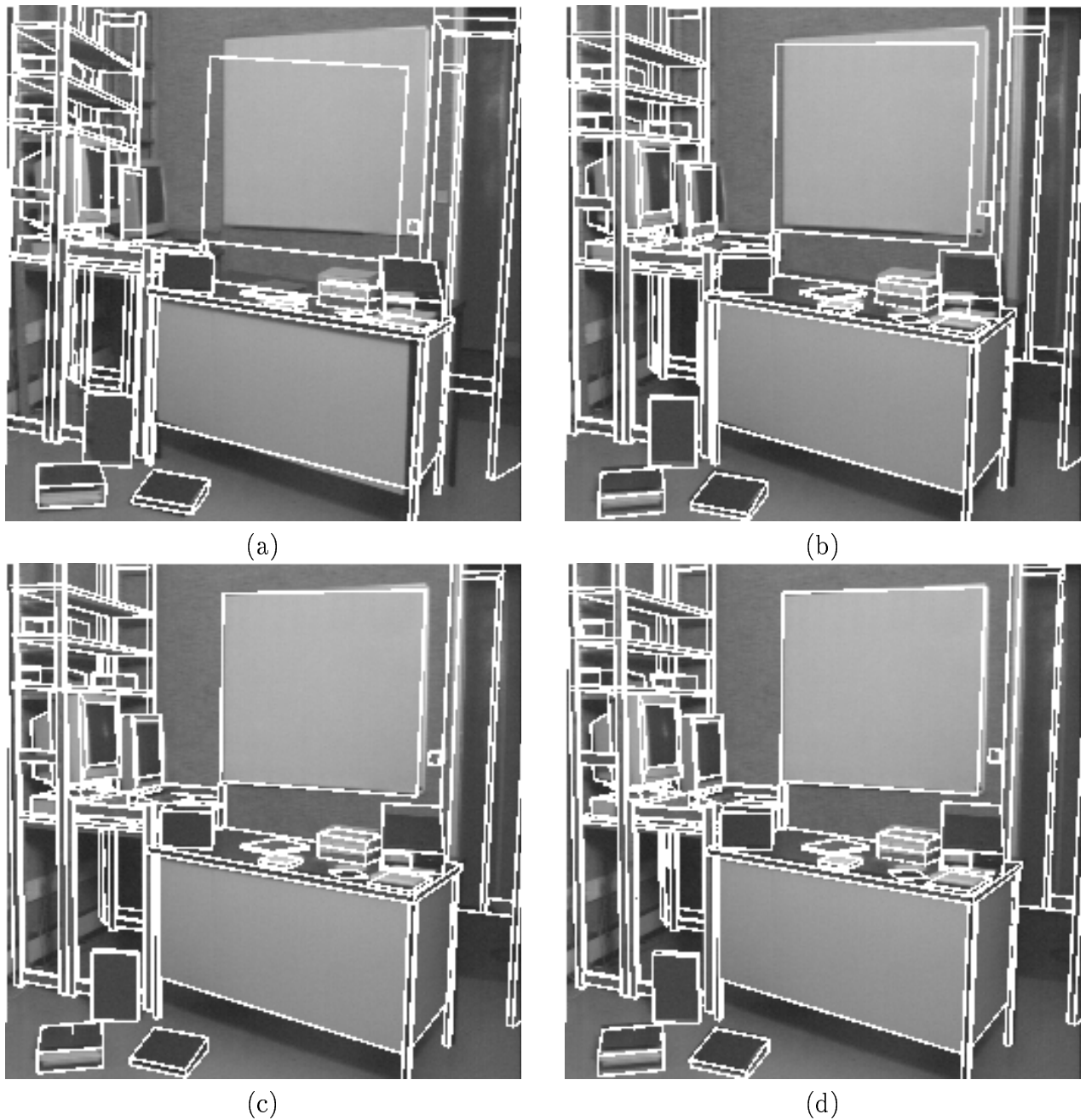


FIG. C.5 - *Reprojection du modèle complet de la scène recalé à partir de la reconstruction (a), après la première étape de recalage fin (b) (2 appariements entre en face 3D et une régions), après la deuxième étape (c) (5 appariements), et après stabilisation (d) (8 appariements).*

très proche de la position réelle. Cet algorithme ne permet donc que de faire du raffinement du recalage et du suivi d'objet 3D qui se déplacent lentement.

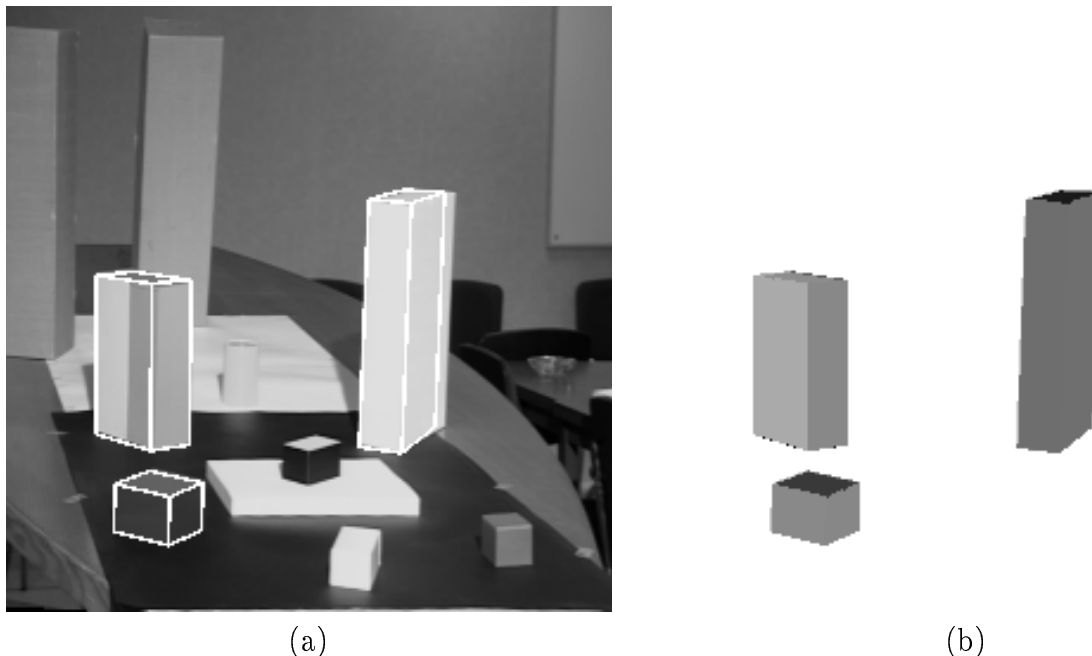


FIG. C.6 - *Superposition des modèles des 3 objets avec l'image originale (a) et segmentation induite par l'ensemble de ces objets (b).*

C.5 Conclusion

Nous avons décrit différentes manières de raffiner le recalage d'un objet 3D par rapport à une paire d'images en utilisant des régions ou des contours. Il est aisé de concevoir une généralisation à un plus grand nombre de vues.

La méthode s'inscrit dans le type d'approche utilisée au chapitre 5, où recalage et appariements sont réalisés et remis en cause tour à tour, itérativement. Les résultats montrent une faible robustesse des méthodes de recalage 3D/2D par rapport à celles qui travaillent dans deux espaces de même dimension.

Nous n'avons pas tenu compte de la présence d'ombres et d'occultations qui peuvent introduire des erreurs dans les méthodes présentées. Mais cette étude peut servir de base pour s'intéresser à ces problèmes. En effet, le traitement des occultations ne semble pas nécessiter d'informations supplémentaires, tandis que pour les ombres, il semble nécessaire de connaître les caractéristiques géométriques des sources présentes dans la scène.

Bibliographie

- [AATRM89] Abi-Ayad (A.), Thiesse (B.), Ragi (C.) et Marthon (Ph.). – Les multiples facettes du problème de calibrage de caméras. *In: 7ème congrès AFCET, Reconnaissance des Formes et Intelligence Artificielle.* – Paris, November 1989.
- [AH88] Ayache (N.) et Hansen (C.). – Rectification of images for binocular and trinocular stereovision. *In: Ninth International Conference on Pattern Recognition (Rome, Italy, November 14–17, 1988).* pp. 11–16. – Washington, DC, 1988.
- [Alo90] Aloimonos (J.Y.). – Perspective approximations. *Image and Vision Computing*, vol. 8, n° 3, August 1990, pp. 179–192.
- [AMG93] Ackah-Miezan (A.) et Gagalowicz (A.). – Discrete models for energy minimizing segmentation. *In: Fourth International Conference on Computer Vision (Berlin, Germany, May 11–14, 1993).* pp. 200–207. – Washington, DC, 1993. <http://www-rocq.inria.fr/syntim/textes/iccv93-eng.html>.
- [And71] Andréani (R.). – *L'objectif photographique.* – Photo-Revue, 1971.
- [Aya88] Ayache (N.). – *Construction et fusion de cartes visuelles 3D : Application à la robotique mobile.* – Thèse de PhD, Université Paris-XI, 1988.
- [Aya93] Ayache (N.). – Computer vision applied to 3D medical imagery: Results, trends and future challenges. *In: Proceedings of the 6th Symposium on Robotics Research.* – MIT Press, also INRIA Tech. Report #2050, 1993.
- [BB82] Ballard (D.H.) et Brown (C.M.). – *Computer Vision.* – Prentice Hall, 1982.
- [BC84] Bolle (R.M.) et Cooper (D.B.). – Bayesian recognition of local 3-D shape by approximating image intensity functions with quadric polynomials. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 6, n° 4, July 1984, pp. 418–429.
- [BCA95] Bardinet (E.), Cohen (L.D.) et Ayache (N.). – *A Parametric Deformable Model to Fit Unstructured 3D Data.* – Rapport technique n° 2617, INRIA, July 1995.
- [BD93] Basclé (B.) et Deriche (R.). – Stereo matching, reconstruction and refinement of 3-D curves using deformable contours. *In: Fourth International Conference on Computer Vision (Berlin, Germany, May 11–14, 1993).* – Washington, DC, 1993.

- [BD94] Blaszkza (T.) et Deriche (R.). – *Recovering and Characterizing Image Features Using An Efficient Model Based Approach*. – Rapport technique n° 2422, INRIA, November 1994.
- [Bey92] Beyer (H.A.). – Accurate calibration of CCD-cameras. *In: CVPR'92 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Champaign, IL, June 15–18, 1992)*. pp. 96–101. – Washington, DC., June 1992.
- [Bez81] Bezdek (J.C.). – *Pattern Recognition with Fuzzy Objective Function Algorithms*. – Plenum Press, 1981.
- [BHST87] Bezdek (J.C.), Hathaway (R. J.), Sabin (M. J.) et Tucker (W. T.). – Convergence theory for fuzzy c-means: Counterexamples and repairs. *IEEE Trans. Systems, Man, and Cybernetics*, vol. 17, n° 5, 1987, pp. 873–877.
- [BM92] Besl (P.J.) et McKay (N.D.). – A method for registration of 3D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, n° 2, February 1992, pp. 239–256.
- [BM94] Bricault (I.) et Monga (O.). – *From volume medical images to quadratic surface patches*. – Rapport technique n° 2380, INRIA, September 1994.
- [BMV93] Boufama (B.), Mohr (R.) et Veillon (F.). – Euclidean constraints for uncalibrated reconstruction. *In: Fourth International Conference on Computer Vision (Berlin, Germany, May 11–14, 1993)*. pp. 466–470. – Washington, DC, 1993.
- [Bou93] Boujemaa (N.). – *Modélisation floue de l'incertitude pour la segmentation d'images*. – Thèse de PhD, Université Paris-V, 1993.
- [Bru59] Bruhat (G.). – *Optique*. – Masson 5ème édition, 1959.
- [BSG90] Bajcsy (R.), Solina (F.) et Gupta (A.). – *Segmentation versus Object Representation—Are They Separable?*, pp. 207–223. – Springer-Verlag, 1990.
- [BSG94] Boujemaa (N.), Stamon (G.) et Gagalowicz (A.). – Modélisation floue pour la segmentation d'images. *In: 9ème congrès AFCET, Reconnaissance des Formes et Intelligence Artificielle*.
- [Bui89] Buil (C.). – *Construction et utilisation des caméras CCD en astronomie amateur*. – Société d'Astronomie Populaire, 1989.
- [BV91] Bolle (R.M.) et Vemuri (B.C.). – On three-dimensional surface reconstruction methods. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, n° 1, January 1991, pp. 1–13.
- [CC94] Cohen (I.) et Cohen (L.D.). – *A Hybrid hyperquadric model for 2-D and 3-D data fitting*. – Rapport technique n° 2188, INRIA, January 1994.

- [CCA91] Cohen (I.), Cohen (L.D.) et Ayache (N.). – *Introducing deformable surfaces to segment 3D images and infer differential structures.* – Rapport technique n° 1403, INRIA, March 1991.
- [CM85] Cochran (S.D.) et Medioni (G.). – 3-D surface description from binocular stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 7, n° 2, 1985, pp. 139–198.
- [CM94] Chabbi (H.) et Masini (G.). – Construction de facettes 3D à partir de triplets stéréoscopiques en utilisant les invariants projectifs. *In : 9ème congrès AFCET, Reconnaissance des Formes et Intelligence Artificielle.*
- [CO85] Cyganski (D.) et Orr (J.A.). – Applications of tensor theory to object recognition and orientation determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 7, n° 6, 1985, pp. 662–674.
- [CR89] Chaumette (F.) et Rives (P.). – Modélisation et calibration d’une caméra. *In : 7ème congrès AFCET, Reconnaissance des Formes et Intelligence Artificielle.* – Paris, November 1989.
- [CT86] Chou (H.L.) et Tsai (W.H.). – A new approach to robot location by house corners. *Pattern Recognition*, vol. 19, n° 6, 1986, pp. 439–451.
- [CT90] Caprile (B.) et Torre (V.). – Using vanishing points for camera calibration. *International Journal of Computer Vision*, vol. 4, 1990, pp. 127–140.
- [Dav91] Dave (R.N.). – Characterization and detection of noise in clustering. *Pattern Recognition Letters*, vol. 12, 1991, pp. 657–664.
- [DB92] Dave (R.N.) et Bhaswan (K.). – Adaptive fuzzy c-shells clustering and detection of ellipses. *IEEE Trans. Neural Networks*, vol. 3, n° 5, 1992, pp. 643–662.
- [Der87] Deriche (R.). – Using canny’s criteria to derive an optimal edge detector recursively implemented. *International Journal of Computer Vision*, vol. 1, n° 2, 1987.
- [Dev95] Devernay (F.). – *A Fast and Efficient Subpixelic Edge Detector.* – Rapport technique n° 2724, INRIA, November 1995.
- [DF95] Deriche (R.) et Faugeras (O.). – *Les EDP en traitement des images et vision par ordinateur.* – Rapport technique n° 2697, INRIA, November 1995.
- [DK86] Dhome (M.) et Kasvand (T.). – Hierarchical approach for polyhedra recognition by hypothesis accumulation. *In : Proceedings, Eighth International Conference on Pattern Recognition (Paris, France, October 27–31, 1986).* pp. 88–91. – Washington, DC, 1986.

- [DRLR88] Dhome (M.), Richetin (M.), Lapreste (J.T.) et Rives (G.). – The inverse perspective problem from a single view for polyhedra location. *In: CVPR'88 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Ann Arbor, MI, June 5–9, 1988)*. pp. 61–66. – Washington, DC., June 1988.
- [DS83] Denis (J.E.) et Schnabel (R.B.). – *Numerical methods for unconstrained optimization and non linear equations*. – Englewood Cliffs, NJ, Prentice-Hall, 1983.
- [EYR80] Eklundh (J.O.), Yamamoto (H.) et Rosenfeld (A.). – A relaxation method for multispectral pixel classification. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 2, n° 1, January 1980, pp. 72–75.
- [FA94] Feldmar (J.) et Ayache (N.). – Locally affine registration of free-form surfaces. *In: CVPR'94 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, WA, June 21–23, 1994)*. pp. 496–501. – Seattle, WA, June 1994.
- [Fau92] Faugeras (O.D.). – What can be seen in three dimensions with an uncalibrated stereo rig? *In: Proceedings, ECCV '92 (Second European Conference on Computer Vision, Santa Margherita Ligure, Italy, May 1992)*, pp. 563–578.
- [FB96] Fua (P.) et Brechbuhler (C.). – Imposing hard constraints on soft snakes. *In: Proceedings, ECCV '96 (Fourth European Conference on Computer Vision, Cambridge, England, April 1996)*, pp. 495–506.
- [Fel95] Feldmar (J.). – *Recalage rigide, non rigide et projectif d'images médicales tridimensionnelles*. – Thèse de PhD, École Polytechnique, 1995.
- [FH83] Faugeras (O.) et Hebert (M.). – The representation, recognition and positioning of 3D shapes from range data. *In: Proceedings of the 8th International Conference On Artificial Intelligence*, pp. 996–1002. – Karlsruhe, BRD, August 1983.
- [FHP83] Faugeras (O.), Hébert (M.) et Pauchon (E.). – Segmentation of range data into planar and quadratic patches. *In: Proceedings, CVPR '83 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Arlington, VA, June, 1983)*. pp. 8–13. – IEEE.
- [FL94] Fua (P.) et Leclerc (Y.G.). – Registration without correspondences. *In: CVPR'94 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, WA, June 21–23, 1994)*. pp. 121–128. – Los Alamitos, CA, June 1994.
- [FLT87] Faugeras (O.), Lustman (F.) et Toscani (G.). – Calcul du mouvement et de la structure à partir de points et de droites. *In: 6ème congrès AFCET, Reconnaissance des Formes et Intelligence Artificielle*, pp. 75–89. – Antibes, November 1987.

- [FLW89] Ferrie (F.P.), Lagarde (J.) et Whaite (P.). – Darboux frames, snake, and superquadrics: Geometry from the bottom-up. *In: Proceedings of Workshop on Interpretation of 3D Scenes*, pp. 170–176. – Austin, TX, 1989.
- [FS88] Faber (T.L.) et Stokely (E.M.). – Orientation of 3-D structures in medical images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, n° 5, 1988, pp. 626–634.
- [FS92] Fua (P.) et Sander (P.T.). – Segmenting unstructured 3D points into surfaces. *In: Proceedings, ECCV '92 (Second European Conference on Computer Vision, Santa Margherita Ligure, Italy, May 1992)*, pp. 676–680.
- [FS93] Flusser (J.) et Suk (T.). – Pattern recognition by affine moment invariants. *Pattern Recognition*, vol. 26, 1993, pp. 167–174.
- [FT86] Faugeras (O.D.) et Toscani (G.). – The calibration problem for stereo. *In: Proceedings, CVPR '86 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Miami Beach, FL, June 22–26, 1986)*. pp. 15–20. – IEEE.
- [FvDFH90] Foley (J. D.), van Dam (A.), Feiner (S. K.) et Hughes (J. F.). – *Computer Graphics: Principles and Practices (2nd Edition)*. – Addison Wesley, 1990.
- [GA92] Guézic (A.) et Ayache (N.). – Smoothing and matching of 3D-space curves. *In: Proceedings, ECCV '92 (Second European Conference on Computer Vision, Santa Margherita Ligure, Italy, May 1992)*.
- [GB88] Gross (A.D.) et Boulton (T.E.). – Error of fit measures for recovering parametric solids. *In: Second International Conference on Computer Vision (Tarpon Springs, FL, December 5–8, 1988)*. pp. 690–694. – Washington, DC, 1988.
- [GB93] Gupta (A.) et Bajcsy (R.). – Volumetric segmentation of range images of 3D objects using superquadric models. *CVGIP - Image Understanding*, vol. 58, n° 3, 1993, pp. 302–326.
- [GLP87] Grimson (W.E.L.) et Lozano-Perez (T.). – Localizing overlapping parts by searching the interpretation tree. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 9, n° 4, July 1987, pp. 469–482.
- [GM86] Gagalowicz (A.) et Monga (O.). – A new approach to image segmentation. *In: Proceedings, Eighth International Conference on Pattern Recognition (Paris, France, October 27–31, 1986)*. pp. 265–267. – Washington, DC, 1986.
- [Gro93] Gros (P.). – *Outils géométriques pour la modélisation et la reconnaissance d'objets polyédriques*. – Thèse de PhD, Institut National Polytechnique de Grenoble, 1993.
- [Har80] Haralick (R.M.). – Using perspective transformations in scenes analysis. *Comp. Graph. and Image Proc.*, vol. 13, 1980, pp. 191–221.

- [HB84] Horaud (P.) et Bolles (R.C.). – 3DPO's strategy for matching three-dimensional objects in range data. *In: International Joint Conference on Artificial Intelligence*, pp. 78–85.
- [HCLL89] Horaud (R.), Conio (B.), Leboulleux (O.) et Lacolle (B.). – An analytic solution for the perspective 4-Point problem. *Computer Vision, Graphics and Image Processing*, vol. 47, n° 1, July 1989, pp. 33–44.
- [Her92] Herman (I.). – *The Use of Projective Geometry in Computer Graphics*. – Springer-Verlag, 1992.
- [HH94] Hoogs (A.) et Hackett (D.). – Model-supported exploitation as a framework for image understanding. *In: Proceedings of the ARPA IU Workshop*.
- [HJL+89] Haralick (R.M.), Joo (H.), Lee (C.N.), Zhuang (X.), Vaidya (V.G.) et Kim (M.B.). – Pose estimation from corresponding point data. *IEEE Trans. Systems, Man, and Cybernetics*, vol. 19, n° 6, November 1989, pp. 1426–1446.
- [HK94] Healey (G.) et Kondepudy (R.). – Radiometric CCD camera calibration and noise estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, n° 3, 1994, pp. 267–276.
- [HLON94] Haralick (R.M.), Lee (C.N.), Ottenberg (K.) et Nolle (M.). – Review and analysis of solutions of the 3-Point perspective pose estimation problem. *International Journal of Computer Vision*, vol. 13, n° 3, December 1994, pp. 331–356.
- [Hor86] Horn (B.K.P.). – *Robot Vision*. – The MIT Press. Mc GRAW-HILL BOOK COMPANY, 1986.
- [HR92] Han (M.) et Rhee (S.). – Camera calibration for three-dimensional measurement. *Pattern Recognition*, vol. 25, n° 2, 1992, pp. 155–164.
- [HRRS86] Hampel (F.R.), Ronchetti (E.M.), Roussecuro (P.J.) et Stahel (W.A.). – *Robust Statistics: An Approach Based on Influence Functions*. – New York: Wiley, 1986.
- [HS85] Haralick (R.M.) et Shapiro (L.G.). – Image segmentation techniques. *Computer Vision, Graphics and Image Processing*, vol. 29, n° 1, January 1985, pp. 100–132.
- [JF92] Jourdan (G.) et Feldmar (J.). – *Reconstruction géométrique et reconnaissance de modèles*. – Rapport de Dea, INRIA, 1992.
- [JMB91] Jolion (J.M.), Meer (P.) et Bataouche (S.). – Robust clustering with applications in computer vision. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, n° 8, August 1991, pp. 791–802.

- [JMN⁺96] Jancène (P.), Meilhac (C.), Neyret (F.), Provot (X.), Tarel (J.-P.), Vézien (J.-M.) et Vérroust (A.). – Réalité enrichie par synthèse. *In: 10ème congrès AF-CET, Reconnaissance des Formes et Intelligence Artificielle.* – Rennes, France, 1996.
- [JNP⁺95] Jancène (P.), Neyret (F.), Provot (X.), Tarel (J.-P.), Vézien (J.-M.), Meilhac (C.) et Vérroust (A.). – RES: computing the interactions between real and virtual objects in video sequences. *In: Second IEEE Workshop on networked Realities.* – Boston, Massachusetts (USA), October 1995. <http://www-rocq.inria.fr/syntim/textes/nr95-eng.html>.
- [JP92] Jaillon (P.) et Peroche (B.). – *Aide à la saisie d'objets en synthèse d'images.* – Rapport technique, Ecole Nationale Supérieure des Mines de Saint-Etienne, 1992.
- [Kan87] Kanatani (K.I.). – 3D recovery from angle clues. *In: Image Analysis*, pp. 167–179. – Stockholm, June 1987.
- [Kan94] Kanatani (K.). – Analysis of 3D rotation fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, n° 5, May 1994, pp. 543–549.
- [KF92] Krishnapuram (R.) et Freg (C.P.). – Fitting an unknown number of lines and planes to image data through compatible cluster merging. *Pattern Recognition*, vol. 25, n° 4, 1992, pp. 385–400.
- [KK93] Krishnapuram (R.) et Keller (M.). – A possibilistic approach to clustering. *IEEE Trans. Fuzzy Systems*, vol. 1, n° 2, May 1993, pp. 98–110.
- [KLM94] Koepfler (G.), Lopez (C.) et Morel (J.M.). – A multiscale algorithm for image segmentation by variational method. *SIAM J. Numerical Analysis*, vol. 31, 1994, pp. 282–299.
- [KN95] Krishnapuram (R.) et Nasraoui (O.). – Crisp interpretations of fuzzy and possibilistic clustering algorithms. *In: Proceedings of Third European Congress Intelligent Techniques and Soft Computing, EUFIT'95*, pp. 1312–1318. – Aachen, Germany, 1995.
- [KNF92] Krishnapuram (R.), Nasraoui (O.) et Frigui (H.). – The fuzzy c-spherical shells algorithm: A new approach. *IEEE Trans. Neural Networks*, vol. 3, n° 5, 1992, pp. 663–670.
- [KPLR91] Kamgar-Parsi (B.), Lones (J.) et Rosenfeld (A.). – Registration of multiple overlapping range images: Scene without distinctive features. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, n° 9, September 1991, pp. 857–870.
- [KVB93] Koivunen (V.), Vézien (J.-M.) et Bajcsy (R.). – Procedural CAD-models from range data. *In: Proceedings of Towards World Class Manufacturing '93.*

- [KWT87] Kass (M.), Witkin (A.) et Terzopoulos (D.). – Snakes: Active contour models. *International Journal of Computer Vision*, vol. 1, 1987, pp. 321–331.
- [LD89] Lo (C.) et Don (H.). – 3-D moment forms: Their construction and application to object identification and positioning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, n° 10, 1989, pp. 1053–1064.
- [Lec89] Leclerc (Y.G.). – Constructing simple stable descriptions for image partitioning. *International Journal of Computer Vision*, vol. 3, 1989, pp. 73–102.
- [LH81] Longuet-Higgins (H.C.). – An computer algorithm for reconstructing a scene from two projections. *Nature*, vol. September, 1981, pp. 133–138.
- [LM89] Lutton (E.) et Maitre (H.). – Etude des symétries du problème de perspective à trois lignes. In: *7ème congrès AFCET, Reconnaissance des Formes et Intelligence Artificielle*, pp. 537–546. – Paris, November 1989.
- [LM94] Liao (C.W.) et Medioni (G.). – Simultaneous segmentation and approximation of complex patterns. In: *CVPR'94 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, WA, June 21-23, 1994)*. pp. 617–623. – Los Alamitos, CA, June 1994.
- [LT88] Lenz (R.K.) et Tsai (R.Y.). – Techniques for calibration of the scale factor and image center for high accuracy 3-D machine vision metrology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, n° 5, 1988, pp. 713–720.
- [LTM96] Lengagne (Richard), Tarel (Jean-Philippe) et Monga (Olivier). – From 2d images to 3d face geometry. In: *Proceedings of IEEE Second International Conference on Automatic Face and Gesture Recognition*. – Killington, USA, 1996. <http://www-rocq.inria.fr/syntim/textes/fg96-eng.html>.
- [Lus87] Lustman (F.). – *Vision stéréoscopique et perception du mouvement en vision artificielle*. – Thèse de PhD, Université Paris-Sud, 1987.
- [LVG92] Lutton (E.), Vézien (J-M.) et Gagalowicz (A.). – Model based stereo reconstruction by energy minimization. In: *Proceedings, ISCIS*. – Bordeaux, France, 1992.
- [Ma93a] Ma (S. De). – A complete two-plane camera calibration method and experimental comparisons. In: *Fourth International Conference on Computer Vision (Berlin, Germany, May 11-14, 1993)*. pp. 439–446. – Washington, DC, 1993.
- [Ma93b] Ma (S. De). – Conics-based stereo, motion estimation and pose determination. *International Journal of Computer Vision*, vol. 10, n° 1, 1993.
- [MDC92] Mukherjee (J.), Das (P.P.) et Chatterji (B.N.). – Segmentation of range images. *Pattern Recognition*, vol. 25, n° 10, 1992, pp. 1141–1156.

- [MF92] Maybank (S.J.) et Faugeras (O.D.). – A theory of self-calibration of a moving camera. *International Journal of Computer Vision*, vol. 8, n° 2, 1992, pp. 123–151.
- [MM91] Mohr (R.) et Morin (L.). – Relative positioning from geometric invariants. *In: CVPR'91 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Lahaina, Maui, HI, June 3-6, 1991)*. pp. 139–144. – Hawaii, June 1991.
- [Moh93] Mohr (R.). – Projective geometry and computer vision. *Handbook of Pattern Recognition and Computer Vision*, 1993, pp. 369–393.
- [MYTG94] Meas-Yedid (Vannary), Tarel (Jean-Philippe) et Gagalowicz (André). – Calibration métrique faible et construction interactive de modèles 3D de scènes. *In: Congrès Reconnaissance des Formes et Intelligence Artificielle. AFCET.* – Paris, France, 1994. <http://www-rocq.inria.fr/syntim/textes/van-eng.html>.
- [Nas94] Nastar (C.). – *Modèles physiques déformables et modes vibratoires pour l'analyse du mouvement non-rigide dans les images multidimensionnelles.* – Thèse de PhD, École Nationale des Ponts et Chaussées, 1994.
- [OK85] Ohta (Y.) et Kanade (T.). – Stereo by intra- and inter-scanline search using dynamic programming. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 7, n° 2, 1985, pp. 139–198.
- [OS91] Oakley (J.P.) et Shann (R.T.). – Efficient method for finding the position of object boundaries to sub-pixel resolution. *Image and Vision Computing*, vol. 9, n° 4, 1991, pp. 262–272.
- [PCO85] Pinjo (Z.), Cyganski (D.) et Orr (J.A.). – Determination of 3-D object orientation from projections. *Pattern Recognition Letters*, vol. 3, n° 5, 1985, pp. 351–356.
- [Pen87] Pentland (A.). – Recognition by parts. *In: First International Conference on Computer Vision, (London, England, June 8–11, 1987)*. pp. 612–620. – Washington, DC, 1987.
- [Pér93] Pérez (J.Ph.). – *Optique géométrique matricielle et ondulatoire.* – Masson, 1993.
- [PHYT95] Phong (T.Q.), Horaud (R.), Yassine (A.) et Tao (P.D.). – Object pose from 2-D to 3-D point and line correspondences. *International Journal of Computer Vision*, vol. 15, n° 3, July 1995, pp. 225–243.
- [PS91] Pentland (A.) et Sclaroff (S.). – Closed-form solutions for physically based shape modelling and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, n° 7, July 1991, pp. 715–729.

- [Rag89] Ragi (C.). – *N.O.I.S.I.V un système pour l'expérimentation en vision passive tridimensionnelle par ordinateur*. – Thèse de PhD, Institut national polytechnique de Toulouse, 1989.
- [Ran92] Randriamasy (S.). – *Segmentation descendante coopérative en régions de paires d'images stéréoscopiques*. – Thèse de PhD, Université Paris-IX Dauphine, 1992.
- [Rei93] Reiss (T.H.). – *Recognizing Planar Object Using Invariant Image Features*. – Lecture Notes in Computer Science, Springer-Verlag, 1993.
- [RG91] Randriamasy (S.) et Gagalowicz (A.). – Region based stereo matching oriented image processing. *In: CVPR'91 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Lahaina, Maui, HI, June 3-6, 1991)*. – Washington, DC., June 1991.
- [RH91] Rigoutsos (I.) et Hummel (R.). – Robust similarity invariant matching in the presence of noise: a data parallel approach. *In: Proceedings of the 8th Israeli Conference on Artificial Intelligence and Computer Vision*.
- [RZMF92] Rothwell (C.A.), Zisserman (A.), Marinos (C.I.) et Forsyth (D.A.). – Relative motion and pose from arbitrary plane curves. *Image and Vision Computing*, vol. 10, n° 4, 1992, pp. 250–262.
- [SAMLG93] Serfaty (V.), Ackah-Miezan (A.), Lutton (E.) et Gagalowicz (A.). – Photometric analysis as an aid to 3D reconstruction of indoor scenes. *In: Image Modeling, L.A. Ray, J.R. Sullivan*, pp. 196–207. – Editors, proc. SPIE 1904.
- [SB95] Serra (B.) et Berthod (M.). – Optimal subpixel matching of contour chains and segments. *In: Fifth International Conference on Computer Vision (Cambridge, MA, June 20–22, 1995)*. – Washington, DC, 1995.
- [Slu95] Sluzek (A.). – Identification and inspection of 2-D objects using new moment-based shape descriptors. *Pattern Recognition Letters*, vol. 16, n° 7, July 1995, pp. 687–697.
- [Sto87] Stockman (G.). – Object recognition and localization via pose clustering. *Computer Vision, Graphics and Image Processing*, vol. 40, n° 2, 1987, pp. 361–387.
- [Sto90] Stockman (George). – *Object Recognition*, pp. 225–253. – Springer-Verlag, 1990.
- [SVCG89] Sander (P.), Vinet (L.), Cohen (L.) et Gagalowicz (A.). – Hierarchical regions based stereo matching. *In: Proceedings of the Sixth Scandinavian Conference on Image Analysis*, pp. 71–78. – Oulu, Finland, June 1989.
- [SW92] Stein (A.) et Werman (M.). – Robust statistics in shape fitting. *In: CVPR'92 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Champaign, IL, June 15–18, 1992)*. pp. 540–546. – Washington, DC., June 1992.

- [Tar94] Tarel (Jean-Philippe). – *Calibration de caméra fondée sur les ellipses*. – Rapport de recherche n° 2200, INRIA, 1994. <http://www-rocq.inria.fr/syntim/textes/calib94-eng.html>.
- [Tar95] Tarel (Jean-Philippe). – *Calibration Radiométrique de Caméra*. – Rapport de recherche n° 2509, INRIA, 1995.
- [Tar96a] Tarel (Jean-Philippe). – Multi-objets interpretation. *In: Proceedings, 13th International Conference on Pattern Recognition (ICPR13, Vienna, Austria, August 25-30, 1996)*. pp. 612–616. – Computer Society Press. <http://www-rocq.inria.fr/syntim/textes/icpr96-tarel-eng.html>.
- [Tar96b] Tarel (Jean-Philippe). – *Reconstruction globale et robuste de facettes 3D*. – Rapport de recherche n° 2813, INRIA, February 1996. <http://www-rocq.inria.fr/syntim/textes/RR-2813-fra.html>.
- [Tar96c] Tarel (Jean-Philippe). – Une méthode de calibration radiométrique de caméra à focale variable. *In: 10ème congrès AFCET, Reconnaissance des Formes et Intelligence Artificielle*. – Rennes, France, 1996.
- [Tau91] Taubin (G.). – Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, n° 11, November 1991, pp. 1115–1138.
- [Tau93] Taubin (G.). – An improved algorithm for algebraic curve and surface fitting. *In: Fourth International Conference on Computer Vision (Berlin, Germany, May 11-14, 1993)*. pp. 658–665. – Washington, DC, 1993.
- [TB95a] Tarel (Jean-Philippe) et Boujemaa (Nozha). – A fuzzy 3D registration method. *In: Proceedings of Third European Congress Intelligent Techniques and Soft Computing, EUFIT'95*, pp. 1359–1365. – Aachen, Germany, 1995.
- [TB95b] Tarel (Jean-Philippe) et Boujemaa (Nozha). – *Une approche floue du recalage 3D : généralité et robustesse*. – Rapport de recherche n° 2716, INRIA, 1995.
- [TB96] Tarel (Jean-Philippe) et Boujemaa (Nozha). – Une approche floue du recalage 3D : généralité et robustesse. *In: 10ème congrès AFCET, Reconnaissance des Formes et Intelligence Artificielle*. – Rennes, France, 1996.
- [TG94] Tarel (Jean-Philippe) et Gagalowicz (André). – Calibration de caméra à base d'ellipses. *In: 9ème congrès AFCET, Reconnaissance des Formes et Intelligence Artificielle*. – Paris, France, 1994.
- [TG95] Tarel (Jean-Philippe) et Gagalowicz (André). – Calibration de caméra à base d'ellipses. *Traitement du Signal*, vol. 12, n° 2, 1995, pp. 177–187. – <http://www-rocq.inria.fr/syntim/textes/calib-eng.html>.

- [Tos87] Toscani (G.). – *Systèmes de calibration et perception du mouvement en Vision Artificielle*. – Thèse de PhD, Université Paris-Sud, 1987.
- [Tsa86] Tsai (R.Y.). – An efficient and accurate camera calibration technique for 3D machine vision. In: *Proceedings, CVPR '86 (IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Miami Beach, FL, June 22–26, 1986)*. pp. 364–374. – IEEE.
- [Tsa87] Tsai (R.Y.). – A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, vol. 3, 1987, pp. 323–344.
- [TV95] Tarel (Jean-Philippe) et Vézien (Jean-Marc). – A generic approach for planar patches stereo reconstruction. In: *Proceedings of the Scandinavian Conference on Image Analysis*, pp. 1061–1070. – Uppsala, Sweden, 1995.
- [Usu94] Usui (Fumiaki). – Zoom lens. – Canon Inc brevet n° JP6250086 (En Japonais), June 1994.
- [Vai90] Vaillant (R.). – *Géométrie différentielle et vision par ordinateur: détection et reconstruction des contours d'occultation de la surface d'un objet non-polyédrique*. – Thèse de PhD, Université Paris-Sud, 1990.
- [Véz95] Vézien (J.M.). – *Techniques de reconstruction globale par analyse de paires d'images stéréoscopiques*. – Thèse de PhD, Université Paris-VII, 1995.
- [Vin91] Vinet (L.). – *Segmentation et mise en correspondance de régions de paires d'images stéréoscopiques*. – Thèse de PhD, Université Paris-IX Dauphine, 1991.
- [Vio95] Viola (P.). – *Entropy, Information, Computer Vision and Image Processing*. – Thèse de PhD, MIT, 1995.
- [VT95] Vézien (Jean-Marc) et Tarel (Jean-Philippe). – *A Generic Approach for Planar Patches Stereo Reconstruction*. – Rapport de recherche n° 2507, INRIA, 1995. <http://www-rocq.inria.fr/syntim/textes/RR-2507-eng.html>.
- [WCH90] Weng (J.), Cohen (P.) et Herniou (M.). – Calibration of stereo cameras using a non linear distortion model. In: *Proceedings, Tenth International Conference on Pattern Recognition (Atlantic City, USA, June 16–21, 1990)*. pp. 246–253. – Washington, DC, 1990.
- [Wei90] Weiss (I.). – Shape reconstruction on a varying mesh. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, n° 4, April 1990, pp. 345–362.
- [WI95] Wheeler (M.D.) et Ikeuchi (K.). – Sensor modeling, probabilistic hypothesis generation, and robust localization for object recognition. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, n° 3, March 1995, pp. 252–265.

- [XB91] Xie (X.L.) et Beni (G.). – A validity measure for fuzzy clustering. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, n° 8, August 1991, pp. 841–847.
- [YL89] Yokoya (N.) et Levine (M.D.). – Range image segmentation based on differential geometry: A hybrid approach. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, n° 6, June 1989, pp. 643–649.
- [Zha94] Zhang (Z.). – Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, vol. 13, n° 2, 1994, pp. 119–152. – also INRIA Tech. Report #1658.
- [ZLF96] Zhang (Z.Y.), Luong (Q.T.) et Faugeras (O.). – Motion of an uncalibrated stereo rig: Self-calibration and metric reconstruction. *IEEE Trans. Robotics and Automation*, vol. 12, n° 1, February 1996, pp. 103–113.
- [ZM90] Zhang (J.) et Modestino (J.W.). – A model-fitting approach to cluster validation with application to stochastic model-based image segmentation. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, n° 10, October 1990, pp. 1009–1017.

Table des matières

Notations	7
Introduction	9
Modélisation polyédrique par stéréoscopie	9
Estimation géométrique et appariement	13
Plan du document	14
1 Couplage entre l'estimation et l'appariement	17
1.1 Introduction	17
1.2 Séparer l'estimation et l'appariement	18
1.2.1 Différents niveaux de difficultés	18
1.2.2 Comment posons-nous un problème?	19
1.3 Estimation avec appariement	20
1.3.1 Translation	21
1.3.2 Déplacement rigide/Rotation	22
1.3.3 Similitude/Rotation et homothétie	23
1.3.4 Étirement/Rotation et changement d'échelle	24
1.3.5 Affinité/Transformation linéaire	25
1.3.6 Homographie	25
1.3.7 Projection perspective	26
1.3.8 Conclusion	27
1.4 Estimation sans appariement	28
1.4.1 Translation	29
1.4.2 Déplacement rigide/Rotation	29
1.4.3 Similitude/Rotation et homothétie	32
1.4.4 Étirement/Rotation et changement d'échelle	33
1.4.5 Affinité/Transformation linéaire	33
1.4.6 Homographie	38
1.4.7 Projection perspective	39
1.4.8 Conclusion	39
1.5 Robustesse dans la pratique	39
1.5.1 Influence du bruit sur l'estimation du déplacement avec ou sans appariement	39

1.5.2	Influence du bruit sur l'estimation de la similitude avec ou sans appariement	40
1.5.3	Influence du bruit sur l'estimation d'une transformation linéaire 2D sans appariement	41
1.5.4	Influence du bruit sur l'estimation d'une transformation homographique 1D sans appariement	42
1.5.5	Estimation d'une transformation perspective avec ou sans appariement	43
1.6	Appariement et estimation couplés	44
1.6.1	Méthodes ICP	45
1.6.2	Classification floue	47
1.7	Conclusion	50
2	Calibration précise de caméra par estimation des moments 'image'	51
2.1	Introduction	51
2.2	Modélisation de la caméra	52
2.2.1	Le modèle sténopé	52
2.2.2	Modèles avec distorsions	53
2.3	Mesures dans l'image	54
2.3.1	Principe	54
2.3.2	Correction des gradients d'intensité	55
2.3.3	Mesures des moments géométriques	58
2.3.4	Précision du calcul des moments	58
2.3.5	Conclusion	60
2.4	Transformation perspective d'une ellipse	60
2.4.1	Caractérisations d'une ellipse	61
2.4.2	Matrice d'inertie homogène	62
2.4.3	projection d'une ellipse	63
2.4.4	Conclusion	64
2.5	Résolution	64
2.5.1	Critère d'adéquation	64
2.5.2	Méthode itérative	65
2.6	Résultats	65
2.6.1	Images de synthèse	66
2.6.2	Images réelles	69
2.7	Conclusion	72
2.8	Calcul de l'erreur sur les moments	72
3	Reconstruction globale de facettes 3D à partir d'images segmentées	75
3.1	Introduction	75
3.2	L'espace des disparités	78
3.2.1	Projection stéréoscopique d'un point	78
3.2.2	Projection stéréoscopique d'un point dans un plan	79
3.2.3	De l'espace des disparités vers celui réel	80
3.2.4	Relation affine entre les images	82

3.2.5	Interprétation géométrique	82
3.2.6	Images intermédiaires	83
3.2.7	Rectification de paires stéréoscopiques	84
3.3	La reconstruction de facettes 3D	86
3.3.1	La reconstruction géométrique	86
3.3.2	Cohérence de la reconstruction géométrique	88
3.3.3	La reconstruction photométrique	89
3.3.4	Cohérence de la reconstruction photométrique	90
3.4	Résultats de reconstructions	90
3.4.1	Images de synthèse	91
3.4.2	Méthodes géométrique et photométrique	91
3.4.3	Sélection des faces	93
3.4.4	Des objets non polygonaux	95
3.4.5	Avec ou sans rectification intermédiaire	97
3.5	Comparaison avec d'autres méthodes	97
3.5.1	Présentation de l'outil de mesure	97
3.5.2	Performance de la reconstruction	99
3.5.3	Comparaison des méthodes de reconstruction	101
3.6	conclusion	103
4	Méthodes Floue et ICP pour un recalage 3D plus précis et robuste	105
4.1	Introduction	105
4.2	Schéma général du recalage 3D avec une approche floue	108
4.2.1	Mise en correspondance de primitives 3D	108
4.2.2	Localisation 3D approchée	109
4.2.3	Raffinement itératif du recalage	115
4.2.4	Résumé	116
4.3	Résultats	117
4.3.1	Réglage des paramètres	117
4.3.2	Les objets symétriques	117
4.3.3	Utilisation de la couleur	118
4.3.4	Recalage avec plusieurs objets	120
4.3.5	Validation par rétroprojection	122
4.3.6	Écart de localisation	122
4.4	Conclusion	125
4.5	Transformée de Hough généralisée aux déplacements rigides	126
5	Modélisation par des prototypes recouvrants	129
5.1	Introduction	129
5.2	L'approche Modélisation Multi-Prototypes (MMP)	130
5.3	Application au recouvrement par des formes géométriques simples	131
5.3.1	Généralisation de la méthode ICP (GICP)	131
5.3.2	Méthodes de classification	135
5.3.3	Unification des algorithmes GICP et FCM	136

5.3.4	Résultats	138
5.3.5	Discussion	150
5.4	Application à la segmentation des images	150
5.4.1	Algorithme MMP adapté à la segmentation d'images	151
5.4.2	Segmentation des cartes de profondeur	151
5.4.3	Segmentation des images d'intensité	154
5.4.4	Segmentation d'images couleur	154
5.4.5	Discussion	156
5.5	Application à plusieurs modèles physiques déformables	156
5.5.1	Modèle déformable constitué de segments articulés	157
5.5.2	Recalage fin sur des contours polyédriques	159
5.5.3	Recouvrement par plusieurs contours	159
5.5.4	Discussion	162
5.6	Conclusion	162
Conclusion		165
	Contributions	165
	Perspectives	166
Annexes		169
A Calibration Radiométrique de Caméra à focale variable		169
A.1	Introduction	169
A.2	Défauts radiométriques	170
A.2.1	Écarts de sensibilité de la matrice CCD	170
A.2.2	Courant d'obscurité	170
A.2.3	Variations d'éclairement dues à l'objectif	171
A.3	Correction radiométrique	173
A.4	Expérimentations	174
A.4.1	Bruit temporel	174
A.4.2	Courant d'obscurité	175
A.4.3	Écart de sensibilité	176
A.4.4	Variation d'éclairement	178
A.4.5	Stabilité de la variation d'éclairement	179
A.4.6	Modèle expérimental simplifié	179
A.5	Exemples de corrections de la radiométrie	180
A.6	Conclusion	183
B Calibration pour la modélisation interactive		185
B.1	Introduction	185
B.2	Rappel des équations de base	187
B.3	Segments	188
B.4	Segments jointifs	190
B.5	Algorithme de résolution	190
B.6	Existence d'une solution	191

B.7	Application à la construction interactive	192
B.7.1	Construction interactive	192
B.7.2	Résultats de calibration	193
B.8	Conclusion	197
C	Recalage 3D/2D fin	201
C.1	Introduction	201
C.2	Mise en correspondance 3D/2D par reprojection	203
C.3	Estimation d'un meilleur recalage	204
C.3.1	À partir de sommets sur les contours	204
C.3.2	Par recouvrement des régions	205
C.4	Exemples	206
C.4.1	Une scène de bureau	206
C.4.2	Des parallélépipèdes	206
C.5	Conclusion	208
	Références bibliographiques	209

Table des figures

0.1	Reconstruction en facettes texturées de deux scènes.	11
0.2	Le processus d'analyse stéréoscopique.	12
1.1	Mises en correspondances possibles d'une facette 3D réduite à ses axes propres (ellipse visualisée par son rectangle englobant).	31
1.2	En (a) est tracée la fonction $x^2y - xy^2 = 1$ qui a trois branches. Les trois axes isotropes de la forme $x^2y - xy^2$ sont les trois asymptotes. En (c), la fonction $x^3 + y^3 = 1$ est visible. Elle est associée à une forme triadique de dos d'âne. Les deux axes propres de la forme $x^3 + y^3$ sont les deux axes (Ox) et (Oy) . L'axe isotrope est l'asymptote. En (b) est tracé: $x^3 + x^2y = 1$, qui a deux asymptotes et qui est associé à une forme triadique dégénérée qui possède deux axes isotropes.	36
1.3	Comparaison de l'erreur d'estimation de l'angle de la rotation entre l'algorithme qui utilise la décomposition SVD et nécessite de connaître les appariements et celui qui utilise les matrices d'inertie où l'appariement est inconnu.	40
1.4	Comparaison de l'erreur d'estimation de l'échelle entre l'algorithme qui utilise la décomposition SVD et celui qui utilise les matrices d'inertie.	41
1.5	Comparaison de l'erreur d'estimation de la transformée linéaire entre notre algorithme qui utilise les tenseurs d'ordre 3 et celui de Cyganski qui utilise les tenseurs jusqu'à l'ordre 4.	42
1.6	Erreur d'estimation de la transformée homographique 1D sans appariement.	43
1.7	Erreur d'estimation de la transformée perspective avec appariement.	44
1.8	Différents types d'appariements possibles entre deux ensembles de quatre points. La figure (a), où l'appariement est bijectif, correspond à celui qui est supposé dans les deux premières catégories. La figure (b) montre le type d'appariement qui apparaît dans les problèmes de la troisième catégorie. Enfin, la figure (c) montre les types d'appariements qui doivent être traités pour résoudre un problème de la dernière catégorie.	50
2.1	Le modèle sténopé de la caméra.	53
2.2	Réflexion diffuse d'une surface lambertienne.	55
2.3	(a) limite des zones claire et sombre, (b) correction de la variation linéaire de l'intensité pour obtenir l'image normalisée.	56
2.4	Distribution réelle de l'intensité d'un disque blanc sur fond noir.	57

2.5	Pour chaque pixel, l'aire de la forme frisée est le même, mais leurs moments géométriques d'ordre 1 et 2 sont différents. Cette indétermination au niveau sous-pixel induit l'erreur e_{local} .	59
2.6	Repère absolu \mathfrak{R} et canonique $\mathfrak{R}'(G, V_1, V_2)$ d'une ellipse.	61
2.7	Image synthétique de la mire.	70
2.8	Image de la mire métallique.	70
2.9	Discretisation de la courbe d'intensité d'un échelon parfait (u_m, u_M) avant $S(u)$ et après $CS(u)$ convolution par la réponse de la caméra.	73
3.1	Reconstruction d'un objet 3D à partir d'une paire segmentée en régions.	76
3.2	Le point U_g est l'image d'un point sur l'axe qui passe par C_g et X . Cette droite se projette sur l'image droite et définit la ligne épipolaire du point U_g . Le point U_d appartient donc nécessairement à la ligne épipolaire $L_d(U_g)$.	78
3.3	Géométrie des caméras rectifiées sur une coupe 2D.	81
3.4	(a) Géométrie des caméras dans le cas rectifié. Les épipolaires sont parallèles. (b) Géométrie cohérente à la géométrie du cas (a) après rejet des centres des caméras à l'infini.	83
3.5	Coupe 2D de l'espace des disparités suivant un plan épipolaire (voir figure 3.4). Pour un point à l'infini $u_g = u_d$. Cette équation définit donc le plan à l'infini. Ainsi, la partie hachurée n'est pas visible par la caméra.	84
3.6	Paire stéréoscopique avant et après rectification. Les épipolaires sont devenues horizontales.	85
3.7	Stabilité de l'invariant Inv_1 et instabilité des invariants Inv_2 et Inv_3 en fonction des changements d'orientation d'une face non occultée.	89
3.8	Efficacité des invariants Inv_1 , Inv_2 et Inv_3 en fonction du degré d'occultation.	89
3.9	Vue de face (a) et de côté (b) de la reconstruction (traits en zigzag) d'un objet de synthèse et du modèle exact associé.	91
3.10	Vue de face (a) et de dessus (b) de la reconstruction du cube de la scène, à partir de la géométrie des régions. Vue de face (c) et de dessus (d) de la reconstruction photométrique du cube de la scène. Vue de face (e) et de dessus (f) de la reconstruction photométrique, après correction radiométrique des images. Les images originales sont dans la figure 3.11	92
3.11	Paire stéréoscopique réelle d'un cube après rectification.	93
3.12	Vue de face (a) et de côté (b) de la reconstruction de la scène visible dans la figure 3.6. Vue de face (c) et de côté (d) de la reconstruction de la même scène après élimination automatique des faces mal segmentées.	94
3.13	Vue de face (a), de côté (b) et de dessus (c) de la reconstruction de la mire de calibration obtenue à partir de l'image (d).	95
3.14	Vue de face (a), de côté (b) et de dessus (c) du ballon reconstruit à partir de l'image originale (d).	96

3.15	Vues originales gauche (a) et droite (b) d'une scène qui forme deux plans. Vues légèrement décalées vers le haut des contours (c) et en faces texturées (d) de la reconstruction sans rectification intermédiaire et donc avec approximation de la projection perspective (Méthode de Jean-Marc Vézien). Vues légèrement de dessus des contours (e) et en faces texturées (f) de la reconstruction avec rectification et notre méthode sans approximation.	98
3.16	Le modèle de référence (a), notre reconstruction (b) et celle gracieusement fournie de B. Serra (c). B. Serra utilise pour faire la reconstruction un algorithme de mise en correspondance sous-pixel des contours des deux images. Ces contours sont représentés par des chaînes ouvertes de segments. Cette méthode permet donc d'obtenir une bonne précision en même temps d'une reconstruction d'une grande partie des contours.	100
3.17	Chaque point représente le sommet d'un déplacement 3D entre un point du modèle et de la reconstruction. La partie centrale, proche de l'origine, est donc associée à la partie commune entre la reconstruction et le modèle de la mire. La partie en noir correspond aux appariements aberrants.	101
3.18	(a) Séparation du modèle parfait de la mire en la partie reconstruite et non reconstruite avec notre méthode (aux artéfacts près). (b) Même résultat mais avec la reconstruction de B. Serra.	102
3.19	Calcul de l'erreur de localisation stéréoscopique d'un objet à une distance p_{objet} avec une base d_{base}	102
3.20	Histogrammes des déplacements entre le modèle et les reconstructions obtenues avec notre méthode et celle de Serra.	103
4.1	Approche générale du recalage 3D mixte. (R, T) est un déplacement. Un coefficient de confiance φ est accordé à chaque déplacement.	107
4.2	Appariement des faces i et j de deux ensembles de faces. Pour une paire (i, j) de rectangles. Quatre déplacements (R_{ij}, T_{ij}) sont au moins possibles, comme le montre la figure 1.1	109
4.3	Variations de la valeur de confiance φ_{ij} avec la différence de la longueur du premier axe propre $\alpha_i - \alpha_j$. e_{recons} est l'erreur moyenne sur les données 3D.	110
4.4	Projection 2D de l'ensemble des rotations 3D à classifier pour obtenir le point d'accumulation central correspondant à une hypothèse intéressante pour le recalage.	110
4.5	(a) variations de $e_{classique}(t_1)$ avec $t_2 = 2$. (b) variations de $e_{FCM}(t_1)$ avec $t_2 = 2$ et $m = 1.5$. On remarque que la le critère flou est plus lisse que celui de la classification classique.	112
4.6	L'appariement est réalisé avec le plus proche voisin, et suivant la longueur du vecteur déplacement, la correspondance est prise ou n'est pas prise en compte.	116
4.7	(a) Objet 3D de synthèse. (b) Une solution de la phase du recalage approché qui possède deux solutions également possibles du fait la symétrie de l'objet par rapport à son axe médian (c).	118
4.8	(a)(b) paire stéréoscopique traitée pour obtenir la reconstruction de figure 4.9 (a).	119

4.9	(a) Vue de la reconstruction 3D obtenue par analyse sur la paire de la figure 4.8 . (b) Le modèle parfait du rubik cube est recalé sur la reconstruction.	119
4.10	(a)(b) Images originales de la scène. Les vues des résultats de reconstruction 3D obtenus avec nos algorithmes de stéréovision sont présentées dans la figure 4.11 (a)(b).	120
4.11	(a) et (b) sont deux points de vue de la reconstruction 3D obtenue à partir de la paire de la figure 4.10 . Dans (c), la mire est recalée sur la reconstruction. Dans (d), seul le ballon est recalé, alors que (e) et (f) montrent les deux objets recalés. Ces résultats ont été obtenus par l'application complète des trois étapes de la méthode pour avoir un recalage très précis.	121
4.12	(a)(b) Point de vue de synthèse des modèles recalés (voir figure 4.11 (e)(f)) sur lesquels l'image originale 4.10 (a) est projetée. (c) Point de vue synthétique de la mire de calibration seule. (d) Point de vue synthétique du ballon obtenu après l'analyse de la paire de la figure 0.2 . (e)(f) Point de vue de synthèse sur la scène du rubik cube de la figure 4.9 (b) sur lequel est projetée l'image de la figure 4.8 (a).	123
4.13	Histogramme des longueurs en mm des déplacements entre les points appariés entre les deux ensembles 3D. Dans cette exemple, l'objet de référence est celui de la figure 4.14 ; le deuxième ensemble est la reconstruction 3D visible dans la figure 4.11 (a)(b).	124
4.14	Segmentation de l'ensemble de référence en sa partie commune avec la reconstruction (en gris) et sa partie non appariaable. La reconstruction 3D associée est visible sur la figure 4.11 (a)(b). Comme le recalage de la mire est très proche de la position obtenue par la calibration, ce résultat est très proche de celui de la figure 3.18(b)	125
4.15	Exemple d'accumulateur de vote de Hough sur l'espace des déplacements représenté par $(\theta_x, \theta_y, \theta_z, t_x, t_y, t_z)$. Le nombre possible de nœuds de chaque niveau est n . À chaque feuille est associé la valeur du vote et la liste des couples votants.	127
5.1	Différentes formes engendrées par 5 types de transformations géométriques appliquées à 3 formes de référence particulières : le cercle, le carré et le triangle.	132
5.2	Résultat produit par l'algorithme GICP où la forme de référence utilisée est le carré unité dans son repère propre et la transformation (a) une translation, (b) un déplacement rigide, (c) une similitude, (d) un étirement, (e) une affinité et (f) une homographie. On remarque que les données peuvent être tronquées de façon importante sans poser de problèmes à l'algorithme. En (a), les appariements entre la forme de référence et les données sont aussi montrés.	134
5.3	Classification obtenue sur les mêmes données avec la méthode de classification floue (a) qui traite mieux les points ambigus, et une méthode classique (b). On remarque en particulier que les centres des deux classes, marqués par des croix, sont mieux localisés par la méthode floue. Le chiffre au dessus point indique son degré d'appartenance à la classe du bas.	135

5.4	Quelques itérations de notre algorithme. Le résultat final est visible sur la figure 5.14 (a).	138
5.5	L'algorithme MMP interprète bien l'ensemble comme trois ellipses qui s'intersectent.	139
5.6	L'algorithme MMP interprète bien l'ensemble comme trois cubes malgré le bruit de fond représentant environ 20% des données.	140
5.7	En (b), l'algorithme MMP interprète bien les trois cubes malgré car le point d'accumulation de points parasites est considéré comme une primitive dégénérée. Alors que si l'on utilise la version robuste, l'algorithme n'est pas capable de séparer les points parasites qui représentent 20% des données.	141
5.8	Images gauches des quatre paires stéréoscopiques originales d'un jeu de Lego.	142
5.9	Les reconstructions du jeu de Lego obtenues à partir des points de vue de la figure 5.8 . Les vues ont été choisies avec des orientations telles que les reconstructions puissent s'assembler pour constituer visuellement la reconstruction complète.	143
5.10	(a) et (b) sont deux vues de la reconstruction complète obtenue en recalant les reconstructions de la figure 5.9 . La texture de la première image originale est retroprojetée sur cette reconstruction. (c) est une vue en fil de fer de cette même reconstruction. (d) montre les 3 parallélépipèdes qui recouvrent la reconstruction.	145
5.11	Les trois parallélépipèdes sont recalés avec la reconstruction complète du jeu de Lego, et sont vus approximativement selon les quatre lieux d'acquisition des images originales. La taille des primitives est légèrement réduite pour permettre une meilleure visualisation des contours reconstruits.	146
5.12	Les trois primitives de la figure 5.11 recalées sur la reconstruction sont vues selon les mêmes positions de prise de vue que celles de la figure 5.8 . De plus, la texture de la première image originale est partiellement retroprojetée sur les primitives afin d'observer la qualité du recalage.	147
5.13	Les points et les normales du modèle 3D d'une tête, sont modélisés comme la superposition de 15 ellipsoïdes par l'algorithme MMP (les ellipsoïdes ne sont pas découpées). Ces ellipsoïdes recouvrent de façon satisfaisante le haut (a)(f) et le bas (b) de l'arrière du crâne, le front (d), et les cotés (e)(c) de la tête.	148
5.14	L'ensemble des points est synthétique, mais il est bruité et incomplet. Les points sont sur un cercle et sur un parallélogramme échantillonnés. Cet ensemble est modélisé par l'algorithme MMP comme 2 parallélogrammes en (a), et comme 2 ellipses en (b).	149
5.15	(a) est la carte de profondeur originale, dont on peut voir deux vues 3D en (c) et (d). (b) est l'approximation par deux fonctions quadratiques (observer les courbures inverses de la région supérieure: front et nez) obtenue par l'approche MMP. (e) et (f) ont les mêmes points de vue que (c) et (d), mais elles présentent l'approximation quadratique.	152

5.16	(a) est la carte de profondeur originale d'un combiné de téléphone, dont une vue 3D est en (e). (b) est l'approximation par cinq fonctions quadratiques obtenue par la méthode MMP dont une vue 3D est en (f). (d) est l'image de labels finale et (c) est l'image de labels initiale qui est obtenue par segmentation avec des régions de profondeur constante.	153
5.17	Les images originales d'un visage avec différentes orientations sont sur la première ligne. Sur la deuxième ligne, ces images sont segmentées en 5 classes qui ont une variation quadratique de l'intensité. La troisième ligne présente les images de labels associées. Sur la quatrième ligne, la segmentation du visage vu de face, avec 5, 4, 3 et 2 classes est visible. Les images de label associés sont sur la dernière ligne.	155
5.18	(a) image originale en couleur. (b) image de labels de la segmentation avec 6 classes. (c) image couleur approximée linéaire.	156
5.19	Structure en 8 digital décrite par ses six sommets (a, b, c, d, e, f) et ses sept arrêtes ($a - b$), ($b - c$), ($c - d$), ($d - e$), ($e - f$), ($f - a$) et ($c - f$). Chaque sommet peut se déplacer en 2D, d'où 12 degrés de linerté.	159
5.20	Superposition, sur une image aérienne d'une résidence de la primitive articulée initiale (a) et finale (b). (c) et (d) montrent la même superposition sur les contours qui ont servi au recalage.	160
5.21	Superposition, sur une image aérienne d'une résidence de la primitive articulée initiale (a) et finale (b). (c) et (d) montrent la même superposition sur les contours qui ont servi au recalage.	161
5.22	(a) est la coupe originale du crâne qui présente trois contours: l'extérieur, l'intérieur du crâne, et le contour du cerveau. (b) montre l'initialisation par trois chaînes de 10 segments qui est relativement approximative. (c) est le résultat après recalage fin réalisé en parallèle.	162
A.1	Système d'acquisition d'images.	170
A.2	Profil de l'assombrissement aux bords d'une lentille en trou d'épingle de tirage $t_{objectif}$	171
A.3	Phénomène de vignettage d'un objectif. Suivant sa position, un point r de l'image ne voit pas la même portion du champ de vue.	172
A.4	Autocorrélation de deux disques pour calculer l'assombrissement dû au vignettage.	173
A.5	Dispositif optique réel de notre objectif Canon J15×9.5B 1:1.8 à focale variable extrait du brevet associé [Usu94]	173
A.6	Histogramme du bruit de transmission entre la caméra et la carte d'acquisition. L'allure est gaussienne avec un écart type de 1.5 niveau de gris.	175
A.7	Variation théorique et réelle de la variance du bruit en fonction du nombre d'images moyennées.	176
A.8	Histogramme pour les canaux Rouge, Vert et Bleu de l'écart de sensibilité de notre caméra CCD.	177
A.9	Image en fausse couleur des écarts de sensibilité $g_{sensibilité}(u, v)$ de la matrice CCD de notre caméra sur le canal Rouge.	177

A.10	Coupe de l'image de la figure A.9 après lissage.	178
A.11	Image en fausse couleur des variations d'éclairement $g_{objectif}(u, v)$ produit par notre objectif sur le canal Rouge.	179
A.12	Variation de l'éclairement avec la focale.	180
A.13	Vue avant (a) et après (b) correction radiométrique des gradients présents dans un ciel bleu près du zénith.	180
A.14	Coupe d'une image du ciel avant (a) et après (b) correction radiométrique.	181
A.15	(a) Détail de l'image d'un bâtiment. (b) coupe du toit de ce bâtiment avant et après correction radiométrique. Le lieu de la coupe est visible en noir sur le détail (a).	182
A.16	Image des contours avant (a) et après correction radiométrique (b) du détails de la figure A.15 (a). Des contours parasites sont éliminés grâce à la correction radiométrique.	182
A.17	Image de segmentation avant (a) et après correction radiométrique (b) du détails de la figure A.15 (a). Une région parasite apparaît sur l'image non corrigée.	183
B.1	Les données d'un rectangle nécessaires à la calibration	194
B.2	En blanc le modèle reconstruit, en noir les primitives utilisées lors des différentes calibrations. Les primitives (points, segments et rectangles) sont essentiellement les parties visibles du bureau et les 2 arêtes verticales du tableau utilisées pour les calibrations mixte et avec segments).	194
B.3	Caméra calibrée avec 6 points. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.	194
B.4	Caméra calibrée avec 8 points. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.	195
B.5	Caméra calibrée avec 9 segments. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.	195
B.6	Caméra calibrée avec 2 rectangles : le plateau et la face avant du bureau (Premier fichier 2D). Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.	195
B.7	Caméra calibrée avec 2 rectangles avec un deuxième fichier 2D. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.	195
B.8	Caméra calibrée avec 1 rectangle (plateau) et 4 segments (appartenant à la face avant du bureau). Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.	196
B.9	Caméra calibrée avec 1 rectangle (plateau) et 4 segments (dont 2 sont les arêtes verticales du tableau). Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.	196
B.10	Caméra calibrée avec 1 rectangle et 5 segments dont 4 sont verticaux. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.	196
B.11	Caméra calibrée avec 2 rectangles et 1 segment vertical. Le modèle 3D reconstruit est projeté sur l'image à travers la vue caméra.	196

B.12	Phase de construction: erreur de construction en profondeur sur le corps du bureau : sa position et ses dimensions paraissent correctes sur les vue caméra et perspective alors qu'elles sont fausses.	197
C.1	Liens entre les objets 3D, les paires d'images et les algorithmes manipulés. .	202
C.2	Image de segmentation originale (a) et segmentation induite par le modèle (b).	203
C.3	Sélection des points sur le contour de la région en fonction de la projection des coins de la facette 3D.	204
C.4	Modèle 3D utilisé pour la mise en correspondance de la scène de bureau. . .	206
C.5	Reprojection du modèle complet de la scène recalé à partir de la reconstruction (a), après la première étape de recalage fin (b) (2 appariements entre en face 3D et une régions), après la deuxième étape (c) (5 appariements), et après stabilisation (d) (8 appariements).	207
C.6	Superposition des modèles des 3 objets avec l'image originale (a) et segmentation induite par l'ensemble de ces objets (b).	208

Liste des tableaux

2.1	Erreur d'estimation du barycentre d'une ellipse dans une image suivant que l'on utilise ou pas les niveaux de gris.	66
2.2	Erreur induite par un bruit gaussien centré de variance σ sur l'estimation du centre d'une ellipse, suivant diamètre en pixels.	66
2.3	Écart d'estimation par rapport aux valeurs exactes avec 3 méthodes sur une mire de synthèse et avec une caméra qui présente uniquement des distorsions radiales.	68
2.4	Erreur relative d'estimation des paramètres de la caméra lors de la calibration suivant que l'on utilise ou pas les termes d'inertie.	69
2.5	Estimations obtenues avec trois méthodes différentes sur les mêmes points 2D.	71
2.6	Stabilité de l'estimation des paramètres intrinsèques suivant le modèle utilisé.	71
3.1	Cohérence des normales reconstruites sur l'objet de synthèse de la figure 3.9	91
3.2	Biais et erreurs moyennes sur la partie en correspondance et taille de la partie sans correspondance entre la mire de référence obtenue par calibration et la reconstruction 3D de la mire.	103
4.1	Déplacements obtenus par la phase de recalage grossier et les valeurs effectives de la figure 4.7 . La rotation est représentée par (p_1, p_2, p_3) qui sont les composantes de l'axe normalisé multiplié par l'angle en radian.	118
4.2	Biais et erreur sur la partie en recouvrement ainsi que la taille de la partie sans correspondance des points de la mire de la figure 4.11 (d)(e)(f).	122
B.1	Points 2D exprimés en pixels pour la calibration avec 2 rectangles : sur chaque point une différence de quelques pixels cf. figures B.6 et B.7 et tableaux B.2 et B.3	193
B.2	Paramètres de la caméra obtenus par calibration avec le nombre minimum de primitives: centre de l'image (u_0, v_0) , facteurs d'échelles (a_u, a_v) , translations (t_x, t_y, t_z) , rotation (r_x, r_y, r_z)	198
B.3	Paramètres de la caméra obtenus par calibration sur d'autres données.	198

Ph-D thesis of Paris IX-Dauphine University

Geometric Fitting and Model Matching in Automatic Modeling

Jean-Philippe TAREL

Summary

This dissertation deals with the development of a region-based stereo analysis system. This system produces a geometric and photometric model of polyhedral scenes.

First, each camera is calibrated using a calibration setup with an algorithm based on region features for a better accuracy. Both images of a stereo pair are segmented into clusters, and the resulting regions are matched two by two. Then, when camera geometry is rectified, the use of these region pairs as inputs allows to compute exactly the location and orientation of 3D patches whose projections on the image planes are assumed to be the two considered regions. Then, multiple-viewpoint reconstructions are robustly and accurately registered to obtain a complete geometric reconstruction. Finally, the resulting reconstruction is interpreted, with a more hi-level model, as the overlap of a set of deformed parallelepipeds. The algorithm developed allows to describe 2D and 3D objects by several deformed geometric prototypes in a very generic way.

During this research, I have noticed that many problems of automatic modeling are difficult to solve because of the overlap between geometric transformation and model matching estimations. Thus, I propose a methodology incorporating both geometric and matching estimations.

Keywords : Computer Vision, Image Analysis, Stereovision, Modeling of polyhedral objects, Pattern Recognition, Matching, Camera Calibration, Rectified Geometry, Image Segmentation, 3D Reconstruction in patches, Rigid and Nonrigid Registration, Fuzzy Clustering, Minimization.

Thèse de doctorat en sciences de l'université Paris IX-Dauphine

Estimation géométrique et appariement en modélisation automatique

Jean-Philippe TAREL

Résumé

Notre but est de concevoir un système de modélisation géométrique et photométrique d'objets polyédriques, à partir de plusieurs couples stéréoscopiques de cet objet.

La procédure que nous avons suivi consiste à calibrer chaque point de vue grâce à l'utilisation d'une mire, en s'appuyant sur des caractéristiques régionales pour avoir une plus grande précision. Chaque image est rectifiée, puis segmentée en régions dans lesquelles l'intensité est soit constante, soit varie linéairement, grâce à des algorithmes de segmentation fondés sur des méthodes de classification. Ensuite, les régions sont appariées entre les images gauche et droite. Chaque paire de régions permet de retrouver le support 3D de la face dont les régions sont les images en s'appuyant uniquement sur les moments géométriques de ces dernières. Alors, les reconstructions d'un même objet obtenues selon différents points de vue sont recalées l'une par rapport à l'autre, pour les fusionner ensemble. Puis, la reconstruction résultante est modélisée comme la composition de parallélépipèdes par un algorithme qui unifie les méthodes de classification géométrique et de recalage rigide ou non-rigide.

L'analyse de ces différentes étapes, nous a permis de reconnaître une difficulté commune à chacune de ces étapes, et nous a donc conduit à proposer une méthodologie pour analyser et poser un grand nombre de des problèmes que l'on peut rencontrer en modélisation automatique.

Mots-Clés : Vision par ordinateur, Analyse d'images, Stéréovision, Modélisation d'objets polyédriques, Reconnaissance des formes, Mise en correspondance, Calibration de caméras, Rectification, Segmentation d'images, Reconstruction 3D en faces, Recalage rigide et non rigide, Classification floue, Minimisation.

Thèse de doctorat en sciences de l'université Paris IX-Dauphine

Estimation géométrique et appariement en modélisation automatique

Jean-Philippe TAREL

Résumé

Notre but est de concevoir un système de modélisation géométrique et photométrique d'objets polyédriques, à partir de plusieurs couples stéréoscopiques de cet objet.

La procédure que nous avons suivie consiste à calibrer chaque point de vue grâce à l'utilisation d'une mire, en s'appuyant sur des caractéristiques régionales pour avoir une plus grande précision. Chaque image est rectifiée, puis segmentée en régions dans lesquelles l'intensité est soit constante, soit varie linéairement, grâce à des algorithmes de segmentation fondés sur des méthodes de classification. Ensuite, les régions sont appariées entre les images gauche et droite. Chaque paire de régions permet de retrouver le support 3D de la face dont les régions sont les images en s'appuyant uniquement sur les moments géométriques de ces dernières. Alors, les reconstructions d'un même objet obtenues selon différents points de vue sont recalées l'une par rapport à l'autre, pour les fusionner ensemble. Puis, la reconstruction résultante est modélisée comme la composition de parallélépipèdes par un algorithme qui unifie les méthodes de classification géométrique et de recalage rigide ou non-rigide.

L'analyse de ces différentes étapes, nous a permis de reconnaître une difficulté commune à chacune de ces étapes, et nous a donc conduit à proposer une méthodologie pour analyser et poser un grand nombre de des problèmes que l'on peut rencontrer en modélisation automatique.

Mots-Clés : Vision par ordinateur, Analyse d'images, Stéréovision, Modélisation d'objets polyédriques, Reconnaissance des formes, Mise en correspondance, Calibration de caméras, Rectification, Segmentation d'images, Reconstruction 3D en faces, Recalage rigide et non rigide, Classification floue, Minimisation.

ISBN - 2 - 7261 - 1002 - 9



* T U - 4 8 6 *