



HAL
open science

Géométrie d'un système de N caméras : théorie, estimation et applications

Stéphane Laveau

► **To cite this version:**

Stéphane Laveau. Géométrie d'un système de N caméras : théorie, estimation et applications. Interface homme-machine [cs.HC]. Ecole Polytechnique X, 1996. Français. NNT : . tel-00267257

HAL Id: tel-00267257

<https://pastel.hal.science/tel-00267257>

Submitted on 26 Mar 2008

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

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

THÈSE

présentée à

L'ÉCOLE POLYTECHNIQUE

pour obtenir

**Le GRADE de DOCTEUR EN SCIENCES
DE L'ÉCOLE POLYTECHNIQUE**

spécialité

INFORMATIQUE

par

Stéphane LAVEAU

Sujet de la thèse :

**GÉOMÉTRIE D'UN SYSTÈME DE N
CAMÉRAS. THÉORIE, ESTIMATION ET
APPLICATIONS.**

soutenue le 31 mai 1996 devant la Commission d'examen :

MM.	Olivier	FAUGERAS	Directeur de Thèse
	Martial	HEBERT	Rapporteurs
	Andrew	ZISSERMAN	
	Fabrice	CLARA	Examineurs
	Luc	VAN GOOL	
	Jean-Marc	STEYAERT	Président du Jury

à Angélique, mon poussin

Je remercie tout d'abord Olivier Faugeras, qui m'a encadré avec passion tout au long de cette thèse. La majorité des travaux exposés ici est le fruit de discussions avec lui.

Je suis reconnaissant à Martial Hébert de m'avoir guidé lors de mes séjours à Carnegie Mellon. Il m'a enseigné la rigueur et le bon sens scientifique. Il s'est aussi volontiers porté au travail de relecture de cette thèse.

Je remercie Andrew Zisserman pour son courage linguistique. Il a accepté d'être rapporteur de ma thèse (environ deux cents pages de français).

Je suis reconnaissant à Jean-Marc Steyaert, Luc Van Gool et Fabrice Clara d'avoir bien voulu faire partie de mon jury.

Je remercie madame Guibert de l'École Polytechnique et la DRET pour le soutien financier qu'ils m'ont apporté.

Enfin, je tiens à remercier tous mes collègues de bureau qui m'ont aidé, soutenu et fait rire pendant ces quatre dernières années : Fred pour son génial sens du hack, Théo pour sa persévérance à chercher les bugs du compilateur, Cyrille pour son optimisme de tous les jours, Jean-Luc pour sa bonne humeur constante, Reyes et les espagnoles pour leurs "Estefan", Luc pour ses nombreuses idées, Michel pour son programme de thèse que j'ai suivi à la lettre depuis le premier jour¹, ainsi que tous les autres...trop nombreux pour être tous cités ici. Je leur suis à tous reconnaissant pour les quatre années passées sous le soleil de Sophia.

Je ne remercie ni Continental Airlines ni Air Inter pour n'avoir jamais fait de geste généreux envers un excellent client, et ce malgré leurs promesses commerciales.

1. Si quelqu'un a conservé le texte original, qu'il me contacte!

Résumé

Cette thèse développe une approche basée sur la géométrie projective pour analyser et traiter des séquences d'images obtenues avec une caméra mobile. Les dérivations et démonstrations sont faites sans supposer qu'une information a priori est disponible sur les images, que ce soit sur le mouvement de la caméra ou sur leurs paramètres intrinsèques, comme la distance focale ou les points principaux. Il peut même s'agir de caméras différentes. La seule hypothèse est que la scène est rigide.

Le premier chapitre introduit le modèle mathématique de la caméra que nous utilisons, c'est à dire le modèle perspectif. À partir d'une formalisation en géométrie projective de la structure d'une caméra, nous dérivons les objets mathématiques nécessaires à la compréhension d'un système de deux caméras, c'est à dire la géométrie épipolaire, puis de trois caméras avec les trilinearités. Une théorie unificatrice est présentée, basée sur des déterminants, qui permet de généraliser les objets utilisés pour le cas de deux et de trois caméras au cas d'un nombre quelconque de caméras. Les liens entre ce formalisme et d'autres théories sont mis en évidence. Nous étudions aussi avec soin comment les objets peuvent se calculer les uns à partir des autres, en prenant soin de n'omettre aucun cas particulier.

Ces modèles auraient une utilité limitée si il n'existait pas de méthodes pour estimer cette géométrie. C'est l'objet du deuxième chapitre. Tout d'abord, nous présentons les algorithmes utilisés pour extraire des images les primitives d'intérêt que nous allons utiliser par la suite. Il s'agit de points et de droites vus dans plusieurs images. Nous distinguerons les procédés d'extraction de ces primitives des procédés de mise en correspondance. Nous abordons le problème de l'estimation de la géométrie des caméras sous plusieurs angles, à partir des matrices fondamentales représentatives de la géométrie épipolaire, à partir du tenseur trifocal, représentatif des trilinearités, ou bien directement, à partir des épipoles seulement. Cette estimation est faite par petits groupes de vues qui ont des primitives en commun. Il faut alors recoller les estimations partielles pour pouvoir exprimer toutes les matrices de projection dans la même base. Plusieurs méthodes sont possibles pour calculer ce changement de base. Nous en décrivons trois, que nous comparons à l'aide de tests statistiques. La géométrie globale peut ensuite être raffinée à l'aide de différents procédés itératifs. Toutes les méthodes sont testées sur des données synthétiques et sur des séquences réelles.

La deuxième partie de cette thèse présente l'application à un système de caméras d'une extension de la géométrie projective, la géométrie projective orientée. Nous introduisons tout d'abord les concepts

mathématiques nécessaires et les théorèmes qui en découlent. Nous explorons aussi les possibilités offertes par la géométrie projective orientée, comme la notion de gauche ou de droite, d'enveloppe convexe et de segment. L'application de ces outils à la vision conduit à l'orientation des caméras et des plans focaux de ces caméras. Tous les objets que nous avons étudiés précédemment possèdent maintenant une orientation. C'est le cas en particulier des matrices fondamentales et des épipoles. Les applications pratiques sont multiples. Nous décrivons comment cette méthode est employée pour supprimer des faux appariements, ou bien pour construire des enveloppes convexes d'objets, ou encore pour déterminer les parties visibles dans une nouvelle vue, ce qui nous sera utile par la suite. Nous montrons les liens qui existent entre cette théorie et une autre, différente, mais conduisant aux mêmes résultats : la chiralité.

La troisième partie de cette thèse montre diverses applications de ces algorithmes à des problèmes concrets.

Le premier chapitre traite des applications dans les images directement. L'exemple abordé est celui de la synthèse de nouvelles vues sans utiliser de modèles tridimensionnels. Cette méthode a l'avantage de permettre un rendu beaucoup plus réaliste des scènes puisqu'elle est basée sur les images et non pas sur des modèles de réflectances et de textures. Cette méthode est aussi avantageuse dans le cas de scènes changeant constamment et pour lesquelles la construction d'un modèle 3-D est inadaptée.

Le dernier chapitre est consacré aux méthodes d'obtention d'un modèle tridimensionnel de la scène à partir de vues non calibrées. L'estimation de la géométrie des caméras permet de simplifier grandement l'estimation de la structure de la scène. L'introduction de quelques contraintes supplémentaires, comme le parallélisme de quelques droites ou l'orthogonalité de quelques autres permet de transformer le modèle 3-D projectif en un modèle 3-D euclidien. Des expériences dans des conditions réelles sont présentées, pour la reconstruction d'un bâtiment à partir de différentes vues.

Table des matières

1	Introduction	1
1.1	Motivations	1
1.2	Contributions de cette thèse	3
1.2.1	Première partie	3
1.2.2	Deuxième partie	4
1.2.3	Troisième partie	4
I	Estimation de la géométrie projective d'un système de caméras	5
2	Géométrie d'un système de N caméras	7
2.1	Le modèle de la caméra	7
2.1.1	Le modèle sténopé	8
2.1.2	Autres modèles	9
2.1.2.1	Le modèle orthographique	9
2.1.2.2	Le modèle paraperspectif	10
2.1.2.3	Prise en compte de la distorsion	11
2.1.3	Formulation projective	12
2.1.4	Géométrie de N caméras	15
2.2	La stéréovision binoculaire	16
2.2.1	La matrice essentielle	16
2.2.2	La matrice fondamentale	19
2.3	Trois caméras	20
2.3.1	Les contraintes entre les trois matrices fondamentales	21

2.3.1.1	Cas où C_1 , C_2 et C_3 ne sont pas alignés . . .	21
2.3.1.2	Cas où C_1 , C_2 et C_3 sont alignés	22
2.3.2	Les trilinearités	23
2.4	Une théorie unificatrice: les déterminants	27
2.4.1	Brèves Notions sur l'algèbre de Grassmann-Cayley . . .	27
2.4.2	Les N -linéarités	30
2.4.2.1	Les bilinéarités	32
2.4.2.2	Les trilinearités	33
2.4.2.3	Les quadrilinearités	35
2.4.3	Trilinearités sur les droites	36
2.4.4	Différentes paramétrisations minimales	38
2.4.5	Lien avec les descriptions de Hartley et de Shashua . .	40
2.5	Relations entre les différentes entités	41
2.5.1	De \mathbf{F} vers \mathbf{P}	42
2.5.1.1	Cas où les épipoles sont disjoints	42
2.5.1.2	Cas où les épipoles sont confondus	46
2.5.2	De T vers \mathbf{P}	47
2.5.3	De 3 à N caméras	48
2.6	Conclusion	49
3	Estimation de la géométrie des caméras	51
3.1	Comment obtenir les appariements?	51
3.1.1	Les primitives d'intérêt utilisées: points et droites . . .	52
3.1.1.1	Les points	52
3.1.1.2	Les droites	53
3.1.2	Mise en correspondance	54
3.1.2.1	Corrélation	54
3.1.2.2	Relaxation	57
	Le critère de relaxation.	57
	Le score de relaxation.	58
	La maximisation.	59
3.2	Estimation de \mathbf{F}	59
3.2.1	Estimation des \mathbf{F}_{ij} paire par paire	59
3.2.2	Estimation de \mathbf{F} sous contrainte	61
3.3	Estimation de \mathbf{T}	62
3.3.1	Méthode linéaire	63
3.3.2	Méthodes non linéaires	64
3.4	Estimation de \mathbf{P}	65
3.4.1	Estimation de \mathbf{P} à partir de \mathbf{F}	65
3.4.2	Estimation de \mathbf{P} à partir de \mathbf{T}	67
3.4.3	Estimation directe de \mathbf{P}	68

3.4.4	Méthodes robustes	71
3.4.5	Raffinements non linéaires	72
3.4.5.1	Minimisation de Levenberg-Marquardt	73
	Itération de Newton	73
	L'itération de Levenberg-Marquardt	74
3.4.5.2	Ajustement des rayons optiques	74
3.4.5.3	Ajustement des lignes épipolaires	80
3.4.5.4	Ajustement des trilinearités	81
3.5	D'une description locale à une description globale	83
3.5.1	Changement de base en utilisant les matrices de projection	84
3.5.2	Changement de base en utilisant le nuage de points	84
3.5.3	Changement de base par recalage dans les images	85
3.5.4	Comparaison	85
3.6	Expériences et comparaison	92
3.6.1	Données synthétiques	92
3.6.1.1	Matrices de projection	92
3.6.1.2	Raffinement	98
3.6.2	Séquences réelles	102
3.6.2.1	La place des arcades à Valbonne	102
3.6.2.2	La bibliothèque de l'INRIA	106
3.7	Conclusion	106

II Géométrie projective orientée appliquée à la vision par ordinateur 111

4	Géométrie projective orientée 113
4.1	Principes de bases 114
4.1.1	Géométrie projective classique et orientée 114
4.1.2	Orientation 117
	4.1.2.1 Approche mathématique 117
	4.1.2.2 Exemples de sous espaces orientés 117
4.1.3	Propagation de l'orientation 118
4.1.4	Droite et gauche, devant et derrière, enveloppes convexes 118
4.2	Application à la vision 119
4.2.1	Plan focal orienté 119
4.2.2	Épipoles et matrices fondamentales orientées 122
4.2.3	Enveloppes convexes, deux approches 124
4.3	Implications 125
4.3.1	Reconstructions possibles et impossibles 125

4.3.1.1	Théorie	125
4.3.1.2	Algorithme	127
4.3.2	Sens de parcours des épipolaires	129
4.4	Liens avec la chiralité de Richard Hartley	132
4.4.1	χ détermine si un point est devant ou derrière une caméra	132
4.4.2	Toutes les reconstructions projectives ne sont pas possibles	133
4.4.3	Visibilité	134
4.5	Conclusion	134
III Applications		137
5	Applications	139
5.1	Synthèse de nouvelles vues sans modèle tridimensionnel	139
5.1.1	Intérêt de la démarche	139
5.1.2	Transfert de primitives d'intérêt	141
5.1.3	Transfert d'image	143
5.1.3.1	A l'aide de la géométrie épipolaire	146
5.1.3.2	A l'aide des trilinearités	147
5.1.4	Résultats	148
5.1.4.1	Avec correspondances denses	148
5.1.4.2	Avec correspondances éparses	148
5.2	Construction de modèles 3-D à partir de séquences d'images .	156
5.2.1	Calcul de l'homographie de retour à l'eulidien	158
5.2.1.1	Du projectif à l'affine	158
5.2.1.2	De l'affine vers l'eulidien	161
5.2.2	Modélisation dans les images	162
5.2.3	Reconstruction	166
6	Conclusion	169
6.1	Contributions et résultats	169
6.1.1	Géométrie des caméras	169
6.1.2	Géométrie projective orientée	170
6.1.3	Applications	170
6.2	Extensions	170
6.2.1	Géométrie des caméras	170
6.2.2	Transfert et synthèse de nouvelles vues	171
6.2.3	Construction de modèles tridimensionnels	171

A	Les algorithmes d'estimation robuste	173
A.1	Moindre médiane des carrés	173
B	Compléments de géométrie projective appliquée à la vision	175
B.1	Les plans de l'espace correspondent à des homographies de l'image	175
B.2	Comment faire un barycentre projectif?	176
B.3	Comment reconstruire des points de \mathcal{P}^3 ?	178
B.3.1	La méthode linéaire	178
B.3.2	La méthode non linéaire	178
C	Calculs relatifs à l'ajustement de rayons	181
D	Notations utilisées	183

Table des figures

2.1	Modèle sténopé. Projection d'un objet de l'espace 3-D sur le plan rétinien.	8
2.2	Comparaison de trois différents modèles utilisés en vision par ordinateur : orthographique, paraperspectif et perspectif. Le même objet est projeté de 3 manières différentes. Les rayons optiques sont indiqués en pointillés.	10
2.3	La géométrie épipolaire. C_1 et C_2 sont les centres optiques. \mathbf{e}_{21} et \mathbf{e}_{12} sont les épipoles.	18
2.4	Faisceau de droites épipolaires	18
2.5	6 épipoles dans un plan. \mathbf{l}_1 , \mathbf{l}_2 et \mathbf{l}_3 représentent la trace du plan trifocal dans les images.	22
2.6	6 épipoles sur une droite	23
2.7	Interprétation géométrique d'une trilinearité. La condition est que la droite passant par C_1 et \mathbf{m}_1 , le plan $\Pi(\mathbf{5} \wedge \mathbf{6}, \mathbf{m}_2)$ construit à partir de la droite $\mathbf{5} \wedge \mathbf{6}$ et du point \mathbf{m}_2 , et le plan $\Pi(\mathbf{8} \wedge \mathbf{9}, \mathbf{m}_3)$ se coupent en un point de l'espace.	34
2.8	Les épipoles vérifient les trilinearités avec un point quelconque. En effet, \mathbf{e}_{21} et C_2 , \mathbf{e}_{31} et C_3 définissent des faisceaux de plans passant par C_1 . Tous les rayons issus du centre optique C_1 coupent donc ces deux faisceaux de plans en un point unique qui est le centre optique de la première caméra.	39
3.1	Si les deux points m_1 de l'image I_1 et m_2 de l'image I_2 sont en correspondance, la valeur de la fonction d'intensité de I_1 dans un voisinage V_1 de m_1 est corrélée à celle de la fonction d'intensité de I_2 dans un voisinage V_2 de m_2	55

3.2	Distance épipolaire entre \mathbf{m}_i et $\mathbf{F}_{ij}^T \mathbf{m}_j$ et entre \mathbf{m}_j et $\mathbf{F}_{ij} \mathbf{m}_i$	61
3.3	Un sous ensemble de matrices fondamentales permettant d'obtenir l'ensemble de la géométrie épipolaire. Chaque trait représente une matrice fondamentale, chaque point une caméra. Cet ensemble contient $2N - 1$ matrices fondamentales.	62
3.4	Critère à minimiser pour obtenir des trilinearités de manière non linéaire. Les points \mathbf{m}_1 et \mathbf{m}_2 engendrent 9 droites dans la troisième image.	65
3.5	Distance entre deux matrices fondamentales. Cette distance est estimée sur la zone de l'espace qui est visible dans les deux images. Des points $(\mathbf{m}_1, \mathbf{m}_2)$ vérifiant la géométrie épipolaire de la première matrice fondamentale \mathbf{F} sont tirés au hasard. On vérifie leur adéquation à la deuxième matrice fondamentale \mathbf{F}'	68
3.6	Matrice jacobienne creuse par blocs. Les parties noires représentent les éléments nuls, les parties blanches représentent les éléments non nuls. Par exemple, la première ligne correspond à ε_{11} et ne dépend que des variables \mathbf{P}_1 et \mathbf{M}_1 , \mathbf{m}_{11} étant fixé. Les dérivées partielles de ε_{11} par rapport aux autres variables sont nulles.	76
3.7	Équations normales par blocs	77
3.8	Critère d'ajustement des droites épipolaires	81
3.9	Droites engendrées dans la troisième image par un tenseur du premier groupe lorsque l'on connaît \mathbf{m}_1 et \mathbf{m}_2 . Les points \mathbf{m}_1 et \mathbf{m}_2 sont présentés ici en situation générale, ils ne vérifient pas la contrainte épipolaire existant entre les images 1 et 2.	82
3.10	Évolution des résultats des différents algorithmes lorsque le nombre de points varie. Le bruit est ici de 0.3 pixel. Les erreurs de la méthode (P) ont été divisées par 10^3 . La distance est la distance angulaire dans \mathcal{P}^{15}	87
3.11	Évolution des résultats des différents algorithmes lorsque le nombre de points varie. Le bruit est ici de 0.3 pixel. Les erreurs de la méthode (P) a été divisées par 10^2 . La distance est la distance euclidienne 3-D.	88
3.12	Évolution des résultats des différents algorithmes lorsque le nombre de points varie. Le bruit est ici de 0.3 pixel. Les erreurs de la méthode (P) a été divisées par 10^2 . La distance est la distance pixel dans les images.	88

3.13	Évolution des résultats des différents algorithmes lorsque le bruit varie. La distance présentée ici est la distance angulaire. Les résultats ont été obtenus avec 100 points. Les erreurs des algorithmes (Mini) et (SVD) ont été multipliées par 10^3 pour obtenir le même ordre de grandeur.	89
3.14	Évolution des résultats des différents algorithmes lorsque le bruit varie. La distance présentée ici est la distance euclidienne de l'espace. Les résultats ont été obtenus avec 100 points. L'erreur de l'algorithme (P) a été multipliée par 10^{-2} pour obtenir le même ordre de grandeur.	90
3.15	Évolution des résultats des différents algorithmes lorsque le bruit varie. La distance présentée ici est la distance entre les points dans les images. Les résultats ont été obtenus avec 100 points. L'erreur de l'algorithme (P) a été multipliée par 10^{-2} pour obtenir le même ordre de grandeur.	91
3.16	Comparaison des différentes méthodes pour 3 vues : La mesure d'erreur utilisée est la distance α . Les abscisses représentent le bruit sur les points.	93
3.17	Comparaison des différentes méthodes pour 3 vues : La distance utilisée est la distance DR. Les abscisses représentent le bruit sur les points.	94
3.18	Comparaison des différentes méthodes pour 3 vues : La distance utilisée est la distance DE. Les abscisses représentent le bruit sur les points.	94
3.19	Comparaison des différentes méthodes pour 6 vues : La distance utilisée est la distance α . Les abscisses représentent le bruit sur les points.	95
3.20	Comparaison des différentes méthodes pour 6 vues : La distance utilisée est la distance DR. Les abscisses représentent le bruit sur les points.	95
3.21	Comparaison des différentes méthodes pour 6 vues : La distance utilisée est la distance DE. Les abscisses représentent le bruit sur les points.	96
3.22	Comparaison des méthodes directes pour 6 vues : La distance utilisée est la distance DE. Ce graphe est le même que le précédent, mais l'échelle est différente et seulement les deux meilleures méthodes sont présentées.	97
3.23	Comparaison des méthodes de raffinement pour 6 vues. La distance utilisée ici est la distance α . Les abscisses représentent le bruit sur les points.	99

3.24	Comparaison des méthodes de raffinement pour 6 vues. La distance utilisée ici est la distance DR. Les abscisses représentent le bruit sur les points.	100
3.25	Comparaison des méthodes de raffinement pour 6 vues. La distance utilisée ici est la distance DE. Les abscisses représentent le bruit sur les points.	101
3.26	Détails de la géométrie épipolaire calculée sur la place des Arcades. Trois des huit images sont présentées. Elles appartiennent à trois blocs différents (deuxième, cinquième et septième images)	103
3.27	Détails de la géométrie épipolaire calculée sur la place des Arcades. Quatre des huit images sont présentées. Le détail est le sommet d'une antenne de télévision.	104
3.28	Vue d'ensemble de la place. Des droites épipolaires sont tracées en certains points. On remarque que les droites épipolaires se coupent en des points uniques. Ces points sont en correspondance entre les images.	105
3.29	Cinq images de la bibliothèque de l'INRIA. Des droites épipolaires sont tracées entre les images à certains points.	107
3.30	Détails des droites épipolaires tracées sur l'image autour d'un point caractéristique. Le point est ici le coin inférieur droit de la façade de la bibliothèque.	108
4.1	Représentation sphérique de \mathcal{P}^2 et \mathcal{T}^2	115
4.2	Passage de la représentation sphérique à la représentation planaire pour \mathcal{P}^1	115
4.3	Représentation planaire de \mathcal{P}^2 et \mathcal{T}^2	116
4.4	Droite orientée dans \mathcal{T}^2	118
4.5	Une caméra orientée. Elle est représentée par un plan orienté, qui se décompose en deux plans (celui de devant et celui de derrière). Ces deux plans ont une droite en commun, qui représente la ligne à l'infini de ces plans.	121
4.6	Une "tranche" de caméra orientée présentée suivant le modèle sphérique. \mathbf{C} et $\neg\mathbf{C}$ sont les centres optiques possibles. La ligne en gras représente l'intersection avec le plan focal Π_f , la ligne fine le rayon optique dessiné ici au départ de \mathbf{C}	121
4.7	Le plan épipolaire de \mathbf{M} représenté suivant le modèle sphérique. Les lignes \mathbf{I}_1 et \mathbf{I}_2 représentent les intersections des plans focaux avec le plan épipolaire de \mathbf{M}	123
4.8	Division de \mathcal{T}^3 en 4 zones : devant et derrière les plans focaux.	126

4.9	Évolution des zones avec le nombre de caméras. L'exemple présenté ici est avec 4 caméras.	127
4.10	Faux appariements détectés grâce à l'orientation. Le point correspondant dans l'espace est reconstruit derrière les plans focaux.	128
4.11	Deux points de l'espace confondus dans une des caméras.	130
4.12	Changement d'orientation lorsque le centre optique de la deuxième caméra traverse le plan focal de la première caméra. Il est noté C_2 et C'_2 suivant la configuration. L'épipole correspondant est noté e_{12} et e'_{12}	130
4.13	2 images synthétisées avec des caméras différant uniquement par l'orientation. L'image de gauche présente des anomalies sur le coté qui sont dues au fait que le sein droit du mannequin est vu par derrière. L'objet n'est pas complet parce que les images sources ne le couvrent pas entièrement.	131
4.14	La caméra qui regardait à travers l'infini. Les rayons traversent le plan à l'infini pour se retrouver de l'autre coté de la caméra.	132
4.15	Position possible du plan à l'infini par rapport aux plans focaux.	134
5.1	Transfert épipolaire simple.	141
5.2	Comparaison de méthodes de prédiction (droite : avec instabilités dans le plan trifocal, gauche : sans instabilités).	142
5.3	Transfert épipolaire à proximité du plan trifocal : vue 2-D.	143
5.4	Transfert épipolaire à proximité du plan trifocal : vue 3-D.	144
5.5	Problème de discrétisation entre deux images.	145
5.6	Présence de trous dans l'image synthétisée. Les trous sont dus au fait qu'un balayage régulier de la scène dans une des images de départ ne garantit pas un balayage régulier de la scène dans l'image synthétisée. On voit donc apparaître des irrégularités.	145
5.7	Solutions multiples. Il existe plusieurs points de la scène sur le même rayon.	146
5.8	Les droites épipolaires dans les images sources ainsi que l'image par les cartes de disparité de la droite épipolaire dans la première image. Deux points de la scène se trouvent sur ce rayon. Seul le point M^a est visible.	147
5.9	Synthèse d'images sans modèle tridimensionnel. Les deux images sources sont présentées au dessus, l'image destination en dessous. Cette expérience est particulièrement convaincante car la position de l'image destination est très différente de la position des images initiales, exacerbant ainsi les défauts possibles.	149
5.10	Configuration utilisée pour l'acquisition d'images	150

5.11	Les 6 images sources utilisées pour calculer les nouvelles images. Elles sont prises de 2 points de vue différents, avec à chaque fois trois orientations différentes. Les 3 premières photos ont été acquises d'un point de vue, les trois suivantes d'un autre point de vue. Les images ont été acquises avec un appareil photo standard.	151
5.12	Les deux images mosaïques engendrées à partir des 6 images précédentes. Les zones noires représentent des parties visibles dans aucune des images.	152
5.13	4 images extraites d'une séquence de 20. Le mouvement est une interpolation entre les mosaïques.	153
5.14	5 images extraites d'une séquence de 50. Le mouvement est un mouvement de translation vers l'avant à partir de la première caméra.	154
5.15	La contrainte d'ordre n'est pas vérifiée. En effet, les points \mathbf{m}_i^a et \mathbf{m}_i^b n'apparaissent pas dans le même ordre dans les deux images.	155
5.16	Triangulation dans l'image.	156
5.17	Ensemble de droites parallèles détectées dans une image.	160
5.18	Reconstruction affine d'une scène, ici des toits d'un village suisse. Vue de haut (droite) et vue de côté (gauche). La scène est très distordue, mais le parallélisme est conservé. La base a été choisie pour que le plan à l'infini ait $T = 0$ pour équation. L'origine est un des points de la scène et les trois directions ont été choisies non coplanaires mais ne représentent rien de particulier.	161
5.19	Droites perpendiculaires sélectionnés pour calculer la transformation de la reconstruction affine vers la reconstruction euclidienne.	163
5.20	Reconstruction de la scène après la minimisation des paramètres de retour à l'euclidien. Vue de dessus (droite), de côté (gauche) et de biais (bas). La scène est désormais reconnaissable, les angles et les rapports de distance sont préservés ainsi que le parallélisme.	164
5.21	Un point dans une image (à gauche) contraint ses correspondants à se trouver sur sa droite épipolaire (à droite)	165
5.22	Le choix d'un deuxième point (contraint sur une droite épipolaire) détermine sans ambiguïté la position de ses correspondants dans toutes les autres images.	165
5.23	Vue de dessus de la bibliothèque de l'INRIA. Vue en fil de fer.	166
5.24	Vue de côté de la bibliothèque de l'INRIA. Vue en fil de fer.	167

5.25	Vue de dessus de la bibliothèque de l'INRIA. Image synthétique.	167
5.26	Vue de coté de la bibliothèque de l'INRIA. Image synthétique.	168
B.1	Le barycentre projectif. Nous présentons ici un exemple de barycentre entre trois points de \mathcal{S}^{n+1} . Les vecteurs sont tous unitaires.	177

Chapitre 1

Introduction

1.1 Motivations

Les images sont de plus en plus présentes dans notre environnement. La demande est forte pour la création d'images, que ce soit pour la fiction, pour la simulation ou bien pour le jeu. Ces images peuvent être créées de manière cinématographique, ou bien à partir de dessins, ou bien à partir d'ordinateurs. Ces images informatiques nouvelles peuvent être soit le traitement d'images réelles, soit une invention complète de l'ordinateur (et de son maître).

Pour que ces images soient visuellement correctes, elles doivent issues de la géométrie tridimensionnelle de la scène. La principale difficulté est alors de déterminer les paramètres des images sources dans la scène, qui sont parfois disponibles, mais le plus souvent inconnues.

Comprendre et calculer la géométrie d'une scène à partir d'images est un problème extrêmement difficile (et n'ayant pas toujours toujours une solution unique). Plusieurs approches existent, notamment en stéréovision binoculaire [Marr et Poggio, 1977, Dhond et Aggarwal, 1989, Hotz, 1991, Nishihara et Poggio, 1983, Nishihara et Poggio, 1984], trinoculaire [Ayache et Lustman, 1987a, Ito et Ishii, 1986b, Kitamura et Yachida, 1990, Pietikainen et Harwood, 1987] et multi-baseline [Ohta et Kanade, 1985, Milenkovic et Kanade, 1985, Kanade et Okutumi, 1990]. D'autres méthodes sont basées sur l'estimation de la structure à partir du mouvement comme [Aggarwal et Nandhakumar, 1988, Huang et Netravali, 1994, Jerian et Jain, 1991, Weng et al., 1992, Lee et al., 1995].

Malheureusement, toutes ces méthodes supposent connus les paramètres

optiques de la caméra. Or, de nombreux facteurs peuvent changer la valeur de ces paramètres : il peut s'agir d'un changement de focale pour permettre de sélectionner un détail de la scène [Enciso et al., 1994, Viéville et al., 1994, Viéville, 1995, Aloimonos et al., 1988, Pahlavan et al., 1993], il peut s'agir d'une mise au point [Krotkov, 1989, Krotkov et Bacjcsy, 1993, Das et Ahuja, 1993].

Ces paramètres peuvent être mesurés à l'aide d'un laser [Willson, 1994, Willson et Shafer, 1993], mais, habituellement, on se sert de l'analyse d'une ou plusieurs images d'une mire dite de calibration. Cette mire peut être fixe ou en mouvement. C'est généralement un cube, un dièdre [Robert, 1993] ou un plan que l'on déplace [Tsai, 1986, Lenz et Tsai, 1987, Tsai, 1987, Tsai, 1989, Lenz et Tsai, 1988]. Malheureusement, obtenir des images de cette mire est parfois malcommode, comme pour les applications sous-marines, mais parfois aussi impossible, comme par exemple pour les applications spatiales et planétaires, où le transport de tels objets présente un coût prohibitif. D'une manière plus générale, cette calibration rend le système plus difficilement utilisable par des non spécialistes, de par sa relative complexité. Il existe toutefois des logiciels commerciaux utilisant une calibration approximative comme *PhotoModeller*.

De plus, l'utilisateur veut parfois avoir un contrôle actif de son système de caméras. Il peut vouloir se concentrer sur des zones de l'image particulièrement intéressantes, ou bien déplacer son champ de vision sur des objets plus proches ou plus lointains et faire une mise au point de l'image. Ces systèmes de vision active [Ma et Olsen, 1990, Pahlavan et al., 1993] ont beaucoup de mal à conserver une calibration au cours du temps.

Pis encore, on veut pouvoir traiter des images qui ont été prises avec une caméra ou un appareil photo dont la calibration n'est pas disponible. Il s'agit de films préexistants, de photos prises par des non spécialistes, de vidéos amateur. Pour toutes ces applications, la calibration est simplement impossible.

Nous allons tenter d'apporter une réponse à ces problèmes dans cette thèse en prenant le parti de ne pas calibrer nos images. Bien sûr, ce choix va entraîner de nombreux problèmes à résoudre. Ce parti pris pour la simplicité d'acquisition va entraîner des difficultés de traitement. En ayant moins d'informations a priori sur la géométrie du système, les résultats possibles seraient moins intéressants. Il va donc falloir que nous obtenions cette information manquante pour faciliter le travail de l'utilisateur.

Tout d'abord, nous allons nous demander quelle est l'information maximale que nous pouvons extraire des images sans calibration. La réponse dépendra bien sûr du nombre de caméras et de leur configuration géométrique,

comme nous le verrons au cours du premier chapitre. Il nous faudra ensuite estimer cette information géométrique, ce que nous ferons au deuxième chapitre. Nous présenterons diverses méthodes ainsi qu'une comparaison basée sur des données exactes ainsi que sur des données réelles. Notre modèle serait malgré tout incomplet si nous n'introduisons pas la notion d'orientation au chapitre 3. En effet, si les notions de devant la caméra et de derrière la caméra sont évidentes lorsque l'on est calibré, elle deviennent beaucoup plus délicates à appréhender sans cette hypothèse. Cette information est pourtant cruciale pour plusieurs applications. Ces applications seront présentées au chapitres 4 et 5. Nous montrerons qu'il est possible de synthétiser de nouvelles vues sans avoir à reconstruire un monde tridimensionnel, de construire une mosaïque à partir de plusieurs images. Les domaines d'utilisation de ces méthodes sont extrêmement nombreux : on imagine un jeu vidéo de courses de voitures où les images ne sont pas synthétiques, mais des images réelles d'un circuit existant. On imagine un développement panoramique de photos d'un paysage pris en plusieurs fois. On imagine un catalogue de vente par correspondance sur votre télévision branché sur une simple ligne de téléphone. Certaines de ces applications sont déjà au stade commercial comme par exemple *QuickTime VR* d'Apple.

Nous montrerons dans un deuxième temps qu'il est possible de reconstruire une scène à partir de différentes vues et ceci sans calibration. Enfin, nous montrerons comment il est possible de calibrer un système de caméras interactivement sans mire, mais avec des images d'une scène quelconque. Là encore, les domaines d'utilisations sont nombreux. On voudrait reconstruire une ville détruite pendant la guerre à partir d'un film. On voudrait construire facilement des modèles tridimensionnels pour l'animation, simplement à partir d'images de ces objets. On voudrait montrer aux gens une cuisine dans leur appartement virtuel obtenu à partir de photos de famille. Des programmes commerciaux existent pour certaines de ces applications, avec toutefois des conditions plus restrictives. Ces logiciels sont développés par des sociétés issues de la photogrammétrie comme *PhotoModeller*.

1.2 Contributions de cette thèse

*Dicunt ei Cæsaris tunc ait illis reddite ergo quæ sunt Cæsaris
Cæsari et quæ sunt Dei Deo.*

1.2.1 Première partie

La section 2.1 est enseignée dans tous les cours de vision par ordinateur. Les personnes y ayant contribué sont trop nombreuses pour être nommées ici. Ce sont principalement des modèles utilisés par les photogrammètres qui ont été repris en vision par ordinateur. La section 2.2 présente principalement le travail de Thomas Huang et Olivier Faugeras ainsi que des parties de la thèse de Quang-Tuan Luong. Je suis responsable pour la première partie de la section 2.3, tandis que la deuxième partie reflète le travail d'Amnon Shashua et de Richard Hartley. La théorie unificatrice des déterminants est l'œuvre d'Olivier Faugeras, de Bernard Mourrain ainsi que de Bill Triggs. Le passage d'un formalisme à un autre correspond à ma recherche.

Les méthodes d'extraction de primitives d'intérêt ainsi que de mise en correspondance sont une compilation des idées et méthodes de Thierry Blaszkowski, Cyril Zeller et Rachid Deriche. L'estimation de \mathbf{F} pour une paire d'images est le travail de Quang-Tuan Luong et de Zhengyou Zhang. L'estimation de T a été réalisée par mes soins à partir d'idées de Richard Hartley. J'ai construit les méthodes exposées à la section 3.4 mis à part l'ajustement de rayons emprunté à la communauté de photogrammétrie.

1.2.2 Deuxième partie

La théorie de la géométrie projective orientée a été construite au siècle dernier. Son introduction en vision computationnelle est due à Jorge Stolfi. Les implications pour la vision par ordinateur ont mûri après de nombreuses discussions avec Olivier Faugeras. Des théories similaires ont été développées par Richard Hartley dans un formalisme différent.

1.2.3 Troisième partie

La synthèse d'images sans modèle tridimensionnel et la compression d'image non destructive est une nouveauté apportée par cette thèse. Les applications de ce domaine sont probablement très vastes, et beaucoup reste encore à faire.

Les applications tridimensionnelles ont été développées au sein d'un projet Européen : REALISE, en collaboration avec Thomson-Sysec, l'université d'Utrecht et le Fraunhofer Institute à Darmstadt. Le travail présenté est partiellement dû à Luc Robert et Cyrille Gauclin.

Première partie

Estimation de la géométrie
projective d'un système de
caméras

Chapitre 2

Géométrie d'un système de N caméras

Dans ce chapitre, nous décrivons d'abord le formalisme qui nous permet de représenter chaque caméra séparément, puis des systèmes de deux et trois caméras, et enfin un ensemble quelconque de caméras. Le formalisme mathématique sous-jacent est celui de la géométrie projective, particulièrement adapté comme nous allons le voir.

2.1 Le modèle de la caméra

L'objet de cette section est de décrire quels modèles de caméras existent et sont appropriés à nos applications et quels modèles de caméras nous allons utiliser. Le modèle de caméra représente le mécanisme mathématique et physique qui permet de produire une image à partir d'une scène tridimensionnelle. Dans tous les cas, il s'agit d'une projection au sens mathématique du terme de l'espace vers le plan rétinien. Nous privilégierons le modèle perspectif ou modèle sténopé pour des raisons qui seront développées plus loin.

Le lecteur trouvera ici peu d'informations sur l'aspect photométrique de la formation des images sur une rétine. Cet aspect est occulté par le fait que nous utilisons des primitives images ou primitives d'intérêt qui ont été extraites à une étape antérieure des calculs. L'extraction et la mise en correspondance de primitives impliquent des hypothèses très précises sur la photométrie, mais ces aspects sortent du champ de nos investigations. Nous reparlerons cependant un tout petit peu de la réflectance au chapitre 5.1.3.

2.1.1 Le modèle sténopé

Ce modèle que nous avons choisi est aussi le plus couramment utilisé. Une caméra est représentée par un plan rétinien Π^r et un centre optique C qui n'appartient pas à Π^r . L'image d'un point M de l'espace est la trace du rayon $\langle C, M \rangle$ sur le plan Π^r (voir figure 2.1). Le plan parallèle à Π^r et passant par C est appelé plan focal et noté Π^f . La distance focale f est Cc où c est la projection orthogonale de C sur Π^r . c est aussi appelé point principal.

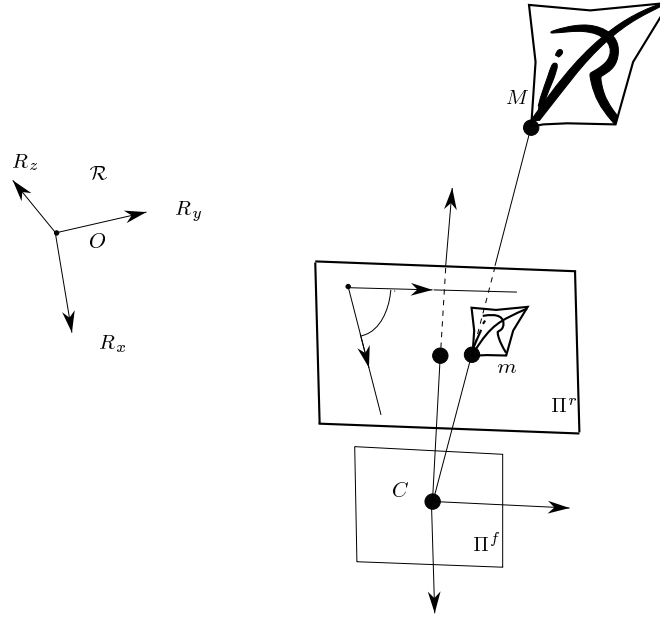


FIG. 2.1 – *Modèle sténopé. Projection d'un objet de l'espace 3-D sur le plan rétinien.*

Soit $\mathcal{R}(O, R_x, R_y, R_z)$ un repère orthonormé de l'espace et $(o, \mathbf{u}, \mathbf{v})$ un repère affine du plan Π^r . L'expression analytique de la projection se décompose en trois parties distinctes:

- Un changement de repère de l'espace.
- Une projection.
- Un changement de repère dans le plan.

Pour un point M donné de l'espace représenté par le vecteur $\mathbf{M}_r(X_r, Y_r, Z_r)$ dans le repère \mathcal{R} , ses coordonnées dans le repère associé au centre optique $\mathcal{R}_C(C, C_x, C_y, C_z)$ (avec C_z orienté suivant Cc) sont $\mathbf{M}_c(X_c, Y_c, Z_c) = \mathbf{R}\mathbf{M}_r +$

\mathbf{t} , \mathbf{R} et \mathbf{t} décrivent le changement de repère entre \mathcal{R} et \mathcal{R}_C . La projection sur Π^r s'obtient en divisant X_c et Y_c par Z_c/f . Le passage du repère induit par \mathcal{R}_C sur Π^r au repère (o, u, v) se fait de la manière suivante:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \alpha_u & -\alpha_u \cot \theta \\ 0 & \alpha_v \sin \theta \end{pmatrix} \begin{pmatrix} x_c \\ y_c \end{pmatrix} + \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} \quad (2.1)$$

θ est l'angle entre les vecteurs \mathbf{u} et \mathbf{v} . α_u et α_v sont les facteurs d'échelle sur les axes. Ils représentent la distance focale exprimée en unités pixel horizontalement et verticalement. u_0 et v_0 sont les coordonnées de c dans le repère (o, u, v) .

En pratique, θ est toujours proche de $\pi/2$ [Vaillant, 1990], α_u et α_v sont donnés par les constructeurs pour les caméras CCD. Le rapport α_u/α_v varie peu [Faugeras, 1993] lorsque l'on change la distance focale.

La caméra est donc représentée par \mathbf{R} , \mathbf{t} , α_u , α_v , θ , u_0 , v_0 . \mathbf{R} et \mathbf{t} sont appelés les paramètres extrinsèques puisqu'elles ne dépendent que de la position de la caméra dans l'espace. \mathbf{R} dépend de 3 paramètres, \mathbf{t} de 3. Les autres paramètres sont appelés les paramètres intrinsèques puisqu'ils ne dépendent que de la caméra.

La description de l'opération de projection peut grandement être simplifiée en utilisant les coordonnées homogènes comme nous le verrons à la section 2.1.3.

2.1.2 Autres modèles

Des modèles différents existent (voir figure 2.2). Ils ont pour certains l'avantage de la simplicité, comme le modèle orthographique ou le modèle paraperspectif. D'autres brillent par leur complétude et une meilleure prise en compte de l'optique de la caméra, au prix de paramètres additionnels. Une comparaison de ces différents modèles sans distorsion peut être trouvée dans [Horaud et al., 1994] par exemple.

2.1.2.1 Le modèle orthographique

Le modèle de projection orthographique est simplement un modèle perspectif dont le centre optique est à l'infini. Les rayons optiques arrivent donc tous parallèlement sur le plan rétinien. Ce modèle a permis de nombreux travaux en structure à partir du mouvement [Tomasi et Kanade, 1991, Ullman et Basri, 1991], mais les auteurs, conscients du manque de réalisme de ce modèle, ont tentés d'étendre leurs théories à la perspective [Basri, 1993] ou à la paraperspective [Poelman et Kanade, 1994, Poelman et Kanade, 1993].

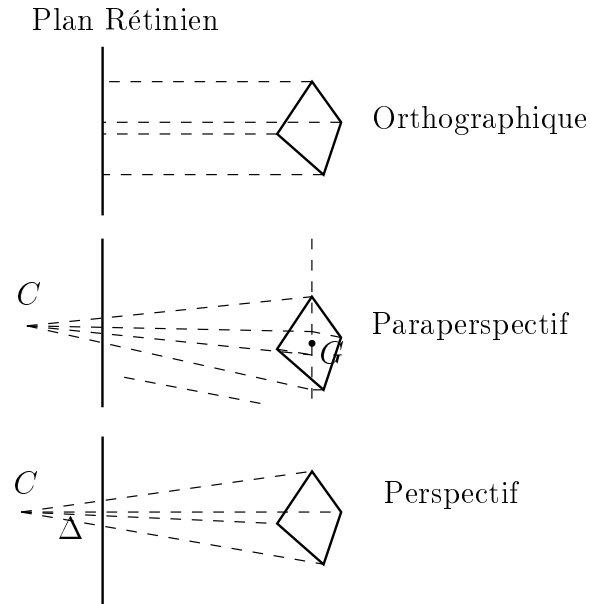


FIG. 2.2 – Comparaison de trois différents modèles utilisés en vision par ordinateur : orthographique, paraperspectif et perspectif. Le même objet est projeté de 3 manières différentes. Les rayons optiques sont indiqués en pointillés.

2.1.2.2 Le modèle paraperspectif

Le modèle paraperspectif est apprécié en vision par ordinateur car il est bien plus complet que le modèle orthographique. Il tient compte du facteur d'échelle, de l'apparence d'objets à la périphérie de la caméra.

Pour une projection paraperspective, on détermine tout d'abord un plan médian de la scène qui est parallèle au plan rétinien de la caméra. On choisit un point G le plus au centre possible de la scène sur ce plan. L'axe $\langle C, G \rangle$ définit une direction de projection Δ .

La projection d'un point M s'effectue alors ainsi : On projette M sur le plan médian suivant la direction Δ en M_Δ . On projette ensuite M_Δ sur le plan rétinien de manière perspective. Toutefois, puisque tous les points M_Δ appartiennent au même plan de l'espace, cette deuxième projection peut s'exprimer plus simplement. Elle est équivalente à une homographie qui transforme les points du plan médian en points du plan image, comme montré à l'appendice B.

Bien qu'il soit utile pour certaines applications [Poelman et Kanade, 1994], ce modèle ne présente pour nous aucun attrait, puisqu'il est moins général que le perspectif, et que les calculs n'y sont pas plus simples.

2.1.2.3 Prise en compte de la distorsion

Trois hypothèses implicites sont contenues dans les sections précédentes:

- L'image se forme sur une surface plane.
- Les pixels sont régulièrement espacés dans le plan image.
- La caméra n'a pas de distorsion optique.

Il est généralement considéré que les deux premières hypothèses sont relativement bien vérifiées pour les caméras CCD. Par contre, la distorsion optique n'est, elle, pas négligeable. Celle-ci transforme les coordonnées image idéales en coordonnées image réelles x_u et y_u :

$$\begin{aligned}x_u &= x_d + \delta_x(x_d, y_d) \\y_u &= y_d + \delta_y(x_d, y_d)\end{aligned}$$

Une littérature relativement abondante traite de divers modèles de distorsion, et de leur utilité pratique. Il n'est néanmoins pas très facile d'en comparer les conclusions, car elles dépendent beaucoup du matériel utilisé pour l'expérimentation, lequel est de qualité extrêmement variable. Les éléments de distorsion le plus souvent considérés sont les éléments radiaux qui s'expriment sous une forme polynômiale:

$$x_u = x_d(1 + \kappa_1 r_d^2 + \kappa_2 r_d^4 + \dots)$$

où

$$r_d = \sqrt{x_d^2 + y_d^2}$$

Plusieurs expériences [Tsai, 1987, Beyer, 1992] ont montré que le premier ordre était généralement suffisant pour que la différence entre le point exact et le point projeté dans l'image soit inférieure à 0.1 pixels.

Il est possible d'annuler la distorsion de ces images en extrayant des contours qui sont sensés être des segments, puis en cherchant la meilleure transformation de ces contours vers des segments de droite. La présence de segments dans l'image est donc tout ce dont nous avons besoin pour passer d'une image avec distorsion à une image sans distorsion. L'avantage de ces méthodes est qu'elles opèrent de manière indépendante sur chaque image et ne nécessitent aucune estimation préalable de paramètres intrinsèques ou/et

extrinsèques. Nous considérerons toujours que cette étape de rectification a été effectuée au moment de l'acquisition des images, et nous nous ramènerons ainsi au modèle projectif simple. Diverses méthodes existent [Brand et al., 1993]. Nous avons utilisé [Devernay et Faugeras, 1995].

2.1.3 Formulation projective

Nous supposons ici que le lecteur possède des connaissances élémentaires en géométrie projective. Si ce n'était pas le cas, une introduction destinée à un public de vision par ordinateur peut être trouvée dans [Faugeras, 1993] ou dans le dernier chapitre de [Mundy et Zisserman, 1992]. Une présentation plus mathématique en est donnée par exemple dans [Kanatani, 1991, Enriques, 1930, Springer, 1964, Semple et Kneebone, 1952].

Nous utiliserons le plongement standard de R^n dans \mathcal{P}^n , c'est à dire que le point M de coordonnées (x_1, \dots, x_n) de R^n sera représenté dans \mathcal{P}^n par les coordonnées $(\lambda x_1, \dots, \lambda x_n, \lambda)$ avec $\lambda \neq 0$.

La géométrie projective et les coordonnées homogènes ont beaucoup d'avantages bien connus par rapport à leur pendants cartésiens. Du point de vue de la vision par ordinateur, leurs qualités principales sont:

- *Des formules plus simples* : L'utilisation des coordonnées homogènes conduit généralement à des formules plus simples qui n'utilisent que les fonctions de base de l'algèbre linéaire: déterminants, produits vectoriels, produits scalaires, multiplication de matrices, etc ... Toutes les transformations affines, euclidiennes et toutes les projections perspectives peuvent s'exprimer comme des applications linéaires agissant sur les coordonnées homogènes des points. Par exemple, les coordonnées cartésiennes du point d'intersection entre les droites D et D' du plan, représentées par $ax + by + c = 0$ et $rx + sy + t = 0$ sont:

$$\left(\frac{bt - cs}{as - br}, \frac{cr - at}{as - br} \right) \quad (2.2)$$

En coordonnées homogènes, l'intersection de $\langle a, b, c \rangle$ et $\langle r, s, t \rangle$ est

$$(bt - cs, cr - at, as - br) \quad (2.3)$$

qui n'est rien d'autre que les coordonnées du produit vectoriel de (a, b, c) et (r, s, t) . Comme le montre cet exemple, on peut éliminer la plupart

des étapes de divisions dans les formules géométriques. Les économies réalisées sont souvent suffisantes pour contrebalancer le coût de traitement d'une coordonnée supplémentaire.

- *Moins de cas particuliers* : Les coordonnées homogènes permettent une représentation efficace des points et des lignes à l'infini d'une manière très naturelle, sans aucun rajout, ni de tests, ni de structures additionnelles. Ces objets sont des données valides pour beaucoup d'applications et sont souvent utilisés en vision par ordinateur. Ils aident aussi à réduire le nombre de cas particuliers à traiter pour une application donnée. Par exemple, un programme calculant l'intersection de deux droites n'aura pas à tester leur parallélisme éventuel. Un point "à l'infini" sera produit dans ce cas. En revanche, les modèles cartésiens ou euclidiens doivent traiter ce cas séparément. Si des procédures de ce type sont imbriquées, le nombre de tests grossit généralement de manière multiplicative plutôt qu'additive.
- *Unification et extension des concepts* : Un autre avantage de la géométrie projective est sa capacité à unifier des concepts apparemment différents. Par exemple, les différences entre les cercles, les ellipses, les hyperboles et les paraboles disparaissent en géométrie projective ou elles deviennent toutes des instances de la même courbe, la conique non dégénérée. Un cas plus intéressant pour notre étude est celui des projections perspectives, qui, compliquées à exprimer en géométrie euclidienne parce qu'elles échangent des points à l'infini et des points finis, deviennent des applications linéaires banales en géométrie projective. Nous en ferons une très grande utilisation par la suite.
- *Dualité* : Considérons la fonction $*$ qui à un point $\mathbf{p} = (a, b, c)$ associe la droite $\mathbf{p}^* = (a, b, c)$ et vice versa. Cette application préserve l'incidence : si le point p est sur la droite l alors la droite p^* passe par le point l^* . L'existence d'une telle application implique que pour toute définition, tout théorème, tout algorithme concernant les points a son équivalent pour les droites et vice versa. Ces dualités, que ce soit dans le plan \mathcal{P}^2 entre droites et points ou dans \mathcal{P}^3 entre plans et points nous seront très utiles.

Reprenons maintenant le modèle développé en 2.1.1 avec une formulation projective : le point M est donc représenté dans \mathcal{R} par \mathbf{M}_r de \mathcal{P}^3 qui a pour coordonnées $(X_r, Y_r, Z_r, 1)$ ¹. le changement de repère est la multiplication

1. Le fait que \mathbf{M} ait des coordonnées dans une base euclidienne impose qu'il ne soit pas à l'infini.

par une matrice \mathbf{K} qui s'exprime facilement en fonction de \mathbf{R} et \mathbf{t} .

$$\mathbf{K} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \quad (2.4)$$

L'étape de projection est une multiplication par \mathbf{P}_0 , avec

$$\mathbf{P}_0 = (\mathbf{I}_3 \mid \mathbf{0}) \quad (2.5)$$

Le changement de coordonnées est exprimé par la matrice \mathbf{A} ,

$$\mathbf{A} = \begin{pmatrix} \alpha_u & -\alpha_u \cot \theta & u_0 \\ 0 & \alpha_v \sin \theta & v_0 \\ 0 & 0 & 1 \end{pmatrix} \quad (2.6)$$

Ces trois opérations sont habituellement combinées en une seule. On représente cette projection par $\mathbf{P} = \mathbf{A}\mathbf{P}_0\mathbf{K}$. Le modèle sténopé de la caméra est donc équivalent à l'équation projective suivante:

$$\mathbf{m} = \mathbf{P} \mathbf{M} \quad (2.7)$$

Cette équation est bien sûr définie seulement à une constante multiplicative près comme toutes les équations projectives. \mathbf{P} est une matrice 3×4 définie à un facteur d'échelle près et \mathbf{P} est de rang 3. On retrouve donc la dépendance en 11 paramètres. Par la suite, nous ferons le plus souvent abstraction de \mathbf{A} , de \mathbf{P}_0 et de \mathbf{K} pour nous concentrer sur \mathbf{P} qui contient toute l'information de projection.

Cette formulation a quelques implications.

- Les points qui vérifient $\mathbf{I}_3^T \cdot \mathbf{M} = 0$ sont les points qui se projettent à l'infini sur la rétine. Cette équation peut être vue comme représentant l'appartenance d'un point \mathbf{M} à un plan représenté par \mathbf{I}_3^T . Or, les points se projetant à l'infini sur la rétine appartiennent au plan focal. \mathbf{I}_3^T représente donc le plan focal. De manière équivalente, la première ligne \mathbf{I}_1^T (respectivement la deuxième ligne \mathbf{I}_2^T) représente le plan ayant \mathbf{u} (respectivement \mathbf{v}) comme trace sur Π^r .
- Le centre optique C vérifie l'équation $\mathbf{P} \cdot \mathbf{C} = \mathbf{0}$. En effet, C appartient aux plans définis par \mathbf{I}_1^T , \mathbf{I}_2^T et \mathbf{I}_3^T . Ces trois plans sont distincts, ils se coupent donc en un point unique. On a donc $\mathbf{P} \cdot \mathbf{C} = \mathbf{0}$ pour le centre

optique C et pour lui seulement. On remarque aussi que $\mathbf{0}$ n'est pas un point projectif. L'image du centre optique n'est donc pas défini par la matrice de projection. On retrouve ceci par la géométrie. L'image du centre optique par une projection perspective n'est pas définie.

2.1.4 Géométrie de N caméras

Dans cette section, nous allons considérer le cas d'un système de N caméras, qui peuvent être semblables ou différentes. Nous allons déterminer quel est le nombre de paramètres nécessaires pour décrire ce système.

Le nombre de paramètres nécessaire pour décrire le système dépend des transformations qui doivent laisser le système invariant. Les paramètres devront être des invariants à une certaine classe de transformations. Par invariant, on entend propriété d'une configuration géométrique qui reste inchangée sous l'action d'une classe de transformation. Par exemple, la distance entre deux points est un invariant pour les transformations euclidiennes, mais n'est pas un invariant pour les transformations projectives. Chaque invariant scalaire indépendant s'appelle un degré de liberté. On dit que le système a α degrés de liberté sous l'action des transformations euclidiennes si la configuration géométrique est caractérisée par α paramètres indépendants invariants aux transformations euclidiennes.

Nous nous sommes placés depuis le début de ce chapitre dans le formalisme des matrices de projection. Chacune de ces matrices de projection dépend de 11 paramètres puisque le facteur d'échelle n'intervient pas dans les équations².

Les transformations qui nous intéressent sont les transformations de l'espace. Si nous considérons les transformations projectives, le nombre de degrés de liberté est donc

$$\alpha = 11N - 15 \tag{2.8}$$

Bien sûr, pour $N = 1$, le nombre de degrés de liberté n'est pas négatif, mais simplement nul.

Si nous considérons les transformations affines de l'espace, alors

$$\alpha = 11N - 12,$$

si nous considérons les transformations euclidiennes, alors

$$\alpha = 11N - 6$$

2. Nous pouvons par exemple décider que le plus grand élément doit être de norme un, ou bien que la somme aux carrés des éléments est un.

Une approche plus complète du comptage des invariants à un groupe donné peut être trouvée dans l'introduction de [Mundy et Zisserman, 1992].

2.2 La stéréovision binoculaire

Après avoir défini le formalisme de la projection sur une caméra, nous nous intéressons aux relations qui existent entre deux vues de la même scène. Nous verrons principalement qu'il existe une relation projective bilinéaire entre deux points en correspondance.

2.2.1 La matrice essentielle

La matrice essentielle a été introduite pour la première fois par Longuet-Higgins [Longuet-Higgins, 1981] dans le contexte de l'étude du mouvement.

La dérivation est relativement simple puisque les deux caméras sont supposées avoir des paramètres intrinsèques connus. On ne travaille alors plus en coordonnées pixels, mais en coordonnées image corrigées. Cela signifie que par un choix judicieux du repère de l'espace et de celui de la première caméra, on peut se ramener au système suivant

$$\begin{aligned} \mathbf{P}_1 &= [\mathbf{I}|\mathbf{0}] \\ \mathbf{P}_2 &= [\mathbf{R}|\mathbf{t}] \end{aligned} \quad (2.9)$$

où \mathbf{R}, \mathbf{t} est le déplacement entre la première et la deuxième caméra. Soit \mathbf{M} un point de l'espace représenté par le vecteur $[X, Y, Z, T]^T$, \mathbf{m}_1 sa projection dans la première image, \mathbf{m}_2 sa projection dans la deuxième image. On voit tout de suite que \mathbf{m}_1 est l'image de la projection de \mathbf{M} sur le plan d'équation $T = 0$. Puisque \mathbf{M} n'est pas à l'infini, il peut s'écrire sous la forme $[X, Y, Z, 1]^T$. En remplaçant 2.9 dans 2.7, on obtient

$$\begin{aligned} \mathbf{m}_1 &= \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \\ \mathbf{m}_2 &= \mathbf{R}\mathbf{m}_1 + \mathbf{t} \end{aligned}$$

On remarque tout d'abord que si $\mathbf{t} = \mathbf{0}$ où si \mathbf{M} est dans le plan d'équation $T = 0$, le système précédent se réduit à

$$\mathbf{m}_2 = \mathbf{R}\mathbf{m}_1$$

On note aussi $\mathbf{R} = \mathbf{H}_\infty$ car \mathbf{R} représente une homographie³ entre points du plan à l'infini. Pour tous les autres points, le théorème suivant s'applique:

Proposition : Il existe une matrice 3×3 notée \mathbf{E}_{12} qui vérifie pour tout \mathbf{M} , associé à ses projections dans les images \mathbf{m}_1 et \mathbf{m}_2

$$\mathbf{m}_2^T \mathbf{E}_{12} \mathbf{m}_1 = 0 \quad (2.10)$$

Preuve: Il suffit de choisir $\mathbf{E}_{12} = [\mathbf{t}]_\times \mathbf{R}$. En effet, puisque $\mathbf{m}_2 = \mathbf{R} \mathbf{m}_1 + \mathbf{t}$,

$$\begin{aligned} \mathbf{m}_2^T \mathbf{E}_{12} \mathbf{m}_1 &= (\mathbf{R} \mathbf{m}_1 + \mathbf{t})^T ([\mathbf{t}]_\times \mathbf{R} \mathbf{m}_1) \\ &= (\mathbf{R} \mathbf{m}_1)^T [\mathbf{t}]_\times (\mathbf{R} \mathbf{m}_1) \\ &= 0. \end{aligned}$$

car $[\mathbf{t}]_\times$ est antisymétrique. Certaines remarques viennent immédiatement à l'esprit : de par sa forme, \mathbf{E}_{12} n'est pas de rang 3 mais de rang 2. On note \mathbf{e}_{12} le noyau de \mathbf{E}_{12} et on l'appelle épipôle de la caméra 2 dans l'image 1. Le vecteur $\mathbf{E}_{12} \mathbf{m}_1$ représente une droite dans le plan de l'image 2. La relation 2.10 signifie donc l'appartenance de \mathbf{m}_2 à une droite donnée, et ce quel que soit le point sur le rayon correspondant \mathbf{m}_1 . Ces droites sont appelées droites épipolaires.

L'existence d'un noyau \mathbf{e}_{12} signifie que toutes les droites épipolaires passent par un même point de l'image. Un tel point est de toute évidence l'image du centre optique de la deuxième caméra, comme illustré par la figure 2.3. \mathbf{e}_{12} appartient à toutes les droites épipolaires engendrées par des points de l'image 2 (voir figure 2.4). En fait, on a $\mathbf{E}_{12} \mathbf{m}_1 = \mathbf{e}_{21} \times \mathbf{m}_2$. D'autre part, en transposant la relation 2.10, on a $\mathbf{E}_{12} = \mathbf{E}_{21}^T$.

Toutefois, la dérivation de l'équation 2.10 masque une explication géométrique plus simple : Connaissant \mathbf{m}_1 , \mathbf{M} se trouve sur le rayon $C_1 \mathbf{m}_1$. \mathbf{m}_2 se trouve donc sur l'image de ce rayon dans la deuxième caméra (figure 2.3).

L'interprétation géométrique nous permet aussi de voir que les lignes épipolaires sont en correspondance. En effet, $\mathbf{E}_{12} \mathbf{m}_1$ et $\mathbf{E}_{21} \mathbf{m}_2$ sont les traces sur les images 1 et 2 du plan défini par C_1 , C_2 et M . Tout les points appartenant à $\mathbf{E}_{12} \mathbf{m}_1$ auront $\mathbf{E}_{21} \mathbf{m}_2$ pour droite épipolaire.

On peut aussi citer d'autres propriétés, sur lesquelles nous ne nous étendrons pas, puisque nous n'en n'aurons pas usage. Citons simplement le fait

3. Le fait qu'un plan de l'espace corresponde à une homographie dans les images est démontré à l'appendice B.

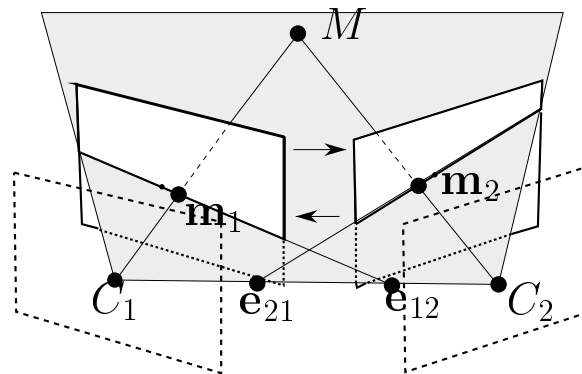


FIG. 2.3 – La géométrie épipolaire. C_1 et C_2 sont les centres optiques. e_{21} et e_{12} sont les épipoles.

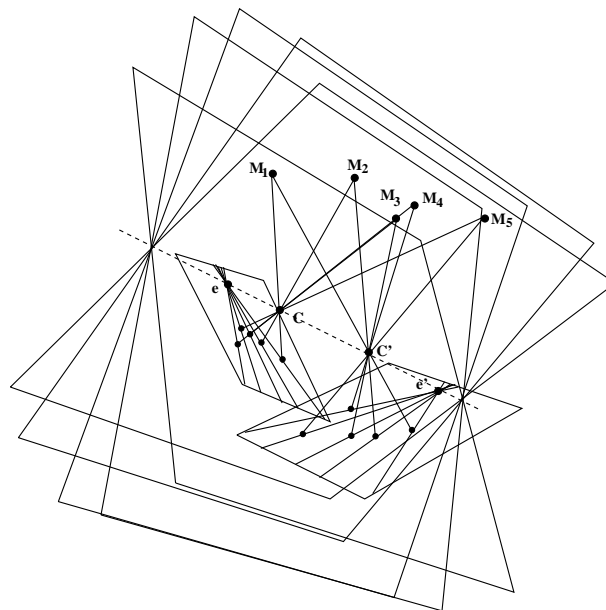


FIG. 2.4 – Faisceau de droites épipolaires

que les faisceaux de droites épipolaires sont en correspondance homographique [Luong, 1992], que \mathbf{E} a deux valeurs propres identiques et une valeur propre nulle [Huang et Faugeras, 1989], que l'on peut retrouver \mathbf{R} et \mathbf{t} à partir de \mathbf{E} [Netravali et al., 1989].

2.2.2 La matrice fondamentale

La matrice essentielle, bien que nous ayant appris beaucoup sur la géométrie d'un système binoculaire, n'est pas suffisante pour ce que nous voulons faire. En effet, nous ne voulons pas nous restreindre au cas où les paramètres intrinsèques sont déjà connus. Luong a démontré que dans ce cas, il existe une entité plus appropriée pour l'étude des caméras qui est une matrice 3×3 appelée matrice fondamentale et notée \mathbf{F} . Elle est reliée à la matrice essentielle par une simple relation linéaire:

$$\mathbf{F}_{12} = \mathbf{A}_2 \mathbf{E}_{12} \mathbf{A}_1^{-1} \quad (2.11)$$

où \mathbf{A}_1 et \mathbf{A}_2 sont les matrices de paramètres intrinsèques définies par l'équation 2.1.

Les propriétés de \mathbf{F} sont relativement semblables à celles de \mathbf{E} . l'équation 2.1 implique que le rang de \mathbf{F} est deux comme \mathbf{E} . Le noyau de \mathbf{F}_{12} est toujours l'épipole \mathbf{e}_{12} et est toujours l'image du centre optique de la deuxième caméra vu dans la première, mais exprimé cette fois en coordonnées pixels au lieu d'être en coordonnées image normalisées. L'équation 2.10 se transforme en

$$\mathbf{m}_2^T \mathbf{F}_{12} \mathbf{m}_1 = 0 \quad (2.12)$$

Toutes les entités sont exprimées en pixels. $\mathbf{F}_{12} \mathbf{m}_1$ est toujours la droite de l'image 2 sur laquelle se trouvent les correspondants possibles de \mathbf{m}_1 . Les droites $\mathbf{F}_{12} \mathbf{m}_1$ et $\mathbf{F}_{21} \mathbf{m}_2$ sont toujours en correspondance homographique. Une différence toutefois : Les deux valeurs singulières non nulles de \mathbf{F} ne sont en général pas identiques.

La matrice \mathbf{F} , bien qu'étant une matrice 3×3 définie à un facteur d'échelle près, ne dépend que de 7 paramètres puisque $|\mathbf{F}| = 0$. Ces 7 paramètres sont en fait les coordonnées des deux épipoles (deux fois deux paramètres) et l'homographie de \mathcal{P}^1 qui fait correspondre une droite épipolaire à une autre (trois paramètres). Cette décomposition est étudiée en détail dans [Luong, 1992]. D'autres décompositions sont bien sûr possibles, certaines étant numériquement plus intéressantes, certaines ayant des significations géométriques plus faciles. Elles comportent toutes 7 paramètres indépendants, ce qui signifie

que la matrice fondamentale contient toute l'information d'un système binoculaire projectif qui dépend de $11 \times 2 - 15 = 7$ paramètres (cf équation 2.8). On peut se reporter à [Viéville et al., 1995] et [Luong et Viéville, 1994] pour de plus amples détails.

Récapitulons les principales propriétés d'un système binoculaire.

- Il existe une matrice (\mathbf{E}_{12} si les paramètres intrinsèques sont connus, \mathbf{F}_{12} si ils sont inconnus) qui vérifie $\mathbf{m}_2 \mathbf{F}_{12} \mathbf{m}_1 = 0$, quels que soient les pixels \mathbf{m}_1 et \mathbf{m}_2 images d'un même point 3-D.
- $\mathbf{F}_{12} \mathbf{m}_1$ est la droite épipolaire engendrée par \mathbf{m}_1 .
- $\mathbf{F}_{21} = \mathbf{F}_{12}^T$.
- \mathbf{e}_{12} (resp. \mathbf{e}_{21}) appartient au noyau de \mathbf{F}_{12} (resp. \mathbf{F}_{12}^T). Il représente l'image de C_2 (resp. C_1) dans l'image 1 (resp. 2).
- La matrice \mathbf{F} dépend de 7 paramètres.

2.3 Trois caméras

Le cas de trois caméras présente un intérêt particulier: c'est la configuration générique avec le plus petit nombre de caméras. Pour bien comprendre la géométrie compliquée d'un système de N caméras, il faut d'abord s'attacher à bien comprendre la géométrie de 3 caméras. En effet, aucun phénomène mathématique nouveau ne va apparaître lorsque l'on passe de 3 à N images comme remarqué en [Beardsley et al., 1992].

La raison en est simple: le groupe des transformations projectives de l'espace agit à droite sur les matrices de projections par $\mathbf{P}_i \mapsto \mathbf{P}_i \mathbf{H}$. Si l'on fixe la géométrie des caméras, mais en laissant varier la base de l'espace \mathcal{P}^3 dans laquelle elle est représentée, pour une seule caméra, \mathbf{P}_1 est quelconque. Pour deux caméras, \mathbf{P}_1 est quelconque et \mathbf{P}_2 a 4 degrés de liberté. Les 7 degrés de liberté restants correspondent à la matrice fondamentale \mathbf{F}_{12} . Pour plus de deux caméras, \mathbf{P}_1 est quelconque, \mathbf{P}_2 a 4 degrés de liberté une fois que \mathbf{P}_1 est choisi, mais $\mathbf{P}_3 \dots \mathbf{P}_N$ sont fixés une fois que \mathbf{P}_1 et \mathbf{P}_2 sont choisis. Le choix de \mathbf{P}_1 et \mathbf{P}_2 n'est que le choix de la base de \mathcal{P}^3 dans laquelle sont exprimées les matrices de projection. En effet, la première colonne de \mathbf{P}_1 représente l'image du point projectif $[1, 0, 0, 0]^T$ dans la première image, la deuxième représente l'image de $[0, 1, 0, 0]^T$, etc...La preuve exacte sera donnée par la suite.

Cela signifie que, si les images 1 et 2 jouent un rôle particulier puisque le choix de leurs matrices de projection associées permet de fixer la base de \mathcal{P}^3 , les images suivantes se comportent algébriquement de manière identique : elles ajoutent 11 paramètres au système. La méthode qui va nous servir à calculer la troisième matrice de projection peut aussi nous servir pour calculer la N ième.

Nous nous attacherons à étudier ce cas avec soin, car il permet une bonne compréhension des problèmes de la géométrie de N caméras.

2.3.1 Les contraintes entre les trois matrices fondamentales

Pour un système trinoculaire, il y a trois matrices fondamentales, qui totalisent donc $3 \times 7 = 21$ paramètres. Or, l'équation 2.8 nous donne $3 \times 11 - 15 = 18$ paramètres. Ces 21 paramètres ne sont donc pas tous indépendants. Nous étudions deux cas, suivant la position relative des centres optiques. Le lecteur ne doit pas penser que le cas où les centres optiques sont alignés est une manie de mathématicien : les instabilités numériques au voisinage de cette situation peuvent être particulièrement importantes comme nous le verrons ultérieurement. Or, il est extrêmement fréquent que l'opérateur déplace la caméra en ligne droite ou lui applique une rotation pure autour de son centre optique.

2.3.1.1 Cas où C_1 , C_2 et C_3 ne sont pas alignés

Nous empruntons ici des calculs à [Faugeras et Robert, 1994]. Si C_1 , C_2 et C_3 ne sont pas alignés, il existe des contraintes entre les épipoles. Prenons par exemple les épipoles \mathbf{e}_{13} et \mathbf{e}_{23} images de C_3 dans la première et la deuxième caméra. La droite épipolaire \mathbf{l}_2 de \mathbf{e}_{13} dans la deuxième image est $\mathbf{F}_{12}\mathbf{e}_{13}$. D'autre part, \mathbf{l}_2 contient \mathbf{e}_{21} et \mathbf{e}_{23} , on a donc $\mathbf{l}_2 = \mathbf{e}_{21} \times \mathbf{e}_{23}$. La relation s'écrit alors :

$$\mathbf{F}_{12}\mathbf{e}_{13} = \mathbf{e}_{23} \times \mathbf{e}_{21} \quad (2.13)$$

L'équation 2.13 fait intervenir les trois matrices fondamentales : \mathbf{F}_{12} , \mathbf{F}_{13} et \mathbf{F}_{23} . Des relations équivalentes existent en choisissant C_1 et C_2 à la place de C_3 :

$$\begin{aligned} \mathbf{F}_{31}\mathbf{e}_{32} &= \mathbf{e}_{12} \times \mathbf{e}_{13} \\ \mathbf{F}_{23}\mathbf{e}_{21} &= \mathbf{e}_{31} \times \mathbf{e}_{32} \end{aligned}$$

Toutes ces contraintes s'interprètent géométriquement par le fait que tous les épipoles appartiennent au même plan, c'est à dire à celui défini par (C_1, C_2, C_3) (voir figure 2.5). Ce plan est appelé plan trifocal. Cette démonstration n'est valable que si C_1, C_2 et C_3 ne sont pas alignés.

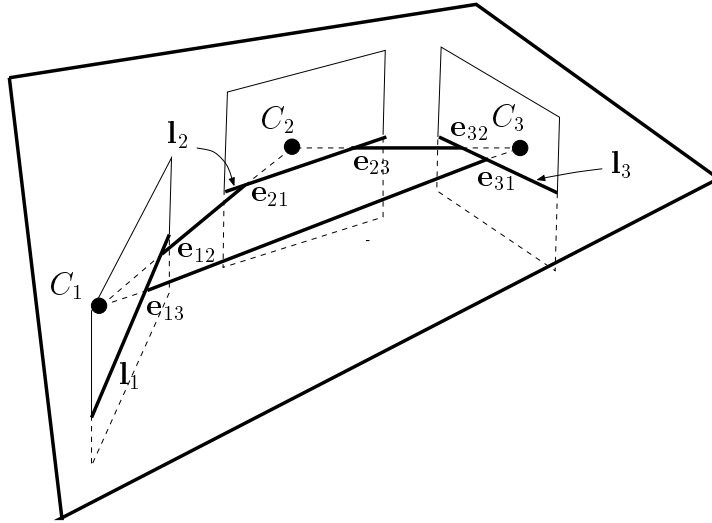


FIG. 2.5 – 6 épipoles dans un plan. l_1, l_2 et l_3 représentent la trace du plan trifocal dans les images.

Ces trois contraintes nous satisfont pleinement puisque le système dépend de 18 paramètres et que les matrices fondamentales en exhibent 21 auxquels on peut soustraire les 3 contraintes.

2.3.1.2 Cas où C_1, C_2 et C_3 sont alignés

Les contraintes prennent alors une forme beaucoup plus simple; on a le système suivant:

$$\begin{aligned} \mathbf{e}_{12} &= \mathbf{e}_{13} \\ \mathbf{e}_{23} &= \mathbf{e}_{21} \\ \mathbf{e}_{31} &= \mathbf{e}_{32} \end{aligned}$$

Les degrés de liberté du système sont alors bien moins nombreux. La détermination d'un épipole dans chacune des images fixe les trois autres. Il suffit par exemple de déterminer $\mathbf{e}_{12}, \mathbf{e}_{23}$ et \mathbf{e}_{31} . Chaque matrice fondamentale dépend alors de cinq paramètres (un épipole, et une homographie de

droites épipolaires). Géométriquement, nous avons tous les épiholes sur la même droite comme dans la figure 2.6. Dans ce cas, les matrices fondamentales ne contiennent pas toute l'information du système trinoculaire. Nous verrons en effet qu'il est impossible de trouver les matrices de projection si l'on ne connaît que les matrices fondamentales. D'autres informations permettent bien sûr de trouver ces matrices de projection, comme par exemple des correspondances entre les trois images, ou bien la connaissance d'une trilinearité, comme nous le verrons plus tard.

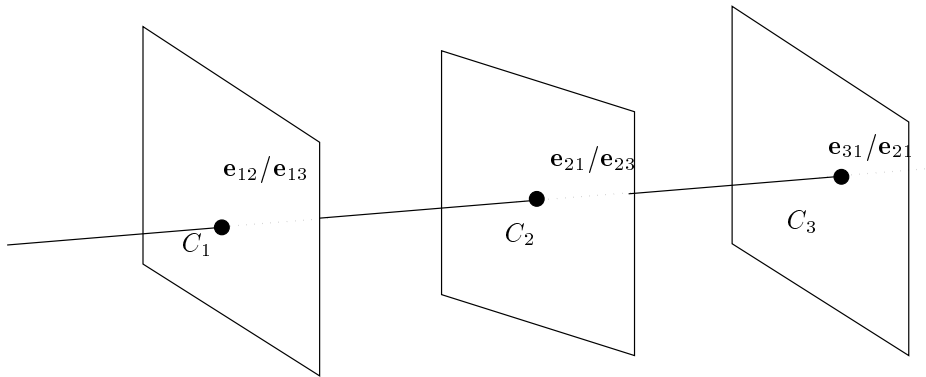


FIG. 2.6 – 6 épiholes sur une droite

Il apparaît désormais clair que les matrices fondamentales ne constituent pas un bon moyen de représenter un système trinoculaire. Elles cumulent deux défauts : Elles peuvent à la fois être redondantes et l'expression des contraintes entre elles est malaisé⁴, ou bien ne pas contenir toute l'information nécessaire pour retrouver la structure projective de la scène.

La situation empire avec le nombre de caméras comme l'intuition le laisse penser. En effet, le nombre de matrices fondamentales est $N(N - 1)/2$ soit un total de $7N(N - 1)/2$ paramètres. Ceci est bien évidemment à comparer avec les $11N - 15$ degrés de liberté du système. Par exemple, pour $N = 20$, le système a 205 degrés de liberté, mais les matrices fondamentales ont 139300 paramètres !

4. Les équations du type 2.13 font intervenir des noyaux d'applications linéaires. Ce sont donc des contraintes polynômiales. En l'occurrence, l'ordre de ces polynômes sont 3 ou 4. Il existe d'autres types de contraintes que celles-ci pour un nombre de caméras supérieur à 4, et ces contraintes sont aussi polynômiales, mais d'ordre 5. Elles sont de la forme $(\mathbf{F}_{14}\mathbf{m}_1 \times \mathbf{F}_{24}\mathbf{m}_2)^T \mathbf{F}_{34}(\mathbf{F}_{13}\mathbf{m}_1 \times \mathbf{F}_{23}\mathbf{m}_2) = 0$.

2.3.2 Les trilinearités

Les trilinearités ont été introduites par Amnon Shashua dans le contexte de la reconnaissance d'objets à partir de vues différentes [Shashua, 1994a]. Shashua étendait ainsi une proposition de Ullman et Basri [Ullman et Basri, 1991] limitée aux projections orthographiques. Par souci de clarté, nous ne démontrerons pas leur existence à la manière de Shashua, mais nous emprunterons des calculs plus simples à Hartley [Hartley, 1994c].

Plaçons nous tout d'abord dans une base projective telle que \mathbf{P}_1 exprimé dans cette base ait la forme particulière $[\mathbf{I}|\mathbf{0}]$. Posons

$$\begin{aligned}\mathbf{P}_2 &= [a_{ik}] \\ \mathbf{P}_3 &= [b_{ik}]\end{aligned}$$

Étant donnée la forme simple de \mathbf{P}_1 , le point \mathbf{M} s'écrit :

$$\mathbf{M} = \begin{pmatrix} \mathbf{m}_1 \\ t \end{pmatrix}$$

avec t à déterminer. On peut vérifier que t est simplement l'invariant affine relatif k développé par Shashua dans [Shashua, 1994b]. Maintenant, si l'on projette \mathbf{M} dans la deuxième image, on obtient

$$\forall i \in \{1, 2, 3\}, \quad m_2^i = \sum_{k=1}^3 a_{ik} m_1^k + a_{i4} t$$

Nous pouvons éliminer le facteur d'échelle inhérent à ces équations pour obtenir

$$m_2^i (a_{jk} m_1^k + a_{j4} t) = m_2^j (a_{ik} m_1^k + a_{i4} t)$$

Chaque choix d'indices i et j distincts donne une équation différente. Parmi ces trois équations, seulement deux sont indépendantes. A partir de chacune de ces équations on peut estimer t de manière indépendante.

$$t = m_1^k (m_2^i a_{jk} - m_2^j a_{ik}) / (m_2^j a_{i4} - m_2^i a_{j4}) \quad (2.14)$$

En substituant cette valeur de t , \mathbf{M} peut s'écrire

$$\mathbf{M} = \begin{pmatrix} (m_2^j a_{i4} - m_2^i a_{j4}) \mathbf{m}_1 \\ (m_2^i a_{jk} - m_2^j a_{ik}) m_1^k \end{pmatrix}$$

En projetant maintenant ce point dans la troisième caméra,

$$\begin{aligned}
m_3^l &= b_{lk} m_1^k (m_2^j a_{i4} - m_2^i a_{j4}) \\
&+ b_{l4} m_1^k (m_2^i a_{jk} - m_2^j a_{ik}) \\
&= m_1^k m_2^i (a_{jk} b_{l4} - a_{j4} b_{lk}) \\
&- m_1^k m_2^j (a_{ik} b_{l4} - a_{i4} b_{lk})
\end{aligned}$$

On peut ranger les éléments dépendants de \mathbf{P}_2 et \mathbf{P}_3 dans un tenseur pour obtenir l'expression suivante

$$\mathbf{m}_3 = \mathbf{m}_1 (m_2^i T_{kjl} - m_2^j T_{kil}) \quad (2.15)$$

avec

$$T_{ijk} = a_{ji} b_{k4} - a_{j4} b_{ki} \quad (2.16)$$

En modifiant 2.15, on obtient

$$\sum_k m_1^k (m_2^i m_3^m T_{kjl} - m_2^j m_3^m T_{kil}) = \sum_k m_1^k (m_2^i m_3^l T_{kjm} - m_2^j m_3^l T_{kim}) \quad (2.17)$$

On peut, à la manière de Bill Triggs, réécrire ces nombreuses équations, en une seule, en considérant tout les indices possibles pour i, j, l et m :

$$[\mathbf{m}_2]_{\times} [T \mathbf{m}_1] [\mathbf{m}_3]_{\times} = \mathbf{0}_{3 \times 3} \quad (2.18)$$

L'équation 2.18 met en évidence la structure de tenseur. Le tenseur est covariant par rapport à \mathbf{m}_1 et contravariant par rapport à \mathbf{m}_2 et \mathbf{m}_3 .

Les équations 2.17 sont exactement les relations trinéaires introduites par Shashua dans [Shashua, 1994b]. Dans ces équations, i, j, l et m sont des variables. Il y a une équation pour chaque choix d'indices $i \neq j$ et $l \neq m$. Étant donné que l'on obtient des équations identiques si l'on inverse i et j ou l et m , nous pouvons considérer sans restriction que $i < j$ et que $l < m$. Il y a donc 9 équations différentes mais pas nécessairement linéairement indépendantes. Ce nombre est implicite dans la dérivation de l'équation 2.18. Ici, nous notons les équations suivant les indices i, j, l et m qui ont servi à les créer. L'équation pour laquelle $i = 1, j = 2, m = 1, l = 2$ est notée $E_{1212} = 0$. Considérons par exemple les équations

$$\begin{aligned}
E_{1212} = \sum_k m_1^k (& m_2^1 m_3^1 T_{k22} \\
& -m_2^2 m_3^1 T_{k12} \\
& -m_2^1 m_3^2 T_{k21} \\
& +m_2^2 m_3^2 T_{k11})
\end{aligned} \tag{2.19}$$

$$\begin{aligned}
E_{2312} = \sum_k m_1^k (& m_2^2 m_3^1 T_{k32} \\
& -m_2^3 m_3^1 T_{k22} \\
& -m_2^2 m_3^2 T_{k31} \\
& +m_2^3 m_3^2 T_{k21})
\end{aligned} \tag{2.20}$$

$$\begin{aligned}
E_{3112} = \sum_k m_1^k (& m_2^3 m_3^1 T_{k12} \\
& -m_2^1 m_3^1 T_{k32} \\
& -m_2^3 m_3^2 T_{k11} \\
& +m_2^1 m_3^2 T_{k31})
\end{aligned} \tag{2.21}$$

alors, on vérifie que l'on a l'identité

$$m_2^3 E_{1212} + m_2^1 E_{2312} + m_2^2 E_{3112} = 0$$

On peut aussi démontrer de la même manière que E_{12ml} est combinaison linéaire de E_{23ml} et de E_{31ml} , mais aussi que E_{ij12} est combinaison linéaire de E_{ij23} et de E_{ij31} . Le nombre d'équations indépendantes est donc 4.

Le rôle de la caméra 1 étant particulier dans notre démonstration, nous pouvons aussi choisir la caméra 2 et la caméra 3 pour obtenir encore 8 autres équations. Nous classifions ces équations comme appartenant au groupe 1, au groupe 2 ou au groupe 3 suivant la caméra choisie pour jouer un rôle particulier.

Le tenseur T comporte 27 coefficients qui sont bien entendu définis à un facteur d'échelle près, soit 26 coefficients si on choisit une normalisation pour T . Ces 26 coefficients ne sont pas indépendants puisque on peut les déduire des 18 coefficients représentant les matrices de projection. En fait, un simple calcul de déterminant montre que T_{1ij} , T_{2ij} et T_{3ij} sont de rang 2. On peut noter \mathbf{u}_k (respectivement \mathbf{v}_k) le noyau de la matrice T_{kij} (respectivement T_{kij}^T) à k fixé. La famille de vecteurs $\{\mathbf{u}_k\}$ (respectivement $\{\mathbf{v}_k\}$) est orthogonale à l'épipôle a_{i4} (respectivement b_{i4}). Ces contraintes déterminent 5 équations (trois pour la nullité des déterminants des matrices T_{kij} à k fixé, et deux pour la dépendance entre les vecteurs \mathbf{u}_k et entre les vecteurs \mathbf{v}_k). Nous n'avons

malheureusement encore que 5 équations, ce qui nous laisse encore avec 21 paramètres. Ceci est dû au fait que les matrices de projection \mathbf{P}_1 et \mathbf{P}_2 ont encore 3 degrés de liberté dont nous ne nous sommes pas préoccupés. Nous verrons dans la section suivante ainsi qu'à la section 2.5 comment exprimer ces dernières contraintes de manière satisfaisante.

Ce tenseur T a aussi une signification pour la mise en correspondance des droites entre 3 images. En effet, connaissant l'image d'une droite dans deux caméras, on peut en déduire son image dans une troisième grâce à la formule

$$\lambda_1^i = \sum_{j,k} \lambda_2^j \lambda_3^k T_{ijk} \quad (2.22)$$

où λ_μ est le représentant projectif de la droite projetée dans l'image μ . Ce résultat est démontré dans [Hartley, 1994b] et plus tard à l'équation 2.36.

2.4 Une théorie unificatrice : les déterminants

Le lecteur inquiet voit le nombre de caméras grandir et les équations se complexifier, apparemment sans limite. Il n'existe pour l'instant pas de relations entre la manière dont nous avons dérivé la géométrie trilinéaire et la géométrie épipolaire. La géométrie de deux caméras était décrite par une matrice 3×3 avec 1 contrainte, la géométrie de trois caméras par un tenseur $3 \times 3 \times 3$ avec 8 contraintes. Il n'est pas envisageable de continuer sur cette voie pour déterminer la géométrie de N caméras.

Heureusement, Faugeras et Mourrain d'une part [Faugeras et Mourrain, 1995], Triggs d'autre part [Triggs, 1994] ont démontré simultanément qu'il n'existait rien au delà des quadrilinéarités, ou plus exactement que les N -linéarités sont des combinaisons des bi-, tri- et quadrilinéarités. De plus, les bilinéarités, les trilinearités et les quadrilinéarités peuvent se dériver de manière identique en suivant la même méthodologie.

Dans cette section et dans cette section seulement, nous noterons

$$\mathbf{P}_i = \begin{bmatrix} \mathbf{i} \\ \mathbf{i} + \mathbf{1} \\ \mathbf{i} + \mathbf{2} \end{bmatrix} \quad (2.23)$$

parce que les lignes des matrices de projection occuperont une place très importante dans notre exposé.

2.4.1 Brèves Notions sur l'algèbre de Grassmann-Cayley

L'algèbre de Grassmann-Cayley connu sous le nom d'*algèbre double* est un formalisme algébrique pour exprimer des relations en géométrie projective. L'algèbre double peut être défini sur un espace vectoriel comme suit :

Definition:

Soit E^n un espace vectoriel de dimension n défini sur un corps \mathcal{K} ⁵. Sur cet espace on va définir la forme n -linéaire alternée qu'on appelle crochet :

$$\mathbf{x}_1 \dots \mathbf{x}_n \mapsto [\mathbf{x}_1 \dots \mathbf{x}_n] = \det(x_{ij}) \in \mathcal{K}$$

où $\mathbf{x}_i = (x_{i1}, \dots, x_{in})^T \in E^n$.

Alors, l'espace E^n doté de cette opération définit un espace Cayley. Un espace Cayley qui possède la propriété suivante

$$\forall \mathbf{x} \in E^n, \exists \mathbf{x}_2 \dots \mathbf{x}_n \text{ tels que } [\mathbf{x} \ \mathbf{x}_2 \dots \mathbf{x}_n] \neq 0$$

est appelé un espace Cayley standard.

Dans cet espace, on peut définir la relation d'équivalence suivante :

Definition:

On dit que deux séquences de k vecteurs indépendants $\mathbf{a}_i, i = 1 \dots k$ et $\mathbf{b}_i, i = 1 \dots k$ sont équivalentes si pour n'importe quel ensemble de $n - k$ vecteurs indépendants $\mathbf{x}_{k+1}, \dots, \mathbf{x}_n$ on a :

$$[\mathbf{a}_1 \dots \mathbf{a}_k \ \mathbf{x}_{k+1} \dots \mathbf{x}_n] = [\mathbf{b}_1 \dots \mathbf{b}_k \ \mathbf{x}_{k+1} \dots \mathbf{x}_n]$$

Les classes d'équivalence de cette relation s'appellent des extenseurs de degré k et sont notés $\mathbf{A} = \mathbf{a}_1 \vee \mathbf{a}_2 \vee \dots \vee \mathbf{a}_k$ ou plus simplement $A = \mathbf{a}_1 \mathbf{a}_2 \dots \mathbf{a}_k$.

Les vecteurs $\mathbf{a}_i, i = 1 \dots k$ étant indépendants, ils définissent un sous espace de E^n de dimension k que l'on va noter par $\overline{A} = \text{span}\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_k\}$. L'opération \vee introduite par la relation d'équivalence est un produit extérieur, multilinéaire, associatif et antisymétrique, que l'on appelle *réunion*.

5. \mathcal{K} est en général le corps des réels ou le corps des complexes.

A l'aide de cette opération on peut définir l'algèbre extérieure suivant (voir [Dieudonné, 1969]) :

Definition:

Soit $\{\mathbf{e}_1, \dots, \mathbf{e}_n\}$ une base de E^n . Pour un quelconque $k \leq n$, on peut définir l'espace vectoriel $\Lambda^k(E^n)$ de dimension C_n^k tel que :

$$\{\mathbf{e}_{j_1} \vee \mathbf{e}_{j_2} \vee \dots \vee \mathbf{e}_{j_k} \mid 1 \leq j_1 < j_2 < \dots < j_k \leq n\}.$$

soit une base pour cet espace. Alors,

$$\Lambda(E^n) = \Lambda^0(E^n) \oplus \Lambda^1(E^n) \oplus \dots \oplus \Lambda^n(E^n)$$

ou $\Lambda^0(E^n) = \mathcal{K}$ est un algèbre extérieure défini par \vee .

Si on considère maintenant k éléments indépendants $\mathbf{a}_i \in E^n, i = 1 \dots k$ tel que $\mathbf{a}_i = \sum_{j=1}^n \alpha_{ij} \mathbf{e}_j$, grâce à la multilinéarité et à l'antisymétrie du produit extérieur, on obtient :

$$\mathbf{A} = \mathbf{a}_1 \vee \mathbf{a}_2 \vee \dots \vee \mathbf{a}_k = \sum_{1 \leq j_1 < j_2 < \dots < j_k \leq n} \det(\mathbf{M}_{j_1 j_2 \dots j_k}) \mathbf{e}_{j_1} \vee \mathbf{e}_{j_2} \vee \dots \vee \mathbf{e}_{j_k}$$

où $\mathbf{M}_{j_1 j_2 \dots j_k} = (\alpha_{m j_n})_{m,n=1 \dots k}$. Ceci montre qu'un extenseur de degré k est un élément de l'espace $\Lambda^k(E^n)$. Si maintenant on prend deux extenseurs $\mathbf{A} = \mathbf{a}_1 \mathbf{a}_2 \dots \mathbf{a}_k$ de degré k et $\mathbf{B} = \mathbf{b}_1 \dots \mathbf{b}_l$ de degré l , leur réunion $\mathbf{A} \vee \mathbf{B} = \mathbf{a}_1 \dots \mathbf{a}_k \mathbf{b}_1 \dots \mathbf{b}_l$ est également un extenseur de degré $k + l$ tel que :

Definition:

La réunion $\mathbf{A} \vee \mathbf{B}$ est non nulle si et seulement si les vecteurs $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_k, \mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_l$ sont indépendants. Dans ce cas, on a :

$$\overline{\mathbf{A}} + \overline{\mathbf{B}} = \overline{\mathbf{A} \vee \mathbf{B}} = \text{span}\{\mathbf{a}_1, \dots, \mathbf{a}_k, \mathbf{b}_1, \dots, \mathbf{b}_l\}$$

Cette proposition montre que la réunion des extenseurs correspond aux réunions des sous espaces vectoriels engendrés par les extenseurs, ce qui explique le choix du nom de réunion pour le produit extérieur \vee . La question qui se pose maintenant est de trouver une opération définie sur les extenseurs qui corresponde aux intersections des sous espaces vectoriels correspondants. On va d'abord identifier l'espace $\Lambda^n(E^n)$ de dimension $C_n^n = 1$

avec l'espace \mathcal{K} par $\mathbf{e}_1 \mathbf{e}_2 \dots \mathbf{e}_n = 1$. En conséquence, un extenseur de degré n peut être exprimé par le déterminant formé par les n vecteurs, c'est à dire $\mathbf{a}_1 \dots \mathbf{a}_n = [\mathbf{a}_1, \dots, \mathbf{a}_n]$.

Definition:

Soit maintenant deux extenseurs \mathbf{A} de degré k et \mathbf{B} de degré l , tel que $k + l > n$. On va définir l'opération appelée intersection, notée par \wedge par :

$$\mathbf{A} \wedge \mathbf{B} = \sum_{\sigma} \text{sgn}(\sigma) [a_{\sigma(1)} \dots a_{\sigma(n-l)} b_1 \dots b_l] a_{\sigma(n-l+1)} \dots a_{\sigma(k)}$$

où σ est une permutation de $\{1, 2, \dots, d\}$ telle que $\sigma(1) < \sigma(2) < \dots < \sigma(n-l)$ et $\sigma(n-l+1) < \dots < \sigma(k)$.

il est évident que $\mathbf{A} \wedge \mathbf{B}$ est un extenseur qui appartient à $\Lambda^{k+l-n}(E^n)$. A l'aide de ces opérations, on peut définir l'algèbre double :

Definition:

L'espace Cayley $(E^n, [.,.])$ doté des opérations \vee et \wedge est appelé l'algèbre Grassmann-Cayley ou algèbre double.

Nous serons souvent amenés à travailler dans l'espace dual E^{*n} des formes linéaires sur E^n . Nous pouvons définir de la même manière l'algèbre de Grassmann-Cayley sur l'espace dual. Le noyau d'un élément H de E^{*n} est un hyperplan de E^n . Nous associons aussi à l'élément $L = H_1 \vee H_2 \vee \dots \vee H_k$ l'espace linéaire L d'hyperplans H tels que $H \vee H_1 \vee H_2 \vee \dots \vee H_k = 0$. Si nous notons \bar{L} l'ensemble des vecteurs appartenant au noyau de H pour $H \in L$, alors $\bar{L} = \cap_{i=1}^k \bar{H}_i$.

On voit alors que dans le dual, les opérations de réunion et d'intersection sont inversées. Cela correspond à la relation bien connue :

$$(A \cap B)^{\perp} = A^{\perp} + B^{\perp}$$

Pour plus de clarté, donnons quelques exemples dans les espaces que nous utiliserons par la suite, c'est à dire \mathcal{P}^2 and \mathcal{P}^3 . La liste exhaustive des propriétés se trouve dans [Csurka, 1996]. Dans \mathcal{P}^2 , les objets sont des points du plan. Leur réunion est une droite du plan, c'est à dire un objet de l'espace dual de \mathcal{P}^2 , que nous pouvons identifier immédiatement avec un objet de \mathcal{P}^2 en utilisant la bijection canonique entre un espace et son espace dual. Dans

\mathcal{P}^2 , les opérations \vee et \wedge sont identiques et équivalentes au produit vectoriel \times .

Dans \mathcal{P}^3 , l'identification entre l'espace et son dual identifie des plans et des points. La réunion de deux points est une droite, tandis que l'intersection de deux plans est aussi une droite.

De plus amples détails peuvent être trouvés dans la littérature de vision par ordinateur [Carlsson, 1994, Csurka, 1996] ou bien dans la littérature mathématique [Sturmfels, 1993, Barnabei et al., 1985].

2.4.2 Les N -linéarités

L'équation de projection de la première caméra s'écrit

$$\mathbf{m}_1 = \mathbf{P}_1 \mathbf{M} \quad (2.24)$$

avec $\mathbf{m}_1 = (x_1, y_1, z_1)^T$. L'équation (2.24) est équivalente aux trois équations non indépendantes:

$$\begin{aligned} x_1 \mathbf{2.M} - y_1 \mathbf{1.M} &= 0 \\ y_1 \mathbf{3.M} - z_1 \mathbf{2.M} &= 0 \\ z_1 \mathbf{1.M} - x_1 \mathbf{3.M} &= 0 \end{aligned} \quad (2.25)$$

En effet, par définition, l'égalité de deux grandeurs projectives \mathbf{a} et \mathbf{b} est équivalente à $\mathbf{a} = \lambda \mathbf{b}$. Dans le cas de points de \mathcal{P}^2 , ceci est aussi équivalent à $\mathbf{a} \times \mathbf{b} = 0$. Ce sont les trois équations décrites ci dessus. Ces équations ne sont pas indépendantes puisque z_1 fois la première plus x_1 fois la deuxième est proportionnel à la troisième fois y_1 .

On choisit deux équations parmi les trois pour chaque caméra et on les regroupe dans une matrice. Nous avons choisi les deux dernières, mais tout autre choix conduit aux mêmes résultats puisque ces équations sont linéairement dépendantes et que nous allons nous intéresser à la nullité de divers déterminants. On obtient alors par exemple pour quatre caméras l'équation matricielle

$$\mathbf{AM} = \begin{bmatrix} y_1 \mathbf{3} & - & z_1 \mathbf{2} \\ z_1 \mathbf{1} & - & x_1 \mathbf{3} \\ y_2 \mathbf{6} & - & z_2 \mathbf{5} \\ z_2 \mathbf{4} & - & x_2 \mathbf{6} \\ y_3 \mathbf{9} & - & z_3 \mathbf{8} \\ z_3 \mathbf{7} & - & x_3 \mathbf{9} \\ y_4 \mathbf{12} & - & z_3 \mathbf{11} \\ z_4 \mathbf{10} & - & x_3 \mathbf{12} \end{bmatrix} \mathbf{M} = 0 \quad (2.26)$$

L'existence d'une solution \mathbf{M} à cette équation implique que tous les mineurs d'ordre 4 de la matrice sont nuls. Chaque mineur va donc nous donner une équation différente. Il existe plusieurs choix possibles: Nous pouvons choisir deux lignes relatives à une image et deux lignes relatives à une autre, deux lignes relatives à une image et une relative à deux autres, ou bien encore une ligne pour quatre images différentes. Nous venons de décrire toutes les combinaisons possibles. Il ne peut intervenir plus de quatre images dans la même équation. Nous sommes donc déjà rassurés en ce qui concerne l'explosion possible de N -linéarités: il n'y a rien au delà des quadrilinéarités. Il est maintenant clair que le choix des deux équations parmi les trois

L'interprétation des vecteurs $\mathbf{1}, \mathbf{2}, \mathbf{3}$ est assez simple. Ce sont des plans passant par le centre optique de la première caméra C_1 . $\mathbf{3}$ représente le plan focal, $\mathbf{1}$ (respectivement $\mathbf{2}$) représente le plan passant par C_1 et ayant comme image \mathbf{u} (respectivement \mathbf{v}), si notre repère image est affine. \mathbf{u} et \mathbf{v} sont les vecteurs définis à la section 2.1.1 comme les axes de coordonnées des pixels dans l'image.

2.4.2.1 Les bilinéarités

Choisissons tout d'abord le mineur formé par les quatre premières lignes. Après un simple développement, la nullité de ce mineur se traduit en

$$z_1 z_2 \mathbf{m}_2^T \mathbf{F}_{12} \mathbf{m}_1 = 0 \quad (2.27)$$

avec

$$\mathbf{F}_{12} = \begin{bmatrix} [\mathbf{2}, \mathbf{3}, \mathbf{5}, \mathbf{6}] & -[\mathbf{1}, \mathbf{3}, \mathbf{5}, \mathbf{6}] & [\mathbf{1}, \mathbf{2}, \mathbf{5}, \mathbf{6}] \\ -[\mathbf{2}, \mathbf{3}, \mathbf{4}, \mathbf{6}] & [\mathbf{1}, \mathbf{3}, \mathbf{4}, \mathbf{6}] & -[\mathbf{1}, \mathbf{2}, \mathbf{4}, \mathbf{6}] \\ [\mathbf{2}, \mathbf{3}, \mathbf{4}, \mathbf{5}] & -[\mathbf{1}, \mathbf{3}, \mathbf{4}, \mathbf{5}] & [\mathbf{1}, \mathbf{2}, \mathbf{4}, \mathbf{5}] \end{bmatrix} \quad (2.28)$$

où $[\mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l}]$ est le déterminant de ces 4 vecteurs.

Le facteur $z_1 z_2$ provient de notre choix des équations. Un autre choix d'équations aurait conduit à un facteur différent, mais à un polynôme $\mathbf{m}_2^T \mathbf{F}_{12} \mathbf{m}_1$ identique. On vérifie en le développant et en simplifiant les déterminants de manière algébrique ([Mourrain et Stolfi, 1994] ou [Sturmfels, 1993]) que le déterminant de \mathbf{F}_{12} est bien nul. Ces calculs sont en fait une réécriture des polynômes en les déterminants de matrices 4×4 en une forme normale. Cette forme normale est nulle si et seulement si le polynôme est une conséquence des relations de Plücker.

Les épipoles se calculent de manière simple comme le produit vectoriel de deux des colonnes ou de deux des lignes. Après quelques réécritures, on obtient

$$\mathbf{e}_{12} = [[\mathbf{1}, \mathbf{4}, \mathbf{5}, \mathbf{6}], [\mathbf{2}, \mathbf{4}, \mathbf{5}, \mathbf{6}], [\mathbf{3}, \mathbf{4}, \mathbf{5}, \mathbf{6}]]^T \quad (2.29)$$

$$\mathbf{e}_{21} = [[\mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{4}], [\mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{5}], [\mathbf{1}, \mathbf{2}, \mathbf{3}, \mathbf{6}]]^T \quad (2.30)$$

On peut démontrer ces expressions de manière différente : En effet, C_2 est représenté par le vecteur $\mathbf{4} \wedge \mathbf{5} \wedge \mathbf{6}$ puisque ce vecteur doit être le noyau de \mathbf{P}_2 . L'image d'un point M de l'espace dans la caméra 1 est $[\mathbf{1.M}, \mathbf{2.M}, \mathbf{3.M}]^T$. Ici, l'image de C_2 dans la caméra 1, c'est à dire

$$\mathbf{e}_{12} = [[\mathbf{1}, \mathbf{4}, \mathbf{5}, \mathbf{6}], [\mathbf{2}, \mathbf{4}, \mathbf{5}, \mathbf{6}], [\mathbf{3}, \mathbf{4}, \mathbf{5}, \mathbf{6}]]^T$$

Une autre expression est obtenue de la même manière avec $C_1 = \mathbf{1} \wedge \mathbf{2} \wedge \mathbf{3}$. On vérifie bien sûr que l'on a toujours $\mathbf{F}_{12} \mathbf{e}_{12} = 0$ et $\mathbf{F}_{12}^T \mathbf{e}_{21} = 0$.

L'interprétation géométrique de ces bilinéarités est simple : Étant donné deux points se correspondant dans deux images, les rayons issus de ces points et passant par les centres optiques doivent se couper. En effet, dire que le déterminant de la matrice dont les lignes sont constituées de quatre plans est nul est équivalent à dire que ses quatre plans se coupent en un point unique. Les quatre plans sont ici deux plans passant par C_1 et deux plans passant par C_2 . Les plans engendrés par une caméra ne sont pas confondus, sinon les deux lignes correspondant à cette caméra seraient liées. Ces deux paires de plans définissent donc deux droites qui se coupent en un point, le point d'intersection des quatre plans.

Cette contrainte est exactement la contrainte que nous avons ici. Elle est plus connue sous le nom de contrainte épipolaire comme nous l'avons mentionné à la section 2.2.

2.4.2.2 Les trilinearités

Si nous prenons maintenant deux lignes relatives à une image puis deux autres relatives à deux autres images, nous obtenons une relation trilinéaire en les coordonnées de \mathbf{m} dans les 3 images.

Choisissons par exemple les deux lignes impliquant la première caméra et la première ligne de la deuxième et de la troisième caméra dans l'équation (2.26). En développant le déterminant on obtient:

$$\begin{aligned}
0 &= (y_1 z_1 \mathbf{3} \wedge \mathbf{1} + z_1^2 \mathbf{1} \wedge \mathbf{2} + x_1 z_1 \mathbf{2} \wedge \mathbf{3}) \wedge (y_2 \mathbf{6} - z_2 \mathbf{5}) \wedge (y_3 \mathbf{9} - z_3 \mathbf{8}) \\
&= z_1 (-y_1 y_2 y_3 [\mathbf{1}, \mathbf{3}, \mathbf{6}, \mathbf{9}] + y_1 y_2 z_3 [\mathbf{1}, \mathbf{3}, \mathbf{6}, \mathbf{8}] \\
&\quad + y_1 y_3 z_2 [\mathbf{1}, \mathbf{3}, \mathbf{5}, \mathbf{9}] - y_1 z_2 z_3 [\mathbf{1}, \mathbf{3}, \mathbf{5}, \mathbf{8}] \\
&\quad + y_2 y_3 z_1 [\mathbf{1}, \mathbf{2}, \mathbf{6}, \mathbf{9}] - y_1 z_1 z_3 [\mathbf{1}, \mathbf{2}, \mathbf{6}, \mathbf{8}] \\
&\quad - y_3 z_1 z_2 [\mathbf{1}, \mathbf{2}, \mathbf{5}, \mathbf{9}] + z_1 z_2 z_3 [\mathbf{1}, \mathbf{2}, \mathbf{5}, \mathbf{8}] \\
&\quad + x_1 y_2 y_3 [\mathbf{2}, \mathbf{3}, \mathbf{6}, \mathbf{9}] - x_2 y_2 z_3 [\mathbf{2}, \mathbf{3}, \mathbf{6}, \mathbf{8}] \\
&\quad - x_1 y_3 z_2 [\mathbf{2}, \mathbf{3}, \mathbf{5}, \mathbf{9}] + x_1 z_2 z_3 [\mathbf{2}, \mathbf{3}, \mathbf{5}, \mathbf{8}])
\end{aligned}$$

On ne tient pas compte du facteur z_1 qui est génériquement non nul. Géométriquement, cette équation signifie que le rayon $\mathbf{C}_1 \mathbf{m}_1$, le plan passant par la droite $\mathbf{5} \wedge \mathbf{6}$ et \mathbf{m}_2 ainsi que le plan passant par $\mathbf{8} \wedge \mathbf{9}$ et \mathbf{m}_3 se coupent (voir figure 2.7).

Avec trois caméras, on peut construire 4 relations indépendantes de ce type, que nous n'énumérerons pas ici pour ne pas ennuyer le lecteur. Il y a deux choix possibles pour la ligne de la deuxième image, deux choix pour la ligne de la troisième image.

À la section 2.3.2, nous avons vu qu'il existait 9 trilinearités qui s'expriment à partir du tenseur T . Ces 9 trilinearités sont combinaisons linéaires des quatre que nous avons obtenus précédemment. En effet, les neuf trilinearités s'obtiennent en prenant les trois lignes de (2.25) au lieu de deux d'entre elles. On a alors deux fois le choix entre trois possibilités au lieu de deux fois le choix entre deux possibilités. Les équations de (2.25) étant linéairement dépendantes, les déterminants qui en sont issus le sont aussi. Nous verrons plus tard à la figure 2.7 que les trilinearités sont en fait issues de deux faisceaux de plans autour des axes $\mathbf{C}_2 \mathbf{m}_2$ et $\mathbf{C}_3 \mathbf{m}_3$. Ces faisceaux de plans peuvent être représentés par deux plans distincts appartenant au faisceau. Plusieurs choix sont bien entendu possibles. C'est le choix de deux combinaisons linéaires des équations (2.25) qui détermine les deux plans. On peut exprimer ces relations pour tous les plans du faisceau. La représentation tensorielle des trilinearités correspond à trois plans du faisceau.

Les trilinearités obtenues à partir des déterminants se classent aussi en trois groupes selon la caméra qui est particularisée. Ici, deux lignes du déterminant comportent des termes de cette caméra. On numérote les équations en fonction des lignes qu'elles font intervenir. Celle explicitée plus haut est par exemple numérotée $t(1, 2, 3, 5)$. Les bilinéarités peuvent aussi être numérotées de la même façon : l'équation $\mathbf{m}_2^T \mathbf{F}_{12} \mathbf{m}_1 = 0$ est numérotée $t(1, 2, 3, 4)$.

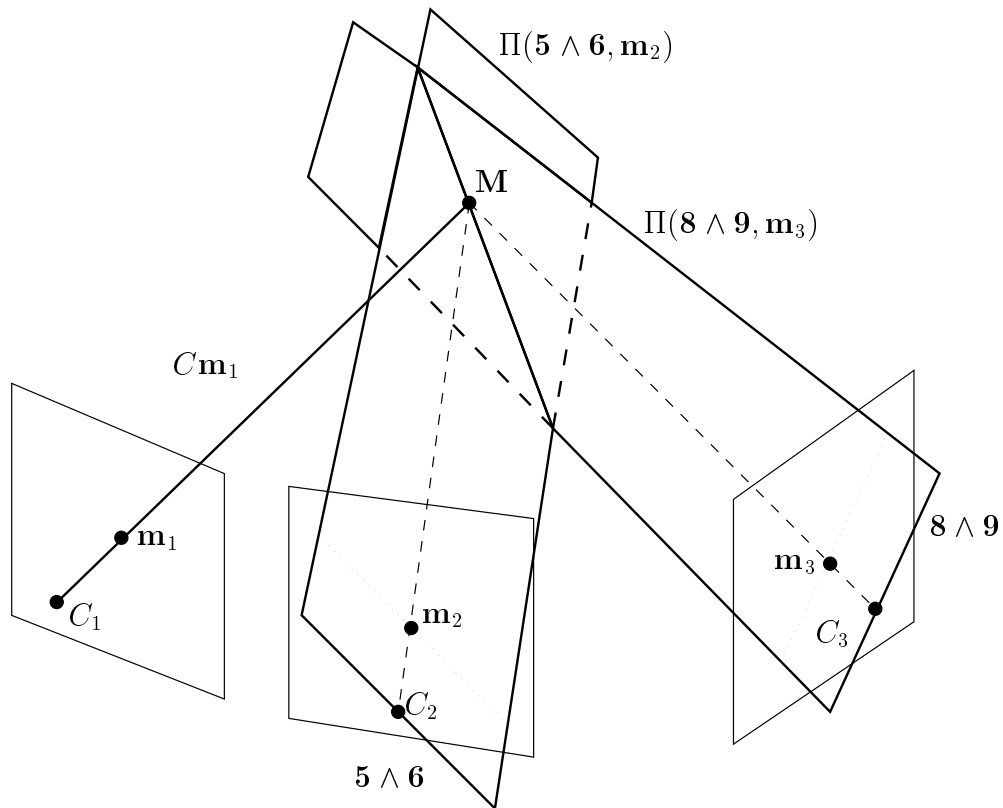


FIG. 2.7 – *Interprétation géométrique d'une trilinearité. La condition est que la droite passant par C_1 et m_1 , le plan $\Pi(\mathbf{5} \wedge \mathbf{6}, m_2)$ construit à partir de la droite $\mathbf{5} \wedge \mathbf{6}$ et du point m_2 , et le plan $\Pi(\mathbf{8} \wedge \mathbf{9}, m_3)$ se coupent en un point de l'espace.*

2.4.2.3 Les quadrilinéarités

Prenons maintenant une ligne par image dans notre déterminant. Nous obtenons alors une relation quadrilinéaire en $\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3$ et \mathbf{m}_4 . Nous allons démontrer que ce type de relation est toujours une combinaison linéaire de bilinéarités et de trilinearités.

Considérons donc la matrice \mathbf{A} $2N \times 4$ présentée à l'équation (2.26). Écrire que tous les mineurs d'ordre 4 sont nuls est équivalent à écrire que la matrice est de rang 3. Pour prouver ceci, il est suffisant de démontrer que toutes les lignes sont combinaisons linéaires de 3 lignes fixées. Ceci peut être fait en calculant le déterminant obtenu par le choix de trois lignes, plus d'une quatrième appartenant aux $2N - 3$ restantes.

Choisissons les deux premières lignes de la matrice comme les deux premières lignes fixes de notre déterminant. Elles ne sont pas collinéaires parce que le rang de \mathbf{P}_1 est 3. La troisième ligne est choisie comme n'étant pas combinaison linéaire des deux premières. Le choix de cette troisième ligne fixe est toujours possible, sinon le rang de notre matrice \mathbf{A} serait 2. Nous reviendrons sur ce cas particulier.

Maintenant, tous les déterminants écrits de cette façon sont des trilinearités ou des bilinéarités (juste dans le cas où la quatrième ligne provient de la même image que la troisième). Tous les autres déterminants, c'est à dire les contraintes quadrilinéaires, mais aussi les bilinéaires et trilinearités pas encore calculés sont combinaison linéaire des expressions déjà calculées. La conséquence algébrique est que toute l'information est contenue dans une bilinéarité et $2N - 4$ trilinearités. Les quadrilinéarités sont des combinaisons linéaires à coefficients de celles ci.

Considérons maintenant le cas particulier où \mathbf{A} est de rang 2. Cela signifie que \mathbf{M} n'est pas un point unique, mais est juste contraint sur une droite. Ce cas correspond à une configuration où tous les centres optiques et le point \mathbf{M} sont alignés.

2.4.3 Trilinearités sur les droites

Le cas des droites est important car elles sont souvent présentes dans les images que nous avons à traiter et leur détection est relativement plus stable que celle des points.

Nous allons devoir introduire des matrices de projection pour droites. Étant donné deux point \mathbf{M}_1 et \mathbf{M}_2 définissant à une droite Λ de l'espace, on représente Λ par un vecteur à 6 dimensions de coordonnées de Plücker $\Lambda = \mathbf{M}_1 \wedge \mathbf{M}_2 = [\Lambda^{12}, \Lambda^{13}, \Lambda^{14}, \Lambda^{23}, \Lambda^{24}, \Lambda^{34}]^T$. Λ est défini à un facteur d'échelle près. L'image λ de Λ est représentée par le vecteur $\boldsymbol{\lambda}$ tel que

$$\boldsymbol{\lambda} = \mathbf{P}\mathbf{M}_1 \wedge \mathbf{P}\mathbf{M}_2 \quad (2.31)$$

On note que dans \mathcal{P}^2 , les opérateurs \wedge et \times sont équivalents. En développant l'équation précédente, on obtient :

$$\begin{aligned} \boldsymbol{\lambda} = & ((\mathbf{2}.\mathbf{M}_1)(\mathbf{3}.\mathbf{M}_2) - (\mathbf{3}.\mathbf{M}_2)(\mathbf{2}.\mathbf{M}_1), \\ & (\mathbf{3}.\mathbf{M}_1)(\mathbf{1}.\mathbf{M}_2) - (\mathbf{1}.\mathbf{M}_1)(\mathbf{3}.\mathbf{M}_2), \\ & (\mathbf{1}.\mathbf{M}_1)(\mathbf{2}.\mathbf{M}_2) - (\mathbf{2}.\mathbf{M}_1)(\mathbf{1}.\mathbf{M}_2)) \end{aligned}$$

Chaque coordonnée de ce vecteur est antisymétrique en $\mathbf{M}_1, \mathbf{M}_2$. De plus, la première coordonnée est antisymétrique en $\mathbf{2}, \mathbf{3}$, la deuxième en $\mathbf{3}, \mathbf{1}$ et la troisième en $\mathbf{1}, \mathbf{2}$. Il est facile de vérifier que $\boldsymbol{\lambda}$ a pour coordonnées

$$\begin{aligned} \boldsymbol{\lambda} = & [(\mathbf{2} \wedge \mathbf{3})^T \Lambda, \\ & (\mathbf{3} \wedge \mathbf{1})^T \Lambda, \\ & (\mathbf{1} \wedge \mathbf{2})^T \Lambda]^T \end{aligned}$$

Nous pouvons réécrire cette équation linéairement en fonction des coordonnées de Plücker :

$$\boldsymbol{\lambda} = \hat{\mathbf{P}}\Lambda \quad (2.32)$$

où $\hat{\mathbf{P}}$ est une matrice 3×6 définie à un facteur d'échelle près. Les rangées de $\hat{\mathbf{P}}$ sont $\mathbf{2} \wedge \mathbf{3}$, $\mathbf{3} \wedge \mathbf{1}$ et $\mathbf{1} \wedge \mathbf{2}$. La matrice $\hat{\mathbf{P}}$ joue le même rôle pour les droites que \mathbf{P} pour les points. De la même manière que pour les points, on obtient un système d'équations du type

$$\begin{bmatrix} \lambda_1^1 \mathbf{1} \wedge \mathbf{2} - \lambda_1^3 \mathbf{2} \wedge \mathbf{3} \\ \lambda_1^2 \mathbf{1} \wedge \mathbf{2} - \lambda_1^3 \mathbf{3} \wedge \mathbf{1} \end{bmatrix} \Lambda = \mathbf{0} \quad (2.33)$$

où λ_j^i est la i ème coordonnée de la droite $\boldsymbol{\lambda}$ vue dans l'image j . Chaque ligne de la matrice qui apparaît dans le terme gauche de l'équation est une droite de l'espace passant par le centre optique. Ceci est extrêmement facile à vérifier : par exemple, pour la première ligne, $\lambda_1^1 \mathbf{1} \wedge \mathbf{2} - \lambda_1^3 \mathbf{2} \wedge \mathbf{3} = -\mathbf{2} \wedge (\lambda_1^1 \mathbf{1} + \lambda_1^3 \mathbf{3})$. Si nous considérons 3 caméras, la matrice de (2.33) devient 6×6 . Chaque ligne de cette matrice est notée L_i et représente une droite de

l'espace. L_1 et L_2 passent par le centre optique de la première caméra, L_3 et L_4 par le centre optique de la deuxième caméra et L_5 et L_6 par le centre optique de la troisième. De plus, si nous considérons la réunion de L_3 et L_4 , nous obtenons le plan :

$$\begin{aligned} L_3 \vee L_4 &= (\lambda_2^1 \mathbf{4} \wedge \mathbf{5} - \lambda_2^3 \mathbf{5} \wedge \mathbf{6}) \vee (\lambda_2^2 \mathbf{4} \wedge \mathbf{5} - \lambda_2^3 \mathbf{6} \wedge \mathbf{4}) \\ &= \lambda_2^1 \mathbf{4} + \lambda_2^2 \mathbf{5} + \lambda_2^3 \mathbf{6}. \end{aligned}$$

Une dérivation identique s'opère pour $L_5 \vee L_6$. Ces deux plans se coupent en Λ . En effet, $L_3 \vee L_4$ (respectivement $L_5 \vee L_6$) représente le plan passant par Λ et le centre optique de la deuxième (respectivement troisième) caméra. Exprimons alors que L_1 et L_2 coupent Λ :

$$\begin{aligned} L_1 \wedge ((L_3 \vee L_4) \wedge (L_5 \vee L_6)) &= 0 \\ L_2 \wedge ((L_3 \vee L_4) \wedge (L_5 \vee L_6)) &= 0 \quad . \end{aligned}$$

Chacune de ces équations est un polynôme de degré 5 en les coefficients de λ_1 , λ_2 et λ_3 . En fait, puisque les 2 termes sont divisibles par $\lambda_2^3 \lambda_3^3$, ces deux conditions sont de degré 3. Elles s'expriment de la manière suivante :

$$\begin{aligned} (\lambda_1^1 \mathbf{1} \wedge \mathbf{2} - \lambda_1^3 \mathbf{2} \wedge \mathbf{3}) \wedge (\lambda_2^1 \mathbf{4} + \lambda_2^2 \mathbf{5} + \lambda_2^3 \mathbf{6}) \wedge (\lambda_3^1 \mathbf{7} + \lambda_3^2 \mathbf{8} + \lambda_3^3 \mathbf{9}) &= 0 \\ (\lambda_1^2 \mathbf{1} \wedge \mathbf{2} - \lambda_1^3 \mathbf{3} \wedge \mathbf{1}) \wedge (\lambda_2^1 \mathbf{4} + \lambda_2^2 \mathbf{5} + \lambda_2^3 \mathbf{6}) \wedge (\lambda_3^1 \mathbf{7} + \lambda_3^2 \mathbf{8} + \lambda_3^3 \mathbf{9}) &= 0 \end{aligned}$$

On peut réécrire ces équations de la façon suivante :

$$\lambda_1^1 \sum_{i,j=1}^3 T_{3ij} \lambda_2^i \lambda_3^j - \lambda_1^3 \sum_{i,j=1}^3 T_{1ij} \lambda_2^i \lambda_3^j = 0 \quad (2.34)$$

$$\lambda_1^2 \sum_{i,j=1}^3 T_{3ij} \lambda_2^i \lambda_3^j - \lambda_1^3 \sum_{i,j=1}^3 T_{2ij} \lambda_2^i \lambda_3^j = 0 \quad (2.35)$$

On en tire l'expression plus compacte :

$$\lambda_1^i = \sum_{j,k} T_{ijk} \lambda_2^j \lambda_3^k \quad (2.36)$$

On reconnaît l'équation (2.22). Des relations symétriques peuvent être obtenues en considérant les caméras 2 et 3. Il faut noter que nous n'avons pas ici directement utilisé les déterminants, mais plutôt les propriétés des droites dans l'algèbre de Grassmann-Cayley.

2.4.4 Différentes paramétrisations minimales

Nous avons vu que les quadrilinéarités étaient des combinaisons linéaires des bilinéarités et des trilinearités, nous pouvons donc les exclure de notre étude puisqu'elles n'apportent pas d'information. Nous avons une description minimale des bilinéarités grâce à l'étude de la matrice fondamentale. Toutefois, les contraintes entre les coefficients des trilinearités restent un peu mystérieuses. Le tenseur T comporte 27 éléments définis à une constante multiplicative près, c'est à dire 26 paramètres tandis que la géométrie projective de 3 caméras ne dépend que de 18 paramètres. Il existe un grand nombre de descriptions minimales de ce tenseur en 18 paramètres. Nous en développerons deux :

- La description à base de matrices de projection
- La description à base de contraintes internes

La description à base de matrices de projection est implicitement exposée à la section 2.5 puisque nous y étudierons la transformation d'un tenseur trifocal en un jeu de matrices de projection. La description à base de contraintes internes permet plus de flexibilité dans le choix de la paramétrisation.

La description à base de contraintes est simplement basée sur le fait que le tenseur trifocal possède 27 paramètres définis à un facteur d'échelle près. Or, le fait que ce tenseur soit issu de trois matrices de projection qui ne dépendent, elles, que de 18 paramètres impose des contraintes. Nous avons donc $27 - 18 = 9$ contraintes internes au tenseur. Ces contraintes devront toujours être vérifiées pour que le tenseur T soit effectivement un tenseur trifocal.

Il est clair géométriquement (voir la figure 2.8) et cela peut être vérifié algébriquement que pour toutes les trilinearités, on a

$$\begin{aligned}\forall \mathbf{m}_1, \quad T(\mathbf{m}_1, \mathbf{e}_{21}, \mathbf{e}_{31}) &= 0 \\ \forall \mathbf{m}_2, \quad T(\mathbf{e}_{12}, \mathbf{m}_2, \mathbf{e}_{32}) &= 0 \\ \forall \mathbf{m}_3, \quad T(\mathbf{e}_{13}, \mathbf{e}_{23}, \mathbf{m}_3) &= 0\end{aligned}$$

où $T(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3)$ représente les trilinearités au point $(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3)$.

Si l'on note \mathbf{T}_{ij} le vecteur obtenu en fixant les deux dernières coordonnées du tenseur à i et j , on démontre [Faugeras et Mourrain, 1995] que ces équations sont équivalentes aux relations suivantes :

$$\forall \{i, j, l, m\}, \quad |\mathbf{T}_{il}\mathbf{T}_{jl}\mathbf{T}_{jm}||\mathbf{T}_{jm}\mathbf{T}_{im}\mathbf{T}_{il}| - |\mathbf{T}_{im}\mathbf{T}_{jm}\mathbf{T}_{jl}||\mathbf{T}_{jl}\mathbf{T}_{il}\mathbf{T}_{im}| = 0 \quad (2.37)$$

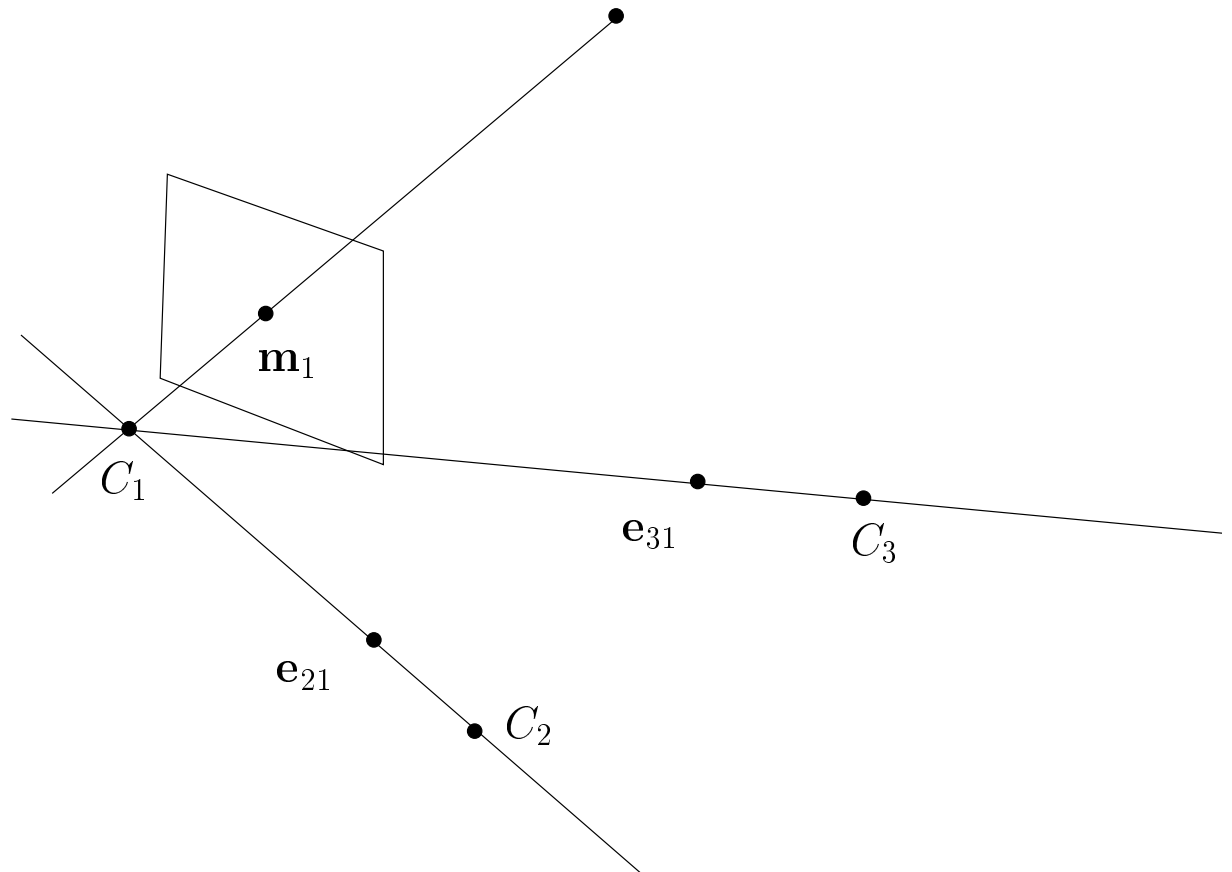


FIG. 2.8 – Les épipoles vérifient les trilinearités avec un point quelconque. En effet, e_{21} et C_2 , e_{31} et C_3 définissent des faisceaux de plans passant par C_1 . Tous les rayons issus du centre optique C_1 coupent donc ces deux faisceaux de plans en un point unique qui est le centre optique de la première caméra.

Ces équations sont trivialement vérifiées pour $i = j$ et pour $l = m$. De plus, échanger les paramètres i et j ou bien l et m conduit à des équations identiques. On peut donc considérer sans restriction que l'on a $i < j$ et $l < m$. On obtient alors 9 contraintes différentes et indépendantes.

Si nous exprimons maintenant par exemple \mathbf{T}_{il} comme fonction linéaire de \mathbf{T}_{jl} , \mathbf{T}_{jm} et \mathbf{T}_{im} qui sont génériquement indépendants

$$\mathbf{T}_{il} = \alpha_{jl}^{il} \mathbf{T}_{jl} + \alpha_{jm}^{il} \mathbf{T}_{jm} + \alpha_{im}^{il} \mathbf{T}_{im}$$

alors l'équation (2.37) se simplifie en :

$$\alpha_{im}^{il} \alpha_{jl}^{il} = \alpha_{jm}^{il},$$

ce qui nous permet de paramétriser \mathbf{T}_{il} en fonction de trois autres vecteurs en utilisant deux paramètres $(\alpha_{im}^{il}, \alpha_{jl}^{il})$ au lieu de trois $(\alpha_{im}^{il}, \alpha_{jl}^{il}, \alpha_{jm}^{il})$. En répétant cette procédure, on obtient une paramétrisation minimale de T .

2.4.5 Lien avec les descriptions de Hartley et de Shashua

Dans [Hartley, 1994a], Richard Hartley démontre que les relations issues de son tenseur trifocal sont identiques aux trilinearités d'Amnon Shashua. Nous nous contenterons alors de démontrer que les trilinearités issues des déterminants sont les mêmes que celles de Richard Hartley. L'identification avec les trilinearités de Shashua s'opérera par transitivité.

Hartley utilise le tenseur T que nous avons introduit à la section 2.3.2. Ses équations ont la forme suivante

$$\sum_{k=1}^3 \mathbf{m}_1^k (\mathbf{m}_2^i \mathbf{m}_3^l T_{kjm} - \mathbf{m}_2^j \mathbf{m}_3^l T_{kim} - \mathbf{m}_2^i \mathbf{m}_3^m T_{kjl} + \mathbf{m}_2^j \mathbf{m}_3^m T_{kil}) \quad (2.38)$$

avec $(i, j, l, m) \in \{1, 2, 3\}^4$. Il y a une équation pour chaque choix d'indices tel que $i \neq j$ et $l \neq m$. On obtient des équations identiques si l'on inverse i et j ou l et m . Il y a donc 9 équations différentes donc 4 seulement sont indépendantes.

En identifiant ces équations avec celles issues des déterminants obtenues à partir des deux lignes correspondant à la première image et une ligne dans chacune des deux images restantes, on remarque que T_{ijk} est construit en prenant l'élément i de la première colonne, l'élément j de la deuxième colonne, l'élément k de la troisième colonne de la matrice suivante:

$$\begin{bmatrix} \mathbf{2} \wedge \mathbf{3} & \mathbf{4} & \mathbf{7} \\ \mathbf{3} \wedge \mathbf{1} & \mathbf{5} & \mathbf{8} \\ \mathbf{1} \wedge \mathbf{2} & \mathbf{6} & \mathbf{9} \end{bmatrix} \quad (2.39)$$

Par exemple, $T_{111} = [\mathbf{2}, \mathbf{3}, \mathbf{4}, \mathbf{7}]$. Ceci montre l'équivalence entre les trilinearités de Hartley et celles induites par les déterminants, et par transitivité avec celles de Shashua.

On vérifie aussi que si l'on choisit $\mathbf{P}_1 = [\mathbf{I}|\mathbf{0}]$, $\mathbf{P}_2 = [a_{ik}]$ et $\mathbf{P}_3 = [b_{ik}]$, les formules de déterminants se simplifient. Ainsi

$$\begin{aligned} [\mathbf{2}, \mathbf{3}, \mathbf{i}, \mathbf{j}] &= a_{i1}b_{j4} - a_{i4}b_{j1} \\ [\mathbf{1}, \mathbf{3}, \mathbf{i}, \mathbf{j}] &= a_{i2}b_{j4} - a_{i4}b_{j2} \\ [\mathbf{1}, \mathbf{2}, \mathbf{i}, \mathbf{j}] &= a_{i3}b_{j4} - a_{i4}b_{j3} \end{aligned}$$

donne les mêmes expressions que l'équation (2.16). On a aussi l'identification entre équations. En effet, pour les trilinearités du premier groupe, $t(1, 2, 3, 5)$ correspond à l'équation de Hartley (2.38) en choisissant $i = 1, j = 3, l = 1$ et $m = 3$, $t(1, 2, 3, 6)$ à $i = 1, j = 3, l = 2$ et $m = 3$, $t(1, 2, 4, 5)$ à $i = 2, j = 3, l = 1$ et $m = 3$, et finalement $t(1, 2, 4, 6)$ correspond au choix $i = 2, j = 3, l = 2$ et $m = 3$.

2.5 Relations entre les différentes entités

Toutes les entités que nous avons manipulées jusqu'à présent ont l'avantage d'opérer linéairement sur les points de l'image. Le prix à payer est qu'elles ne sont pas exprimables sans contraintes, mais surtout que leur nombre n'est pas proportionnel au nombre de caméras, comme la formule 2.8. C'est pourquoi il est particulièrement intéressant de revenir au formalisme des matrices de projection. Elles contiennent toute l'information relative à la géométrie du système et ne sont pas contraintes entre elles. Il est extrêmement aisé de passer des matrices de projection (que nous noterons $\mathbf{P} = [\mathbf{Q}|\mathbf{t}]$) aux matrices fondamentales par la formule suivante

$$\mathbf{F}_{ij} = [\mathbf{t}_j - \mathbf{Q}_j \mathbf{Q}_i^{-1} \mathbf{t}_i]_{\times} \mathbf{Q}_i \mathbf{Q}_j^{-1} \quad (2.40)$$

Cette relation est vraie si \mathbf{Q}_i et \mathbf{Q}_j sont inversibles.

Le passage des matrices de projection aux trilinearités s'effectue grâce aux formules 2.16 si $\mathbf{P}_1 = [\mathbf{I}|\mathbf{0}]$ ou aux formules 2.39 sinon. Toutefois, la transformation des bi- et trilinearités en matrices de projection est plus complexe, d'un point de vue théorique *et* d'un point de vue pratique.

2.5.1 De \mathbf{F} vers \mathbf{P}

Dans cette section nous supposons connues toutes les matrices fondamentales \mathbf{F}_{ij} entre les images et nous cherchons un jeu possible de matrices de projection (définies à une transformation projective de \mathcal{P}^3 près). Nous distinguerons deux cas, selon que les centres optiques sont alignés (épipoles confondus) ou en position générale. Encore une fois (comme à la section 2.3.1), cette distinction est amenée par une étude approfondie des cas pratiques. Cette distinction n'existe pas dans les travaux de Luong et Viéville [Luong et Viéville, 1994]. Leurs calculs ne s'appliquent pas si les centres optiques sont alignés.

Nous restreindrons nos démonstrations aux cas de trois caméras par simplicité, $\mathbf{P}_i, i > 3$ pouvant être calculé de la même manière que \mathbf{P}_3 .

2.5.1.1 Cas où les épipoles sont disjoints

Nous avons vu à la section 2.3.1 que, dans ce cas, les matrices fondamentales dépendent de 18 paramètres, exactement comme les matrices de projection. Toutefois, en pratique, les 21 paramètres ne vérifient pas exactement les trois contraintes. Ils nous faudra en tenir compte dans nos calculs.

Tout d'abord, nous pouvons choisir

$$\begin{aligned}\mathbf{P}_1 &= [\mathbf{I}|\mathbf{0}] \\ \mathbf{P}_2 &= [[\mathbf{e}_{21}]_{\times} \mathbf{F}_{12} | \mathbf{e}_{21}] \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \alpha^T & \alpha_4 \end{bmatrix}\end{aligned}$$

Ce choix prend en compte la contrainte épipolaire symbolisée par \mathbf{F}_{12} . En effet, si \mathbf{F} est la matrice fondamentale issue de \mathbf{P}_1 et \mathbf{P}_2 , en utilisant les formules 2.40,

$$\begin{aligned}\mathbf{F} &= [\alpha_4 \mathbf{e}_{21}]_{\times} ([\mathbf{e}_{21}]_{\times} \mathbf{F}_{12} + \mathbf{e}_{21} \alpha^T) \\ &= [\mathbf{e}_{21}]_{\times} [\mathbf{e}_{21}]_{\times} \mathbf{F}_{12}\end{aligned}$$

En utilisant le fait que pour tout \mathbf{x} ,

$$\begin{aligned}
\mathbf{F}\mathbf{x} &= \mathbf{e}_{21} \times (\mathbf{e}_{21} \times (\mathbf{F}_{12}\mathbf{x})) \\
&= (\mathbf{e}_{21}^T \mathbf{F}_{12}\mathbf{x})\mathbf{e}_{21} - (\mathbf{e}_{21}^T \mathbf{e}_{12})\mathbf{F}_{12}\mathbf{x} \\
&= \mathbf{F}_{12}\mathbf{x}
\end{aligned}$$

puisque $\mathbf{e}_{21}^T \mathbf{F}_{12}\mathbf{x} = 0$ pour tout \mathbf{x} .

Notons $\bar{\alpha}$ le vecteur $[\alpha^T \ \alpha_4]^T$. \mathbf{P}_1 et \mathbf{P}_2 vérifient bien la géométrie épipolaire imposée. La présence du vecteur $\bar{\alpha}$ permet de tenir compte des degrés de liberté existant. En effet, nous avons choisi arbitrairement les 11 paramètres de la matrice \mathbf{P}_1 , la matrice fondamentale détermine 7 paramètres de \mathbf{P}_2 , il reste donc 4 paramètres libres que nous regroupons dans le vecteur $\bar{\alpha}$. Le vecteur $\bar{\alpha}$ n'intervient bien sûr pas dans l'expression de \mathbf{F} . On remarque aussi que le centre optique de la première caméra est représenté par $[0, 0, 0, 1]^T$

Ensuite, nous montrons que, à partir d'un choix arbitraire de $\bar{\alpha}$, la forme de \mathbf{P}_3 est déterminée sans ambiguïté.⁶ De la même manière que nous avons contraint \mathbf{P}_2 à l'aide de la première caméra, exprimons \mathbf{P}_3 .

$$\mathbf{P}_3 = [[\mathbf{e}_{31}]_{\times} \mathbf{F}_{13} | \mathbf{e}_{31}] \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \beta^T & \beta_4 \end{bmatrix} \quad (2.41)$$

Comme nous l'avons montré plus haut, cette forme nous assure que les contraintes épipolaires entre les images 1 et 3 sont vérifiées. Nous devons nous assurer que \mathbf{F}_{23} peut aussi se calculer à partir de \mathbf{P}_2 et \mathbf{P}_3 . Décomposons \mathbf{P}_2 et \mathbf{P}_3 sous la forme $[\mathbf{Q} | \mathbf{t}]$.

$$\begin{aligned}
\mathbf{Q}_2 &= [\mathbf{e}_{21}]_{\times} \mathbf{F}_{12} + \mathbf{e}_{21} \alpha^T \\
\mathbf{t}_2 &= \alpha_4 \mathbf{e}_{21} \\
\mathbf{Q}_3 &= [\mathbf{e}_{31}]_{\times} \mathbf{F}_{13} + \mathbf{e}_{31} \beta^T \\
\mathbf{t}_3 &= \beta_4 \mathbf{e}_{31}
\end{aligned}$$

De ces expressions, nous pouvons déduire celles de \mathbf{e}_{32} et de \mathbf{F}_{23}

$$\lambda \mathbf{e}_{32} = \mathbf{t}_3 - \mathbf{Q}_3 \mathbf{Q}_2^{-1} \mathbf{t}_2 \quad (2.42)$$

$$\mu \mathbf{F}_{23} = [\mathbf{e}_{32}]_{\times} \mathbf{Q}_3 \mathbf{Q}_2^{-1} \quad (2.43)$$

6. Géométriquement, on voit que la position de C_3 peut être déduite de la position de C_1 et C_2 avec \mathbf{e}_{13} , \mathbf{e}_{23} et la position des plans image 1 et 2.

Nous avons introduit les paramètres λ et μ pour prendre en compte l'inconnue d'échelle dans ces équations. A partir de la deuxième équation, on déduit⁷

$$\mu \mathbf{F}_{23} \mathbf{Q}_2 = [\mathbf{e}_{32}]_{\times} \mathbf{Q}_3 \quad (2.44)$$

et en développant

$$\mu \mathbf{F}_{23} ([\mathbf{e}_{21}]_{\times} \mathbf{F}_{12} + \mathbf{e}_{21} \alpha^T) = [\mathbf{e}_{32}]_{\times} ([\mathbf{e}_{31}]_{\times} \mathbf{F}_{13} + \mathbf{e}_{31} \beta^T) \quad (2.45)$$

puis

$$(\mathbf{e}_{32} \times \mathbf{e}_{31}) \beta^T = \mu \mathbf{F}_{23} [\mathbf{e}_{21}]_{\times} \mathbf{F}_{12} - [\mathbf{e}_{32}]_{\times} [\mathbf{e}_{31}]_{\times} \mathbf{F}_{13} + \mu \mathbf{F}_{23} \mathbf{e}_{21} \alpha^T \quad (2.46)$$

où, en séparant colonne par colonne

$$\beta_i (\mathbf{e}_{32} \times \mathbf{e}_{31}) = \mathbf{l}_i + \mu \alpha_i \mathbf{F}_{23} \mathbf{e}_{21} \quad (2.47)$$

Or, les vecteurs $\mathbf{e}_{32} \times \mathbf{e}_{31}$ et $\mathbf{F}_{23} \mathbf{e}_{21}$ sont proportionnels puisqu'ils représentent tous les deux la trace du plan trifocal $C_1 C_2 C_3$ sur la troisième image. Il s'agit de l'expression des contraintes qui existent entre les matrices fondamentales, comme nous l'avons décrit à la section 2.3.1. Si les matrices fondamentales sont cohérentes entre elles, il existe un μ tel que les vecteurs \mathbf{l}_i soient proportionnels à l'image du plan trifocal. Ceci est vrai parce que les deux termes de \mathbf{l}_i représentent des droites passant par \mathbf{e}_{32} ⁸ et, avec une combinaison linéaire, nous devons être capables de produire une droite qui passe à la fois par \mathbf{e}_{32} et \mathbf{e}_{31} . Nous devons donc trouver la même valeur pour μ quel que soit i .

On peut éliminer μ entre ces trois équations. L'équation 2.47 est une équation liant deux grandeurs projectives. Ces deux grandeurs projectives sont dites égales si et seulement si elles sont proportionnelles. Si elles appartiennent à \mathcal{P}^2 , alors ceci est équivalent à dire que leur produit vectoriel est nul. En transformant de cette manière l'équation 2.47, on obtient

7. Nous avons choisi \mathbf{Q}_2 de déterminant non nul, ce qui nous est autorisé par un choix approprié de α . En effet, puisque $\mathbf{Q}_2 = [\mathbf{e}_{21}]_{\times} \mathbf{F}_{12} + \mathbf{e}_{21} \alpha^T$, pour tout vecteur \mathbf{x} , $\mathbf{Q}_2 \mathbf{x}$ sera la somme d'un vecteur orthogonal à \mathbf{e}_{21} et d'un vecteur de même direction que \mathbf{e}_{21} .

8. Les deux termes de \mathbf{l}_i sont de la forme $\mathbf{e}_{32} \times \mathbf{x}$ où \mathbf{x} est soit $\beta_i \mathbf{e}_{31}$ soit $\mathbf{e}_{31} \times \mathbf{F}_{13}^i$. On a $\mathbf{e}_{32} \cdot \mathbf{l}_i = \mathbf{e}_{32} \cdot (\mathbf{e}_{32} \times \mathbf{x}) = 0$. Chaque terme de \mathbf{l}_i représente donc une droite passant par \mathbf{e}_{32} .

$$\beta_i \mathbf{u} + \alpha_i \mathbf{v} + \mathbf{w} = 0 \quad (2.48)$$

avec

$$\begin{aligned} \mathbf{u} &= (\mathbf{e}_{32} \times \mathbf{e}_{31}) \times (\mathbf{F}_{23}(\mathbf{e}_{21} \times \mathbf{F}_{12}^i)) \\ \mathbf{v} &= (\mathbf{e}_{32} \times (\mathbf{e}_{31} \times \mathbf{F}_{13}^i)) \times \mathbf{F}_{23} \mathbf{e}_{21} \\ \mathbf{w} &= (\mathbf{e}_{32} \times (\mathbf{e}_{31} \times \mathbf{F}_{13}^i)) \times (\mathbf{F}_{23}(\mathbf{e}_{21} \times \mathbf{F}_{12}^i)) \end{aligned}$$

Cette équation (2.48) vectorielle d'inconnue β_i a une solution qui peut s'exprimer ainsi :

$$\beta_i = -\frac{\mathbf{u}^T(\alpha_i \mathbf{v} + \mathbf{w})}{\mathbf{u}^T \mathbf{u}} \quad (2.49)$$

Nous pouvons donc trouver les β_i pour $i \in \{1, 2, 3\}$ à partir des α_i correspondants. Il faut toutefois noter que l'équation 2.48 est une équation entre vecteurs de dimension 3 qui n'a pas de solution en général si les vecteurs \mathbf{u} , \mathbf{v} et \mathbf{w} sont quelconques. Ici, il existe une solution par construction. Les équations liant les éléments des vecteurs \mathbf{u} , \mathbf{v} et \mathbf{w} sont induites par les contraintes existant entre les matrices fondamentales comme celles vues à la section 2.3.1. Les matrices \mathbf{Q}_2 et \mathbf{Q}_3 sont alors définies de manière unique. Nous procédons de même pour calculer β_4 en utilisant l'équation 2.42.

$$\lambda \mathbf{e}_{32} = \beta_4 \mathbf{e}_{31} - \alpha_4 \mathbf{Q}_3 \mathbf{Q}_2^{-1} \mathbf{e}_{21} \quad (2.50)$$

Encore une fois, on peut éliminer λ dans cette équation vectorielle pour obtenir β_4 en fonction de α_4 . On obtient alors

$$\beta_4(\mathbf{e}_{31} \times \mathbf{e}_{32}) - \alpha_4(\mathbf{Q}_3 \mathbf{Q}_2^{-1} \mathbf{e}_{21}) \times \mathbf{e}_{32} = 0 \quad (2.51)$$

Le calcul des β_i en fonction des α_i est donc achevé. Nous avons donc obtenu un jeu de matrices $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ qui engendrent les matrices fondamentales $\mathbf{F}_{12}, \mathbf{F}_{23}, \mathbf{F}_{31}$.

2.5.1.2 Cas où les épipoles sont confondus

La démonstration précédente ne tient plus dans le cas où les centres optiques sont alignés. En effet, il n'existe plus de plan trifocal, puisque tous les plans appartenant au faisceau défini par la droite contenant C_1 , C_2 et C_3 contiennent les 3 centres optiques.

Pour calculer les matrices de projection à partir des matrices fondamentales, nous pouvons nous servir du fait que les centres optiques se trouvent tous sur un même droite. Si nous fixons C_1 et C_2 , alors C_3 a un degré de liberté, sa position sur la droite C_1C_2 , contre 3 auparavant, ses coordonnées dans l'espace. Le système de trois caméras a donc 16 inconnues au lieu de 18. Toutefois, nous avons démontré à la section 2.3.1 que les trois matrices fondamentales ne dépendaient plus que de 15 paramètres. Il n'y a pas d'issue si nous n'injectons pas de l'information à ce stade.

L'interprétation géométrique est la suivante : C_3 est représenté dans les images 1 et 2 par \mathbf{e}_{13} et \mathbf{e}_{23} . Si C_3 n'appartient pas à la droite C_1C_2 , nous pouvons le reconstruire. Cependant, aucun point de la droite C_1C_2 ne peut être reconstruit. Nous nous trouvons alors dans l'impossibilité de déterminer les coordonnées spatiales de C_3 à partir de ses images dans les caméras 1 et 2.

Ce problème n'est absolument pas insurmontable. Dans ce cas, il nous faut trois bilinéarités et une trilinearité pour être capable de prédire un point générique dans la troisième image à partir de ses images dans les deux autres. En effet, pour tout point représenté par \mathbf{m}_1 , \mathbf{m}_2 et \mathbf{m}_3 , les lignes épipolaires générées par \mathbf{m}_1 et \mathbf{m}_2 dans la troisième image vont être confondues. Il n'est alors pas possible de déterminer la position de \mathbf{m}_3 sur cette ligne. La connaissance d'une trilinearité suffit.

On peut aussi résoudre ce problème si nous avons accès à quelques triplets de points réputés correspondre au même point de l'espace. L'existence de tels triplets de points est vérifié dans la quasi totalité des cas pratiques. Soit \mathbf{M} un tel point de l'espace. A partir de \mathbf{P}_1 et \mathbf{P}_2 , nous pouvons reconstruire ce point \mathbf{M} dans l'espace, comme décrit à l'appendice B.3. Or, ce point vérifie l'équation de projection vers la caméra 3, donc $\mathbf{m}_3 = \mathbf{P}_3\mathbf{M}$. Nous avons donc pour \mathbf{m}_3 les équations suivantes :

$$\begin{aligned} \mathbf{m}_3 &= \mathbf{P}_3\mathbf{M} \\ \mathbf{m}_3^T \mathbf{F}_{13} \mathbf{m}_1 &= 0 \\ \mathbf{m}_3^T \mathbf{F}_{23} \mathbf{m}_2 &= 0 \end{aligned}$$

Les deux dernières équations sont redondantes puisque l'on a $\mathbf{F}_{13}\mathbf{m}_1 = \mathbf{F}_{23}\mathbf{m}_2$. La première équation vectorielle n'apporte pas deux équations sca-

lares comme on pourrait s'y attendre, mais une seule. En effet, si l'on change le système de coordonnées tels que l'axe \mathbf{x} soit parallèle à la droite épipolaire et \mathbf{y} perpendiculaire, l'équation $\mathbf{m}_3 = \mathbf{P}_3\mathbf{M}$ se décompose en $m_3^y = 0$ et $m_3^x = \mathbf{AM}$. $m_3^y = 0$ signifie simplement l'appartenance de \mathbf{m}_3 à la droite épipolaire. Donc, l'information apportée par $\mathbf{m}_3 = \mathbf{P}_3\mathbf{M}$ n'est que la position de \mathbf{m}_3 sur la droite épipolaire de \mathbf{m}_1 et \mathbf{m}_2 . Il faut donc trois points \mathbf{M} pour déterminer complètement \mathbf{P}_3 .

2.5.2 De T vers P

Deux solutions se présentent à nous : soit nous supposons que nous avons une représentation de T qui est minimale, c'est à dire qu'elle ne dépend que de 18 paramètres, ou formulé différemment, que les paramètres dont elle dépend vérifient les contraintes qui doivent exister entre eux, soit nous supposons que notre représentation dépend de plus de paramètres et ne vérifie aucune contrainte spécifique à la géométrie trinoculaire. Tous les tenseurs $3 \times 3 \times 3$ ne sont pas possibles, mais seulement un sous espace de dimension 18. La détermination de ce sous espace est décrite à la section 2.4.4.

Le premier cas a l'avantage d'être simple et bien maîtrisé. Le deuxième cas est malheureusement le plus courant si nous n'avons pas fait une minimisation de paramètres de T sous contraintes. Toutefois, nous présenterons pour ces deux approches une seule et même méthode, qui s'apparente à une projection de l'espace des tenseurs T généraux vers la sous-variété des tenseurs T que l'on peut obtenir à partir des matrices de projection. Si les contraintes sont déjà vérifiées, la transformation consiste simplement en un changement de repère, si elles ne le sont pas, la transformation est une projection sur l'espace des matrices de projection suivie d'un changement de repère à l'intérieur de cette variété. En l'absence de métrique sur l'espace des tenseurs définis à un facteur d'échelle près, il est impossible de dire si il s'agit de la meilleure projection possible ou non.

Les matrices de projection représentent donc une paramétrisation possible de l'espace des tenseurs trifocaux. Nous reprenons ici les notations de la section 2.3.2. Notons tout d'abord que si $\mathbf{P}_1 = [\mathbf{I}|\mathbf{0}]$, alors $\mathbf{e}_{21} = [a_{i4}]$ et $\mathbf{e}_{31} = [b_{i4}]$. En effet, on utilise la formule 2.40,

$$\mathbf{e}_{ij} = \mathbf{t}_i - \mathbf{Q}_i \mathbf{Q}_j^{-1} \mathbf{t}_j \quad (2.52)$$

or, $\mathbf{t}_1 = \mathbf{0}$ donc

$$\mathbf{e}_{i1} = \mathbf{t}_i \quad (2.53)$$

Nous savons (2.16) que $T_{ijk} = a_{ji}b_{k4} - a_{j4}b_{ki}$. Si nous arrivons à calculer les épipoles \mathbf{e}_{21} et \mathbf{e}_{31} , c'est à dire $\mathbf{a}_{.4}$ et $\mathbf{b}_{.4}$, le problème du calcul de \mathbf{P}_2 et \mathbf{P}_3 en fonction de T sera un problème linéaire.

On note \mathbf{a}_i la i ème colonne de la matrice \mathbf{P}_2 et \mathbf{b}_i la i ème colonne de la matrice \mathbf{P}_3 . On a alors

$$\mathbf{T}_i^T = \mathbf{a}_i \mathbf{b}_4^T - \mathbf{a}_4^T \mathbf{b}_i$$

On remarque que $\mathbf{r}_i = \mathbf{a}_i \times \mathbf{a}_4$ appartient au noyau de \mathbf{T}_i et que $\mathbf{s}_i = \mathbf{b}_i \times \mathbf{b}_4$ appartient au noyau de \mathbf{T}_i^T . On peut donc calculer les vecteurs \mathbf{r}_i et \mathbf{s}_i en fonction de T . On peut bien sûr utiliser la méthode de son choix pour résoudre le système $\mathbf{T}_i^T \mathbf{s}_i = 0$. Si le tenseur T est issu de matrices de projections, le système aura une solution. Dans le cas contraire, on peut par exemple chercher le vecteur correspondant à la plus petite valeur singulière de \mathbf{T}_i .

Par construction,

$$\forall i \in \{1, 2, 3\}, \mathbf{r}_i^T \mathbf{a}_4 = 0$$

Nous connaissons les vecteurs \mathbf{r}_i , nous pouvons en déduire le vecteur \mathbf{a}_4 qui n'est rien d'autre que \mathbf{e}_{21} . Encore une fois, si le tenseur est issu de matrices de projections, le système aura une solution exacte et sinon nous devons nous contenter d'une solution approchée.

Les équations 2.16 deviennent alors linéaire en les inconnues \mathbf{a}_i et \mathbf{b}_i pour $i \in \{1, 2, 3\}$. Ce système de 18 inconnues et 27 équations peut être alors résolu. Bien sûr, la solution sera exacte si le tenseur vérifie les contraintes internes, et approchée sinon.

Il est important de noter que si aucune précaution n'est prise, les éléments de T auront des ordres de grandeurs très différents. Il est donc souhaitable de normaliser les points avant de calculer les trilinearités. La normalisation s'effectue en choisissant un repère image tel que les coordonnées x et y des points soient comprises entre -1 et 1 . Il est aussi souhaitable de normaliser les équations avant d'effectuer une résolution aux moindres carrés en normalisant par exemple les vecteurs \mathbf{r}_i à 1 avant de résoudre $\mathbf{r}_i \cdot \mathbf{a}_4 = 0$. Cela donne un poids identique à tous les vecteurs \mathbf{r}_i .

2.5.3 De 3 à N caméras

Nous avons exposé au cours de ce chapitre de nombreuses méthodes pour calculer la géométrie d'un système de 3 caméras. Comment généralise t'on

9. On remarque au passage que la matrice formée des vecteurs \mathbf{r}_i n'est rien d'autre que \mathbf{F}_{12} et que la matrice formée des vecteurs \mathbf{s}_i n'est que \mathbf{F}_{13} . Ceci nous donne une méthode efficace pour passer de T à une représentation partielle des bilinéarités.

ces calculs à un nombre quelconque de caméras ?

Nous avons vu à la section 2.4 qu'il n'existe pas de relations linéaires entre des points images d'un même point tridimensionnel faisant intervenir plus de quatre images. De plus, les quadrilinéarités sont des combinaisons linéaires de bilinéarités et de trilinéarités. Les relations linéaires entre points sont donc limitées à trois caméras.

Ces relations entre deux ou trois caméras peut se transformer en matrices de projection. Toutefois, si nous menons nos calculs de manière indépendante pour chaque triplet, les matrices de projection que nous allons obtenir seront exprimées dans des bases projective de l'espace différentes. Les objets reconstruits dans \mathcal{P}^3 le seront dans cette base. Or, si nous connaissons l'expression de plusieurs objets¹⁰ dans des bases différentes, il nous est possible de calculer la matrice de changement de base. Plus spécifiquement, si \mathbf{P} et \mathbf{P}' représentent la même matrice de projection dans deux bases différentes, la matrice de changement de base \mathbf{H} qui est une matrice 4×4 définie à une facteur multiplicatif près, vérifie :

$$\mathbf{P}' = \mathbf{P}\mathbf{H}^{-1}.$$

Si \mathbf{M} et \mathbf{M}' sont des points de \mathcal{P}^3 , la matrice de changement de base devra satisfaire :

$$\mathbf{M}' = \mathbf{H}\mathbf{M}$$

Il est donc possible, en estimant par exemple les matrices de projection par sous-ensembles de trois, de calculer les matrices de changement de base entre les différents sous-ensembles, et donc de construire une représentation des N caméras dans une même base de l'espace.

2.6 Conclusion

Dans ce chapitre, nous avons étudié les différents modèles de caméras. Notre choix s'est porté sur le modèle perspectif, et ce pour plusieurs raisons. Contrairement au modèles orthographique et paraperspectif, il décrit bien les déformations dues à un changement de profondeur dans la scène. Il est associé à un formalisme mathématique agréable, la géométrie projective. nous n'avons pas choisi un modèle plus compliqué, comme ceux qui intègrent les distorsions des images car nous pouvons rectifier indépendamment chacune des images et nous ramener à un modèle perspectif.

Partant de ce modèle de la géométrie d'une caméra, nous avons entrepris de construire une théorie pour une paire de vues. L'objet central est alors une

10. un objet pouvant être ici une matrice de projection, un point ou une droite de l'espace

matrice 3×3 de déterminant nul et définie à un facteur d'échelle près appelée matrice fondamentale. Cette matrice concentre l'information épipolaire, qui a un point fait correspondre une droite dans l'autre image. Cette droite est l'image du rayon issu de ce point.

Nous nous sommes ensuite penchés sur le cas un peu plus complexe de trois caméras, et nous avons vu premièrement que les matrices fondamentales étaient inadaptées et que la géométrie pouvait être capturée, entre autres par un tenseur $3 \times 3 \times 3$ défini à un facteur d'échelle près et vérifiant certaines contraintes internes. Ce tenseur caractérise les relations linéaires existant entre les coordonnées 2-D des points.

Toutefois, les dérivations des deux précédentes sections n'ayant rien de systématique, nous sommes revenus aux équations initiales de la projection pour établir une théorie des relations linéaires entre les points des images. Cette théorie est basée sur les déterminants. Elle nous fournit non seulement une méthode de calcul de toutes les relations linéaires dans les images, mais aussi une explication pour les contraintes existantes entre les différents objets manipulés.

Finalement, nous avons présenté des méthodes à la fois théoriques et pratiques pour passer d'une représentation possible (matrice fondamentale, trilinearités, matrices de projection), vers une autre. Il doit maintenant être clair que le passage des matrices de projection vers les trilinearités ou les bilinearités est une opération facile. Les opérations inverses, des trilinearités ou des bilinearités vers les matrices de projection sont des opérations difficiles, qui nécessitent certaines précautions. Nous n'avons pas étudié les méthodes pour passer directement des trilinearités aux bilinearités et vice versa puisque nous pouvons les relier en passant par un jeu de matrices de projection.

Chapitre 3

Estimation de la géométrie des caméras

3.1 Comment obtenir les appariements?

Nous allons maintenant nous intéresser à l'estimation de la géométrie du système de caméras. Nos données vont être des correspondances de primitives entre les images. Ces correspondances doivent être obtenues uniquement à partir des images. Aucune supposition n'est faite sur la structure de la scène ni sur la géométrie des caméras, ce sont des inconnues toutes les deux.

Le problème de la mise en correspondance, comme celui de la segmentation, est l'un des problèmes les plus difficiles, et bien sûr loin d'être résolu en vision par ordinateur et en vision pour la robotique. Ici, nos données vont être des primitives dans des images différentes et leurs attributs et nous devons identifier si elles proviennent de la même primitive 3-D. On appelle aussi ce problème *appariement*.

La recherche dans ce domaine est toujours très active (On peut consulter, entre autres, [Brown, 1992, Dhond et Aggarwal, 1989, Goshtasby et al., 1984, Hummel et Zucker, 1983, Kahl et al., 1980, Zhang et al., 1994, Maître et Wu, 1987, Robert et Faugeras, 1991, Zhang, 1993, Zhang et Faugeras, 1992], mais aussi bien d'autres ...), même si une énorme quantité de travail a été accomplie au cours de ces 15 dernières années. Hélas, les résultats dépendent toujours d'heuristiques plus ou moins bien vérifiées, comme la similitude des profils d'intensité. En effet, ce qui apparaît simple à l'oeil humain est assez difficile à réaliser avec des machines. Il faut à la fois détecter des primitives,

et les appairer alors quelles sont généralement différentes. On regarde plusieurs images pour justement en apprécier les différences. On cherche donc alors une relative similitude vérifiant quelques contraintes pour restreindre les possibilités. Il n'existe pas à l'heure actuelle de système mettant en correspondance plusieurs images qui soit robuste à des changements d'aspect ou d'orientation des objets. Certains systèmes s'en approchent, mais personne ne prétend avoir trouvé une solution satisfaisante à ce problème difficile.

Ces difficultés sont parfois contournées en prenant des séquences finement discrétisées dans le temps. En effet, puisque l'intervalle de temps entre les images est petit et que la vitesse des objets est contrainte par des lois physiques à être bornée, on peut obtenir des déplacements très petits d'une image à l'autre. L'espace dans lequel on va devoir chercher le correspondant est restreint à un voisinage de la primitive d'intérêt en question.

3.1.1 Les primitives d'intérêt utilisées : points et droites

3.1.1.1 Les points

Les points saillants de la surface d'intensité de l'image sont des points appropriés pour la mise en correspondance. En effet, ils sont facilement identifiables d'une image à l'autre puisqu'ils représentent des "événements" de la surface d'intensité, le plus souvent des discontinuités. Ces points représentent souvent (mais pas exclusivement) des coins des objets dans l'image. Leur mise en correspondance sera basée sur la similitude de leur profil d'intensité.

Les coins dans une image de fonction d'intensité $I(x, y)$ sont obtenus comme maxima d'un opérateur $\mathcal{O}(x, y)$ appliqué à l'image : ce sont les points (x_0, y_0) de l'image tels que

$$\forall (x, y) \in V, \mathcal{O}(x_0, y_0) > \mathcal{O}(x, y)$$

où V est un voisinage de (x_0, y_0) , généralement, un rectangle centré en (x_0, y_0) ; seuls les points dont la valeur $\mathcal{O}(x_0, y_0)$ est supérieure à un seuil sont réellement considérés comme des coins.

L'opérateur que nous avons utilisé est celui décrit dans [Harris et Stephens, 1988]. Il est une version légèrement modifiée de l'opérateur de Plessey [Harris, 1987, Noble, 1988] :

$$\mathcal{O}(x, y) = \det(\hat{\mathbf{C}}(x, y)) - k(\text{trace}(\hat{\mathbf{C}}(x, y)))^2$$

avec

$$\widehat{\mathbf{C}}(x, y) = \begin{pmatrix} \widehat{\frac{\partial I(x, y)^2}{\partial x}} & \widehat{\frac{\partial I(x, y)}{\partial x} \frac{\partial I(x, y)}{\partial y}} \\ \widehat{\frac{\partial I(x, y)}{\partial x} \frac{\partial I(x, y)}{\partial y}} & \widehat{\frac{\partial I(x, y)^2}{\partial y}} \end{pmatrix}$$

Dans la formule ci-dessus, $\widehat{F}(x, y)$ est la fonction obtenue à partir de $F(x, y)$ après lissage, c'est-à-dire, convolution par un noyau de convolution $G_\sigma(x, y)$:

$$\widehat{F}(x, y) = \int_{u=-\infty}^{u=\infty} \int_{v=-\infty}^{v=\infty} G_\sigma(x - u, y - v) F(u, v) du dv$$

Les noyaux de convolution sont choisis séparables en x et y , c'est-à-dire que $G_\sigma(x, y) = g_\sigma(x)g_\sigma(y)$. Le plus souvent utilisé est le noyau gaussien

$$g_\sigma(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}$$

où σ correspond au rayon approximatif du noyau.

Cette façon de procéder produit les coordonnées des coins à un pixel près, ce qui est suffisant pour la procédure de corrélation utilisée qui ne prend en compte que des coordonnées entières. Une précision subpixelique est obtenue a posteriori en raffinant chacun des coins extraits à partir de l'opérateur précédent, à l'aide d'une méthode basée sur un modèle paramétrique local du coin [Blaszka et Deriche, 1994, Blaszka et Deriche, 1995, Rohr, 1992b]. On peut aussi utiliser des méthodes de mise en correspondance ayant une précision inférieure au pixel comme [Devernay et Faugeras, 1994a, Devernay et Faugeras, 1994b].

Il existe bien sûr d'autres extracteurs de coins avec des caractéristiques plus ou moins différentes. Ils sont recensés dans [Deriche et Giraudon, 1993, Rohr, 1992a] par exemple.

3.1.1.2 Les droites

Les droites sont au moins aussi intéressantes que les points, parce que certaines de leurs caractéristiques peuvent être détectées avec plus de stabilité que les points. En effet, si les extrémités d'un segment peuvent varier grandement d'une image à l'autre, l'orientation ainsi que la distance à l'origine sont extrêmement stables. Ces qualités sont particulièrement intéressantes puisque la précision des primitives d'intérêt détermine la précision de nos estimations. On utilise alors comme paramètres des lignes l'orientation θ , la longueur l , la distance à l'origine a et la coordonnée du point milieu sur le

segment b . θ et a ont une variance faible par rapport à celle de l et de b . On note aussi que plus un segment est long, plus son orientation est précise.

Les droites sont extraites à partir d'un détecteur de contours comme ceux décrits dans [Canny, 1986, Deriche, 1987, Spacek, 1985, Devernay, 1993]. Ces approches fonctionnent toutes en combinant des lissages avec une dérivation pour déterminer les plus grandes variations de l'intensité dans l'image. Les contours sont extraits à partir de l'image d'intensité, puis on les approxime par des segments qui vont constituer nos droites. On prend bien sûr soin de retirer les segments trop courts qui sont moins significatifs. On peut aussi utiliser des approches à base de modèles paramétriques pour obtenir une estimation plus précise [Blaszka et Deriche, 1994, Blaszka et Deriche, 1995].

3.1.2 Mise en correspondance

La mise en correspondance de primitives d'intérêt entre plusieurs images se base tout d'abord sur la similitude des profils d'intensité avoisinants. Cette similitude va être mesurée par la corrélation entre les deux vecteurs associés à ces profils d'intensité. Cette corrélation est une grandeur statistique représentant la Cette première étape va permettre d'associer un point d'une image à un ou plusieurs points dans une autre image. Une deuxième étape permet généralement de lever cette ambiguïté en supposant que deux points près l'un de l'autre dans une image doivent avoir des correspondants peu éloignés dans une autre image. Il s'agit alors d'une contrainte sur l'ensemble de la scène et non plus sur des primitives séparées. Dans le cas du suivi, cette contrainte est prise en compte au moment de la mise en correspondance, puisque l'on va chercher le correspondant dans un voisinage proche de l'ancienne position de la primitive, et même dans certains cas, dans une direction seulement si l'on connaît la loi d'évolution de la position de ce point. Les algorithmes de suivi sont souvent basés sur des méthodes issues du filtrage de Kalman, comme dans [Deriche et Faugeras, 1990, Sethi et Jain, 1987, Zhang, 1994, Bascle et al., 1994, Crowley et Stelmaszyk, 1990, Gamba, 1989, Gai-Checa et al., 1993]. Nous développerons ici uniquement et brièvement la mise en correspondance à base de corrélation et la relaxation pour lever les ambiguïtés.

3.1.2.1 Corrélation

Lorsqu'aucune information n'est connue sur le modèle, ni sur le système, ni sur la scène observée, la seule information disponible pour déterminer si deux points, appartenant à deux images différentes I_1 et I_2 , se correspondent est la fonction d'intensité définissant chacune des images.

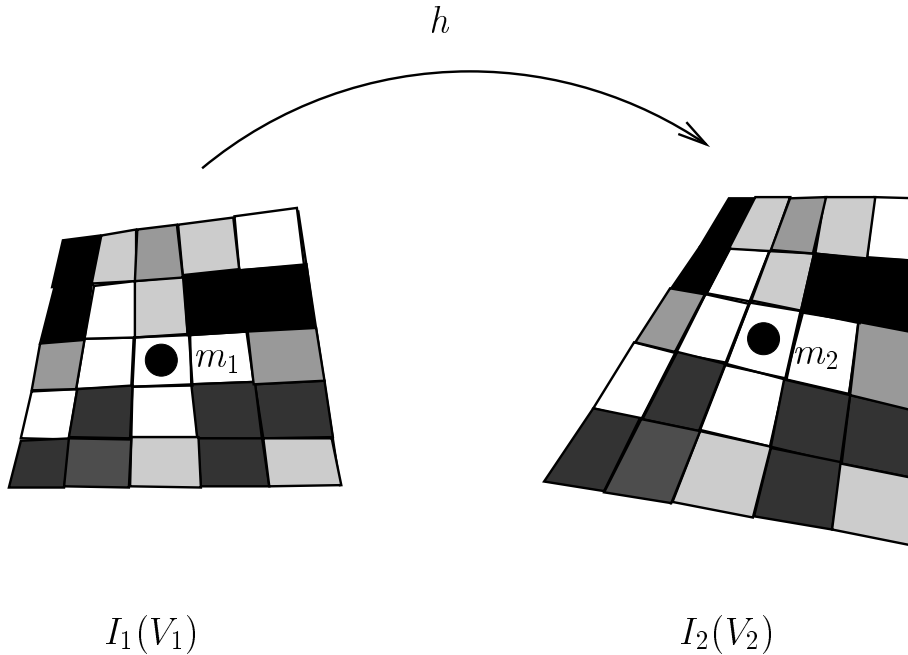


FIG. 3.1 – Si les deux points m_1 de l'image I_1 et m_2 de l'image I_2 sont en correspondance, la valeur de la fonction d'intensité de I_1 dans un voisinage V_1 de m_1 est corrélée à celle de la fonction d'intensité de I_2 dans un voisinage V_2 de m_2 .

Considérons une correspondance de points (m_1, m_2) entre I_1 et I_2 . Soit M le point de la scène qui se projette en m_1 et m_2 . En considérant que M appartient à une surface lambertienne, si I_1 et I_2 sont issues d'une même caméra, l'intensité de m_1 est égale à celle de m_2 . Si I_1 et I_2 sont issues de deux caméras différentes, la fonction reliant l'intensité de m_1 à l'intensité de m_2 est indépendante de M et monotone. En conséquence, les vecteurs \mathbf{i}_1 et \mathbf{i}_2 représentant, respectivement, les valeurs de la fonction d'intensité de I_1 dans un voisinage V_1 de m_1 et les valeurs de la fonction d'intensité de I_2 dans un voisinage V_2 de m_2 , sont fortement corrélés puisque ces voisinages sont, en général, eux-mêmes en correspondance (voir la figure 3.1). Dans le cas où V_1 et V_2 correspondent à une portion de plan de la scène, par exemple, nous pouvons déduire l'un des vecteurs \mathbf{i}_1 et \mathbf{i}_2 de l'autre, en nous appuyant sur le fait que, d'après l'appendice B, V_1 et V_2 sont alors en correspondance homographique.

Lorsque seules les fonctions d'intensité sont connues, la relation générale entre \mathbf{i}_1 et \mathbf{i}_2 est impossible à déterminer et nous ne pouvons qu'exhiber un critère de corrélation qui mesure le degré de corrélation entre \mathbf{i}_1 et \mathbf{i}_2 . Les critères les plus communément utilisés découlent des approximations suivantes

qui consistent à supposer que :

- la fonction reliant l'intensité i_2 de \mathbf{m}_2 à l'intensité i_1 de \mathbf{m}_1 est affine, c'est-à-dire, de la forme $i_2 = ai_1 + b$,
- V_1 et V_2 correspondent à une portion de plan, et donc, la surface autour de \mathbf{M} est localement assimilée à un plan,
- l'homographie h associée à ce plan est la translation de vecteur $\mathbf{m}_1\mathbf{m}_2$, ce qui revient à considérer que les images sont localement *rectifiées par rapport à ce plan*; cette dernière approximation conduit à prendre exactement la même forme pour V_1 et V_2 , généralement, un rectangle, centré en m_1 pour V_1 et en m_2 pour V_2 et appelé *fenêtre de corrélation*.

Les quatre critères classiques suivants s'en déduisent alors selon les hypothèses supplémentaires faites sur a et b :

- $|\mathbf{i}_2 - \mathbf{i}_1|$, qui suppose que $a = 1$ et $b = 0$, ce qui est une bonne approximation si I_1 et I_2 viennent d'une même caméra;
- $|(\mathbf{i}_2 - \bar{\mathbf{i}}_2) - (\mathbf{i}_1 - \bar{\mathbf{i}}_1)|$, où \bar{i}_j est l'intensité moyenne de I_j sur V_j ; ce critère suppose seulement que $a = 1$;
- $\cos(\mathbf{i}_2, \mathbf{i}_1)$, qui suppose seulement que $b = 0$;
- $\cos((\mathbf{i}_2 - \bar{\mathbf{i}}_2), (\mathbf{i}_1 - \bar{\mathbf{i}}_1))$, qui ne fait aucune hypothèse sur les valeurs de a et b .

Moins il y a d'hypothèses faites sur a et b , plus le critère correspondant est général, mais aussi, plus son calcul nécessite un nombre élevé d'opérations. Nous avons utilisé le dernier des critères exposés ci-dessus.

Des critères plus généraux que les précédents sont obtenus en relâchant la contrainte sur h , c'est-à-dire, en autorisant h à être n'importe quelle homographie, ou au moins, n'importe quelle application linéaire affine. Dans ce cas, nous avons $V_2 = h(V_1)$ et \mathbf{i}_2 se calcule à partir de \mathbf{i}_1 par interpolation. Le calcul du critère de corrélation consiste alors à trouver l'homographie h_{max} qui conduit au meilleur score de corrélation calculé à partir d'un des quatre critères définis précédemment. La procédure est donc beaucoup plus complexe puisqu'elle comprend une phase de minimisation par un processus itératif. En contre partie, la connaissance de h_{max} constitue, en général, une information supplémentaire précieuse, en relation avec la normale à la surface en M [Devernay et Faugeras, 1994b]. Une étude comparative de différents critères de corrélation se trouve dans [Aschwanden et Guggenbühl, 1992].

3.1.2.2 Relaxation

Le but de l'étape de relaxation est d'extraire de l'ensemble incohérent de correspondances de point potentielles produit par l'étape de corrélation décrite dans la section 3.1.2.1, l'ensemble des bonnes correspondances de point. Cet ensemble doit être :

- cohérent, dans le sens où un coin ne doit être impliqué que dans une seule correspondance de points,
- grand, car nous supposons que les bonnes correspondances de points sont présentes dans l'ensemble produit par l'étape de corrélation décrite à la section 3.1.2.1. Les points en correspondance ont une forte corrélation entre eux.
- cohérent avec les hypothèses qui justifient l'utilisation du critère de corrélation, c'est-à-dire que les coins en correspondance doivent être localement approximativement en correspondance homographique, ou au moins, affine ou euclidienne.

Un *critère de relaxation* est défini qui tient compte de ces caractéristiques, le but de l'étape de relaxation devenant alors la maximisation de ce critère.

Le critère de relaxation. Le critère de relaxation est défini pour un ensemble \mathcal{S} de correspondances de point. A chaque correspondance de points (m_1, m_2) de \mathcal{S} , est attribué un *score de relaxation* $s(m_1, m_2)$ qui est toujours positif et mesure le degré de cohérence de (m_1, m_2) avec les caractéristiques, décrites ci-dessus, voulues pour les correspondances finales. Le critère de relaxation est alors défini comme la somme des scores de relaxation de toutes les correspondances de point de \mathcal{S} :

$$\mathcal{C}(\mathcal{S}) = \sum_{(m_1, m_2) \in \mathcal{S}} s(m_1, m_2)$$

Ainsi, maximiser ce critère revient à maximiser les scores de relaxation et donc choisir l'ensemble des bonnes correspondances de point.

Le score de relaxation. Le score de relaxation d'une correspondance de points (m_1, m_2) est lui-même défini comme une somme de scores toujours positifs :

$$s(m_1, m_2) = \sum_{(n_1, n_2) \in \mathcal{V}(m_1, m_2)} \delta(m_1 \vec{n}_1, m_2 \vec{n}_2) \quad (3.1)$$

où $\mathcal{V}(m_1, m_2)$ est le voisinage de (m_1, m_2) et $\delta(\vec{u}, \vec{v})$ est le *score de similarité* entre \vec{u} et \vec{v} .

$\mathcal{V}(m_1, m_2)$ est défini comme suit :

$$\mathcal{V}(m_1, m_2) = \{(n_1, n_2) \in \mathcal{S}, \|m_1 \vec{n}_1\| < r \text{ et } \|m_2 \vec{n}_2\| < r\}$$

où r est un seuil fixant le rayon du voisinage. Ainsi, plus il y a de correspondances de point dans le voisinage de (m_1, m_2) , plus $s(m_1, m_2)$ est élevé. Cette définition a pour but de favoriser les ensembles à nombre élevé de correspondances de point.

$\delta(\vec{u}, \vec{v})$ est défini comme suit :

$$\delta(\vec{u}, \vec{v}) = \begin{cases} \frac{1}{1 + \frac{\|\vec{u}\| + \|\vec{v}\|}{2}} & \text{si } \frac{1}{k_0} < \frac{\|\vec{v}\|}{\|\vec{u}\|} < k_0 \text{ et } |(\vec{u}, \vec{v})| < \theta_0, \\ 0 & \text{sinon.} \end{cases}$$

où $k_0 > 1$ et $\theta_0 > 0$. Cette définition montre que \vec{u} et \vec{v} sont considérés comme similaires si \vec{v} est l'image de \vec{u} par une similitude d'angle compris entre $-\theta_0$ et θ_0 et de rapport compris entre $\frac{1}{k_0}$ et k_0 . Elle favorise donc les correspondances de point (m_1, m_2) telles que le voisinage de m_2 se déduit approximativement de celui de m_1 par une application affine proche de la translation de vecteur $m_1 \vec{m}_2$, ce qui est cohérent avec le critère de corrélation utilisé. L'approximation est d'autant plus valable que les voisinages considérés sont restreints : c'est pourquoi, d'une part, $s(m_1, m_2)$ ne prend en compte que le voisinage de (m_1, m_2) et d'autre part, $\delta(m_1 \vec{n}_1, m_2 \vec{n}_2)$ est d'autant plus grand que n_1 est proche de m_1 et n_2 est proche de m_2 . De la même manière que pour le critère de corrélation, la définition du score de similarité peut être rendue plus générale en faisant une approximation plutôt par une homographie que par une similitude. Dans ce cas, plus de deux correspondances de point potentielles doivent être considérées.

Enfin, dans la somme présente dans la définition de $s(m_1, m_2)$ donnée par l'équation (3.1), les voisins n_1 de m_1 ou n_2 de m_2 ne sont comptés qu'une seule fois : si deux scores de similarité $\delta(m_1 \vec{n}_1, m_2 \vec{n}_2)$ et $\delta(m_1 \vec{n}'_1, m_2 \vec{n}'_2)$ ont un point en commun, c'est-à-dire que $n_1 = n'_1$ ou $n_2 = n'_2$, alors, seul le plus grand des deux est pris en compte. Ceci empêche $s(m_1, m_2)$ d'être grossi, de façon indésirable, du fait de la présence de nombreuses correspondances de point potentielles contradictoires au voisinage de (m_1, m_2) .

La maximisation. La maximisation est faite de manière itérative. L'ensemble \mathcal{S} des correspondances de point est initialement celui produit par l'étape de corrélation décrite dans la section 3.1.2.1. A chaque itération, dans

un premier temps, tous les scores de relaxation sont calculés; ils permettent, dans un second temps, d'extraire de \mathcal{S} de bonnes correspondances de point, de telle sorte que \mathcal{S} évolue vers un ensemble cohérent qui maximise $\mathcal{C}(\mathcal{S})$.

Pour cela, l'extraction de \mathcal{S} de bonnes correspondances de points est faite de la façon suivante : une correspondance de points (m_1, m_2) potentielle est considérée comme bonne s'il n'existe pas d'autre correspondance de points de \mathcal{S} impliquant m_1 ou m_2 qui ait un score de relaxation supérieur à $s(m_1, m_2)$; pour toute bonne correspondance de points (m_1, m_2) , toutes les correspondances de point potentielles impliquant m_1 ou m_2 sont ôtées de \mathcal{S} .

Le processus itératif est stoppé dès lors qu'aucune nouvelle bonne correspondance de points n'a été trouvée. Il est nécessairement stoppé à un moment donné, puisqu'à chaque itération, au moins une correspondance de points potentielle est ôtée de \mathcal{S} et le nombre total de correspondances de point potentielles est limité.

Une fois le processus itératif stoppé, toutes les correspondances de points potentielles de \mathcal{S} qui ont un coin en commun sont ôtées de \mathcal{S} de façon à obtenir un ensemble cohérent.

Une description plus complète de cet algorithme se trouve dans [Zhang et al., 1994]. D'autres relaxations sont utilisées dans [Rosenfeld et al., 1976, Zucker et al., 81, Pollard et al., 1985, Li, 1990, Faugeras et Berthod, 1980, Faugeras et Berthod, 1981].

3.2 Estimation de \mathbf{F}

Les matrices fondamentales sont les objets les plus simples que nous avons à manipuler. Il va nous être souvent utile de les estimer pour pouvoir initialiser d'autres estimations.

3.2.1 Estimation des \mathbf{F}_{ij} paire par paire

Avant de penser obtenir une représentation globale de la géométrie projective du système de caméras, il est souvent utile d'obtenir une représentation paire par paire, de manière indépendante. Soient i et j une telle paire de caméras. Ces matrices fondamentales ne respecteront bien sûr pas les contraintes qui existent entre elles, comme celles que nous avons vues à la section 2.3.1. Elles seront néanmoins utiles comme point de départ pour de nombreux algorithmes. La méthode se décompose habituellement en deux parties:

1. une estimation linéaire (aux moindres carrés) de \mathbf{F}_{ij} .

2. un raffinement itératif (une minimisation avec un critère non linéaire) de \mathbf{F}_{ij} .

Les équations nécessaires à la première partie dérivent directement de l'équation épipolaire fondamentale $\mathbf{m}_j^T \mathbf{F}_{ij} \mathbf{m}_i = 0$. On obtient ainsi une équation par appariement $(\mathbf{m}_i, \mathbf{m}_j)$. Cette équation fait intervenir les 9 coefficients de \mathbf{F}_{ij} de manière homogène de degré 1. Pour N appariements, on obtient un système de N équations linéaires homogènes de degré 1 à 9 inconnues. 8 appariements sont donc nécessaires pour obtenir un résultat. Pour un plus grand nombre d'appariements, on utilise une méthode de moindres carrés linéaires pour résoudre le système surdéterminé. Il existe des méthodes pour calculer une matrice fondamentale avec moins d'appariements (7 est bien sûr le minimum, la matrice fondamentale ayant 7 paramètres indépendants), mais nous ne les utiliserons pas ici. Le lecteur intéressé peut se reporter à [Torr et al., 1995] ou à des articles plus anciens [Hesse, 1863, Sturm, 1869].

La méthode linéaire a cependant plusieurs inconvénients, comme mentionné dans la thèse de Quang-Tuan Luong [Luong, 1992]. Tout d'abord, elle ne prend pas en compte les contraintes internes à la matrice fondamentale. Les équations $\mathbf{m}_j^T \mathbf{F}_{ij} \mathbf{m}_i = 0$ n'imposent pas que le déterminant de \mathbf{F}_{ij} soit nul. L'expérience prouve d'ailleurs que les matrices \mathbf{F}_{ij} obtenues par cette méthode ne sont pas de rang deux en général. Ensuite, le résultat minimise $\sum |\mathbf{m}_j^T \mathbf{F}_{ij} \mathbf{m}_i|^2$ qui ne représente pas une distance avec une interprétation physique. Les différents termes de l'équation peuvent avoir des ordres de grandeurs différents et donner ainsi des résultats différents à partir des mêmes données. Le critère n'est pas invariant à un changement de représentant projectif pour les points \mathbf{m}_i ou \mathbf{m}_j . On substitue alors à ce critère un autre, plus "physique" mais non quadratique. Il s'agit de la distance d'un point à la droite engendrée par son correspondant dans l'autre image.

Le résultat du calcul linéaire est toutefois utile parce qu'il donne une initialisation le plus souvent correcte et obtenue à un coût de calcul moindre. L'étape suivante est la minimisation d'un critère non linéaire.

Les critères que l'on cherche maintenant à minimiser sont des critères avec une signification géométrique dans les images. L'un d'entre eux, le plus utilisé, est :

$$C(\mathbf{F}_{ij}) = \sum_{\mathbf{m}_i, \mathbf{m}_j} \left(\frac{\mathbf{m}_j^T \mathbf{F}_{ij} \mathbf{m}_i}{\sqrt{(\mathbf{m}_j^T \mathbf{F}_{ij} \mathbf{m}_i)_1^2 + (\mathbf{m}_j^T \mathbf{F}_{ij} \mathbf{m}_i)_2^2}} \right)^2 + \left(\frac{\mathbf{m}_i^T \mathbf{F}_{ij}^T \mathbf{m}_j}{\sqrt{(\mathbf{m}_i^T \mathbf{F}_{ij}^T \mathbf{m}_j)_1^2 + (\mathbf{m}_i^T \mathbf{F}_{ij}^T \mathbf{m}_j)_2^2}} \right)^2 \quad (3.2)$$

Ce critère mesure en fait la distance entre un point et la droite épipolaire du point correspondant dans l'autre image comme montré en figure 3.2. Le critère comprend deux termes afin d'être symétrique par rapport à i et à j .

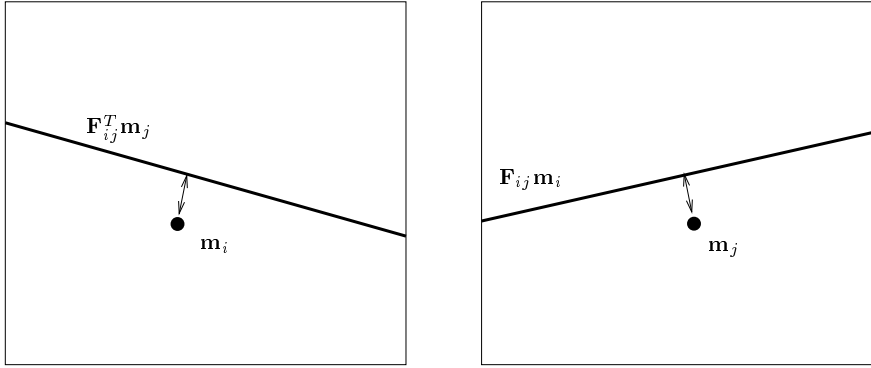


FIG. 3.2 – Distance épipolaire entre \mathbf{m}_i et $\mathbf{F}_{ij}^T \mathbf{m}_j$ et entre \mathbf{m}_j et $\mathbf{F}_{ij} \mathbf{m}_i$.

Une fois que le critère n'est plus linéaire, il n'y a aucune raison de maintenir une paramétrisation de \mathbf{F}_{ij} qui ne tienne pas compte de la contrainte $|\mathbf{F}_{ij}| = 0$. Différentes paramétrisations existent et sont répertoriées dans [Luong, 1992]. Il existe par exemple une paramétrisation par les épipoles \mathbf{e}_{ij} et \mathbf{e}_{ji} et l'homographie qui relie les droites épipolaires correspondantes entre elles.

Le calcul de \mathbf{F}_{ij} apparaît alors comme une minimisation de C par rapport à un vecteur de 7 paramètres.

3.2.2 Estimation de \mathbf{F} sous contrainte

La méthode précédente est valable pour un calcul entre deux images. Comment peut-on généraliser le calcul des matrices fondamentales pour un nombre quelconque d'images ?

La première extension évidente de ce procédé est de calculer les matrices \mathbf{F}_{ij} en utilisant la même paramétrisation, mais en opérant une minimisation sous contraintes. Nous voulons ici calculer toutes les matrices \mathbf{F}_{ij} de manière simultanée, sans passer par les matrices de projection ou les trilinearités. Des méthodes pour calculer l'ensemble des matrices fondamentales à partir de certaines d'entre elles seront présentées à la section 2.5.1.

Cette méthode apparaît vite comme inutilisable en pratique. En effet, il existe $7N(N-1)/2$ paramètres, mais le nombre de contraintes est gigantesque : Il existe 3 contraintes de plan trifocal par triplet, donc $N(N-1)(N-2)/2$ contraintes de ce type. Il existe d'autres contraintes pour 4 caméras, qui fournissent une équation, donc $N(N-1)(N-1)(N-3)/24$. Puisque le nombre de contraintes varie en $O(N^3)$ et $O(N^4)$, il existe un N tel que ces contraintes soient algébriquement dépendantes. En effet, pour $N > 7$, il

y a plus de contraintes que de paramètres. Les relations entre ces contraintes sont complexes et font intervenir les coefficients des matrices \mathbf{F} à des degrés élevés. A titre d'exemple, pour 7 caméras, le nombre de paramètres du système est de 62, le nombre de paramètres des matrices fondamentales est de 87 et le nombre de contraintes d'ordre 3 et 4 est de 140 ! Cet exemple nous montre que lorsque nous voudrions estimer la géométrie d'un système à plus de trois ou quatre caméras, il nous faut absolument le paramétrer avec des matrices de projection ou avec une autre représentation, mais qui dépende linéairement du nombre de caméras.

Il faut toutefois modérer ce discours en signalant qu'il est possible de choisir un sous-ensemble de matrices fondamentales à partir desquelles elles peuvent être toutes calculées, au besoin en passant par des matrices de projection. Le nombre de contraintes est alors beaucoup moins grand puisque le nombre de matrices fondamentales varie en $O(N)$ et non plus en $O(N^2)$. Le nombre exact de paramètres varie en fonction du sous ensemble choisi. La Figure 3.3 montre un sous ensemble possible à $2N - 1$ matrices fondamentales.

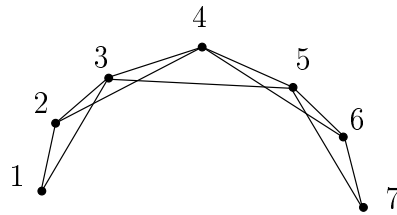


FIG. 3.3 – *Un sous ensemble de matrices fondamentales permettant d'obtenir l'ensemble de la géométrie épipolaire. Chaque trait représente une matrice fondamentale, chaque point une caméra. Cet ensemble contient $2N - 1$ matrices fondamentales.*

Cette minimisation reste cependant lente à cause des calculs à faire à chaque itération : calcul des \mathbf{P} à partir du sous ensemble des \mathbf{F} , puis calcul de l'ensemble des \mathbf{F} puis calcul du critère dans les $O(N^2)$ paires. Ceci est équivalent à ce qui sera discuté plus tard à la section 3.4.1.

3.3 Estimation de \mathbf{T}

Comme les bilinéarités, les trilinearités nous sont utiles comme point de départ de nos estimations. C'est pour cette raison que nous nous attacherons à leur estimation comme moyen pour retrouver la géométrie épipolaire d'un ensemble de caméras.

3.3.1 Méthode linéaire

Rappelons nous l'équation (2.17).

$$\sum_k m_1^k (m_2^i m_3^m T_{kjl} - m_2^j m_3^m T_{kil}) = \sum_k m_1^k (m_2^i m_3^l T_{kjm} - m_2^j m_3^l T_{kim})$$

Si on considère \mathbf{m}_1 , \mathbf{m}_2 et \mathbf{m}_3 connus, ce qui est le cas puisque nous supposons donnés un certain nombre d'appariements, les différents choix pour i, j, l et m donnent quatre équations en les éléments de T . Avec 7 points, nous obtenons 28 équations, ce qui est suffisant pour résoudre notre système de 27 inconnues définies à un facteur d'échelle près. Bien sûr de meilleurs résultats seront obtenus avec un nombre de points très supérieur à 7 en utilisant des méthodes aux moindres carrés. Cette estimation pose cependant certains problèmes:

- Le tenseur T résultant n'a aucune raison particulière de vérifier les contraintes qui existent entre ses coefficients. En effet, ils sont tous considérés de manière indépendante dans l'équation (2.17).
- La quantité minimisée est relativement mal définie puisqu'il s'agit d'un produit de trois coordonnées de points image et d'un élément de T . Cette quantité minimisée n'a aucune signification physique. De plus, nous n'avons aucune garantie que les équations ont le même ordre de grandeur. En effet, les éléments m_i^j intervenant dans les équations (2.17) n'ont pas tous le même ordre de grandeur. En effet, il est courant que les vecteurs \mathbf{m} soient de la forme $(x, y, 1)^T$ où x et y varient dans toute l'image. Les deux premières composantes sont donc le plus souvent en $O(10^2)$ tandis que la troisième reste à 1. Les équations faisant intervenir ces termes d'ordre de grandeurs différents auront elles-mêmes des ordres de grandeurs différents. Si les ordres de grandeurs des équations sont différents, alors leur poids dans le résultat final va aussi être différent.

Une première solution pour rendre la méthode plus robuste est d'introduire des équations trinéaires entre droites comme en 2.3.2. Puisque les droites sont détectées de manière plus stable dans les images, les équations qui en découlent sont moins sensibles au bruit.

Une deuxième précaution utile est de normaliser les droites et les points pour que les différentes coordonnées soient de tailles approximativement identiques. Pour les points, on utilise un changement de repère qui ramène les coordonnées x et y des points entre -1 et 1. Les résultats obtenus avec normalisation sont meilleurs que les résultats sans normalisation de plusieurs ordres de grandeur. La raison en est simple: les coefficients T_{ijk} sont multipliés soit

par des coordonnées (x,y) entre 0 et la taille de l'image en pixels (quelques centaines) soit par 1 (z), et ceci trois fois. Il en résulte que le tenseur T a des entrées 10^8 fois plus grandes que d'autres. Bien sûr les entrées les plus petites ont autant d'importance que les grandes, mais elles sont négligées lors des calculs. Cette normalisation est aussi souhaitable pour le calcul des matrices fondamentales, mais il est moins critique dans ce cas car la différence est de deux ordres de grandeur au lieu de trois.

Une troisième amélioration consiste à prendre en compte les 9 équations, même si certaines sont des combinaisons linéaires des autres. Les résultats ne sont pas différents en cas de bruit faible, mais se stabilisent mieux si le bruit sur les points augmente.

3.3.2 Méthodes non linéaires

Comme pour l'estimation des matrices fondamentales, on peut, au lieu de minimiser les carrés des équations (2.17), minimiser des distances physiques dans les images pour avoir de meilleurs résultats. Le prix à payer est en général que le critère n'est plus quadratique en fonction des variables et qu'il n'existe alors plus de solution non itérative. Puisque nous avons introduit ainsi des non linéarités, on peut aussi transformer une représentation non minimale en une représentation minimale. Cela permet d'obtenir en fin de minimisation un objet qui vérifie les contraintes qui lui sont imposées. C'est aussi ce que nous allons faire ici.

Le critère d'ajustement des trilinearités est un critère simple : Étant donné \mathbf{m}_1 , \mathbf{m}_2 et \mathbf{m}_3 une correspondance, nous calculons le critère comme suit : à partir de \mathbf{m}_1 et \mathbf{m}_2 , les 9 trilinearités génèrent 9 droites. Nous calculons la somme des carrés des distances entre \mathbf{m}_3 et ces 9 droites. Pour rendre symétrique le critère, nous opérons pareillement avec \mathbf{m}_2 et \mathbf{m}_3 ainsi qu'avec \mathbf{m}_1 et \mathbf{m}_3 . Une idée de ce critère est donnée dans la figure 3.4. Bien sûr, si ce triplet vérifie les contraintes épipolaires, les 9 droites se coupent en un point et plusieurs d'entre elles sont confondues. Malheureusement, cette situation est rare puisque les points qui nous sont donnés pour l'estimation de la géométrie peuvent être bruités. Les 9 droites se coupent alors en trois points distincts correspondant aux trois choix possibles pour le plan passant par \mathbf{m}_2 (voir figure 2.7).

Nous pouvons choisir plusieurs types de paramétrisations minimales. Le choix devra surtout être conduit par la simplicité de la paramétrisation et par la facilité à la déduire d'une configuration qui ne vérifie pas les contraintes. Nous avons principalement étudié la paramétrisation par matrices de projection ainsi que la paramétrisation du tenseur par des méthodes issues de la théorie des déterminants.

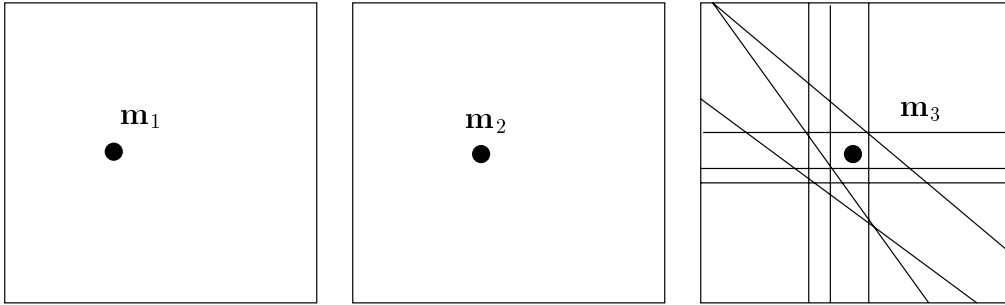


FIG. 3.4 – Critère à minimiser pour obtenir des trilinearités de manière non linéaire. Les points \mathbf{m}_1 et \mathbf{m}_2 engendrent 9 droites dans la troisième image.

La minimisation qui en résulte comporte 18 variables et $27 \times M$ fonctions (3 fois 9 par point), où M est le nombre de triplets présents dans les images.

3.4 Estimation de \mathbf{P}

Nous avons vu précédemment (à la section 2.5) que les matrices de projection étaient l'élément fédérateur de la géométrie des caméras. Les autres entités (bilinéarités et trilinearités) peuvent être calculées simplement à partir des \mathbf{P} . Il existe de nombreuses manières de les estimer. Nous allons les passer en revue. Elles se divisent en trois familles : on peut partir des matrices fondamentales ou des trilinearités que l'on aurait estimées auparavant, ou bien les calculer directement à partir de correspondances de points ou de droites.

3.4.1 Estimation de \mathbf{P} à partir de \mathbf{F}

Il est clair que ce calcul des matrices \mathbf{P} en fonction des matrices \mathbf{F} ne peut pas toujours être mené à bien. En effet, la situation va se bloquer dès que l'on va se trouver face à trois centres optiques alignés. Elle peut se débloquent si l'on peut se permettre de choisir les caméras dans un ordre tel que trois centres optiques consécutifs ne soient jamais alignés ou bien si l'on a des appariements entre triplets d'images, et pas seulement entre paires d'images. Nous supposons cette hypothèse réalisée dans cette section. Elle est effectivement réalisée dans l'immense majorité des cas pratiques.

Nous procéderons alors comme suit : à partir d'une paire de caméras (que nous noterons 1 et 2), nous pouvons calculer \mathbf{P}_i en nous aidant de \mathbf{F}_{1i} et \mathbf{F}_{2i} comme nous l'avons décrit à la section 2.5.1. Il faut d'abord fixer \mathbf{P}_1 et \mathbf{P}_2 en

fonction de \mathbf{F}_{12} , puis \mathbf{P}_i en fonction de \mathbf{F}_{1i} et \mathbf{F}_{2i} . En faisant varier i de 3 à N , nous pouvons obtenir une description complète et cohérente de la géométrie. Nous avons choisi ici un sous ensemble des matrices fondamentales qui permet d'engendrer l'ensemble de la géométrie épipolaire. D'autres sous-ensembles existent, comme celui décrit dans la figure 3.3. Pour qu'un sous-ensemble de matrices fondamentales puisse engendrer l'ensemble de la géométrie épipolaire, il suffit qu'il engendre toutes les matrices de projection. En effet, une fois les matrices de projection connues, les matrices fondamentales se déduisent grâce aux équations (2.40). Pour pouvoir calculer les matrices de projection à partir des matrices fondamentales, il faut que chaque image soit reliée à deux autres images reliées entre elles. Il existe un grand nombre de graphes vérifiant ces contraintes. La figure 3.3 est un exemple parmi d'autres.

En pratique, on a à notre disposition beaucoup plus de matrices fondamentales que le nombre minimal nécessaire. Si l'on calcule des matrices de projection en utilisant différentes matrices fondamentales, on obtient plusieurs valeurs a priori différentes des matrices de projection. On utilise alors une approximation pour combiner ces valeurs, le barycentre projectif. La méthode de calcul de ce vecteur est détaillée dans l'appendice B. Il nous permet de combiner plusieurs valeurs estimées pour le même objet projectif. Il faut aussi bien sûr faire attention d'exprimer toutes les matrices de projection dans la même base de \mathcal{P}^3 .

Le sous-ensemble que nous avons choisi présente plusieurs inconvénients :

- Le calcul est extrêmement sensible à la qualité de \mathbf{F}_{12} , qui intervient directement dans le calcul de tous les \mathbf{P}_i . On peut bien sûr choisir les caméras 1 et 2 en fonction de la qualité des résidus obtenus à la fin de la minimisation non linéaire, mais cette mesure ne garantit qu'une adéquation aux données, pas une mesure d'erreur par rapport à un absolu non disponible. Le choix est alors risqué, et il est bon de faire plusieurs tentatives avec des caméras de référence différentes.
- Elle utilise uniquement les matrices \mathbf{F}_{12} , \mathbf{F}_{1i} , \mathbf{F}_{2i} , et ignore toutes les autres qui, simplement redondantes en théorie, assureraient en pratique au système une robustesse et permettraient une amélioration notable de la qualité des résultats. Ce défaut peut être compensé par une moyenne projective des valeurs obtenues en utilisant différents sous ensembles.

On peut aussi faire une minimiser la distance entre les matrices fondamentales engendrées par les matrices de projection et les matrices fondamentales initiales. La difficulté réside principalement dans le choix d'une distance entre matrices fondamentales. On peut en définir plusieurs types :

- La distance relative entre les épipoles comme dans [Luong, 1992]. Cette

distance a l'avantage d'être simple, et l'inconvénient de peu pénaliser une différence d'orientation dans les droites épipolaires par rapport à un simple décalage de l'épipole.

- Une distance sur les vecteurs de \mathcal{P}^9 . On range les neuf éléments de la matrice dans un vecteur que l'on normalise à 1. La distance est alors définie comme le cosinus de deux tels vecteurs. Cette distance ne tient pas compte des contraintes internes à la matrice fondamentale comme $\det(\mathbf{F}_{ij}) = 0$.
- La distance sur les 7 paramètres indépendants de la matrice fondamentale. L'inconvénient de cette distance est qu'elle ne représente aucune grandeur physique palpable et que les différents paramètres n'ont pas le même ordre de grandeur.
- La distance moyenne d'un point à l'épipolaire générée à partir d'une correspondance parfaite selon la première matrice fondamentale et de la deuxième matrice fondamentale. Cette distance est une distance statistique. En effet, il n'est pas possible de parcourir tout l'espace des correspondances parfaites au sens de la première matrice fondamentale. Nous devons donc échantillonner cet espace et effectuer une moyenne sur les échantillons. Cette distance n'est pas calculable non plus de manière formelle à partir d'intégrales. Elle est décrite dans la figure 3.5.

C'est véritablement la distance qui nous intéresse, elle mesure quelque chose de géométrique qui a une signification dans l'image. Elle a cependant un grave défaut : elle n'est pas calculable simplement et on doit avoir recours à des méthodes statistiques. Le temps de calcul de cette distance la rend inutilisable en pratique à l'intérieur d'une minimisation.

Nous avons utilisé la troisième distance mentionnée. En effet, quelques essais avec la dernière distance ont donné des résultats similaires à la troisième, mais au bout de quelques heures.

3.4.2 Estimation de \mathbf{P} à partir de \mathbf{T}

Il suffit de reprendre les calculs de la section 2.5 pour transformer un ensemble de trilinearités en trois matrices de projection. Si l'on cherche à calculer plus de trois matrices de projection, il faut utiliser une estimation en cascade. On estimera d'abord \mathbf{P}_1 , \mathbf{P}_2 et \mathbf{P}_3 puis \mathbf{P}_2 , \mathbf{P}_3 et \mathbf{P}_4 et ainsi de suite jusqu'à \mathbf{P}_{N-2} , \mathbf{P}_{N-1} et \mathbf{P}_N . Ces groupes de matrices seront estimés

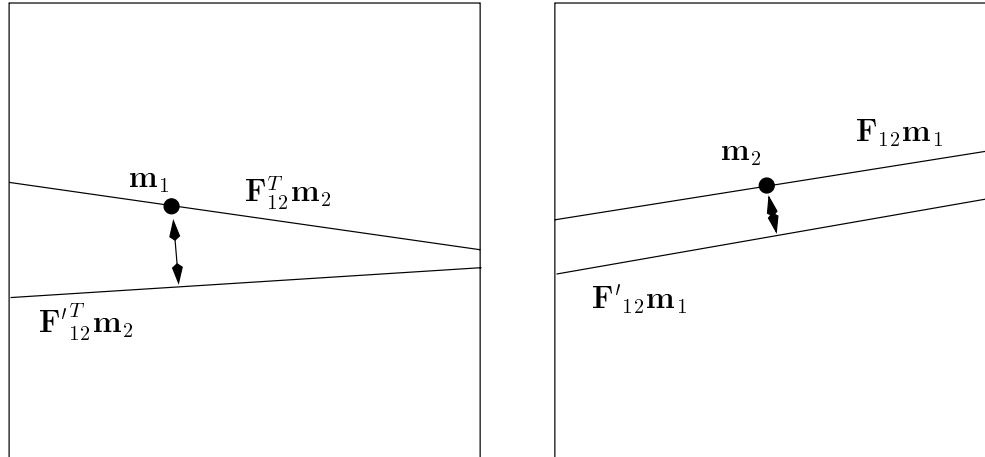


FIG. 3.5 – Distance entre deux matrices fondamentales. Cette distance est estimée sur la zone de l'espace qui est visible dans les deux images. Des points $(\mathbf{m}_1, \mathbf{m}_2)$ vérifiant la géométrie épipolaire de la première matrice fondamentale \mathbf{F} sont tirés au hasard. On vérifie leur adéquation à la deuxième matrice fondamentale \mathbf{F}' .

dans des bases projectives différentes, il faudra donc utiliser les techniques décrites plus loin (à la section 3.5) pour ramener ces estimations dans des bases différentes à une base unique de \mathcal{P}^3 . Les différentes estimations de la matrice de projection pour une caméra seront moyennées suivant la méthode développée à l'appendice B.

3.4.3 Estimation directe de \mathbf{P}

Pourquoi ne calculerions nous pas les matrices de projection directement ? En effet, une matrice de projection n'est rien d'autre qu'une matrice de changement de base multipliée par une matrice de projection canonique, multipliée par une autre matrice de changement de base. Il nous suffirait donc de choisir une base de \mathcal{P}^3 , la même pour toutes les caméras, et de connaître les images des points la composant dans toutes les images. Pour que cet ensemble de points soit une base de \mathcal{P}^3 , il faut qu'il n'y ait pas de groupe de quatre points qui soient coplanaires. Il ne nous resterait plus alors qu'à “calibrer” la caméra à l'aide de ces cinq points pour obtenir une représentation de la géométrie. Malheureusement, les cinq points engendrent dix équations pour une matrice qui compte onze paramètres. Il nous faut donc une donnée

supplémentaire. Nous avons choisi d'utiliser un épipole plutôt qu'un sixième point par facilité, car les équations sont plus simples dans ce cas. La méthode n'est donc pas véritablement directe dans ce cas puisqu'elle utilise les épipoles calculés à partir des matrices fondamentales. On peut la rendre directe en utilisant six points quelconques.

Les matrices de projection vont être calculées par paires, mais il va de soi que si toutes les paires sont calculées dans la même base, la matrice \mathbf{P}_i ne dépendra pas de l'image j avec laquelle elle a été associée, sauf en ce qui concerne l'épipole. Si nous estimons les matrices de projection à partir de 6 points, ce n'est plus le cas. La matrice \mathbf{P}_i dépend alors uniquement des points présents dans l'image i . Cette variante permet de ne plus dépendre de l'épipole dont l'estimation est parfois instable [Luong, 1992].

Nous présentons ici le calcul de \mathbf{P}_i et \mathbf{P}_j en fonction de 5 points \mathbf{a}_i^k et \mathbf{a}_j^k , k variant de 1 à 5 et des épipoles \mathbf{e}_{ij} et \mathbf{e}_{ji} estimés à partir des matrices fondamentales ou bien directement. Les points 3-D correspondants aux \mathbf{a}^k seront notés \mathbf{A}^k . Leur coordonnées sont pour l'instant inconnues.

Nous choisissons maintenant comme base canonique de \mathcal{P}^3 la base \mathbf{A}^k , $k \in \{1, \dots, 5\}$. On a $\mathbf{A}^1 = [0, 0, 0, 1]^T, \dots, \mathbf{A}^5 = [1, 1, 1, 1]^T$. On va aussi effectuer un changement de base projective dans les images pour que les points \mathbf{a}_i^k et \mathbf{a}_j^k , pour k variant de 1 à 4 représentent la base projective standard de \mathcal{P}^2 . On a alors $\mathbf{a}_i^1 = [0, 0, 1]^T$ par exemple. Ces changements de bases vont nous permettre d'exprimer simplement les matrices \mathbf{P} . En effet, si on note

$$\mathbf{P}\mathbf{A}^k = \rho_k \mathbf{a}^k \quad k = 1, \dots, 4 \quad (3.3)$$

On obtient une forme simple de \mathbf{P} pour les images i et j :

$$\mathbf{P} = \begin{pmatrix} \rho_1 & 0 & 0 & \rho_4 \\ 0 & \rho_2 & 0 & \rho_4 \\ 0 & 0 & \rho_3 & \rho_4 \end{pmatrix} \quad (3.4)$$

Si $\mathbf{a}^5 = [\alpha, \beta, \gamma]^T$, alors la relation $\mathbf{P}\mathbf{A}^5 = \rho_5 \mathbf{a}^5$ donne les trois équations

$$\begin{aligned} \rho_1 + \rho_4 &= \rho_5 \alpha \\ \rho_2 + \rho_4 &= \rho_5 \beta \\ \rho_3 + \rho_4 &= \rho_5 \gamma \end{aligned}$$

Les matrices de projections peuvent alors s'exprimer différemment en posant $x = \rho_5 / \rho_4$

$$\mathbf{P} = \begin{pmatrix} \alpha x - 1 & 0 & 0 & 1 \\ 0 & \beta x - 1 & 0 & 1 \\ 0 & 0 & \gamma x - 1 & 1 \end{pmatrix} \quad (3.5)$$

Cette expression est bien entendu valable pour les images i et j avec un paramètre x différent. On peut en déduire les coordonnées de C_i et de C_j puisque $\mathbf{PC} = 0$. On obtient alors

$$\mathbf{C} = \left(\frac{1}{1 - \alpha x}, \frac{1}{1 - \beta x}, \frac{1}{1 - \gamma x}, 1 \right)^T \quad (3.6)$$

Nous pouvons donc déterminer les épipoles \mathbf{e}_{ij} et \mathbf{e}_{ji} . En effet, $\mathbf{e}_{ij} = \mathbf{P}_i \mathbf{C}_j$, soit

$$\mathbf{e}_{ij} = \left(\frac{\alpha_i x_i - \alpha_j x_j}{1 - \alpha_j x_j}, \frac{\beta_i x_i - \beta_j x_j}{1 - \beta_j x_j}, \frac{\gamma_i x_i - \gamma_j x_j}{1 - \gamma_j x_j} \right)^T \quad (3.7)$$

La connaissance de \mathbf{e}_{ij} est suffisante pour déterminer les deux paramètres x_i et x_j . Les α , β et γ sont connus, puisque les coordonnées des points \mathbf{a}^5 sont connues. La démonstration exacte se trouve dans [Faugeras, 1992]. Il se trouve alors que les solutions pour x_i et x_j ,

$$\begin{aligned} x_i &= \frac{\mathbf{e}_{ji} \cdot (\mathbf{a}_i^5 \times \mathbf{a}_j^5)}{\mathbf{v}_1 \cdot (\mathbf{a}_i^5 \times \mathbf{a}_j^5)} \\ x_j &= \frac{\mathbf{a}_i^5 \cdot \mathbf{v}_2}{\mathbf{a}_j^5 \cdot \mathbf{v}_2} x_i \end{aligned}$$

avec

$$\begin{aligned} \mathbf{v}_1 &= \begin{pmatrix} \alpha_i \mathbf{e}_{ji_x} \\ \beta_i \mathbf{e}_{ji_y} \\ \gamma_i \mathbf{e}_{ji_z} \end{pmatrix} \\ \mathbf{v}_2 &= \begin{pmatrix} (\gamma_i - \beta_i) \mathbf{e}_{ji_y} \mathbf{e}_{ji_z} \\ (\alpha_i - \gamma_i) \mathbf{e}_{ji_z} \mathbf{e}_{ji_x} \\ (\beta_i - \alpha_i) \mathbf{e}_{ji_x} \mathbf{e}_{ji_y} \end{pmatrix} \end{aligned}$$

On remplace x_i et x_j dans les expressions de \mathbf{P}_i et \mathbf{P}_j .

3.4.4 Méthodes robustes

Le problème de l'approche développée à la section précédente est que si l'un des points est faux pour une image alors tout notre raisonnement s'écroule et les matrices de projection calculées pour cette image sont fausses. De plus, il est à première vue difficile de garantir que les points choisis dans les images ne forment pas une structure dégénérée. En effet, nous avons vu que ces cinq points formaient une base de l'espace projectif \mathcal{P}^3 si et seulement si il n'existait pas de sous-ensemble de quatre points coplanaires. Certaines approches à base d'invariants existent pour détecter les configurations dégénérées [Rothwell et al., 1993], mais il n'existe pas de métrique pour déterminer l'éloignement d'une base de la configuration dégénérée. Dans le monde euclidien une telle mesure serait le volume engendré par la base ou de manière équivalente le déterminant des trois vecteurs unitaires composant la base. Une telle mesure n'existe malheureusement pas en géométrie projective.

Il nous faut donc une mesure qui nous permette de décider si une base choisie est bonne ou mauvaise. Si la base est bonne, elle va correctement expliquer la scène tridimensionnelle en fonction des images. Cela signifie que les points en correspondance vont donner lieu à des triangulations pour lesquelles les rayons se couperont en un seul et même point. Si les matrices de projection sont fausses, ces reconstructions vont être fantaisistes, puisque pour la majorité de points, les rayons issus des images ne se couperont pas. On peut cependant effectuer une reconstruction approchée de ces points 3-D comme expliqué à l'appendice B.3.

Afin de tirer une grandeur significative de cette conjecture, nous pouvons reprojeter les points triangulés dans les images. Si les rayons se coupent bien en un point, alors pour toutes les images, les points reprojétés doivent se trouver près des points originaux. Si tel n'est pas le cas, la distance des points reprojétés au points originaux va être élevée. La formulation mathématique de cette distance est la suivante:

$$C(\mathbf{P}_1, \dots, \mathbf{P}_N, \mathbf{M}_j) = \sqrt{\sum_{i=0}^N \left(\frac{\mathbf{m}_{ij}^1}{\mathbf{m}_{ij}^3} - \frac{(\mathbf{P}_i \mathbf{M}_j)^1}{(\mathbf{P}_i \mathbf{M}_j)^3} \right)^2 + \left(\frac{\mathbf{m}_{ij}^2}{\mathbf{m}_{ij}^3} - \frac{(\mathbf{P}_i \mathbf{M}_j)^2}{(\mathbf{P}_i \mathbf{M}_j)^3} \right)^2} \quad (3.8)$$

Celle ci permet de mesurer si un point est correct sous l'hypothèse que les matrices de projection le soient. On peut aussi la raffiner en choisissant de séparer les termes suivant les images. Le critère devient alors :

$$C(\mathbf{P}_i, \mathbf{M}_j) = \sqrt{\left(\frac{\mathbf{m}_{ij}^1}{\mathbf{m}_{ij}^3} - \frac{(\mathbf{P}_i \mathbf{M}_j)^1}{(\mathbf{P}_i \mathbf{M}_j)^3} \right)^2 + \left(\frac{\mathbf{m}_{ij}^2}{\mathbf{m}_{ij}^3} - \frac{(\mathbf{P}_i \mathbf{M}_j)^2}{(\mathbf{P}_i \mathbf{M}_j)^3} \right)^2} \quad (3.9)$$

On peut alors dire si un point est faux, et dans quelles images. En effet, un point faux dans l'image i seulement aura un critère élevé dans cette image et faible dans les autres.

La procédure que nous allons suivre est une procédure classique d'estimation robuste, l'estimation à la moindre médiane des carrés. Le lecteur est reporté à l'appendice A pour plus d'informations concernant les méthodes d'estimation robuste. Nous allons tirer au hasard un certain nombre de bases possibles (en pratique de l'ordre de la centaine au millier), ce nombre dépendant de la probabilité de trouver une base correcte dans toutes les images. Nous allons ensuite calculer les matrices de projection pour toutes ces bases, puis les distances de reprojection pour tous les autres points de la scène. Nous choisirons alors la base pour laquelle la médiane des distances de reprojection est la plus faible. Les points pour lesquels la distance de reprojection est bien supérieure à la médiane ("bien supérieur" est déterminé comme à l'appendice A) sont considérés comme faux et éliminés des calculs pour la suite.

Cette méthode a l'avantage de sélectionner une bonne base sur ses résultats (faible distance de reprojection) et non sur sa possibilité de bons résultats (non dégénérescence). L'expérience prouve que les bases choisies par l'algorithme sont en général composées de points bien répartis dans la scène tant au niveau de la profondeur que des directions. De plus, cette méthode permet non seulement de sélectionner la meilleure base possible parmi celles tirées au sort, mais aussi d'éliminer les points qui sont manifestement faux.

Les deux façons de calculer les matrices de projection de manière directe (avec l'épipole comme sixième point ou non) ont une complexité et des résultats différents suivant les cas. Avec l'épipole, la procédure est plus rapide puisque moins de tirages sont nécessaires pour obtenir une base correcte. Par contre, si l'épipole est mal estimé, la méthode à six points donne bien sûr de meilleurs résultats.

3.4.5 Raffinements non linéaires

Dans cette section, nous allons nous raffiner la géométrie des caméras obtenues par n'importe laquelle des méthodes décrites depuis le début de ce chapitre. Ces méthodes fournissent toutes une estimée initiale de la géométrie qui peut être améliorée.

Le raffinement n'est pas toujours nécessaire. En effet, si l'estimée initiale est bonne, on peut se dispenser de l'étape de raffinement. Toutefois, les méthodes d'estimation initiale que nous avons décrites nécessitent le plus souvent un raffinement, et ceci pour plusieurs raisons :

- Pour la méthode directe, avec ou sans l'estimation robuste, nous n'avons

pris en compte que cinq correspondances et un épipôle. Les autres correspondances à notre disposition sont ignorées. Il est important de les faire intervenir dans le résultat final en vérifiant que la géométrie est correcte sur tout les points et qu'il n'en n'existe pas qui jouent un rôle particulier. En utilisant tous les points lors de l'étape de raffinement, nous éliminons ce défaut.

- Pour les méthodes basées sur les trilinearités ou sur les matrices fondamentales, tous les points sont pris en compte au moment du calcul des trilinearités ou des matrices fondamentales. Toutefois, le passage d'une représentation \mathbf{F} ou T à celle des matrices \mathbf{P} s'apparente à une projection sur une sous variété. Plusieurs projections sur cette sous variété peuvent être utilisées. Il n'est pas garanti que la nôtre donne un bon résultat par rapport au point dans les images. L'étape de raffinement consiste alors en un déplacement sur la variété.

Ces raffinements peuvent ne pas être nécessaires si

- la méthode robuste donne une erreur médiane très faible pour la base choisie
- la projection sur la sous variété a entraîné une faible variation des paramètres, ou autrement dit, si les matrices \mathbf{F} ou les trilinearités vérifiaient déjà les contraintes.

L'algorithme de Levenberg-Marquardt est un algorithme bien connu pour effectuer des minimisations non-linéaires. Toutefois, il est ici décrit en détail car c'est un ingrédient fondamental des méthodes présentées à cette section.

3.4.5.1 Minimisation de Levenberg-Marquardt

Étant donnée une relation vectorielle $\mathbf{y} = f(\mathbf{x})$ où \mathbf{y} et \mathbf{x} sont des vecteurs d'espace \mathbb{R}^n et \mathbb{R}^m et une mesure $\hat{\mathbf{y}}$ pour \mathbf{y} , nous désirons trouver le vecteur $\hat{\mathbf{x}}$ qui satisfait le mieux la relation vectorielle. Plus précisément, nous cherchons le vecteur $\hat{\mathbf{x}}$ satisfaisant $\hat{\mathbf{y}} = f(\hat{\mathbf{x}}) + \hat{\mathbf{e}}$ pour lequel $\|\hat{\mathbf{e}}\|$ est minimal.

Itération de Newton La méthode de l'itération de Newton démarre avec une estimée initiale \mathbf{x}_0 et raffine cette estimée avec l'hypothèse que f est localement linéaire. Si l'on développe $f(\mathbf{x}_0 + \mathbf{\Delta})$ au premier ordre en \mathbf{x}_0 , on obtient :

$$f(\mathbf{x}_0 + \mathbf{\Delta}) = f(\mathbf{x}_0) + \mathbf{J}\mathbf{\Delta}$$

où \mathbf{J} est la matrice jacobienne et Δ un petit déplacement dans \mathbb{R}^m . Si nous choisissons $\mathbf{x}_1 = \mathbf{x}_0 + \Delta$, nous obtenons $\hat{\mathbf{y}} - f(\mathbf{x}_1) = \hat{\mathbf{y}} - f(\mathbf{x}_0) - \mathbf{J}\Delta = \hat{\boldsymbol{\epsilon}}_0 - \mathbf{J}\Delta$. Nous devons donc choisir Δ pour que $\|\hat{\boldsymbol{\epsilon}}_0 - \mathbf{J}\Delta\|$ soit minimum. Ceci est un problème de moindres carrés linéaires. Une simple dérivation nous donne la contrainte

$$\mathbf{J}^T \mathbf{J} \Delta = \mathbf{J}^T \hat{\boldsymbol{\epsilon}}$$

On appelle cette équation l'équation normale. La solution au problème de minimisation est donc obtenue en commençant par une estimée initiale \mathbf{x}_0 et en la raffinant en calculant des approximations successives par

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \Delta_i$$

où Δ_i est la solution des équations normales

$$\mathbf{J}^T \mathbf{J} \Delta_i = \mathbf{J}^T \hat{\boldsymbol{\epsilon}}_i.$$

La matrice \mathbf{J} est la matrice jacobienne évaluée à \mathbf{x}_i et $\hat{\boldsymbol{\epsilon}}_i = \hat{\mathbf{y}} - f(\mathbf{x}_i)$. On espère que cet algorithme converge vers la solution désirée. Malheureusement, il est possible que la procédure itérative converge vers un minimum local ou ne converge pas du tout. Le comportement de cet algorithme itératif dépend beaucoup de l'estimée initiale \mathbf{x}_0 .

L'itération de Levenberg-Marquardt La minimisation de Levenberg-Marquardt est une petite variation de la minimisation de Newton. Les équations normales $\mathbf{N}\Delta = \mathbf{J}^T \mathbf{J} \Delta = \mathbf{J}^T \hat{\boldsymbol{\epsilon}}$ sont remplacées par les équations normales augmentées $\mathbf{N}'\Delta = \mathbf{J}^T \hat{\boldsymbol{\epsilon}}$ où $N'_{ii} = (1 + \delta_{ij}\lambda)N_{ii}$ où δ_{ij} est le symbole de Kronecker.

La valeur λ est initialisée à une valeur donnée, typiquement 10^{-3} . Si la valeur de Δ obtenue en résolvant les équations normales augmentées entraîne une réduction de l'erreur alors l'incrément est accepté et λ est divisé par 10 avant la prochaine itération. Si au contraire Δ entraîne une augmentation de l'erreur, alors λ est multiplié par 10 et les équations normales sont résolues à nouveau jusqu'à ce qu'une valeur de Δ qui réduise l'erreur soit trouvée. La procédure de résolutions répétées des équations normales augmentées jusqu'à ce que Δ réduise l'erreur est une itération de Levenberg-Marquardt.

3.4.5.2 Ajustement des rayons optiques

Cette méthode est connue sous le nom de "bundle adjustment" en anglais dans les communautés de vision par ordinateur et de photogrammétrie. C'est

une méthode ancienne [Brown, 1958], utilisée par les photogrammètres de manière manuelle au début [Slama, 1980], elle a surtout été utilisée dans le cas d'une calibration forte [Brown, 1971, Gruen, 1978, Gruen et Beyer, 1992], avec tous les paramètres intrinsèques et extrinsèques connus, mais rien dans la méthode n'en fait un usage direct. Nous l'avons adapté à la reconstruction projective. La présentation que nous en faisons ici est inspirée de Hartley [Hartley, 1994a].

Le but est simple. Il s'agit d'obtenir à la fois des points \mathbf{M}_j de l'espace et des matrices de projection \mathbf{P}_i qui vérifient au mieux les correspondances trouvées dans les images, c'est à dire qui minimisent la distance D entre les points $\hat{\mathbf{m}}_{ij}$, projetés dans les images des points \mathbf{M}_j , et les points \mathbf{m}_{ij} , seule donnée du problème.

$$\begin{aligned} D &= \sum_i \sum_j (d(\mathbf{m}_{ij}, \mathbf{P}_i \mathbf{M}_j))^2 \\ &= \sum_i \sum_j \varepsilon_{ij}^2 \end{aligned}$$

où d est la distance entre deux points du plan rétinien.

Cette minimisation est énorme. Si l'on note M le nombre de points, on minimise NM fonctions de $11N - 15 + 3M$ variables. Les scènes que nous traitons comporte environ une dizaine de vues et quelques centaines de points. Pour par exemple 10 vues et 400 points, cela ne représente pas moins de 4000 fonctions de 1300 variables. Une telle minimisation, effectuée sans précaution ou modification est bien sûr hors de question.

Lorsque les matrices de projection et la localisation des points de l'espace interviennent à la fois dans la matrice jacobienne qui est la matrice des dérivées partielles de ε_{ij} par rapport aux variables \mathbf{M}_j et \mathbf{P}_i , celle-ci a une structure spécifique dont nous allons faire usage. Il s'agit d'une structure en blocs, creuse, qui va donner des équations pour le calcul des nouvelles itérations très facilement simplifiables .

Dans notre cas, les variables sont de deux types : il s'agit soit des coefficients des matrices de projection, soit des coordonnées de points de \mathcal{P}^3 . Une variation des coordonnées de \mathbf{M}_j va résulter en un changement des \mathbf{m}_{ij} ayant le même indice j . Pareillement, une variation de \mathbf{P}_i va affecter les \mathbf{m}_{ij} de même indice i . En conséquence, la matrice jacobienne \mathbf{J} a une structure creuse ressemblant à celle de la figure 3.6.

La figure montre le cas de quatre points et de trois caméras, mais elle peut aisément être étendue au cas de N caméras et M points. Dans le cas où certains des paramètres sont fixés à une valeur donnée, il disparaissent simplement de la matrice jacobienne. Il s'agira pour nous de \mathbf{P}_1 que nous laisserons constant, puisque son changement équivaldrait à un changement

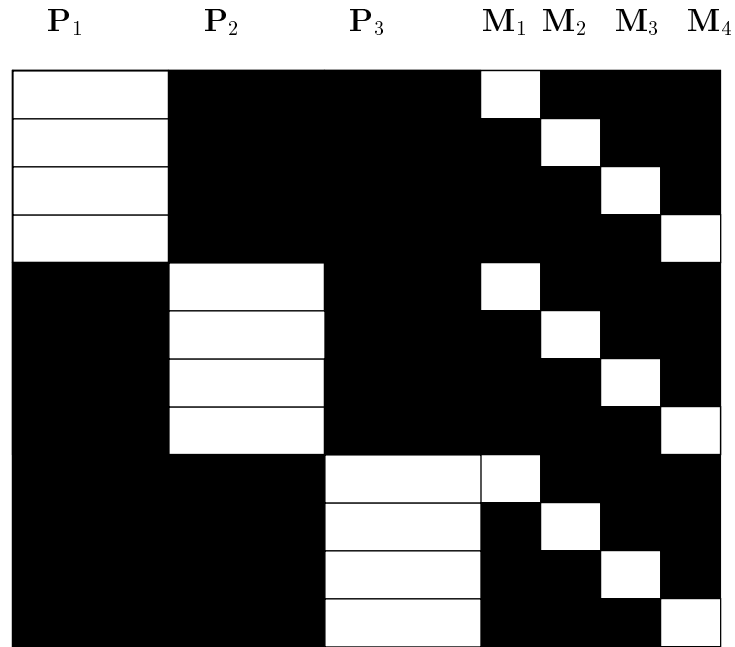


FIG. 3.6 – Matrice jacobienne creuse par blocs. Les parties noires représentent les éléments nuls, les parties blanches représentent les éléments non nuls. Par exemple, la première ligne correspond à ε_{11} et ne dépend que des variables \mathbf{P}_1 et \mathbf{M}_1 , \mathbf{m}_{11} étant fixé. Les dérivées partielles de ε_{11} par rapport aux autres variables sont nulles.

de base projective. Dans certains cas, \mathbf{P}_1 est fixé à sa valeur initiale \mathbf{P}_1^0 , dans d'autre cas, l'ensemble est ramené dans une base telle que $\mathbf{P}_1 = [\mathbf{I}|\mathbf{0}]$. Ceci a relativement peu d'importance pour la suite de la discussion.

A cause de la structure par blocs de \mathbf{J} , la résolution du système $\mathbf{J}^T \mathbf{J} \mathbf{x} = \mathbf{b}$ se présente comme à la figure 3.7. La résolution de ce système est indispensable pour calculer le vecteur d'incrément entre la position courante et la nouvelle itération, qui s'exprime comme $(\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \varepsilon$.

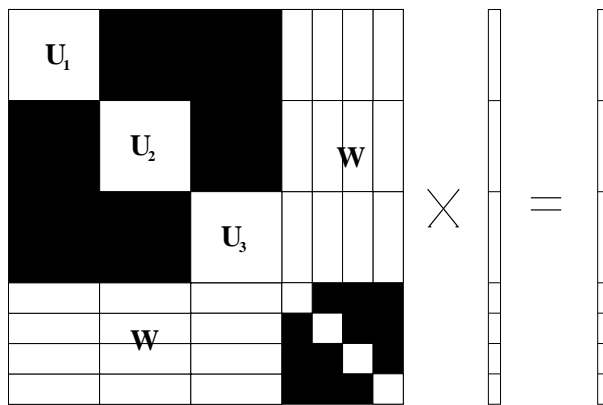


FIG. 3.7 – Équations normales par blocs

Il est possible de donner des formules explicites pour chacun des blocs. Pour ceci, il nous faut définir la matrice des dérivées partielles d'un vecteur par rapport à un autre. Si $\mathbf{f}(\mathbf{x})$ est une fonction de E^n vers E^p , alors, la matrice des dérivées partielles de \mathbf{f} par rapport à \mathbf{x} est une matrice $n \times p$ dont l'élément i, j est la dérivée partielle de la i -ème coordonnée de \mathbf{f} par rapport à la j -ième coordonnée de \mathbf{x} , c'est à dire $\frac{\partial f_i}{\partial x_j}$.

Donc, si $\frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{P}_i}$ est la matrice des dérivées partielles de $\hat{\mathbf{m}}_{ij}$ par rapport au coordonnées de \mathbf{P}_i de dimension $2N \times 11M$ et $\frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{M}_j}$ la matrice des dérivées partielles de $\hat{\mathbf{m}}_{ij}$ par rapport aux coordonnées de \mathbf{M}_j de dimension $2N \times 4M$, si de plus, nous notons ε_{ij} l'erreur résiduelle au point $\hat{\mathbf{m}}_{ij}$, nous pouvons écrire

$$\begin{aligned}
\mathbf{U}_i &= \sum_j \left(\frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{P}_i} \right)^T \frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{P}_i} \\
\mathbf{V}_j &= \sum_i \left(\frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{M}_j} \right)^T \frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{M}_j} \\
\mathbf{W}_{ij} &= \left(\frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{P}_i} \right)^T \frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{M}_j} \\
\varepsilon(\mathbf{P}_i) &= \sum_j \left(\frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{P}_i} \right)^T \varepsilon_{ij} \\
\varepsilon(\mathbf{M}_j) &= \sum_i \left(\frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{M}_j} \right)^T \varepsilon_{ij}
\end{aligned}$$

Les matrices \mathbf{U}_i sont des matrices 11×11 , \mathbf{V}_j des matrices 4×4 , \mathbf{W}_{ij} des matrices 11×4 . Les vecteurs $\varepsilon(\mathbf{M}_j)$ et $\varepsilon(\mathbf{P}_i)$ sont respectivement de dimensions 4 et 11. Les équations normales peuvent s'écrire sous la forme compacte

$$\begin{pmatrix} \mathbf{U} & \mathbf{W} \\ \mathbf{W}^T & \mathbf{V} \end{pmatrix} \begin{pmatrix} \Delta(\mathbf{P}) \\ \Delta(\mathbf{M}) \end{pmatrix} = \begin{pmatrix} \varepsilon(\mathbf{P}) \\ \varepsilon(\mathbf{M}) \end{pmatrix} \quad (3.10)$$

où les matrices \mathbf{U} , \mathbf{V} , les vecteurs $\Delta(\mathbf{P})$, $\Delta(\mathbf{M})$, $\varepsilon(\mathbf{P})$ et $\varepsilon(\mathbf{M})$ sont eux même composés de blocs.

Si nous supposons que la matrice \mathbf{V} est inversible, nous pouvons multiplier chaque membre de l'équation précédente par

$$\begin{pmatrix} \mathbf{I} & -\mathbf{WV}^{-1} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \quad (3.11)$$

pour obtenir

$$\begin{pmatrix} \mathbf{U} - \mathbf{WV}^{-1}\mathbf{W}^T & \mathbf{0} \\ \mathbf{W}^T & \mathbf{V} \end{pmatrix} \begin{pmatrix} \Delta(\mathbf{P}) \\ \Delta(\mathbf{M}) \end{pmatrix} = \begin{pmatrix} \varepsilon(\mathbf{P}) - \mathbf{WV}^{-1}\varepsilon(\mathbf{M}) \\ \varepsilon(\mathbf{M}) \end{pmatrix} \quad (3.12)$$

Ce qui peut être séparé en deux groupes d'équations. La première est

$$(\mathbf{U} - \mathbf{WV}^{-1}\mathbf{W}^T)\Delta(\mathbf{P}) = \varepsilon(\mathbf{P}) - \mathbf{WV}^{-1}\varepsilon(\mathbf{M}) \quad (3.13)$$

que l'on peut résoudre pour trouver $\Delta(\mathbf{P})$. La solution peut donc être substituée dans la deuxième équation, ce qui donne

$$\Delta(\mathbf{M}) = \mathbf{V}^{-1}(\varepsilon(\mathbf{M}) - \mathbf{W}^T \varepsilon(\mathbf{P})) \quad (3.14)$$

Grâce à la forme spécifique de \mathbf{V} , il est possible de calculer très efficacement les quantités intervenant dans l'équation (3.13). Plus spécifiquement, si l'on note \mathbf{A} la matrice $\mathbf{U} - \mathbf{W}\mathbf{V}^{-1}\mathbf{W}^T$ et \mathbf{b} le vecteur $\varepsilon(\mathbf{P}) - \mathbf{W}\mathbf{V}^{-1}\varepsilon(\mathbf{M})$, la matrice bloc \mathbf{A}_{ij} et le vecteur bloc \mathbf{b}_i vérifient

$$\begin{aligned} \mathbf{A}_{ij} &= \delta_{ij} \mathbf{U}_i - \sum_k \mathbf{W}_{ik} \mathbf{V}_k^{-1} \mathbf{W}_{jk}^T \\ \mathbf{b}_i &= \varepsilon(\mathbf{P}_i) - \sum_j \mathbf{W}_{ij} \mathbf{V}_j^{-1} \varepsilon(\mathbf{M}_j) \end{aligned}$$

où δ_{ij} est le symbole de Kronecker.

La matrice \mathbf{A} et le vecteur \mathbf{b} peuvent aussi être calculés directement à partir des dérivées partielles sans qu'il y ait besoin de calculer la matrice jacobienne ou les équations normales. La force vient de ce que nous n'inversons plus une matrice jacobienne fois sa transposée de taille 1300×1300 , mais un grand nombre de matrices 4×4 et de matrices 11×11 , ce qui est une tâche autrement plus facile et plus rapide !

Armés de ces calculs préliminaires, nous pouvons maintenant exposer notre algorithme de minimisation, connu sous le nom de minimisation de Newton. On part d'une solution initiale \mathbf{X} au problème. \mathbf{X} regroupe toutes les variables, c'est à dire tout les \mathbf{M}_j et tout les \mathbf{P}_j . Nous calculons alors la valeur du vecteur ε en ce point et nous estimons la matrice jacobienne où, plus exactement, les matrices \mathbf{U}_i , \mathbf{V}_j et \mathbf{W}_{ij} . Ces matrices peuvent être calculés par approximation des dérivées partielles ou bien par un calcul exact des dérivées partielles. Le calcul exact est ici facile à mener à bien, nous le préférons aux calculs approchés. Ce calcul est présenté à l'appendice C. Nous calculerons alors le vecteur Δ à partir des équations développées plus haut. Ce vecteur Δ et le vecteur \mathbf{X} nous permettent de calculer la nouvelle estimation des variables, c'est à dire un nouveau vecteur \mathbf{X} . On répète l'opération jusqu'à ce que la norme du vecteur Δ soit inférieure à la précision désirée.

L'algorithme vient d'être décrit dans le cas d'une itération de Newton. Il est facile de voir que ceci peut être transformé en une itération de Levenberg-Marquardt en augmentant la matrice jacobienne d'un coefficient λ , ce qui revient à augmenter les valeurs de \mathbf{U} et \mathbf{V} de manière similaire. Ceci nous

permet d'assurer que la matrice \mathbf{V} reste inversible. Dans notre configuration particulière, la matrice \mathbf{V} n'est en général pas inversible car \mathbf{M} peut changer (en particulier d'échelle) sans que $\hat{\mathbf{m}}_{ij}$ change. La matrice \mathbf{V} a donc en général une valeur propre nulle avant d'être augmentée.

Il existe plusieurs extensions de cet algorithme [Okamoto, 81, Forstner et Pertl, 1986, Gruen, 1978, Slama, 1980, Zeller, 1952, Okamoto, 84, Brown, 1976, Zeller, 1996], par exemple pour tenir compte de l'incertitude initiale sur les points et leur donner un poids en conséquence. Cela permet de tenir plus compte des données dont on est sûr que des données douteuses ou peu précises. Dans notre cas, l'incertitude sur les points n'est pas connue, nous n'utiliserons donc pas ces méthodes. D'autres algorithmes sont plus adaptés lorsque des contraintes existent sur les paramètres des différentes caméras. Les paramètres intrinsèques peuvent être constants par exemple. Nous n'avons pas ici fait de supposition particulière sur les caméras utilisées, contrairement à [Zeller, 1996], où l'ajustement de faisceaux se fait avec des paramètres intrinsèques constants au cours de la séquence.

3.4.5.3 Ajustement des lignes épipolaires

La méthode est ici très proche de celle utilisée pour calculer une matrice fondamentale entre deux images. La minimisation non linéaire que nous effectuons a comme paramètres \mathbf{F}_{12} et les matrices de projection $\mathbf{P}_3, \dots, \mathbf{P}_N$. Les matrices de projection sont normalisées de manière à ne conserver que 11 paramètres. Pour ce faire, on peut par exemple fixer un de leurs éléments à 1 ou bien fixer la somme des carrés des éléments à 1. L'ensemble a bien $11N - 15$ degrés de liberté comme souhaité. Les paramètres sont tous indépendants.

Le critère choisi est la somme des distances d'un point aux droites épipolaires générées par ses correspondants dans les autres images. Ce critère s'exprime de la manière suivante:

$$C(\mathbf{F}_{12}, \mathbf{P}_3, \dots, \mathbf{P}_N) = \sum_{i=1}^N \sum_{\mathbf{m}} \sum_{j \neq i} \left(\frac{\mathbf{m}_j^T \mathbf{F}_{ij} \mathbf{m}_i}{\sqrt{(\mathbf{m}_j^T \mathbf{F}_{ij} \mathbf{m}_i)_1^2 + (\mathbf{m}_j^T \mathbf{F}_{ij} \mathbf{m}_i)_2^2}} \right) \quad (3.15)$$

Ce critère n'est en fait que la généralisation de celui choisi pour l'estimation non linéaire de \mathbf{F} (équation 3.2). La figure 3.8 représente le critère pour un choix de i . Ce critère ne privilégie aucune image, même si la paramétrisation privilégie les images 1 et 2. Si une correspondance est absente d'une ou de plusieurs images, les termes sont mis à zéro.

Il faut noter que si la convergence des rayons optiques implique la convergence des lignes épipolaires, la réciproque n'est pas toujours vraie. En effet,

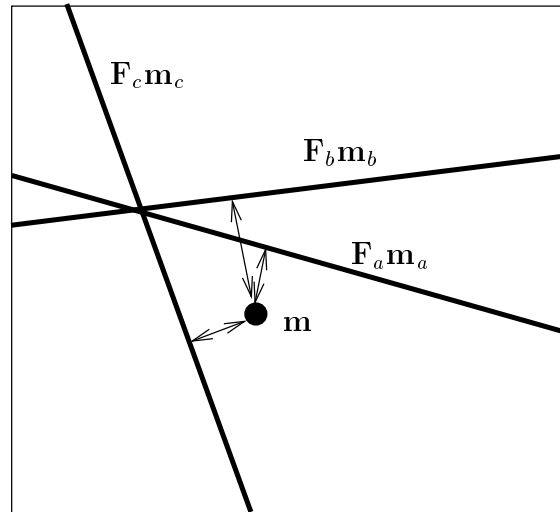


FIG. 3.8 – Critère d'ajustement des droites épipolaires

lorsque les centres optiques sont alignés (ou quasiment alignés), la distance minimisée n'est plus la distance à plusieurs droites, mais à une seule et unique droite. La convergence peut donc avoir lieu en n'importe quel point de la droite alors qu'un seul d'entre eux achève la convergence au sens des rayons. Cette méthode est donc à déconseiller si le centre optique se déplace sur une ligne au cours de grandes parties de la séquence. On peut toutefois utiliser cette méthode puis la faire suivre d'un ajustement de rayons pour vérifier que l'on ne se trouve pas dans un tel cas. l'ajustement de rayons terminera immédiatement, puisque les minima des deux méthodes sont les mêmes.

3.4.5.4 Ajustement des trilinearités

Nous avons vu au chapitre précédent que les trilinearités sont des équations liant les coordonnées des images d'un point de l'espace. Ces relations sont, comme leur nom l'indique, trilineaires. D'un point de vue pratique, si l'on considère deux des points comme fixés, un tenseur trifocal va engendrer 9 équations différentes suivant les indices que nous déciderons de prendre en compte. Ces équations seront linéaires en les coordonnées du troisième point, elles représentent donc des droites dans la troisième image. Elles sont au nombre de 9 parce que nous avons trois choix possibles pour les coordonnées utilisées dans la deuxième image, c'est à dire (x, y) , (y, z) et (z, x) . Le même choix existe dans la troisième image.

Si l'on revient à l'interprétation géométrique des trilinearités (figure 2.7),

on voit que les trilinearités que nous considérons ici avec notre tenseur T privilégie la première image. En effet, chaque trilinearité est l'expression de l'intersection d'une droite issue de la première image et de deux plans issus des images 2 et 3. Il y a trois choix possibles pour le plan dans la deuxième image et trois choix possibles pour le plan dans la troisième image. Selon le choix du plan dans la deuxième image, le point d'intersection dans l'espace va être différent. Ce point peut être déterminé à partir des deux premières images seulement comme l'intersection d'un plan et d'une droite. Si le couple de points $\mathbf{m}_1, \mathbf{m}_2$ vérifie la géométrie épipolaire, alors les trois plans possibles de la deuxième image couperont la droite issue de la première image en un seul et même point. Si ce n'est pas le cas, ces trois points seront distincts.

Dans la troisième image, chacun de ces trois points engendre trois droites suivant le choix du plan associé à la troisième image. Encore une fois, si les points sont en correspondance épipolaire, les neuf droites se couperont en un seul et même point.

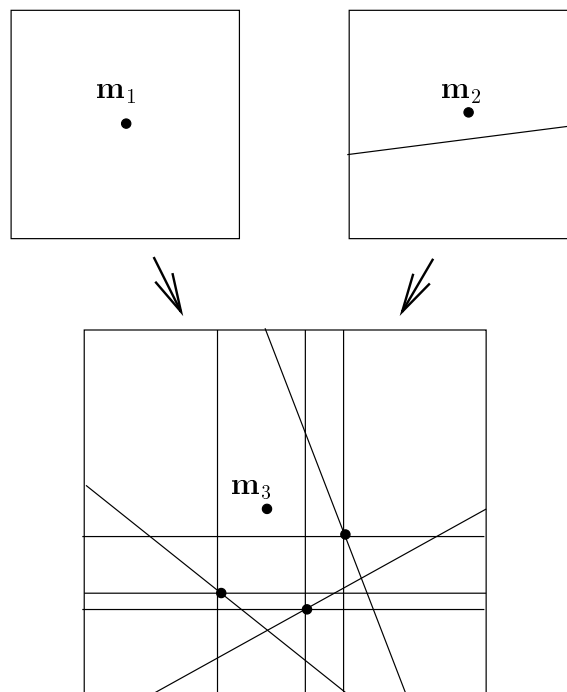


FIG. 3.9 – Droites engendrées dans la troisième image par un tenseur du premier groupe lorsque l'on connaît \mathbf{m}_1 et \mathbf{m}_2 . Les points \mathbf{m}_1 et \mathbf{m}_2 sont présentés ici en situation générale, ils ne vérifient pas la contrainte épipolaire existant entre les images 1 et 2.

Dans la figure 3.9, de nombreuses droites sont représentées comme étant soit verticales, soit horizontales. Ceci n'est pas le fait du hasard, mais simplement du choix des coordonnées (x, z) et (y, z) dans la troisième image. La base image choisie pour cette figure étant un repère pixel orthogonal dans l'image, ces droites sont donc soit verticales soit horizontales. L'orientation des droites engendrées par le troisième choix de coordonnées n'est pas constant.

Le critère que nous utiliserons pour cette étape de raffinement de la géométrie du système est simplement la somme des carrés des distances du point mesuré \mathbf{m}_3 à ces neuf droites obtenues par les trilinearités. Il est bien entendu que, le critère devant être symétrique par rapport à toutes les images, nous allons additionner les résultats obtenus en considérant les paires $(\mathbf{m}_1, \mathbf{m}_2)$, $(\mathbf{m}_2, \mathbf{m}_3)$ et $(\mathbf{m}_3, \mathbf{m}_1)$. Un autre critère possible est la distance de \mathbf{m}_3 à ces trois points particuliers. Ce critère doit aussi être symétrisé. Les résultats seront similaires, mais le temps de calcul sera plus faible.

Il y a six ordres possibles pour choisir les trois images. Nous devons considérer ces six ordres car les trois images jouent des rôles différents suivant leur place dans le triplet. Chacune de ces 6 configurations va générer 9 droites, soit un total de 54 distances pour chaque triplet de points. Le nombre de triplets étant de l'ordre de quelques centaines, la taille du vecteur d'erreur est de quelques milliers.

Lorsque nous avons plus de trois images, nous devons calculer ces distances pour tous les triplets possibles, c'est à dire $N(N-1)(N-2)$ possibilités où N est le nombre d'images.

3.5 D'une description locale à une description globale

En règle générale, nous n'estimerons pas les matrices de projection pour la séquence entière, mais pour des sous-séquences. Il peut se trouver que notre méthode d'estimation marche mal ou ne marche pas pour un certain nombre d'images. C'est le cas par exemple pour l'estimation à base de trilinearités, où l'on considère les matrices par blocs de trois. C'est le cas aussi de la méthode robuste puisque le nombre d'essais varie exponentiellement avec le nombre de caméras. Il peut se trouver aussi que notre séquence couvre un grand nombre d'images et qu'il ne soit pas possible de trouver suffisamment de points en commun dans ces images. C'est par exemple le cas si on cherche à modéliser un objet vu sous toutes ses faces. Il n'y aura alors pas ou peu de points communs à toutes les images.

Il nous faut donc un procédé pour mettre bout à bout des mini-séquences. Mettre bout à bout des sous-ensembles de matrices de projection estimées dans des bases différentes signifie seulement que l'on veut être capable de calculer la matrice de passage d'une base à l'autre de proche en proche pour obtenir l'ensemble de la séquence exprimé dans une même base de l'espace.

Nous allons décrire deux méthodes pour trouver la transformation projective qui ramène deux jeux de matrices de projection sur une même base. Il est clair que si on peut exprimer deux jeux de matrices de projection dans une base commune, on pourra alors sans trop de difficulté exprimer tous les jeux dans une même base en procédant de proche en proche. D'autres méthodes existent si la base d'arrivée est quasiment euclidienne comme dans [Kanatani, 1992].

3.5.1 Changement de base en utilisant les matrices de projection

Supposons que nos deux jeux de matrices se recouvrent sur au moins deux images. Ils peuvent se recouvrir sur plus de deux, les calculs n'en seront que plus stables. Soit $\mathbf{P}_1, \dots, \mathbf{P}_i$ le premier jeu et $\mathbf{Q}_j, \dots, \mathbf{Q}_N$ le deuxième, avec $i - j \geq 2$. Soit \mathbf{H} la collinéation recherchée, celle qui nous fait passer de la base du deuxième jeu à la base du premier. On a pour tout $k \in \{j, \dots, i\}$

$$\mathbf{P}_k = \mathbf{Q}_k \mathbf{H} \quad (3.16)$$

L'équation 3.16 est définie à un facteur d'échelle près, elle est donc équivalente à 11 équations scalaires, linéaires en les coefficients de \mathbf{H} . Il suffit de 15 équations pour résoudre le problème puisque \mathbf{H} dépend de 15 paramètres. Il nous faut donc au minimum deux caméras. Le système est alors résolu avec une méthode aux moindres carrés.

3.5.2 Changement de base en utilisant le nuage de points

Supposons maintenant que les ensembles de points tridimensionnels reconstruits à partir de nos jeux de matrices et les points \mathbf{m} dans les images aient 5 points M_l en commun. Les points sont réputés être en commun si leurs images sont en correspondance. Les ensembles d'images associées aux deux jeux peuvent être disjoints. Il peut y avoir plus de 5 points en commun, les calculs n'en seront que plus stables. Soient \mathbf{M}_l les points estimés dans la base du premier jeu, et \mathbf{M}'_l les points estimés dans la base du deuxième jeu. On a alors, en utilisant les mêmes notations qu'au paragraphe précédent

$$\mathbf{M}_l = \mathbf{H}^{-1}\mathbf{M}'_l \quad (3.17)$$

L'équation 3.17 est aussi linéaire en les coefficients de \mathbf{H}^{-1} , et est aussi définie à un facteur d'échelle près. Elle est donc équivalente à 3 équations scalaires. Il faut alors 5 points pour déterminer complètement \mathbf{H} . En présence d'un plus grand nombre de points, nous utiliserons une méthode aux moindres carrés pour résoudre le système surdéterminé.

3.5.3 Changement de base par recalage dans les images

Il existe d'autres méthodes pour résoudre ce système. En effet, nous avons cherché la collinéation qui transforme le mieux un ensemble de points projectifs en un autre ensemble de points projectifs. La distance que nous minimisons n'a pas de signification géométrique dans les images. On peut alors envisager de chercher la meilleure collinéation dans un sens différent, par exemple, celle qui, une fois les points transformés va assurer une bonne re-projection dans les images. Les points reconstruits dans le deuxième jeu, transformés dans le premier et projetés dans les images du premier jeu devront être près de leur correspondants. La distance mesurée est alors une distance euclidienne en pixels.

3.5.4 Comparaison

Nous présentons ici des résultats statistiques pour comparer les diverses méthodes. Le mode opératoire est le suivant: les matrices de projection initiales sont des matrices de projection euclidiennes, et les points sont dans une sphère devant les caméras¹. Une transformation homographique aléatoire de l'espace est alors imprimée aux points et aux matrices de projection. A partir des deux jeux de points, on calcule cette homographie de 3 manières différentes:

- En utilisant des matrices de projection calculées à partir des points 2-D bruités et des jeux de points 3-D. (P). Cette méthode est celle qui est décrite à la section 3.5.1. Ici, les matrices de projection ont été calculées à partir des équations $\mathbf{m} = \mathbf{PM}$ en connaissant les points \mathbf{M} .
- En utilisant les nuages de points 3-D directement et une résolution aux moindres carrés. (SVD). Cette méthode est celle qui est décrite dans la première partie de la section 3.5.2.

1. Cette information n'est bien sûr pas utilisée dans les procédures de recalage, mais seulement dans le calcul de certaines distances.

- En minimisant l’erreur de reprojection des points 3-D. (Mini). L’estimation initiale utilisée pour cette minimisation est le résultat de la méthode (SVD). Cette méthode est celle qui est décrite dans la section 3.5.3.

Les estimations données par les différents algorithmes sont alors évaluées par rapport aux données exactes par trois distances. On note \mathbf{H}_0 l’homographie exacte, \mathbf{H} l’homographie calculée, \mathbf{h}_0 et \mathbf{h} sont les vecteurs de \mathcal{P}^{15} leur correspondant, \mathbf{M} les points de l’espace dans la première base (euclidienne), \mathbf{m} les points dans les images.

- Distance Euclidienne 3-D. Nos points originaux étaient des points euclidiens. Cette distance mesure la moyenne des distances des points entre eux une fois tous dans la même base euclidienne. Son expression est

$$e = \sum_{\text{points}} \frac{d_{\mathbb{R}^3}(\mathbf{M}, \mathbf{H}_0 \mathbf{H}^{-1} \mathbf{M}) + d_{\mathbb{R}^3}(\mathbf{M}, \mathbf{H} \mathbf{H}_0^{-1} \mathbf{M})}{2}$$

- Distance Angulaire dans \mathcal{P}^{15} . Les homographies de \mathcal{P}^3 vers \mathcal{P}^3 peuvent être représentées comme des vecteurs unitaires de dimension 16. L’angle entre ces vecteurs est une distance. Cette distance existe indépendamment des points choisis. Son expression est

$$a = \arccos\left(\frac{\mathbf{h} \cdot \mathbf{h}_0}{\|\mathbf{h}\| \|\mathbf{h}_0\|}\right)$$

- Distance en pixels dans les images. Cette distance mesure la moyenne des distances entre les points 2-D images des points 3-D ramenés dans la même base. C’est la distance que minimise la méthode Mini. Son expression est

$$p = \sum_{\text{points images}} \sum \frac{d_{\mathbb{R}^2}(\mathbf{PM}, \mathbf{PH}_0 \mathbf{H}^{-1} \mathbf{M}) + d_{\mathbb{R}^2}(\mathbf{PM}, \mathbf{PH} \mathbf{H}_0^{-1} \mathbf{M})}{2}$$

Nous avons fait ces expériences pour 40 homographies différentes et 9 niveaux de bruit (0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.7, 1.0, 1.5 pixels). Les expériences ont aussi été faites pour un nombre de points variable (7, 20, 50, 100). Il faut noter qu’une détection subpixelique des points donnera une précision de l’ordre de 0.2 pixel tandis qu’une détection classique donnera une précision de l’ordre de 1.0 pixel. Les expériences ont été conduites sur 3 images.

40 tirages ont été effectués pour chaque niveau de bruit. Nous avons utilisé un bruit uniforme. Le bruit uniforme nous a paru plus adapté pour modéliser l'erreur de localisation des points d'intérêt dans une image que le modèle gaussien. En effet, le modèle gaussien autorise des erreurs infinies, ce qui n'arrive pas en pratique. Ce modèle est utilisé en reconnaissance d'objets par d'autres auteurs désireux de faire des statistiques probantes avec des tirages de Monte-Carlo [Grimson et al., 1994, Rothwell, 1995]. Les résultats présentés sont les moyennes sur tous les essais. Il faut noter que les erreurs médianes différaient très faiblement des erreurs moyennes dans nos expériences. Les deux jeux de points ont été bruités différemment. L'initialisation de la méthode Mini a été la méthode Svd. Des résultats différents auraient été obtenus avec les valeurs données par P pour point de départ.

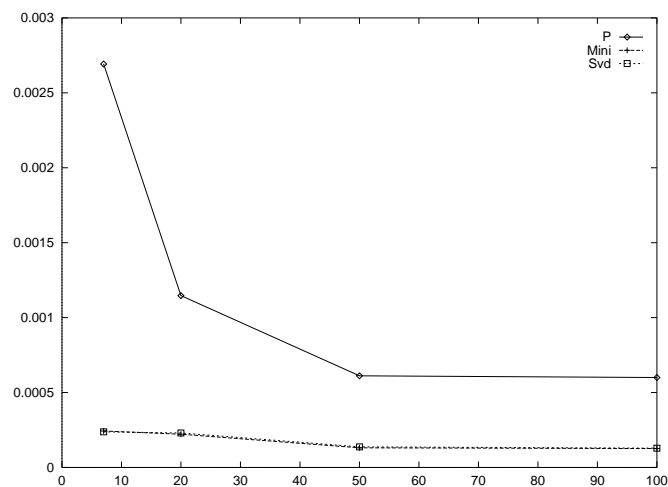


FIG. 3.10 – Évolution des résultats des différents algorithmes lorsque le nombre de points varie. Le bruit est ici de 0.3 pixel. Les erreurs de la méthode (P) ont été divisées par 10^3 . La distance est la distance angulaire dans \mathcal{P}^{15} .

Les méthodes SVD et Mini semblent donc relativement peu sensibles au nombre de points, même si les résultats s'améliorent légèrement avec le nombre de points comme on peut le voir sur les figures 3.10, 3.11 et 3.12. La méthode basée sur les matrices de projection marche nettement moins bien. Toutes les méthodes fonctionnent mieux avec un plus grand nombre de points, ce qui n'est pas surprenant. Les algorithmes SVD et Mini se comportent tout deux de manière similaire.

De l'interprétation des résultats présentées dans les figures 3.13, 3.14 et 3.15, il ressort que la méthode P est instable lorsque le bruit augmente (supérieur à 0.5 pixel). Les deux autres méthodes sont relativement robustes au

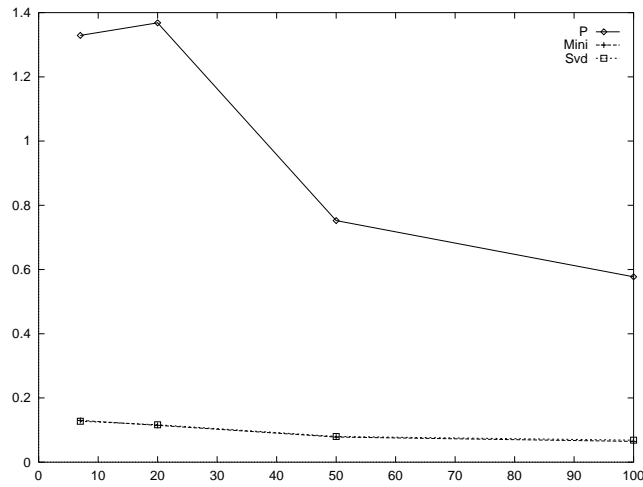


FIG. 3.11 – Évolution des résultats des différents algorithmes lorsque le nombre de points varie. Le bruit est ici de 0.3 pixel. Les erreurs de la méthode (P) a été divisées par 10^2 . La distance est la distance euclidienne 3-D.

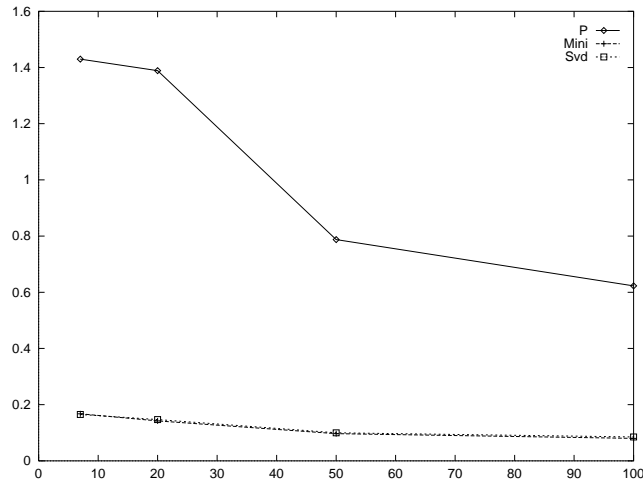


FIG. 3.12 – Évolution des résultats des différents algorithmes lorsque le nombre de points varie. Le bruit est ici de 0.3 pixel. Les erreurs de la méthode (P) a été divisées par 10^2 . La distance est la distance pixel dans les images.

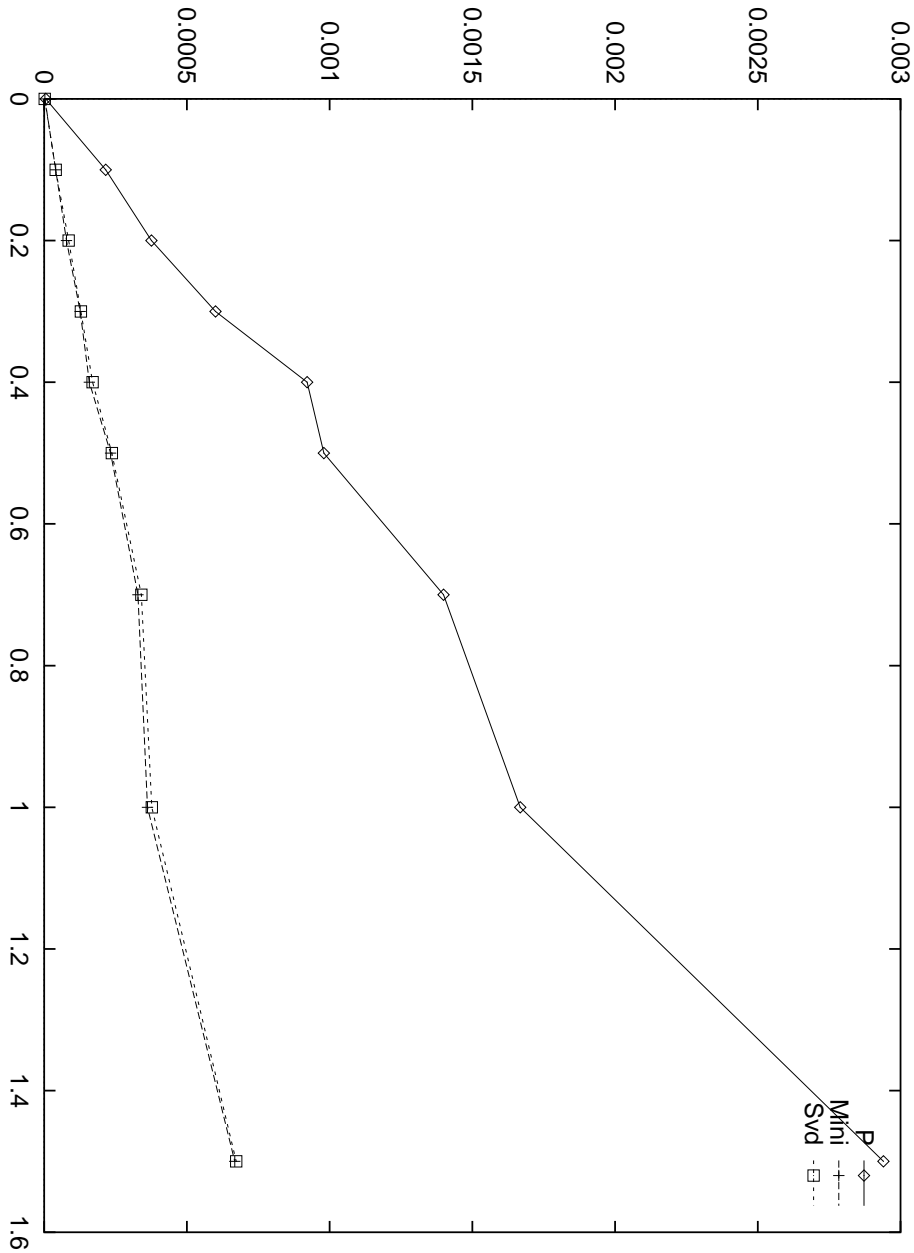


FIG. 3.13 – Évolution des résultats des différents algorithmes lorsque le bruit varie. La distance présentée ici est la distance angulaire. Les résultats ont été obtenus avec 100 points. Les erreurs des algorithmes (Mini) et (SVD) ont été multipliées par 10^3 pour obtenir le même ordre de grandeur.

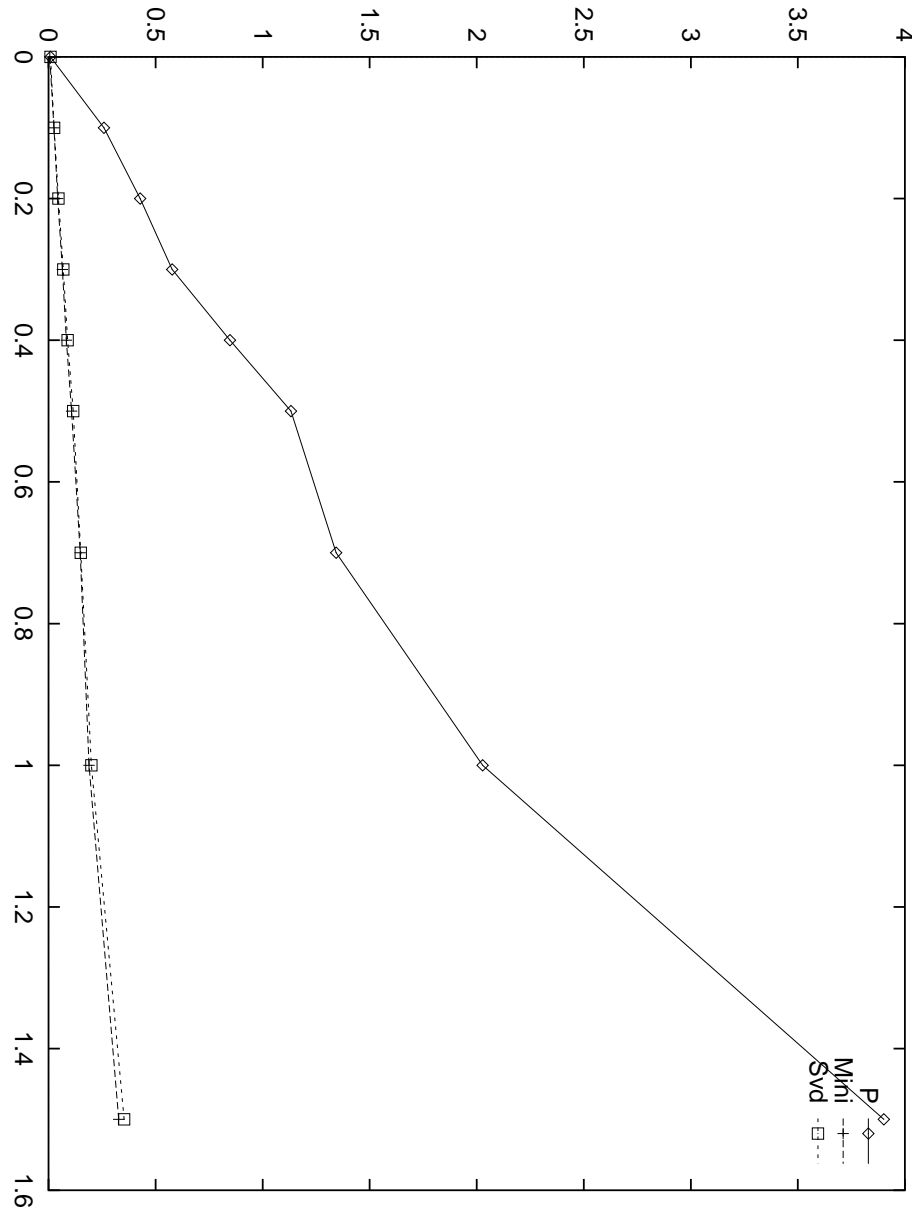


FIG. 3.14 – Évolution des résultats des différents algorithmes lorsque le bruit varie. La distance présentée ici est la distance euclidienne de l'espace. Les résultats ont été obtenus avec 100 points. L'erreur de l'algorithme (P) a été multipliée par 10^{-2} pour obtenir le même ordre de grandeur.

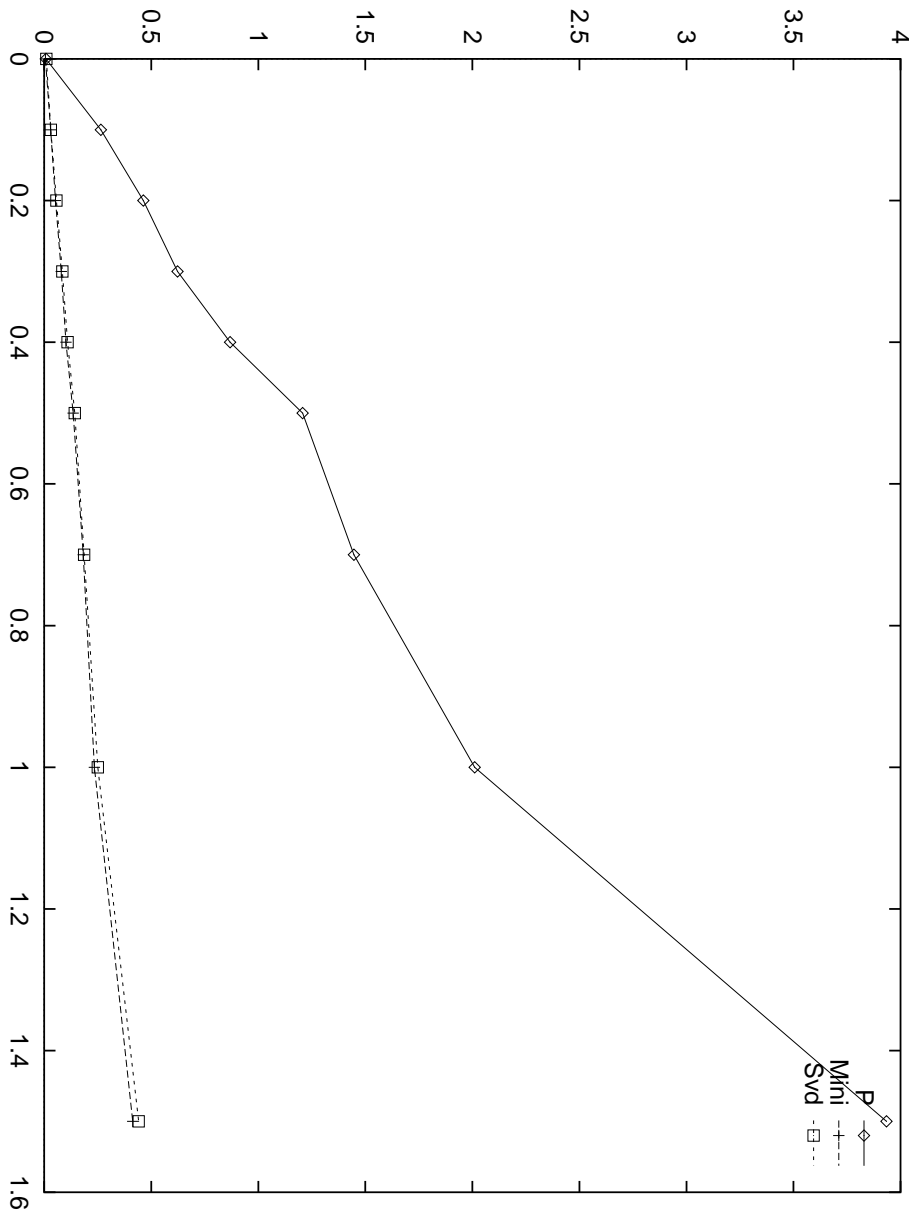


FIG. 3.15 – Évolution des résultats des différents algorithmes lorsque le bruit varie. La distance présentée ici est la distance entre les points dans les images. Les résultats ont été obtenus avec 100 points. L'erreur de l'algorithme (P) a été multipliée par 10^{-2} pour obtenir le même ordre de grandeur.

bruit, les erreurs restent très petites pour tous les niveaux de bruit.

La différence de qualité entre les méthodes Mini et SVD est très peu sensible. Or, l'utilisation d'une minimisation non linéaire rend les temps de calcul nettement plus longs. L'intérêt de la méthode Mini est donc minime. Des résultats différents auraient pu être obtenus avec une initialisation différente.

Une des raisons pour laquelle la méthode P fonctionne moins bien est qu'elle opère sur $3N$ plans qui sont les trois lignes des N matrices de projection au lieu d'opérer sur les M points. Plus précisément, les équations (3.16) sont des équations liant deux plans de l'espace tridimensionnel. En effet, comme nous l'avons vu à la section 2.1.3, les lignes des matrices de projection représentent des plans passant par le centre optique. Les équations (3.16) sont la formulation algébrique du fait que ces plans exprimés dans la base du premier jeu doit se transformer en le même plan exprimé dans la base du second jeu. On applique en fait le changement de base aux trois plans qui définissent la caméra. Le fait d'opérer sur les données originales (les points) semble être un avantage pour les méthodes SVD et Mini.

3.6 Expériences et comparaison

Dans cette section nous présentons les résultats et la comparaison des différentes méthodes sur des jeux de données synthétiques avec un bruit ajouté variable, et sur des séquences réelles.

3.6.1 Données synthétiques

Les niveaux de bruit choisis sont de 0.1, 0.2, 0.3, 0.4, 0.5, 0.7, 1.0 et 1.5 pixels. Le bruit attendu pour une détection subpixelique est de l'ordre de 0.2 pixel tandis qu'une détection classique donnera des résultats comparables à ceux de 1.0 pixel. Aucune fausse correspondance n'a été incluse. Nous avons travaillé avec trois et six images. Les données sont similaires à celles utilisées à la section précédente.

3.6.1.1 Matrices de projection

Nous présenterons tout d'abord les résultats des différentes méthodes de calcul des matrices de projection. Les méthodes présentées sont:

- Méthode à partir des matrices fondamentales (F).
- Méthode à partir des trilinearités normalisées (TN).

- Méthode directe à 2000 essais (D2).
- Méthode directe à 10000 essais (D10).

Nous jugerons de leur qualité sur plusieurs critères:

- La distance entre les matrices de projection calculées et réelles, calculée comme un angle sur la sphère de \mathcal{P}^{11N-15} , après les avoir ramenées à une même base de \mathcal{P}^3 . Cette distance n'a pas d'unité (α).
- La distance entre les points projections des points 3-D estimés et les points projections des points 3-D réels. Cette distance est en pixels (DR).
- La distance entre les points reconstruits une fois appliquée la transformation pour les ramener dans la même base de \mathcal{P}^3 , qui se trouve être une base euclidienne. Cette distance est en millimètres. (DE).

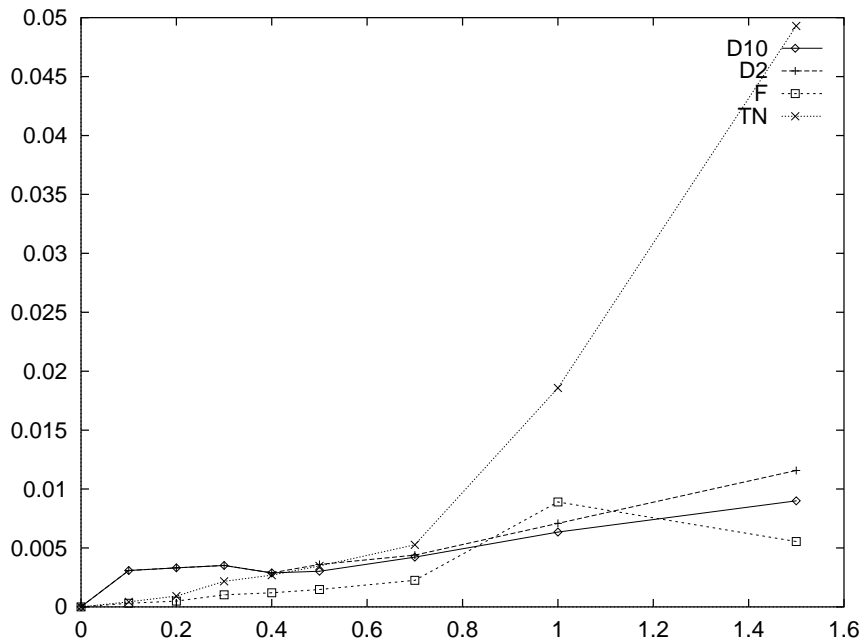


FIG. 3.16 – Comparaison des différentes méthodes pour 3 vues : La mesure d'erreur utilisée est la distance α . Les abscisses représentent le bruit sur les points.

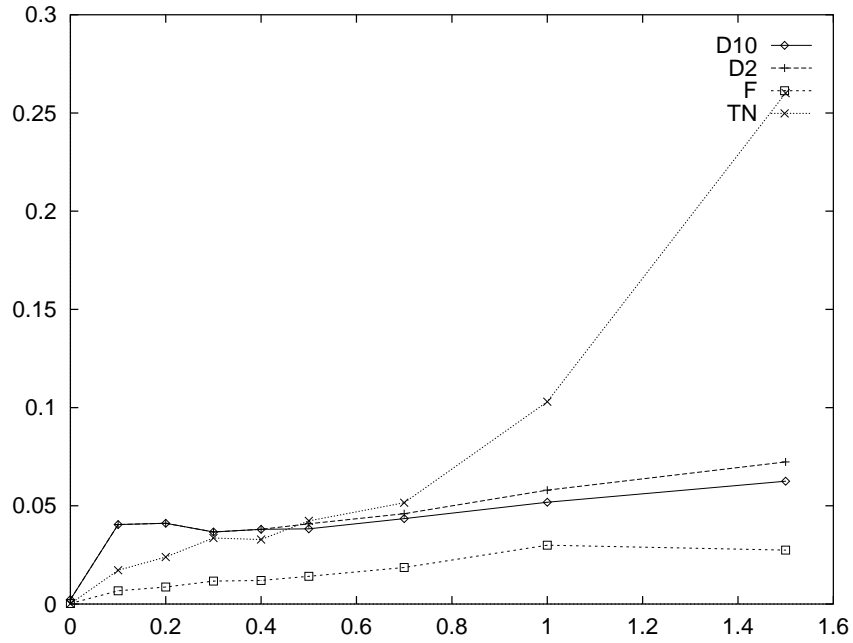


FIG. 3.17 – Comparaison des différentes méthodes pour 3 vues : La distance utilisée est la distance DR. Les abscisses représentent le bruit sur les points.

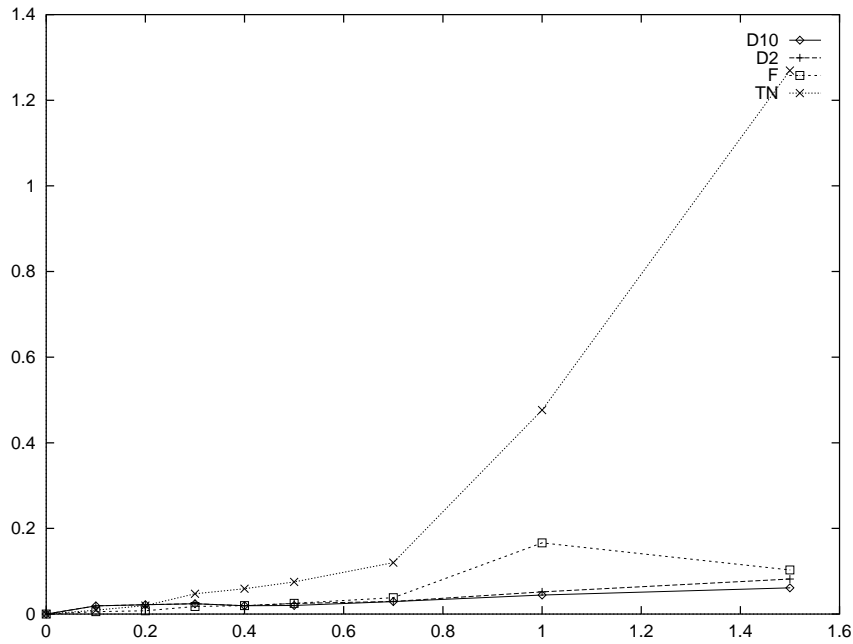


FIG. 3.18 – Comparaison des différentes méthodes pour 3 vues : La distance utilisée est la distance DE. Les abscisses représentent le bruit sur les points.

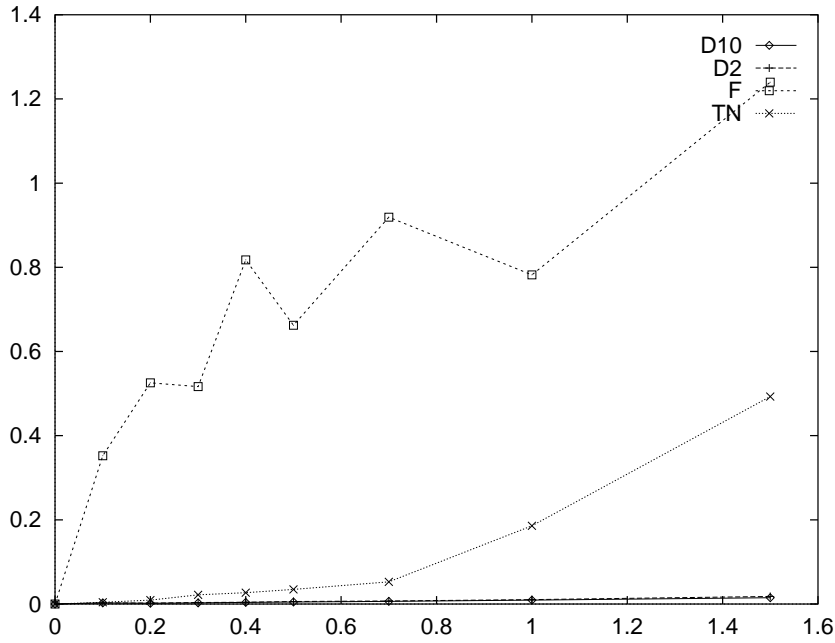


FIG. 3.19 – Comparaison des différentes méthodes pour 6 vues : La distance utilisée est la distance α . Les abscisses représentent le bruit sur les points.

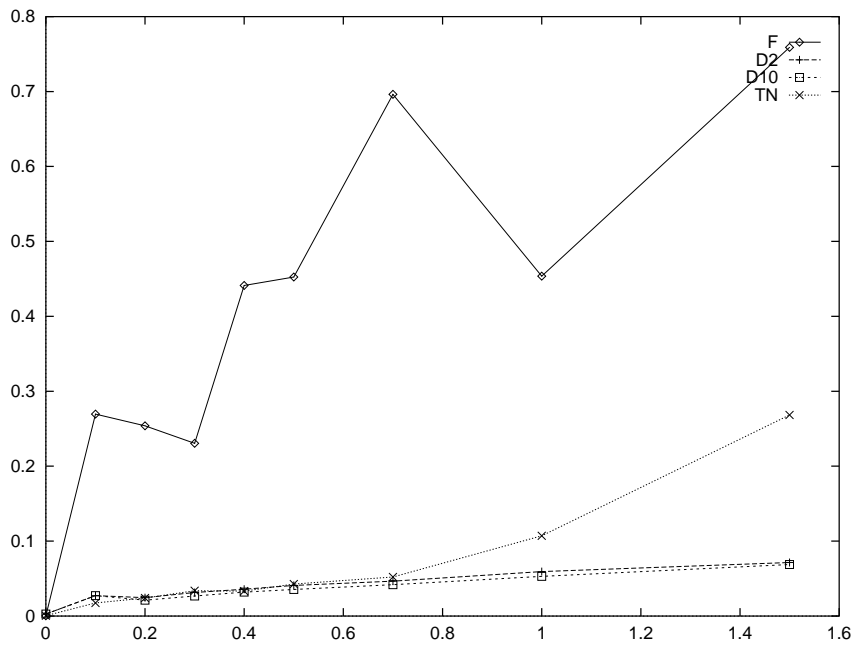


FIG. 3.20 – Comparaison des différentes méthodes pour 6 vues : La distance utilisée est la distance DR . Les abscisses représentent le bruit sur les points.

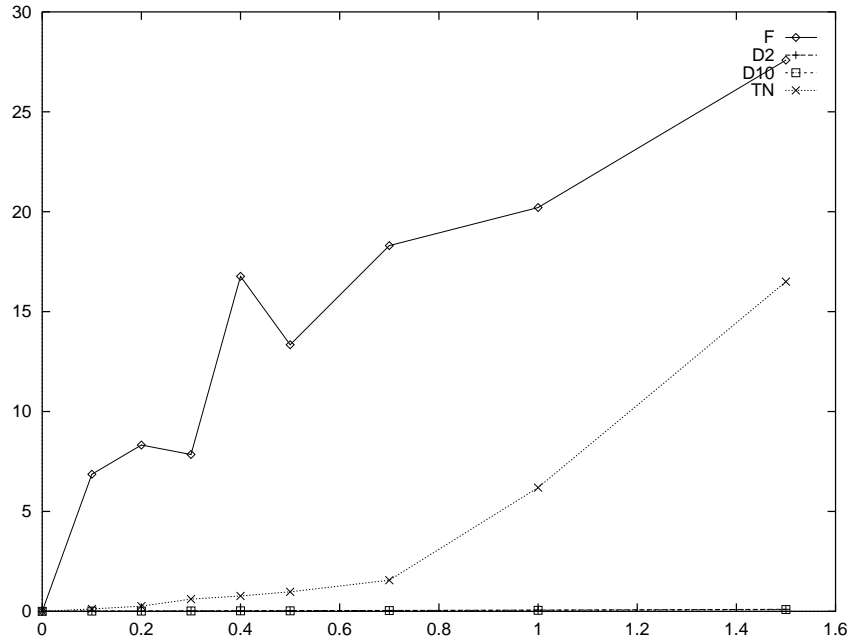


FIG. 3.21 – *Comparaison des différentes méthodes pour 6 vues : La distance utilisée est la distance DE . Les abscisses représentent le bruit sur les points.*

La méthode basée sur la moindre médiane des carrés est la plus fiable comme il apparaît dans les graphes des figures 3.16, 3.17, 3.18, 3.19, 3.20 et 3.21. En fait, elle réalise précisément le but recherché : Obtenir des matrices de projection qui expliquent les images de manière satisfaisante. Les images sont bien expliqués lorsque l'on est capable de trouver de points de l'espace et des matrices de projection dont les points dans les images se déduisent. En théorie, deux jeux de matrices de projection exprimées dans deux bases différentes expliquent la scène de manière identique. En pratique, le conditionnement numérique varie beaucoup suivant le choix de la base. En fait, si l'on regarde les bases choisies par l'algorithme, les points les composant sont en général bien répartis dans l'espace. D'une certaine manière, ils englobent la scène.

La méthode basée sur les matrices fondamentales fonctionne bien (voir les figures 3.16, 3.17 et 3.18) pour un faible nombre de caméras (entre 2 et 4). Elle donne les meilleurs résultats pour un ensemble de trois caméras². Mais ses performances se détériorent rapidement comme on peut le voir entre les figures pour 3 caméras et les figures pour 6 caméras. Ceci est principalement dû à l'initialisation algébrique exposée à la section 2.5.1 qui prend en compte

² pour la configuration que nous avons tiré au hasard.

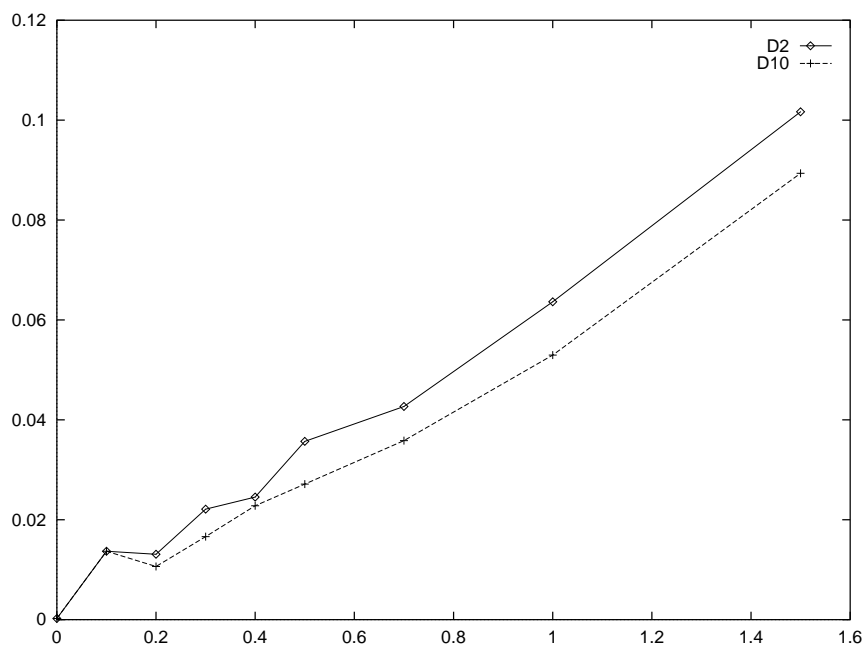


FIG. 3.22 – *Comparaison des méthodes directes pour 6 vues : La distance utilisée est la distance DE. Ce graphe est le même que le précédent, mais l'échelle est différente et seulement les deux meilleures méthodes sont présentées.*

certaines caméras plus que d'autres. Il suffit alors d'une faible perturbation sur des données sensibles pour que l'ensemble de l'estimation devienne erroné. Cet effet est plus sensible lorsque le nombre de caméras augmente. Pour les mêmes raisons, TN conduit à de bons résultats à des faibles niveaux de bruit, mais se détériore avec l'augmentation du bruit. Sa relative stabilité par rapport à la méthode F s'explique simplement : le nombre de caméras pris en compte dans chaque bloc est plus important pour TN (3) que pour F (2). L'initialisation sera donc bonne pour au moins 3 caméras, au lieu de l'être pour seulement 2. En revanche, quand le bruit devient trop important, la projection de T sur l'espace des tenseurs trifocaux ne se passe pas très bien, et le résultat devient rapidement mauvais.

Les méthodes D10 et D2 donnent des résultats similaires à tous les niveaux de bruit comme on peut le voir sur le graphe de la figure 3.22. Les résultats sont même identiques aux faibles bruits parce que la base choisie est la même.

En pratique, D10 est environ 5 fois plus lent que D2, pour une différence de qualité marginale par rapport au temps de calcul. De même, TN est beaucoup plus rapide que D2 et donne des résultats satisfaisants dans la majorité des cas. Ceci n'est bien sûr vrai que si la scène ne comporte pas de faux appariements. Si ce n'est pas le cas (ce n'est jamais le cas dans les scènes réelles), il faudra adapter une moindre médiane des carrés sur TN, ce qui ramènera cette méthode à des temps de calculs similaires.

3.6.1.2 Raffinement

Les distances et les niveaux de bruit sont les mêmes qu'à la section précédente. Nous comparons ici trois résultats :

- Le meilleur résultat sans raffinement (-).
- Avec ajustement de droites épipolaires (E).
- Avec ajustement de rayons (R).
- Avec ajustement de trilinearités (T).

Les critères de comparaison et les niveaux de bruit sont identiques. On voit que l'ordre de grandeur des résultats reste le même quelque soit la méthode de raffinement utilisée. La raison principale est que le résultat initial est relativement bon dans la majorité des cas. La méthode basée sur les trilinearités fonctionne moins bien que les autres, mais ceci est du principalement à sa très lente convergence. Le temps nécessaire à la mise en oeuvre de cette méthode est 15 fois supérieur aux autres méthodes. En effet, le nombre de

fonctions critères à calculer à chaque itération est bien plus important. Cela augmente aussi la taille de la matrice jacobienne à conserver³.

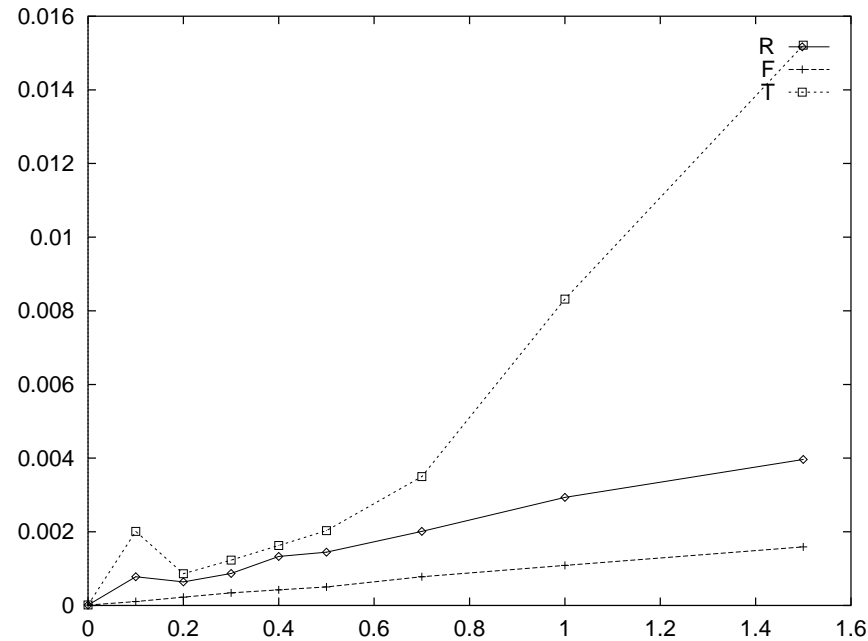


FIG. 3.23 – *Comparaison des méthodes de raffinement pour 6 vues. La distance utilisée ici est la distance α . Les abscisses représentent le bruit sur les points.*

Au regard des figures 3.23, 3.24 et 3.25, la méthode F semble donner de meilleurs résultats que la méthode à base d'ajustement de rayons, et ceci de manière consistante. Ceci n'aurait vraisemblablement pas été le cas si notre configuration avait eu plusieurs centres optiques alignés. Cela n'indique pas tant que la convergence est meilleure dans la méthode F que dans les autres, mais qu'elle converge plus vite. En effet, par souci d'égalité, toutes les méthodes ont été bridées à 2000 itérations.

3.6.2 Séquences réelles

Nous avons ensuite testé nos algorithmes sur deux séquences différentes, la première est la place des Arcades à Valbonne et la deuxième est un bâtiment (la bibliothèque de l'INRIA à Sophia-Antipolis). Aucune donnée exacte

³. Pour plus de détails sur les algorithmes de minimisation utilisés, le lecteur est reporté à l'appendice 3.4.5.1

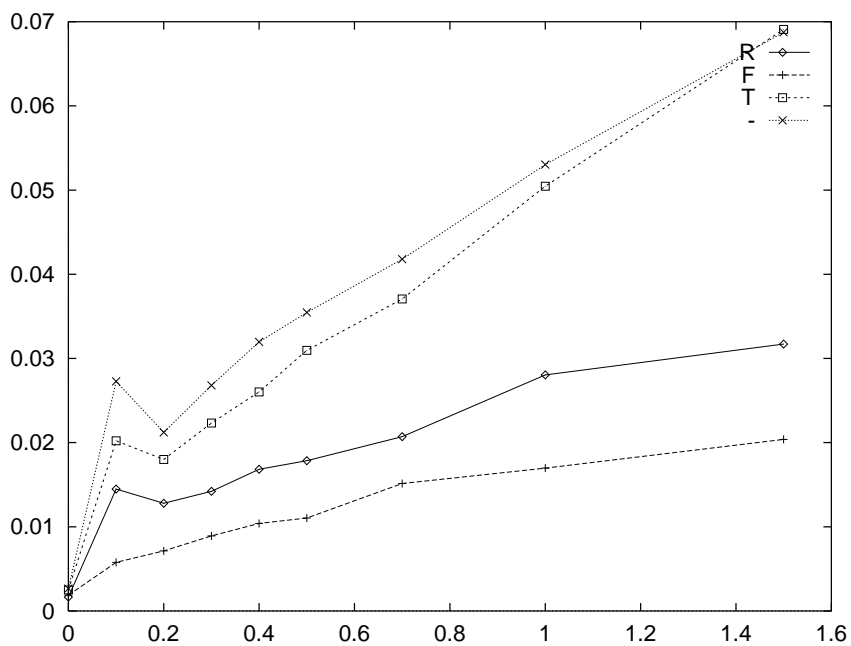


FIG. 3.24 – Comparaison des méthodes de raffinement pour 6 vues. La distance utilisée ici est la distance DR . Les abscisses représentent le bruit sur les points.

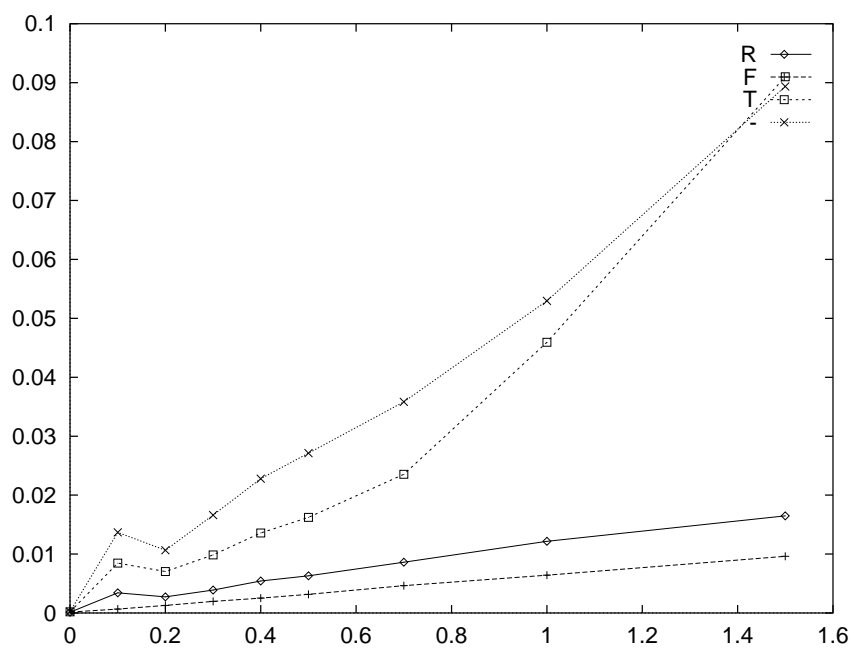


FIG. 3.25 – *Comparaison des méthodes de raffinement pour 6 vues. La distance utilisée ici est la distance DE . Les abscisses représentent le bruit sur les points.*

n'a été fournie avec ces séquences. Il est cependant possible de s'assurer visuellement de la qualité des résultats à l'aide de la géométrie épipolaire.

Les images utilisées pour ces séquences ont été prises avec un appareil photo commercial Canon, puis ont été développées sur PhotoCD. Nous n'avons pas utilisé les images à leur résolution maximale (3072×2048), mais à une résolution intermédiaire : 768×512 pixels. Implicitement, cela veut dire que nous pourrions obtenir des résultats quatre fois plus précis en utilisant des primitives détectées dans les images à pleine résolution. Les images originales sont en couleur, nous avons utilisé seulement l'information de luminance pour extraire nos points d'intérêt et les mettre en correspondance.

3.6.2.1 La place des arcades à Valbonne

La séquence présentée ici est une séquence de 8 images de la place des Arcades à Valbonne. L'estimation de la géométrie a été faite en 3 blocs de 4 images. Nous avons utilisé la méthode directe robuste avec 2000 bases essayées. Les blocs ont été ensuite recollés les uns avec les autres en utilisant les méthodes décrites en 3.5.2.

La figure 3.26 et la figure 3.27 montrent que les droites épipolaires sont bien estimées. La distance entre un point et les droites épipolaires est généralement inférieure à deux pixels. Les résultats obtenus avec cette séquence en utilisant le fait que les paramètres intrinsèques n'ont pas varié [Zeller, 1996] sont légèrement meilleurs, mais du même ordre de grandeur.

La figure 3.28 montre une vue d'ensemble de la scène avec quelques droites épipolaires tracées entre les images.

3.6.2.2 La bibliothèque de l'INRIA

L'estimation de la géométrie a été faite en deux étapes, tout d'abord sur les trois premières images, puis sur les trois dernières. Les deux jeux de matrices de projection ont été rapportés à la base utilisée dans le premier jeu en utilisant les points reconstruits en commun comme à la section 3.5.2. Nous avons utilisé la méthode directe robuste avec 2000 bases essayées.

La figure 3.29 montre que les droites épipolaires sont bien estimées. Ceci est important parce que la géométrie épipolaire est un invariant projectif de la structure de l'ensemble de caméras. L'exactitude de la géométrie épipolaire est une condition nécessaire⁴ de la géométrie projective du système de

4. mais pas forcément suffisante. En effet, si les droites épipolaires sont confondues dans une image, la géométrie épipolaire ne contraint pas le point à se trouver à un pixel donné, mais simplement sur une droite donnée. Sa position exacte sur la droite peut être fautive. La géométrie projective des caméras sera alors fautive, mais la géométrie épipolaire sera



FIG. 3.26 – Détails de la géométrie épipolaire calculée sur la place des Arcades. Trois des huit images sont présentées. Elles appartiennent à trois blocs différents (deuxième, cinquième et septième images)

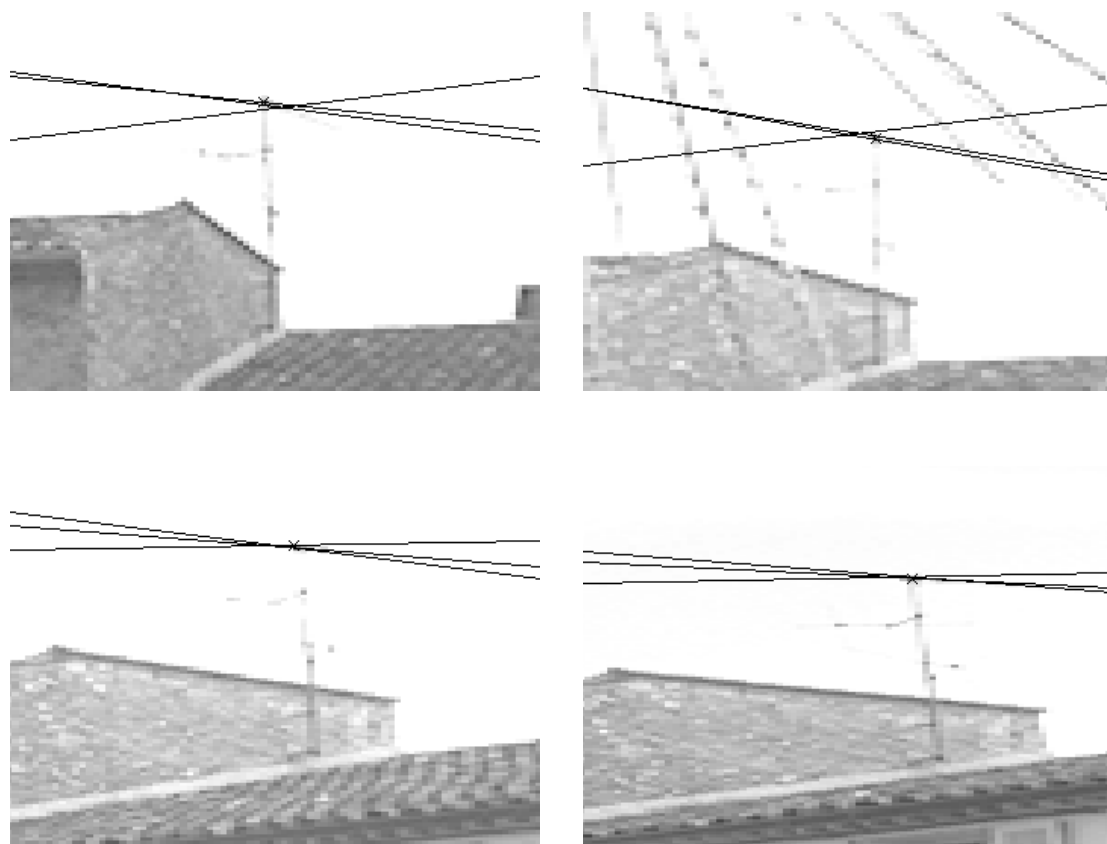


FIG. 3.27 – Détails de la géométrie épipolaire calculée sur la place des Arcades. Quatre des huit images sont présentées. Le détail est le sommet d'une antenne de télévision.

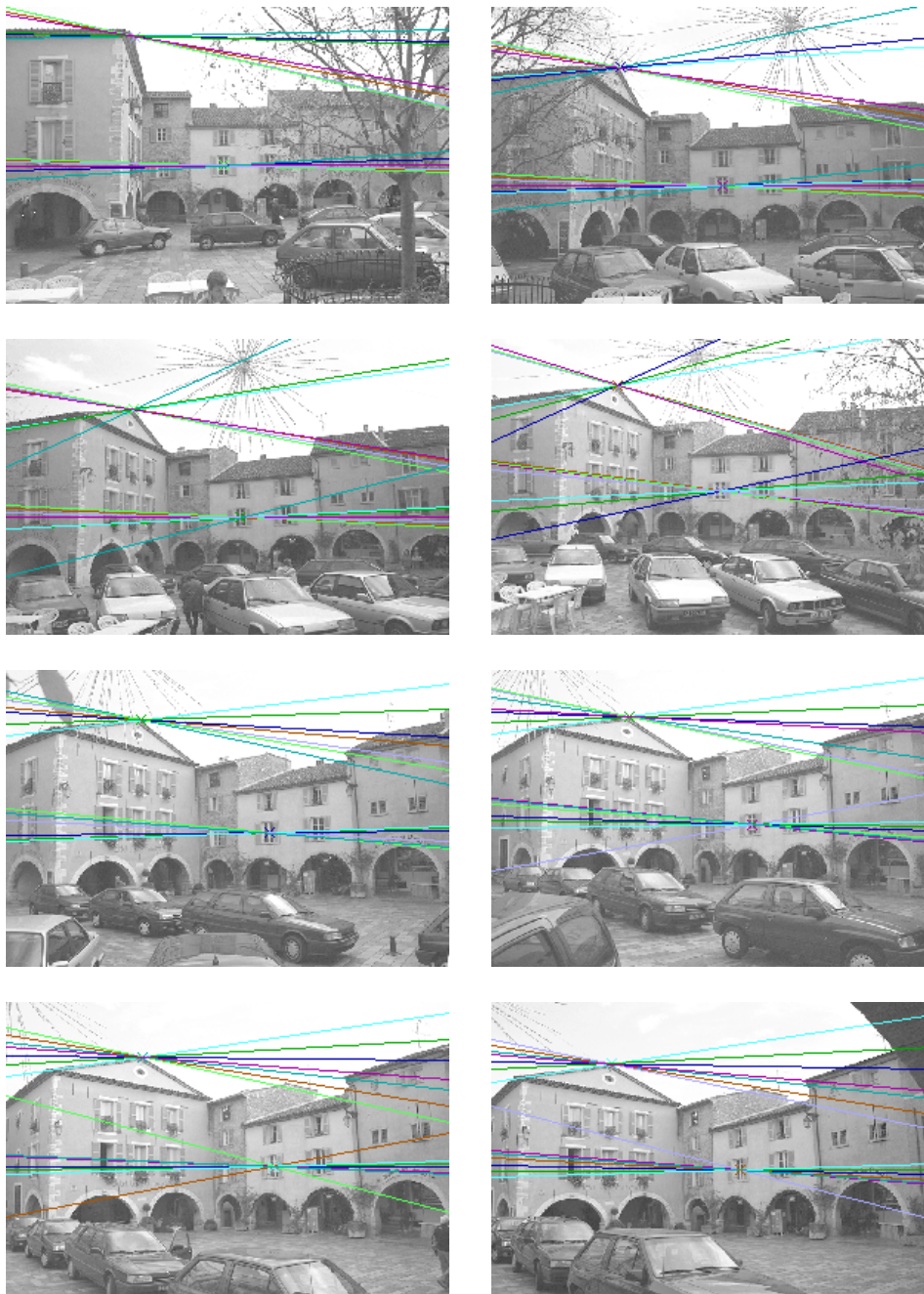


FIG. 3.28 – *Vue d'ensemble de la place. Des droites épipolaires sont tracées en certains points. On remarque que les droites épipolaires se coupent en des points uniques. Ces points sont en correspondance entre les images.*

caméras. La figure 3.30 présente une vue grossie autour d'un point particulier. On remarque que la distance entre le point et les droites épipolaires est inférieure au pixel dans tous les cas.

3.7 Conclusion

Dans ce chapitre, nous avons développé des méthodes d'estimation des objets représentant la géométrie d'un système de caméras, jusqu'au résultat final qui assure une cohérence entre les données, c'est à dire les matrices de projection.

Le point de départ est toujours l'estimation de correspondances entre les images. C'est la partie la plus critique du système. Des bonnes correspondances permettent généralement d'obtenir une bonne géométrie du système. Toutefois, ces bonnes correspondances sont difficiles à obtenir.

L'estimation à base de matrices fondamentales souffre de défauts majeurs : elle ne permet pas l'obtention d'un résultat à coup sûr et elle est très dépendante de la qualité de l'estimation d'une des matrices fondamentales.

Les trilinearités sont très sensibles au bruit. Un bruit faible peut facilement perturber l'estimation de ces dernières, même en prenant toutes les précautions nécessaires pour normaliser les données.

Au contraire, l'estimation directe des matrices de projection est assez robuste et permet l'estimation de la géométrie pour un nombre de caméras plus grand (6-8 au lieu de 3). Cette méthode ne marche absolument pas sans les méthodes statistiques robustes. En effet, les méthodes robustes servent non seulement à éliminer les faux appariements, mais aussi à choisir une paramétrisation des matrices de projections.

Une estimation sur un petit nombre de caméras ne suffit pas forcément pour nos applications. Nous estimons donc alors la géométrie de caméras par sous-ensembles que nous recollons ensuite. Le changement de base est effectué en mettant en correspondance les points. Il est clair que l'algorithme de découpage en sous-ensembles doit dépendre de la scène observée et de la continuité des appariements dans cette scène. Plus de recherches devraient être faites pour déterminer un algorithme automatique.

La géométrie ainsi obtenue est raffinée par des méthodes itératives. Il en existe principalement deux classes, qui minimisent l'une une distance entre points, l'autre une distance entre point et droite. Les deux sont mesurées dans les images.

Les résultats montrent clairement la supériorité des méthodes directes et robustes, même en l'absence de fausses correspondances. Ces méthodes ont

correcte.

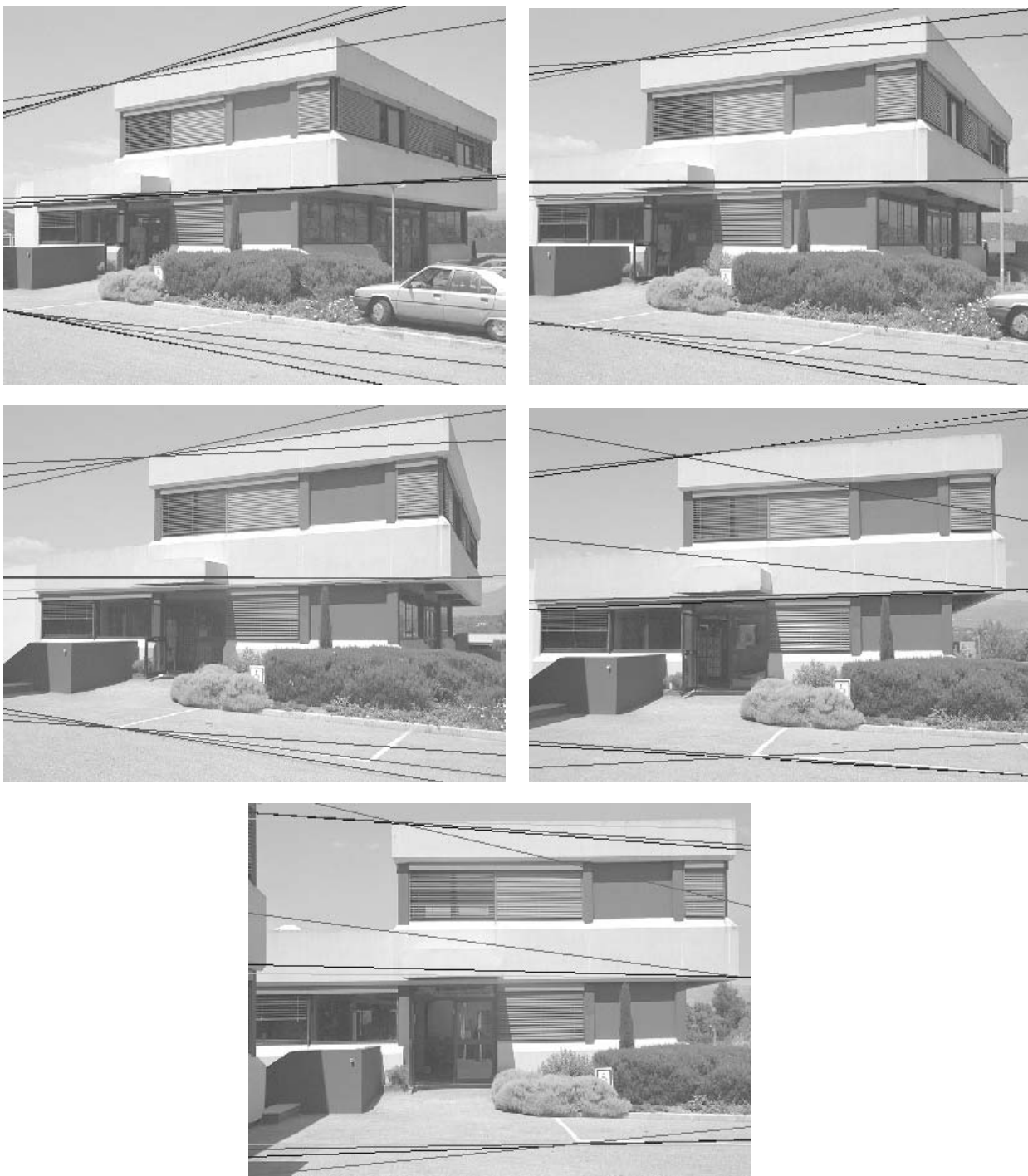


FIG. 3.29 – Cinq images de la bibliothèque de l'INRIA. Des droites épipolaires sont tracées entre les images à certains points.

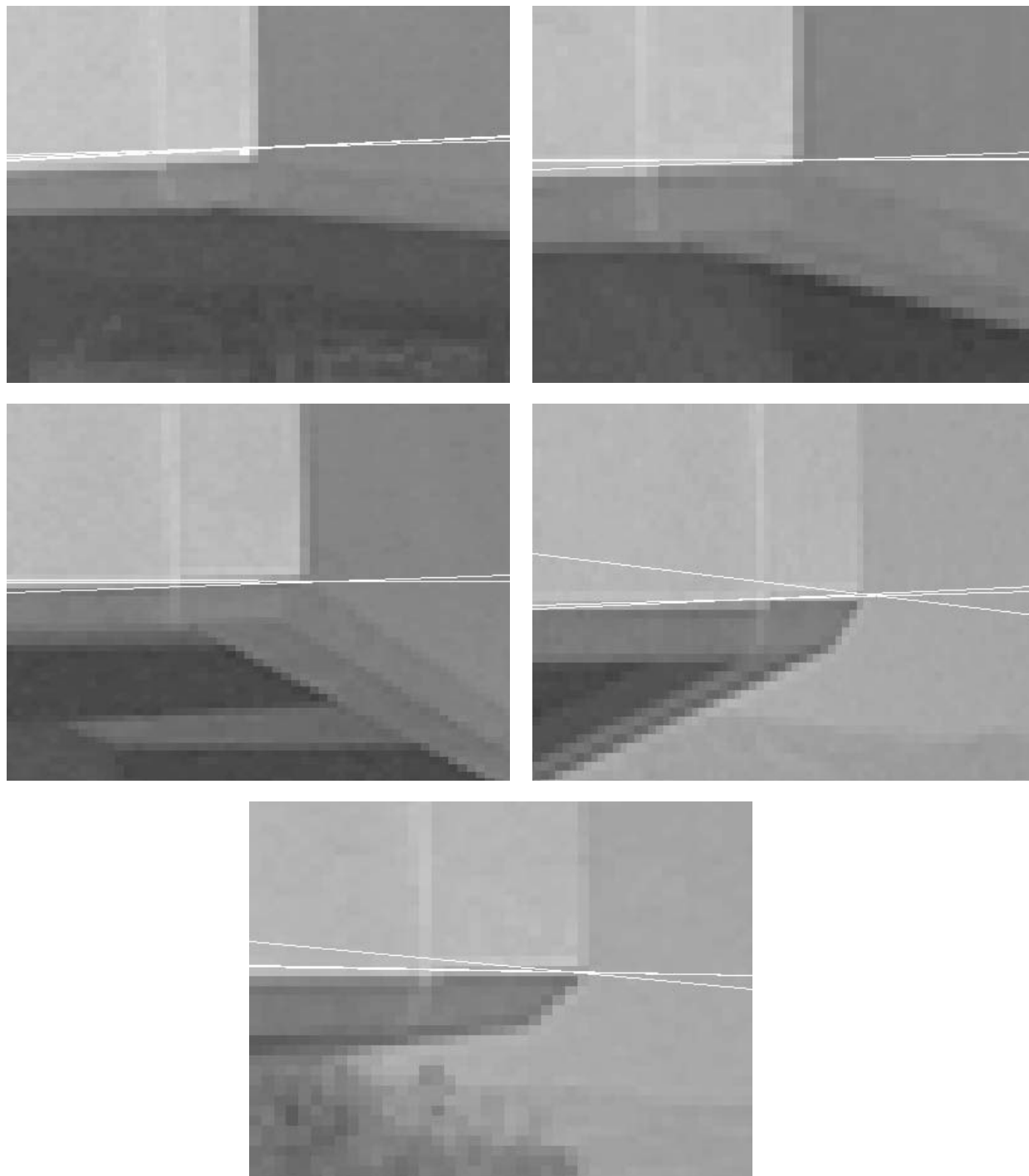


FIG. 3.30 – Détails des droites épipolaires tracées sur l'image autour d'un point caractéristique. Le point est ici le coin inférieur droit de la façade de la bibliothèque.

toutefois un coût exponentiel en le nombre d'images. Il faut donc les associer à des méthodes de recalage entre sous-séquences pour pouvoir les utiliser sur plus de 5 à 10 images.

Pour le raffinement, l'ajustement de faisceaux est la méthode donnant des bons résultats le plus rapidement. De plus, il est facile d'implémenter des extensions de cette méthodes à des cas plus particuliers.

Deuxième partie

Géométrie projective orientée
appliquée à la vision par
ordinateur

Chapitre 4

Géométrie projective orientée

Dans tout ce qui précède, nous avons utilisé le formalisme de la géométrie projective. Ceci nous a permis de dériver de nombreux résultats, mais il reste néanmoins un problème important que nous n'avons pas abordé.

Lorsque nous allons prédire quels points des images sources doivent être utilisés pour calculer l'intensité d'un pixel de l'image synthétisée, il se peut fort que nous n'ayons pas une solution unique. En effet, nous n'avons aucun moyen de discriminer entre les différents points présents sur le rayon. L'ordre dans lequel les points sont vus dans une image est simplement l'ordre réel des points auxquels on a appliqué une transformation projective. C'est à dire que quatre points peuvent se retrouver dans un ordre quelconque.

Ces problèmes d'orientation vont se retrouver lorsque nous allons essayer d'éliminer les faces cachées dans notre synthèse d'image. Ceci est la principale motivation pour établir cette théorie. En effet, il n'est pas possible de savoir quel point est visible en utilisant la géométrie projective non orientée.

Nous allons introduire dans ce chapitre un outil qui va nous permettre de résoudre ces problèmes, la géométrie projective orientée.

La géométrie projective orientée et les problèmes qui y sont reliés sont encore relativement rarement évoqués dans la littérature de vision par ordinateur. Les premiers articles sur le sujet sont de Richard Hartley ([Hartley, 1993a] et d'autres articles non publiés). Hartley introduit le concept d'invariant chiral, et développe une théorie similaire à la nôtre mais sans se placer dans un formalisme mathématique bien défini. Son travail ne concerne que les reconstructions projectives à partir de paires d'images, alors que nous abordons ici le problème pour un nombre d'images quelconque.

4.1 Principes de bases

Nous présenterons dans cette section la géométrie projective orientée en supposant une connaissance approfondie de la géométrie projective classique. Comme Jorge Stolfi dans son livre [Stolfi, 1991], notre exposé sera didactique et s'adresse plutôt aux utilisateurs qu'aux mathématiciens.

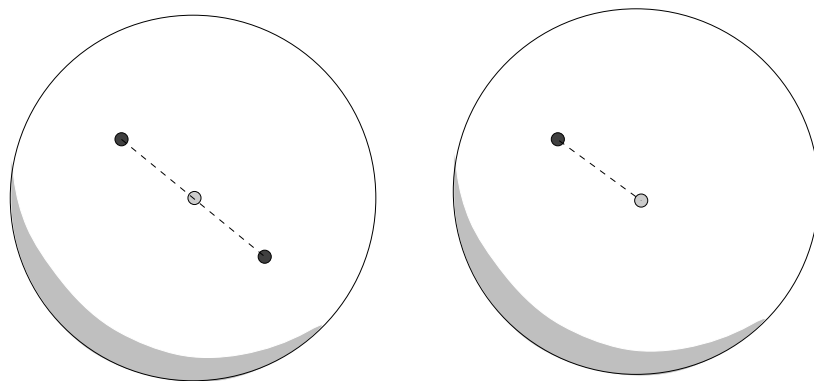
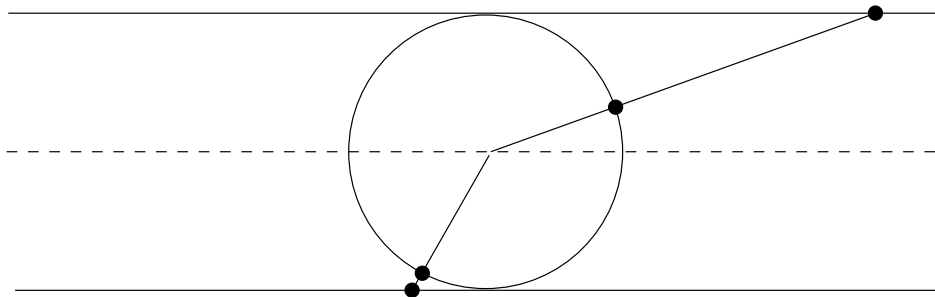
4.1.1 Géométrie projective classique et orientée

Dans la première partie de ce document, nous louions les mérites de la géométrie projective. La géométrie projective classique, malgré toute sa puissance, a un inconvénient majeur qu'il est bon de citer ici. Les plans et les droites ne sont pas orientables. De manière informelle, cela signifie qu'il n'est pas possible de définir le sens des aiguilles d'une montre pour un plan donné. Il n'est pas non plus possible de définir un sens sur une droite, comme par exemple le sens de la propagation de la lumière sur un rayon. De la même manière, la gauche et la droite, le devant et le derrière n'existent pas en géométrie projective classique puisqu'il n'est pas possible de distinguer une face du plan de l'autre, un coté de la droite de l'autre. Il existe toutefois une adaptation de la géométrie projective qui permet d'éviter ces problèmes. La géométrie projective orientée part d'une définition légèrement différente de la géométrie projective classique. La géométrie projective classique considère que deux vecteurs représentent le même point si et seulement si ils sont proportionnels à une constante non nulle près. La géométrie projective orientée restreint cette définition à une constante strictement positive. On remarque tout de suite que cela équivaut à un dédoublement de l'espace. En géométrie classique, le plan \mathcal{P}^2 est représenté par une sphère pour laquelle on a identifié les points antipodaux. En géométrie orientée, \mathcal{T}^2 est représenté simplement par une sphère (voir figure 4.1).

On peut aussi représenter \mathcal{P}^2 par un plan auquel on a adjoint une droite à l'infini. \mathcal{T}^2 est représenté par deux plans auxquels on a adjoint une droite à l'infini commune (voir figure 4.3). Un de ces plans est le devant, l'autre est le derrière. On peut passer facilement de la représentation sphérique à la représentation planaire comme indiqué à la figure 4.2 pour \mathcal{P}^1 .

Une ligne (un objet de \mathcal{T}^1) est en fait la superposition de 2 droites, l'une étant le devant, l'autre le derrière.

Ces principes s'étendent bien sûr à \mathcal{T}^n pour tout n , mais les exemples sont plus difficiles à visualiser pour $n > 3$.

FIG. 4.1 – Représentation sphérique de \mathcal{P}^2 et \mathcal{T}^2 FIG. 4.2 – Passage de la représentation sphérique à la représentation planaire pour \mathcal{P}^1 .

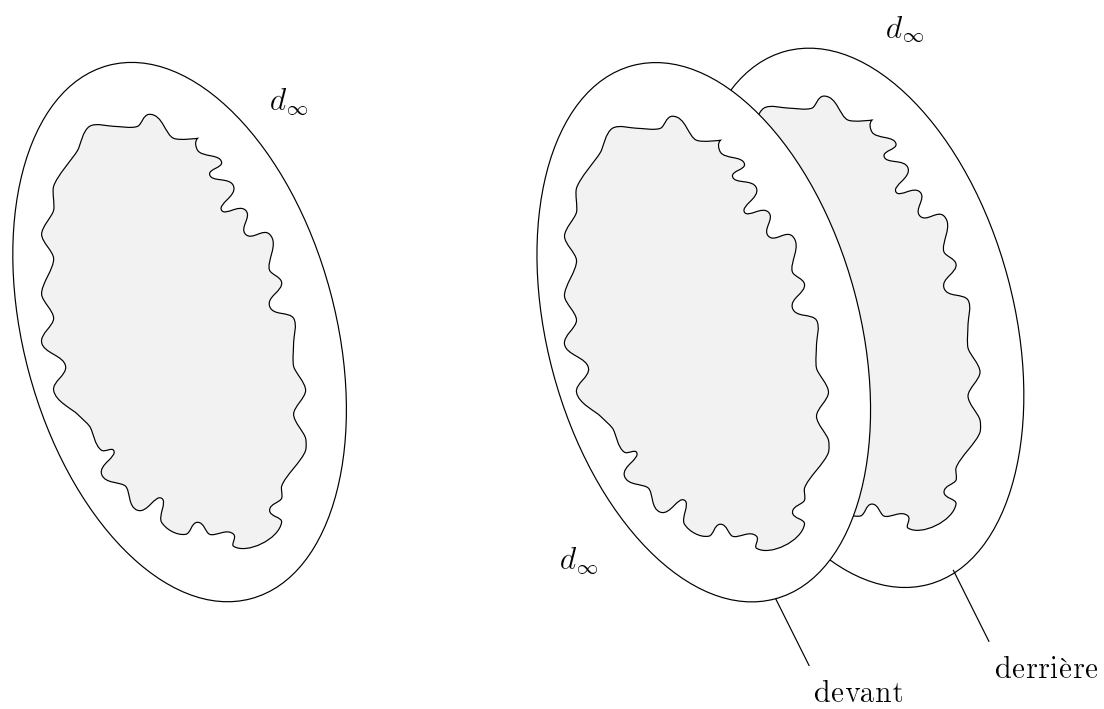


FIG. 4.3 – Représentation planaire de \mathcal{P}^2 et \mathcal{T}^2

4.1.2 Orientation

4.1.2.1 Approche mathématique

Definition:

On appelle simplexe à k points un k -uplet ordonné de points de \mathcal{T}^n . En représentation sphérique, un simplexe s est un $n + 1$ -uplet ordonné de vecteurs unitaires de R^{n+1} . Un simplexe est dit dégénéré si ces vecteurs ne sont pas linéairement indépendants.

La notion de simplexe est bien sûr très reliée à la notion de bases dans un espace vectoriel. On appelle espace engendré par un simplexe s l'intersection de \mathcal{T}^n avec l'espace engendré par les vecteurs de s dans R^{n+1} . La dimension de s est par définition celle de l'espace qu'il engendre. De manière évidente, un simplexe à k points est non dégénéré si et seulement si sa dimension est $k - 1$. Deux simplexes sont dits équivalents si il existe un moyen de passer de l'un à l'autre de manière continue en les déformant et si tous les états intermédiaires sont non dégénérés. Un simplexe à trois points est équivalent à un autre simplexe à trois points si il est possible de déplacer les trois points du premier simplexe sur la sphère jusqu'au deuxième simplexe sans que les trois points ne soient alignés à aucun moment.

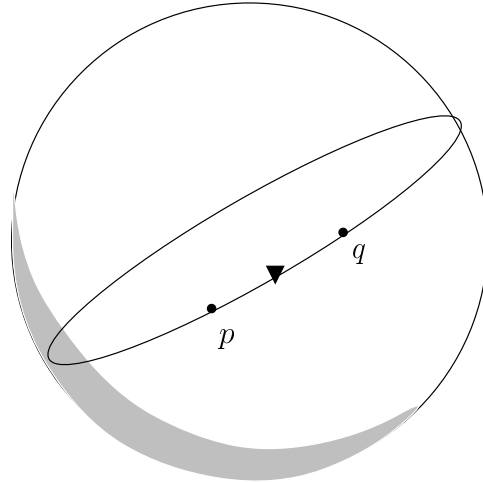
Les simplexes engendrant \mathcal{T}^n se divisent naturellement en deux classes d'équivalence. Nous identifierons ces deux classes avec les deux orientations de \mathcal{T}^n en choisissant un représentant *positif* dans une des classes.

4.1.2.2 Exemples de sous espaces orientés

L'exemple le plus simple est celui des (sous-)espaces de dimension 0. Selon la définition, un sous espace de dimension zéro est une paire ordonnée de points antipodaux sur la sphère. Un sous espace orienté est une paire (a, b) de ce type pour laquelle on a choisi l'un des points avec une orientation positive. On notera $a = -b$. On voit que leur représentants \mathbf{a} et \mathbf{b} vérifient $\mathbf{a} = -\mathbf{b}$.

Un autre exemple est celui des droites dans \mathcal{T}^2 . Deux simplexes (p, q) et (r, s) sont équivalents si et seulement si il est possible de transformer l'un en l'autre sans passer par un simplexe dégénéré. A chaque instant au cours de cette transformation, (p_t, q_t) détermine un ordre circulaire sur la droite $p_t \times q_t$, celui allant de p_t à q_t par le chemin le plus court (voir figure 4.4).

On note que, entre autre, $p \vee q = \neg(q \vee p) = (-p) \vee (-q)$, et que $\neg p \vee q = p \vee (\neg q)$.

FIG. 4.4 – Droite orientée dans \mathcal{T}^2

4.1.3 Propagation de l'orientation

Après les exemples que nous venons de voir, on peut légitimement se poser la question de la propagation de l'orientation au cours des opérations projectives usuelles. L'orientation se propage lorsque l'on joint deux points, la droite étant celle qui relie les deux points par le plus court chemin. Le plus court chemin peut croiser l'hyperplan à l'infini si les deux points sont l'un sur le devant, l'autre sur le derrière. L'orientation se propage aussi lorsque l'on joint un point et une droite. Le plan est alors orienté de manière à ce que la droite tourne autour du point dans le sens positif.

L'orientation se propage aussi lors des opérations d'intersection. Deux plans orientés Π_1 et Π_2 se coupent en une droite orientée l . On détermine l'orientation de cette droite de la manière suivante : on considère le sens de rotation pour passer d'un plan à l'autre par le plus petit chemin et l'on applique la règle de la main droite. Tous les objets propagent leur orientation à travers les opérations projectives.

4.1.4 Droite et gauche, devant et derrière, enveloppes convexes

A partir de ces quelques notions, nous pouvons aisément définir la droite et la gauche d'une ligne. Un point est dit à *gauche* si il est sur le coté gauche lors d'un parcours de la droite dans le sens de son orientation. De même, un point est dit *devant* un plan si le simplexe formé par un simplexe positif du

plan et le point est une base directe de l'espace. On dira qu'il est *derrière* si la base est indirecte.

La définition d'une enveloppe convexe d'un ensemble de points \mathbf{X} est aussi possible. Les algorithmes classiques d'estimation d'enveloppes convexes peuvent s'appliquer puisque nous avons un moyen de décider si deux points sont d'un même côté d'un plan ou non.

4.2 Application à la vision

Les principaux problèmes d'orientation rencontrés en vision stéréoscopique et en structure à partir du mouvement sont reliés à l'orientation de la caméra et à l'orientation d'un point de l'espace par rapport à un point à partir d'informations dans les images.

Nous nous attacherons tout d'abord à l'orientation d'une caméra. Jusqu'ici, aucun algorithme projectif ne pouvait spécifier qu'une caméra ne voyait que le monde devant elle, et non pas le monde derrière elle.

4.2.1 Plan focal orienté

L'orientation d'une caméra se fait en pratique d'une manière assez simple. Il suffit de choisir un point visible dans cette caméra et dont on a les coordonnées spatiales (elles pourront avoir été obtenues par triangulation avec une autre caméra, ou être une donnée, comme dans la calibration). Nous décidons que ce point se trouve *devant* la caméra.

L'existence de telles données (les coordonnées d'un point dans l'espace et dans l'image) est vérifiée dans tous les cas pratiques. Si il s'agit d'une caméra calibrée, les points de la mire de calibration sont devant la caméra, et nous connaissons leurs coordonnées spatiales puisque l'objet est connu. Dans le cas d'une calibration dite faible, la scène n'est pas connue, mais il est toujours possible de reconstruire au moins un point de la scène (vue par au moins deux caméras) dans l'espace¹. Ce point est lui aussi bien sûr devant la caméra. Il est même en fait devant les deux caméras dans lesquelles il apparaît, ce qui nous permet aisément de propager l'orientation sur l'ensemble des caméras du système.

Pour obtenir cette orientation, nous avons juste besoin d'une seule hypothèse : les points de l'espace que nous considérons ne sont pas sur le plan focal, ou bien, de manière équivalente, ils ne se projettent pas sur la droite à

1. La calibration faible, ou calcul de la matrice fondamentale, nécessite un minimum de 7 points.

l'infini dans l'image. Cette condition est en fait toujours vérifiée puisqu'aucune caméra ne voit les points qui sont sur son plan focal.

On suppose fixé le repère dans lequel sont exprimés les points de l'image. Soit \mathbf{M} notre point de référence qui se projette en \mathbf{m} en suivant l'équation $\mathbf{m} = \mathbf{P}\mathbf{M}$. \mathbf{P} peut s'écrire de la manière suivante

$$\mathbf{P} = \begin{pmatrix} \mathbf{I}_1^T \\ \mathbf{I}_2^T \\ \mathbf{I}_3^T \end{pmatrix}$$

\mathbf{I}_3 représente le plan focal. En effet, les points vérifiant $\mathbf{I}_3^T \mathbf{M} = 0$ se projettent à l'infini dans l'image, donc sont dans le plan focal. La dernière coordonnée de \mathbf{m} n'est rien autre que $\mathbf{I}_3^T \mathbf{M}$. Le signe de cette expression nous renseigne sur la position relative de \mathbf{M} et de \mathbf{I}_3 . Suivant nos conventions, $\mathbf{I}_3^T \mathbf{M}$ doit être positif puisque \mathbf{M} est devant le plan focal. Ceci détermine donc le signe de \mathbf{I}_3 et donc le signe de \mathbf{P} . Pour tout point \mathbf{M}' de \mathcal{T}^3 , la dernière coordonnée de \mathbf{m}' nous donne la position de \mathbf{M}' par rapport à la caméra, en prenant \mathbf{M} comme un représentant du devant. Si la dernière coordonnée de \mathbf{m}' est positive, \mathbf{M}' est du même côté de la caméra que \mathbf{M} . Si elle est négative, \mathbf{M}' est de l'autre côté. Si elle est nulle, alors \mathbf{M}' appartient au plan focal.

L'analogie géométrique la plus appropriée ici est celle du double plan pour représenter le sous espace projectif qui correspond à Π_f . La caméra est alors représentée par deux plans (l'un recevant les projections du devant et l'autre du derrière) et une droite représentant les points se projetant à l'infini comme à la figure 4.5. Une autre représentation est la représentation sphérique. Si l'on se place dans un plan contenant C et \mathbf{M} , on obtient le schéma présenté à la figure 4.6.

Le signe de \mathbf{P} détermine donc sans équivoque l'orientation de notre caméra. Une caméra orientée de manière opposée regardera exactement dans l'autre direction, avec les mêmes propriétés quant à la projection. Ces deux caméras générant des images très différentes de la même scène. Une caméra en Corse regardant vers le Sud verrait la Sardaigne (orientation correcte) ou la France et l'Italie (orientation opposée). Il est rassurant de voir qu'elles sont représentées par des objets mathématiques différents.

Il faut cependant être conscient que nous n'avons pas pour l'instant la capacité de connaître a priori l'orientation des points de l'espace \mathcal{T}^3 . Ainsi, à partir de l'expression de \mathbf{C} dans \mathcal{P}^3 , il ne nous est pas possible de décider lequel, de C ou de $\neg C$ est le point qui nous intéresse. Ce choix ne va pas avoir d'incidence sur l'orientation de la caméra mais change l'orientation des rayons optiques. En effet, $(\neg C) \wedge \mathbf{M} = \neg(C \wedge \mathbf{M})$.

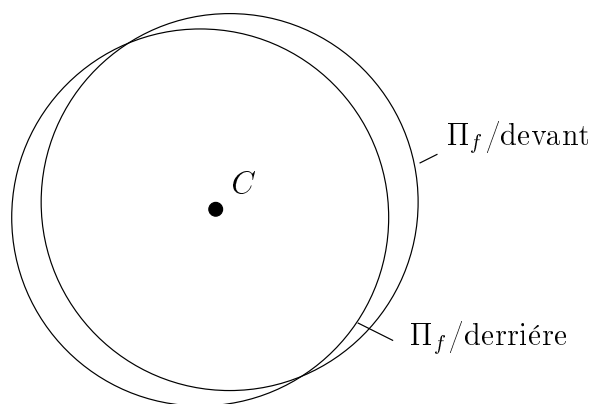


FIG. 4.5 – Une caméra orientée. Elle est représentée par un plan orienté, qui se décompose en deux plans (celui de devant et celui de derrière). Ces deux plans ont une droite en commun, qui représente la ligne à l'infini de ces plans.

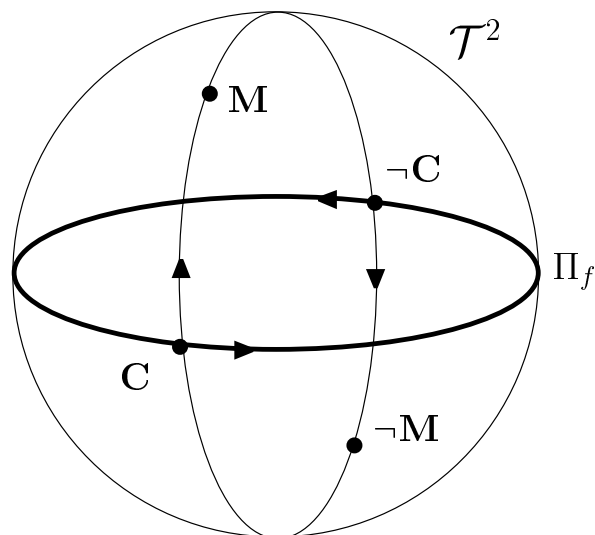


FIG. 4.6 – Une “tranche” de caméra orientée présentée suivant le modèle sphérique. C et $\neg C$ sont les centres optiques possibles. La ligne en gras représente l'intersection avec le plan focal Π_f , la ligne fine le rayon optique dessiné ici au départ de C .

4.2.2 Épipoles et matrices fondamentales orientées

L'orientation de notre matrice de projection n'est pas bien sûr sans conséquence sur les autres objets de notre formalisme. Les épipoles, par exemple, sont définis comme les projections d'autres centres optiques sur notre caméra. Ces épipoles vont donc être orientés comme les autres points. Si le centre optique de l'autre caméra est derrière le plan focal de la première, l'épipole aura une orientation négative (une troisième coordonnée négative). Pratiquement, on dispose d'une expression orientée pour \mathbf{P}_1 et \mathbf{P}_2 . \mathbf{C}_1 et \mathbf{C}_2 se calculent en utilisant la propriété développée à la section 2.1.1 : $\mathbf{P}\mathbf{C} = 0$. Le problème est ici que nous ne disposons d'aucune information sur l'orientation du centre optique. En effet, \mathbf{C}_2 et $-\mathbf{C}_2$ sont deux solutions possibles de $\mathbf{P}_2\mathbf{C}_2 = 0$. La figure 4.7 représente le plan épipolaire de \mathbf{M} en utilisant le modèle sphérique de \mathcal{T}^2 .

On remarque que l'espace est divisé en quatre zones par les deux plans focaux. On note ++ le quartier devant les deux plans focaux, +- le quartier devant la première caméra et derrière la seconde, -+ le quartier derrière la première caméra et devant la seconde et -- le quartier derrière les deux caméras. Pour N caméras, on aura une liste de N fois + et -.

Cependant, contrairement au cas monoculaire où le changement d'orientation du centre optique est sans importance, ici le choix de \mathbf{C}_2 ou $-\mathbf{C}_2$ va changer le signe de leur projections dans les images, c'est à dire les épipoles. L'orientation des épipoles va donc dépendre de ce choix, qui va placer C_2 soit devant soit derrière la première caméra. Si les épipoles sont orientés, les droites épipolaires le sont aussi puisque l'orientation se propage. L'équation de la droite épipolaire engendrée par \mathbf{m}_1 dans l'image 2 est $\mathbf{l}_2^{\mathbf{m}_1} = \mathbf{e}_{21} \times \mathbf{m}_2$ qui hérite des orientations de \mathbf{e}_{21} et de \mathbf{m}_2 . Notons que nous avons ici implicitement choisi le sens de parcours sur la droite épipolaire comme allant de l'épipole vers le point considéré. Cela implique naturellement que le sens de parcours sur le rayon optique est de C_1 vers \mathbf{M} . Le choix de l'autre orientation aurait été possible si les deux termes du calcul de $\mathbf{l}_2^{\mathbf{m}_1}$ avaient été inversés.

Si les droites épipolaires sont orientées, il apparaît normal que les matrices fondamentales le soient aussi. En effet, on a $\mathbf{l}_2^{\mathbf{m}_1} = \mathbf{F}_{12}\mathbf{m}_1$ où \mathbf{m}_1 et $\mathbf{l}_2^{\mathbf{m}_1}$ sont des quantités orientées. Deux points \mathbf{m}_1 différant juste par l'orientation généreront deux droites épipolaires d'orientations opposées.

Le signe de \mathbf{F}_{12} est fixé par le choix de l'orientation d'un épipole, puisque $\mathbf{l}_2^{\mathbf{m}_1} = \mathbf{F}_{12}\mathbf{m}_1 = \mathbf{e}_{21} \times \mathbf{m}_2$. Cependant, nous ne pouvons pas pour autant en déduire le signe de \mathbf{e}_{12} . En effet, nous n'avons plus $\mathbf{F}_{12} = \mathbf{F}_{21}^T$, mais $\mathbf{F}_{12} = \pm\mathbf{F}_{21}^T$. En particulier, si C_2 est devant Π_f^1 et C_1 est derrière Π_f^2 , on a $\mathbf{F}_{12} = -\mathbf{F}_{21}^T$.

Nous ne pouvons pas calculer l'orientation des droites épipolaires à partir

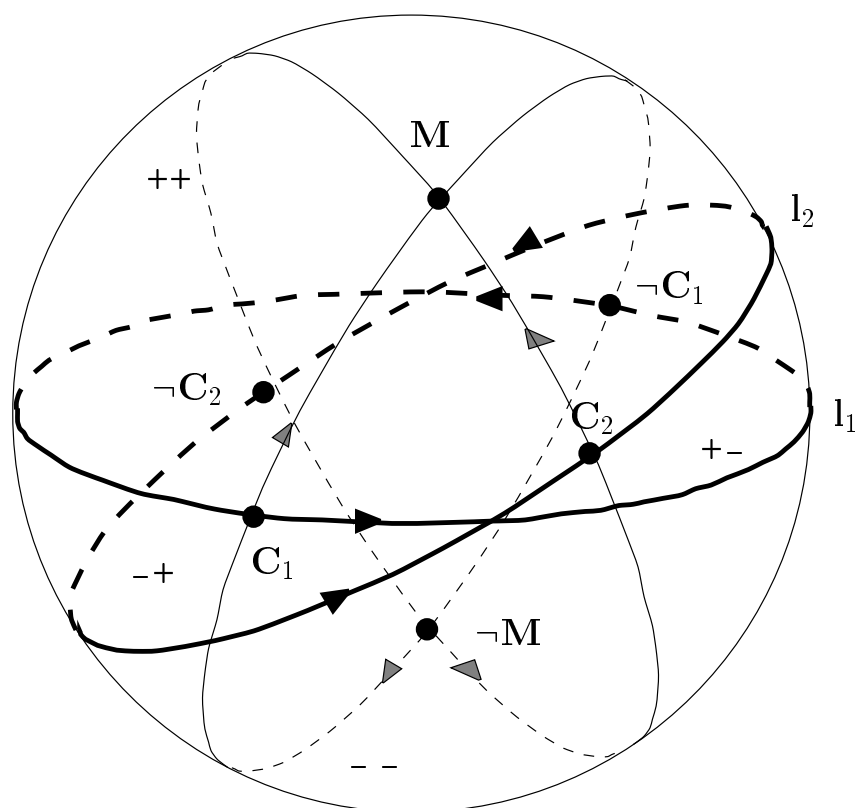


FIG. 4.7 – Le plan épipolaire de M représenté suivant le modèle sphérique. Les lignes l_1 et l_2 représentent les intersections des plans focaux avec le plan épipolaire de M .

de l'orientation des caméras si nous nous limitons aux hypothèses précédentes (connaissance d'un point devant les caméras). Par contre, si nous connaissons l'ordre de deux points sur une épipolaire, nous pouvons déterminer son orientation sans avoir à la choisir arbitrairement. L'orientation de la droite épipolaire $\mathbf{I}_2^{\mathbf{m}_1}$ et celle du point \mathbf{m}_1 étant connu, on peut en déduire l'orientation de \mathbf{F}_{12} . La connaissance de l'orientation de $\mathbf{I}_1^{\mathbf{m}_2}$ nous donne l'orientation de \mathbf{F}_{21} .

Le problème du choix est aussi résolu si nous travaillons dans un espace affine, puisque \mathbf{C}_1 et \mathbf{C}_2 étant des points réels, ils sont contraints à être devant le plan à l'infini, ce qui élimine donc $\neg\mathbf{C}_1$ et $\neg\mathbf{C}_2$. En effet, un seul point parmi une paire de points antipodaux peut se trouver devant le plan à l'infini (en fait devant n'importe quel plan). On connaît alors l'orientation des points représentant les centres optiques. L'orientation de \mathbf{C}_1 et \mathbf{C}_2 étant déterminée, l'orientation des épipoles et de la matrice fondamentale se déduit des formules ci-dessus. En effet, $\mathbf{e}_{21} = \mathbf{P}_2\mathbf{C}_1$ par exemple, ce qui nous permet d'obtenir le signe de \mathbf{e}_{21} puisque nous connaissons l'orientation des deux autres entités.

4.2.3 Enveloppes convexes, deux approches

Une autre propriété agréable de la géométrie projective orientée est d'autoriser le positionnement d'un point par rapport à un plan ou à une droite, autorisant ainsi la construction d'enveloppes convexes dans le plan ou dans l'espace. Ces enveloppes convexes peuvent être utilisées en vision pour obtenir une description géométrique plus simple d'un objet pour un algorithme de reconnaissance utilisant des invariants affines ou projectifs, ou bien pour évaluer l'espace libre en face d'un robot.

Une méthode pour calculer l'enveloppe convexe d'un objet a été proposé par Luc Robert [Robert et Faugeras, 1995]. Il est malheureusement clair, comme le notent d'ailleurs les auteurs, que ses calculs et interprétations géométriques s'effondrent si les points ne sont pas tous dans le quart d'espace devant les deux caméras ou tous dans le quart d'espace derrière les deux caméras. Son approche est basée sur les homographies qui existent entre les points d'un plan vus dans deux caméras.

Notre approche est plus analytique et plus classique d'une certaine manière. Un plan dans l'espace est défini par les reconstructions de trois points qui sont vus dans les images. Ces points de l'espace n'étant pas orientés, le plan ne l'est pas non plus. Par contre, il est extrêmement facile de dire si deux points \mathbf{M} et \mathbf{M}' , dont on sait qu'ils sont visibles, sont du même côté de ce plan Π en comparant le signe des deux produits scalaires du vecteur représentant Π avec \mathbf{M} et \mathbf{M}' orientés de telle manière qu'ils appartiennent à la zone ++.

Les mêmes algorithmes s'appliquent alors : il s'agit par exemple de l'algorithme du paquet cadeau [Preparata et Shamos, 1985]. Les résultats sont bien sûr identiques. Toutefois, notre algorithme peut s'appliquer dans les cas où la scène n'est pas entièrement devant les caméras, ce qui arrive fréquemment si l'on se déplace à l'intérieur une scène.

4.3 Implications

Dans cette section, nous montrons deux cas où la géométrie projective orientée permet une meilleure résolution de deux problèmes pratiques en vision par ordinateur. Dans la première partie, nous apprendrons à faire la différence entre une reconstruction 3-D à partir de la stéréovision possible et impossible. Dans la deuxième partie, nous appliquerons la géométrie projective orientée à la détermination du sens de parcours sur les épipolaires.

4.3.1 Reconstructions possibles et impossibles

4.3.1.1 Théorie

Dans cette section, nous ne mentionnerons que le cas de deux caméras pour garder un discours clair et concis. L'extension de 2 à N caméras est immédiate.

Nous avons vu dans les sections précédentes que nous pouvions déterminer si un point d'une scène est devant ou derrière la caméra une fois que l'on a choisi la position d'un point de référence. Il est clair que pour un système de stéréovision, tous les points reconstruits doivent être devant toutes les caméras. Ceci va nous permettre d'éliminer des faux appariements qui donneraient lieu à des points impossibles, la scène traversant un des deux plans focaux.

Les plans focaux des deux caméras divisent l'espace en quatre zones comme montré à la figure 4.8 et à la figure 4.7. La reconstruction doit être dans une et une seule zone. Les plans focaux doivent être considérés comme des murs infranchissables. En effet, la scène ne peut pas se situer de part et d'autre d'un plan focal. Les points doivent donc tous être reconstruits dans une seule zone.

Nous n'essayons pas ici de démontrer que nous allons pouvoir nous débarrasser de tous les faux appariements, mais simplement que cette étape peu coûteuse (une reconstruction projective) va améliorer les résultats à peu de frais.

La méthode est extrêmement simple. Fixons nous un devant et un derrière de référence pour chaque caméra, arbitrairement si besoin est. Nous

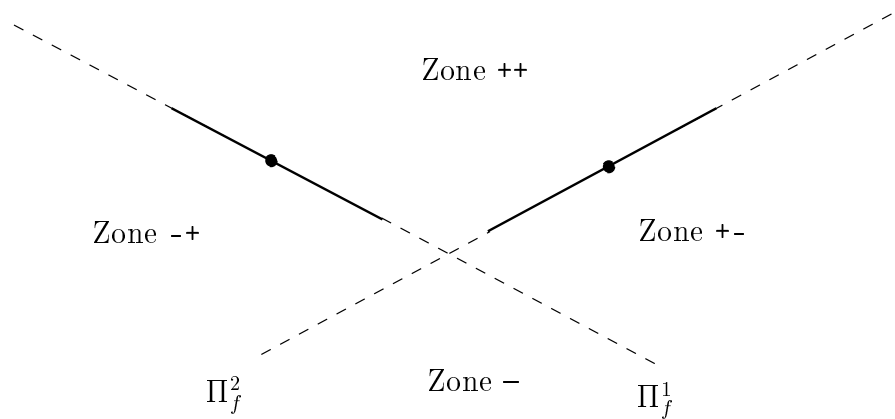


FIG. 4.8 – Division de \mathcal{T}^3 en 4 zones : devant et derrière les plans focaux.

pouvons alors obtenir à partir des correspondances deux matrices de projection orientées représentant nos deux caméras. La reconstruction des appariements dans l'espace nous donne un nuage de points de \mathcal{P}^3 . Ces points de \mathcal{P}^3 correspondent à des paires de points antipodaux $(\mathbf{M}_+, \mathbf{M}_-)$ de \mathcal{T}^3 . Nous ne savons pas pour l'instant lequel de ces points correspond à \mathbf{M} et lequel correspond à $\neq \mathbf{M}$. Un seul de ces points se reconstruit devant Π_f^1 . Notons ce point \mathbf{M}_+ . \mathbf{M}_+ peut être devant ou derrière Π_f^2 . Si il est derrière, il est alors considéré comme faux puisque \mathbf{M}_+ et \mathbf{M}_- appartiennent à des zones impossibles, c'est à dire des zones où les points visibles ne peuvent pas se trouver. Si \mathbf{M}_+ se trouve devant Π_f^2 , alors il peut être juste ou bien faux. En effet, nous ne pouvons pas distinguer entre un point qui se trouve dans la zone $-$ et un point qui se trouve dans la zone $++$. Dans notre cas, le point effectivement reconstruit pourrait être \mathbf{M}_- et il n'y a pas de moyen de décider lequel de \mathbf{M}_+ ou de \mathbf{M}_- est le point physique. Encore une fois, cette ambiguïté est levée si nous connaissons la structure affine de la scène. Les points sont physiques et doivent donc se trouver devant le plan à l'infini. Nous pouvons alors immédiatement décider lequel de \mathbf{M}_+ et \mathbf{M}_- est le point physique \mathbf{M} .

Nous venons de démontrer que nous pouvons éliminer deux zones en géométrie projective orientée, mais trois zones si nous connaissons le plan à l'infini. La situation s'améliore avec le nombre de caméras, car les zones possibles sont toujours $++ \dots +$ et $-- \dots -$ tandis que le nombre de zones augmente en 2^N , N étant le nombre de caméras. Cette situation est décrite à la figure 4.9.

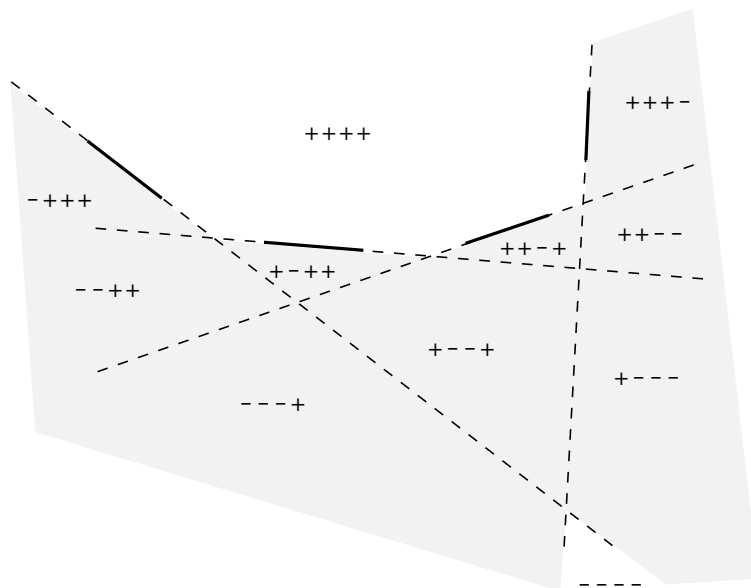


FIG. 4.9 – Évolution des zones avec le nombre de caméras. L'exemple présenté ici est avec 4 caméras.

4.3.1.2 Algorithme

Notre algorithme de rejet est simple : Une des zones est majoritaire puisqu'elle correspond à la scène réelle (Nous faisons ici l'hypothèse implicite que les mises en correspondance correctes sont plus nombreuses que les mises en correspondance incorrectes, ce qui est heureusement le cas courant). Les points reconstruits dans les trois autres zones sont alors marqués (temporairement) comme incorrects. À partir des points marqués corrects, on peut recalculer une matrice fondamentale et réestimer les matrices de projection. On reconstruit alors tous les points (corrects et incorrects) pour pouvoir de nouveau les séparer en quatre zones.

Les deux images ci-dessous ont été prises avec une caméra CCD conventionnelle. La géométrie épipolaire a été calculée à partir de correspondances de points. Malgré les méthodes de relaxation et de rejet de faux appariements ([Torr et Murray, 1993, Torr et Murray, 1992, Torr, 1995] [Zhang et al., 1994]), il reste encore quelques mauvais appariements. Ces appariements vérifient la géométrie épipolaire, ont un profil d'intensité semblable. En utilisant l'orientation, nous avons pu détecter deux faux appariements numérotés 41 et 251 dans les images (figure 4.10). Ces points satisfont les contraintes épipolaires. Ceci peut ne paraître pas beaucoup, mais c'est principalement parce que la majorité des autres appariements incorrects ont été éliminés aux étapes pré-

cédentes. Cette méthode est à utiliser en complément des autres (relaxation, méthodes robustes).

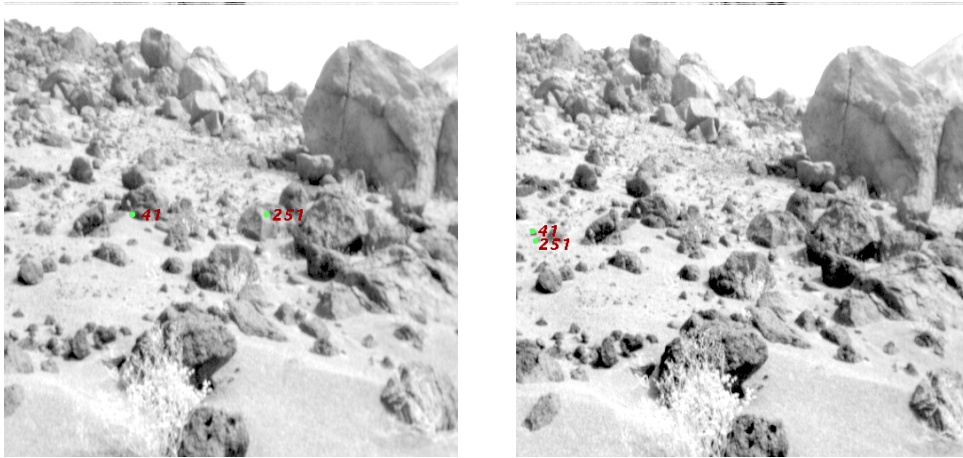


FIG. 4.10 – *Faux appariements détectés grâce à l'orientation. Le point correspondant dans l'espace est reconstruit derrière les plans focaux.*

Cette méthode peut être étendue à tout plan qui ne traverse pas la scène et dont l'on connaît l'équation. Par exemple, on désire, pour certaines applications que la scène ne soit ni trop loin ni trop près. La reconstruction possible va alors se situer entre deux plans parallèles² aux plans focaux. C'est ce que fait Francis Lustman dans [Lustman, 1988]. Il rajoute une contrainte à la mise en correspondance qu'il appelle contrainte de disparité. Cette contrainte n'est que la spécification de deux plans entre lesquels la scène doit se trouver.

Cette contrainte a été développée par Cyrille Zeller [Zeller, 1996]. Dans ces travaux, les images sont supposées être rectifiées, c'est à dire que les épipoles sont à l'infini dans la direction horizontale. On peut alors définir la disparité d comme la différence de position sur la droite épipolaire de deux points en correspondance. La contrainte $d_{min} \leq d \leq d_{max}$ force les points en correspondance à se trouver entre deux plans parallèles au plan de rectification. Une étude complète sur la rectification peut être trouvée dans [Robert, 1993, Zeller, 1996].

On peut généraliser cette approche à des images non rectifiées en utilisant la géométrie projective orientée. En effet, une fois notre géométrie épipolaire

2. Attention. Une telle notion n'existe pas en géométrie projective. Des contraintes supplémentaires sont nécessaires pour pouvoir exprimer cette propriété. Dire que deux plans sont parallèles est équivalent à dire qu'ils se coupent en une droite qui appartient au plan à l'infini, qui n'est défini qu'une fois la structure affine connue.

orientée, nous pouvons ordonner les points sur le rayon optique. On note \mathbf{H}_i et \mathbf{H}'_j les homographies associées aux plans Π_i et Π'_j entre lesquels les points doivent se trouver. Les points devront se trouver derrière les plans Π_i et devant les plans Π'_j . Soit un point \mathbf{M} de l'espace qui est visible dans les caméras en \mathbf{m}_1 et \mathbf{m}_2 .

Regardons la droite épipolaire $\mathbf{F}_{12}\mathbf{m}_1$ de \mathbf{m}_1 dans l'image 2. Sur cette droite, on a le point \mathbf{m}_2 , mais aussi tous les points $\mathbf{H}_i\mathbf{m}_1$ et $\mathbf{H}'_j\mathbf{m}_1$. Cette droite est orientée puisque l'orientation de la matrice fondamentale est connue. Si \mathbf{M} est entre les plans Π_i et Π'_j , alors, en parcourant depuis l'épipole \mathbf{e}_{21} la droite épipolaire, on va d'abord trouver tous les points $\mathbf{H}_i\mathbf{m}_1$ puis \mathbf{m}_2 puis tous les points $\mathbf{H}'_j\mathbf{m}_1$.

Si les images sont rectifiées et si les plans Π_i et Π'_j sont parallèles au plan de rectification alors les homographies sont simplement des translations de direction horizontale. On retrouve alors la contrainte $d_{min} \leq d \leq d_{max}$.

4.3.2 Sens de parcours des épipolaires

Le problème est le suivant : Étant donné deux points d'une scène qui sont représentés par un seul point dans une vue, on veut savoir de quel point il s'agit en s'aidant d'une deuxième vue (voir figure 4.11). Ce problème est central en transfert de vues et en compression d'images [Faugeras et Laveau, 1994, Laveau et Faugeras, 1994, McMillan et Bishop, 1995] pour la détermination des surfaces cachées.

Il n'est pas possible d'identifier le point en utilisant la géométrie épipolaire simple, puisque les deux points appartiennent à la même droite épipolaire. Il faut identifier le point qui est le plus près du centre optique sur le rayon. Il s'agit bien sûr du premier point atteint par le rayon lorsqu'il est parcouru du centre optique vers l'infini dans le demi-espace devant la caméra. Il faut se défaire de l'impression fautive qu'il s'agit toujours du point le plus près de l'épipole. En effet, la situation s'inverse dès que le centre optique de la deuxième caméra franchit le plan focal de la première comme le lecteur peut s'en convaincre à l'aide de la figure 4.12.

Il est suffisant de faire intervenir la géométrie projective orientée pour résoudre ce problème de manière simple et élégante. Nous avons vu à la section précédente que tous les points de \mathcal{T}^3 se projetaient sur les images avec un signe qui dépendait de leur position par rapport au plan focal. Soit C_2 le centre optique de la caméra 2 où les deux points de \mathcal{T}^3 apparaissent comme confondus. C_2 se projette sur la caméra 1 en \mathbf{e}_{12} . Le signe de \mathbf{e}_{12} nous renseigne sur la position relative de C_2 et Π_f^1 , mais cette information n'est pas celle que nous allons essayer d'extraire directement. Considérons en effet le segment orienté $\mathbf{e}_{12} \times \mathbf{m}_1^a$ où \mathbf{m}_1^a est l'un des points ambigus. Le segment

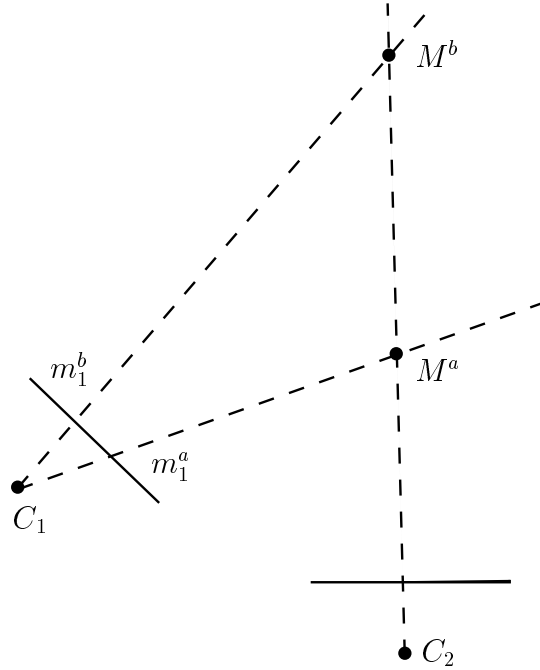


FIG. 4.11 – Deux points de l'espace confondus dans une des caméras.

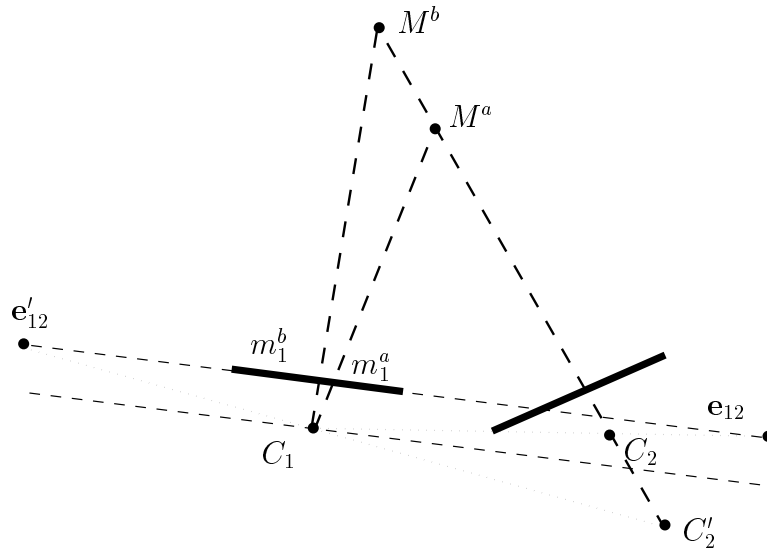


FIG. 4.12 – Changement d'orientation lorsque le centre optique de la deuxième caméra traverse le plan focal de la première caméra. Il est noté C_2 et C'_2 suivant la configuration. L'épipole correspondant est noté e_{12} et e'_{12} .

est orienté de la même manière que le rayon qui lui est associé dans l'espace, c'est à dire partant de C_2 et se dirigeant vers la scène et l'infini. Il suffit de suivre le segment orienté à partir de \mathbf{e}_{12} pour trouver le premier point qui est visible à partir de C_2 dans la caméra 2.

Dans le cas de la figure 4.12, il nous faut tout d'abord connaître l'orientation de la géométrie épipolaire. Elle ne sera pas la même dans les deux cas. En effet, \mathbf{e}'_{12} aura une orientation négative tandis que \mathbf{e}_{12} aura une orientation positive. Pour connaître l'orientation de la géométrie épipolaire, il suffit de connaître l'orientation d'une seule droite épipolaire.

Que se passe-t-il lorsque l'orientation n'est pas respectée? Pour mieux le comprendre, la figure 4.13 montre deux fois la même vue, synthétisée une fois avec l'orientation correcte, une fois avec une orientation erronée. La vue synthétisée avec l'orientation erronée apparaît comme étant vue "de l'autre côté". L'interprétation géométrique est simple : la caméra inversée regarde de l'autre côté, mais suffisamment loin pour passer à travers l'infini et revenir de l'autre côté de l'objet (voir figure 4.14).

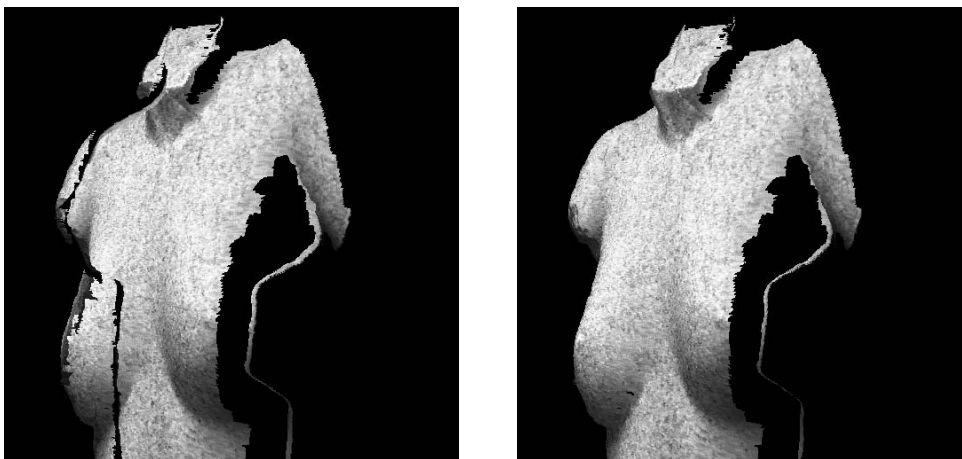


FIG. 4.13 – 2 images synthétisées avec des caméras différant uniquement par l'orientation. L'image de gauche présente des anomalies sur le côté qui sont dues au fait que le sein droit du mannequin est vu par derrière. L'objet n'est pas complet parce que les images sources ne le couvrent pas entièrement.

Ceci nous montre qu'il n'y a pas un choix possible pour chaque ligne épipolaire ambiguë, mais un seul choix global entre deux orientations possibles. Il suffit de choisir l'orientation d'une ligne épipolaire dans une image pour fixer l'orientation de toutes les lignes épipolaires dans cette image, et donc, de toute l'image synthétisée. Un programme peut facilement créer les deux

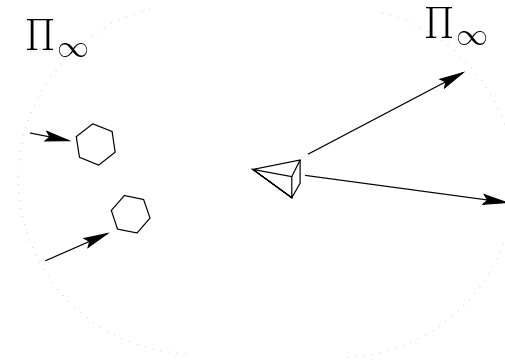


FIG. 4.14 – La caméra qui regardait à travers l’infini. Les rayons traversent le plan à l’infini pour se retrouver de l’autre côté de la caméra.

images, puis l’utilisateur sélectionne celle qui est plausible visuellement. On peut aussi procéder par continuité, si le centre optique de la nouvelle caméra se déplace lentement, et conserver la même orientation tant que le centre optique de la nouvelle caméra ne traverse pas un plan focal des caméras source, ce qui peut être détecté facilement.

4.4 Liens avec la chiralité de Richard Hartley

Richard Hartley a développé une théorie parallèle à la nôtre en utilisant un formalisme différent. Ses résultats sont les mêmes, mais les calculs sont différents. En particulier, il n’utilise pas la géométrie projective orientée mais une géométrie euclidienne plongée dans un espace projectif non orienté. Dans son formalisme, toutes les transformations projectives ne sont pas autorisés. En particulier, la multiplication par un nombre négatif est évitée tout au long du document en considérant que tout les points qui ne sont pas à l’infini ont leur dernière coordonnée projective égale à un.

Implicitement, ceci signifie que nous travaillons toujours dans \mathcal{T}^{3+} , le devant de \mathcal{T}^3 . Bien sûr, cela est rendu difficile par le fait que nous ne connaissons pas l’équation du plan à l’infini, donc nous ne savons pas identifier \mathcal{T}^{3+} .

Au cours de cette section, nous ferons toujours référence aux notations de [Hartley, 1996] et nous les comparerons à nos démonstrations des sections précédentes.

4.4.1 χ détermine si un point est devant ou derrière une caméra

Dans [Hartley, 1996], $\chi(\mathbf{x}, \mathbf{P})$ est défini de la manière suivante :

$$\chi(\mathbf{x}, \mathbf{P}) = (\det \mathbf{M})^{1/3} t/w$$

où \mathbf{x} est un point de l'espace \mathcal{P}^3 de coordonnées (x, y, z, t) qui se projette en \mathbf{u} de coordonnées (u, v, w) sur la caméra ayant pour matrice de projection $\mathbf{P} = [\mathbf{M}|\mathbf{v}]$. Seul le signe de χ nous intéresse. On remarque que ce signe est invariant à un changement de coordonnées homogènes. En fait, ce signe est exactement celui de $\mathbf{1}_3^T \cdot \mathbf{M}$ dans nos calculs. Précisément, si l'on désire garder les mêmes conditions aux limites, c'est-à-dire que $\chi = \infty$ sur le plan focal et $\chi = 0$ sur le plan à l'infini, alors on peut considérer que

$$\chi \doteq \frac{1}{\mathbf{1}_3^T \cdot \mathbf{M}}$$

Les mêmes résultats sont alors obtenus :

- Le signe des objets mathématiques (matrices et vecteurs) est important et ne doit pas être négligé. Pour nous, il s'agit de savoir à quelle partie de \mathcal{T}^3 le point appartient. Pour Richard Hartley, le monde est fondamentalement euclidien, le signe des vecteurs est donc important.
- Un point est dit devant une caméra si $\chi(\mathbf{x}, \mathbf{P}) > 0$ ou, de manière équivalente si $\mathbf{1}_3^T \cdot \mathbf{M} > 0$. Ces deux expressions représentent la même propriété physique. Le point 3-D est d'un côté ou de l'autre du plan focal.

4.4.2 Toutes les reconstructions projectives ne sont pas possibles

Hartley définit les réalisations fortes comme des reconstructions possibles au sens de l'orientation. Par reconstruction possible, on entend la même chose qu'à la section précédente, c'est à dire une reconstruction qui se trouve devant les plans focaux. Il en déduit qu'il doit exister un plan Π_∞ tel que tous les points reconstruits soient devant la caméra et (de manière implicite) devant le plan à l'infini. La position de ce plan à l'infini n'est malheureusement pas connue en général. Ce plan peut donc être déplacé à loisir. La position laissant le plus de liberté est celle d'un plan contenu dans les zones $-+$ et $+ -$ et passant par la droite d'intersection des plans focaux (voir figure 4.15). On

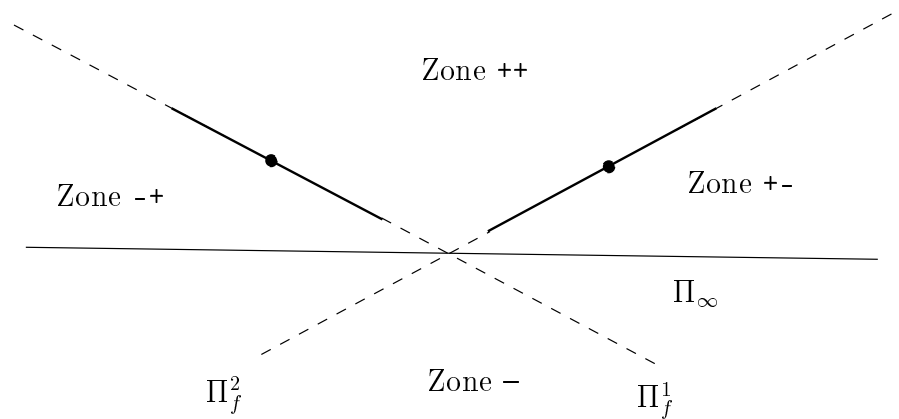


FIG. 4.15 – Position possible du plan à l’infini par rapport aux plans focaux.

retrouve alors l’impossibilité de distinguer les zones ++ et – sans information sur la position du plan à l’infini.

Les réalisations faibles sont des reconstructions projectives quelconques qui ne vérifient aucune propriété particulière, c’est à dire qu’elles peuvent traverser les plans focaux, être partiellement ou entièrement derrière les images, etc.... Comme nous, Hartley démontre que toutes les réalisations faibles ne sont pas possibles. Les réalisations fortes, par contre, le sont.

4.4.3 Visibilité

Hartley démontre aussi qu’il existe seulement deux orientations possibles de la scène, que l’on peut désambiguer dès que l’on connaît la position du plan à l’infini. Il rajoute cependant un élément intéressant : si il n’existe pas de plan séparant la scène des deux centres optiques, alors l’orientation peut être déterminée de manière unique. Ceci découle directement du fait que, dans ce cas précis, nous pouvons positionner le plan à l’infini par rapport à la scène et aux caméras.

En effet, le plan à l’infini ne peut pas se trouver entre la scène et les caméras, par hypothèse. Les caméras et la scène sont donc du même côté du plan à l’infini. On peut alors décider lequel de C_2 ou $-C_2$ est le point correct une fois le représentant de C_1 choisi. Le signe de l’épipole est alors fixé. La visibilité en découle.

4.5 Conclusion

Dans cette partie, nous avons introduit dans la vision par ordinateur un nouveau concept, celui d'orientation projective. A l'aide de modifications très simples (garder un signe) aux théories existantes, nous obtenons des propriétés très utiles pour un certain nombre d'applications.

Les modifications de la théorie projective en vision entraînent des modifications de son interprétation. Le plan focal possède désormais un "devant" et un "derrière". Les épipoles et les droites épipolaires ont une orientation, qui est reliée au sens du parcours de la lumière sur le rayon. Les plans de l'espace peuvent eux aussi être orientés, et ceci nous permet de définir des enveloppes convexes sans abandonner l'agréable formalisme de la géométrie projective.

Les applications sont multiples. Celles que nous avons développées ici sont au nombre de deux, mais on peut en imaginer d'autres :

- Suppression de points aberrants lors d'une reconstruction projective.
- Organisation d'une géométrie épipolaire pour tenir compte des contraintes de visibilité.
- Raffinement des invariants à des fins de reconnaissance comme dans [Hartley, 1996] où dans les travaux de Charlie Rothwell.

Troisième partie

Applications

Chapitre 5

Applications

5.1 Synthèse de nouvelles vues sans modèle tridimensionnel

L'utilisation de matrices de projection projectives est particulièrement appropriée pour les applications dont le domaine d'action se limite aux images. En effet, si il semble manquer de l'information entre notre monde 3-D projectif et le monde 3-D euclidien, pour les images les matrices de projection contiennent autant d'information qu'une calibration complète.

Nous étudierons ici deux applications différentes : Tout d'abord la synthèse de vues nouvelles sans modèle tridimensionnel, puis nous présenterons une méthode pour compresser les séquences d'images de manière non destructive.

5.1.1 Intérêt de la démarche

La synthèse d'images correspond traditionnellement à l'opération suivante : on projette un monde 3-D sur le plan rétinien d'une caméra virtuelle, en utilisant les attributs photométriques attachés au modèle 3-D. Ces attributs sont généralement la couleur, la réflectance, voire la bi-réflectance, la direction des sources lumineuses par rapport à l'objet, etc..

Dans les applications qui nous intéressent, le modèle 3-D est obtenu à partir d'images, qui ont été mises en correspondance, probablement prises avec des caméras préalablement calibrées. Le modèle 3-D a été reconstruit par

une intersection de rayons ou par n'importe quelle autre méthode. En suivant cette démarche, on voit que ces différentes images sources sont compilées en un modèle 3-D dont on peut donner un rendu facilement.

Il est clair aussi que cette démarche perd tout son intérêt si le modèle 3-D change en permanence. Les approches classiques sont donc bien adaptées à l'observation d'une maison ou d'une bouteille à partir d'une multitude de points de vue, mais inadaptées à la restitution d'un match de basket filmé par plusieurs caméras, où l'utilisateur se déplacerait virtuellement sur le terrain. De tels systèmes à partir d'images réelles sont actuellement à l'étude et correspondent à un développement probable des techniques de réalité virtuelle. L'université Carnegie-Mellon à Pittsburgh a un projet de ce type destiné à plusieurs applications, dont l'imagerie médicale. Ce projet est décrit dans [Kanade et al., 1995] ou sur le Web¹. D'autres projets existent comme celui de Mc Millan [McMillan et Bishop, 1995, McMillan, 1995].

Dans ce cas, la compilation de la collection d'images a peu de sens puisqu'il faut la refaire à chaque fois. Nous proposons ici une méthode qui se base sur la compilation de la structure de l'ensemble de caméras, qui lui, dans cette application, reste fixe. Pour reprendre l'exemple du match de basket, le spectateur reste immobile pendant de longues périodes par rapport aux joueurs qui se déplacent.

Cette compilation de la structure des images revient à court-circuiter l'étape intermédiaire de la génération du modèle 3-D. Au lieu d'avoir le système:

$$\begin{aligned}\text{modèle 3-D} &= F(\text{images sources}) \\ \text{image résultats} &= G(\text{modèle 3-D})\end{aligned}$$

on a

$$\text{image résultats} = G \circ F(\text{images sources})$$

L'essentiel de ce chapitre repose simplement sur la simplification de $G \circ F$.

Nous allons présenter dans ce chapitre principalement du transfert entre deux vues (dites vues sources) et une troisième (vue destination ou nouvelle). Ceci ne doit pas être vue comme étant en opposition avec les chapitres précédents où nous avons estimé la géométrie de plus de trois caméras. En effet, pour rendre le système utilisable, il faut prendre plus de deux vues de la scène, quitte à sélectionner deux d'entre elles au moment de la synthèse en

1. <http://www.ius.cs.cmu.edu/afs/cs/user/pjn/www/VirtualizedR.html>

fonction de la position de la vue destination. Toutes les images sources et l'image destination devront avoir une géométrie cohérente, ce qui explique nos calculs des chapitres précédents sur plus de trois caméras.

5.1.2 Transfert de primitives d'intérêt

Cette simplification qui consiste à inférer une image à partir d'autres images est communément appelée *transfert* dans la littérature de vision par ordinateur. Ces techniques sont issues de la stéréovision trinoculaire et de la reconnaissance d'objets à partir de vues multiples : [Barrett et al., 1992], [Ito et Ishii, 1986b], [Pietikainen et Harwood, 1986], [Pietikainen et Harwood, 1987], [Ayache et Lustman, 1987b], [Gurewitz et al., 1986], [Hansen et al., 1988], [Ito et Ishii, 1986a], [Milenkovic et Kanade, 1985], [Ohta et al., 1986].

L'idée est relativement simple : si on connaît deux points en correspondance dans les images 1 et 2, le point de l'image 3 leur correspondant se trouve à l'intersection des droites épipolaires engendrées par ces deux points, c'est à dire $\mathbf{F}_{13}\mathbf{m}_1$ et $\mathbf{F}_{23}\mathbf{m}_2$. La figure 5.1 illustre ce procédé. Ces méthodes ont été exploitées pour lever une ambiguïté entre des correspondances entre les deux premières images à l'aide de la troisième ou pour obtenir des cartes de disparité comportant moins de défauts et d'aberrations, mais aussi pour reconnaître une troisième vue d'un objet à partir de deux vues connues.

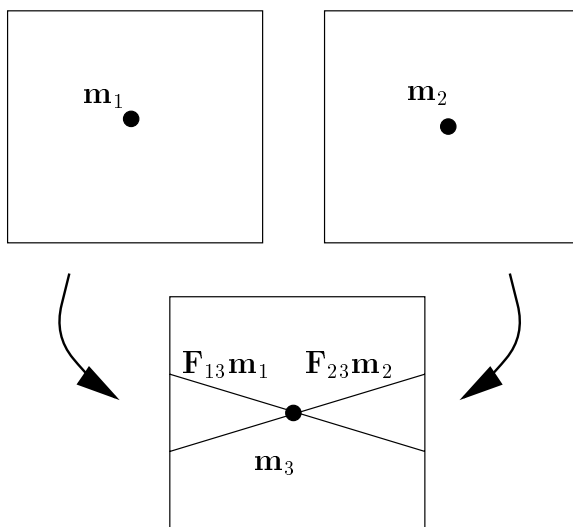


FIG. 5.1 – *Transfert épipolaire simple.*

Ces méthodes fonctionnent bien lorsque les caméras sont disposés en triangle, deux sur une base horizontale et une troisième plus en hauteur, ce

dont la plupart des gens s'assuraient naturellement dans le but d'obtenir une faible incertitude sur leur reconstruction, mais elle marche mal si le plan trifocal (plan contenant C_1 , C_2 et C_3) vient à couper le plan image (voir la comparaison de la figure 5.2). En effet, sur le plan trifocal, la droite épipolaire engendrée par \mathbf{m}_1 est confondue avec celle engendrée par \mathbf{m}_2 . L'intersection de ces deux droites est alors très sensible au bruit, qui peut venir soit de l'imprécision de la correspondance, soit de l'inexactitude de la géométrie épipolaire. Il faut alors utiliser une méthode différente, comme celle décrite dans [Okutomi et Kanade, 1991] par exemple. Nous allons présenter ici une méthode qui conviendra pour les deux cas de figures, sans que l'on ait à tester la configuration et choisir une méthode ou une autre.

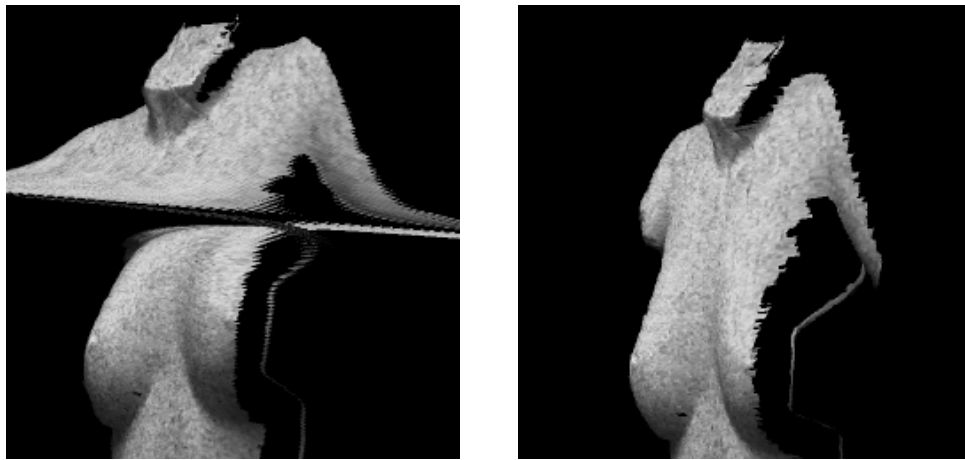


FIG. 5.2 – *Comparaison de méthodes de prédiction (droite : avec instabilités dans le plan trifocal, gauche : sans instabilités).*

Choisissons deux paires de lignes épipolaires qui sont “loin” de la trace du plan trifocal dans les images. “Loin” pouvant si nécessaire être hors des images, le choix d’une coordonnée ayant un ordre de grandeur différent conviendra. Nous notons ces deux paires (l_1^P, l_2^P) et (l_1^Q, l_2^Q) . Choisissons un point \mathbf{p}_1 sur l_1^P et un point \mathbf{p}_2 sur l_2^P . Ceci représentera un point \mathbf{P} de l’espace. Si \mathbf{M} est le point de l’espace correspondant à \mathbf{m}_1 et \mathbf{m}_2 , la ligne (\mathbf{PM}) rencontre le plan défini par (l_1^Q, l_2^Q) en un point \mathbf{Q} . Les images de \mathbf{Q} dans les images 1 et 2 sont \mathbf{q}_1 et \mathbf{q}_2 . Nous pouvons maintenant construire \mathbf{p}_3 (respectivement \mathbf{q}_3) comme l’intersection des lignes épipolaires engendrées par \mathbf{p}_1 et \mathbf{p}_2 (respectivement \mathbf{q}_1 et \mathbf{q}_2). \mathbf{p}_3 et \mathbf{q}_3 sont bien définis puisque \mathbf{P} et \mathbf{Q} sont loin du plan trifocal. \mathbf{m}_3 peut alors être facilement calculé comme l’intersection de $\mathbf{e}_{31} \times \mathbf{e}_{32}$ et $\mathbf{p}_3 \times \mathbf{q}_3$.

Cette construction est décrite dans la figure 5.3 dans les images et dans la figure 5.4 dans l'espace. On peut aussi calculer directement $\mathbf{p}_3 \times \mathbf{q}_3$ comme l'image de $\mathbf{p}_1 \times \mathbf{q}_1$ et de $\mathbf{p}_2 \times \mathbf{q}_2$ en utilisant les méthodes de [Faugeras et Robert, 1994]. Cette dernière méthode permet une prédiction légèrement plus rapide puisque plus de données peuvent être précalculées.

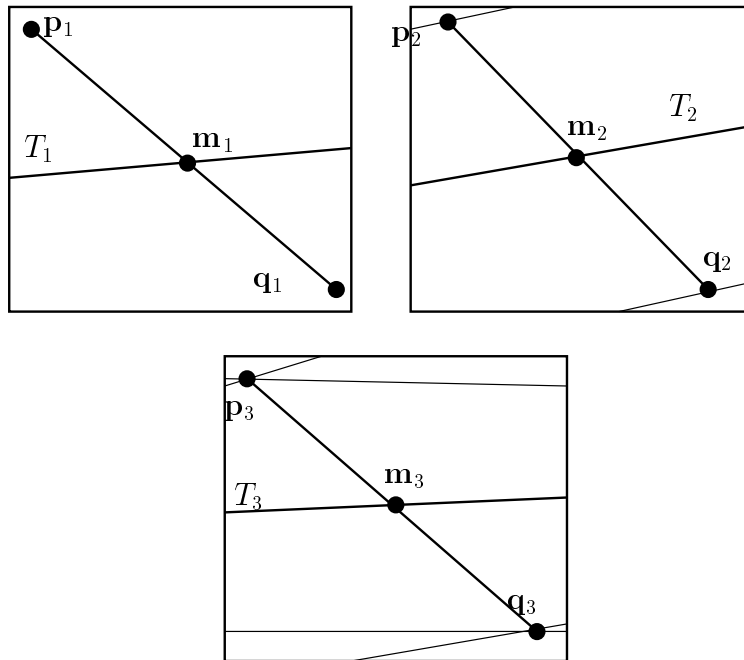


FIG. 5.3 – Transfert épipolaire à proximité du plan trifocal: vue 2-D.

Il faut noter que si le plan trifocal n'est pas défini, c'est à dire si les trois centres optiques sont alignés, ces méthodes ne s'appliquent pas. Elles ne sont pas nécessaires non plus si l'on dispose des trilinearités. Avec \mathbf{T} , le transfert est immédiat puisque l'on dispose d'équations pour trouver les coordonnées de \mathbf{m}_3 directement en fonction de \mathbf{m}_2 et \mathbf{m}_1 . Les trilinearités devront donc préférentiellement être utilisées si on n'utilise pas le côté géométrique de la prédiction. La prédiction des droites peut aussi se faire à l'aide des trilinearités [Hartley, 1993b].

5.1.3 Transfert d'image

La généralisation directe de ce schéma de prédiction où l'on construit une troisième image à partir des deux autres ne fonctionne malheureusement pas bien si l'on essaye de prédire tous les points d'une image. En effet, les

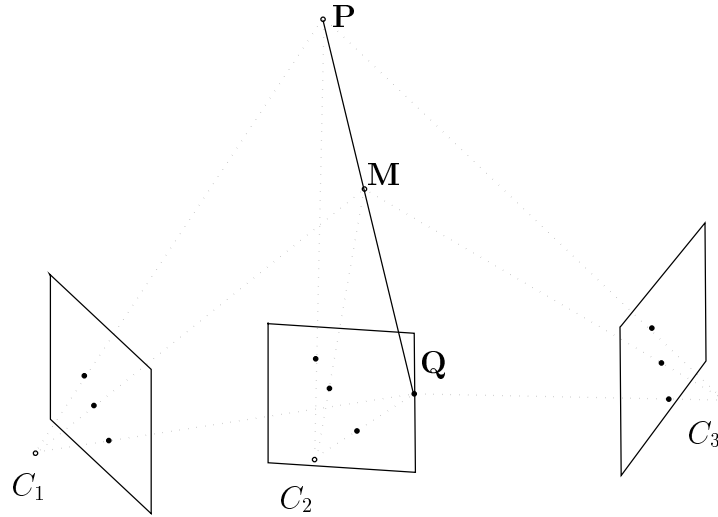


FIG. 5.4 – *Transfert épipolaire à proximité du plan trifocal : vue 3-D.*

points qui ont une distribution régulière dans la première image n'ont aucune raison d'être distribués régulièrement dans les autres. Considérons l'exemple simplifié de la figure 5.5. Un maillage régulier dans les deux images ne donne pas forcément le même échantillonnage sur l'objet. Ceci se traduit dans les images synthétisées par la présence de trous (voir figure 5.6).

Pour pallier ce problème, il faut parcourir non pas les pixels des images sources mais les pixels des images destinations. De cette manière, la discrétisation des images sources ne créera pas de vide dans l'image à synthétiser. Si on se place du point de vue spatial, cela revient à balayer le rayon issu du pixel de l'image destination au lieu de reconstruire un point à partir des images sources puis de le projeter dans l'image destination. Le parallèle avec le lancer de rayons est immédiat. Ceci complique notablement nos affaires, puisqu'au lieu d'avoir une inconnue (\mathbf{m}_3), nous en avons plusieurs ($\mathbf{m}_1, \mathbf{m}_2$). Lorsque nous parcourons les pixels de la troisième image (\mathbf{m}_3), nous parcourons les droites épipolaires dans les deux premières images. Il nous faut ensuite parcourir les pixels le long de ces droites épipolaires. Ces pixels sont liés par la carte de disparité. On va donc chercher les pixels qui se trouvent sur la droite épipolaire engendrée par \mathbf{m}_3 et dont tous les correspondants se trouvent sur une droite épipolaire engendrée par \mathbf{m}_3 .

Il peut y avoir plusieurs solutions à ce problème comme le montre la figure 5.7. Le choix entre les solutions n'est pas possible sans l'aide de la géométrie projective orientée. En effet, le point que nous recherchons est le premier que le rayon rencontre lorsqu'il part du centre optique vers l'infini, mais ce

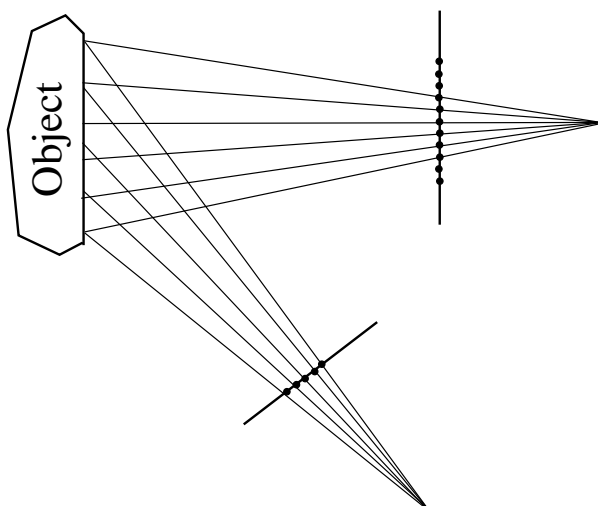


FIG. 5.5 – *Problème de discrétisation entre deux images.*

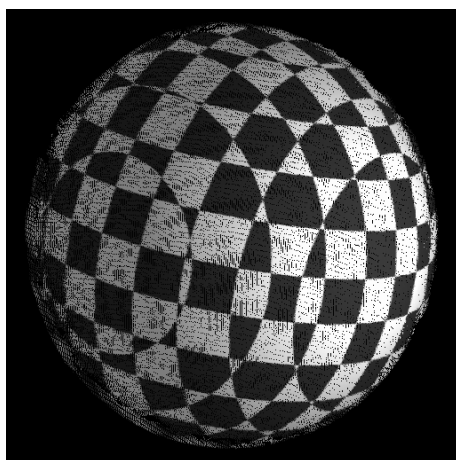


FIG. 5.6 – *Présence de trous dans l'image synthétisée. Les trous sont dus au fait qu'un balayage régulier de la scène dans une des images de départ ne garantit pas un balayage régulier de la scène dans l'image synthétisée. On voit donc apparaître des irrégularités.*

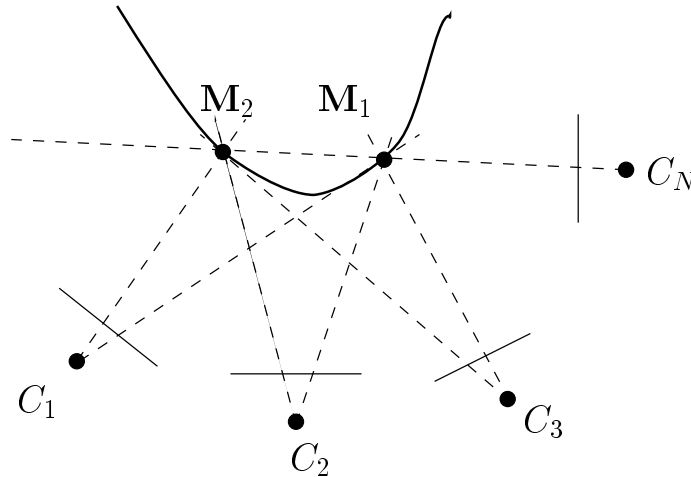


FIG. 5.7 – *Solutions multiples. Il existe plusieurs points de la scène sur le même rayon.*

rayon n'est pas orientable en géométrie projective classique et il n'est donc pas possible de distinguer le premier point du dernier.

5.1.3.1 A l'aide de la géométrie épipolaire

La situation est simple. Nous connaissons le point recherché dans l'image destination, mais dans aucune des images sources. A partir de la géométrie épipolaire, nous pouvons construire les lignes correspondant à ce point dans toutes les images sources. Il va falloir parcourir ces lignes dans le sens correspondant au rayon issue de l'image destination. Ce sens nous est donné par la matrice fondamentale orientée.

Si nous sommes loin du plan trifocal, pour chaque point dans la première image source, calculons ses correspondants à partir des cartes de disparité ou cartes de correspondances que nous avons obtenues au préalable. Ces correspondants se trouvent sur les droites épipolaires engendrées par le point source uniquement si le point spatial correspondant est sur le rayon partant du point source.

Le point que nous recherchons

- est sur les droites épipolaires dans toutes les images sources (pour être sur le bon rayon).
- est en correspondance par les cartes de disparité (pour être un point physique).

- est le premier sur le rayon à vérifier ces contraintes (pour être le premier point visible).

Nous pouvons alors parcourir le rayon orienté dans la première image source. A chaque point nous associons ses correspondants dans les autres images sources. Cette image par la carte de disparité du rayon va être une courbe dans les autres images. Cette courbe va couper la droite épipolaire en un ou plusieurs points (voir figure 5.8). Seul le premier de ces points nous intéresse. Les autres représentent des points de la scène sur le même rayon, mais invisibles depuis notre image.

Si nous nous trouvons à proximité du plan trifocal, il faut alors encore une fois utiliser une méthode avec un point extérieur. Le parcours ne se fait plus sur la droite épipolaire du point destination, parce que tous les points sur cette droite ont leur correspondant sur cette droite vue dans les autres images. Nous parcourons alors en fait l'espace des droites passant par un point donné extérieur au plan trifocal.

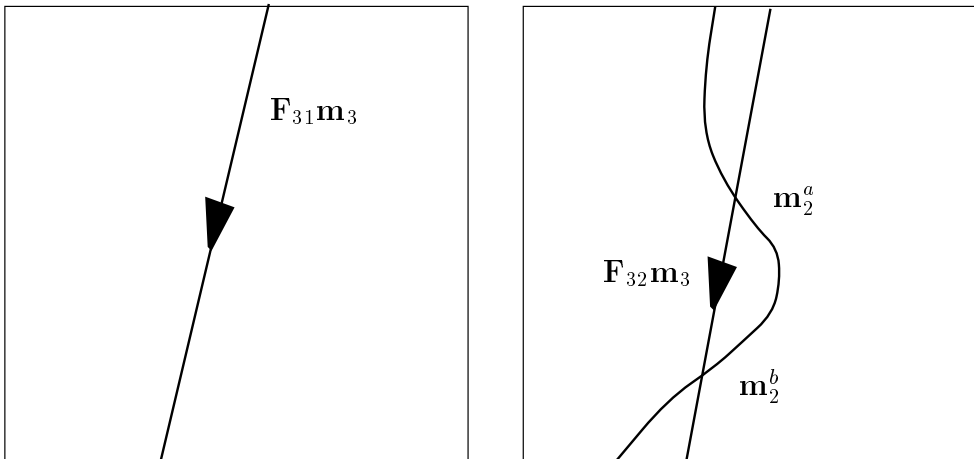


FIG. 5.8 – Les droites épipolaires dans les images sources ainsi que l'image par les cartes de disparité de la droite épipolaire dans la première image. Deux points de la scène se trouvent sur ce rayon. Seul le point \mathbf{M}^a est visible.

5.1.3.2 A l'aide des trilinearités

La situation avec les trilinearités est un petit peu différente. En effet, si nous substituons \mathbf{m}_3 dans les équations (2.17) nous obtenons 2 équations indépendantes où les coordonnées de \mathbf{m}_2 sont des fonctions linéaires des coordonnées de \mathbf{m}_1 . On obtient donc non pas une correspondance de droite à

droite, mais une correspondance de point à point, ces points étant contraints à être sur deux droites. La méthode n'a plus besoin d'être géométrique. Il suffit de trouver tous les points vérifiant les contraintes de disparité et les contraintes trilinéaires. Parmi ces points, le seul visible est celui qui est le plus près de l'épipole si celui-ci est orienté positivement, ou le plus loin si il est orienté négativement.

5.1.4 Résultats

5.1.4.1 Avec correspondances denses

La figure 5.9 présente un exemple de synthèse sans modèle tridimensionnel. Les deux images ont été acquises face à l'objet de la scène. L'image reconstruite est une vue de coté. On voit que les faces cachées sont correctement traitées, en particulier au niveau du cou et de la poitrine droite. Les manques dans la reconstitution sont dus à l'absence d'information dans les images sources. Le coté gauche du cou, par exemple, n'est pas visible dans les images sources, il est normal qu'il ne soit pas présent dans l'image finale. La fine bande sur la gauche est une erreur de la carte de corrélation. Si la mise en correspondance est fautive, alors l'information spatiale est fautive et l'image est forcément inexacte. La précision de nos calculs est inférieure au pixel.

5.1.4.2 Avec correspondances éparses

Parfois, l'obtention d'une carte de disparité est malaisée. Il est quand même possible de donner un rendu très réaliste de nouvelles vues sans avoir de cartes de correspondances dense. Pour les expériences qui vont suivre, seulement une cinquantaine de correspondances étaient disponibles sur toute l'image.

Tout d'abord, on effectue une triangulation de Delaunay entre les correspondances éparses dans les deux images. On obtient ainsi un nombre de triangles qui sont en correspondance affine les uns avec les autres, ce qui implicitement est équivalent à considérer que les zones définies par les triangles sont planes (ou apparaissent comme planes dans nos images). Pour chaque sommet de triangle, on calcule sa position dans la nouvelle image à partir de ses positions dans les images source et des trilinearités. Les pixels présents à l'intérieur du triangle peuvent être interpolés à partir des intensités dans les images sources. Cette interpolation n'est ni plus ni moins que du "texture mapping", qui peut être réalisé en temps réel sur des machines de type Silicon Graphics. Cela consiste simplement en une transformation affine des

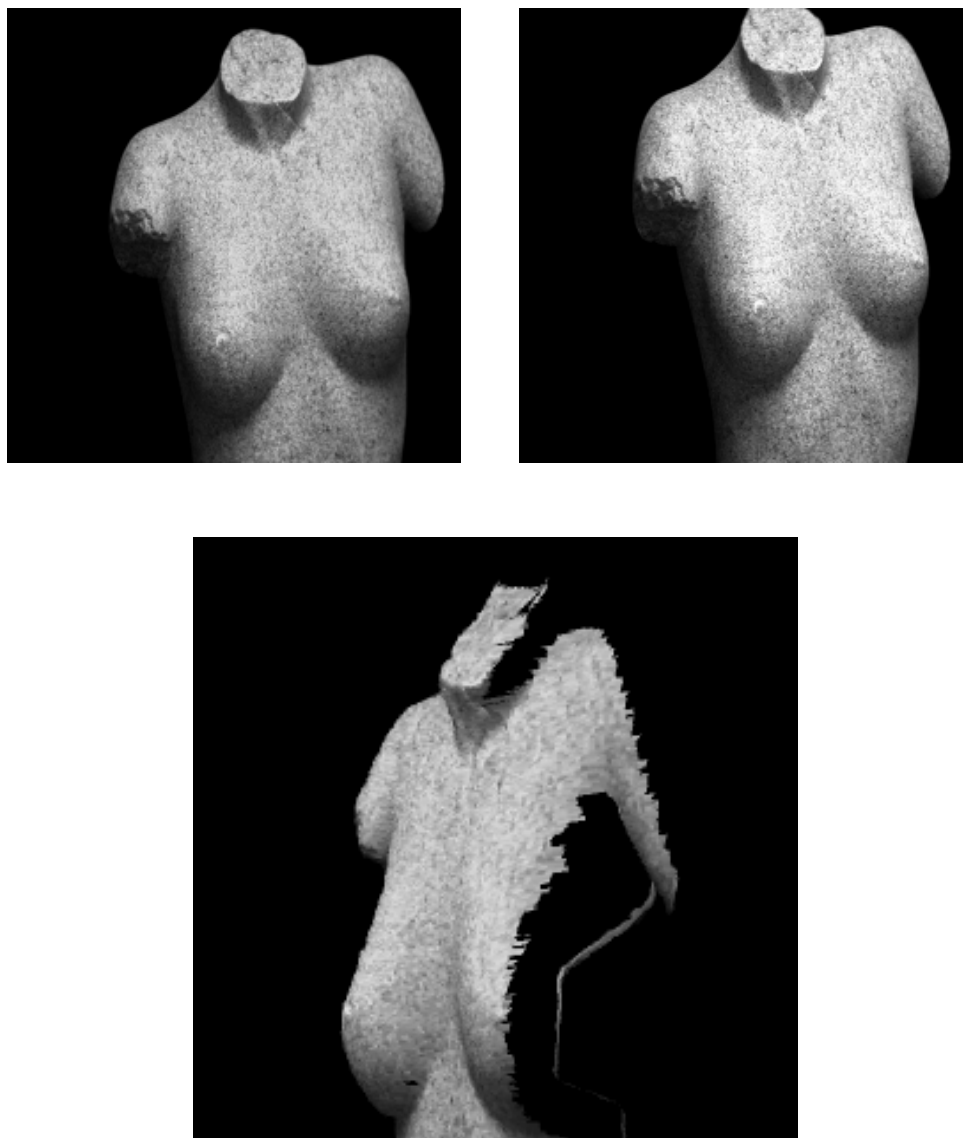


FIG. 5.9 – *Synthèse d'images sans modèle tridimensionnel. Les deux images sources sont présentées au dessus, l'image destination en dessous. Cette expérience est particulièrement convaincante car la position de l'image destination est très différente de la position des images initiales, exacerbant ainsi les défauts possibles.*

intensités des images source dans la nouvelle image.

La figure 5.11 montre les 6 images initiales utilisées pour générer la séquence qui va suivre. Pour obtenir un champ de vue assez large, nous avons regroupé ces images trois par trois comme montré à la figure 5.10. Ceci est possible puisque les images ont été prises du même point de vue [Szeliski, 1994, Hartley, 1994c]. Les deux images mosaïques sont présentées à la figure 5.12.



FIG. 5.10 – Configuration utilisée pour l'acquisition d'images

Le mode opératoire a été le suivant : On a d'abord calibré les caméras partiellement, en supposant que le centre de l'image était le point principal et 90 degrés l'angle entre les axes. On sait qu'entre deux photos prises du même point de vue il existe une homographie qui transforme une image dans le repère de la deuxième ([Szeliski, 1994, Hartley, 1994c] mais aussi [Irani et al., 1995, Anandan et al., 1994] et bien d'autres). Si on sait de plus que les deux images ont les mêmes paramètres intrinsèques, alors il existe de plus des contraintes liant cette homographie aux paramètres intrinsèques comme le démontre Hartley dans [Hartley, 1994c]. A partir de ces contraintes, on calcule la distance focale utilisée lors de la prise de vue. Le calcul de l'homographie se fait en mettant en correspondance des points dans les trois vues et en calculant la meilleure homographie possible entre elles. Le résultat est ensuite raffiné en utilisant les intensités dans l'image comme dans [Zoghلامي, 1995].

On a alors deux images mosaïques prises de deux points de vue différents. On connaît les paramètres intrinsèques de ces deux vues. On calcule la matrice fondamentale. A partir de la matrice fondamentale et des paramètres intrinsèques, on en déduit la matrice essentielle, que l'on décompose en une rotation et une translation. On peut alors spécifier la nouvelle vue dans le repère des deux vues de référence.

Quelques images des séquences engendrées sont présentés ici. La première séquence est une interpolation du mouvement entre les deux mosaïques. La séquence comporte 20 images, mais seulement 4 sont visualisées à la figure 5.13. La deuxième séquence est un mouvement inventé, une translation vers



FIG. 5.11 – *Les 6 images sources utilisées pour calculer les nouvelles images. Elles sont prises de 2 points de vue différents, avec à chaque fois trois orientations différentes. Les 3 premières photos ont été acquises d'un point de vue, les trois suivantes d'un autre point de vue. Les images ont été acquises avec un appareil photo standard.*



FIG. 5.12 – *Les deux images mosaïques engendrées à partir des 6 images précédentes. Les zones noires représentent des parties visibles dans aucune des images.*

l'avant à partir de la première caméra. La séquence vidéo comporte 50 images, mais seulement 5 sont présentées à la figure 5.14.

Le rendu visuel est bon, les séquences sont fluides, mais on peut remarquer quelques défauts dus à la représentation que nous avons utilisé. En effet, notre triangulation dans une image n'autorise pas les discontinuités de profondeur.

Les algorithmes de stéréovision supposent souvent l'existence d'une contrainte dite contrainte d'ordre qui dit que si deux points \mathbf{M}^a et \mathbf{M}^b sont situés sur le même objet opaque et dans un même plan épipolaire, alors leur projections sont nécessairement rencontrées dans le même ordre si l'on parcourt les droites épipolaires de gauche à droite dans l'une ou l'autre des images. Cette contrainte n'est pas toujours vérifiée comme le montre la figure 5.15.

Dès que les contraintes d'ordre ne sont plus vérifiées, des choses étranges peuvent se passer. En effet, la représentation suppose que l'ordre des triangles est le même dans toutes les images. Il suffit de regarder le lampadaire sur la gauche de l'image et son interaction avec l'arbre qui est derrière lui pour avoir une vision plus nette du problème. L'extrême gauche de l'image cause aussi des problèmes. En effet, plusieurs triangles (présentés à la figure 5.16) traversent la route et ont donc une grande différence de disparité d'un coin à l'autre. Ces coins vont se déplacer avec une amplitude très différente, ce qui va causer un étirement excessif du triangle. Dans les images présentées, les problèmes se posent au niveau du lampadaire sur la gauche et entre la jonction entre les arbres du fond et ceux du devant sur la droite. On a l'impression que le poteau se tord au fur et à mesure et que les arbres du devant traversent la route (ce qu'ils font effectivement dans cette représentation).

Cette modélisation de la scène en triangle n'est pas excellente non plus d'un point de vue théorique, puisqu'elle suppose que la scène est plane par morceaux, ce qui n'est évidemment pas vrai. Elle suppose aussi que la transformation affine qui permet de passer d'un triangle dans une image source à son correspondant dans l'image destination est une bonne approximation de l'homographie correspondant au plan du triangle. Comme démontré à l'appendice B, ceci signifie que cette transformation affine doit laisser l'épipole invariant. Ceci est rarement vérifié.

Cependant, malgré toutes ces petites incorrections, la séquence reste parfaitement crédible quant au mouvement 3-D. Il n'y a pas d'a-coups et le rendu visuel est bon.



FIG. 5.13 – 4 images extraites d'une séquence de 20. Le mouvement est une interpolation entre les mosaïques.



FIG. 5.14 – 5 images extraites d'une séquence de 50. Le mouvement est un mouvement de translation vers l'avant à partir de la première caméra.

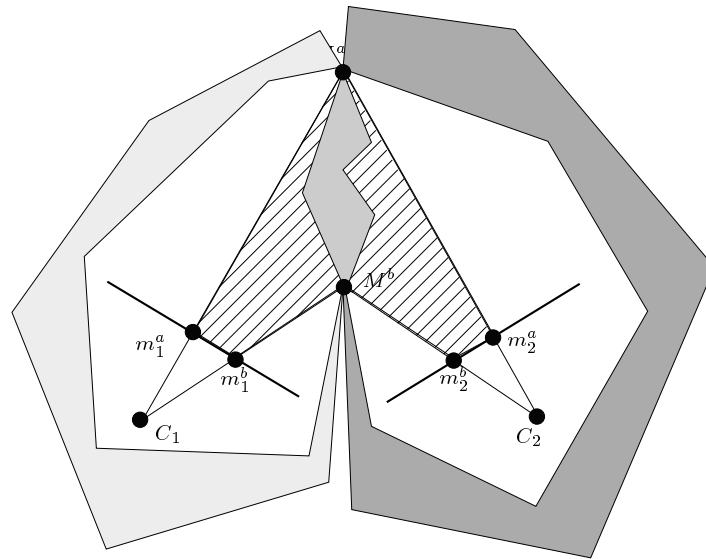


FIG. 5.15 – La contrainte d'ordre n'est pas vérifiée. En effet, les points \mathbf{m}_i^a et \mathbf{m}_i^b n'apparaissent pas dans le même ordre dans les deux images.

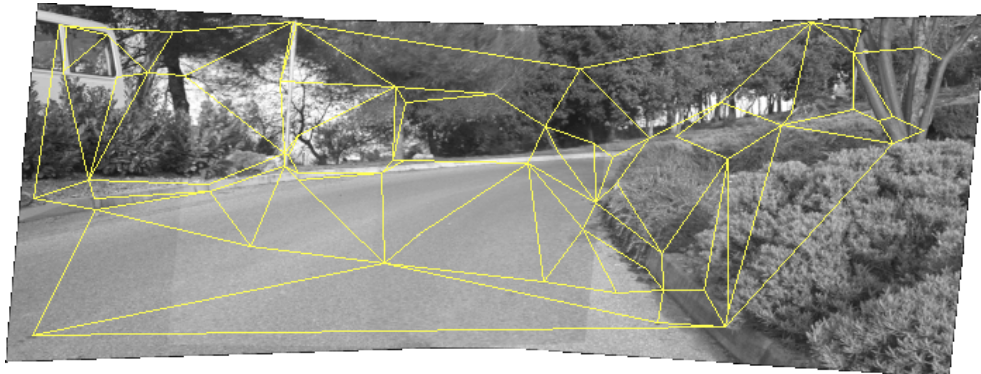


FIG. 5.16 – Triangulation dans l'image.

5.2 Construction de modèles 3-D à partir de séquences d'images

Malgré tout le bien que l'on peut penser des images, il existe de nombreuses applications où un modèle 3-D est désirable, voire indispensable. Il peut s'agir d'architecture, de réalité virtuelle pour laquelle seul un modèle tridimensionnel permet un rendu rapide de la scène, ou bien même de cartographie. Traditionnellement, les modèles 3-D sont obtenus manuellement, en spécifiant le modèle facette par facette, dans la majorité des cas, ou à partir de vues calibrées avec le plus grand soin dans les autres cas.

Les méthodes manuelles ont bien entendu comme inconvénient majeur leur aspect fastidieux. Les architectes passent des heures à dessiner et redessiner ce qui pourrait être construit à partir d'images. Mais le défaut le plus souvent constaté est le manque total de réalisme. L'environnement cubique à plans uniformes ne suffit pas pour des applications de rendu. Il est par ailleurs relativement difficile de choisir une texture (souvent trop parfaite) et de la coller sur des facettes. Il apparaît donc naturel de s'inspirer d'images et photos pour obtenir des rendus plus réalistes.

Les méthodes semi automatiques comme celles utilisées par les photogramètres ont l'inconvénient de nécessiter un équipement de grande précision et donc fort coûteux comme les caméras très haute résolution utilisées en cartographie aérienne. Ces caméras doivent être calibrées avec le plus grand soin. La recherche de la qualité lors de cette calibration est un problème encore d'actualité, comme en témoignent les nombreux articles parus sur ce sujet au cours de ces dernières années comme par exemple [Gruen et Beyer, 1992, Beyer, 1992]. Ces caméras sont toujours des caméras à focale fixe avec une mise au point à l'infini, ce qui est assez contraignant pour les applications grand public.

Si ces spécifications sont parfaitement adaptées pour la cartographie aérienne, elles ne correspondent pas à des problèmes pour lesquels moins de précision est requise, mais où la flexibilité et le prix ont une importance déterminante. Pour permettre une grande diffusion de ces outils dans le grand public, la calibration doit être inexistante ou cachée, les appareils photo ou caméras disponibles en grandes surfaces. Nous avons donc choisi de tester nos algorithmes sur deux types de matériels:

- une caméra vidéo Grundig VHS SECAM. VHS (marque déposée de JVC) est le standard en ce qui concerne les caméras vidéo non professionnelles (par opposition à Betamax de Sony qui est utilisé par des cameramen spécialistes). La caméra est dotée d'un autofocus (lent!).

- un appareil photo Canon EOS 100 (ou EOS Elan) avec un objectif à focale variable 28-105 mm. L'appareil est autofocus. Les films utilisés sont des films commerciaux à 100 ou 200 ASA de Kodak ou de Fuji. Le développement a été réalisé sur PhotoCD Kodak, un système de plus en plus répandu (disponible à la FNAC).

Le choix de ces deux techniques a été motivé par leur faible coût et leur très grande disponibilité dans le public.

Le système que nous allons présenter est en grande partie basé sur l'interactivité avec l'utilisateur. Certaines parties peuvent bien sûr être automatisées, mais au prix d'une moins grande robustesse. Les différentes étapes de la modélisation sont

- Le calcul de la géométrie projective des caméras.
- Le calcul de l'homographie qui transforme notre monde projectif en un monde euclidien à un facteur d'échelle près. Ce calcul nécessite l'intervention de l'utilisateur. En effet, nous savons que la géométrie projective est la seule information que nous pouvons trouver à partir des images seulement. L'utilisateur doit donc faire des hypothèses sur la structure tridimensionnelle de la scène. Ici, les droites parallèles permettront de déterminer le plan à l'infini, tandis que les angles ou les rapports de distances induiront un monde euclidien.
- La détermination des primitives d'intérêt dans les images de manière interactive. L'utilisateur veut modéliser un polygone dans l'espace, il doit cliquer les points de ce polygone dans deux images. La connaissance de la géométrie projective permet de trouver le polygone dans toutes les autres images. Nous utiliserons pour ce faire, soit les trilinearités soit de manière équivalente, une reconstruction puis une reprojction.

Le premier point a été traité aux chapitres 2 et 3. Le calcul de l'homographie est développé dans la section 5.2.1. La modélisation proprement dite sera traitée à la section 5.2.2.

5.2.1 Calcul de l'homographie de retour à l'euclidien

Nous suivons ici la philosophie développée dans [Faugeras, 1995] pour la stratification des représentations euclidienne, affine et projective de l'espace.

Le calcul se décompose naturellement en deux parties. Nous allons tout d'abord passer du monde 3-D projectif à un monde 3-D affine où nous aurons identifié le plan à l'infini. Nous nous ramènerons ensuite à un espace euclidien

à une échelle près en identifiant des angles et des rapports de distances. Si on le désire, le choix d'une valeur pour une distance nous ramènera à un modèle 3-D euclidien. Ceci n'est pas toujours nécessaire, en particulier pour les applications liées à la synthèse d'images et la réalité virtuelle.

5.2.1.1 Du projectif à l'affine

Nous utilisons le plongement standard de \mathcal{A}^3 dans \mathcal{P}^3 , obtenu en identifiant \mathcal{A}^3 à \mathcal{P}^3/Π_∞ , où Π_∞ est un plan appelé plan à l'infini et qui peut être vu comme l'ensemble des directions des droites de \mathcal{A}^3 . Ce plongement est équivalent au choix de $T = 0$ pour plan à l'infini.

La connaissance de Π_∞ dans notre espace projectif nous permet de choisir une base telle que le plan à l'infini ait effectivement $T = 0$ pour équation. Cette base sera alors une base affine de la scène.

Une propriété de l'espace affine par rapport à l'espace projectif est d'avoir identifié l'ensemble des points qui sont à l'infini. Pour nous, la meilleure façon de déterminer cet ensemble est de trouver le point d'intersection de droites parallèles dans la scène. Une détection automatique est possible. On peut, en passant par l'espace dual des droites du plan, c'est à dire les points du plan, faire passer une droite par un maximum de points. Cette droite est le dual du point d'intersection des droites. Cette méthode est relativement risquée et peut mener à des erreurs qui seront fatales pour notre retour à l'espace affine. Plutôt, nous utiliserons une méthode interactive, où l'utilisateur, aidé de la méthode automatique si besoin est, choisit les droites parallèles dans les images. Avec trois directions non coplanaires, nous pouvons retrouver l'équation du plan à l'infini Π_∞ puisque nous pouvons trouver trois points non alignés sur le plan à l'infini. Il nous suffit ensuite de choisir pour notre homographie une application qui envoie Π_∞ sur le plan d'équation $T = 0$.

Plus précisément, soit $\langle D_i \rangle$ les droites parallèles de l'espace dont nous mesurons les projections $\langle d_{ij} \rangle$ dans les images. $\langle d_{ij} \rangle$ est représenté par le vecteur \mathbf{d}_{ij} . Nous voulons calculer le point d'intersection $\mathbf{V} = \cap_i \langle D_i \rangle$ que nous savons être sur Π_∞ . Le problème que nous cherchons à résoudre est le suivant : étant donné \mathbf{d}_{ij} , trouver \mathbf{V} . Puisque l'image de \mathbf{V} dans la caméra j est sur $\langle d_{ij} \rangle$,

$$\forall (i, j) \quad \mathbf{d}_{ij}^T \mathbf{P}_j \mathbf{V} = 0 \quad (5.1)$$

Ce système comporte autant d'équations qu'il y a de lignes visibles dans les images. Il est linéaire et homogène en fonction des quatre coordonnées de \mathbf{V} . Nous pouvons le résoudre avec une méthode d'extraction des valeurs

singulières. En effet, \mathbf{V} est le vecteur singulier correspondant à la plus petite valeur singulière de la matrice dont les rangées sont $\mathbf{d}_{ij}^T \mathbf{P}_j$.

On aurait aussi pu calculer les images de \mathbf{V} dans les images comme meilleur point d'intersection des \mathbf{d}_{ij} , mais les point \mathbf{v}_j n'auraient alors pas vérifié les contraintes épipolaires. Expérimentalement, les résultats sont moins stables.

Le procédé précédent peut être appliqué pour autant de familles de droites parallèles que désiré. On obtient alors une collection de points \mathbf{V}_k qui sont sur le plan à l'infini. Le calcul de Π_∞ se fait grâce aux équations suivantes:

$$\forall k \quad \mathbf{V}_k \Pi_\infty = 0 \quad (5.2)$$

Ce système d'équations est résolu de la même manière que précédemment, avec un calcul de valeurs singulières.

La figure 5.17 montre un ensemble de droites parallèles utilisé pour nos calculs. La figure 5.18 montre une reconstruction affine de la scène. Une figure montrant une reconstruction projective quelconque est inutile, parce que la scène est en général méconnaissable.

5.2.1.2 De l'affine vers l'eulidien

A cette étape, nous allons demander à l'utilisateur de sélectionner dans les images des paires de droites perpendiculaires dans l'espace. Leurs intersections avec le plan à l'infini $\mathbf{V}_i, \mathbf{V}_j$ représentent leurs directions. Alors, leur orthogonalité s'exprime de la manière suivante:

$$\mathbf{V}_i^T \mathbf{V}_j = 0 \quad (5.3)$$

Nous cherchons une transformation de la base affine obtenue précédemment en une base euclidienne orthonormée. Cette transformation que nous cherchons à évaluer possède 5 paramètres: En effet, nous pouvons transformer un point quelconque en l'origine du repère \mathbf{O} , la première direction en $[1, 0, 0, 0]$, la deuxième direction en $[\lambda, 1, 0, 0]$, la troisième en $[\mu, \nu, 1, 0]$. Il reste alors les facteurs d'échelle suivant ces axes: α, β, γ puisque nous désirons obtenir une base orthonormée et non pas une base seulement orthogonale. Nous savons qu'il est possible de retrouver la scène à un facteur d'échelle près seulement. Nous ne pouvons calculer que β/α et γ/α . Nous pouvons donc décider, par exemple, que $\alpha = 1$.

Les paramètres de notre transformation sont donc $\lambda, \mu, \nu, \beta, \gamma$. Le critère que nous allons chercher à optimiser est



FIG. 5.17 – Ensemble de droites parallèles détectées dans une image.

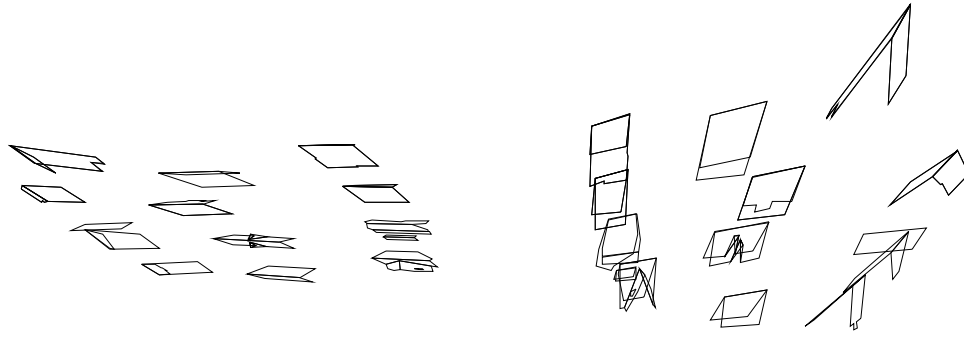


FIG. 5.18 – *Reconstruction affine d’une scène, ici des toits d’un village suisse. Vue de haut (droite) et vue de coté (gauche). La scène est très distordue, mais le parallélisme est conservé. La base a été choisie pour que le plan à l’infini ait $T = 0$ pour équation. L’origine est un des points de la scène et les trois directions ont été choisies non coplanaires mais ne représentent rien de particulier.*

$$C = \sum_{(i,j)} (\mathbf{V}_i^T \mathbf{A}^T(\beta, \gamma, \lambda, \mu, \nu) \mathbf{A}(\beta, \gamma, \lambda, \mu, \nu) \mathbf{V}_j)^2 \quad (5.4)$$

avec

$$\mathbf{A}(\beta, \gamma, \lambda, \mu, \nu) = \begin{pmatrix} \alpha & \beta\lambda & \gamma\mu & 0 \\ 0 & \beta & \gamma\nu & 0 \\ 0 & 0 & \gamma & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Il faut bien sûr normaliser les vecteurs \mathbf{V} pour éviter que certains prennent plus d’importance que d’autres. Si les vecteurs sont normalisés, le critère représente la somme des carrés des cosinus des angles entre vecteurs. La minimisation de C en fonction des paramètres de la transformation n’est pas linéaire et nous utilisons une méthode itérative de Levenberg-Marquardt comme celle décrite à la section 3.4.5.1 pour obtenir les meilleures valeurs possibles des paramètres. L’initialisation de la minimisation a peu d’influence sur les résultats finaux. Une initialisation possible est $\alpha = \beta = \gamma = 1$ et $\lambda = \mu = \nu = 0$. En combinant les deux transformations (projectif vers affine et affine vers euclidien), on obtient une homographie \mathbf{H} transformant notre reconstruction projective en une reconstruction euclidienne.

La figure 5.19 montre des droites perpendiculaires sélectionnées dans les images pour calculer la transformation affine vers euclidien. La figure 5.20 montre une reconstruction euclidienne obtenue après la minimisation. La scène est désormais reconnaissable.



FIG. 5.19 – *Droites perpendiculaires sélectionnés pour calculer la transformation de la reconstruction affine vers la reconstruction euclidienne.*

5.2.2 Modélisation dans les images

La connaissance de la géométrie projective de l'ensemble des caméras peut aider l'utilisateur au cours de la modélisation. En effet, un point sélectionné dans plusieurs images doit satisfaire certaines contraintes. Imposer ces contraintes au moment de la sélection du point permet

- de simplifier la modélisation pour l'utilisateur en minimisant le nombre de clics souris nécessaires.
- de s'assurer que les points dans les images représentent effectivement des points physiques. La reconstruction est plus simple dans ce cas.

La modélisation d'un polygone, d'un segment ou d'un point peut être vue comme plusieurs modélisations de points. Nous décrivons donc seulement la

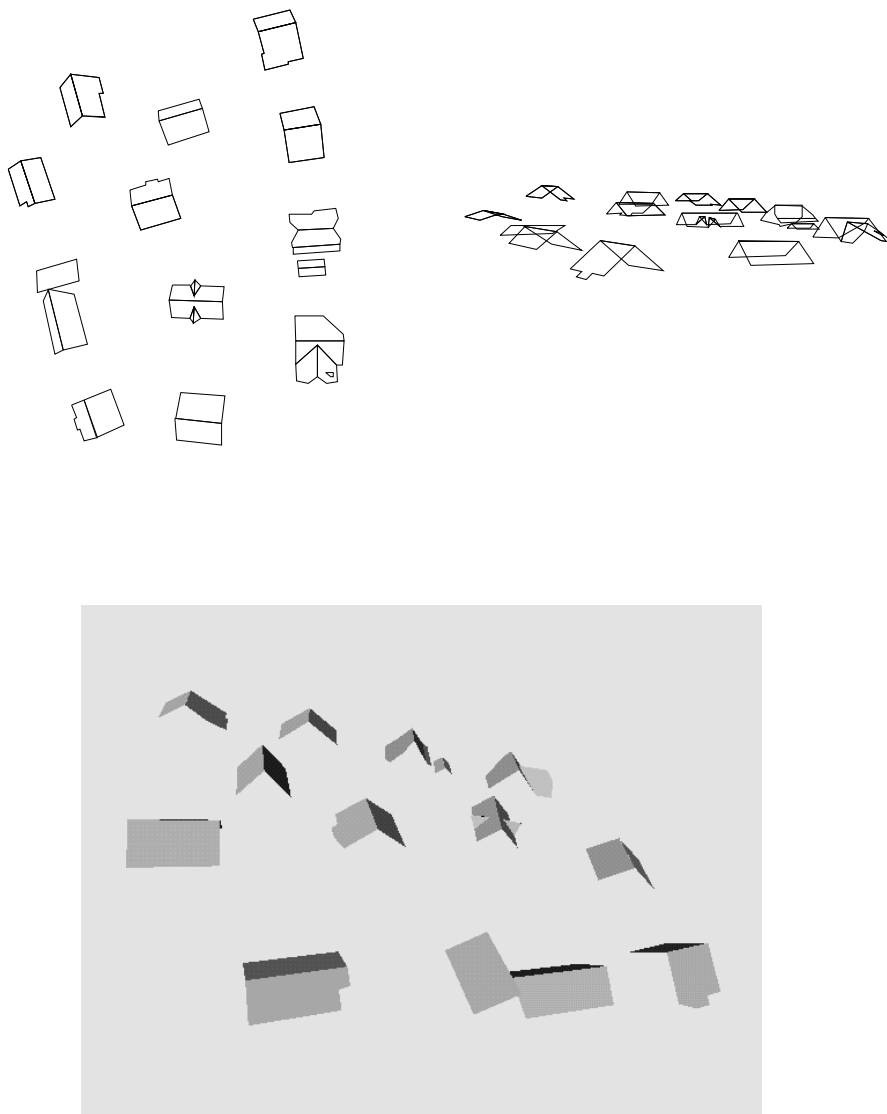


FIG. 5.20 – *Reconstruction de la scène après la minimisation des paramètres de retour à l'eulidien. Vue de dessus (droite), de côté (gauche) et de biais (bas). La scène est désormais reconnaissable, les angles et les rapports de distance sont préservés ainsi que le parallélisme.*

modélisation d'un point 3-D à partir des images, élément de base de toutes les autres modélisations.

L'utilisateur sélectionne le point qui l'intéresse dans une des images. Les contraintes données par la géométrie des caméras imposent que dans toutes les autres images, le point se trouve sur la droite épipolaire engendrée par le point sélectionné comme le montre la figure 5.21.



FIG. 5.21 – *Un point dans une image (à gauche) contraint ses correspondants à se trouver sur sa droite épipolaire (à droite)*

L'utilisateur choisit alors une deuxième image. Le point qu'il désire sélectionner est contraint à se trouver sur la droite épipolaire. Une fois ce point déterminé, nous pouvons calculer les coordonnées des projections du point 3-D correspondant dans toutes les autres images (voir figure 5.22), soit en utilisant des trilinearités, qui donneront directement le troisième point en fonction des deux premiers, soit par une reconstruction à partir de deux images puis une projection dans les images restantes. Ces deux méthodes sont équivalentes d'un point de vue géométrique, mais la méthode utilisant les trilinearités est plus rapide car plus compacte. De plus, les trilinearités peuvent être partiellement précalculées après le choix du premier point. La fonction liant le deuxième point à un point d'une autre image est alors linéaire.

5.2.3 Reconstruction

Une fois modélisée, la scène est reconstruite par simple intersection de rayons (les rayons se coupent car nous avons imposé cette contrainte au moment de la modélisation). On associe alors à chaque face des propriétés photométriques pour permettre un rendu réaliste. Idéalement, ces attributs photométriques sont déduits des images.



FIG. 5.22 – *Le choix d'un deuxième point (contraint sur une droite épipolaire) détermine sans ambiguïté la position de ses correspondants dans toutes les autres images.*

Les figures 5.23 et 5.25 présentent une vue de dessus de la bibliothèque de l'INRIA à Sophia-Antipolis. Les figures 5.24 et 5.26 présentent un autre angle de vue. Ce modèle 3-D a été obtenu à partir de cinq images prises avec l'appareil photo mentionné plus haut.

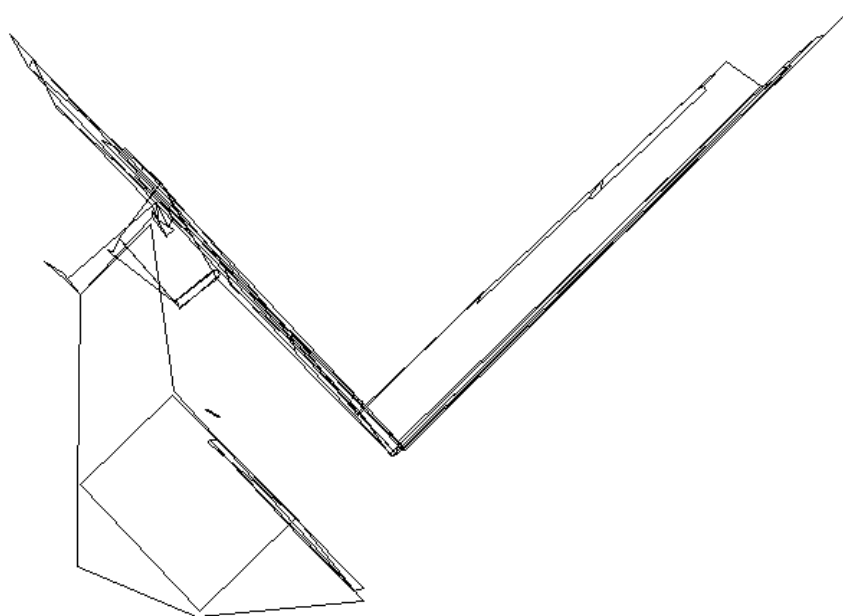


FIG. 5.23 – *Vue de dessus de la bibliothèque de l'INRIA. Vue en fil de fer.*

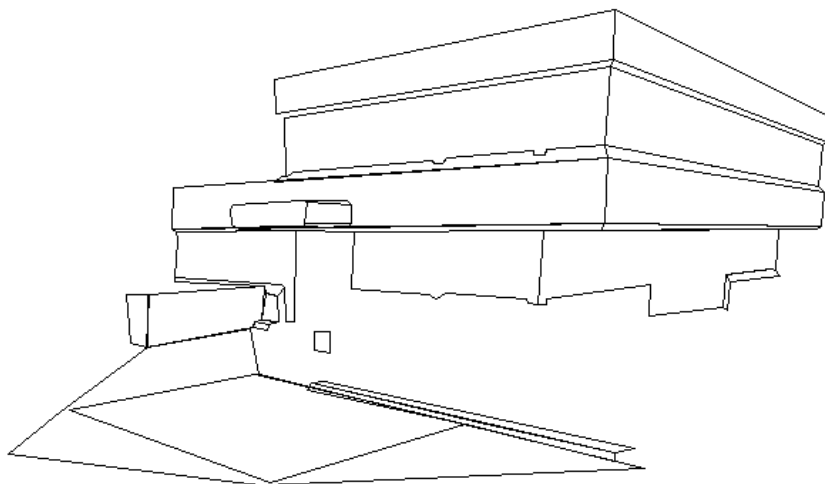


FIG. 5.24 – *Vue de côté de la bibliothèque de l'INRIA. Vue en fil de fer.*

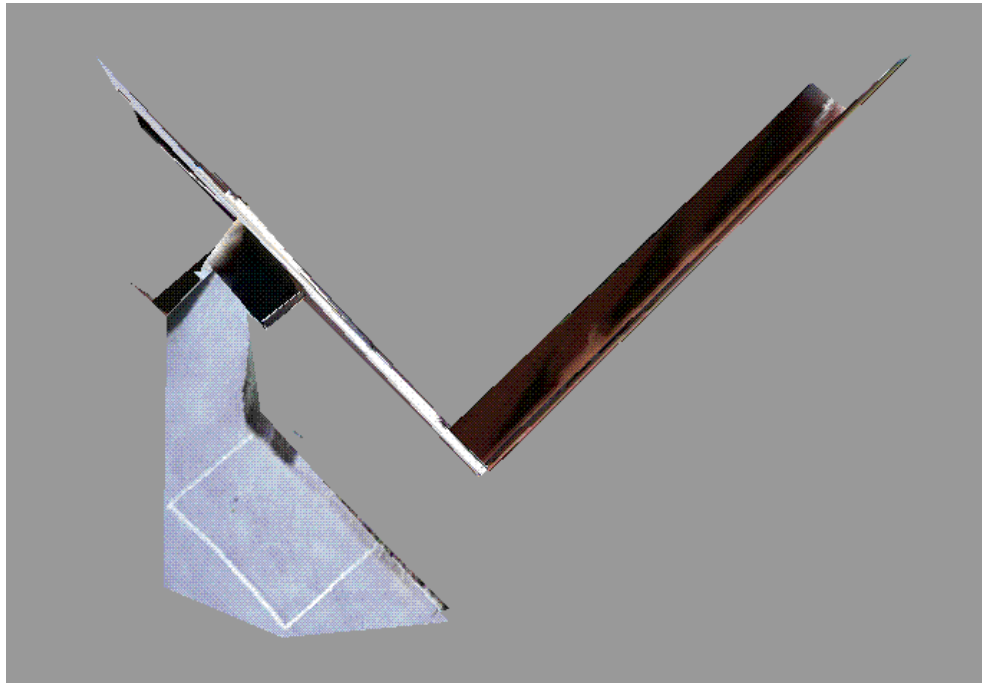


FIG. 5.25 – *Vue de dessus de la bibliothèque de l'INRIA. Image synthétique.*



FIG. 5.26 – *Vue de côté de la bibliothèque de l'INRIA. Image synthétique.*

Chapitre 6

Conclusion

6.1 Contributions et résultats

6.1.1 Géométrie des caméras

Le chapitre 2 nous a montré quelles sont les relations entre les différents objets apparaissant dans la théorie d'un système multi-caméras et comment ces objets apparaissent et sont utilisés. On retiendra que la représentation la plus générale est celle des matrices de projection, qui ne souffre d'aucune contrainte algébrique liant ces paramètres ni de dégénérescences. De plus, les matrices fondamentales et les matrices de projection se calculent facilement à partir des matrices de projection. Nous avons aussi montré comment tous nos objets se dérivent de manière systématique à travers la théorie des déterminants.

Au cours du chapitre 3 nous avons expérimenté diverses méthodes pour calculer cette géométrie. Les méthodes basées sur les matrices fondamentales et sur les trilinearités donnent des résultats corrects, mais la méthode basée sur une estimation directe des matrices de projection fonctionne encore mieux. Si cela est nécessaire, on peut raffiner les résultats en utilisant une ou plusieurs méthodes présentées ici. La méthode de l'ajustement de rayon a ma faveur, de part sa simplicité et des bons résultats qu'elle donne. Nous avons aussi présenté différentes méthodes pour recaler des sous-séquences entre elles. Avec chacune de ces étapes sont présentés des résultats sur des images synthétiques et sur des images réelles.

6.1.2 Géométrie projective orientée

Le chapitre 4 introduit une nouveauté rendue nécessaire par la synthèse de vues, l'orientation en géométrie projective. Ce concept, nouveau en vision par ordinateur, s'applique facilement et de manière élégante. Diverses applications en découlent, comme l'élimination de fausse correspondance et la définition d'enveloppe convexe. Les droites épipolaires sont alors orientées, ce qui rend la synthèse de nouvelles vues possibles en désambiguant les points visibles et ceux qui ne le sont pas.

6.1.3 Applications

Nous avons fait au chapitre 5 une présentation de différentes méthodes de synthèse de nouvelles vues à partir de vues connues et ce sans modèle tridimensionnel. La faisabilité est démontrée et plusieurs exemples de séquences sont disponibles. Les occultations sont convenablement traitées grâce à la géométrie projective orientée.

La construction de modèles tridimensionnels euclidiens est vue au chapitre 6. Pour construire un modèle euclidien à partir d'un modèle projectif, il suffit de rajouter certaines contraintes comme le parallélisme ou l'orthogonalité. On peut alors obtenir des modèles utilisables en réalité virtuelle ou en synthèse d'images.

6.2 Extensions

Plusieurs extensions sont possibles à cette thèse, aussi bien au niveau de l'estimation de la géométrie des caméras que des applications.

6.2.1 Géométrie des caméras

- Dans cet ouvrage, nous considérons le cas d'une scène rigide et d'une caméra mobile ou bien une scène rigide en mouvement et une caméra fixe. Dans un certain nombre d'applications, ceci peut s'avérer insuffisant puisqu'il n'est pas toujours possible d'avoir un seul objet rigide dans la scène. Les méthodes présentées ici sont robustes à la présence de petits objets mobiles dans la scène qui sont alors éliminés comme fausses correspondances, mais la présence de deux mouvements rigides d'égale importance n'est pas traitée.
- Nous avons choisi de nous placer dans le cas le plus général, où les paramètres intrinsèques de la caméra varient librement d'une image à

l'autre. En réalité, au cours d'une séquence d'images prises à des instants successifs, les paramètres intrinsèques varient relativement lentement. On peut donc utiliser des méthodes de suivi comme le filtrage de Kalman pour obtenir une estimée initiale des matrices de projection.

- Le cas des trilinearités mis à part, nous n'avons pas utilisé les droites dans les images pour notre estimation. Tout porte à croire que ce serait bénéfique, puisque certains paramètres des droites (l'orientation par exemple) sont estimés avec beaucoup de précision à partir des images.
- On veut aussi pouvoir déterminer automatiquement le découpage en sous-séquences qui va permettre la meilleure estimation. Les critères à prendre en compte sont entre autres le nombre de points suivis entre les différentes images d'une sous-séquence (doit être suffisamment grand), le déplacement des points en pixels (doit être suffisamment grand aussi). D'autres paramètres peuvent aussi entrer en ligne de compte.

6.2.2 Transfert et synthèse de nouvelles vues

- Pour la synthèse à base de correspondances éparses, nous avons utilisé des points mis en correspondance manuellement. Ce processus pourrait être partiellement automatisé en effectuant une segmentation de la scène en zones localement planes.
- Le problème des discontinuités reste à traiter. En effet, notre représentation n'autorise pas de discontinuités de profondeur dans la scène. On peut éliminer ce problème en séparant la scène en plusieurs couches visuelles comme dans le dessin animé. Ces couches ne sont pas nécessairement liées à la profondeur. Le découpage en couche peut être une opération interactive (facile) ou le résultat d'une segmentation (plus difficile).
- Nous n'avons pas étudié ici comment sélectionner la paire de vue la plus appropriée parmi les N disponibles. Il faudrait bien sûr prendre en compte la proximité des centres optiques, mais aussi l'orientation des caméras.

6.2.3 Construction de modèles tridimensionnels

- Dans notre approche, nous avons tout d'abord estimé la géométrie du système de caméras automatiquement puis la géométrie de la scène à l'aide de l'intervention de l'utilisateur. Nous pourrions utiliser cette

information pour estimer à nouveau la géométrie de la scène et obtenir de meilleurs résultats.

Annexe A

Les algorithmes d'estimation robuste

Une méthode d'estimation de paramètres à partir de données est « robuste » si son résultat est peu influencé par la présence de données « aberrantes », c'est-à-dire, dont la probabilité d'apparition est faible dans le cadre du modèle. Ces données aberrantes sont, soit faiblement aberrantes, mais en nombre important, soit fortement aberrantes, mais en faible nombre.

A.1 Moindre médiane des carrés

Cette méthode est fondée sur le fait que, pour une variable aléatoire dont la distribution est fortement centrée, la médiane est un estimateur de la valeur centrale plus robuste que la moyenne. Au lieu de minimiser la moyenne des carrés des erreurs, comme le fait la méthode aux moindres carrés ordinaires, elle minimise donc leur médiane :

$$\min_{\mathbf{a}} \chi^2(\mathbf{a}) \quad \text{avec} \quad \chi^2(\mathbf{a}) = \text{med}_{i \in [1, n]} \left(\frac{y_i - f_i(\mathbf{x}_i, \mathbf{a})}{\sigma_i} \right)^2 \quad (\text{A.1})$$

Comme il n'y a pas d'expression analytique de la médiane, la minimisation se fait en cherchant aléatoirement dans l'espace des solutions. Plus précisément, si q est un entier supérieur ou égal au nombre minimum de données nécessaires pour calculer \mathbf{a} , N solutions $\mathbf{a}_1, \dots, \mathbf{a}_N$ sont calculées par une méthode non robuste (moindres carrés linéaires ou non linéaires) à

partir de N ensembles de q données tirés aléatoirement dans l'ensemble des n données initiales. La solution \mathbf{a}_j qui minimise $\chi^2(\mathbf{a})$ est alors choisie. Les données telles que

$$\frac{y_i - f_i(\mathbf{x}_i, \mathbf{a}_j)}{\sigma_i} > \sigma$$

où σ est une estimation de la variance (voir [Rousseeuw et Leroy, 1987]), sont considérées comme aberrantes et rejetées. Un raffinement de la solution \mathbf{a}_j est finalement effectué en appliquant une méthode aux moindres carrés ordinaires aux données restantes.

N est choisi de telle sorte que la probabilité P que le tirage des N ensembles de q données contienne au moins un ensemble exempt de toute donnée aberrante soit très proche de 1. Nous sommes ainsi quasiment sûr qu'au moins une solution parmi les N solutions $\mathbf{a}_1, \dots, \mathbf{a}_N$ est bonne.

P se calcule comme suit : si a est le pourcentage estimé de données aberrantes présentes dans les données initiales, $1 - a$ est la probabilité qu'une donnée ne soit pas aberrante; $(1 - a)^q$ est la probabilité qu'un ensemble de q données soit exempt de toute donnée aberrante; $1 - (1 - a)^q$ est la probabilité qu'un ensemble de q données contienne au moins une donnée aberrante; $(1 - (1 - a)^q)^N$ est la probabilité qu'un tirage de N ensembles de q données contienne au moins un ensemble de q données contenant au moins une donnée aberrante; donc, $P = 1 - (1 - (1 - a)^q)^N$. N est donc donné par la formule suivante :

$$N = \frac{\ln(1 - P)}{\ln(1 - (1 - a)^q)} \quad (\text{A.2})$$

Annexe B

Compléments de géométrie projective appliquée à la vision

Cet appendice est consacré à certains détails et petits calculs de géométrie projective permettant d'exprimer des propriétés intéressantes pour le modèle sténopé.

B.1 Les plans de l'espace correspondent à des homographies de l'image

Considérons des points \mathbf{M}_i de l'espace se trouvant en configuration générale dans un même plan, et leur image \mathbf{m}_i par une relation projective linéaire de \mathcal{P}^3 vers \mathcal{P}^2 . Celle-ci dans ce cas précis se restreint à une relation entre les deux plans, qui, à titre de relation projective de \mathcal{P}^2 , est une homographie h . Si nous disposons de deux images \mathbf{m}_i et \mathbf{m}'_i des mêmes points \mathbf{M}_i de la scène, nous pouvons considérer la relation $h' \circ h^{-1}$ entre ces deux images. C'est également une relation homographique c'est à dire qu'il existe une matrice 3×3 inversible \mathbf{H} telle que nous ayons l'égalité projective

$$\mathbf{m}'_i = \mathbf{H}\mathbf{m}_i \tag{B.1}$$

Le fait que \mathbf{H} soit inversible découle aussi du fait qu'une relation du même type doit exister si nous échangeons les rôles des deux images. L'homographie inverse a pour matrice \mathbf{H}^{-1} .

Quatre points sont nécessaires pour calculer une homographie, mais seulement trois points définissent un plan. La réponse à ce paradoxe apparent est dans la matrice fondamentale. En effet, les 4 points doivent vérifier la contrainte épipolaire et donc, il existe une relation entre la matrice fondamentale et toute les homographies possibles entre les deux images. Si on reprend l'expression B.1 en la remplaçant dans l'équation épipolaire 2.12, on obtient

$$(\mathbf{H}\mathbf{m}_i)^T \mathbf{F}\mathbf{m}_i = 0 \quad (\text{B.2})$$

d'où

$$\mathbf{m}_i^T \mathbf{H}^T \mathbf{F}\mathbf{m}_i = 0 \quad (\text{B.3})$$

et ceci, pour tout \mathbf{m}_i . On en déduit alors

$$\mathbf{H}^T \mathbf{F} + \mathbf{F}^T \mathbf{H} = 0 \quad (\text{B.4})$$

Cette équation plus trois points permet de définir l'homographie.

B.2 Comment faire un barycentre projectif?

Ce qui va suivre ne constitue pas un traité mathématique, la notion de barycentre n'existant pas en géométrie projective. Toutefois, il existe des problèmes où le barycentre (ou les moyennes) sont utiles. On a par exemple deux estimations \mathbf{v}_1 et \mathbf{v}_2 d'un même vecteur, auxquelles peuvent être associées des mesures d'incertitude et l'on désire combiner ces deux estimations de manière cohérente. En géométrie classique, le vecteur estimé est un barycentre de \mathbf{v}_1 et \mathbf{v}_2 avec des poids dépendants de leur incertitudes relatives.

La formule du barycentre \mathbf{G} de N points \mathbf{M}_i de poids α_i est

$$\mathbf{G} = \frac{\sum_i \alpha_i \mathbf{M}_i}{\sum_i \alpha_i}$$

Or, la somme de vecteurs n'est pas définie en géométrie projective. Cette formule n'est donc pas directement applicable.

Revenons donc à la représentation de vecteurs projectifs. Ces vecteurs sont des classes d'équivalence qui peuvent être représentées par un de leurs éléments de norme 1. On identifie alors \mathcal{P}^n à la sphère \mathcal{S}^{n+1} où les points

antipodaux sont identifiés. Si l'on travaille en géométrie projective orientée comme au chapitre 4, alors, on identifie exactement \mathcal{T}^n avec \mathcal{S}^{n+1} . Sur la sphère, une distance qui ne dépend pas de la base choisie est la longueur de l'arc du grand cercle entre deux points. On définit le barycentre projectif comme étant le point \mathbf{G} de \mathcal{S}^{n+1} (voir figure B.1) minimisant le critère suivant

$$C = \frac{\sum_i \alpha_i \text{Arclength}(\mathbf{M}_i, \mathbf{G})}{\sum_i \alpha_i}$$

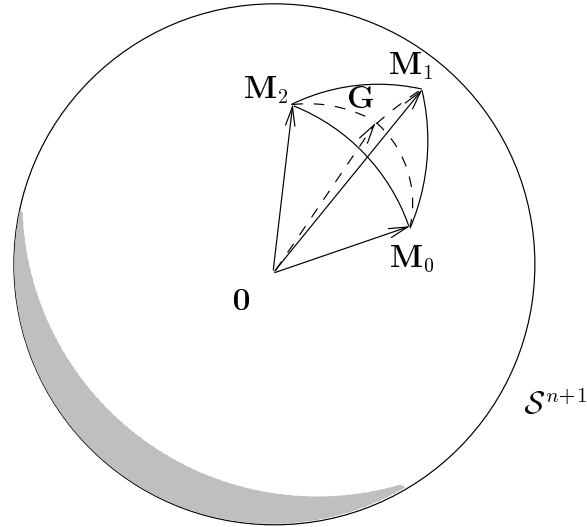


FIG. B.1 – *Le barycentre projectif. Nous présentons ici un exemple de barycentre entre trois points de \mathcal{S}^{n+1} . Les vecteurs sont tous unitaires.*

Diverses remarques peuvent être faites :

- Le point déterminé ainsi n'est pas forcément unique. En effet, considérons 2 points antipodaux. Leur point milieu peu se trouver sur tout l'équateur leur correspondant. Nous aurons cependant rarement à traiter ce cas en pratique car les points que nous devons moyenner sont des points proches les uns des autres.
- Le calcul d'un tel point résulte d'une minimisation non linéaire. En effet, si l'on dérive C par rapport à \mathbf{G} ,

$$\frac{\partial C}{\partial \mathbf{G}} = \sum_i \alpha_i \frac{\mathbf{M}_i}{\sqrt{1 - (\mathbf{M}_i \cdot \mathbf{G})^2}}$$

Le calcul des zéros de cette expression conduit à des calculs non linéaires. Le barycentre projectif n'a donc pas de formule explicite.

En dépit de ces défauts et du côté très pragmatique de sa définition, le barycentre projectif permet de résoudre les problèmes de combinaison de plusieurs estimées d'une manière plus satisfaisante que le choix aléatoire d'une des deux estimées.

B.3 Comment reconstruire des points de \mathcal{P}^3 ?

Le problème est le suivant : étant donné des matrices de projection \mathbf{P}_i pour les images i de 1 à N , et les images \mathbf{m}_i d'un point tridimensionnel \mathbf{M} inconnu, comment calculer \mathbf{M} ?

Il existe plusieurs méthodes, qui sont comparées dans [Rothwell et al., 1995] et dans [Hartley et Sturm, 1994]. Nous allons nous concentrer sur deux d'entre elles : la méthode linéaire à base de décomposition en valeurs singulières et la méthode non linéaire.

B.3.1 La méthode linéaire

Les seules équations dont on dispose sont les équations (2.7) :

$$\mathbf{m}_i = \mathbf{P}_i \mathbf{M}$$

Nous pouvons réécrire ces équations comme à la section 2.4 pour obtenir l'équation matricielle (2.26) que l'on résume sous la forme

$$\mathbf{A} \mathbf{M} = \mathbf{0}$$

où \mathbf{A} est une matrice $2N \times 4$ qui ne dépend que des vecteurs \mathbf{m}_i et des matrices \mathbf{P}_i . Si les points \mathbf{m}_i et les matrices \mathbf{P}_i engendrent des rayons qui se coupent en un point unique, ce système aura une solution unique et \mathbf{A} sera de rang trois, comme démontré à la section 2.4. Toutefois, ce système peut ne pas avoir de solution exacte si les points \mathbf{m}_i ou les matrices \mathbf{P}_i sont sujettes aux bruits ou aux erreurs. Dans ce cas, nous allons essayer de minimiser $\|\mathbf{A} \mathbf{M}\|$ sous la contrainte que $\|\mathbf{M}\| = 1$. La solution \mathbf{M} de ce problème de minimisation linéaire est le vecteur propre de norme 1 associé à la plus petite valeur propre en valeur absolue de $\mathbf{A}^T \mathbf{A}$. Ce vecteur propre est exactement le vecteur singulier associé à la plus petite valeur singulière de \mathbf{A} .

B.3.2 La méthode non linéaire

La méthode linéaire, bien que donnant des bons résultats dans la majorité des cas, souffre de quelques défauts :

- Le résultat est sensible à la multiplication par une constante d'une des matrices de projection. En effet, si l'on multiplie \mathbf{P}_i par λ , cela a

pour effet de multiplier les deux lignes de \mathbf{A} correspondantes par λ . Les valeurs singulières de \mathbf{A} ainsi que leurs vecteurs associés vont être modifiés.

- La distance minimisée ($\|\mathbf{AM}\|$) ne représente rien d'un point de vue géométrique, ce qui rend le résultat difficilement prévisible.

La solution est alors de changer le critère de minimisation pour le rendre significatif géométriquement et indépendant d'un changement d'échelle des matrices de projection. Le critère que nous avons choisi est la distance dans les images entre la projection du point \mathbf{M} suivant \mathbf{P}_i et \mathbf{m}_i . Ce critère n'est pas linéaire en fonction des données et nous devons utiliser une minimisation itérative (comme celles décrites à l'appendice 3.4.5.1) pour obtenir le minimum. Le critère s'exprime sous la forme suivante :

$$C(\mathbf{M}) = \sum_{i=0}^N \left(\frac{m_i^1}{m_i^3} - \frac{(\mathbf{P}_i \mathbf{M})^1}{(\mathbf{P}_i \mathbf{M})^3} \right)^2 + \left(\frac{m_i^2}{m_i^3} - \frac{(\mathbf{P}_i \mathbf{M})^2}{(\mathbf{P}_i \mathbf{M})^3} \right)^2$$

Une initialisation possible pour \mathbf{M} est le résultat de la méthode linéaire.

Annexe C

Calculs relatifs à l'ajustement de rayons

Nous cherchons ici à calculer les matrices \mathbf{U}_i , \mathbf{V}_j , et \mathbf{W}_{ij} en fonction de la position courante.

Calculons tout d'abord $\frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{M}_j}$. On a les formules

$$\begin{aligned}\hat{m}_{ij}^1 &= \frac{\mathbf{P}_i^1 \mathbf{M}}{\mathbf{P}_i^3 \mathbf{M}} \\ \hat{m}_{ij}^2 &= \frac{\mathbf{P}_i^2 \mathbf{M}}{\mathbf{P}_i^3 \mathbf{M}}\end{aligned}$$

où \mathbf{P}_i^l représente la l -ième ligne de la matrice \mathbf{P}_i . En dérivant par rapport à \mathbf{M} , on obtient

$$\begin{aligned}\frac{\partial \hat{m}_{ij}^1}{\partial \mathbf{M}_j} &= \frac{(\mathbf{P}_i^3 \mathbf{M}) \mathbf{P}_i^1 - (\mathbf{P}_i^1 \mathbf{M}) \mathbf{P}_i^3}{(\mathbf{P}_i^3 \mathbf{M})^2} \\ \frac{\partial \hat{m}_{ij}^2}{\partial \mathbf{M}_j} &= \frac{(\mathbf{P}_i^3 \mathbf{M}) \mathbf{P}_i^2 - (\mathbf{P}_i^2 \mathbf{M}) \mathbf{P}_i^3}{(\mathbf{P}_i^3 \mathbf{M})^2}.\end{aligned}$$

On calcule de même $\frac{\partial \hat{\mathbf{m}}_{ij}}{\partial \mathbf{P}_i}$. On a les formules

$$\hat{m}_{ij}^1 = \frac{\mathbf{P}_i^1 \mathbf{M}}{\mathbf{P}_i^3 \mathbf{M}}$$

$$\hat{m}_{ij}^2 = \frac{\mathbf{P}_i^2 \mathbf{M}}{\mathbf{P}_i^3 \mathbf{M}}$$

Nous décomposerons le calcul en deux parties. La première partie est le calcul de la dérivée de $\hat{\mathbf{m}}_{ij}$ par rapport à un élément d'une des deux premières lignes de \mathbf{P}_i , la deuxième traitant la dérivation par rapport à un élément de la troisième ligne.

Pour les deux premières lignes ($l = 1, 2$), on a simplement :

$$\frac{\partial \hat{\mathbf{m}}_{ij}^k}{\partial \mathbf{P}_i^l} = \delta_{kl} \frac{\mathbf{M}}{\mathbf{P}_i^3 \mathbf{M}}$$

Pour la troisième,

$$\frac{\partial \hat{\mathbf{m}}_{ij}^k}{\partial \mathbf{P}_i^3} = -\frac{(\mathbf{P}_i^k \mathbf{M}) \mathbf{P}_i^3}{(\mathbf{P}_i^3 \mathbf{M})^2}$$

De ces expressions, on peut déduire celles de \mathbf{U}_i , \mathbf{V}_j et \mathbf{W}_{ij} .

Annexe D

Notations utilisées

- \neg : l'opposition en géométrie projective orientée.
- $\mathbf{a} \times \mathbf{b}$: le produit vectoriel de \mathbf{a} par \mathbf{b} .
- $[\mathbf{a}]_{\times}$: la matrice associée au produit vectoriel vérifiant $\forall \mathbf{m}, [\mathbf{a}]_{\times} \mathbf{m} = \mathbf{a} \times \mathbf{m}$.
- A : la matrice des paramètres intrinsèques.
- \mathcal{A}^n : l'espace affine de dimension n .
- α_u, α_v : les facteurs d'échelle pixels.
- c : le point principal.
- C : le centre optique.
- \mathbf{e}_{ij} : l'épipole de l'image i générée par l'image j
- \mathbf{E}_{ij} : la matrice essentielle entre les images i et j .
- f : la distance focale.
- \mathbf{F}_{ij} : la matrice fondamentale entre les images i et j .
- \mathbf{G} : le barycentre de plusieurs points.
- \mathbf{H} : une homographie, c'est à dire une relation linéaire projective.

- \mathbf{I} : la matrice identité.
- λ : une droite dans les images.
- Λ : une droite dans l'espace.
- \mathbf{m} : un point des images.
- \mathbf{M} : un point de l'espace.
- \mathbf{P}_i : la matrice de projection de la caméra i .
- \mathcal{P}^n : l'espace projectif de dimension n .
- Π^r : le plan rétinien.
- Π^f : le plan focal.
- Π_∞ : le plan à l'infini.
- R^n : l'espace euclidien de dimension n .
- T : le tenseur trifocal.
- \mathcal{T}^n : l'espace projectif orienté de dimension n .
- u_0, v_0 : les coordonnées du point principal.

Sauf indication contraire, toutes les équations vectorielles et matricielles de cet ouvrage s'entendent à un facteur multiplicatif près. Les numéros d'images sont en indices tandis que les numéros de coordonnées sont en exposant.

Bibliographie

- [Aggarwal et Nandhakumar, 1988] Aggarwal, J. et Nandhakumar, N. (1988). On the computation of motion from sequences of images — a review. *Proc. IEEE*, 76(8):917–935.
- [Aloimonos et al., 1988] Aloimonos, J., Weiss, I., et Bandopadhyay, A. (1988). Active vision. *The International Journal of Computer Vision*, 1(4):333–356.
- [Anandan et al., 1994] Anandan, P., Burt, P., Dana, K., Hansen, M., et van der Wal, G. (1994). Real-time scene stabilization and mosaic construction. In *Proceedings of the ARPA Image Understanding Workshop*, pages 1:457–465. Defense Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc.
- [Aschwanden et Guggenbühl, 1992] Aschwanden, P. et Guggenbühl, W. (1992). Experimental results from a comparative study on correlation-type registration algorithms. In *ISPRS Workshop*, Bonn, Germany.
- [Ayache et Lustman, 1987a] Ayache, N. et Lustman, F. (1987a). Fast and reliable passive stereovision using three cameras. In *Proceedings of International Workshop of Industrial Applications of Machine Vision and Machine Intelligence, Tokyo*. IEEE.
- [Ayache et Lustman, 1987b] Ayache, N. et Lustman, F. (1987b). Trinocular stereovision: Recent results. In *Proceedings of the 10th International Joint Conference on Artificial Intelligence*, Milan, Italy.

- [Barnabei et al., 1985] Barnabei, M., Brini, A., et Rota, G.-C. (1985). On the exterior calculus of invariant theory. *Journal of Algebra*, 96:pp 120–160.
- [Barrett et al., 1992] Barrett, E. B., Brill, M. H., Haag, N. N., et Payton, P. M. (1992). Invariant linear methods in photogrammetry and model-matching. In Mundy, J. L. et Zimmerman, A., editors, *Geometric Invariance in Computer Vision*, chapter 14. MIT Press.
- [Bascle et al., 1994] Bascle, B., Bouthemy, P., Deriche, R., et Meyer, F. (1994). Tracking complex primitives in an image sequence. In *Proceedings of the 12th IAPR International Conference On Pattern Recognition*, pages 426–431, Jerusalem, Israel.
- [Basri, 1993] Basri, R. (1993). On the uniqueness of correspondence under orthographic and perspective projections. In *Proceedings of the ARPA Image Understanding Workshop*, pages 875–884, Washington, DC. Defense Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc.
- [Beardsley et al., 1992] Beardsley, P., Murray, D., et Zissermann, A. (1992). Camera calibration using multiple images. In Sandini, G., editor, *Proceedings of the 2nd European Conference on Computer Vision*, volume 588 of *Lecture Notes in Computer Science*, pages 312–320, Santa Margherita Ligure, Italy. Springer-Verlag.
- [Beyer, 1992] Beyer, H. A. (1992). Accurate calibration of CCD-cameras. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, Urbana Champaign, IL. IEEE.
- [Blaszka et Deriche, 1994] Blaszka, T. et Deriche, R. (1994). Recovering and characterizing image features using an efficient model based approach. Technical Report 2422, INRIA.
- [Blaszka et Deriche, 1995] Blaszka, T. et Deriche, R. (1995). A model based method for characterization and location of curved image features. In *Proceedings of the 8th ICIAP*, San-Remo, Italy.
- [Brand et al., 1993] Brand, P., Mohr, R., et Bobet, P. (1993). Distorsions optiques : correction dans un modèle projectif. Technical Report 1933, LIFIA–INRIA Rhône-Alpes.
- [Brown, 1976] Brown, D. (1976). The bundle adjustment - progress and prospect. In *XIII Congress of the ISPRS*, Helsinki.

- [Brown, 1958] Brown, D. C. (1958). A solution to the general problem of multiple station analytical stereotriangulation. Technical Report 43, RCA Data Reduction Technical Report, Patrick Air Force base, Florida.
- [Brown, 1971] Brown, D. C. (1971). Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866.
- [Brown, 1992] Brown, L. (1992). A survey of image registration techniques. *ACM Computing Surveys*, 24(4):325–376.
- [Canny, 1986] Canny, J. F. (1986). A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):769–798.
- [Carlsson, 1994] Carlsson, S. (1994). Multiple image invariance using the double algebra. In Mundy, J. L., Zissermann, A., et Forsyth, D., editors, *Applications of Invariance in Computer Vision*, volume 825 of *Lecture Notes in Computer Science*, pages 145–164. Springer-Verlag.
- [Crowley et Stelmaszyk, 1990] Crowley, J. et Stelmaszyk, P. (1990). Measurement and integration of 3-D structures by tracking edge lines. In Faugeras, O., editor, *Proceedings of the 1st European Conference on Computer Vision*, pages 269–280, Antibes, France. Springer, Berlin, Heidelberg.
- [Csurka, 1996] Csurka, G. (1996). *Modelisation projective des objets tridimensionnels en vision par ordinateur*. PhD thesis, Université de Nice Sophia Antipolis.
- [Das et Ahuja, 1993] Das, S. et Ahuja, N. (1993). A comparative study of stereo, vergence and focus as depth cues for active vision. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 194–199, New-York, NY. IEEE Computer Society, IEEE.
- [Deriche, 1987] Deriche, R. (1987). Using canny's criteria to derive a recursively implemented optimal edge detector. *The International Journal of Computer Vision*, 1(2):167–187.
- [Deriche et Faugeras, 1990] Deriche, R. et Faugeras, O. (1990). Tracking line segments. *Image and Vision Computing*, 8(4):261–270. A shorter version appeared in the Proceedings of the 1st ECCV.
- [Deriche et Giraudon, 1993] Deriche, R. et Giraudon, G. (1993). A computational approach for corner and vertex detection. *The International Journal of Computer Vision*, 10(2):101–124.

- [Devernay, 1993] Devernay, F. (1993). A fast and efficient subpixelic edge detector. In *Quatrièmes Journées Orasis*, Mulhouse, France.
- [Devernay et Faugeras, 1994a] Devernay, F. et Faugeras, O. (1994a). Computing differential properties of 3-D shapes from stereoscopic images without 3-D models. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 208–213, Seattle, WA. IEEE.
- [Devernay et Faugeras, 1995] Devernay, F. et Faugeras, O. (1995). Automatic calibration and removal of distortion from scenes of structured environments. In Rudin, L. I. et Bramble, S. K., editors, *Investigative and Trial Image Processing*, volume 2567 of *Proc. SPIE*, San Diego, CA. SPIE.
- [Devernay et Faugeras, 1994b] Devernay, F. et Faugeras, O. D. (1994b). Computing differential properties of 3-D shapes from stereoscopic images without 3-D models. Technical Report 2304, INRIA.
- [Dhond et Aggarwal, 1989] Dhond, U. R. et Aggarwal, J. (1989). Structure from stereo - a review. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1489–1510.
- [Dieudonné, 1969] Dieudonné, J. (1969). Annexe. In *Éléments d'analyse*, Collection Cahiers scientifiques. Fasc. XXVIII. Gauthier-Villars, Paris.
- [Enciso et al., 1994] Enciso, R., Viéville, T., et Faugeras, O. (1994). Approximation du changement de focale et de mise au point par une transformation affine à trois paramètres. *Traitement du Signal*, 11(5).
- [Enriques, 1930] Enriques, F. (1930). *Leçons de géométrie projective*. Gauthier-Villars, Paris. Reprinted by Jacques Gabay in 1993.
- [Faugeras, 1992] Faugeras, O. (1992). What can be seen in three dimensions with an uncalibrated stereo rig. In Sandini, G., editor, *Proceedings of the 2nd European Conference on Computer Vision*, volume 588 of *Lecture Notes in Computer Science*, pages 563–578, Santa Margherita Ligure, Italy. Springer-Verlag.
- [Faugeras, 1993] Faugeras, O. (1993). *Three-Dimensional Computer Vision: a Geometric Viewpoint*. The MIT Press.
- [Faugeras, 1995] Faugeras, O. (1995). Stratification of 3-D vision: projective, affine, and metric representations. *Journal of the Optical Society of America A*, 12(3):465–484.

- [Faugeras et Berthod, 1980] Faugeras, O. et Berthod, M. (1980). Scene labelling: an optimization approach. *Pattern Recognition*, 12:339–347.
- [Faugeras et Berthod, 1981] Faugeras, O. et Berthod, M. (1981). Improving consistency and reducing ambiguity in stochastic labelling: An optimization approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3:412–424.
- [Faugeras et Laveau, 1994] Faugeras, O. et Laveau, S. (1994). Representing three-dimensional data as a collection of images and fundamental matrices for image synthesis. In *Proceedings of the International Conference on Pattern Recognition*, pages 689–691, Jerusalem, Israel. Computer Society Press.
- [Faugeras et Mourrain, 1995] Faugeras, O. et Mourrain, B. (1995). On the geometry and algebra of the point and line correspondences between n images. In *Proceedings of the 5th Proc. International Conference on Computer Vision*, pages 951–956, Boston, MA. IEEE Computer Society Press.
- [Faugeras et Robert, 1994] Faugeras, O. et Robert, L. (1994). What can two images tell us about a third one? *The International Journal of Computer Vision*. to appear, also INRIA Technical report 2018.
- [Forstner et Pertl, 1986] Forstner, W. et Pertl, A. (1986). Photogrammetric standard methods and digital image matching techniques for high precision surface measurements. In Gelsema, E.S. and Kanal, L.N., editor, *Pattern Recognition in Practice II*, pages 57–72. Elsevier Science Publishers.
- [Gambotto, 1989] Gambotto, J. (1989). Tracking points and line segments in image sequences. In *Proc. IEEE Workshop Visual Motion*, pages 38–45, Irvine, CA. IEEE.
- [Giai-Checa et al., 1993] Giai-Checa, B., Deriche, R., Viéville, T., et O.Faugeras (1993). Suivi de segments dans une séquence d’images monoculaire. Technical Report 2113, INRIA Sophia-Antipolis, France.
- [Goshtasby et al., 1984] Goshtasby, A., Gage, S. H., et Bartholic, J. F. (1984). A two-stage cross correlation approach to template matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(3):374–378.
- [Grimson et al., 1994] Grimson, W. E., Huttenlocher, D. P., et Jacobs, D. W. (1994). A study of affine matching with bounded sensor error. *The International Journal of Computer Vision*, 13(1):7–32.

- [Gruen, 1978] Gruen, A. (1978). Accuracy, reliability and statistics in close-range photogrammetry. In *Proceedings of the Symposium of the ISP Commission V*, Stockholm.
- [Gruen et Beyer, 1992] Gruen, A. et Beyer, H. A. (1992). System calibration through self-calibration. In *Proceedings of the Workshop on Calibration and Orientation of Cameras in Computer Vision*, Washington D.C.
- [Gurewitz et al., 1986] Gurewitz, E., Dinstein, I., et Sarusi, S. (1986). More on the benefit of a third eye for machine stereo perception. In *International Conference on Pattern Recognition*, pages 966–968. IEEE. Paris, France.
- [Hansen et al., 1988] Hansen, C., Ayache, N., et Lustman, F. (1988). Towards real-time trinocular stereo. In *Proceedings of the 2nd Proc. International Conference on Computer Vision*, pages 129–133, Tampa, FL. IEEE Computer Society Press.
- [Harris, 1987] Harris, C. (1987). Determination of ego-motion from matched points. In *Proceedings of the 3rd Alvey Conference, Cambridge*, pages 189–192.
- [Harris et Stephens, 1988] Harris, C. et Stephens, M. (1988). A combined corner and edge detector. In *Proc. 4th Alvey Vision Conf.*, pages 189–192.
- [Hartley, 1994a] Hartley, R. (1994a). Lines and points in three views—an integrated approach. In *Proceedings of the ARPA Image Understanding Workshop*. Defense Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc.
- [Hartley, 1994b] Hartley, R. (1994b). Projective reconstruction from line correspondences. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 903–907, Seattle, WA. IEEE.
- [Hartley, 1994c] Hartley, R. (1994c). Self-calibration from multiple views with a rotating camera. In Eklundh, J.-O., editor, *Proceedings of the 3rd European Conference on Computer Vision*, volume 800-801 of *Lecture Notes in Computer Science*, pages 471–478, Stockholm, Sweden. Springer-Verlag.
- [Hartley et Sturm, 1994] Hartley, R. et Sturm, P. (1994). Triangulation. In *Proceedings of the ARPA Image Understanding Workshop*, pages 957–966. Defense Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc.

- [Hartley, 1993a] Hartley, R. I. (1993a). Cheirality invariants. In *Proceedings of the ARPA Image Understanding Workshop*, pages 745–753, Washington, DC. Defense Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc.
- [Hartley, 1993b] Hartley, R. I. (1993b). Invariants of lines in space. In *Proceedings of the ARPA Image Understanding Workshop*, pages 737–744, Washington, DC. Defense Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc.
- [Hartley, 1996] Hartley, R. I. (1996). Cheirality. *International Journal of Computer Vision*. to appear.
- [Hesse, 1863] Hesse, O. (1863). Die cubische Gleichung, von welcher die Lösung des Problems der Homographie von M. Chasles abhängt. *J. reine angew. Math.*, 62:188–192.
- [Horaud et al., 1994] Horaud, R., Christy, S., et Dornaika, F. (1994). Object pose: The link between weak perspective, para perspective, and full perspective. Technical Report 2356, INRIA.
- [Hotz, 1991] Hotz, B. (1991). Etude de techniques de stéréovision par corrélation - application au programme véhicule autonome planétaire (V.A.P.). Rapport de stage de D.E.A TE/AE/SE/SR No 91/242, Robotique et Vision Artificielle, Université de Nice.
- [Huang et Netravali, 1994] Huang, T. et Netravali, A. (1994). Motion and structure from feature correspondences: A review. *Proc. IEEE*, 82(2):252–268.
- [Huang et Faugeras, 1989] Huang, T. S. et Faugeras, O. D. (1989). Some properties of the E matrix in two-view motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(12):1310–1312.
- [Hummel et Zucker, 1983] Hummel, R. et Zucker, S. (1983). On the foundations of relaxation labeling processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5:267–287.
- [Irani et al., 1995] Irani, M., Anandan, P., et Hsu, S. (1995). Mosaic based representations of video sequences and their applications. In *Proceedings of the 5th Proc. International Conference on Computer Vision*, pages 605–611, Boston, MA. IEEE Computer Society Press.

- [Ito et Ishii, 1986a] Ito, M. et Ishii, A. (1986a). Range and shape measurements using three-view stereo analysis. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 9–14. IEEE. Miami Beach, Florida.
- [Ito et Ishii, 1986b] Ito, M. et Ishii, A. (1986b). Three-view stereo analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(4):524–532.
- [Jerian et Jain, 1991] Jerian, C. et Jain, R. (1991). Structure from motion. a critical analysis of methods. *IEEE Transactions on Systems, Man, and Cybernetics*, 21(3):572–587.
- [Kahl et al., 1980] Kahl, D., Rosenfeld, A., et Danker, A. (1980). Some experiments in point pattern matching. *IEEE Trans. SMC*, 10:105–116.
- [Kanade et al., 1995] Kanade, T., Narayanan, P. J., et Rander, P. (1995). Virtualized reality: Concepts and early results. In *IEEE Workshop on Representation of Visual Scenes*. IEEE.
- [Kanade et Okutumi, 1990] Kanade, T. et Okutumi, M. (1990). A stereo matching algorithm with an adaptative window: Theory and experiments. In *Proceedings of the ARPA Image Understanding Workshop*, PA. Defense Advanced Research Projects Agency, Science Applications International Corporation.
- [Kanatani, 1991] Kanatani, K. (1991). Computational projective geometry. *CVGIP: Image Understanding*, 54(3):333–348.
- [Kanatani, 1992] Kanatani, K. (1992). *Geometric computation for machine vision*. Oxford university press.
- [Kitamura et Yachida, 1990] Kitamura, Y. et Yachida, M. (1990). Three-dimensional data acquisition by trinocular vision. *Advanced Robotics*, 4(1):29–42. Robotics Society of Japan.
- [Krotkov et Bacjcsy, 1993] Krotkov, E. et Bacjcsy, R. (1993). Active vision for reliable ranging: Cooperating focus, stereo and vergence. *The International Journal of Computer Vision*, 11(2):187–203.
- [Krotkov, 1989] Krotkov, E. P. (1989). *Active Computer Vision by Cooperative Focus and Stereo*. Springer-Verlag.

- [Laveau et Faugeras, 1994] Laveau, S. et Faugeras, O. (1994). 3-D scene representation as a collection of images and fundamental matrices. Technical Report 2205, INRIA.
- [Lee et al., 1995] Lee, M., Medioni, G., et Deriche, R. (1995). Structure and motion from a sparse set of views. In *IEEE International Symposium on Computer Vision*, Biltmore Hotel, Coral Gables, Florida, USA.
- [Lenz et Tsai, 1987] Lenz, R. et Tsai, R. (1987). Techniques for calibrating of the scale factor and image center for high accuracy 3D machine vision metrology. In *International Conference on Robotics and Automation*, pages 68–75, Raleigh, NC.
- [Lenz et Tsai, 1988] Lenz, R. K. et Tsai, R. Y. (1988). 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*, 10:713–720.
- [Li, 1990] Li, S. Z. (1990). Invariant surface segmentation through energy minimization with discontinuities. *The International Journal of Computer Vision*, 5(2):161–194.
- [Longuet-Higgins, 1981] Longuet-Higgins, H. (1981). A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135.
- [Luong, 1992] Luong, Q.-T. (1992). *Matrice Fondamentale et Calibration Visuelle sur l'Environnement-Vers une plus grande autonomie des systèmes robotiques*. PhD thesis, Université de Paris-Sud, Centre d'Orsay.
- [Luong et Viéville, 1994] Luong, Q.-T. et Viéville, T. (1994). Canonic representations for the geometries of multiple projective views. In Eklundh, J.-O., editor, *Proceedings of the 3rd European Conference on Computer Vision*, volume 800-801 of *Lecture Notes in Computer Science*, pages 589–599, Vol. 1, Stockholm, Sweden. Springer-Verlag.
- [Lustman, 1988] Lustman, F. (1988). *Vision Stéréoscopique Et Perception Du Mouvement En Vision Artificielle*. PhD thesis, Université de Paris-Sud Centre d'Orsay.
- [Ma et Olsen, 1990] Ma, J. et Olsen, S. (1990). Depth from zooming. *Journal of the Optical Society of America*, 7(10):1883–1890.
- [Maître et Wu, 1987] Maître, H. et Wu, Y. (1987). Improving dynamic programming to solve image registration. *Pattern Recog.*, 20(4):443–462.

- [Marr et Poggio, 1977] Marr, D. et Poggio, T. (1977). A theory of human stereo vision. Memorandum AIM 451, MIT Artif. Intell. Lab.
- [McMillan, 1995] McMillan, L. (1995). Acquiring immersive visual environments with an uncalibrated camera. Technical Report TR95-006, University of North Carolina.
- [McMillan et Bishop, 1995] McMillan, L. et Bishop, G. (1995). Plenoptic modeling: An image-based rendering system. In *SIGGRAPH*, Los Angeles, CA.
- [Milenkovic et Kanade, 1985] Milenkovic, V. et Kanade, T. (1985). Trinocular vision using photometric and edge orientation constraints. In Baumann, L., editor, *Proceedings of the ARPA Image Understanding Workshop*, pages 163–175, Miami Beach, FL. Defense Advanced Research Projects Agency, Morgan Kaufmann Publishers, Inc.
- [Mourrain et Stolfi, 1994] Mourrain, B. et Stolfi, N. (1994). Computational symbolic geometry. In N., W., editor, *Invariants methods in Discrete and Computational Geometry*, chapter 1. Kluwer acad. pub. to appear.
- [Mundy et Zisserman, 1992] Mundy, J. L. et Zisserman, A., editors (1992). *Geometric Invariance in Computer Vision*. MIT Press.
- [Netravali et al., 1989] Netravali, A., Huang, T., Krishnakumar, A., et Holt, R. (1989). Algebraic methods in 3-D motion estimation from two-view point correspondences. *Int. J. Imaging Syst. Tech.*, 1:78–99.
- [Nishihara et Poggio, 1983] Nishihara, H. et Poggio, T. (1983). Stereo vision for robotics. In *ISRR83 Conference*, Bretton Woods, New Hampshire.
- [Nishihara et Poggio, 1984] Nishihara, H. et Poggio, T. (1984). Stereo vision for robotics. In Brady, M. et Paul, R., editors, *Robotics research*, pages 489–505. MIT Press.
- [Noble, 1988] Noble, J. (1988). Finding corners. *Image and Vision Computing*, 6:121–128.
- [Ohta et Kanade, 1985] Ohta, Y. et Kanade, T. (1985). Stereo by intra- and inter-scanline search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7:139–154.
- [Ohta et al., 1986] Ohta, Y., Watanabe, M., et Ikeda, K. (1986). Improving depth map by right-angled trinocular stereo. In *International Conference on Pattern Recognition*, pages 519–521. IEEE. Paris, France.

- [Okamoto, 81] Okamoto, A. (81). Orientation and construction of models, part i: The orientation problem in close-range photogrammetry. *Photogrammetric Engineering and remote Sensing*, 47(10):1437–1454.
- [Okamoto, 84] Okamoto, A. (84). The model construction problem using the collinearity condition. *Photogrammetric Engineering and remote Sensing*, L(6):705–711.
- [Okutomi et Kanade, 1991] Okutomi, M. et Kanade, T. (1991). A multiple-baseline stereo. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 63–69, Lahaina, Hawaii. IEEE.
- [Pahlavan et al., 1993] Pahlavan, K., Uhlin, T., et Eklundh, J. (1993). Active vision as methodology. In Aloimonos, Y., editor, *Active Perception*, Computer Vision, chapter 1, pages 19–46. L. Erlbaum Associates.
- [Pietikainen et Harwood, 1986] Pietikainen, M. et Harwood, D. (1986). Depth from three-camera stereo. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 2–8. IEEE. Miami Beach, Florida.
- [Pietikainen et Harwood, 1987] Pietikainen, M. et Harwood, D. (1987). Progress in trinocular stereo. In *Proceedings NATO Advanced Workshop on Real-time Object and Environment Measurement and classification, Maratea, Italy*.
- [Poelman et Kanade, 1993] Poelman, C. J. et Kanade, T. (1993). A paraperspective factorization method for shape and motion recovery. Technical Report CMU-CS-93-219, Carnegie Mellon University, School of Computer Science.
- [Poelman et Kanade, 1994] Poelman, C. J. et Kanade, T. (1994). A paraperspective factorization for shape and motion recovery. In Eklundh, J.-O., editor, *Proceedings of the 3rd European Conference on Computer Vision*, volume B of *Lecture Notes in Computer Science*, pages 97–108, Stockholm, Sweden. Springer-Verlag.
- [Pollard et al., 1985] Pollard, S., Mayhew, J., et Frisby, J. (1985). PMF : a stereo correspondence algorithm using a disparity gradient constraint. *Perception*, 14:449–470.
- [Preparata et Shamos, 1985] Preparata, F. et Shamos, M. (1985). *Computational Geometry*. Springer-Verlag, New-York.

- [Robert, 1993] Robert, L. (1993). *Perception stéréoscopique de courbes et de surfaces tridimensionnelles. Applications à la robotique mobile*. PhD thesis, École Polytechnique, Paris, France.
- [Robert et Faugeras, 1991] Robert, L. et Faugeras, O. (1991). Curve-based stereo: Figural continuity and curvature. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 57–62, Lahaina, Hawaii. IEEE.
- [Robert et Faugeras, 1995] Robert, L. et Faugeras, O. (1995). Relative 3-D positioning and 3-D convex hull computation from a weakly calibrated stereo pair. *Image and Vision Computing*, 13(3):189–197. also INRIA Technical Report 2349.
- [Rohr, 1992a] Rohr, K. (1992a). Modelling and identification of characteristic intensity variations. *Image and Vision Computing*, 10(2):66–76.
- [Rohr, 1992b] Rohr, K. (1992b). Recognizing corners by fitting parametric models. *The International Journal of Computer Vision*, 9(3):213–230.
- [Rosenfeld et al., 1976] Rosenfeld, A., Hummel, R., et Zucker, S. (1976). Scene labeling by relaxation operators. *IEEE Transactions on Systems, Man, and Cybernetics*, 6:420–433.
- [Rothwell, 1995] Rothwell, C. (1995). *Object recognition through invariant indexing*. Oxford University Science Publications. Oxford University Press.
- [Rothwell et al., 1995] Rothwell, C., Csurka, G., et Faugeras, O. (1995). A comparison of projective reconstruction methods for pairs of views. In *Proceedings of the 5th Proc. International Conference on Computer Vision*, pages 932–937, Boston, MA. IEEE Computer Society Press.
- [Rothwell et al., 1993] Rothwell, C. A., Forsyth, D. A., Zisserman, A., et Mundy, J. (1993). Extracting projective structure from single perspective views of 3D point sets. In *Proceedings of the 4th Proc. International Conference on Computer Vision*, pages 573–582, Berlin, Germany. IEEE Computer Society Press.
- [Rousseeuw et Leroy, 1987] Rousseeuw, P. et Leroy, A. (1987). *Robust Regression and Outlier Detection*. John Wiley & Sons, New York.
- [Semple et Kneebone, 1952] Semple, J. et Kneebone, G. (1952). *Algebraic Projective Geometry*. Oxford: Clarendon Press. Reprinted 1979.

- [Sethi et Jain, 1987] Sethi, I. et Jain, R. (1987). Finding trajectories of Feature Points in a Monocular Image Sequence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(1):56–73.
- [Shashua, 1994a] Shashua, A. (1994a). Algebraic functions for recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Use shashua:95.
- [Shashua, 1994b] Shashua, A. (1994b). Trilinearity in visual recognition by alignment. In Eklundh, J.-O., editor, *Proceedings of the 3rd European Conference on Computer Vision*, volume 800-801 of *Lecture Notes in Computer Science*, pages 479–484, Stockholm, Sweden. Springer-Verlag.
- [Slama, 1980] Slama, C. C., editor (1980). *Manual of Photogrammetry*. American Society of Photogrammetry, fourth edition.
- [Spacek, 1985] Spacek, L. A. (1985). *The Detection of Contours and their Visual Motion*. PhD thesis, University of Essex at Cochester.
- [Springer, 1964] Springer, C. (1964). *Geometry and Analysis of Projective Spaces*. Freeman.
- [Stolfi, 1991] Stolfi, J. (1991). *Oriented Projective Geometry, A Framework for Geometric Computations*. Academic Press, Inc., 1250 Sixth Avenue, San Diego, CA.
- [Sturm, 1869] Sturm, R. (1869). Das problem der projektivität und seine anwendung auf die flächen zweiten grades. *Math. Ann.*, 1:533–574.
- [Sturmfels, 1993] Sturmfels, B. (1993). *Algorithms in Invariant Theory*. Springer-Verlag Wien New york.
- [Szeliski, 1994] Szeliski, R. (1994). Image mosaicing for tele-reality applications. Technical Report CRL94/2, DEC-CRL.
- [Tomasi et Kanade, 1991] Tomasi, C. et Kanade, T. (1991). Factoring image sequences into shape and motion. In *Proc. IEEE Workshop on Visual Motion*, pages 21–28, Princeton, NJ.
- [Torr, 1995] Torr, P. (1995). *Motion Segmentation and Outlier Detection*. PhD thesis, Department of Engineering Science, University of Oxford.
- [Torr et Murray, 1992] Torr, P. et Murray, D. (1992). Statistical detection of independent movement from a moving camera. In *British Machine Vision Conference*, pages 79–88, Leeds, UK.

- [Torr et Murray, 1993] Torr, P. et Murray, D. (1993). Outlier detection and motion segmentation. In S, editor, *Sensor Fusion VI, SPIE Vol.2059*, pages 432–443, Boston.
- [Torr et al., 1995] Torr, P., Zisserman, A., et Maybank, S. (1995). Robust detection of degenerate configurations for the fundamental matrix. In *Proceedings of the 5th Proc. International Conference on Computer Vision*, pages 1037–1042, Boston, MA. IEEE Computer Society Press.
- [Triggs, 1994] Triggs, B. (1994). The geometry of projective reconstruction I: Matching constraints and the joint image. Technical report, LIFIA.
- [Tsai, 1986] Tsai, R. (1986). Multiframe image point matching and 3d surface reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5:159–174.
- [Tsai, 1989] Tsai, R. (1989). Synopsis of recent progress on camera calibration for 3D machine vision. In Khatib, O., Craig, J. J., et Lozano-Pérez, T., editors, *The Robotics Review*, pages 147–159. MIT Press.
- [Tsai, 1987] Tsai, R. Y. (1987). 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*, 3(4):323–344.
- [Ullman et Basri, 1991] Ullman, S. et Basri, R. (1991). Recognition by linear combinations of models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(10):992–1006.
- [Vaillant, 1990] Vaillant, R. (1990). *Géométrie différentielle et vision par ordinateur: détection et reconstruction de contours d'occultation de la surface d'un objet non-polyédrique*. PhD thesis, University of Paris XI, Orsay. in french.
- [Viéville, 1995] Viéville, T. (1995). *A few steps towards 3D Active Vision*. Springer Series in Information Sciences. to appear.
- [Viéville et al., 1994] Viéville, T., Clergue, E., Enciso, R., et Mathieu, H. (1994). Experimenting 3-D vision on a robotic head. In *Proceedings of the International Conference on Pattern Recognition*, pages 739–743, Jerusalem, Israel. Computer Society Press.
- [Viéville et al., 1995] Viéville, T., Luong, Q., et Faugeras, O. (1995). Motion of points and lines in the uncalibrated case. *International Journal of Computer Vision*, 17(1).

- [Weng et al., 1992] Weng, J., Cohen, P., et Rebibo, N. (1992). Motion and structure estimation from stereo image sequences. *IEEE Trans. RA*, 8(3):362–382.
- [Willson, 1994] Willson, R. (1994). *Modeling and Calibration of Automated Zoom Lenses*. PhD thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University.
- [Willson et Shafer, 1993] Willson, R. et Shafer, S. (1993). What is the center of the image? In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 670–671, New-York, NY. IEEE Computer Society, IEEE.
- [Zeller, 1996] Zeller, C. (1996). *Calibration Projective Affine et Euclidienne en Vision par Ordinateur*. PhD thesis, École Polytechnique.
- [Zeller, 1952] Zeller, M. (1952). *Textbook of Photogrammetry*. H.K. Lewis & Company: London.
- [Zhang, 1993] Zhang, Z. (1993). Recalage de deux nuages de points 3D. *Traitement du Signal*, 10(4):263–281.
- [Zhang, 1994] Zhang, Z. (1994). Token tracking in a cluttered scene. *Int'l J. of Image and Vision Computing*, 12(2):110–120. also Research Report No.2072, INRIA Sophia-Antipolis, 1993.
- [Zhang et al., 1994] Zhang, Z., Deriche, R., Faugeras, O., et Luong, Q.-T. (1994). A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence Journal*, 78(1-2):87–119. Appeared in October 1995, also INRIA Research Report No.2273, May 1994.
- [Zhang et Faugeras, 1992] Zhang, Z. et Faugeras, O. (1992). Estimation of displacements from two 3D frames obtained from stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(12):1141–1156.
- [Zoghliami, 1995] Zoghliami, I. (1995). Traitement des occlusions pour la modification d'objet dans une séquence d'image. rapport de dea, INRIA.
- [Zucker et al., 81] Zucker, S., Leclerc, Y., et Mohammed, J. (81). Continuous relaxation and local maxima selection: Conditions for equivalence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3:117–127.