



Méthodes robustes en vision : application aux appariements visuels

Zhong-Dan Lan

► To cite this version:

Zhong-Dan Lan. Méthodes robustes en vision : application aux appariements visuels. Interface homme-machine [cs.HC]. Institut National Polytechnique de Grenoble - INPG, 1997. Français. NNT : . tel-00004950

HAL Id: tel-00004950

<https://theses.hal.science/tel-00004950>

Submitted on 20 Feb 2004

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 par

Zhong-Dan LAN

pour obtenir le grade de **DOCTEUR**

de l'**INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE**

Spécialité : **Informatique**

**MÉTHODES ROBUSTES EN VISION :
APPLICATION AUX APPARIEMENTS
VISUELS**

Date de soutenance : Mai 1997

Composition du jury :

Président :	James Crowley
Rapporteurs :	Peter Sander Jun Shen
Examineurs :	Long Quan Roger Mohr

Thèse préparée au sein du laboratoire GRAVIR - IMAG - INRIA Rhône - Alpes
sous la direction de Roger MOHR

A mes parents

Remerciements

Je tiens tout d'abord à remercier l'Institut National de Recherche en Informatique et en Automatique (INRIA) qui a permis la préparation de cette thèse.

Je remercie

- Messieurs Peter Sander, Professeur à ESSI, université de Nice et Jun Shen professeur à l'Institut de Géodynamique de Talence pour avoir accepté de juger ce travail en tant que rapporteurs, pour leurs conseils scientifiques fructueuses et pour leur participation au jury de thèse.
- Monsieur James Crowley, professeur à INPG et Long Quan, chercheur à CNRS qui m'ont fait l'honneur de participer au jury de ma thèse.
- Roger Mohr, pour avoir dirigé ma thèse tout en m'accordant une grande liberté dans mon travail, pour les discussions scientifiques que nous avons eues, pour ses conseils qui m'ont guidé le long de cette thèse et pour sa chaleur humaine et relation amicale.
- Jérôme Blanc, Patrick Gros, Bart Lamiroy, Long Quan, Peter Sturm pour leur relecture de ma thèse,
- toutes les personnes de l'équipe MOVI pour leurs conseils, discussions fructueuses et travail en commun. et leurs aides lors des difficiles moments de fin de thèse et pour leur amitié chaleureuse.

Table des matières

Remerciements	3
Introduction	10
1 Estimation robuste	14
1.1 Problème de l'analyse de régression	16
1.2 Quelques méthodes préliminaires	17
1.2.1 La transformation de Hough	17
1.2.2 La technique de RanSaC	19
1.2.2.1 Compatibilité entre les données et le modèle : seuil d'erreur	20
1.2.2.2 Le nombre maximal de tirages aléatoires	20
1.2.2.3 Le seuil de dimension d'un ensemble de consensus acceptable	21
1.3 Des estimateurs robustes de régression dans la statistique robuste	22
1.3.1 Critères évaluation d'un estimateur de régression	22
1.3.2 M-, R-, L- et LMedS-estimateurs	23
1.3.2.1 M-, R- et L- estimateurs	23
1.3.2.2 LMedS-estimateur	24
1.4 Autres méthodes robustes	26
1.4.1 CBD : consensus by decomposition	27
1.4.2 MF-estimateur	28

1.4.3	Méthodes robustes pour la reconstruction de surfaces . . .	29
1.4.3.1	MINPRAN	29
1.4.3.2	RESC	31
1.4.3.3	ALKS	32
1.4.3.4	MUSE	33
1.4.4	Méthode des moments	34
1.4.4.1	Méthode des moments classique	34
1.4.4.2	Méthode des moments généralisée	35
1.5	Sommaire d'estimateurs robustes	37
1.6	Domaines d'applications en vision	37
2	Appariement visuel : état de l'art	40
2.1	Méthodes discrètes	42
2.1.1	Corrélation	42
2.1.1.1	Corrélations classiques [AG92, BM95b]	42
2.1.1.2	Corrélations non paramétriques	48
2.1.1.3	Résumé	53
2.1.2	Invariant	54
2.1.3	Distance de Hausdorff [HR93, HJ95]	56
2.1.4	Les relations entre ces trois techniques	57
2.2	Méthodes de programmation dynamique	58
2.2.1	Probabilité d'appariement	59
2.2.2	L'unicité et le processus d'occultation	59
2.2.3	Fonction de coût	60
2.2.4	La contrainte d'occultation	61
2.2.5	La contrainte de monotonicité	61
2.2.6	Programmation dynamique	62

2.2.7	L'utilisation des points de contrôle	63
2.3	Résumé	63
3	Corrélation partielle	65
3.1	Corrélation partielle basée sur la régression	66
3.1.1	Corrélation et régression en probabilité et en statistique . .	66
3.1.1.1	Corrélation	66
3.1.1.2	Régression	68
3.1.2	Corrélation partielle basée sur la régression partielle	69
3.1.2.1	Analyse de corrélation et régression	70
3.1.2.2	Procédure de la corrélation partielle	70
3.1.2.3	Appariements des points dans deux images par corrélation partielle	71
3.1.3	Tests simples et résultats statistiques sur des images pla- naires ¹	73
3.1.3.1	Corrélation partielle pour l'appariement d'images	73
3.1.3.2	Résultats expérimentaux	75
3.2	Corrélation partielle basée sur MVE	82
3.2.1	Analyse de ZNCC au point de vue de localisation	84
3.2.2	Estimation robuste de localisation et de matrice de covariance	84
3.2.2.1	Estimation de localisation et de dispersion	85
3.2.2.2	Quelques définitions et un exemple	85
3.2.2.3	Des estimateurs robustes	87
3.2.3	Corrélation partielle basée sur la localisation robuste	89
3.2.3.1	Analyse de corrélation et de localisation	89
3.2.3.2	Procédure de la corrélation partielle	89
3.2.3.3	Les contraintes	90

1. Le terme «images planaires» employé dans ce mémoire représente les images d'une scène planaire.

3.2.4	Application en vision et résultats expérimentaux	92
3.2.4.1	Application à l'estimation du coefficient de corrélation	92
3.2.4.2	Résultats expérimentaux	93
3.3	Conclusion	95
4	Corrélation précise	97
4.1	Introduction	97
4.2	Le cas d'une translation pure : algorithme rapide	98
4.2.1	Échantillonnage de signal et décalage sous-pixelique	98
4.2.2	Modélisation	99
4.2.2.1	Translation sous-pixelique par convolution : cas unidimensionnel	99
4.2.2.2	Translation sous pixelique par convolution : cas bidimensionnel	100
4.2.3	Prendre en compte les incertitudes des décalages estimés	101
4.2.3.1	Des notations	102
4.2.3.2	Cas de quatre-voisinage	103
4.2.3.3	Cas de huit voisinage	104
4.2.3.4	Explication de l'estimation de la matrice de covariance	105
4.2.3.5	Résumé	105
4.2.4	Liens avec d'autres modèles	105
4.2.5	Comment raffiner la première estimation?	106
4.2.6	Quelques problèmes d'implémentation	107
4.2.7	Résultats expérimentaux	107
4.3	Le cas général : algorithme robuste	111
4.3.1	Obtenons la transformation affine entre les points bidimensionnels correspondants	112

4.3.1.1	LMedS	113
4.3.1.2	RanSaC	114
4.3.2	Calculons le décalage sous pixelique	115
4.3.3	Estimation de l'incertitude de décalage sous-pixelique . .	116
4.3.4	Résultats expérimentaux	117
4.4	Post traitement : incorporant la géométrie épipolaire	121
4.4.1	Réestimation de la géométrie épipolaire	121
4.4.2	Projeter sur la ligne épipolaire	122
4.5	Validation via reconstruction tridimensionnelle	122
4.6	Conclusion	126
5	Conclusion	129
A	Preuve de l'efficacité relative	135
B	Une analyse statistique de SSD et ZSSD	139
C	Résultats expérimentaux de la corrélation précise	141
C.1	Résultats expérimentaux de la corrélation précise : cas translationel	141
C.2	Résultats expérimentaux de la corrélation précise : cas général . .	145
C.2.1	Résultats par la méthode des moindres carrés médians . .	145
C.2.2	Résultats par la technique de RanSaC	148
C.2.2.1	RanSaC	148
C.2.2.2	RanSaC2	151
C.3	Résultats expérimentaux de la corrélation précise après le post traitement	154
	Liste des tableaux	159
	Table des figures	164

Bibliographie

167

Index

177

Introduction

Deux tâches principales en vision, la *navigation* et la *reconnaissance d'objets tridimensionnels*, sont basées sur l'estimation du mouvement tridimensionnel et la reconstruction tridimensionnelle, respectivement.

Depuis vingtaine d'années, la vision par ordinateur a fait des progrès considérables [Fau93, HM93]. On peut obtenir de manière quasi-automatique la reconstruction tridimensionnelle (3D) et l'estimation du mouvement tridimensionnel, à partir des images stéréo [Aya90] ou des séquences d'images [HN94].

La reconstruction 3D se divise en deux classes principales :

- reconstruction avec calibrage [Zha93],
qui consiste en trois étapes [Sin90] :
 1. calibrage [FT87],
 2. appariement stéréo [AG92, RBM94, LMR95]
 3. et calcul de la profondeur et interpolation [Ter88, SS90] ;
- reconstruction sans calibrage,
qui est une approche plus récente [MA90, Fau92, HGC92].

L'estimation du mouvement 3D se base sur le mouvement apparent qu'on appelle aussi le flot optique, qui résulte de l'appariement des points d'images consécutives de la séquence.

Ces deux avancées, ouvrent d'importantes perspectives à la robotique : elles permettent d'envisager dans un prochain futur, des robots beaucoup plus autonomes, et adaptables aux changements imprévus de leurs environnements. Toutes ces tâches ne permettent être opérées que si des appariements entre les images ont été réalisés. Pour cette raison, le problème d'*appariement visuel*, est souvent considéré comme le noyau des ces processus [HA89]. L'appariement visuel sera le sujet principal de ce rapport.

Selon Fischler [FB81], Haralick [Har86] et Schunck [Sch90], les méthodes en vision doivent être *robustes* pour avoir de bonnes performances dans un domaine non contrôlé, car lors de l'extraction de caractéristiques, la reconnaissance et l'appariement, des erreurs grossières sont inévitables. Or ces erreurs perturbent de façon incontrôlable les processus ci-dessus.

Dans ce contexte, ce rapport traite des applications de méthodes robustes aux appariements visuels. Nos contributions se constituent essentiellement de deux parties :

– *Appariement entre images et occultation partielle*

Nous abordons le problème de la mise en correspondance par corrélation pour traiter le cas de présence d'occultations partielles. Nous présentons l'idée de la *corrélation partielle* à l'aide de la *statistique robuste*. Nous avons réalisé deux types de corrélation. Ils se basent sur la régression robuste et la localisation robustes, respectivement. Nous avons fait des expérimentations sur des images d'une scène planaire², et des comparaisons avec les méthodes classiques et les méthodes de Zabih [ZW94] ont aussi été effectuées. Nous avons obtenu de bons résultats pour les points proches de contours d'occultation.

– *Appariement précis*

Nous présentons une méthode linéaire pour effectuer une mise en correspondance sous-pixellique. Cette méthode se compose en plusieurs phases : la première phase consiste à estimer la déformation des fenêtres de manière robuste, en utilisant une carte d'appariements dense ; la deuxième consiste à estimer le décalage local en supposant un développement de premier ordre de signal, ce développement étant exprimé en termes d'une convolution, et nous estimons le masque de cette convolution en résolvant un système d'équations linéaires ; la dernière consiste à réestimer la géométrie épipolaire à partir de ces appariements précis, et projeter le correspondant de chaque pixel de la première image sur la ligne épipolaire associée, calculée à partir de la géométrie précise. Étant une méthode linéaire, elle est applicable aux problèmes réels, où des méthodes non-linéaires et itératives ne satisfont souvent pas les contraintes de temps réel imposées.

Ce mémoire est organisé de la manière suivante :

- Le chapitre 1 est consacré à des notions fondamentales de l'estimation robuste. Les techniques classiques comme la transformée de Hough, la technique de RanSaC (*Random Sample Consensus*), la méthode LMedS (*Least*

2. Ce type d'images sont aussi appelées "images planaires" dans ce mémoire.

Median Square), et aussi des nouvelles méthodes appliquées en vision, comme MINPRAN (*MINimization of the Probability of RANdomness*), RESC (*RE-Sidual Consensus*) et MUSE (*Minimum Unbiased Scale Estimator*) seront présentées.

- Le chapitre 2 donne une vue générale des méthodes d'appariement entre deux images. Ce chapitre contient des méthodes locales et discrètes comme la corrélation, la distance de Hausdorff, la technique des invariants locaux, et aussi une méthode globale : celle de programmation dynamique.
- Le chapitre 3 traite le problème de l'occultation partielle dans l'appariement ; il présente le principe de la *corrélation partielle*. Nous présentons deux méthodes basées sur la régression robuste et la localisation robuste respectivement. Nous avons fait des expérimentations qui montrent qu'à l'aide de ces méthodes, nous pouvons obtenir de bons résultats pour les points proches de contours d'occultation.
- Le chapitre 4 est consacré au problème de l'appariement précis ; un algorithme direct et un robuste sont présentés. Celui direct est basé sur une hypothèse d'approximation locale du signal pour son développement au 1^{er} ordre ; il s'implante directement en résolvant un système d'équations linéaire. Celui robuste estime d'abord la déformation locale des fenêtres de corrélation d'une manière robuste, puis estime le décalage local sur la fenêtre déformée à une précision sous-pixelique. Ces deux algorithmes peuvent être suivis par un post traitement en incorporant la contrainte épipolaire pour améliorer le résultat. On atteint ainsi dans des conditions assez générales des précisions d'ordre 0,08 pixel, et un taux de faux appariements très faible. Les résultats sont atteints avec des changements d'échelle et des rotations d'images qui souvent mettent en défaut les corrélations habituelles.
- Enfin, la conclusion générale présentée dans le chapitre 5 résume cette thèse et présente quelques perspectives de recherche qui s'offrent à la suite de ce travail.

Chapitre 1

Estimation robuste

Dans ce chapitre, nous présentons une synthèse des méthodes d'estimation robuste dans l'ordre chronologique. Tout d'abord, nous présentons la transformation de Hough [Hou62] et la technique de RanSaC [FB80] (*Random Sample Consensus*) comme des méthodes préliminaires, puisqu'elles ont été développées et utilisées les premières dans le contexte de la vision par ordinateur. Avec les progrès de la statistique robuste, d'autres estimateurs robustes comme les M-, R-, L- et LMedS-estimateurs (*Least Median Square*) se sont développés, parmi lesquels les M-estimateurs [BBW88] et le LMedS-estimateur [MMRK91] sont fréquemment employés en vision.

Ces méthodes sont applicables pour un modèle donné, et non pour une classe de modèles, comme dans le problème d'ajustement de surfaces polynomiales où l'on doit connaître a priori l'ordre de surface. Un estimateur CBD (*Consensus By Decomposition*) a été proposé pour traiter ce problème [MMR90]. Quelques méthodes qui tolèrent plus de 50% d'erreurs grossières se sont proposées depuis des années, comme le MF-estimateur [ZH94] (*Model Fitting*), le MINPRAN-estimateur [Ste93] (*MINimization of the Probability of RANdomness*), le RESC-estimateur [YBK94] (*RESidual Consensus*), le ALKS-estimateur [LMP96] (*Adaptive Least k-th order Square*) et le MUSE-estimateur [MS96] (*Minimum Unbiased Scale Estimator*).

Elles sont nécessaires dans la reconstruction de surfaces multiples. Enfin, nous présentons une méthode MOM [NW94] (*Method Of Moments*) qui utilise les moments, cette méthode suppose un mélange de bruit pour considérer à la fois les petites erreurs et les erreurs grossières.

Ces méthodes se classent en deux familles : dans la première famille, les méthodes comme les M-, R- et L-estimateurs essaient de compenser l'influence des erreurs grossières ; dans la deuxième, les méthodes distinguent explicitement les

données correctes et les données aberrantes par des mesures robustes. Cette deuxième famille contient des techniques classiques comme la transformation de Hough, RanSaC et LMedS, ainsi que des estimateurs robustes développés récemment en vision, comme les CBD-, MF-, MINPRAN-, RESC-, ALKS-, MUSE- et MOM-estimateurs.

Les M-, R- et L-estimateurs sont des estimateurs proposés dans la statistique robuste pour compenser les erreurs grossières. Les M-estimateurs utilisent une fonction de croissance plus modérée que la fonction carrée, et les L- et R- estimateurs utilisent les rangs d'erreurs, qui sont moins influencés par les erreurs grossières.

La transformation de Hough consiste à maximiser, de manière robuste, la cohérence dans l'ensemble de toutes les solutions possibles, elle est réalisée par une recherche exhaustive et est donc très couteuse. RanSaC maximise la cohérence dans un ensemble de solutions obtenues par des tirages aléatoires de données et elle est donc plus applicable que la transformation de Hough. Toutes les deux techniques supposent que l'on connaisse a priori l'écart-type des erreurs des données correctes. Cet inconvénient est surmonté par le LMedS- estimateur, qui estime de manière robuste, l'écart-type du bruit par l'estimation de la médiane des erreurs. Le LMedS-estimateur est similaire à la transformation de Hough et à la technique RanSaC, mais il est plus autonome : l'écart-type du bruit est estimé simultanément à l'estimation de la régression.

L'usage du LMedS-estimateur a dominé la vision depuis longtemps, car celui-ci possède un seuil de rupture haut : 50%. Néanmoins, il n'est capable de choisir que les paramètres d'un modèle mais pas le modèle lui-même ; l'estimateur CBD permet d'y remédier. De nouvelles techniques avant un seuil de rupture supérieure à 50% sont disponibles, par exemple, les MF-, MINPRAN-, RESC-, ALKS- et MUSE-estimateurs. Un autre estimateur MOM est un essai pour avoir une méthode robuste analytique.

Remarquons que la notion de «rang» est très utilisée avec estimateurs robustes, que ce soient les anciens R-, L-, LMedS-, ou les nouveaux ALKS- et MUSE-estimateurs. Cette notion est fondamentale dans la statistique non paramétrée et elle trouve aussi sa place dans la statistique robuste.

Dans les sections suivantes, nous formulons le problème de régression et présentons les différentes méthodes d'estimation robuste. Dans la section 1.5, on donnera un résumé, suivi d'une brève introduction aux applications des méthodes robustes en vision.

1.1 Problème de l'analyse de régression

L'analyse de régression est un outil statistique important, fréquemment employé en vision par ordinateur. La méthode des moindres carrés a été populaire pour cette analyse, car elle est facile à utiliser et correspond à l'estimation du maximum de vraisemblance sous l'hypothèse que la distribution des erreurs suit une loi gaussienne centrée. Néanmoins, cette méthode est fragile face à des données aberrantes. En vision, ces données peuvent être provoquées par des erreurs de segmentation ou de mise en correspondance.

On peut illustrer l'absence de robustesse de cette méthode à travers d'un exemple d'ajustement de droite, qui est décrit dans la figure 1.1.

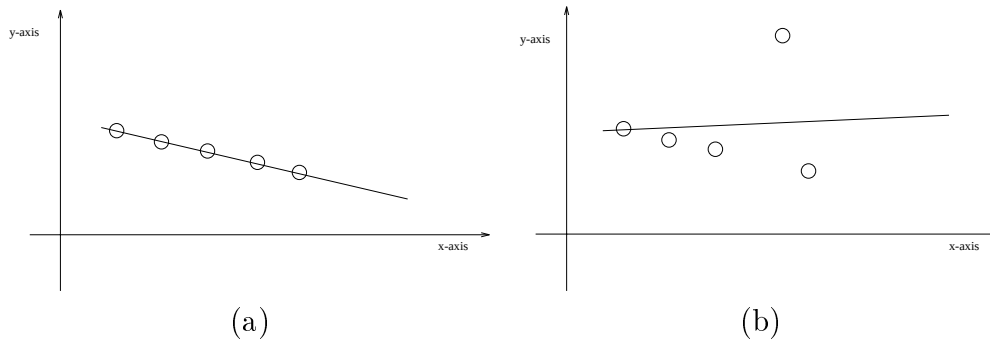


FIG. 1.1 – (a) Données originales avec 5 points et la droite obtenue par régression des moindres carrés. (b) Mêmes données mais avec une donnée aberrante dans la direction y .

On voit que si toutes les données sont correctes comme dans la figure 1.1 (a), la droite estimée est correcte; mais si y a une donnée aberrante comme dans la figure 1.1 (b), la droite estimée peut être très fausse.

Plus mathématiquement, un modèle de régression linéaire d'ordre n est représenté par l'équation :

$$\hat{y}_i = \hat{\theta}_1 x_{i1} + \hat{\theta}_2 x_{i2} + \cdots + \hat{\theta}_n x_{in}.$$

Une telle équation exprime la dépendance entre les coordonnées du $i^{\text{ième}}$ point de données $(x_{i1}, \dots, x_{in}, y_i)$, $\hat{\theta}_j$ étant les estimations des paramètres du modèle. Le résidu, pour chaque point de données, est $r_i = y_i - \hat{y}_i$. Dans la régression des moindres carrés, l'estimation des paramètres du modèle est obtenue par la minimisation de la somme des résidus au carré :

$$\min_{\hat{\theta}_j} \sum_{i=1}^n r_i^2$$

et les $\hat{\theta}_i$ sont obtenus en résolvant un système d'équations linéaires.

1.2 Quelques méthodes préliminaires

1.2.1 La transformation de Hough

En traitement d'image, la technique de Hough a sans doute été la première méthode robuste utilisée. Elle est robuste dans le sens où elle tolère les erreurs grossières. On l'a d'abord introduite pour la détection de droites [Hou62], par exemple, pour la détection de segments de droite dans une image.

Considérons le problème de la recherche d'une droite $y = mx + c$, étant donnés des points dans l'espace image (x, y) . Chaque point (x_0, y_0) correspond à une droite $mx_0 + c = y_0$ dans l'espace des paramètres (m, c) de la droite recherchée. L'intersection de ces droites dans l'espace des paramètres donne les valeurs de (m, c) recherchées.

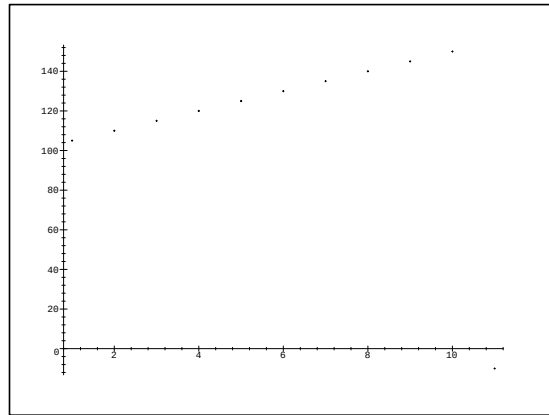
L'algorithme est le suivant :

1. quantification de l'espace des paramètres entre le maximum et le minimum adéquats de c et m ,
2. construction d'un tableau $A(c, m)$ et initialisation de ce tableau à zéro,
3. pour chaque point (x, y) , incrémentation de $A(c, m)$ si la droite correspondante passe par (x, y) ,
4. recherche du maximum dans le tableau $A(c, m)$.

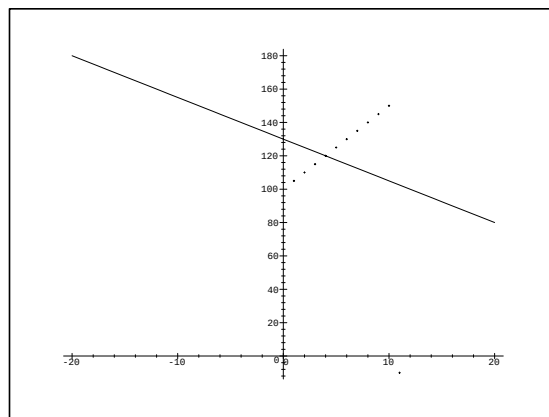
La figure 1.2 montre un exemple illustrant la robustesse de la transformation de Hough. Sur la figure (a) sont affichés des points ($y = 5x + 100$) plus un point aberrant $(11, -20)$; dans la figure (b) est affichée la droite ajustée par estimation aux moindres carrés ; dans la figure (c) sont affichés les droites de vote, dont le point d'accumulation donnent la droite cherchée : $y = 5x + 100$.

On a étendu cette technique au problème de détection de courbes [KBS75].

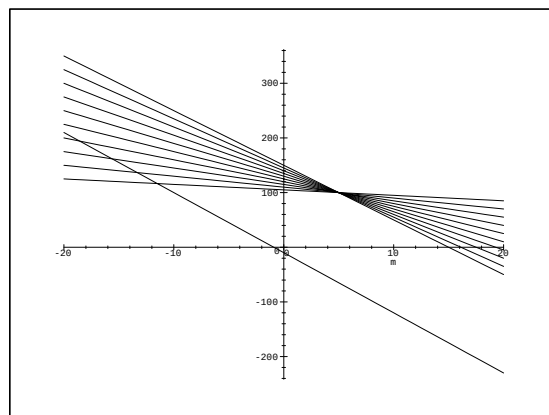
Son inconvénient principal est la complexité en espace et en temps. Si l'on obtient n sous-espaces après la quantification et si l'on a m données, alors sa



(a)



(b)



(c)

FIG. 1.2 – (a) : Les points dont un aberrant ; (b) : La droite obtenue par estimation aux moindres carrés ; (c) : Les coordonnées du point d'accumulation (5, 100) donnent la droite ajustée aux points, $y = 5x + 100$. Pour ce point, la valeur de l'accumulation est 10.

complexité en espace est d'ordre $O(n)$, et sa complexité en temps est d'ordre $O(nm)$.

Sa complexité en temps peut être réduite si la direction du gradient est intégrée [KBS75]. Car dans ce cas, on diminue le nombre de paramètres d'une unité, et on réduit également n , le nombre de sous-espaces quantifiés.

Un aperçu général de cette technique est donné par Illingworth [IK88]. Cette technique est utilisée pour reconnaître des objets partiellement occultés [TMV85].

1.2.2 La technique de RanSaC

Au début des années 80, Fischler et Bolles [FB81] ont proposé un algorithme pour l'ajustement du modèle qui maximise l'ensemble de *consensus*. Cet ensemble se constitue des données dont l'erreur par rapport au modèle trouvé est inférieure à un seuil.

Étant donné un ensemble de points P , on cherche un modèle paramétré avec n paramètres, l'algorithme se déroule comme suit :

1. Choisir aléatoirement un sous-ensemble $S1$ de n données de P , trouver le modèle $M1$.
2. Déterminer le sous-ensemble $S1^*$ de points de P dont l'erreur par rapport à $M1$ est inférieure à un seuil.

Cet ensemble est appelé *l'ensemble de consensus de $S1$* .

3.
 - Si le cardinal de $S1^*$, $\#(S1^*)$, est supérieur à un seuil t , utiliser $S1^*$ pour calculer le nouveau modèle $M1^*$.
 - Sinon,
 - (a) si le nombre d'essais est inférieur à un seuil k , aller à l'étape 1;
 - (b) sinon, échec.

Cette technique s'avère robuste aux erreurs grossières. Sa comparaison avec l'estimateur de LMedS est donnée dans [MMRK91] et aussi à la page 25.

RanSaC nécessite de fixer trois paramètres

- le seuil d'erreur tolérée utilisé pour déterminer si un point est compatible avec un modèle
- le nombre maximal de tirages aléatoires

- le seuil t , qui est le nombre minimal de bons points, permettant de décider si un modèle trouvé est correct.

Des méthodes pour déterminer des valeurs raisonnables de ces paramètres sont discutées ci-après.

1.2.2.1 Compatibilité entre les données et le modèle : seuil d'erreur

Il est généralement possible de déterminer le seuil d'erreur tolérée expérimentalement. En perturbant les données, on calcule le modèle et on mesure l'erreur. Le seuil peut être pris égal à quelques fois l'écart-type des erreurs. En supposant une distribution d'erreurs, ce seuil correspond à une certaine probabilité. Il dépend donc aussi de la distribution des erreurs supposée, mais l'hypothèse de la distribution gaussienne des erreurs est habituellement suffisante.

1.2.2.2 Le nombre maximal de tirages aléatoires

L'arrêt du choix du nombre de nouveaux sous-ensembles de P peut être basé sur l'espérance du nombre de tirages k , qui est le nombre nécessaire pour avoir un sous-ensemble de n bons points de données. Si w est la fréquence de bonnes données, alors la probabilité que $k = i$ est $(1 - w^n)^{i-1}w^n$, et son espérance est :

$$E(k) = \sum_{i=1}^{+\infty} i(1 - w^n)^{i-1}w^n = w^n \sum_{i=1}^{+\infty} i(1 - w^n)^{i-1} = w^n \frac{1}{(w^n)^2} = \frac{1}{w^n}. \quad (1.1)$$

L'écart-type de k , $SD(k)$ est donné par $SD(k) = \sqrt{E(k^2) - E(k)^2}$.

En notant $b = w^n$ et $a = 1 - b$, on a

$$E(k^2) = \sum_{i=1}^{+\infty} bi^2a^{i-1} = \frac{2 - b}{b^2},$$

et

$$SD(k) = \frac{\sqrt{1 - w^n}}{w^n}.$$

Le nombre maximal de tirages qu'on utilise est souvent compris entre $E(k) + SD(k)$ et $E(k) + 2SD(k)$.

D'un autre point de vue, on peut analyser le nombre nécessaire de tirages pour avoir une probabilité d'erreur faible. La probabilité que tous les k n -uplets

différents choisis au hasard contiennent au moins une donnée corrompue est $Prob = (1 - w^n)^k$.

Donc, pour avoir une probabilité d'erreur inférieure à $Q \ll 1$, il suffit que $Prob < Q$, ce qui est équivalent à

$$k > \frac{\log Q}{\log(1 - w^n)} = k_{min}. \quad (1.2)$$

Par exemple, si $w = 0.5$ et $n = 4$, alors $W^n = \frac{1}{16}$. Ainsi, pour être sûr d'avoir choisi au moins un bon sous-ensemble de données avec un taux de confiance de 90%, il suffit que $k > \frac{\log Q}{\log(1 - w^n)} = \frac{\log 0.1}{\log \frac{15}{16}} = 35,7$. C'est-à-dire, si au plus 50% de données sont corrompues, alors en choisissant 36 quadruples pour le calcul d'un ajustement tridimensionnel, la probabilité que tous les ensembles sont corrompus est inférieure à 10%. Notons que si $w^n \ll 1$ alors $k_{min} \approx -\log Q \times E(k)$. Donc si $Q = 0.10$ et $w^n \ll 1$, alors $k_{min} \approx 2,3E(k)$; si $Q = 0,05$ et $w^n \ll 1$, alors $k_{min} \approx 3,0E(k)$.

1.2.2.3 Le seuil de dimension d'un ensemble de consensus acceptable

On utilise le seuil t pour déterminer si l'ensemble de consensus est assez grand pour terminer l'échantillonnage. On doit choisir t assez grand pour satisfaire à deux conditions :

- le modèle trouvé est correct ;
- il fournit un nombre de points suffisamment grand pour la procédure d'estimation finale.

Pour assurer qu'on trouve un modèle correct, supposons que y est la probabilité qu'une donnée appartienne à un faux modèle. Nous espérons que y^{t-n} est très petit. Supposons que $y < 0,5$, alors si $t - n$ est supérieur ou égal à 5, on obtient une probabilité supérieure à $1 - 0,5^5 > 95\%$ d'avoir un modèle correct.

Pour avoir un nombre de points suffisamment grand pour la procédure d'estimation finale, on a proposé des méthodes dans [Sor70],

1.3 Des estimateurs robustes de régression dans la statistique robuste

1.3.1 Critères évaluation d'un estimateur de régression

Trois concepts sont souvent utilisés pour évaluer une méthodes de régression : *l'efficacité relative*, *le point de rupture* (ou *le seuil de rupture*), *la complexité en temps* [Hub81].

- *L'efficacité relative* d'une méthode de régression est définie comme le rapport entre la variance la plus basse atteignable pour les paramètres à estimer et la variance actuelle obtenue par la méthode donnée. Il dépend de la distribution du bruit supposée.

Par exemple : en présence d'un bruit gaussien, l'estimateur de la moyenne a une efficacité asymptotique de 1, alors celle de la médiane est réduit à $\frac{2}{\pi}$ [Hub81].

La preuve est donnée dans l'annexe A.

- *Le point de rupture* ou *le seuil de rupture* d'une méthode de régression est le plus bas pourcentage d'erreurs grossières qui peut forcer la méthode à donner un résultat complètement faux.

Par exemple : le point de rupture de la moyenne est $\frac{1}{n}$ (asymptotiquement 0), car une seule erreur aberrante peut corrompre le résultat d'une valeur arbitrairement grande ; la médiane reste fiable si moins de la moitié des données sont contaminées, elle donne un point de rupture asymptotique maximal de 0,5.

- *La complexité en temps* de l'estimation aux moindres carrés est $O(np^2)$ où n est le nombre de données et p est le nombre des paramètres à estimer. Dans la pratique, une complexité en temps inférieure à $O(n^2)$ est demandée.

Une bonne méthode de régression doit posséder une bonne efficacité relative, un haut point de rupture et une complexité en temps raisonnable.

La notion du seuil de rupture Soit Z un ensemble de n points de données, et Z' la version de Z avec m points remplacés par des valeurs arbitrairement éloignées du modèle. Notons $\hat{\theta} = T(Z)$ l'estimateur de régression. Le biais dans l'estimation, à cause des erreurs grossières, est donné par :

$$\text{biais}(m, T, Z) = \sup_{Z'} \|T(Z') - T(Z)\|.$$

Le seuil de rupture est le plus petit pourcentage d'erreurs grossières qui peut forcer cette méthode à donner n'importe quel résultat. Il est défini comme suit :

$$\epsilon_n^* = \min \left\{ \frac{m}{n} : \text{biais}(m, T, Z) \text{ est infini} \right\}.$$

Pour la régression des moindres carrés, $\epsilon_n^* = \frac{1}{n}$, et à la limite $\epsilon_\infty^* = 0$: la moyenne n'est pas robuste du tout. On présente maintenant des méthodes robustes pour la régression linéaire.

1.3.2 M-, R-, L- et LMedS-estimateurs

1.3.2.1 M-, R- et L- estimateurs

Dans les années 70, on a développé des estimateurs robustes comme les M-estimateurs, R-estimateurs et L-estimateurs [Hub81].

– M-estimateurs

Les M-estimateurs [BBW88] sont les plus populaires. Ils minimisent la somme d'une fonction symétrique et définie positive $\rho(r_i)$ du résidu r_i , avec un seul minimum à $r_i = 0$. La M-estimation des paramètres est obtenue en convertissant la minimisation de $\sum_i \rho(r_i)$ en un problème de moindres carrés pondérés. L'initialisation est importante et la convergence n'est pas garantie. Récemment, Li [LI96] a développé le *AM* estimateur (*Annealing M-estimator*), une combinaison de M-estimateur et *GNC* (*Graduated Non-Convexity* [BZ87]) pour améliorer la précision, la stabilité et le point de rupture. Black [BR94] a décrit une équivalence entre la M-estimation et la régularisation.

– R-estimateurs

Les R-estimateurs sont basés sur un tri de l'ensemble des résidus. Ils proposent d'estimer les paramètres en minimisant $\sum_i a_n(R_i)r_i$, où r_i est le résidu, R_i son rang, et a_n est une fonction de coût. a_n doit être monotone et vérifier la condition $\sum_i a_n(R_i) = 0$ [Hub81].

– L-estimateurs

Les L-estimateurs utilisent la combinaison linéaire de la statistique d'ordre. Les méthodes basées sur la médiane et la moyenne coupée appartiennent à

cette classe. Des études simulées [Hei81] ont démontré que les L-estimateurs donnent des résultats moins satisfaisants que ceux des deux autres classes.

Tous ces M-, R- et L- estimateurs ont un point de rupture inférieur à $\frac{1}{p+1}$, où p est le nombre de paramètres [HRRS86].

1.3.2.2 LMedS-estimateur

On a proposé des estimateurs robustes ayant un point de rupture proche de 0,5. Parmi eux, celui des moindres carrés médians est devenu le plus populaire [Rou84]. L'efficacité relative de cette méthode peut être améliorée en la combinant avec des techniques basées sur les moindres carrés. La complexité en temps peut être réduite par un échantillonnage aléatoire. La procédure est expliquée dans l'algorithme 1.

La procédure d'estimateur LMedS

Algorithme 1 *Supposons qu'il y a n points de données et p paramètres dans le modèle linéaire.*

1. *Choisir au hasard des ensembles de p points de l'ensemble de n points de données.*
2. *Calculer l'ajustement du modèle aux p points pour chaque ensemble ¹.*
3. *Calculer la médiane des résidus au carré et prendre le modèle qui donne la médiane minimale.*

Cette procédure est répétée jusqu'à ce qu'un ajustement soit trouvé avec une médiane assez petite ou jusqu'à ce que le nombre d'échantillonnages soit assez grand.

Cette méthode a un seuil de rupture de 50%. un exemple d'application de cette méthode est montré à la figure 1.3.

Dans cette figure, contrairement à ce qui est montré dans la figure 1.1, grâce à la robustesse de la méthode LMedS, une donnée abberante n'a pas beaucoup d'influence au résultat final.

1. À chaque itération, $\hat{\beta}_0(i_1 \dots i_p)$ peut être trouvé en minimisant $\text{med}_1 r_i^2$, étant donnés $\hat{\beta}_j(i_1 \dots i_p)$ pour $j = 1 \dots p - 1$.

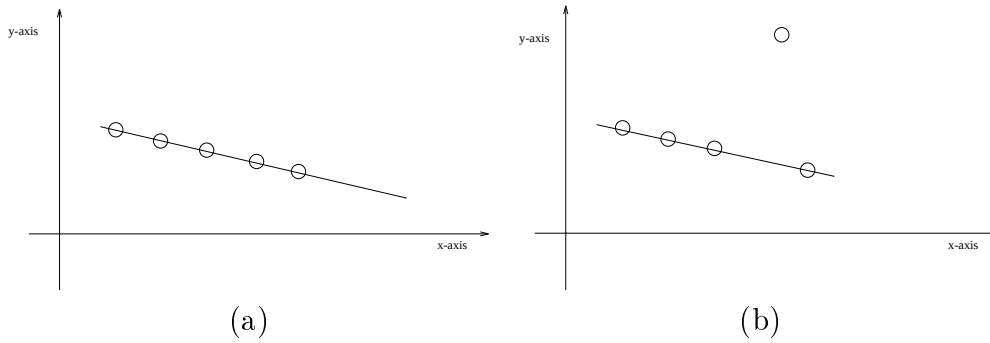


FIG. 1.3 – (a) Données originales avec cinq points et la droite obtenue par régression des moindres carrés médians. (b) Mêmes données mais avec une erreur grossière dans la direction y .

Le nombre nécessaire de tirages aléatoires pour la procédure de LMedS peut être calculé comme dans la technique de RanSaC (page 20, équation 1.2).

Après avoir obtenu le modèle correct et la médiane des résidus au carré associée $\text{med}_i e_i^2$, l'écart-type du bruit σ peut être estimé comme [RL87] :

$$\sigma = 1,4826 \left(1 + \frac{5}{n-p}\right) \sqrt{\text{med}_i e_i^2},$$

où n est le nombre de points et p est le nombre de paramètres, 1,4826 est un facteur dans la présence d'un bruit gaussien et $\frac{5}{n-p}$ est proposé pour la correction d'un échantillonnage fini. Les points ayant de résidu supérieur à $2,5\sigma$ sont rejetés comme des données aberrantes. Et une estimation des moindres carrés peut être faite sur les données restantes pour raffiner le résultat.

Comparaison avec RanSaC Malgré quelques différences, LMedS est une technique très similaire à la technique de RanSaC. L'estimateur LMedS a un seuil de rupture de 0,5, sans aucune information a priori sur les données. Dans la technique de RanSaC, la dimension de l'ensemble consensus acceptable ne peut pas être inférieure à la moitié non plus. Le nombre maximal d'itérations est établi par l'équation (1.2) pour ces deux méthodes. Dans LMedS, le nombre de tirages aléatoires est toujours le même ; dans RanSaC, l'ensemble de consensus peut être trouvé plus tôt quand le cardinal dépasse le seuil t . Néanmoins, si l'on emploie un seuil pour le résidu d'erreur tolérée dans LMedS, un critère d'arrêt similaire à la technique RanSaC est obtenu. En termes statistiques, LMedS minimise une mesure robuste d'erreurs du modèle, et RanSaC maximise le nombre de bonnes données. Ces deux critères ne sont pas mathématiquement équivalents ; dans le cas général, au moins la moitié des données sont bonnes et les deux méthodes

gènèrent des résultats très similaires.

Les techniques de LMedS et RanSaC se sont développées indépendamment dans les domaines de la statistique et de la vision par ordinateur respectivement ; elles sont basées sur des concepts similaires. La seule différence significative est que le seuil pour distinguer une «bonne» donnée et une «mauvaise» donnée est calculé automatiquement pour LMedS et il est choisi a priori pour RanSaC. Car dans la méthode LMedS, la médiane des erreurs au carré est calculée, puis l'écart-type de bruit est estimé selon cette médiane, et le seuil est normalement 2,5 fois l'écart-type ; dans la méthode de RanSaC, l'écart-type de bruit doit être obtenu autrement.

Dans [RL93], Roth a donné un algorithme aléatoire qui regroupe la méthode des moindres carrés, la M-estimation, l'estimateur des moindres carrés médians et la technique de Hough comme des minimisations de fonctions de coût différentes. Cet algorithme peut donner des comparaisons de ces estimateurs différents.

1.4 Autres méthodes robustes

Bien que la méthode de LMedS ait dominé en vision pendant des années, elle montre des inconvénients : elle ne peut pas choisir le modèle, elle tolère au plus 50% d'erreurs grossières, et elle n'a pas d'expression analytique. D'autres méthodes robustes ont aussi été étudiées.

Pour traiter le choix automatique du modèle, on a présenté une méthode appelée *consensus* dans [MNMR90], cette méthode est réalisée par un ensemble d'opérateurs suivis d'une procédure de *consensus*. On a présenté un tel estimateur dans [MMR90]. On a démontrée que cet estimateur n'est pas sensible au choix du modèle, et on a présenté son application à la détection de contours dans [Min94].

Beaucoup de travaux se sont effectués pour dépasser le seuil de rupture de 50%, en ajoutant des hypothèses raisonnables. Un des premiers est peut-être le travail de Zhuang, la MF- (*Model Fitting*) estimation dans [ZWZ92], qui est réalisé par la modélisation partielle du logarithme de la fonction de vraisemblance. Cet algorithme maximise le logarithme de la fonction de vraisemblance en prenant compte des contraintes heuristiques et de la décision de Bayes. Stewart a présenté un estimateur MINPRAN (*Minimization of the PRObability of RANdomness*) dans [Ste93], en supposant que les données aberrantes sont uniformément distribuées dans la portée du capteur. Yu a présenté une méthode de reconstruction qui s'appelle RESC (*RESidual Consensus*) en mesurant le consensus résiduel [YBK94]. Miller [MS96] a développé une méthode MUSE (*Minimum Unbiased Scale Estimator*) qui cherche l'ajustement via la minimisation de l'esti-

mation d'échelle non biaisée. Cette méthode est similaire à celle développée par Lee [LMP96], la méthode ALKS (*Adaptive Least K-th order Square*).

Enfin, une autre méthode robuste analytique MOM (*Method Of Moments*) est réalisée en utilisant les *moments* [NW94] sous l'hypothèse que le bruit a une distribution composée d'une partie gaussienne et d'une partie uniforme. La direction de recherche des estimateurs analytiques et robustes a rarement été abordée jusqu'à maintenant.

1.4.1 CBD : consensus by decomposition

Nous avons expliqué le principe des moindres carrés médians dans l'algorithme 1.

Calculer un modèle candidat en choisissant un petit sous-ensemble de données. Appliquer ce modèle à l'ensemble des données. Calculer une mesure de qualité globale pour ce modèle. Répéter cette procédure pour optimiser la mesure de qualité.

Pour avoir une bonne estimation aux moindres carrés médians, il est nécessaire qu'au moins un modèle candidat (un p -uplet) apporte des valeurs correctes pour les paramètres. Quand le bruit aléatoire est important, un p -uplet donne une estimation non biaisée mais avec une grande variance. Donc le p -uplet ne donne plus une estimation fiable des paramètres du modèle.

Le consensus dans LMedS est obtenu en minimisant la médiane des erreurs au carré et en maximisant le nombre de bonnes données dans RanSaC.

Le consensus peut être obtenu autrement. Pour l'estimateur CBD, la «structure d'échantillonnage» est définie comme l'ensemble d'index des données. Par exemple, la «structure d'échantillonnage» de données bidimensionnelles est l'ensemble des valeurs (x, y) où les données existent. Le principe de CBD est le suivant [MMR90] :

Calculer un modèle candidat pour un petit voisinage de chaque point dans la «structure d'échantillonnage». Définir une «image» pour chaque paramètre. C'est-à-dire, pour chaque point dans la «structure d'échantillonnage», estimer une valeur pour chaque paramètre à partir d'un petit voisinage de ce point. Estimer les paramètres à partir de leurs «images» en appliquant récursivement la procédure de détection de mode [Min94].

On a démontré que CBD est insensible au choix du modèle, et il peut s'utiliser dans l'ajustement d'une classe de modèles. Dans [MMR90], on l'a appliqué au traitement d'image, et l'a comparé avec LMedS. Dans [Min94], on l'a appliqué à la détection de contours, et l'a comparé avec le détecteur de Canny. Ces comparaisons montrent qu'il est moins sensible à un bas rapport signal par bruit.

1.4.2 MF-estimateur

Le MF-estimateur a été un des premiers estimateurs qui ont essayé de dépasser le seuil de rupture de 50%. Son idée principale est de supposer que les données sont triées dans deux classes : «bonnes» données et «mauvaises» données selon une décision bayésienne. Le problème se formule comme suit :

On cherche à estimer p paramètres inconnus $\beta_1, \dots, \beta_p$, représentés par un vecteur β à partir de N observations $y_1, \dots, y_N$, qui sont de dimension m . Supposons qu'il y a N fonctions $s_k : R^p \rightarrow R^m$, $k = 1, \dots, N$, telles que pour chaque β , $s_k(\beta)$ peut être calculé. Le résidu entre y_k et $s_k(\beta)$ est noté $r_k = y_k - s_k(\beta)$. On suppose, de plus, que chaque observation y_k a une probabilité ϵ d'être «mauvaise». Les r_k ont une distribution identique avec une fonction de densité f :

$$f(r_k) = \frac{1 - \epsilon}{(\sqrt{2\pi}\sigma)^m} e^{-\frac{|r_k|^2}{2\sigma^2}} + \epsilon h(r_k) \quad (1.3)$$

où h représente la densité de probabilité inconnue. Si les r_k sont indépendants entre eux, le logarithme de la fonction de vraisemblance des observations $y_1, \dots, y_N$ qui dépende de $\beta, \sigma, h(r_1), \dots, h(r_N)$ peut être exprimé par Q comme suit :

$$Q = \log P(y_1, \dots, y_N | \beta, \sigma, \epsilon, h(r_1), \dots, h(r_N)) = \sum_k \log f(r_k)$$

D'après l'équation (1.3), la probabilité conditionnelle que «l'observation y_k est bonne» est λ_k :

$$\lambda_k = \frac{\frac{1 - \epsilon}{(\sqrt{2\pi}\sigma)^m} e^{-\frac{|r_k|^2}{2\sigma^2}}}{f(r_k)}$$

En se basant sur la décision bayésienne, on classe l'observation comme «bonne» si $\lambda_k > 0.5$, comme «mauvaise» sinon. Notons G l'ensemble des bonnes observations et B l'ensemble des mauvaises :

$$G = \{k | \lambda_k > 0.5\} = \{k | \frac{1 - \epsilon}{(\sqrt{2\pi}\sigma)^m} e^{-\frac{|r_k|^2}{2\sigma^2}} > \epsilon h(r_k)\}$$

$$B = \{k | \lambda_k \leq 0.5\} = \{k | \frac{1 - \epsilon}{(\sqrt{2\pi}\sigma)^m} e^{-\frac{|r_k|^2}{2\sigma^2}} \leq \epsilon h(r_k)\}$$

Pour bien conditionner ce système, on utilise quatre contraintes.

- La première est la condition heuristique suivante :

$$\frac{\sharp G}{N} = 1 - \epsilon, \quad \frac{\sharp B}{N} = \epsilon.$$

- La deuxième est que, pour deux observations différentes, celle possédant un résidu de valeur absolue plus petite a une plus grande probabilité d'être bonne que l'autre, autrement dit :

$$\max\{|r_k| \mid k \in G\} < \min\{|r_k| \mid k \in B\}.$$

- La troisième contrainte est qu'on doit avoir assez de bonnes données : si on note L le nombre minimal de bonnes données, alors $\sharp G \geq L$.
- La quatrième contrainte est que l'écart-type du bruit doit être limité, disons par U : $\sigma \leq U$.

En résumé, supposant que h peut être modélisée comme suit : $h(r_1) = h(r_2) = \dots = h(r_N) = \delta$, on cherche à maximiser :

$$\max Q(\theta, \sigma, \epsilon, \delta) = \sum_k \log \left(\frac{1 - \epsilon}{(\sqrt{2\pi}\sigma)^m} e^{-\frac{|r_k|^2}{2\sigma^2}} + \epsilon\delta \right)$$

sous les contraintes $\sharp G = (1 - \epsilon)N$, $N \geq L$ et $\sigma \leq U$.

Pour plus de détails, voir [ZWZ92].

On a appliqué cette méthode à l'estimation de pose [ZH94], et a montré qu'elle tolère une grande portion (proche de 90%) de données aberrantes.

1.4.3 Méthodes robustes pour la reconstruction de surfaces

1.4.3.1 MINPRAN

Lors de la reconstruction de surfaces, on rencontre souvent des cas où plus de la moitié des données sont mauvaises, les données appartiennent à plusieurs surfaces, et, quelques fois, les données sont totalement aléatoires et aucune surface existe pour les ajuster. On cherche donc un estimateur acceptant plus d'une

moitié des données aberrantes, pour construire des surfaces multiples, et ne pas produire de surfaces imaginaires. Un estimateur MINPRAN (*MINimization of the Probability of RANdomness*) introduit par Stewart [Ste93] vérifie bien ces exigences.

Cet estimateur s'est basé sur l'hypothèse que les données aberrantes sont uniformément distribuées dans la portée du capteur. Considérons un ajustement $\phi(x)$ et une bande de $\pm r$ autour de ϕ illustrés dans la figure 1.4. Si les données aberrantes sont uniformément distribuées dans l'intervalle $[Z_{\min}, Z_{\max}]$, alors la probabilité que plus de k points se trouvent dans $\phi(x) \pm r$ est :

$$\mathcal{F}(r, k, N) = \sum_{i=k}^N C_N^i \left(\frac{r}{Z_0}\right)^i \left(1 - \frac{r}{Z_0}\right)^{N-i}$$

où $Z_0 = (Z_{\max} - Z_{\min})/2$.

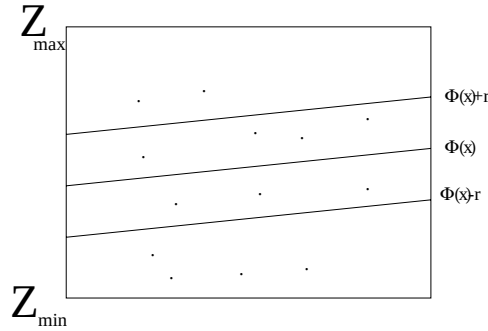


FIG. 1.4 – Si les données sont uniformément distribuées de $Z_{\min}$ à $Z_{\max}$, alors la probabilité que plus de k points se trouvent dans $\phi \pm r$ est donnée par la somme binomiale.

Étant donnés N points, MINPRAN cherche l'ajustement ϕ et les valeurs de r et k qui minimisent $\mathcal{F}$.

$\mathcal{F}(r, k, N)$ est une distribution binomiale, qui est équivalente à la fonction bêta incomplète :

$$\sum_{i=k}^N C_N^i \left(\frac{r}{Z_0}\right)^i \left(1 - \frac{r}{Z_0}\right)^{N-i} = \frac{N!}{(k-1)!(N-k)!} \int_0^{\frac{r}{Z_0}} t^{k-1} (1-t)^{N-k} dt$$

À partir de cette équation, on déduit des propriétés importantes de $\mathcal{F}$:

1. étant donné k , $\mathcal{F}(r, k, N)$ est une fonction croissante en r ;

2. étant donné r , $\mathcal{F}(r, k, N)$ est une fonction décroissante en k ;
3. étant donné un ajustement, supposons que les résidus sont $r_i, \dots, r_N$ dans l'ordre croissant ; alors, il y a au plus N minima locaux dans $\mathcal{F}$ pour l'ajustement donné, qui apparaissent aux couples (i, r_i) .

Sur la base de ces propriétés, on réalise l'estimateur MINPRAN par un échantillonnage aléatoire. On a montré qu'il tolère plus de 50% de fausses données, il est applicable quand les données viennent de surfaces multiples et il ne trouve pas d'ajustement imaginaire quand il n'y en a pas.

Un inconvénient de MINPRAN est que l'hypothèse de distribution uniforme des erreurs grossières n'est pas suffisante pour extraire des surfaces multiples. Son coût de calcul est d'ordre $O(N^2 + SN \log N)$, où S est le nombre de tirages, et N le nombre des données ; elle est donc un peu coûteuse.

1.4.3.2 RESC

Une méthode qui s'appelle *RESidual Consensus* (RESC) [YBK94] est employée pour la segmentation et la reconstruction d'images de distance. On choisit un ensemble de points qui détermine le meilleur ajustement en utilisant l'histogramme des erreurs.

L'algorithme se présente comme suit :

1. choisir aléatoirement K sous-ensembles de p points de l'ensemble de données S ;
2. pour chaque ensemble de points, calculer le résidu avec les primitives déterminées par ces points, fabriquer un histogramme des résidus ;
3. choisir l'histogramme qui a la meilleure qualité (voir dessous) parmi ces K ensembles ;
4. déterminer l'écart-type σ des résidus de l'histogramme ;
5. trouver les bonnes données et les supprimer de S ;
6. répéter les étapes 1–5 jusqu'à ce que $S = \phi$.

Pour l'optimisation de l'histogramme, on doit déterminer une fonction de coût basée sur l'analyse de l'histogramme. Il y a deux critères possibles :

- le nombre de points appartenant à la surface doit être le plus grand possible (par exemple : RanSaC, Transformation de Hough) ;

- les résidus des points appartenant à la surface doivent être les plus petits possibles (par exemple : moindres carrés).

Le critère ψ qu'on emploie combine ces deux critères : $\psi = \sum_{i=1}^m \frac{h_i^\alpha}{|r_i|^\beta}$, où m est le nombre de colonnes dans l'histogramme, h_i est le nombre de points dont l'erreur est compris entre r_{i-1} et r_i , et r_i est proportionnel à i . Les auteurs ont choisi $\alpha = 1.3$ et $\beta = 1.0$ pour les deux coefficients. On cherche alors à maximiser ψ .

L'expérience montre que cette méthode tolère plus de 80% de données aberrantes. Néanmoins, elle possède l'inconvénient d'avoir des paramètres comme α et β à choisir.

1.4.3.3 ALKS

L'estimateur ALKS *Adaptive Least K-th order Square* minimise la statistique d'ordre k des résidus au carré. Il tolère mieux les erreurs aberrantes structurées que MINPRAN et RESC [LMP96].

On présente d'abord le calcul de LKS (*Least K-th order Square*).

On choisit un p -uplet de données pour définir un modèle, on utilise tous les paramètres du modèle sauf l'intercept pour calculer les résidus par rapport à ce modèle, on trie les résidus u_i en ordre croissant, et trouve la position de la fenêtre la plus petite contenant au moins k résidus, soit :

$$d_{lk} = \frac{u_{(l+k-1):(n-p)} - u_{l:(n-p)}}{2}$$

où $u_{i:n-p}$ dénote le $i^{\text{ième}}$ résidu parmi tous les $n - p$ résidus.

Cette procédure est répétée pour plusieurs p -uplets. Le plus petit $d_{lk} = \hat{d}_k$ nous donne l'estimation LKS du modèle.

L'estimation robuste de l'écart-type du bruit peut être approximée par

$$\hat{s}_k = \frac{\hat{d}_k}{\Phi^{-1}(0, 5(1 + \frac{k}{n}))}.$$

Les bonnes données sont celles ayant des résidus $|r_{ik}| \leq 2, 5\hat{s}_k$. La variance de l'erreur normalisée ϵ_k^2 est calculée :

$$\epsilon_k^2 = \frac{1}{q_k - p} \sum_{i=1}^{q_k} \left(\frac{r_{ik}}{\hat{s}_k} \right)^2 = \frac{\hat{\sigma}_k^2}{\hat{s}_k^2} \quad (1.4)$$

où seulement les q_k bons points sont utilisés.

La procédure utilisant LKS avec le k qui minimise (1.4) s'appelle ALKS.

Son avantage par rapport aux procédures MINPRAN et RESC est qu'il n'a ni besoin de paramètres a priori comme α et β dans RESC, ni de l'hypothèse d'une distribution uniforme des erreurs grossières comme dans MINPRAN. Les erreurs abberantes peuvent donc avoir n'importe quelle distribution.

1.4.3.4 MUSE

LMedS cherche un ajustement qui minimise la médiane des erreurs au carré : $\min_{\hat{\theta}} \text{med}_i r_{i\hat{\theta}}^2$, où $r_{i\hat{\theta}}$ est le résidu du $i^{\text{ième}}$ point de donnée mesuré relativement à l'ajustement $\hat{\theta}$. Il génère une estimation robuste de l'écart-type de bonnes données comme $\hat{\sigma}_{\hat{\theta}} = C \sqrt{\text{med}_i r_{i\hat{\theta}}^2}$, où C est une constante dépendant de la distribution des erreurs de bonnes données. Si les bonnes données suivent une distribution gaussienne, alors $C = 1.4826(1 + 5/(n - p))$ [RL87]. Si le pourcentage des erreurs grossières dépasse 50%, cette estimation augmente dramatiquement. Pourtant, une estimation d'échelle peut encore être construite en utilisant la statistique d'ordre, qui peut être vue comme une généralisation de la médiane, car aucune hypothèse sur le pourcentage de bonnes données est faite.

MUSE (*Minimum Unbiased Scale Estimator*) se base sur cette intuition. Cet estimateur calcule une série d'estimations d'échelle et les combine pour produire une nouvelle estimation d'échelle [MS96].

Pour n'importe quel ajustement, on estime l'échelle non biaisée des k plus petits résidus, pour $1 \leq k \leq N - p$. Cette valeur est utilisée pour estimer l'échelle et trouver le meilleur ajustement.

Étant donnée une fonction de densité du résidu $\phi(r, \sigma)$, où ϕ est la densité de la variable aléatoire r centrée et ayant l'écart-type σ , et N points de $\phi(r, \sigma)$, le $k^{\text{ième}}$ résidu $r_{k:N}$ peut être exprimé en termes de celui d'une distribution standard, $r_{k:N} = \sigma u_{k:N}$. Cela donne l'estimation d'échelle :

$$s_k = \frac{r_{k:N}}{E(u_{k:N})}$$

où $E(u_{k:N})$ peut être calculé directement à partir de $\phi(u, 1)$.

Si $\phi(u, 1)$ est symétrique, $E(u_{k:N})$ peut être approché par :

$$E(u_{k:N}) = \Phi^{-1}(0, 5(1 + \frac{k}{N+1}))$$

où Φ est la fonction de distribution de résidu.

MUSE se montre capable d'extraire les surfaces contenant moins de 50% de bonnes données et de détecter de petites discontinuités.

1.4.4 Méthode des moments

En statistique, deux types de méthodes existent pour un problème d'estimation. L'une est l'estimation par maximisation de la vraisemblance, qui est basée sur les résidus ; l'autre est celle des moments [MGB63].

Nous présentons d'abord cette dernière méthode dans le cadre de la statistique classique, et puis nous la généralisons pour notre cas. Cette méthode, appelée MOM (*Method Of Moments*), donne un estimateur analytique en supposant que la distribution des erreurs est la combinaison linéaire d'une partie gaussienne et d'une partie uniforme.

1.4.4.1 Méthode des moments classique

Soit $f_{\theta_1, \dots, \theta_k}$ une fonction de densité d'une variable aléatoire X qui a k paramètres $\theta_1, \dots, \theta_k$. Notons μ_r son $r^{\text{ième}}$ moment autour de 0, c'est-à-dire, $\mu_r = E(X^r) = \int x^r f(x) dx$. C'est une fonction de $\theta_1, \dots, \theta_k$, et on note $\mu_r = \mu_r(\theta_1, \dots, \theta_k)$. Soient $X_1, \dots, X_n$ un échantillonnage aléatoire de la densité $f_{\theta_1, \dots, \theta_k}$ et notons M_j le $j^{\text{ième}}$ moment de l'échantillon, c'est-à-dire :

$$M_j = \frac{1}{n} \sum_{i=1}^n X_i^j$$

Identifions les k premiers moments de la fonction de densité avec ceux des échantillons, et nous obtenons

$$M_j = \mu_j(\theta_1, \dots, \theta_k), j = 1, \dots, k \quad (1.5)$$

Nous supposons qu'il y a une seule solution $\hat{\theta}$ pour les $\theta_j (j = 1, \dots, k)$. Nous disons que $(\hat{\theta}_j)_{j=1}^k$ est l'estimateur de $(\theta_j)_{j=1}^k$ par la méthode des moments. Cet

estimateur est obtenu en identifiant les moments de la population par les moments des échantillons. Donnons un exemple.

Soient $X_i, i = 1, \dots, n$ un échantillon aléatoire d'une distribution normale de moyenne μ et de variance σ^2 . Donc ici $(\theta_1, \theta_2) = (\mu, \sigma)$. Estimons μ et σ par la méthode des moments. Rappelons que $\sigma^2 = \mu_2 - \mu_1^2$ et $\mu = \mu_1$, alors les équations (1.5) deviennent :

$$\begin{cases} M_1 &= \mu_1 &= \mu \\ M_2 &= \mu_2 &= \sigma^2 + \mu^2 \end{cases}$$

et nous obtenons

$$\mu = M_1 = \bar{X} = \frac{1}{n} \sum_{i=1}^n X_i$$

et

$$\sigma = \sqrt{M_2 - \mu^2} = \sqrt{\frac{1}{n} \sum_{i=1}^n X_i^2 - \bar{X}^2} = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n}}.$$

Ce résultat est conforme à l'intuition.

1.4.4.2 Méthode des moments généralisée

La méthode des moments classique peut se généraliser. Les moments employés peuvent être pondérés par une autre fonction de densité $g(x)$, et les équations deviennent :

$$\frac{1}{n} \sum_{i=1}^n X_i^j g(X_i) = E(X^j g(X)) = \int x^j g(x) f(x) dx$$

Ceci est nécessaire dans le cas où les moments non pondérés n'existent pas, comme celui traité dans [NW94]. Grâce à la fonction de pondération g , pour cette généralisation, f n'est plus forcément une fonction de densité, elle peut même être non intégrable.

Netanyahu [NW94] a présenté une méthode analytique robuste pour l'ajustement de droite. Il suppose que les données peuvent être modélisées par un mélange de deux distributions statistiques, l'une gaussienne et l'autre uniforme.

Dans le cas unidimensionnel, étant donné un ensemble de mesures, on suppose que celles-ci suivent la distribution mélangée suivante :

$$f(x) = \alpha_u f_u(x) + \alpha_n f_n(x) \tag{1.6}$$

où α_u, α_n sont positifs, f_u et f_n sont les parties uniforme et normale de f , respectivement. Plus précisément, $f_u(x) = 1$ et $f_n(x) = e^{-\frac{(x-\mu)^2}{2\sigma^2}}$, $\sigma > 0$.

Comme la partie uniforme n'est pas intégrable, on emploie la méthode des moments à $f(x)g(x)$, où $g(x) = C_g e^{-\frac{x^2}{2\sigma_g^2}}$ ($C_g > 0$ et $\sigma_g > 0$).

Soit $m^{(k)}$ le moment d'ordre k de $f(x)g(x)$,

$$m^{(k)} = \int_{-\infty}^{\infty} x^k f(x)g(x)dx.$$

Notons $m_u(k) = \alpha_u \int_{-\infty}^{\infty} x^k f_u(x)g(x)dx$, $m_n(k) = \alpha_n \int_{-\infty}^{\infty} x^k f_n(x)g(x)dx$. Alors $m^{(k)} = m_u^{(k)} + m_n^{(k)}$.

Ces moments théoriques peuvent être comparés avec ceux observés. Soit $M^{(k)}$ le moment observé d'ordre k , $M^{(k)} = \sum_{i=1}^N x_i^k g(x_i)$, où $\{x_i, i = 1, \dots, N\}$ est l'ensemble donné. En notant $\hat{\sigma}^2 = \frac{\sigma^2 \sigma_g^2}{\sigma^2 + \sigma_g^2}$, $\hat{\mu} = \frac{\mu \hat{\sigma}^2}{\sigma^2}$, $m_u^{(0)} = C_u$, $m_n^{(0)} = C_n$, après quelques calculs algébriques, on obtient :

$$\begin{aligned} m_u^{(1)} &= 0, \\ m_n^{(1)} &= C_n \hat{\mu}, \\ m_u^{(2)} &= C_u \sigma_g^2, \\ m_n^{(2)} &= C_n (\hat{\mu}^2 + \hat{\sigma}^2), \\ m_u^{(3)} &= 0, \\ m_n^{(3)} &= C_n (\hat{\mu}^3 + 3\hat{\mu}\hat{\sigma}^2). \end{aligned}$$

En comparant les moments empiriques avec les théoriques, on obtient :

$$\begin{aligned} M^{(0)} &= m^{(0)} = m_u^{(0)} + m_n^{(0)} = C_u + C_n \\ M^{(1)} &= m^{(1)} = m_u^{(1)} + m_n^{(1)} = C_n \hat{\mu} \\ M^{(2)} &= m^{(2)} = m_u^{(2)} + m_n^{(2)} = C_u \sigma_g^2 + C_n (\hat{\mu}^2 + \hat{\sigma}^2) \\ M^{(3)} &= m^{(3)} = m_u^{(3)} + m_n^{(3)} = C_n (\hat{\mu}^3 + 3\hat{\mu}\hat{\sigma}^2) \end{aligned}$$

avec $\hat{\sigma}^2 = \frac{\sigma^2 \sigma_g^2}{\sigma^2 + \sigma_g^2}$, $\hat{\mu} = \frac{\mu \hat{\sigma}^2}{\sigma^2}$, et C_u, C_n contiennent α_u, α_n .

En résolvant ce système, on obtient une équation quadrique en $\hat{\mu}$:

$$2M^{(1)}\hat{\mu}^2 - 3(M^{(2)} - M_{(0)}\sigma_g^2)\hat{\mu} + (M^{(3)} - 3M^{(1)}\sigma_g^2) = 0$$

Une fois $\hat{\mu}$ trouvé, les quatre inconnues α_u, α_n, μ et σ peuvent être aussi facilement trouvées.

Cette méthode peut s'étendre au cas bidimensionnel, et son application à un exemple montre clairement qu'elle est meilleure que la méthode des moindres carrés [NW94] sous l'hypothèse que le bruit est un mélange d'une partie gaussienne et une partie uniforme.

1.5 Sommaire d'estimateurs robustes

À partir de l'utilisation de la transformation de Hough et de la technique de RanSaC, beaucoup de méthodes robustes se sont proposées en vision. Pendant la dernière décennie, les M-estimateurs et le LMedS estimateur se sont largement employés. Depuis le début de cette décennie, on a inventé des nouveaux estimateurs pour surmonter des limites des anciens estimateurs. Parmi eux, CBD cherche à choisir automatiquement le bon modèle parmi une classe de modèles, un problème encore ouvert en statistique. Les MF-, MINPRAN-, RESC-, ALKS-, MUSE-estimateurs essaient de trouver un bon ajustement où le pourcentage d'erreurs grossières dépasse 50%. Et MOM est un estimateur analytique qui trouve bien l'ajustement quand le bruit est un mélange d'une partie gaussienne et une partie uniforme.

Dans des applications, on se trouve souvent dans un cadre où les données appartiennent à plusieurs classes, où aucun modèle ne correspond à la majorité des données, et où les divers modèles sont de types différents, comme, par exemple, dans le cas de points appartenant à des droites et des plans, où l'on veut chercher ces droites et ces plans. C'est encore un problème ouvert, mais qui se rencontre très souvent en vision.

1.6 Domaines d'applications en vision

En vision, il y a deux types d'erreurs : les erreurs liées à la mesure, qui sont souvent modélisées comme des erreurs gaussiennes centrées ; et les erreurs aberrantes ou grossières, venant de faux appariements, ou de cas où plusieurs modèles interviennent et où la segmentation n'est pas bien faite. On peut citer des exemples d'erreurs aberrantes :

- Un mauvais pixel blanc ou noir

Dans une caméra, une mauvaise valeur de niveau de gris d'un pixel peut être produite dans chaque image, par un défaut du capteur électronique. Ce genre de valeurs n'obéissent à aucune distribution d'erreurs avec de bonnes

propriétés. Ce type d'erreurs peut provoquer de faux éléments de contour, par exemple.

– Faux appariements

En stéréo, les faux appariements sont inévitables. Ils peuvent détruire le résultat de la reconstruction de surface, sauf si l'on utilise des méthodes robustes [SS90].

– L'occultation dans l'estimation du flot optique

L'estimation du flot optique concerne le calcul de deux composantes de la vitesse à partir d'une seule contrainte du flot optique d'image [HS80]. L'occultation crée de fausses mesures [Sch86] et la combinaison des contraintes du flot optique de différentes régions du mouvement peut produire des erreurs extrêmes dans l'estimation du champ de flot optique [Sch89].

– Structure à partir du mouvement

On suppose souvent que tous les points appartiennent au même objet rigide ou quasiment rigide [Ull84, GH87]. Si les objets dans la scène ont des mouvements différents, les points d'objets différents sont aberrants.

Grâce aux méthodes robustes, on peut espérer de surmonter les difficultés causées par telles erreurs. Les méthodes robustes sont employées pour le traitement d'images [BBW88], la détection de contours [Min94], la segmentation [JMB91], l'estimation de la matrice fondamentale [ZDFL94], l'appariement visuel [LMR95, LM97], l'estimation de pose [HJL⁺89], la reconstruction de surface [SS90], et enfin l'estimation du tenseur trilinéaire [TZ96].

Chapitre 2

Appariement visuel : état de l'art

Considérons un système de vision binoculaire qui observe une scène tridimensionnelle. Un point dans l'espace tridimensionnel se projette sur les plans images des deux caméras. Pour une projection donnée dans une image, celle correspondante dans une autre image est appelée son homologue. On appelle *disparité* le déplacement à effectuer sur la projection pour qu'elle se superpose sur son homologue. En général, la disparité est définie par un vecteur bidimensionnel. Le calibrage faible (la détermination de la géométrie épipolaire) permet de réduire la recherche de correspondance à un segment d'une droite appelée droite épipolaire. Après une transformation géométrique, la rectification [Aya90] peut s'appliquer aux deux images, pour que les droites épipolaires conjuguées associées à deux pixels homologues $m^g(i, j)$ et $m^d(i, j')$ se confondent avec la ligne i de chaque image. La disparité peut être représentée par un nombre réel dans ce cas et on est donc réduit à un appariement unidimensionnel.

L'appariement visuel entre les images, aussi appelé problème de mise en correspondance, est l'étape décisive de la reconstruction tridimensionnelle (3D) et l'estimation du mouvement 3D [HA89].

Malgré des efforts considérables qu'on a déjà mis par la communauté de vision, ce problème continue à être étudié et le sera encore largement dans le futur. C'est un problème difficile et qui n'aura peut-être pas de réponse définitive. Les raisons en sont les suivantes :

1. Bruit de l'image :

Le bruit de l'image vient de l'échantillonnage (on a seulement les niveaux de gris sur les points des coordonnées entières), de la quantification (le niveau de gris est normalement représenté sur 8 bits), et du bruit électronique de la caméra (habituellement les 2 derniers bits ne sont pas significatifs).

2. Occultation :

Un point dans une image n'apparaît pas forcément dans une autre. Soit parce qu'il y a des obstacles entre ce point et l'autre, soit parce que sa projection sur l'autre est en dehors du support de l'image. L'occultation pose souvent des difficultés pour l'appariement visuel.

3. Distorsion perspective :

L'image d'un objet peut être géométriquement déformée dans une autre, quand l'angle entre deux vues est grande. Par exemple, un rectangle dans une image peut apparaître comme un trapèze dans une autre image.

4. Variation photométrique :

Entre deux vues, les conditions d'éclairage de prise de vue sont différentes ; cette variation peut parfois être approchée par une translation et un changement d'échelle des niveaux de gris entre les deux images. Un pixel dans une image peut être plus clair que dans une autre, à cause du changement de l'éclairage ou de l'ouverture d'iris, et ce changement peut être approximé par une addition de niveau de gris. Le contraste dans une image peut être plus fort que dans une autre, à cause des différences des dynamiques des caméras, et ce changement peut être approximé par un facteur d'échelle. Localement, les niveaux de gris entre deux images peuvent être approximativement liés par une relation affine, c'est-à-dire, une multiplication plus une addition. C'est une hypothèse implicite de beaucoup de corrélations classiques.

5. Répétition de textures :

Des textures similaires peuvent exister, ce qui crée des ambiguïtés pour l'appariement. Pour ces répétitions, il faut utiliser des contraintes globales pour surmonter les ambiguïtés.

6. Absence de texture :

Dans les zones uniformes, il est impossible de déterminer quel point dans une image doit être mis en correspondance avec un point dans une autre image : ils sont tous localement identiques. La solution consiste à faire des hypothèses globales sur ces régions pour déduire des solutions de mise en correspondance, comme dans le cas de répétitions de textures. C'est un cas spécial de la répétition de textures, dans une zone où la texture est partout nulle.

Dans ce chapitre, nous présentons d'abord les méthodes discrètes dans le domaine spatial (section 2.1), ensuite nous présentons la technique de programmation dynamique pour l'appariement visuel (section 2.2), un résumé est donné dans la section 2.3.

2.1 Méthodes discrètes

Ces méthodes se divisent en deux familles [FFH⁺92] : l'appariement de pixels [AG92, HR93, SM95] et l'appariement de caractéristiques [Aya90]. Les méthodes d'appariement de pixels essaie d'apparier les points dans deux images en comparant leur ressemblance, soit par corrélation [AG92], soit par la distance de Hausdorff [HR93], soit par les invariants locaux [SM95]. Ces méthodes donnent des appariements relativement denses si la scène observée est suffisamment texturée. Les méthodes d'appariement de caractéristiques emploient des informations extraites des images, souvent des segments [Aya90]. Ils sont plus efficaces en temps et en espace, mais donnent des appariements épars ; des interpolations sont donc nécessaires pour la reconstruction tridimensionnelle.

Les méthodes de caractéristiques sont présentées dans [Aya90]. Les méthodes d'appariement de pixels sont présentées dans ce chapitre, qui se divisent en trois classes : la corrélation [AG92], la distance de Hausdorff [HR93] et les invariants locaux [SM95].

2.1.1 Corrélation

La corrélation est une technique fréquemment utilisée pour le problème d'appariement. On classe les corrélations en quatre familles : corrélations classiques [Ack84, AG92, Bra95, BM95b, For82, For84, Gru85, GS91, HB95, OK92, LG94, RBM94] corrélations non paramétriques [ZW94, BN96], corrélation partielle [LMR95, LM97], corrélation avec masque de convolution [LM96]. Nous présentons les corrélation des deux premières classes dans ce chapitre, et les deux dernières seront présentées dans les chapitre 3 et 4 respectivement.

2.1.1.1 Corrélations classiques [AG92, BM95b]

Avec quelques variantes, toutes ces méthodes utilisent les intensités lumineuses (ou leurs gradients, ...).

Notations On introduit d'abord des notations (Figure 2.1) :

- $S(x, y)$: fenêtre de recherche d'intérêt (RDI), la seule partie d'image considérée pour l'appariement.
- $R(u, v)$: modèle, une partie d'image de dimension $(ulen + 1)(vlen + 1)$.

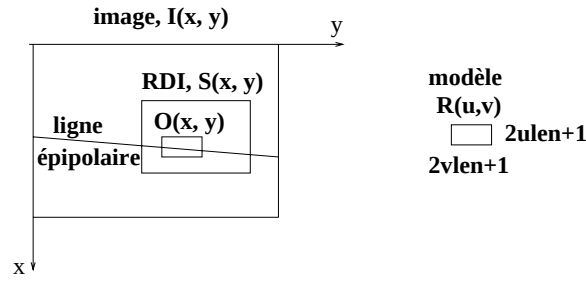


FIG. 2.1 – Définitions pour la corrélation.

- $O(x+x_0, y+y_0)$: fenêtre d'objet. partie de la RDI, ayant la même dimension que le modèle et considérée comme la vraie position d'appariement.
- $\overline{S(x, y)}, \overline{R}$: moyenne locale de $S(x, y)$ et $R(u, v)$.

$$\overline{S(x, y)} = \frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} S(x+u, y+v)}{(vlen+1)(ulen+1)} \quad \overline{R} = \frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} R(u, v)}{(vlen+1)(ulen+1)}$$

- $S_{sobel}(x, y), R_{sobel}(u, v)$: orientation d'image, calculée par l'opérateur Sobel.¹

Mesure de ressemblance Différentes mesures de ressemblance entre deux signaux sont rappelées dans le tableau 2.1 (emprunté de [AG92]).

1. L'orientation à un pixel peut être calculé comme suit :

$$\begin{aligned} H_{sobel}(x, y) &= (S(x+1, y+1) + 2S(x, y+1) + S(x-1, y+1)) - \\ &\quad (S(x+1, y-1) + 2S(x, y-1) + S(x-1, y-1)) \\ V_{sobel}(x, y) &= (S(x+1, y+1) + 2S(x+1, y) + S(x+1, y-1)) - \\ &\quad (S(x-1, y+1) + 2S(x-1, y) + S(x-1, y-1)) \\ S_{sobel}(x, y) &= \arctan\left(\frac{V_{sobel}(x, y)}{H_{sobel}(x, y)}\right) \end{aligned}$$

TAB. 2.1 – Définition des mesures de ressemblance. $R(u, v)$ définit le modèle de niveau de gris, $S(x, y)$ définit un signal de niveau de gris dans une fenêtre contenant dans la région d'intérêt (RDI, une partie d'image) d'une autre image, $ulen+1$ et $vlen+1$ sont le hauteur et le largeur de R et S , $n = (vlen+1)(ulen+1)$.

Nom	Abréviation	Définition
Corrélation normalisée	$NCC(x, y)$	$\frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} R(u, v) \cdot S(x+u, y+v)}{\sqrt{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} R^2(u, v) \sum_{v=0}^{vlen} \sum_{u=0}^{ulen} S^2(x+u, y+v)}}$
Corrélation normalisée centrée	$ZNCC(x, y)$	$\frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} (R(u, v) - \bar{R}) \cdot (S(x+u, y+v) - \bar{S}(x, y))}{\sqrt{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} (R(u, v) - \bar{R})^2 \cdot \sum_{v=0}^{vlen} \sum_{u=0}^{ulen} (S(x+u, y+v) - \bar{S}(x, y))^2}}$
Somme des différences aux carrées	$SSD(x, y)$	$\sqrt{\frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} (R(u, v) - S(x+u, y+v))^2}{n}}$
Somme des valeurs absolues des différences	$SAD(x, y)$	$\frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} R(u, v) - S(x+u, y+v) }{n}$
Somme des différences aux carrées centrées	$ZSSD(x, y)$ ²	$\sqrt{\frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} ((R(u, v) - \bar{R}) - (S(x+u, y+v) - \bar{S}(x, y)))^2}{n}}$
Somme des valeurs absolues des différences centrées	$ZSAD(x, y)$	$\frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} (R(u, v) - \bar{R}) - (S(x+u, y+v) - \bar{S}(x, y)) }{n}$
Somme des différences aux carrées normalisées	$LSSD(x, y)$	$\sqrt{\frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} (\frac{R(u, v)}{\bar{R}} - \frac{S(x+u, y+v)}{\bar{S}(x, y)})^2}{n}}$
Somme des valeurs absolues des différences normalisée	$LSAD(x, y)$	$\frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} \frac{R(u, v)}{\bar{R}} - \frac{S(x+u, y+v)}{\bar{S}(x, y)} }{n}$
Seitz 1	$SES1(x, y)$	$\sqrt{\frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} (R_{sobel}(u, v) - S_{sobel}(x+u, y+v))^2}{n}}$
Seitz 3	$SES2(x, y)$	$\frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} R_{sobel}(u, v) - S_{sobel}(x+u, y+v) }{n}$

2. $ZSSD$ dans cette formule est une estimation optimale de l'écart type du bruit dans le cas gaussien, mais elle est biaisée. L'estimation non biaisée est

$$\sqrt{\frac{\sum_{v=0}^{vlen} \sum_{u=0}^{ulen} ((R(u, v) - \bar{R}) - (S(x+u, y+v) - \bar{S}(x, y)))^2}{n-1}}.$$

Dans le tableau 2.1, normalisation signifie la division de chaque élément d'un signal par une constante pour avoir une norme (L_2 ou L_1) de 1, et centralisation signifie la soustraction de chaque élément d'un signal par sa moyenne pour avoir la moyenne 0.

Nous discutons maintenant ces corrélations. Dans les discussions suivantes, les signaux unidimensionnels sont pris, ce qui simplifie l'explication et ne change *rien* essentiel.

Supposons que nous avons deux signaux quelconques a et b , et notons $a = (a_i)_{i=1}^n, b = (b_i)_{i=1}^n$. Alors :

- $a_c(i) = a(i) - \bar{a}$ est la centralisation de a , et $\bar{a}_c = 0$.
- $a_{cn2}(i) = \frac{a_c(i)}{\|a_c\|_{L_2}}$ est la normalisation L_2 de a_c et $\overline{a_{cn2}} = 0, \|a_{cn2}\|_{L_2} = 1$.
- $a_{n1}(i) = \frac{a(i)}{\bar{a}}$ est la normalisation L_1 de a et $\|a_{n1}\|_{L_1} = 1.0$.

1. SSD, NCC et SAD

Ces trois mesures définies dans le tableau 2.1 peuvent s'interpréter géométriquement selon la figure 2.2.

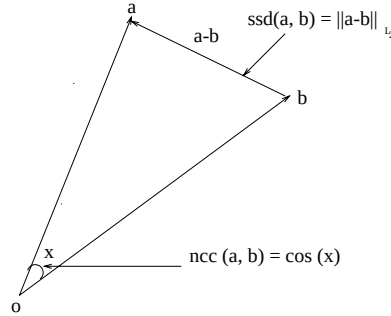


FIG. 2.2 – Sens géométrique de SSD et NCC.

- $SSD(a, b) = \|a - b\|_{L_2}$ est la distance euclidienne L_2 entre a et b .
- $NCC(a, b) = \frac{(a, b)}{\|a\|_{L_2} \|b\|_{L_2}} = \cos(x)$ est le cosinus de l'angle entre a et b .
- $SAD(a, b) = \|a - b\|_{L_1}$ est la distance L_1 entre a et b .

2. ZSSD,ZNCC, ZSAD, LSSD et LSAD

Leur interprétation est la suivante :

- $ZSSD(a, b) = \|a_c - b_c\|_{L_2}$ est la distance euclidienne L_2 entre a_c et b_c .

- $ZNCC(a, b) = \cos(ang_c) = \cos(ang_{cn2})$, où ang_c est l'angle entre a_c et b_c , ang_{cn2} est l'angle entre a_{cn2} et b_{cn2} .
- $ZSAD = \|a_c - b_c\|_{L_1}$ est la distance L_1 entre a_c et b_c .
- $LSSD = \|a_{n1} - b_{n1}\|_{L_2}$ est la distance euclidienne L_2 entre a_{n1} et b_{n1} .
- $LSAD = \|a_{n1} - b_{n1}\|_{L_1}$ est la distance L_1 entre a_{n1} et b_{n1} .

3. SES1 et SES2

Ce sont les «SAD» et «SSD» de signaux de *direction d'image*.

On peut aussi proposer $LSAD$, $ZSAD$, $LSSD$, $ZSSD$... de ces signaux.

4. Une relation

On voit qu'il y a deux type de mesures : *distance* et *angle*.

SSD , $ZSSD$, $LSSD$, SAD , $ZSAD$, $LSAD$ sont de type *distance*, NCC et $ZNCC$ sont de type *angle*.

Et on a la relation suivante :

Soient a et b deux vecteurs unitaires (pour la norme euclidienne), alors plus que leur angle $\angle(a, b)$ est petit, plus leur distance est faible. Ici, $\angle(a, b)$ définit l'angle entre eux, et (a, b) est leur produit scalaire, on a $\cos(\angle(a, b)) = (a, b)$.

En termes plus mathématiques : si $\|a\|_{L_2} = \|b\|_{L_2} = 1.0$, alors $\|a - b\|_{L_2}^2 = 2 - 2(a, b) = 2 - 2\cos \angle(a, b)$, donc, plus $\angle(a, b)$ est petit, plus $\cos \angle(a, b)$ est grand et plus $\|a - b\|_{L_2}^2$ est petit.

Donc $ZNCC(a, b) = \cos \angle(a_{cn2}, b_{cn2})$ est équivalent à $SSD(a_{cn2}, b_{cn2})$ (aussi à $ZSSD(a_{cn2}, b_{cn2})$).

5. Prétraitement du signal

a_c , a_{cn2} et a_{n1} (resp. b_c , b_{cn2} et b_{n1}) sont des prétraitements du signal, pour ne pas être affecté par l'effet de la *transformation affine* des signaux.

6. Une propriété de $ZSSD(a, b)$

$$ZSSD(a, b) = \min_{\lambda} SSD(a - b - \lambda e) \quad e = (1, \dots, 1).$$

En termes statistiques, SSD et ZSSD ont une meilleure propriété que nous allons détailler. Si a (resp. b) est un échantillonnage indépendant de distribution A (resp. B), alors $SSD(a, b)$ (resp. $ZSSD(a, b)$) est une estimation non biaisée de $\|A - B\|_{L_2}$ (resp. $\|A - E(A) - (B - E(B))\|_{L_2}$). L'inconvénient de SSD et ZSSD est qu'ils comptent toujours l'effet d'échelle (a et b sont différents si $b(i) = \lambda a(i)$ et $\lambda \neq 1$). Pour plus de discussions de SSD et ZSSD, voir l'annexe B.

On donne un résumé de ces mesures de corrélation dans le tableau 2.2. Quatre termes sont décrits pour chaque corrélation :

- *signaux* utilisés (niveaux de gris ou son gradient),
- *mesure* utilisée (angle, distance L_2 , distance L_1),
- invariance à une translation des intensités lumineuses, ce phénomène est essentiel dans le cas de variations des intensités des sources lumineuses,
- invariance à un effet d'échelle des intensités lumineuses, ce phénomène est essentiellement causé par le changement de dynamique des caméras.

	signaux utilisés	mesure	translation des intensités lumineuses	échelle des intensités lumineuses
NCC	niveaux de gris	angle	V	I
ZNCC	niveaux de gris	angle	I	I
SSD	niveaux de gris	distance L_2	V	V
SAD	niveaux de gris	distance L_1	V	V
ZSSD	niveaux de gris	distance L_2	I	V
ZSAD	niveaux de gris	distance L_1	I	V
LSSD	niveaux de gris	distance L_2	V	I
LSAD	niveaux de gris	distance L_1	V	I
SES1	direction des gradients	distance L_2	I	I
SES2	direction des gradients	distance L_1	I	I

TAB. 2.2 – *Invariance des corrélations classiques par la transformation de signal : I : invariant, V : variant.*

Corrélation de Brunelli Brunelli a proposé des corrélations moins sensibles au bruit dans [BM95b]. Il y en a trois types :

$$\begin{aligned}
 g(a, b) &= 1 - \frac{\sum_i |a_{nc,i} - b_{nc,i}|}{\sum_i |a_{nc,i}| + |b_{nc,i}|}, \\
 l(a, b) &= \frac{1}{n} \sum_i \left(1 - \frac{|a_{nc,i} - b_{nc,i}|}{|a_{nc,i}| + |b_{nc,i}|} \right), \\
 R(a, b) &= \frac{\sigma_{th}^2 (a+b) - \sigma_{th}^2 (a-b)}{\sigma_{th}^2 (a+b) + \sigma_{th}^2 (a-b)}.
 \end{aligned}$$

Ici, $a_{nc,i}$ (resp. $b_{nc,i}$) est le $i^{\text{ième}}$ composant de a_{nc} (resp. b_{nc}), la normalisation et centralisation de a (resp. b). σ_{th} est le *tanh*-estimation de paramètre d'échelle [HRRS86]. Brunelli a utilisé ces corrélations pour des applications de reconnaissance de visages et de livres [BM95b].

Quelques remarques

- Distorsion perspective.

Les méthodes citées ci-dessus supposent qu'il n'y a que peu de distorsions perspectives entre deux images. Dans le cas contraire, il est nécessaire de considérer la déformation des fenêtres entre deux images. Avec cette considération, on peut espérer d'obtenir la précision sous-pixelique pour l'appariement [Ack84, Bra95, Gru85, GS91, RBM94].

- Méthodes pyramidales.

Comme pour beaucoup d'autre algorithmes de vision en bas niveau, il est aussi possible d'utiliser des méthodes pyramidales pour la corrélation. Ces méthodes accélèrent beaucoup la vitesse, elles sont décrites dans des ouvrages, comme [HM93].

- Discontinuité de disparités.

Les corrélations classiques, utilisées depuis longtemps, se dégradent lorsqu'il y a des discontinuités de la disparité [ZW94, LMR95]. Elles supposent que la disparité reste quasiment constante et elles échouent près des contours d'occultation où la discontinuité de la disparité se présente. Ce phénomène, appelée par Zabih [ZW94] le *factionalisme*, est appelé par nous [LMR95, LM97] l'*occultation partielle*. Ces auteurs ainsi que Bhat [BN96] ont donc proposé autres méthodes «non paramétriques» et «partielles» pour vaincre cette difficulté ; elles sont expliquées ci-dessous et dans le chapitre 3 respectivement.

2.1.1.2 Corrélations non paramétriques

Nous présentons d'abord les méthodes de Zabih [ZW94] : rang et census, puis la méthode de Bhat [BN96].

Les méthodes de Zabih Zabih [Zab94] a développé une approche basée sur des mesures de transformations locales *non paramétriques*. Les mesures non paramétriques sont tolérantes aux données aberrantes. Ces mesures sont basées sur la statistique non paramétrique qui utilise l'ordre des données, au lieu des données elles-mêmes. Nous expliquons d'abord ces mesures originales, puis introduisons une modification.

Les transformations de rang et de census Deux transformations locales non paramétriques sont introduites. La première, qui s'appelle la transformation

de *rang*, est une mesure locale des intensités locales ; et la deuxième, appelé la transformation de *census*, est un résumé non paramétrique de la structure spatiale locale.

Soit P un pixel, $I(P)$ son intensité, et $N(P)$ l'ensemble de pixels dans son voisinage. Toutes les transformations locales non paramétriques considérées dépendent de la comparaison de $I(P)$ et les intensités de pixels dans $N(P)$. Soit $\xi(P, P')$ tel que $\xi(P, P') = 1$ si $I(P') < I(P)$ et 0 sinon. Les transformations locales non paramétriques dépendent seulement de l'ensemble de comparaison, qui est :

$$\Xi = \bigcup_{P' \in N(P)} (P', \xi(P, P'))$$

La transformation de *rang* est définie comme le nombre de pixels dans la fenêtre locale dont l'intensité est inférieure à celle du pixel central. Formellement, elle est :

$$R(P) = ||P' \in N(P) | I(P') < I(P)|| = \sum_{P' \in N(P)} \xi(P, P')$$

Pour la transformation de *rang*, la corrélation L_1 (minimisation de la somme des valeurs absolues des différences) est utilisée sur l'image transformée.

La transformation de *census* est une fonction du voisinage local d'un pixel P à une chaîne de bits représentant l'ensemble de pixels voisins dont l'intensité est inférieure à celle de P . Soit $N(P) = P \oplus D$ où $\oplus$ est la somme de Minkowski et D l'ensemble de déplacement, et $\otimes$ la concaténation. La transformation de *census* est définie comme :

$$R_\tau(P) = \bigotimes_{[i,j] \in D} \xi(P, P \oplus [i, j])$$

La distance de Hamming est employée après la transformation de *census*, et l'algorithme est réalisé par la minimisation de cette distance.

Donnons maintenant un exemple : Ces deux transformations tolèrent les données aberrantes de la manière illustrée dans l'exemple ci-dessous. Considérons une fenêtre de dimension 3×3 d'une image dont les intensités sont :

$$\begin{array}{ccc} 127 & 127 & 129 \\ 126 & 128 & 129 \\ 127 & 131 & A \end{array}$$

Considérons l'ensemble d'effet de plusieurs mesures paramétriques et non paramétriques, calculées au centre de la fenêtre, quand A parcourt ses 256 valeurs possibles.

La moyenne de cette fenêtre varie de 114 à 142, et la variance se place entre 2 et 1823. Ces mesures paramétriques présentent une variation continue quand A change.

Les transformations non paramétriques sont plus stables. Tous les éléments de Ξ sauf un restent fixés quand A change. Ξ sera :

$$\begin{array}{ccc} 1 & 1 & 0 \\ 1 & & 0 \\ 1 & 0 & a \end{array}$$

où a est 1 si $A < 128$, 0 sinon. La transformation de *census* est $\{1, 1, 0, 1, 0, 1, 0, a\}$ et celle de *rang* est 5 si $A < 128$, 4 sinon.

Cette comparaison montre la tolérance des mesures non paramétriques : une minorité de pixels peuvent être aberrants, mais ses effets sur la transformation de *census* et de *rang* sont limités par la taille de cette minorité.

Dans cette approche, le centre de la transformation joue un rôle plus important que les pixels autours, cette importance a des effets négatifs s'il y a des bruits importants sur le centre, mais cette importance donne aussi plus de discriminance pour trouver le bon appariement entre les candidats voisins assez similaires [Page 41],[Zab94].

Notre modification Basée sur son idée, nous avons développé une transformation locale qui capture plus d'informations, mais plus compliquée. Au lieu de classer $I(P) < I(P')$ par 0 ou 1, nous remplaçons cette valeur binaire par une probabilité de $I(P) < I(P')$, plus précisément :

Étant donné un pixel P et un pixel voisin $P \oplus [i, j]$, plus la valeur observée de $I(P) - I(P \oplus [i, j])$ est grande, plus la confiance nous avons en $I(P \oplus [i, j]) < I(P)$ est large. Bien sûr, cette confiance dépend aussi de l'écart-type de $I(P) - I(P \oplus [i, j])$.

Posons la probabilité que $I(P \oplus [i, j]) < I(P)$ d'être $G(s/\sigma)$, où s est la valeur observée de $I(P) - I(P \oplus [i, j])$, σ est l'estimation de l'écart-type de $I(P) -$

$I(P \oplus [i, j])$, et G la fonction de distribution gaussienne. σ est estimé comme suit :

- estimer σ_1 =l'écart-type de $I(P)$
- estimer σ_2 =l'écart-type de $I(P \oplus [i, j])$
- estimer r =le coefficient de corrélation entre $I(P)$ et $I(P \oplus [i, j])$
- l'écart-type de $I(P) - I(P \oplus [i, j])$ peut être calculé comme $\sqrt{\sigma_1^2 + \sigma_2^2 - 2r\sigma_1\sigma_2}$

Selon notre observation, nous avons mis $\sigma_1 = \sigma_2 = 5$, $r = \rho_h^i \rho_v^j$, où $\rho_h = \rho_v = 0.9$.

Cette méthode est plus souple, car elle considère aussi l'effet du bruit dans l'image, mais elle est aussi beaucoup plus coûteuse. De plus, σ_1 , σ_2 et r sont difficiles à estimer. Cette méthode n'est donc pas appliquée pour la suite.

Les méthodes de Bhat Bhat a développé une autre méthode de corrélation non paramétrique [BN96]. Il ordonne les valeurs d'intensité dans chaque fenêtre, et obtient une permutation. En utilisant une nouvelle distance entre les permutations, un nouveau coefficient de corrélation est obtenu. Donnons plus de détails maintenant.

Pour deux ensembles des valeurs d'intensités $(I_1^i)_{i=1}^n$ et $(I_2^i)_{i=1}^n$. Soient π_1^i et π_2^i le rang de I_1^i (I_2^i) parmi les données I_1 (I_2).

Définissons une permutation composée comme suit : $s^i = \pi_2((\pi_1^{-1})(i))$, où π_1^{-1} note la permutation inverse de π_1 . Intuitivement, s^i est le rang de pixel dans I_2 qui correspond au pixel dont le rang est i dans I_1 .

Si I_1 et I_2 sont parfaitement corrélés, S doit être $u = (1, 2, \dots, n)$.

Nous obtenons une distance entre π_1 et π_2 en définissant une distance entre s et u . La déviation d_m^i sur chaque s_i est définie comme le nombre de s_j ($j = 1 \dots i$) plus grand que i . Formellement, $d_m^i = \sum_{j=1}^i 1_{[i+1, +\infty[}(s_j)$, où 1 est la fonction indicatrice.

Une mesure de corrélation $\kappa = \kappa(I_1, I_2)$ est définie comme :

$$\kappa(I_1, I_2) = 1 - \frac{2 \max_{i=1}^n d_m^i}{\lfloor \frac{n}{2} \rfloor}.$$

Si I_1 et I_2 sont parfaitement corrélées, alors $\kappa = 1$. κ possède les propriétés suivantes :

- il est indépendant d'effet d'échelle et de translation des intensités lumineuses.
- il est normalisé : $-1 \leq \kappa \leq 1$.
- il est symétrique, c'est-à-dire $\kappa(I_1, I_2) = \kappa(I_2, I_1)$.
- $\kappa(f(I_1), h(I_2)) = \kappa(I_1, I_2)$ si f et h sont deux fonctions strictement croissantes, ce qui le rend invariant aux effets non linéaires, comme la variation de gamma [BN96].

Une autre mesure $\chi(I_1, I_2)$ moins coûteuse est définie comme :

$$\chi(I_1, I_2) = 1 - \frac{2d_m^{\lfloor \frac{n}{2} \rfloor}}{\lfloor \frac{n}{2} \rfloor}$$

Nous montrons l'insensibilité de ces deux mesures au bruit et à la distorsion du rang qui arrive dans le cas de réflexion spéculaire et des discontinuités de disparité.

Un exemple Considérons l'exemple suivant d'une fenêtre de référence de dimension 3×3 avec les intensités I_1 :

	M	
10	30	70
20	50	80
40	60	100

Sa matrice de rang π_1 est :

1	3	7
2	5	8
4	6	9

Si une autre fenêtre dans la deuxième image est :

10	30	70
20	50	80
40	60	A

Si A est entre 81 et 255, π_2 est toujours :

1	3	7
2	5	8
4	6	9

Il est identique à π_1 , donc $\kappa = 1$.

Si A prend la valeur 75, alors κ descend à 0,8. Si A prend une valeur entre 0 et 10, alors π_2 devient

2	4	8
3	6	9
5	7	1

Heureusement, κ est encore 0,8, au contraire du τ de Kendall et de ρ de Spearman, qui descendent à 0,556 et 0,4, respectivement. Si A prend 0, alors NCC devient 0,311.

Nous pouvons montrer les propriétés similaires pour χ . En résumé, ces deux mesures sont robustes à la réflexion spéculaire et aux discontinuités.

2.1.1.3 Résumé

On réalise la corrélation en plusieurs étapes :

- Choix de fenêtre

On peut choisir le plus souvent une fenêtre rectangulaire (ou carrée) ou non rectangulaire ([LG94]) et aussi la dimension de la fenêtre [OK92]).

- Choix de la région d'intérêt : limitation de la zone d'appariement

On peut incorporer la contrainte épipolaire et la contrainte d'ordre. La contrainte épipolaire réduit le problème en deux dimensions à un problème en une dimension, donc augmente la vitesse et aussi la robustesse, mais il exige de plus une bonne géométrie épipolaire. La contrainte d'ordre est une contrainte heuristique, qui peut être intégrée dans la technique de programmation dynamique, présentée dans la section 2.2.

- Choix des primitives à appairer

On peut faire la corrélation non seulement sur l'intensité de l'image, mais aussi sur la direction du gradient (SES1 et SES2), etc.

- Choix de la mesure de ressemblance

Pour les mises en correspondance, il s'agit d'un problème encore ouvert.

ZSSD est une estimation non biaisée, c'est sa supériorité (Annexe B). La supériorité de ZNCC est qu'il incorpore non seulement le bruit additif, mais aussi le bruit multiplicatif (qui n'est pas traité par ZSSD).

On cherche une mesure de ressemblance λ invariante par une transformation affine, non biaisée et symétrique:

1. invariante par une transformation affine (satisfaite par ZNCC)

$$\lambda(a, b) = \lambda(k_1 a + k_2, b)$$

où $(k_1 a + k_2)(i) = k_1 a(i) + k_2$ et k_1, k_2 sont deux constantes.

2. non biaisée (satisfaite par ZSSD)

$$E\lambda(a, b) = \phi(A, B)$$

où a_i (resp. b_j) sont des échantillons de A (resp. B), a_i et b_j sont indépendants deux à deux pour $i \neq j$.

3. symétrique $\lambda(a, b) = \lambda(b, a)$.

2.1.2 Invariant

Les méthodes d'invariant sont employées en vision depuis quelques années [MKM⁺92, GQ92]. Les invariants différentiels sont étudiés par Koenderink et ses confrères [KvD87, Flo93]. Récemment, Schmid [SM95] a proposé une méthode d'appariement en utilisant des invariants différentiels d'intensités lumineuses. Dans cette technique, on essaie d'appliquer la corrélation sur les vecteurs d'invariants des niveaux de gris pour l'appariement. Sa comparaison avec les techniques de corrélation est donnée dans section 2.1.4.

Les invariants différentiels des niveaux de gris allant jusqu'à l'ordre 3 sont utilisés ; en notant L la convolution entre une image et une fonction gaussienne, on calcule les dérivations de L , et les sauvegarde dans un vecteur $\vec{v}_i[0 \dots 8]$, comme suit :

$$\vec{v}_i[0 \dots 4] = \begin{bmatrix} L \\ L_x L_x + L_y L_y \\ L_{xx} L_x L_x + 2L_{xy} L_x L_y + 2L_{yy} L_y L_y \\ L_{xx} + L_{yy} \\ L_{xx} L_{xx} + 2L_{xy} L_{yx} + L_{yy} L_{yy} \end{bmatrix} = \begin{bmatrix} L \\ L_i L_i \\ L_i L_{ij} L_j \\ L_{ii} \\ L_{ij} L_{ji} \end{bmatrix}$$

Les indices x et y représentent les dérivées par rapport aux variables x et y , respectivement, par exemple : $L_{xy} = \frac{\partial^2}{\partial x \partial y} L$. On emploie aussi la notation d'Einstein, où une indice i (ou j) signifie la somme des dérivées par rapport à l'ensemble des variables, comme $L_i = \sum_i L_i = L_x + L_y$, $L_{ij} = L_{xx} + L_{xy} + L_{yx} + L_{yy}$, $L_{ii} = L_{xx} + L_{yy}$, etc.

Si on va jusqu'aux dérivations d'ordre 3, on obtient :

$$\vec{v}_i[5 \dots 8] = \begin{bmatrix} \epsilon_{ij}(L_{jkl}L_iL_kL_l - L_{jkk}L_iL_lL_l) \\ L_{iij}L_jL_kL_k - L_{ijk}L_iL_jL_k \\ -\epsilon_{ij}L_{jkl}L_iL_kL_l \\ L_{ijk}L_iL_jL_k \end{bmatrix}$$

où ϵ représente le tenseur canonique anti-symétrique $\epsilon_{21} = \epsilon_{21} = 1, \epsilon_{11} = \epsilon_{22} = 0$.

Le vecteur $\vec{v}$ présente ci-dessus est invariant pour un déplacement rigide. Pour qu'il s'adapte au changement d'échelle géométrique, c'est-à-dire, une transformation de similitude, le vecteur d'invariants doit être calculé à plusieurs échelles [Sch96].

Distance de Mahalanobis La distance utilisée pour comparer deux vecteurs d'invariants est celle de Mahalanobis. Cette distance statistique utilise la matrice de covariance Λ des composants et prend ainsi en compte la différence de magnitude des composants et leur corrélation éventuelle :

$$d_M(\vec{b}, \vec{a}) = \sqrt{(\vec{b} - \vec{a})^T \Lambda^{-1} (\vec{b} - \vec{a})}.$$

En seuillant cette distance, il est possible de décider si deux vecteurs sont similaires. Le carré de cette distance est une variable aléatoire de distribution χ . Il est donc possible d'utiliser une table de cette distribution pour seuiller la distance et rejeter les appariements qui ont la plus grande probabilité d'être faux.

Pour diminuer la probabilité de faux appariements, on peut filtrer les mises en correspondance par l'addition des contraintes basées sur le voisinage. Les contraintes semi-locales sont donc utilisées [Sch96].

Les expériences ont montré que ces contraintes servent à beaucoup diminuer le nombre de faux appariements.

En résumé, cette méthode a pour avantage la robustesse contre le changement d'échelle et la rotation géométriques des fenêtres. Mais les invariants utilisés dans cette méthode sont seulement pour le groupe de similitudes, et ils ne sont pas utilisables dans le cas où une transformation affine des fenêtres entre deux images

est marquée ; cette méthode n'atteint qu'une précision pixelique ; elle ne s'emploie que pour des points d'intérêts et n'est pas robuste aux occultations partielles.

2.1.3 Distance de Hausdorff [HR93, HJ95]

L'idée est de d'abord prétraiter les images d'intensités lumineuses pour avoir deux images binaires. On peut avoir deux images binaires denses (signes du Laplacien du gaussien [Nis84]) ou deux images binaires éparses (contours de Canny [Can86]). On compare les deux ensembles de points en utilisant la mesure de Hausdorff généralisée [HJ95].

La mesure de Hausdorff classique est définie pour deux ensembles de points $A = \{a_1, \dots, a_p\}$ et $B = \{b_1, \dots, b_q\}$ comme suit :

Soit

$$h(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\|$$

et $\|X\|$ est une norme de X pour les ensembles A et B (par exemple, la norme L_2 euclidienne).

La distance H est le maximum de $h(A, B)$ et $h(B, A)$. Elle mesure le degré de dissimilarité entre deux ensembles, en mesurant la distance du point de A qui est le plus loin de chaque point de B et vice versa. Concrètement, si la distance de Hausdorff est d , alors chaque point de A se trouve forcément à l'intérieur d'une distance d d'un point de B et vice versa.

Cette distance est naturellement étendue au problème de trouver la meilleure distance *partielle* entre deux ensembles :

$$h_k(B, A) = k^{\text{ième}} \min_{a \in A} \|a - b\|$$

où $k^{\text{ième}}$ dénote la $k^{\text{ième}}$ valeur ordonnée dans l'ensemble des distances, c'est-à-dire, pour chaque point de B , la distance au point le plus proche de A est calculée, et puis, les points de B sont ordonnés par leurs valeurs de cette distance respectivement. La $k^{\text{ième}}$ valeur ordonnée d signifie que k des points du modèle B sont à l'intérieur d'une distance d des points d'image A .

La distance de Hausdorff généralisée bidirectionnelle est donc définie comme :

$$H_{lk}(A, B) = \max(h_l(A, B), h_k(B, A))$$

De plus, on peut définir les distances de Hausdorff (classique et généralisée) sur un groupe G :

$$\begin{aligned} M_G(A, B) &= \min_{g_1, g_2 \in G} H(g_1 A, g_2 B) && \text{pour la distance classique} \\ M'_{G_{lk}}(A, B) &= \min_{g_1, g_2 \in G} H_{lk}(g_1 A, g_2 B) && \text{pour la distance généralisée} \end{aligned}$$

Si :

$$\|g x_1 - g x_2\| = \|x_1 - x_2\| \quad \forall g \in G, \forall x_1, x_2 \quad (2.1)$$

alors $M_G(A, B) = \min_{g \in G} H(A, gB)$ et $M'_{G_{lk}}(A, B) = \min_{g \in G} H_{lk}(A, gB)$.

La propriété (2.1) est vérifiée si G est le groupe de translation ou G est le groupe euclidien (rotations et translations) et $\|\cdot\|$ est la norme euclidienne.

Des techniques pour calculer les distances de Hausdorff sous le groupe euclidien sont présentées dans [HR93].

La distance de Hausdorff généralisée a pour avantage sa robustesse aux occultations partielles. Elle est néanmoins limitée dans le cas de la déformation euclidienne des fenêtres. Bien que Rucklidge [Ruc95] ait développé une technique pour le cas affine, cela reste encore un problème ouvert d'en faire une technique efficace. Un autre problème ouvert est de développer des modèles pour comparer les fonctions qui pourront être utilisés pour des mesures similaires opérant directement sur les images de niveaux de gris.

2.1.4 Les relations entre ces trois techniques

Nous venons de discuter les trois techniques pour l'appariement : la corrélation, les invariants locaux et la distance de Hausdorff. Ce sont des techniques d'appariement comparant la ressemblance de deux signaux.

La comparaison pour la technique d'invariants est réalisé à l'aide de la distance de Mahalanobis ; cette distance peut être considérée comme la généralisation de la mesure de corrélation SSD, au point de vue que SSD est en effet la distance de Mahalanobis avec la matrice de covariance d'identité. Donc, dans un sens plus général, la technique d'invariants est aussi une technique de corrélation des vecteurs d'invariants locaux. On peut présenter les invariants comme une expression invariante des caractéristiques d'une décomposition locale d'une base du signal ; la technique d'invariants locaux est discutée séparément car elle est invariante à un certain niveau de distorsion perspective, à une similitude entre

les deux images, par exemple. En revanche, la technique de corrélation requiert une distorsion perspective négligeable.

La corrélation et la technique d'invariants prennent d'abord des vecteurs caractéristiques, puis mesurent la ressemblance entre eux. Pour la technique d'invariants, ces vecteurs sont des invariants différentiels ; et pour la corrélation, ce sont des niveaux de gris, des gradients de niveaux de gris, des Laplaciens, de rang (corrélation de Bhat et corrélation rang de Zabih), ou bien des bits représentant le résultat de la comparaison des niveaux de gris entre un pixel et ses voisins (census de Zabih). La dimension des vecteurs caractéristiques pour la corrélation est plus grande que celle de la technique d'invariants, ce qui donne à la technique de corrélation une discriminance plus grande ; la technique d'invariants est moins discriminante et elle s'emploie seulement pour les points d'intérêts, mais elle est très robuste aux distorsions géométriques.

La méthode de distance de Hausdorff se réalise en comparant deux images binaires, autrement dit, deux ensembles de points. Cette méthode compare directement les formes représentées par des images binaires et elle est plus significative de ce point de vue. Sa version généralisée peut traiter l'occultation partielle, qui pose souvent des problèmes pour l'appariement visuel. Comme une technique alternative que les deux autres, elle est néanmoins plus coûteuse et difficile à s'employer pour traiter les distorsions perspectives, ce qui peut être considéré comme un inconvénient.

Un dernier avantage des techniques de corrélation de niveaux de gris est qu'on ne perd aucune information ; il semble donc que seulement à l'aide des corrélations de niveaux de gris on peut appairer les pixel en haute précision. Une telle technique est présentée dans le chapitre 4.

Notons que ces trois techniques d'appariement sont applicables seulement pour les points dans une zone assez texturée. Dans le cas contraire, des méthodes globales sont indispensables (page 41). Dans la section 2.2, nous présentons une des méthodes globales : la technique de programmation dynamique.

2.2 Méthodes de programmation dynamique

En général, les méthodes discrètes ne prennent pas en compte la régularité du champs de disparités, et les méthodes continues ne considèrent guère les variations lumineuses entre les deux signaux. Geiger [GLY92] a développé un algorithme qui considère à la fois la régularité du champ de disparités et les variations du signal. Cet algorithme est une méthode globale qui peut appairer les pixels dans une

zone peu texturée. Il se sert des deux contraintes suivantes :

- la contrainte d'occultation :

Une discontinuité dans une image correspond à une occultation dans l'autre image, et vice versa.

- la contrainte monotonique :

C'est une extension de la contrainte d'ordre en incorporant explicitement l'occultation, la discontinuité et l'unicité.

Enfin, la technique de programmation dynamique est utilisée pour implémenter cette technique. Nous allons illustrer cet algorithme en détail maintenant.

2.2.1 Probabilité d'appariement

Pour chaque pixel g dans l'image de gauche, et chaque pixel d dans celle de droite, on peut avoir un vecteur caractéristique W_g^G et W_d^D (par exemple : la liste d'intensités lumineuses dans une fenêtre autour), respectivement. Si les deux pixels se correspondent, pour une fonction de coût C , $C(W_g^G, W_d^D)$ doit être petite. On utilise un processus d'appariement M_{gd} , qui est 1, si g correspond à d , 0 sinon.

On définit la probabilité qu'une paire de W^G et W^D soient appariés, étant donné le processus d'appariement M par :

$$P_{input}(W^L, W^R | M) = \frac{e^{-\sum_{gd} M_{gd} C(W_g^G, W_d^D) + \epsilon(1 - M_{gd})}}{Z}$$

où le deuxième terme représente le coût de points non appariés ($M_{gd} = 0$), avec ϵ un paramètre positif à estimer et Z une constante de normalisation.

2.2.2 L'unicité et le processus d'occultation

Pour interdire l'appariement multiple, on impose que :

$$\sum_{g=0}^{N-1} M_{g,d} = 0, 1 \text{ et } \sum_{d=0}^{N-1} M_{g,d} = 0, 1.$$

Ces contraintes garantissent qu'il y a au plus un appariement pour chaque pixel, et elles autorisent l'existence de non appariement.

On considère de plus un processus d'occultation O , pour l'image de gauche O^G et un pour celle de droite O^D , tels que :

$$O_g^G = 1 - \sum_{d=0}^{N-1} M_{g,d} \text{ et } O_d^D(M) = 1 - \sum_{g=0}^{N-1} M_{g,d}.$$

Ce processus est 0 dans le cas d'appariement, 1 dans le cas contraire. La disparité est définie pour l'image à gauche D^G et celle à droite D^D comme :

$$D_g^G(M)(1 - O_g^G) = \sum_{d=0}^{N-1} M_{g,d}(d - g) \text{ et } D_d^D(M)(1 - O_d^D) = \sum_{g=0}^{N-1} M_{g,d}(d - g).$$

Elles sont définies seulement dans le cas d'appariement.

2.2.3 Fonction de coût

Comme le changement de surface est normalement petit, sauf aux discontinuités de profondeur, on impose que le champ de la disparité doit être une fonction lissée avec des discontinuités [BZ87]. Une telle fonction de coût est donnée dans [GG91] :

$$U(M) = 2\gamma - \sum_l \ln(1 + e^{\gamma - \mu(D_{l+1}^L - D_l^L)^2}) - \sum_R \ln(1 + e^{\gamma - \mu(D_{r+1}^R - D_r^R)^2})$$

où μ et γ sont des paramètres à estimer.

En l'incorporant avec la probabilité $P_{input}(W^L, W^R|M)$, on obtient :

$$P_{stereo}(M|W^G, W^D) = \frac{e^{-V_{eff}(M)}}{Z}$$

où Z est une constante de normalisation et

$$V_{eff}(M) = \sum_{gd} M_{gd}(C(W_g^G, W_d^D) - \frac{1}{N} \ln(1 + e^{\gamma - \mu(D_{g+1}^G - D_g^G)}) - \frac{1}{N} \ln(1 + e^{\gamma - \mu(D_{d+1}^D - D_d^D)})) + \frac{1}{N} \frac{\epsilon}{2} (O_g^G + O_d^D). \quad (2.2)$$

2.2.4 La contrainte d'occultation

Pour qu'un modèle de stéréo autorise les discontinuités des disparités, il doit admettre aussi les occultations, et vice versa. En effet, la plupart des discontinuités de disparité le long de la ligne épipolaire dans une image correspondent à une région d'occultation dans une autre image. Un bon modèle de stéréo doit être symétrique aux occultations et discontinuités dans l'image de gauche et de droite.

L'occultation peut être mieux comprise dans l'espace d'appariement. On suppose que la ligne épipolaire à gauche est l'abscisse horizontal de l'espace d'appariement, celle à droite est l'abscisse vertical de cet espace. Un chemin peut être coupé verticalement quand une discontinuité est détectée dans l'image de gauche, et peut être coupé horizontalement quand une occultation est trouvée. Souvent un saut vertical dans une image correspond à un saut horizontal dans une autre.

Considérons la figure 2.3, dans cette figure, la courbe en trait plein est le graphe d'appariement le long de deux lignes épipolaires. Les zones [a b] et [a' b'], [b d] et [c' d'] sont mises en correspondance. Le point e est occulté dans l'image droite, ce qui correspond à un saut horizontal; il y a une discontinuité au point b, ce qui correspond à un saut vertical.

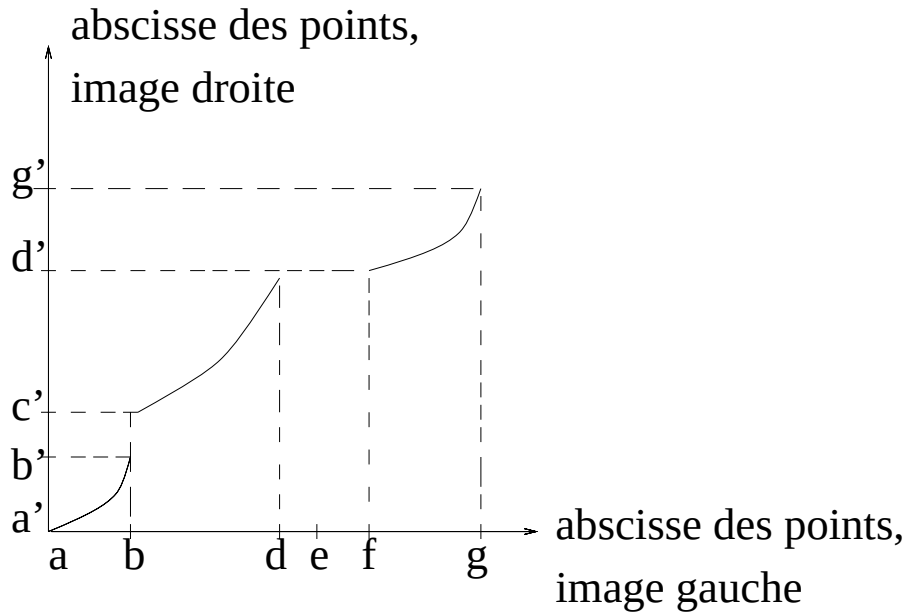
La contrainte d'occultation impose qu'une discontinuité dans une image corresponde à une occultation dans l'autre image, et vice versa.

2.2.5 La contrainte de monotonicité

Par extension de la contrainte d'ordre, on peut aussi imposer la monotonicité de la fonction $F_g^G = g + D_g^G$ pour l'image de gauche ou la monotonicité de $F_d^D = d + D_d^D$. En considérant la contrainte d'occultation et l'unicité, on peut supposer la monotonicité de F_g^G pour une occultation de taille o :

$$F_{g+o+1}^G > F_g^G \quad \forall g$$

où $O_{g+o+1}^G = O_g^G = 0$ et $\sum_{g'=g+1}^{g+o} (g - O_{g'}^g) = 0$, et similaire pour F_d^D . Cette contrainte peut être appliquée pour simplifier l'optimisation de la fonction de coût dans l'équation (2.2).

FIG. 2.3 – *L'espace d'appariement.*

2.2.6 Programmation dynamique

Comme les interactions du champ de disparité D_g^G et D_d^D sont restreints à un petit voisinage, on peut employer la programmation dynamique pour résoudre le problème.

La programmation dynamique fonctionne en résolvant des sous-problèmes de la forme : quel est le chemin avec le coût le plus bas du début à (g, d) et quel est son coût ? Ces sous-problèmes sont résolus colonne par colonne de gauche à droite et finalement la solution du problème est trouvée. À chaque étape, la résolution du sous-problème dépend de l'ensemble de sous problèmes précédemment résolus. Grâce à la contrainte de monotonie, l'ensemble des sous problèmes précédemment résolus est réduit. Plus précisément, la résolution du sous problème (g, d) demande des informations des solutions des sous problèmes précédents (x, y) où $x < g$ et $y < d$. Notons que la contrainte de monotonie est utilisée pour réduire l'ensemble de sous problèmes précédemment résolus, et rendre l'algorithme plus efficace.

2.2.7 L'utilisation des points de contrôle

Intille [IB94] a proposé un algorithme de stéréo qui intègre l'appariement et l'analyse d'occultation dans un seul processus, en introduisant la notion des *points de contrôle* (GCP : Ground Control Points en anglais). Ces points sont des appariements très fiables. La sensibilité du coût de non appariement et la complexité en temps sont significativement réduites, quand ces points sont incorporés dans le processus de mise en correspondance.

Plusieurs critères heuristiques sont employés pour identifier les GCP. Le premier demande qu'un point de contrôle soit le meilleur de gauche à droite et aussi le meilleur de droite à gauche. Le deuxième exige les points de contrôle d'avoir des valeurs d'appariement plus petites que le coût d'occultation. Le troisième demande que ces GCP n'aient pas des voisins immédiats qui sont aussi des GCPs.

Une fois qu'on a choisi l'ensemble de points de contrôle, on force l'algorithme de programmation dynamique à choisir un chemin passant par ces points, en donnant un coût zéro aux appariements avec un point de contrôle et un très grand coût aux autres chemins.

Les expérimentations montrent que ces points de contrôle améliorent le résultat, réduisent la complexité, et la dépendance du coût d'occultation.

2.3 Résumé

Dans ce chapitre, nous avons présenté de différentes méthodes d'appariement de deux images. Dans la section 2.1, nous avons présenté des méthodes discrètes d'appariement des pixels, et dans la section 2.2, nous avons présenté une méthode globale pour l'appariement dense de deux images. Normalement les techniques discrètes d'appariement des pixels sont applicables seulement pour les zones texturées, et pour les zones non texturées, le problème d'appariement peut être résolu seulement à l'aide des techniques globales, comme la programmation dynamique. Dans le processus d'appariement dense, un problème très gênant est celui des occultations partielles [BLM96]. Pour cela, une solution sera apportée dans le chapitre 3. La programmation dynamique ne donne que des appariements à précision pixelique. Basés sur ces appariements initiaux, des appariements précis peuvent être obtenus à l'aide de techniques présentées dans le chapitre 4.

Chapitre 3

Corrélation partielle

La technique de comparer la ressemblance¹ entre deux fenêtres s'appelle la *corrélation*, par extension de la corrélation entre deux fonctions. Cette technique peut être utilisée pour l'appariement de pixels dans deux images. Le terme «corrélation» a aussi une autre origine en probabilité et en statistique où l'on trouve *coefficient de corrélation* et *coefficient de corrélation de Pearson* respectivement [Fel86].

Beaucoup de techniques de corrélation ont des difficultés près des discontinuités, car une partie de la fenêtre d'image considérée subit de larges perturbations. Une partie de pixels correspondent au modèle à considérer et d'autres viennent d'autres parties de la scène. Cela entraîne un problème pour beaucoup de techniques de corrélation, car les corrélations standards sont basées sur les méthodes statistiques standards, qui conviennent mieux à une seule population. Nous appelons ce phénomène l'*occultation partielle*, et nous proposons la *corrélation partielle* pour le surmonter.

Nous présentons deux méthodes de la *corrélation partielle* dans ce chapitre, elles sont basées sur la *régression robuste* (section 3.1) et la *localisation robuste* (section 3.2) respectivement. Pour chaque méthode, nous montrons des expérimentations sur des images réelles d'une scène planaire qui valident l'approche.

Nous utilisons aussi d'autres mesures de corrélation pour la comparaison expérimentale, qui sont définies à la page 48 et dans le chapitre 2, tableau 2.1.

1. Des mesures donnent la *différence* de deux signaux, au sens que plus que le score est grand, moins que les deux signaux se ressemblent. Dans la suite, on utilise le mot “ressemblance” ou “corrélation” au sens d’une mesure de la ressemblance ou la différence de deux signaux.

3.1 Corrélation partielle basée sur la régression

Nous commençons par l'introduction des deux définitions en probabilité et en statistique (3.1.1) et puis nous introduisons notre principe de *corrélation partielle* (3.1.2).

3.1.1 Corrélation et régression en probabilité et en statistique

Dans cette section, nous commençons par le rappel de la définition du *coefficient de corrélation* en probabilité et en statistique, puis nous introduisons des autres mesures de ressemblance. Nous expliquons un peu leurs sens selon leurs noms. Nous terminons par une discussion de la relation entre la régression et la corrélation pour déduire la conclusion :

Une corrélation robuste doit se baser sur une régression robuste.

3.1.1.1 Corrélation

Rappel de corrélation en probabilité et en statistique

Définition de la corrélation 1 Soient X et Y deux variables réelles aléatoires carrément intégrables, et non constantes. Alors, $\rho(X, Y) = \frac{\sigma_{XY}}{\sqrt{\sigma_X^2 \sigma_Y^2}}$ est appelé le coefficient de corrélation de X et Y où σ_{XY} est la covariance entre X et Y , σ_X^2 , σ_Y^2 sont les variances de X et de Y .²

Cette notion se transpose en statistique :

Soient $X = (x_i)$ et $Y = (y_i)$ deux listes de n nombres réels, alors $\frac{\sigma_{XY}}{\sqrt{\sigma_X^2 \sigma_Y^2}}$ est appelé le coefficient de corrélation de X et Y où σ_{XY} est la covariance de X et de Y , σ_X^2 , σ_Y^2 sont les variances de X et Y .³

On peut montrer facilement que :

$|\rho(X, Y)| = 1 \iff$ Il existe une relation affine entre X et Y , c'est-à-dire $\exists c, \exists(a, b), ab \neq 0$ tels que $aX + bY = c$.

2. Nous notons également σ_X^2 et σ_Y^2 pour les variances de X et de Y , respectivement.

3. Formellement, $\sigma_{XY} = \frac{\sum_{i=1}^n (x_i - \bar{X})(y_i - \bar{Y})}{n}$, $\sigma_X^2 = \frac{\sum_{i=1}^n (x_i - \bar{X})^2}{n}$, $\sigma_Y^2 = \frac{\sum_{i=1}^n (y_i - \bar{Y})^2}{n}$ où $\bar{X}$ et $\bar{Y}$ sont les moyennes de X et de Y respectivement.

De plus, on a

- $\rho(X, Y) = 1 \iff ab < 0$ (X et Y sont *positivement* liées).
- $\rho(X, Y) = -1 \iff ab > 0$ (Elles sont *néativement* liées).

En résumé, ce coefficient mesure l'association linéaire entre deux variables aléatoires X et Y , ou entre deux réalisations (x_i) et (y_i) .

Dans les discussions suivantes, on se limite au cas statistique. Le cas en probabilité est similaire. On emploiera des termes moins stricts, par exemple, «une liste de nombres réels» plutôt que «une réalisation d'un échantillon».

Supposons que $y = ax + b + \epsilon$ et que l'on observe x_i, y_i , si l'on cherche à minimiser :

$$\epsilon^2 = \min_{a,b} \sum_{i=1}^n (y_i - ax_i - b)^2$$

la variance du bruit sous l'hypothèse que le bruit soit gaussien et indépendant, on trouve que $a = \frac{\sigma_{XY}}{\sigma_{X^2}}$, $b = \bar{Y} - \frac{\sigma_{XY}}{\sigma_{Y^2}}\bar{X}$, et $\epsilon^2 = \sigma_{Y^2} - \frac{\sigma_{XY}^2}{\sigma_{X^2}} = \sigma_{X^2}(1 - \rho(X, Y)^2)$,

donc

$$\frac{\epsilon^2}{\sigma_{Y^2}} = 1 - \rho(X, Y)^2 \quad (3.1)$$

où $\frac{\epsilon^2}{\sigma_{Y^2}}$ est le *rapport bruit par signal*.

Donc, minimiser le *rapport bruit par signal* est équivalent à maximiser le carré du coefficient de la corrélation. Le signe du coefficient donne la tendance de leur dépendance.

Il existe d'autres méthodes pour mesurer la ressemblance de deux listes de nombres réels. Elles sont présentées dans le tableau 2.1. On donne leurs définitions ici brièvement.

- ressemblances utilisant la norme L_2

. ZSSD

Soient $X = (x_i)$ et $Y = (y_i)$ deux listes de nombres réels, alors $\sqrt{\frac{\sum_{i=1}^n ((x_i - \bar{X}) - (y_i - \bar{Y}))^2}{n}}$ est appelée leur ZSSD(X, Y).

. SSD

Soient $X = (x_i)$ et $Y = (y_i)$ deux listes de nombres réels, alors $\sqrt{\frac{\sum_{i=1}^n (x_i - y_i)^2}{n}}$ est appelée leur $SSD(X, Y)$.

– ressemblances utilisant la norme L_1

. ZSAD

Soient $X = (x_i)$ et $Y = (y_i)$ deux listes de nombres réels, alors $\sqrt{\frac{\sum_{i=1}^n |(x_i - \bar{X}) - (y_i - \bar{Y})|}{n}}$ est appelée leur $ZSAD(X, Y)$.

. SAD

Soient $X = (x_i)$ et $Y = (y_i)$ deux listes de nombres réels, alors $\sqrt{\frac{\sum_{i=1}^n |x_i - y_i|}{n}}$ est appelée leur $SAD(X, Y)$.

En vision, ces méthodes sont appelées *fonction de corrélation*, et $\rho(X, Y)$ est appelée $ZNCC(X, Y)$. $ZNCC$ vient de l'anglais «*Zero mean Normalised Cross Correlation*», qui se traduit en français comme «*corrélation centrée normalisée*»; $ZSSD$ vient de «*Zero mean Sum of Squared Differences*», qui est «*somme des différences centrées aux carrées*» en français; SSD , venant de «*Sum of Squared Difference*», est la somme des différences aux carrées; similairement, $ZSAD$ et SAD , venant de *Zero mean Sum of Absolute Difference* et *Sum of Absolute Difference*, sont la somme des valeurs absolues des différences centrées et la somme des valeurs absolues des différences, respectivement.

Maintenant, nous discutons la relation entre la régression et ces fonctions de corrélation.

3.1.1.2 Régression

Donnons les relations entre la régression et les fonctions de corrélation différentes ci-dessous.

- On voit que $ZNCC(X, Y)$ donne le rapport bruit/signal et la tendance de dépendance de X et Y .
- $ZSSD = \min_C \sqrt{\sum_{i=1}^n (y_i - x_i - C)^2}$ donne l'estimation de *l'écart-type* de bruit gaussien sous l'hypothèse que $y_i = x_i + C + e_i$ où C est une constante et e_i sont des variables aléatoires indépendantes de même distribution gaussienne centrée.

- *SSD* donne l'estimation de *l'écart-type* de bruit gaussien sous l'hypothèse que $y_i = x_i + e_i$ et e_i sont des variables aléatoires indépendantes de même distribution gaussienne centrée.
- *ZSAD* donne l'estimation de *la norme L_1* du bruit sous l'hypothèse que $y_i = x_i + \bar{y} - \bar{x} + e_i$, e_i sont des variables aléatoires indépendantes intégrables de même distribution.
- *SAD* donne l'estimation de *la norme L_1* de bruit sous l'hypothèse que $y_i = x_i + e_i$, e_i sont des variables aléatoires indépendantes intégrables de distribution identique.

On voit donc qu'il y a un lien fort entre la régression et ces fonctions de corrélation. En résumé, ZNCC correspond à la régression *affine*, ZSSD et ZSAD correspondent à celle *translationnelle*, SSD et SAD correspondent à celle *d'égalité*. C'est pour cette raison que ZNCC est invariant aux transformations affines de niveaux de gris, ZSSD et ZSAD sont invariantes aux translations de niveaux de gris, et SSD et SAD sont sensibles aux translations et aux effets d'échelle de niveaux de gris.

En résumé, la *régression* donne l'estimation de relation entre deux ensembles de données, et la qualité de cette relation est exprimée en terme de la fonction de *corrélation*.

Malheureusement, la régression classique est fragile face aux erreurs aberrantes (chapitre 1, section 1.3). D'où : Une corrélation robuste doit se baser sur une régression robuste.

Nous expliquons le principe de la corrélation robuste dans la section 3.1.2, qui est basé sur la régression robuste. Ici, la robustesse signifie l'insensibilité aux corruptions, pour cela, il faut qu'on soit capable de trouver la partie corrompue, et puis accomplir la corrélation sur la partie saine. Cette corrélation robuste, est donc appelée la *corrélation partielle*.

3.1.2 Corrélation partielle basée sur la régression partielle

Nous commençons par la discussion de la relation entre la corrélation et la régression dans la section 3.1.2.1, puis nous en déduisons la procédure de la corrélation partielle dans la section 3.1.2.2, à la fin nous introduisons deux contraintes *la contrainte du point principale* et *la contrainte de la fenêtre principale* qui peuvent s'employer dans l'appariement de deux images dans la section 3.1.2.3.

3.1.2.1 Analyse de corrélation et régression

Supposons que nous avons deux listes de nombres réels (x_i) et (y_i) , nous cherchons une relation entre elles :

$$y_i = f_\alpha(x_i) + e_i \quad (3.2)$$

où α est l'ensemble de paramètres et f est une fonction donnée, qui dépend de α , les e_i (bruit) sont des variables aléatoires indépendantes entre elles, de distribution identique.

Donnons un exemple : dans une régression translationnelle, $y_i = x_i + \alpha + e_i$, les e_i peuvent être supposées gaussiennes centrées d'écart-type identique et indépendantes entre elles.

Le problème de régression est la recherche de α , et la mesure de corrélation peut être l'écart-type de e_i , ou sa norme L_1 , etc.⁴

3.1.2.2 Procédure de la corrélation partielle

Si des occultations sont présentées, la relation (3.2) n'est valable que sur la partie non occultée. On ne doit donc calculer la mesure de corrélation que sur cette partie non occultée.

La procédure de la *corrélation partielle* peut être décrite comme suit :

1. *Estimer α de façon robuste.*
2. *Obtenir des informations robustes à partir de α robuste.*
3. *Trouver la partie occultée selon un test statistique.*
4. *Calculer la mesure de corrélation sur la partie non occultée.*

Nous expliquons ces quatre étapes en détail :

1. Pour ZNCC, on a deux paramètres à estimer ; pour ZSSD et ZSAD, on en a un ; et pour SSD et SAD, on n'en a aucun. On verra qu'avec la contrainte du *point principal* (équation (3.4) à la page 72), on peut réduire le nombre de paramètres à estimer par un.

4. Si l'on cherche à minimiser la norme L_1 de e_i , on trouve que α doit être la médiane de $y_i - x_i$, qui est en général différent de $\bar{y} - \bar{x}$; si on cherche à minimiser sa norme L_2 , on trouve bien que $\alpha = \bar{y} - \bar{x}$.

2. Les informations robustes sont celles non sensibles aux erreurs aberrantes, par exemple la médiane des erreurs aux carrées ; au contraire, la moyenne, ou la moyenne des erreurs aux carrées ne le sont pas, et sont sensibles aux erreurs grossières.
3. Après l'étape précédente, on peut estimer la fonction de distribution et décider si une erreur appartient à cette distribution selon un intervalle de confiance.

. Exemple 1.

Si l'on suppose que le bruit est gaussien centré, et $m = \sqrt{\text{med}_i e_i^2}$ où e_i sont les erreurs, alors $\sigma = 1,4826m$ [RL87] donne l'estimation de l'écart type du bruit, et si $|e_i| > 2,5\sigma$ [RL87], elle est déclarée comme une erreur grossière.

. Exemple 2.

Si l'on suppose que le bruit a une fonction de densité $\frac{e^{-\frac{|x|}{\sigma}}}{2\sigma}$, $m = \text{med}_i |e_i|$, où e_i sont les erreurs, alors σ est $\frac{m}{\ln 2}$.⁵

Pour chaque e_i , la confiance qu'elle soit une erreur grossière est

$$\frac{1}{2\sigma} \int_{-|e_i|}^{|e_i|} e^{-\frac{|x|}{\sigma}} dx = 1 - e^{-\frac{|e_i|}{\sigma}}.$$

4. La corrélation partielle est réalisée par la corrélation pondérée. Par exemple, si l'on laisse $\omega_i = 1$ (bon point) ou $\omega_i = 0$ (mauvais point), la corrélation partielle de $ZSSD$ peut s'exprimer sous la forme de la corrélation pondérée comme suit :

$$ZSSD_p(X, Y) = \sqrt{\frac{\sum_{i=1}^n ((x_i - \bar{x}) - (y_i - \bar{y}))^2 \omega_i}{\sum_{i=1}^n \omega_i}}. \quad (3.3)$$

où $\bar{x}$ (respectivement $\bar{y}$) sont les moyennes pondérées de x_i (respectivement y_i), de pondérations ω_i , c'est-à-dire : $\bar{x} = \frac{\sum_{i=1}^n x_i \omega_i}{\sum_{i=1}^n \omega_i}$ et $\bar{y} = \frac{\sum_{i=1}^n y_i \omega_i}{\sum_{i=1}^n \omega_i}$.

3.1.2.3 Appariements des points dans deux images par corrélation partielle

Nous discutons maintenant le problème d'appariement des points dans deux images, par la *corrélation partielle* de deux fenêtres comme cela a été présenté

5. $\frac{1}{2\sigma} \int_{-m}^m e^{-\frac{|x|}{\sigma}} dx = \frac{1}{2}$ donne $m = \sigma \ln 2$.

dans la section 3.1.2.2, et nous allons présenter la contrainte du *point principal* et la contrainte de la *fenêtre principale*.

Dans le cas d'appariement de pixels de deux images, nous supposons que :

Hypothèse 3.1 *Le point à apparier n'est pas occulté.*

On l'appelle le *point principal*.

Hypothèse 3.2 *Plus qu'un point est proche du point principal, moins il est probable qu'il soit occulté.*

Pour obtenir des informations de la distribution des erreurs, au lieu de considérer la fenêtre totale, nous pouvons prendre une petite fenêtre (de forme rectangulaire normalement) autour du *point principal*. Elle est appelée la *fenêtre principale*. .

Contrainte du point principal Selon la première hypothèse 3.1, on peut imposer une contrainte dans l'équation (3.2) ci-dessous :

$$y_{pp} = f_{\alpha}(x_{pp}) \quad (3.4)$$

où x_{pp} et y_{pp} sont les intensités lumineuses des points principaux. Avec cette contrainte, on réduit un nombre de paramètres à estimer.

Cette contrainte a deux avantages :

- Elle incorpore le fait que le *point principal* n'est pas occulté, et elle rend la détection d'occultation partielle plus fiable.
- Elle diminue le nombre de paramètres à estimer par un et accélère la vitesse de la détection de parties occultées.

Pour ZSSD et ZSAD, cette accélération est significative : on a besoin de faire plusieurs tirages aléatoires pour estimer un paramètre sans cette contrainte ; et avec cette contrainte, on obtient immédiatement la régression robuste.

L'inconvénient de cette contrainte est qu'elle n'est pas assez souple : car en effet, on a $y_{pp} = f_{\alpha}(x_{pp}) + e_{pp}$, où e_{pp} n'est pas une erreur grossière, mais elle n'est pas forcément nulle. En résumé, avec cette contrainte, la détection dépend plus du *point principal*.

Contrainte de la fenêtre principale Selon la deuxième hypothèse 3.2, nous pouvons estimer la médiane des erreurs dans la fenêtre principale. Nous l'appelons *contrainte de la fenêtre principale*.

L'avantage de procéder ainsi est l'accélération de vitesse car le calcul de la médiane pour la *fenêtre principale* est plus rapide que celui pour la fenêtre totale, et son inconvénient est sa forte dépendance de la *fenêtre principale*. Ce procédé ne sera donc pas retenu.

3.1.3 Tests simples et résultats statistiques sur des images planaires¹

3.1.3.1 Corrélacion partielle pour l'appariement d'images

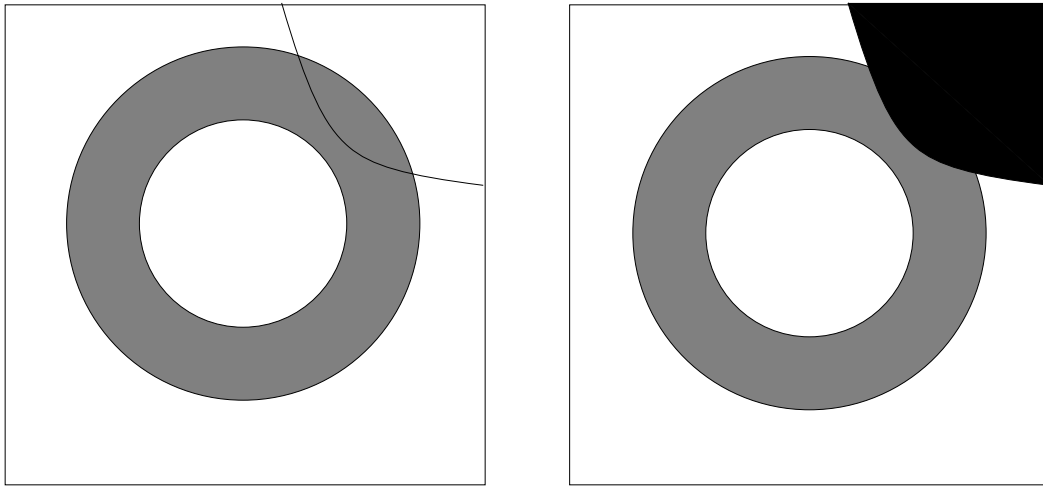


FIG. 3.1 – *Occultation. Gauche: I_1 , droite: I_2 .*

Considérons les deux fenêtres dans la figure 3.1, elles s'apparient sauf pour la partie en haut à droite dans les images I_1 et I_2 . Nous devons retrouver cette partie contaminée. Nous supposons que localement, le signal obéit une relation affine d'une image à une autre :

$$I_2(s, t) = kI_1(s + dx, t + dy) + l + \epsilon$$

où (dx, dy) est la disparité du point (x, y) , $s = x + u$, $t = y + v$, et $s \in \{x - N \dots x + N\}$, $t \in \{y - M \dots y + M\}$. $2N + 1$ et $2M + 1$ sont la largeur et l'hauteur

1. Le terme «images planaires» employé dans ce mémoire représente les images d'une scène planaire.

de la fenêtre, respectivement. ϵ est un bruit gaussien d'image. Cette relation affine est vraie pour la fenêtre sauf pour la partie occultée (comme la partie en haut à droite indiquée dans la figure 3.1).

Un modèle simplifié est celui translationel, c'est-à-dire, l'échelle k est supposé d'être 1 :

$$I_2(s, t) = I_1(s + dx, t + dy) + l + \epsilon \quad (3.5)$$

Si nous employons la *contrainte du point principal*, nous pouvons avoir :

$$I_2(x, y) = kI_1(x + dx, y + dy) + l \text{ (cas affine) ,} \quad (3.6)$$

$$I_2(x, y) = I_1(x + dx, y + dy) + l \text{ (cas translationel).} \quad (3.7)$$

La partie occultée peut être trouvée par la *statistique robuste*, comme montré dans la figure 3.2.

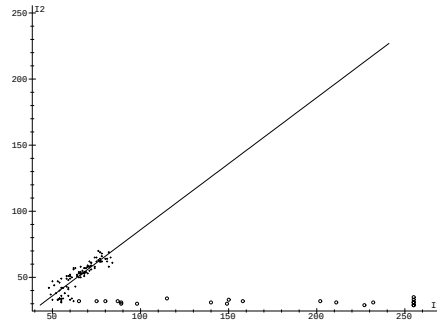


FIG. 3.2 – *Régression robuste entre I_1 and I_2 , les petits cercles sont des couples de points non appariés.*

Cette figure affiche les couples de niveaux de gris des pixels appariés dans deux images. Chaque couple est représenté par un point, et les points représentant un couple de pixels appariés doivent se situer autour d'une droite. Les points représentant un couple de pixels non appariés sont éloignés de cette droite.

Après avoir trouvé la partie occultée, la corrélation est limitée sur la partie restante de deux fenêtres. C'est la *corrélation partielle*, qui est un cas spécial de la *corrélation pondérée*. Comme nous laissons $\omega(u, v) = 1$ («bon» point) ou $\omega(u, v) = 0$ («mauvais» point).

Par exemple, ZSSD pondéré peut être exprimé comme suit :

$$ZSSD_p(X, dX) = \sqrt{\frac{\sum_{\Delta \in W} (((I_1(X + \Delta) - \bar{I}_1(X)) - (I_2(X + \Delta) - \bar{I}_2(X)))^2 \omega(\Delta))}{\sum_{\Delta \in W} \omega(\Delta)}} \quad (3.8)$$

3.1.3.2 Résultats expérimentaux

Dans cette section, nous montrons d'abord des exemples sur des points isolés et puis donnons des comparaisons avec les méthodes standard ZSSD et non standards : rang et census.

Exemples Nous testons notre approche d'abord sur des points d'images où les méthodes classiques comme ZNCC (Zero mean Normalised Cross Correlation), ZSSD (Zero mean Sum of Squared Difference) échouent [RBM94]. Dans la figure 3.3, deux exemples sont montrés. Les deux rectangles dans l'image droite représentent les régions d'intérêt choisies. Les techniques de corrélations classiques ne fonctionnent pas dans ces cas. Dans le premier cas, une grande occultation apparaît et dans le deuxième cas, la surface est presque tangente à la direction de vue et les deux fenêtres ne se correspondent pas correctement.

Les lignes épipolaires sont affichées dans la figure droite. La corrélation partielle réussit dans ces cas. Par exemple, sur le doigt de Paolo, la méthode de *corrélation partielle* trouve l'appariement correct à droite, alors que ZSSD trouve celui faux à gauche.

Dans la figure 3.4, les zones non appariées sont affichées pour chaque modèle, quand l'appariement correct est trouvé. Dans la figure 3.4 (a) est affichée une fenêtre où une partie à gauche est blanche, cela signifie que les deux fenêtres du point de doigt se correspondent SAUF pour une partie à gauche. Car dans l'image (a) de la figure 3.3, cette partie est l'image d'une cube blanche ; et dans l'image (b) de la figure 3.3, cette partie est l'image d'un fond noir. Le cas est similaire pour la figure (b).

À travers de ces deux exemples, nous voyons que notre méthode fonctionne où celle classique échoue. Nous allons montrer des résultats statistiques.

Résultats statistiques Nous avons fait des expérimentations sur des paires d'images d'une scène planaire. Dans un tel cas, il y a une homographie entre ces deux images d'une même scène planaire, car il y a une homographie entre chaque image et la scène respectivement. Cette homographie peut être estimée à partir des cibles appariées dans les deux images. Elle donne le correspondant de chaque

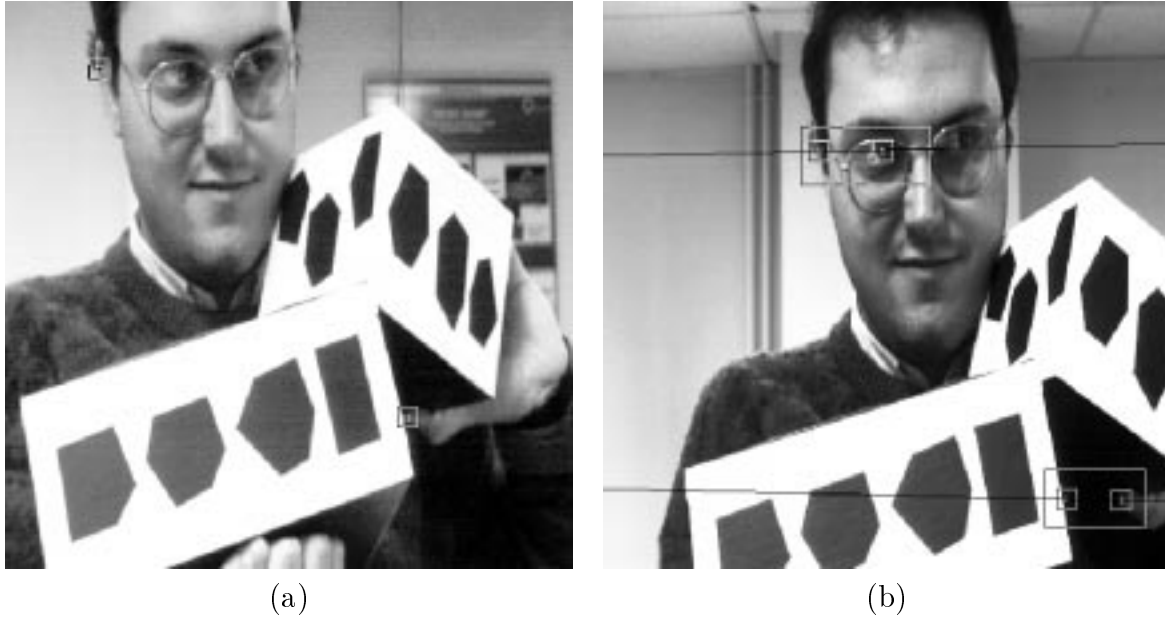


FIG. 3.3 – *Paolo. Deux tests d'appariements: l'un sur le doigt et l'autre sur l'oreille. Les petits carrés dans l'image à gauche sont des modèles.*

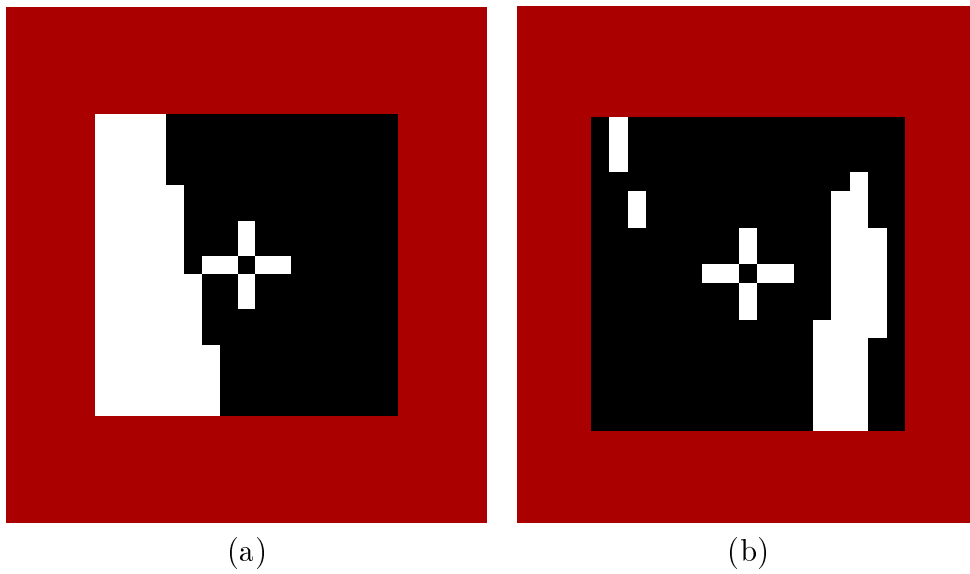


FIG. 3.4 – *En blanc sont les parties non appariées pour les deux exemples de la figure 3.3, l'image (a) correspond au doigt où retrouve le bord du cube occultant, l'image (b) correspond à l'oreille.*

point d'une image dans une autre image et donc elle nous permet de valider les résultats d'un appariement. Donc, on peut calculer l'appariement correct en estimant précisément l'homographie entre les deux images.

Dans ce mémoire, les images d'une scène planaire sont utilisées pour la validation d'une méthode d'appariement, nous les appelons aussi «images planaires».

Nous avons testé la méthode en utilisant la relation translationnelle (équation (3.5)) et la *contrainte du point principal* (équation (3.7)). Nous l'appelons *rzssdp* ici, qui signifie *robuste ZSSD avec la contrainte du point principal*. Cette méthode est comparée avec la méthode standard ZSSD, la méthode *rang* et la méthode *census*.

Deux paires d'images sont utilisées pour ce test. Elles sont affichées dans la figure 3.5 et la figure 3.6.

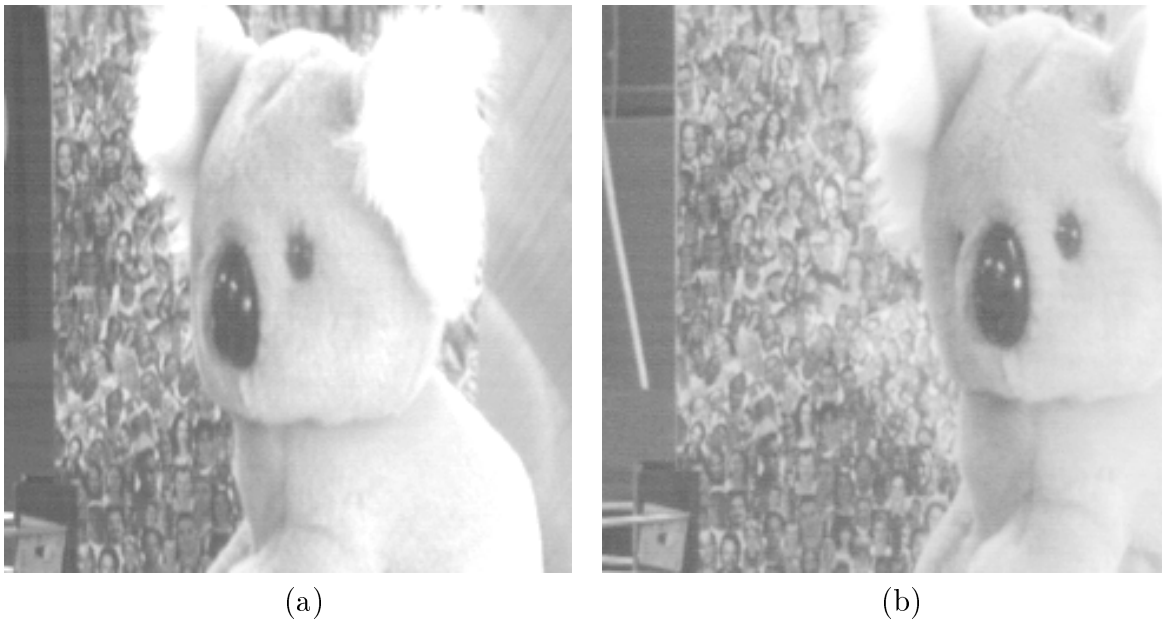


FIG. 3.5 – *La paire stéréo 1: benetton.*

Les images sont de taille 512×512 pixels et elles ont un fond planaire avec des textures différentes. Quatre tests sont faits, la figure 3.7 montre deux ensembles de points pour le fond *benetton*, et la figure 3.8 montre deux ensembles de points pour le fond *world*; pour le troisième test, les points sont choisis le long des contours d'occultations : un cas difficile à traiter.

Nous avons testé des méthodes différentes sur ces quatre ensembles de points. Les candidats sont contraints de rester sur la ligne épipolaire et d'avoir la disparité

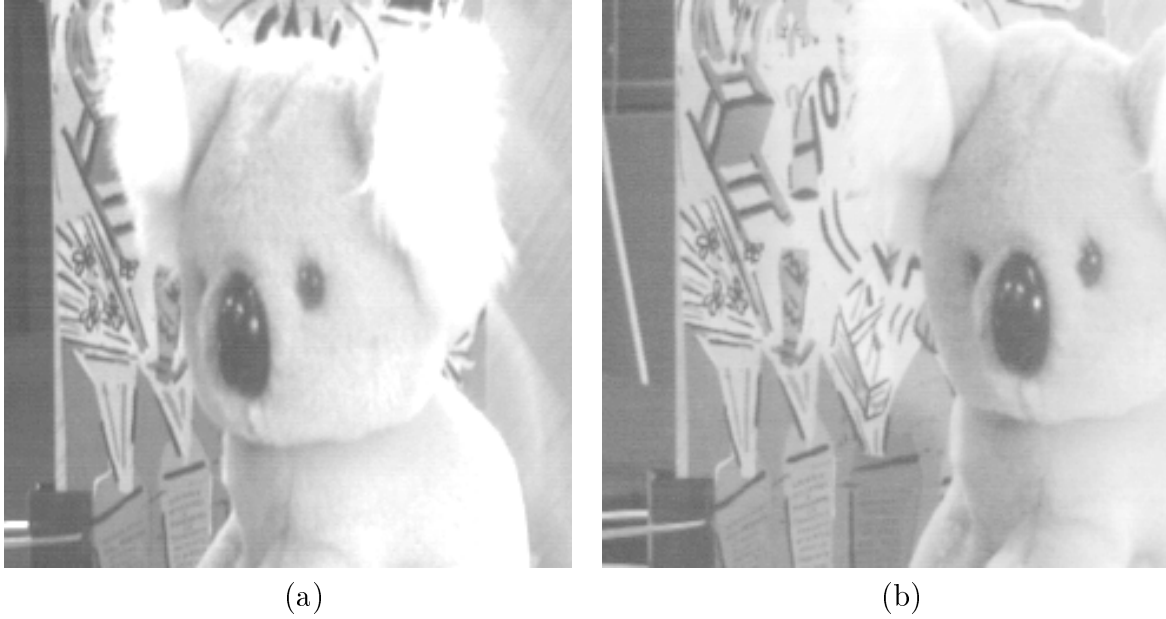
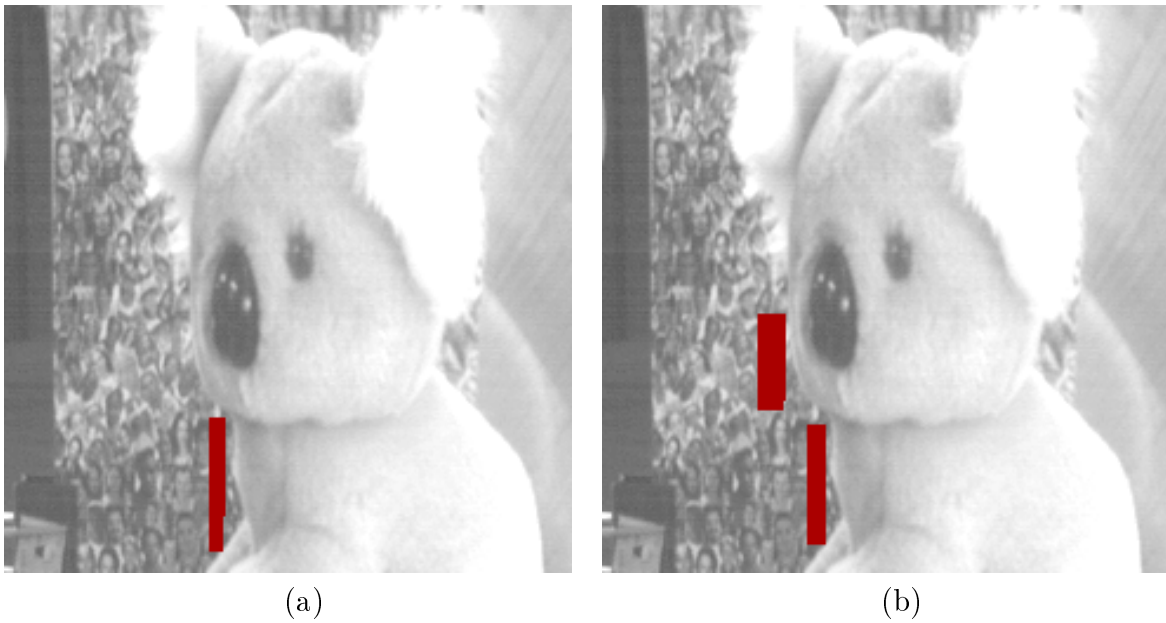
FIG. 3.6 – *La paire stéréo 2: world.*

FIG. 3.7 – *Fond benetton. Points choisis (noirs) dans la Figure 3.5 (a) pour test 1 et 2. Gauche: le test 1 correspondant au tableau 3.1; Droite: le test 2 correspondant au tableau 3.2 .*

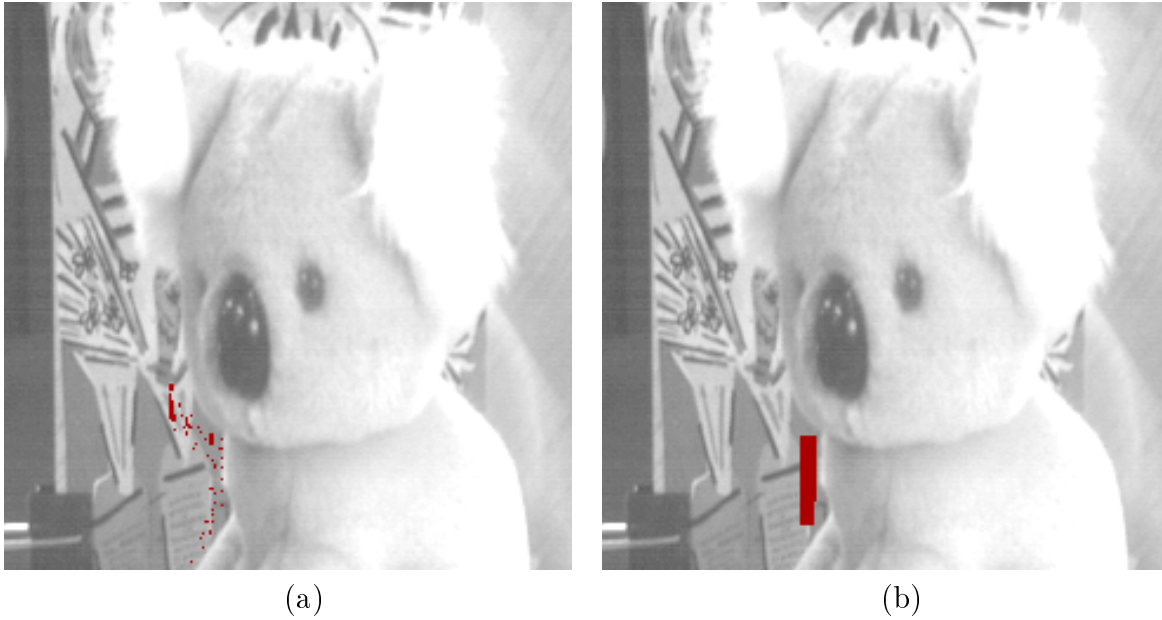


FIG. 3.8 – *Fond world*. Points choisis (noirs) dans la Figure 3.6 (a) pour test 3 et 4. Gauche : le test 3 correspondant au tableau 3.3; Droite : le test 4 correspondant au tableau 3.4 .

inférieure à 200 : presque l'image entière. Pour chaque méthode, nous comparons les résultats trouvés avec la valeur précise donnée par l'homographie.

Les résultats sont montrés dans les tableaux 3.1, 3.2, 3.3 et 3.4. Pour chaque méthode (rang, census, rzssdp, et ZSSD) sont indiqués :

1. le nombre d'appariements ayant une erreur inférieure à un pixel (appariement précis).
2. le nombre d'appariements ayant une erreur entre un et deux pixels (bon appariement).
3. le nombre d'appariements ayant une erreur entre deux et trois pixels (mauvais appariement).
4. le nombre d'appariements ayant une erreur supérieure à trois pixels (faux appariement).

Discussions À travers des tableaux 3.1, 3.2, 3.3 et 3.4, nous constatons voyons que la méthode rzssdp donne de meilleurs résultats, avec moins de 10% de faux appariements. La méthode census donne le deuxième meilleur résultat, et comme

<i>rang</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
246	96	0	57

<i>census</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
305	63	0	31

<i>rzssdp</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
316	63	0	20

<i>zssd</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
188	37	0	174

<i>rang</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
713	98	0	89

<i>census</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
773	83	0	44

<i>rzssdp</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
796	74	0	30

<i>zssd</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
658	38	3	201

TAB. 3.1 – Le premier exemple sur l'image *benetton*.
 TAB. 3.2 – Le deuxième exemple sur l'image *benetton*.

<i>rang</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
77	14	0	49

<i>census</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
89	12	0	39

<i>rzssdp</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
118	10	0	12

<i>zssd</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
56	0	0	84

<i>rang</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
129	15	0	122

<i>census</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
185	6	0	75

<i>rzssdp</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
234	29	2	1

<i>zssd</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
71	1	0	194

TAB. 3.3 – Le premier exemple sur l'image *world*.
 TAB. 3.4 – Le deuxième exemple sur l'image *world*.

<i>rang</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
12500	2053	3	214

<i>census</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
12885	1745	0	140

<i>rzssdp</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
12585	1930	4	251

<i>zssd</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
12615	1580	0	575

<i>rang</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
11946	2512	299	968

<i>census</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
12960	2237	105	423

<i>rzssdp</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
11845	1985	193	1702

<i>zssd</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
13140	1941	116	528

TAB. 3.5 – Exemple pour le cas général sur l'image Benetton. TAB. 3.6 – Exemple pour le cas général sur l'image world.

mentionné par Zabih, la méthode de rang est encore moins bonne. ZSSD est la pire, et il donne beaucoup trop de faux appariements.

Mais dans le cas des régions non occultées, nous obtenons le meilleur résultat avec census ou ZSSD.

Il y a plusieurs explications pour ces résultats. Comme census et ZSSD considèrent la fenêtre entière, le support pour leurs réponses est plus grand et ils sont plus robustes au bruit. Au lieu de considérer le signal global, la corrélation partielle peut choisir une fausse fenêtre avec un bon score d'appariement. Ce phénomène est illustré dans la figure 3.9 sur un signal de dimension 1.

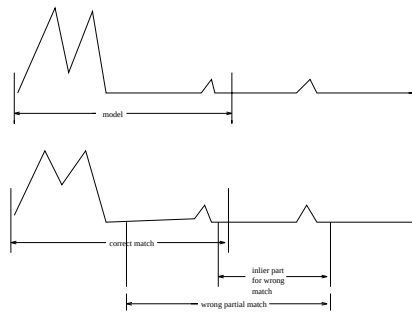


FIG. 3.9 – La translation du signal. La partie à gauche est trouvée fausse et rejetée et la partie droite appariée très bien. S'il n'y a pas de translation, nous ne trouverons pas de fausse partie et les fenêtres entières s'apparient moins bien qu'après la translation.

Donc, nous pensons que la corrélation partielle est plutôt applicable pour les zones d'occultations partielles. La difficulté fondamentale pour cette technique est le conflit entre la *similitude* et la *complétude* de la corrélation :

Étant donné un modèle et plusieurs candidats, quel est le plus similaire au modèle ?

Deux sous ensembles des fenêtres peuvent s'apparier parfaitement (*similitude*), les fenêtres entières peuvent s'apparier aussi (*complétude*), mais moins bien. Un critère doit être choisi quand ces deux critères s'opposent. Ce n'est pas facile car la complétude n'est pas importante proche des contours d'occultations, mais importante dans les autres régions.

Mais au moins, on a montré que la corrélation partielle est bien adaptée aux points près des contours d'occultations. Si l'on applique l'appariement pour chaque pixel, on obtient une carte de disparité dense. Les contours de cette carte correspondent aux points de discontinuités des disparités, ceux sur les contours d'occultations.

Si l'on applique la détection de contours par le détecteur de Canny [Can86] (avec les paramètres $\alpha = 2$, $\omega = 0.001$, $up_{thr} = 13.1174$, $down_{thr} = 1.31174$ et $min_{length} = 10$) sur une carte de disparité, on peut obtenir les points sur les contours d'occultations. Dans les figures 3.1.3.2 et 3.1.3.2 sont affichés les contours d'occultations qu'on obtient pour la paire d'images *Benetton*, à partir des cartes de disparité différentes venant des corrélations différentes utilisées.

Les quatre techniques ont toutes des difficultés pour les zones non texturées. Pour le coté à gauche de jouet, les technique ZSSD et rang donnent des résultats éloignées aux vrais contours, tandis que census et rzssdp donnent de meilleurs résultats, entre ces deux techniques, rzssdp donne des résultat encore plus proches des vrais contours.

3.2 Corrélation partielle basée sur MVE

La fonction de corrélation ZNCC est fréquemment utilisée pour l'appariement visuel. Mais elle n'est pas robuste face aux occultations partielles. Dans cette section, nous analysons d'abord cette fonction au point de vue de la localisation de l'estimation de dispersion, puis nous en déduisons une fonction robustifiée, basée sur la localisation robuste.

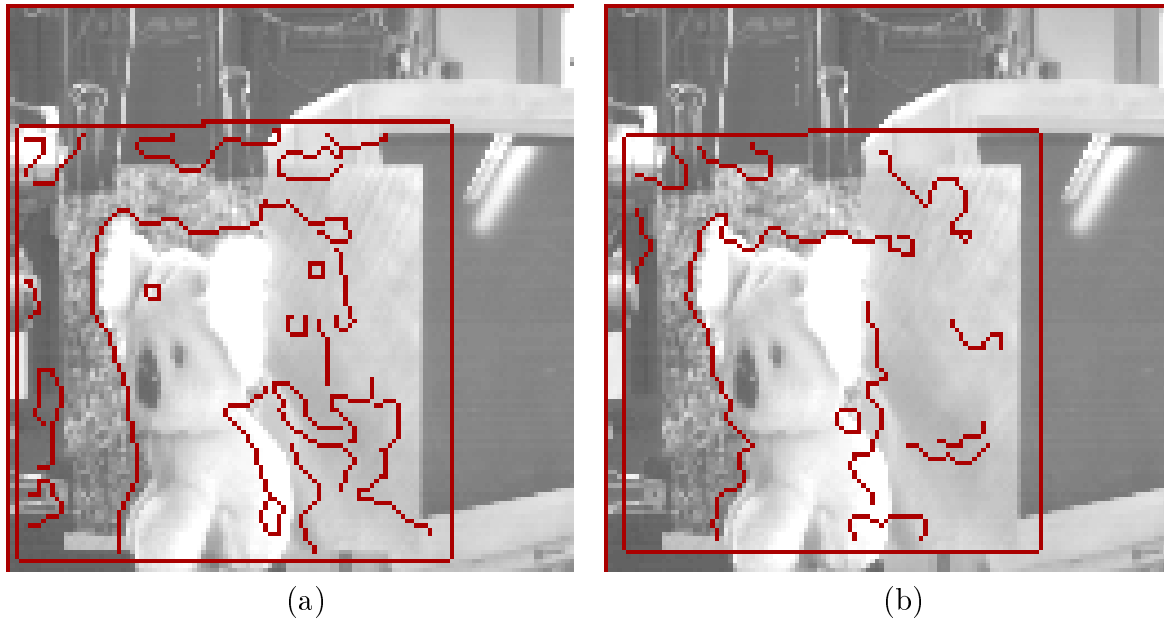


FIG. 3.10 – Contours d'occultations à partir des techniques de corrélation différentes. (a): zssd; (b): rang.

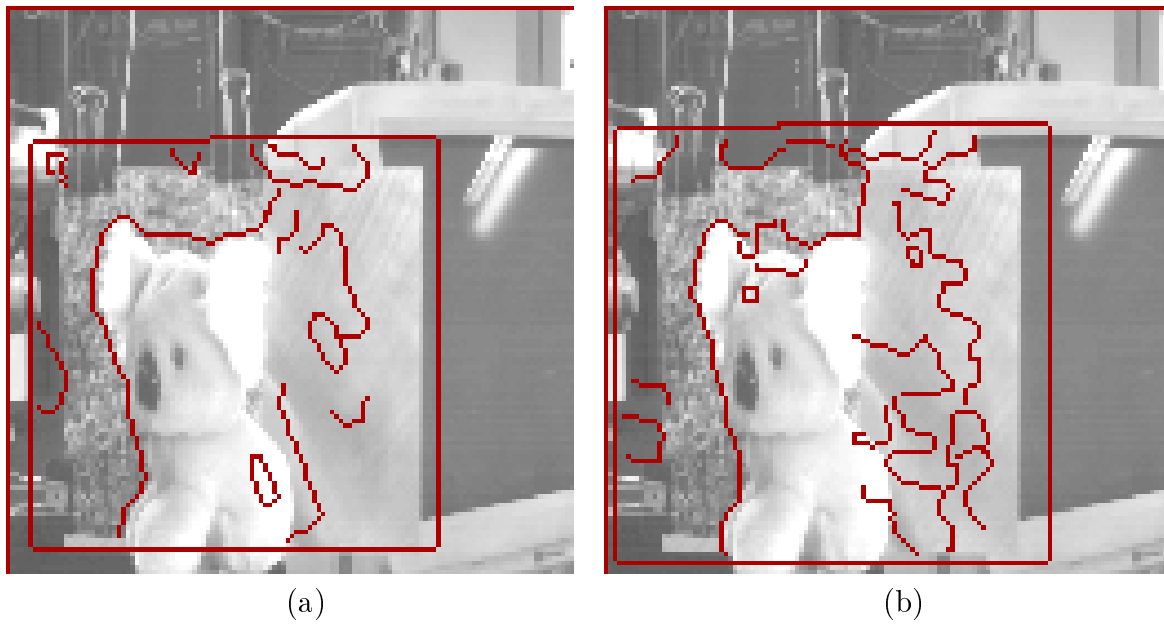


FIG. 3.11 – Contours d'occultations à partir des techniques de corrélation différentes. (a): census; (b): rzssdp.

3.2.1 Analyse de ZNCC au point de vue de localisation

La fonction *ZNCC* de $X = (x_i)$ et $Y = (y_i)$ est définie comme :

$$\frac{\sum_{i=1}^n \frac{(x_i - \bar{X})(y_i - \bar{Y})}{n-1}}{\sqrt{\sum_{i=1}^n \frac{(x_i - \bar{X})^2}{n-1} \sum_{i=1}^n \frac{(y_i - \bar{Y})^2}{n-1}}},$$

qui peut aussi être exprimée comme le rapport de la covariance σ_{XY} entre X et Y par la moyenne géométrique de leurs variances σ_X^2 et σ_Y^2 .

Si l'on constitue une liste de points bidimensionnels $P = (P_i) = (x_i, y_i)^t$, alors le centre de cette liste est de coordonnées $(\bar{X}, \bar{Y})$, donnés par $\bar{P} = (\bar{X}, \bar{Y})^t = (\frac{\sum_{i=1}^n x_i}{n}, \frac{\sum_{i=1}^n y_i}{n})^t$; sa dispersion en terme de la matrice de covariance est donnée par :

$$C = \frac{\sum_{i=1}^n (P_i - \bar{P})(P_i - \bar{P})^t}{n-1} = \begin{pmatrix} \frac{\sum_{i=1}^n (x_i - \bar{X})^2}{n-1} & \frac{\sum_{i=1}^n (x_i - \bar{X})(y_i - \bar{Y})}{n-1} \\ \frac{\sum_{i=1}^n (x_i - \bar{X})(y_i - \bar{Y})}{n-1} & \frac{\sum_{i=1}^n (y_i - \bar{Y})^2}{n-1} \end{pmatrix}$$

et $ZNCC(X, Y) = \frac{C_{12}}{\sqrt{C_{11}C_{22}}}$, où C_{ij} est l'élément de C à la ligne i et la colonne j .

Nous voyons que la fonction *ZNCC* de $X = (x_i)$ et $Y = (y_i)$ peut se calculer à partir de la dispersion de $P = (P_i) = (x_i, y_i)^t$.

Cette procédure est constituée en trois étapes :

1. Estimation du centre des données.
2. Estimation de la matrice de covariance de données.
3. Estimation du coefficient de corrélation en calculant le rapport de la covariance de deux variables par la moyenne géométrique de leurs variances.

Dans le cas d'occultations partielles, il y a des données aberrantes, l'estimation du centre et de la matrice de covariance de donnée doit être faite de manière robuste pour que l'estimation du coefficient de corrélation soit fiable.

3.2.2 Estimation robuste de localisation et de matrice de covariance

Dans cette section, nous introduisons d'abord la notion d'un estimateur de localisation. Puis nous introduisons les notions de quelques propriétés comme in-

variance par les translations, par les permutations, par les transformations affines, et aussi la notion du seuil de rupture. À la fin, nous montrons des estimateurs robustes, et nous entrons dans plus de détail pour l'estimateur MVE, qui est invariant par les transformations affines. Cet estimateur est appliqué pour déduire une fonction de corrélation robuste face aux occultations partielles.

3.2.2.1 Estimation de localisation et de dispersion

Nous discutons l'estimation du «centre» d'un ensemble de points, où toutes les variables sont traitées de la même manière. Aussi nous nous intéressons à la dispersion de données autour du «centre», qui est exprimée par la matrice de covariance de cet ensemble de points.

Supposons que nous avons un ensemble de données :

$$\begin{aligned}\mathbf{X} &= \{\mathbf{x}_1, \dots, \mathbf{x}_n\} \\ &= \{(\mathbf{x}_{11}, \mathbf{x}_{12}, \dots, \mathbf{x}_{1p})^t, \dots, (\mathbf{x}_{n1}, \mathbf{x}_{n2}, \dots, \mathbf{x}_{np})^t\}\end{aligned}$$

de n points en dimension p , et nous allons à estimer son «centre» par un estimateur $\mathbf{T}$, et sa dispersion de son «centre» exprimée par une matrice de covariance C .

Plus exactement, on cherche un estimateur $\mathbf{T}$, tel que les $\mathbf{e}_i = \mathbf{x}_i - \mathbf{T}(\mathbf{X})$ suivent une distribution supposée et vérifient de bonnes propriétés. Si l'on tolère des exceptions parmi les $\mathbf{e}_i$ qui ne suivent pas la distribution, cet estimateur devient robuste face aux données aberrantes.

3.2.2.2 Quelques définitions et un exemple

Nous introduisons des notions des propriétés d'un estimateur robuste : les notions d'invariance, et la notion du seuil de rupture.

Définition 1 *Un estimateur de localisation $\mathbf{T}$ est appelé invariant par les translations, si et seulement si pour n'importe quel vecteur $\mathbf{b}$,*

$$\mathbf{T}(\mathbf{x}_1 + \mathbf{b}, \dots, \mathbf{x}_n + \mathbf{b}) = \mathbf{T}(\mathbf{x}_1, \dots, \mathbf{x}_n) + \mathbf{b}.$$

Définition 2 *Un estimateur de localisation $\mathbf{T}$ est dit invariant par les permutations si et seulement si pour chaque permutation π sur $\{1, 2, \dots, n\}$,*

$$\mathbf{T}(\mathbf{x}_{\pi(1)}, \dots, \mathbf{x}_{\pi(n)}) = \mathbf{T}(\mathbf{x}_1, \dots, \mathbf{x}_n).$$

Donnons un exemple d'estimateur qui vérifie ces deux propriétés :

$$\mathbf{T}(\mathbf{X}) = \bar{\mathbf{X}} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i.$$

C'est la moyenne d'échantillons, l'estimateur le plus connu, qui est l'estimateur aux moindres carrés, car il minimise $\sum_{i=1}^n \|\mathbf{x}_i - \mathbf{T}\|^2$, où $\|\dots\|$ est la norme normale euclidienne. Néanmoins, cet estimateur n'est pas *robuste*, car un seul mauvais point peut totalement corrompre $\mathbf{T}(\mathbf{X}) = \bar{\mathbf{X}}$.

Précisons la notion de la robustesse pour un estimateur de localisation. Considérons tous les échantillons corrompus obtenus en remplaçant n'importe quels m points de données par des valeurs arbitraires, et nous définissons le biais maximal par :

$$bias(m; \mathbf{T}, \mathbf{X}) = \sup_{\mathbf{X}'} \|\mathbf{T}(\mathbf{X}') - \mathbf{T}(\mathbf{X})\|$$

et le seuil de rupture est défini comme :

$$\epsilon_n^*(\mathbf{T}, \mathbf{X}) = \min \{m/n; bias(m; \mathbf{T}, \mathbf{X}) \text{ est infini}\}.$$

Ce seuil est le pourcentage minimal de contaminations qui peuvent totalement abîmer $\mathbf{T}$.

La moyenne a un seuil de rupture $\frac{1}{n}$. Nous considérons souvent le seuil de rupture pour $n \rightarrow \infty$, donc, la moyenne a un seuil de rupture asymptote de 0%.

Il est clair qu'un estimateur $\mathbf{T}$ invariant par les translations ne peut pas avoir un seuil de rupture supérieure à 50%, car on peut avoir les fausses données traduites par les bons points de données, ce qui rend le bon choix de $\mathbf{T}$ impossible.

Dans le cas unidimensionnel, cette bande 50% peut être atteinte facilement, par exemple, par la médiane des échantillons. Ses généralisations ont été considérées dans le cas multidimensionnel pour avoir des estimateurs robustes [RL87].

Les estimateurs de localisation sont classés dans deux familles : ceux qui sont invariants par les transformations affines et ceux qui ne le sont pas. Dans beaucoup de situations, on espère d'avoir un estimateur qui commute avec les transformations affines. On dit que T est invariant par les transformations affines si et seulement si :

$$T(\mathbf{x}_1 \mathbf{A} + \mathbf{b}, \dots, \mathbf{x}_n \mathbf{A} + \mathbf{b}) = T(\mathbf{x}_1, \dots, \mathbf{x}_n) \mathbf{A} + \mathbf{b}$$

pour n'importe quel vecteur $\mathbf{b}$ et n'importe quelle matrice $\mathbf{A}$ régulière. Par exemple, la moyenne est un tel estimateur. Notons que toutes les translations

et permutations sont affines, donc, l'invariance par les transformations affines implique l'invariance par les translations et les permutations.

3.2.2.3 Des estimateurs robustes

Nous introduisons d'abord des estimateurs qui ne sont pas invariants par les transformations affines, puis un qui est invariant.

Estimateurs robustes affinement non invariants L'idée la plus simple pour avoir un estimateur robuste de localisation dans le cas multidimensionnel est de considérer chaque composante séparément. Pour chaque variable j , les nombres $\mathbf{x}_{1j}, \mathbf{x}_{2j}, \dots, \mathbf{x}_{nj}$ constituent un ensemble de n réels. On applique donc un estimateur robuste à chaque ensemble et combine les résultats. Ce type d'estimateurs hérite du seuil de rupture de l'estimateur original. Par exemple, l'estimateur de la médiane des coordonnées ($\underset{i}{\text{med}} \mathbf{x}_{i1}, \underset{i}{\text{med}} \mathbf{x}_{i2}, \dots, \underset{i}{\text{med}} \mathbf{x}_{in}$) possède un seuil de rupture 50%. Il n'est pourtant pas invariant par les transformations affines.

Quelques fois, on veut avoir seulement l'invariance euclidienne pour un estimateur de localisation, c'est-à-dire, la propriété préservant la distance euclidienne. Il suffit que l'estimateur commute avec les transformations de type $\mathbf{x} \rightarrow \mathbf{x}, +b$, où b est une matrice orthogonale. Ces transformations contiennent les translations, les rotations, et les réflexions. Par exemple, l'estimateur de localisation L_1 donné par la solution de $\min_T \sum_{i=1}^n \|\mathbf{x}_i - T\|$ est orthogonalement invariant. Il est la généralisation de la médiane unidimensionnelle, sachant que $\min_T \sum_{i=1}^n |x_i - T|$ donne $T = \underset{i}{\text{med}} x_i$ pour les nombres réels x_i .

Cet estimateur peut encore se généraliser facilement à un M -estimateur, donné par $\min_T \sum_{i=1}^n \rho(\|\mathbf{x}_i - T\|)$, où ρ est une fonction définie positive.

Un estimateur robuste invariant par des transformations affines Rousseeuw [RL87] a introduit un estimateur ayant le seuil de rupture maximal, défini comme :

$\mathbf{T}(\mathbf{X}) =$ centre de l'ellipse de volume minimal contenant au moins h points de $\mathbf{X}$.

où h peut être pris comme $\lceil \frac{n}{2} \rceil + 1$. Il est appelé *ellipse de volume minimal* (MVE en anglais). L'estimation de la matrice de covariance est donnée par l'ellipse elle-même, multiplié par un facteur d'échelle convenable.

L'invariance affine de cet estimateur suit du fait que l'image d'un ellipse par une transformation affine $\mathbf{x} \rightarrow \mathbf{A}\mathbf{x} + \mathbf{b}$ est encore un ellipse, avec le volume égal à $|\det \mathbf{A}|$ multiplie le volume original. $|\det \mathbf{A}|$ étant constant, la dimension relative de l'ellipse ne change pas sous les transformations affines.

Il n'est pas pratique de considérer toutes les moitiés des données et de calculer le volume de la plus petite ellipse autour d'elles. Comme pour l'estimateur LMedS, pour implémenter cet estimateur, nous choisissons un ensemble de $p+1$ observations différentes, indexées par $\mathbf{J} = \{i_1, \dots, i_{p+1}\}$. Pour ce sous-ensemble, nous déterminons la moyenne et la matrice de covariance correspondante par :

$$\overline{\mathbf{X}}_{\mathbf{J}} = \frac{1}{p+1} \sum_{i \in \mathbf{J}} \mathbf{x}_i \quad \text{et} \quad \mathbf{C}_{\mathbf{J}} = \frac{1}{p} \sum_{i \in \mathbf{J}} (\mathbf{x}_i - \overline{\mathbf{X}}_{\mathbf{J}})(\mathbf{x}_i - \overline{\mathbf{X}}_{\mathbf{J}})^t,$$

où $\mathbf{C}_{\mathbf{J}}$ est régulière si $\mathbf{x}_{i_1}, \dots, \mathbf{x}_{i_{p+1}}$ sont dans une position générale. Par la définition de la médiane, l'ellipse contenant la moitié des données correspond à la matrice $m_{\mathbf{J}}^2 \mathbf{C}_{\mathbf{J}}$, où $m_{\mathbf{J}}^2 = \underset{i}{\text{med}} (\mathbf{x}_i - \overline{\mathbf{X}}_{\mathbf{J}})^t \mathbf{C}_{\mathbf{J}}^{-1} (\mathbf{x}_i - \overline{\mathbf{X}}_{\mathbf{J}})$. Il suffit de voir que la moitié de $\mathbf{x}_i$ vérifient $(\mathbf{x}_i - \overline{\mathbf{X}}_{\mathbf{J}})^t (m_{\mathbf{J}}^2 \mathbf{C}_{\mathbf{J}})^{-1} (\mathbf{x}_i - \overline{\mathbf{X}}_{\mathbf{J}}) = \frac{(\mathbf{x}_i - \overline{\mathbf{X}}_{\mathbf{J}})^t \mathbf{C}_{\mathbf{J}}^{-1} (\mathbf{x}_i - \overline{\mathbf{X}}_{\mathbf{J}})}{m_{\mathbf{J}}^2} \leq 1$.

Le volume d'ellipse est proportionnel à :

$$\sqrt{\det(m_{\mathbf{J}}^2 \mathbf{C}_{\mathbf{J}})} = \sqrt{\det(\mathbf{C}_{\mathbf{J}})} m_{\mathbf{J}}^p. \quad (3.9)$$

Nous répétons ce processus pour plusieurs $\mathbf{J}$, et nous prenons le $\mathbf{J}$ qui donne la valeur minimale dans l'équation (3.9).

Puis nous calculons :

$$\mathbf{T}(\mathbf{X}) = \overline{\mathbf{X}}_{\mathbf{J}} \quad \text{et} \quad \mathbf{C}(\mathbf{X}) = (\chi_{p,0.5}^2)^{-1} m_{\mathbf{J}}^2 \mathbf{C}_{\mathbf{J}}$$

où $\chi_{p,0.5}^2$ est la médiane de la distribution chi-deux de degré de liberté p .

L'estimateur MVE peut être utilisé comme une solution initiale suivie d'un pas d'amélioration. Comme pour le cas de l'estimation aux moindres carrés pondérés, nous donnons un poids à chaque observation par la règle :

$$\omega_i = \begin{cases} 1 & \text{si } (\mathbf{x}_i - \mathbf{T}(\mathbf{X}))^t \mathbf{C}(\mathbf{X})^{-1} (\mathbf{x}_i - \mathbf{T}(\mathbf{X})) \leq c \\ 0 & \text{sinon} \end{cases}$$

où c peut prendre la valeur $\chi_{p,0.975}^2$, qui signifie qu'on rejette une donnée quand on a 97.5% de confiance qu'elle soit aberrante.

Puis nous pouvons employer l'estimateur de la moyenne repondérée :

$$\mathbf{T}_1(\mathbf{X}) = \frac{\sum_{i=1}^n \omega_i \mathbf{x}_i}{\sum_{i=1}^n \omega_i} \quad \text{et} \quad \mathbf{C}_1(\mathbf{X}) = \frac{\sum_{i=1}^n \omega_i (\mathbf{x}_i - \mathbf{T}_1(\mathbf{X}))(\mathbf{x}_i - \mathbf{T}_1(\mathbf{X}))^t}{\sum_{i=1}^n \omega_i - 1},$$

qui signifie simplement que le calcul classique se fait sur les observations dont le poids est égal à 1.

3.2.3 Corrélation partielle basée sur la localisation robuste

Nous commençons par la discussion de la relation entre la corrélation et la localisation, puis nous en déduisons la procédure de la corrélation partielle, à la fin, nous introduisons deux contraintes : la contrainte des *points principaux* (page 91) et la contrainte de la *fenêtre principale* (page 91), qui peuvent s'employer dans l'appariement de deux images.

3.2.3.1 Analyse de corrélation et de localisation

Supposons que nous avons une liste de n points en dimension p : $\mathbf{X} = (\mathbf{x}_i)_{i=1}^n$, où $\mathbf{x}_i = (\mathbf{x}_{i1}, \dots, \mathbf{x}_{ip})$, nous cherchons un estimateur $\mathbf{T} = \mathbf{T}(\mathbf{X})$, tel que :

$$\mathbf{x}_i = \mathbf{T} + \mathbf{e}_i \tag{3.10}$$

où les $\mathbf{e}_i = (\mathbf{e}_{i1} \dots \mathbf{e}_{ip})$ sont des variables aléatoires vectorielles (de distribution identique).

Le problème de localisation est la recherche d'un tel estimateur. Sous l'hypothèse que les $\mathbf{e}_i$ suivent une distribution identique, le coefficient de corrélation entre $\mathbf{e}_{ij}$ et $\mathbf{e}_{ik}$ ne dépendent que de j et k . Il peut se calculer à l'aide de la matrice de covariance de $\mathbf{e}_i$, comme le rapport de la covariance entre $\mathbf{e}_{ij}$ et $\mathbf{e}_{ik}$ par la moyenne géométrique des variances de $\mathbf{e}_{ij}$ et de $\mathbf{e}_{ik}$.

3.2.3.2 Procédure de la corrélation partielle

Dans la présence d'occultation, la relation (3.10) n'est retenue que pour la partie non occultée. Donc on doit calculer la mesure de corrélation seulement sur cette partie.

Nous proposons donc la procédure de la *corrélation partielle* :

1. Estimer $\mathbf{T}$ de manière robuste, et aussi la matrice de covariance C associée.
2. Récupérer des informations robustes à partir de $\mathbf{T}$ obtenu.
3. Trouver la partie occultée selon un test statistique.
4. Calculer la mesure de corrélation sur la partie non occultée.

Nous expliquons ces quatre étapes en détails par des exemples.

1. On peut prendre $\mathbf{T}$ comme l'estimateur de la médiane des coordonnées ($\text{med}_i \mathbf{x}_{i1}, \text{med}_i \mathbf{x}_{i2}, \dots, \text{med}_i \mathbf{x}_{ip}$), ou un estimateur plus sophistiqué comme celui de MVE.
2. Les informations robustes sont celles insensibles aux erreurs grossières. Par exemple, la médiane des $(\mathbf{x}_i - \mathbf{T})^t \mathbf{C}^{-1} (\mathbf{x}_i - \mathbf{T})$ est robuste, et la moyenne des $(\mathbf{x}_i - \mathbf{T})^t \mathbf{C}^{-1} (\mathbf{x}_i - \mathbf{T})$ ne l'est pas.
3. Après l'étape précédente, on estime la fonction de distribution d'erreurs et décide si une erreur appartient à cette distribution selon un test statistique.
Exemple : dans l'estimateur MVE, $(\mathbf{x}_i - \mathbf{T})^t \mathbf{C}^{-1} (\mathbf{x}_i - \mathbf{T})$ est une variable aléatoire de distribution chi-deux de degré p , si toutes les $\mathbf{e}_i$ sont gaussiennes centrées, et si la valeur de cette expression dépasse un certain seuil c ($c = \chi_{0.975}^2$ par exemple), la donnée $\mathbf{x}_i$ est déclarée aberrante.
4. La corrélation partielle peut se réaliser à l'aide de la corrélation pondérée binaire. Par exemple, si l'on laisse $\omega_i = 1$ (bon point) ou $\omega_i = 0$ (mauvais point), la corrélation partielle de $ZNCC$ peut s'exprimer sous la forme de la corrélation pondérée comme suit :

$$ZNCC_p(X, Y) = \frac{\frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})\omega_i}{\sum_{i=1}^n \omega_i - 1}}{\sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2 \omega_i}{\sum_{i=1}^n \omega_i - 1}} \sqrt{\frac{\sum_{i=1}^n (y_i - \bar{y})^2 \omega_i}{\sum_{i=1}^n \omega_i - 1}}}$$

où $\bar{x}$ (respectivement $\bar{y}$) sont les moyennes pondérées de x_i (respectivement y_i), de pondérations ω_i .

3.2.3.3 Les contraintes

Nous introduisons deux contraintes pour les appariements des points dans deux images par la corrélation partielle basée sur l'estimateur MVE.

La contrainte des points principaux Comme indiqué à la page 21, pour avoir une confiance de Q , il suffit que le nombre de tirage aléatoire soit supérieur à $\frac{\log Q}{\log(1-(1-\epsilon)^p)}$. Ici, p le nombre de paramètres est 3, et ϵ est le pourcentage d'erreur aberrantes ($\epsilon = 1 - \omega$). Supposons que $\epsilon = 0.5$ sa valeur maximale autorisée, et $Q = 0.01$, nous avons $\frac{\log Q}{\log(1-(1-\epsilon)^p)} = 34.48$ donc le nombre de tirages est 35. Ce qui indique que l'algorithme est assez coûteux.

Dans nos expériences, nous divisons la petite fenêtre 3×3 autour du pixel à apparier en quatre parties, comme suit :

3	4	4
3	c	1
2	2	1

Les chiffres dans ce tableau numérotent les régions différentes, et c note le pixel à apparier. Nous supposons que le point à apparier n'est pas occulté, et de plus, au moins deux pixels d'une région ne sont pas occultés. C'est-à-dire : les deux pixels numérotés 1 ou 2 ou 3 ou 4 ne sont pas occultés.

Cette contrainte, ressemblant à la contrainte du point principal dans la corrélation partielle par la régression partielle (équation (3.4) à la page 72), s'appelle *la contrainte des points principaux*. Et les points autour du point C s'appellent les points principaux.

En employant cette contrainte, au lieu de faire 35 échantillonnages aléatoires, nous prenons c avec ces quatre groupes de pixels pour les échantillonnages, et le nombre de tirages se réduit à 4.

Les avantages de cette contrainte est 1) elle réduit le nombre de tirages et le temps de calcul ; 2) elle incorpore le fait que beaucoup de points voisins au point à apparier ne sont pas occultés. Son désavantage, est qu'elle rend l'estimation de localisation dépend plus des points principaux.

La contrainte de la fenêtre principale Similaire à la contrainte de la fenêtre principale dans la corrélation partielle basée sur la régression partielle (page 73), nous observons que les points proches du point à apparier ont moins de chances d'être occultés. Et pour le calcul de $m_{\mathbf{J}}^2$, on peut prendre la médiane des

$$(\mathbf{x}_i - \overline{\mathbf{X}_{\mathbf{J}}})^t \mathbf{C}_{\mathbf{J}}^{-1} (\mathbf{x}_i - \overline{\mathbf{X}_{\mathbf{J}}})$$

dans une sous fenêtre au lieu de le faire pour la fenêtre entière, comme dans la corrélation partielle basée sur la régression partielle (page 73).

Cette contrainte est appelée *la contrainte de la fenêtre principale* et la fenêtre associée est appelée *la fenêtre principale*.

Son avantage est l'accélération de vitesse. Son inconvénient est la forte dépendance de la fenêtre principale. Elle ne sera pas retenue dans la suite.

3.2.4 Application en vision et résultats expérimentaux

L'estimateur MVE a été déjà utilisé dans la vision, sous la forme d'un algorithme de classification. Par exemple, dans [JMB91], il a été utilisé pour le seuillage d'images, l'analyse d'espace de Hough, et la segmentation d'images.

3.2.4.1 Application à l'estimation du coefficient de corrélation

Nous utilisons l'estimateur MVE pour déduire une fonction de corrélation pour l'appariement.

Dans le cas d'appariement d'images, nous prenons $\mathbf{x}_i = (I_{1i}, I_{2i})^t$, où I_{1i} (I_{2i}) sont les intensités lumineuses du $i^{\text{ème}}$ point de la première (deuxième) fenêtre d'images respectivement. Soit $\mathbf{C}_1(X) = \begin{pmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{12} & \sigma_2^2 \end{pmatrix}$ la matrice de covariance de $\mathbf{x}_i$, estimée de l'estimateur MVE, nous estimons le coefficient de corrélation de I_1 et I_2 comme $\frac{\sigma_{12}}{\sigma_1 \sigma_2}$.

Nous appelons cette corrélation basée sur l'estimateur de localisation MVE, la corrélation MVE.

Cette estimation est la généralisation du coefficient de corrélation classique si $\mathbf{C}_1(\mathbf{X})$ est remplacée par celle classique, qui est calculée comme :

$$\begin{pmatrix} \frac{\sum_{i=1}^n (I_{i1} - \bar{I}_1)^2}{n-1} & \frac{\sum_{i=1}^n (I_{i1} - \bar{I}_1)(I_{i2} - \bar{I}_2)}{n-1} \\ \frac{\sum_{i=1}^n (I_{i1} - \bar{I}_1)(I_{i2} - \bar{I}_2)}{n-1} & \frac{\sum_{i=1}^n (I_{i2} - \bar{I}_2)^2}{n-1} \end{pmatrix}$$

où $\bar{I}_1 = \frac{\sum_{i=1}^n I_{i1}}{n}$ et $\bar{I}_2 = \frac{\sum_{i=1}^n I_{i2}}{n}$ sont les moyennes des niveaux de gris de deux fenêtres. Ce coefficient de corrélation classique, appelée souvent *ZNCC*, est une fonction de corrélation fréquemment employée en vision pour le problème d'appariement.

Un exemple réel est montré dans la figure 3.12.

Dans la figure 3.12 sont affichés des couples de niveaux de gris des pixels appariés dans deux images. Chaque couple est représenté par un point, et l'ensemble de points représentant un couple de pixels appariés doivent se situer dans une ellipse. Cette ellipse est similaire à l'ellipse d'incertitude de (I_1, I_2) . Les points

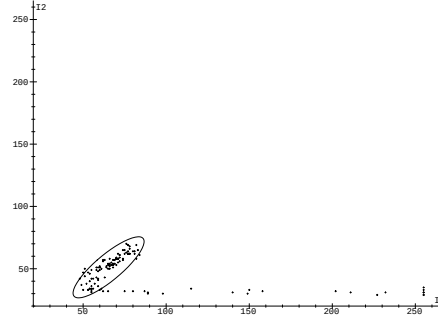


FIG. 3.12 – Localisation robuste par l'estimateur MVE de (I_1, I_2) , les points dehors de l'ellipse représentent des couples de points non appariés.

non appariés sont en dehors d'elle.

3.2.4.2 Résultats expérimentaux

On a employé cette technique sur les paires d'images *Benetton* et *World* dans les figures 3.5 et 3.6, respectivement. Les points de test utilisés sont les mêmes que dans 3.7 et 3.8. Des comparaisons sont faites avec la technique de ZNCC, car cette technique peut être vue comme la version robuste de ZNCC.

Dans les tableaux 3.7, 3.8, 3.11 3.9, 3.10, 3.12 sont montrés les résultats. Nous pouvons aussi voir les tableau 3.1, 3.2, 3.3, 3.4, 3.5, 3.6 pour comparer avec les autres méthodes.

<i>MVE</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
270	79	0	50

<i>ZNCC</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
183	37	0	179

TAB. 3.7 – Le résultat de MVE pour le test du tableau 3.1.

<i>MVE</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
730	108	0	56

<i>ZNCC</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
656	38	1	205

TAB. 3.8 – Le résultat de MVE pour le test du tableau 3.2.

Les résultats des points généraux sont affichés dans les tables 3.11 et 3.12. Similaire que la méthode rzssdp, il y a un conflit entre la *similitude* et la *complétude* à résoudre (page 82).

<i>MVE</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
94	10	1	35

<i>ZNCC</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
56	0	0	84

TAB. 3.9 – Le résultat de MVE pour le premier test du tableau 3.3.

<i>MVE</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
173	49	7	37

<i>ZNCC</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
78	0	0	188

TAB. 3.10 – Le résultat de MVE pour le deuxième test du tableau 3.4.

<i>MVE</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
12378	2183	40	169

<i>ZNCC</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
12771	1550	0	449

TAB. 3.11 – Le résultat de MVE pour le test du tableau 3.5.

<i>MVE</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
11102	2512	414	1697

<i>ZNCC</i>			
0 – -1	1 – -2	2 – -3	3 – -∞
13216	1970	80	459

TAB. 3.12 – Le résultat de MVE pour le test du tableau 3.6.

3.3 Conclusion

Dans ce chapitre, nous avons présenté deux corrélations partielles qui écartent la partie occultée. Ces deux méthodes, l'une basée sur la régression robuste, l'autre sur la localisation robuste, fonctionnent bien pour les points proches de contours d'occultation.

Plusieurs perspectives de ce travail sur la corrélation partielle peuvent être envisagées :

- L'intégration dans un processus de programmation dynamique pour l'appariement dense pour avoir plus d'appariements fiables proches de contours d'occultation.
- Localisation précise de cibles partielles occultées en détectant l'occultation partielle.
- Application à la détection de la déformation perspective.

Dans ce chapitre, nous n'avons pas encore tenu compte de la précision de l'appariement, ni la déformation perspective. Dans le chapitre suivant, nous allons présenter une méthode précise et robuste pour l'appariement, qui tient en compte de la déformation perspective par l'approximation d'une transformation affine.

Chapitre 4

Corrélation précise

4.1 Introduction

Dans ce chapitre, nous nous intéressons non seulement à la fiabilité de l'appariement en stéréo vision mais aussi à sa précision. Cette précision a une importance décisive pour la qualité de la reconstruction tridimensionnelle.

Nous proposons une méthode linéaire, qui incorpore les informations des pixels voisins. À partir de cette méthode, deux algorithmes seront présentés, l'un *rapide* et l'autre *robuste*.

L'*algorithme rapide* de s'applique à une paire d'images où la déformation entre les fenêtres de deux images est négligeable. Une précision d'ordre 0,05 pixel pour les points d'intérêts et 0,06 pour les points généraux est obtenue avec cet algorithme.

L'*algorithme robuste* est plus général et précis. L'algorithme proposé ici est robuste car il élimine de faux appariements dans les initialisations en utilisant l'estimation robuste de la transformation affine locale ; il est plus général dans le sens où il est aussi applicable quand la déformation des fenêtres entre les deux images est grande ; il donne une précision d'ordre 0,05 pixel quand la déformation est petite, et 0.08 quand elle est grande.

Ce chapitre est organisé comme suit : après cette introduction, nous allons discuter la théorie d'échantillonnage du signal, puis de notre modèle de corrélation. Nous expliquons comment nous pouvons obtenir des informations sous pixeliques à partir des coefficients de corrélation, comment nous fusionnons les décalages estimés en utilisant leurs incertitudes estimées ; ce qui constitue l'algorithme rapide. Quand la déformation des fenêtres est non négligeable, nous proposons d'estimer d'abord cette déformation à partir d'une carte de disparité dense, et ensuite d'es-

timer le décalage local ; cette procédure constitue l'algorithme robuste. Pour cet algorithme, le qualificatif de robustesse s'applique à sa capacité d'éliminer de faux appariements dans la carte de disparité initiale, en utilisant la statistique robuste. Des expérimentations sur des images réelles valident l'approche.

4.2 Le cas d'une translation pure : algorithme rapide

Dans cette section, nous considérons le cas où il y a principalement une translation pure entre les deux images à appairer. Cet algorithme est très rapide et la moyenne des erreurs est de l'ordre de 0,06 pixel.

Dans cette section, d'abord nous montrons qu'une translation de signal peut être approximée par une convolution, puis nous montrons que nous pouvons estimer la translation en estimant le masque de cette convolution. Dépendant de la direction de convolution (de l'image 1 à l'image 2 ou de l'image 2 à l'image 1) qu'on choisit, nous pouvons avoir deux estimations de translation. Pour fusionner ces deux estimations, nous estimons aussi leurs incertitudes. À la fin, nous montrons une étape de raffinement. Des expérimentations valident l'approche.

4.2.1 Échantillonnage de signal et décalage sous-pixelique

La translation de signal peut-être modélisée en utilisant la convolution :

Soit δ_t la distribution Dirac centré au point t :

$$\delta_t(x) = \begin{cases} 0 & x \neq t \\ +\infty & x = t \end{cases}, \quad \int \delta_t = 1,$$

alors pour une fonction f , son décalage peut être exprimé comme :

$$f(x+t) = f * \delta_{-t}(x) = \int f(u) \delta_{-t}(x-u) du.$$

Dans le cas discret, quand le décalage est un entier, soit $\delta_t(i) = \begin{cases} 0 & \text{si } i \neq t \\ 1 & \text{si } i = t \end{cases}$,

alors $f(i+t) = \sum_j f(j) \delta_{-t}(i-j) = f * \delta_{-t}(i)$.

Pour un décalage non entier, il n'y a pas de réponse exacte, à cause de la perte de données pendant l'échantillonnage. Néanmoins, nous pouvons supposer la linéarité locale : $f(i+\epsilon) \approx (1-\epsilon)f(i) + \epsilon f(i+1)$ ($0 \leq \epsilon \leq 1$).

Cette approximation linéaire peut être exprimée par l'approximation de $(1 - \epsilon)\delta_0 + \epsilon\delta_{-1}$ vers δ_{-t} , au regard de l'effet de convolution avec f .

Plus généralement, si nous avons b_i, λ_i , tels que $\sum_i b_i = 1$, et $f(x + \lambda_i) \approx k(x + \lambda_i) + l$ (k, l sont constants), alors nous avons $\sum_i b_i f(x + \lambda_i) \approx f(x + \sum_i \lambda_i b_i)$, c'est-à-dire, $\sum_i b_i \delta_{-\lambda_i}$ est une approximation de $\delta_{-\sum_i \lambda_i b_i}$ pour une telle fonction f .

Donc sous cette condition de régularité de signal, un décalage peut-être approximativement modélisé par une convolution, et l'estimation du décalage de signal peut être remplacé par l'estimation de la fonction de convolution correspondante.

4.2.2 Modélisation

Nous introduisons d'abord notre modèle d'appariement précis de signal dans le cas unidimensionnel, et puis nous l'étendons dans le cas bidimensionnel.

4.2.2.1 Translation sous-pixellique par convolution : cas unidimensionnel

Supposons que nous avons deux signaux discrets f_1 et f_2 , échantillonnés des signaux F_1 et F_2 . F_1 et F_2 sont définis sur l'ensemble réel $\mathbb{R}$, f_1 et f_2 sont définis sur des valeurs entières.

Le problème d'appariement est modélisé comme :

Étant données f_1 et f_2 , chercher $t \in \mathbb{R}$, tel que :

$$F_1(x) \sim F_2(x + t) = F_2 * \delta_{-t}.$$

Ici, $\sim$ signifie que les deux signaux sont égaux modulo des transformations telles qu'une transformation affine plus un bruit. Pour nous, nous considérons que : $F_2(x) = SF_1(x + t) + O + \epsilon$ (S : scale; O : offset). Dans ce chapitre, nous sommes intéressés à l'appariement précis. Nous supposons qu'un appariement initial est donné ; sans perte de généralité, nous pouvons supposer que $t \in [-1, 1]$.

Ne connaissant que f_1 et f_2 , nous sommes amenés à chercher $t \in [-1, 1]$, tel que $f_1 \sim f_2 * \delta_{-t}$. Comme f_1 et f_2 sont seulement définies sur des valeurs entières, donc, au lieu de δ_{-t} , nous utilisons l'hypothèse de l'approximation linéaire discrète dans la section 4.2.1. Nous cherchons une combinaison linéaire de δ_{-1} , δ_0 et δ_1 : $b_1\delta_{-1} + b_0\delta_0 + b_{-1}\delta_1$ ($b_1 + b_0 + b_{-1} = 1$), telle que $f_1 \sim f_2 * (b_1\delta_{-1} + b_0\delta_0 + b_{-1}\delta_1)$. Et une fois que les b_i ($i = -1, 0, 1$) sont trouvés, nous approximons $b_1\delta_{-1} + b_0\delta_0 + b_{-1}\delta_1$

par $\delta_{b_{-1}-b_1}$, et le décalage estimé est alors $b_1 - b_{-1}$.

Prenons un exemple, si nous avons deux signaux suivants :

$$\begin{array}{cccccc} f_1 & & & & & f_2 \\ 50 & 50 & 75 & 100 & 100 & 50 & 50 & 50 & 100 & 100 \end{array}$$

alors nous pouvons obtenir $b_1 = b_0 = 0.5, b_{-1} = 0$, le décalage est estimé d'être $b_1 - b_{-1} = 0.5$.

Dans le problème d'appariement du signal, l'effet d'échelle et de translation des intensités lumineuses sont aussi souvent considérés. Donc nous sommes amenés à estimer a_1, a_0 et a_{-1} , tels que

$$f_1 = f_2 * (a_1\delta_{-1} + a_0\delta_0 + a_{-1}\delta_1) + O + \epsilon. \quad (4.1)$$

En posant $b_i = \frac{a_i}{\sum_{i=-1}^1 a_i}$ ($i = -1, 0, 1$), le décalage est estimé comme étant $b_1 - b_{-1}$.

4.2.2.2 Translation sous pixellique par convolution : cas bidimensionnel

Dans le cas bidimensionnel, nous modélisons le problème de l'estimation du décalage comme suit :

$$I_1(X) = \sum_{K \in N} a_K I_2(X + K) + O + \epsilon_X. \quad (4.2)$$

Les deux signaux des niveaux de gris dans deux fenêtres rectangulaires d'images sont notés I_1 et I_2 respectivement, X parcourt une certaine fenêtre choisie, c'est-à-dire $X \in \{(x, y) | s - s_{len} < x < s + s_{len}, t - t_{len} < y < t + t_{len}\} = W_1$ la fenêtre à apparier, N un voisinage d'origine (par exemple $N = \{(i, j) | |i| + |j| \leq 1\} = \{(0, 0), (0, 1), (1, 0), (0, -1), (-1, 0)\}$). $(a_K)_{K \in N}$ (matrice des coefficients de corrélation) représente l'effet du décalage et comme nous verrons plus tard, l'effet d'échelle de niveaux de gris, et O représente une translation de niveau de gris, ϵ est supposé d'être bruit blanc.

À partir de deux signaux I_1 et I_2 , a_K et O peuvent être estimés en minimisant la moyenne des carrés des résidus.

Et l'écart-type de ϵ_X est estimé par

$$\sigma = \sqrt{\sum_{X \in W_1} \frac{(I_1(X) - \sum_{K \in N} a_K I_2(X + dX + K) - O)^2}{(2s_{len} + 1)(2t_{len} + 1)}} \quad (4.3)$$

Nous pouvons aussi établir :

$$I_2(X) = \sum_{K \in N} a'_K I_1(X + K) + O' + \epsilon'_X. \quad (4.4)$$

Maintenant, nous discutons comment nous pouvons récupérer les informations sous-pixeliques à partir des coefficients de corrélation.

Dans l'équation (4.2), nous notons $\sum_{K \in N} a_K = S \neq 0$. $S \approx 1$ si les deux caméras ne se diffèrent pas trop, car S représente l'effet d'échelle, qui vient de la différence des sensibilités des caméras. $\frac{a_K}{S} = b_K$, b_K représente l'effet du décalage, dont la somme $\sum_{K \in N} b_K$ est 1, nous avons :

$$I_1(X) = S \sum_{K \in N} b_K I_2(X + K) + O + \epsilon_X.$$

Sous l'hypothèse de la linéarité locale de niveaux de gris d'images, nous pouvons avoir :

$$I_1(X) \approx S I_2(X + \sum_{K \in N} b_K K) + O + \epsilon.$$

Donc nous obtenons la disparité sous-pixelique $\sum_{K \in N} b_K K$.

Aussi à partir de l'équation (4.4), nous avons :

$$I_2(X) \approx S' I_1(X + \sum_{K \in N} b'_K K) + O' + \epsilon'_X.$$

où S' et b'_K sont définis de façon similaire ; la disparité sous-pixelique est estimée comme $-\sum_{K \in N} b'_K K$.

4.2.3 Prendre en compte les incertitudes des décalages estimés

Dans cette section, nous discutons les incertitudes de deux décalages de l'équation (4.2) et l'équation (4.4). Et puis, nous pouvons les fusionner pour améliorer le résultat.

Soient les deux décalages dX_1 et dX_2 , avec respectivement les matrices de covariance d'estimation de C_1 et C_2 . Alors nous les fusionnons comme suit :

$$(C_1^{-1} + C_2^{-1})^{-1}(C_1^{-1}dX_1 + C_2^{-1}dX_2).$$

Cette fusion est optimale si la distribution d'erreurs est gaussienne et pour n'importe quelle distribution, elle est optimale parmi toutes les combinaisons linéaires de dX_1 et dX_2 au sens des moindres carrés [Aya90].

L'estimation symétrique obtenue à partir de deux estimations de décalage $\sum_{K \in N} b_K K$ et $\sum_{K \in N} b'_K K$ suivies de cette fusion, est appelé *l'algorithme rapide*. Elle ne prend pas en compte l'effet de la déformation des fenêtres dans l'appariement.

Donc, nous sommes amenés à estimer C_1 et C_2 . D'abord, nous introduisons des notations et puis nous discutons le calcul de la matrice de covariance dans le cas de quatre-voisinage et huit-voisinage, respectivement.

4.2.3.1 Des notations

Considérons l'équation (4.2). Soit A le vecteur colonne de tous les $\{a_K\}_{K \in N}$ et $\mathbb{I}_2(X)$ celui des $\{I_2(X + K)\}_{K \in N}$, alors :

$$I_1(X) = (\mathbb{I}_2(X), 1) \begin{pmatrix} A \\ O \end{pmatrix} + \epsilon_X.$$

Soit $\mathbb{I}_1$ le vecteur de colonne de $\{I_1(X)\}_{X \in W}$, $\mathbb{I}_2$ celui de $\{(\mathbb{I}_2(X), 1)\}_{X \in W}$, et $T = \begin{pmatrix} A \\ O \end{pmatrix}$. Nous devons résoudre le système d'équations $\mathbb{I}_1 = \mathbb{I}_2 T$, sous l'hypothèse que ϵ_X sont indépendants et identiquement distribués. Nous résolvons ce système en minimisant la moyenne des erreurs au carré et nous obtenons $T = (\mathbb{I}_2^T \mathbb{I}_2)^{-1} \mathbb{I}_2^T \mathbb{I}_1$. Notons $L = (\mathbb{I}_2^T \mathbb{I}_2)^{-1} \mathbb{I}_2^T = \mathbb{I}_2^+$, le pseudo inverse de $\mathbb{I}_2$, alors $T = L \mathbb{I}_1$. Supposons que $\mathbb{I}_1$ est gaussien, alors T et A le sont aussi.

Supposons que la dimension de W_1 (la fenêtre que X parcourt) soit m , et la dimension de N soit n ; $\mathbb{I}_2$ est de dimension $m \times (n + 1)$, et L est une matrice

de dimension $(n + 1) \times m$, écrite comme $L = \begin{pmatrix} L_1 \\ L_2 \\ \vdots \\ L_{n+1} \end{pmatrix}$, où L est un vecteur de

ligne en dimension m . Notons la matrice de covariance de $dX = (dx, dy)$ comme $\begin{pmatrix} \text{var}(dx) & \text{cov}(dx, dy) \\ \text{cov}(dx, dy) & \text{var}(dy) \end{pmatrix}$.

4.2.3.2 Cas de quatre-voisinage

Dans le cas de quatre-voisinage, nous utilisons

- $N = \{(-1, 0), (0, -1), (0, 0), (0, 1), (1, 0)\}$,
- $A = (a_{(-1,0)}, a_{(0,-1)}, a_{(0,0)}, a_{(0,1)}, a_{(1,0)})$
- et $\mathbb{I}_2(X) = (I_2(x-1, y), I_2(x, y-1), I_2(x, y), I_2(x, y+1), I_2(x+1, y))$,

comme montrés dans la figure 4.1.

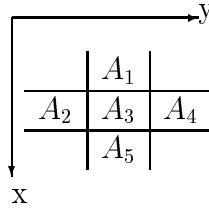


FIG. 4.1 – Vecteur A dans le cas de quatre voisinage.

Le décalage sous-pixellique que nous obtenons est

$$\begin{aligned}
 (dx, dy) &= \frac{1}{S}(a_{(-1,0)}(-1, 0) + a_{(0,-1)}(0, -1) + a_{(0,0)}(0, 0) + a_{(0,1)}(0, 1) + a_{(1,0)}(1, 0)) \\
 &= \left(\frac{a_{(1,0)} - a_{(-1,0)}}{S}, \frac{a_{(0,1)} - a_{(0,-1)}}{S} \right) \\
 &= \frac{1}{S}((L_5 - L_1)\mathbb{I}_1, (L_4 - L_2)\mathbb{I}_1)
 \end{aligned}$$

où $S = \sum_{K \in N} a_K$.

Négligeons le changement de S , nous estimons la variance de dX comme :

$$var(dx) = var\left(\frac{1}{S}(L_5 - L_1)\mathbb{I}_1\right) \approx \frac{1}{S^2}(L_5 - L_1)var(\mathbb{I}_1)(L_5 - L_1)^T.$$

Supposons que le bruit d'image est gaussien centré et indépendant de distribution identique d'écart-type σ , nous avons $var(\mathbb{I}_1) = \sigma^2 I_{m \times m}$, où $I_{m \times m}$ est la matrice identique de dimension $m \times m$ et σ est estimé de l'équation (4.3). Enfin nous obtenons $var(dx) = \frac{\sigma^2}{S^2}(L_5 - L_1)(L_5 - L_1)^T$.

De façon similaire, nous avons $var(dy) = \frac{\sigma^2}{S^2}(L_4 - L_2)(L_4 - L_2)^T$ et $cov(dx, dy) = \frac{\sigma^2}{S^2}(L_4 - L_2)(L_5 - L_1)^T$.

4.2.3.3 Cas de huit voisinage

Dans le cas de huit voisinage, nous utilisons

$$- N = \{(-1, -1), (-1, 0), (-1, 1), (0, -1), (0, 0), (0, 1), (1, -1), (1, 0), (1, 1)\},$$

$$- A = (a_{(-1,-1)}, a_{(-1,0)}, a_{(-1,1)}, a_{(0,-1)}, a_{(0,0)}, a_{(0,1)}, a_{(1,-1)}, a_{(1,0)}, a_{(1,1)})$$

- et

$$\begin{aligned} \mathbb{I}_2(X) = & (I_2(x-1, y-1), I_2(x-1, y), I_2(x-1, y+1), \\ & I_2(x, y-1), I_2(x, y), I_2(x, y+1), \\ & I_2(x+1, y-1), I_2(x+1, y), I_2(x+1, y+1)), \end{aligned}$$

comme montrés dans la figure 4.2.

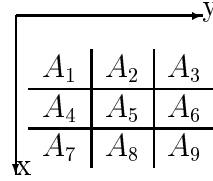


FIG. 4.2 – Vecteur A dans le cas de huit voisinage.

Le décalage sous-pixelique que nous obtenons est :

$$\begin{aligned} (dx, dy) &= \frac{\sum_{i,j=-1}^1 a_{(i,j)}(i, j)}{S} \\ &= \left(\frac{\sum_{j=-1}^1 a_{(1,j)} - a_{(-1,j)}}{S}, \frac{\sum_{i=-1}^1 a_{(j,1)} - a_{(j,-1)}}{S} \right) \\ &= \frac{1}{S} ((L_7 + L_8 + L_9 - L_1 - L_2 - L_3)\mathbb{I}_1, (L_3 + L_6 + L_9 - L_1 - L_4 - L_7)\mathbb{I}_1) \end{aligned}$$

De même comme dans la section 4.2.3.2, nous estimons la variance de décalage comme :

$$var(dx) = \frac{\sigma^2}{S^2} (L_7 + L_8 + L_9 - L_1 - L_2 - L_3)(L_7 + L_8 + L_9 - L_1 - L_2 - L_3)^T,$$

$$var(dy) = \frac{\sigma^2}{S^2} (L_3 + L_6 + L_9 - L_1 - L_4 - L_7)(L_3 + L_6 + L_9 - L_1 - L_4 - L_7)^T,$$

et

$$cov(dx, dy) = \frac{\sigma^2}{S^2} (L_7 + L_8 + L_9 - L_1 - L_2 - L_3)(L_3 + L_6 + L_9 - L_1 - L_4 - L_7)^T.$$

4.2.3.4 Explication de l'estimation de la matrice de covariance

Donnons des explications des estimations mentionnées ci-dessus. Reprenons les notations à la page 102. $\mathbb{I}_2$ est une matrice contenant des informations des textures de I_2 , appelée la matrice de *texture* de I_2 , et L est son pseudo inverse. Nous trouvons que $\text{var}(dX)$, $\text{var}(dY)$ et $\text{cov}(dX, dY)$ dépendent de l'écart-type du bruit σ et la texture de I_2 . Plus précisément, plus σ est petit, plus I_2 est texturée, plus $\text{var}(dX)$ et $\text{var}(dY)$ sont petits, qui est concorde à l'intuition.

La matrice $\mathbb{I}_2$ est une mesure de degré de texture de la fenêtre $I_2(X)_{X \in W}$, elle doit être étudiée plus en détail.

4.2.3.5 Résumé

Dans les sections précédentes, nous avons discuté l'estimation d'incertitudes de décalages locaux dans le cas de 4-voisinage et 8-voisinage pour l'algorithme rapide. Nous allons discuter les liens entre notre modèle et d'autres modèles comme celui de Förstner [For82], celui d'Ackermann [Ack84] et celui de Gruen [Gru85]. Nous allons introduire une étape de raffinement pour cet algorithme, où l'estimation peut être naturellement étendue comme elle est dans l'algorithme robuste présenté dans la section 4.3. L'algorithme rapide avec cette étape sera appelé *4nei2* et *8nei2* dans le cas du quatre-voisinage et huit-voisinage, respectivement, et l'algorithme sans cette étape sera appelé *4nei* et *8nei* dans ces deux cas respectivement.

4.2.4 Liens avec d'autres modèles

Nous discutons les liens entre notre modèle et autres modèles. Comme une analyse théorique, nous nous limitons au cas unidimensionnel, le cas bidimensionnel peut être traité de manière similaire.

Reprenons le modèle unidimensionnel exprimé par l'équation (4.1). Si nous supposons qu'il n'y a pas d'effet d'échelle et de translation, nous obtenons :

$$f_1 = f_2 * (a_1 \delta_{-1} + (1 - a_1 - a_{-1}) \delta_0 + a_{-1} \delta_1) + \epsilon,$$

c'est-à-dire :

$$f_1 = a_1 f_2 * (\delta_{-1} - \delta_0) + a_{-1} f_2 * (\delta_1 - \delta_0) + f_2 + \epsilon.$$

Si nous prenons $f_2(x+1) - f_2(x) = (f_2 * (\delta_{-1} - \delta_0))(x)$ et $f_2(x) - f_2(x-1) = (f_2 * (\delta_0 - \delta_1))(x)$ comme deux approximations de $F'_2(x)$, nous obtenons $f_1(x) \approx$

$f_2(x) + (a_1 - a_{-1})F'_2(x)$. C'est le modèle de Förstner expliqué dans [For82]. Donc, notre modèle peut être traité comme une généralisation de celui de Förstner.

Il y a aussi d'autres modèles plus généraux comme celui d'Ackermann [Ack84] et celui de Gruen [Gru85]. Ces deux modèles considèrent la déformation des fenêtres entre les deux images, comme Brand [Bra95].

Plus précisément, Ackermann considère que les niveaux de gris de deux fenêtres d'images suivent une relation affine, et Gruen considère qu'elles suivent simplement une relation translationnelle :

- Ackermann : $f_1(x) = r + hf_2(ax + b)$,
- Gruen : $f_1(x) = r + f_2(ax + b)$.

Les deux méthodes fonctionnent seulement après beaucoup d'itérations. Elles ne donnent pas un résultat précis après la première itération. En effet, pour implémenter les modèles, la fonction $f_2(ax + b)$ est approximée par $f_1(x) + (a - 1)xf'_2(x) + bf'_2(x)$. Cette approximation est bonne seulement quand b et $(a - 1)x$ sont tous petits. Si nous avons un appariement initial assez bon, b est normalement petit, mais $(a - 1)x$ n'est pas sûr d'être petit.

Cette analyse montre que la déformation des fenêtres ne peut pas être très bien approximée seulement selon la relation des niveaux de gris.

Notre modèle, comme une généralisation de celui de Förstner peut être vu comme un modèle direct «maximal», dans le sens qu'il est le modèle le plus général qui peut fournir un résultat direct sans itération. Dans ce modèle, la transformation des fenêtres est négligée, il n'est donc applicable que pour le cas translationnel. Pour le cas des transformations affines, nous proposons la méthode robuste présentée dans la section 4.3.

4.2.5 Comment raffiner la première estimation?

Le décalage peut être raffiné après sa première estimation. Cette réestimation est accomplie en imposant des coefficients à être zéro, selon la première estimation. Comme dans le cas unidimensionnel, $\delta_{-\epsilon}$ peut-être approximée par $(1 - \epsilon)\delta_0 + \epsilon\delta_{-1}$, au lieu d'être approximée par $b_1\delta_{-1} + (1 - 2b_1 - \epsilon)\delta_0 + (b_1 + \epsilon)\delta_{-1}$.

Dans le cas bidimensionnel, par exemple, si la première estimation de la disparité sous-pixelle est placée dans le premier quadrant. Notons $I = \{(x, y) | x, y \geq 0\}$ et $I' = \{(x, y) | (x, y) \leq 0\}$, alors nous pouvons imposer que $a_K = 0$ pour tous les $K \notin I'$ dans l'équation (4.4) et $a_K = 0$ pour tous les $K \notin I$ dans l'équa-

tion (4.2), puis résolvons le système d'équations linéaires simplifié, avec plus de redondances et stabilités.

Les autres cas peuvent être traités similairement. Nous appelons cette méthode raffinée *4nei2* et *8nei2*, respectivement. L'estimation d'incertitude peut-être obtenue comme dans la première étape.

Les résultats expérimentaux montrent que *8nei2* est un peu meilleur que *8nei* mais *4nei2* n'est pas meilleur que *4nei*, voir les tableaux C.1, C.2, C.3, C.4, C.5, C.6 pour la comparaison.

En absence d'explication de ce phénomène, nous pensons que ces améliorations légères ne sont pas significatives et qu'elles ne méritent pas d'être retenues.

4.2.6 Quelques problèmes d'implémentation

Les processus principaux dans cet algorithme sont le calcul du pseudo inverse d'une matrice, et l'estimation des niveaux de gris des pixels de coordonnées flottantes. Leurs implémentations sont discutées ici.

Le calcul du pseudo inverse d'une matrice A peut se faire par la décomposition LU ou Cholesky de $A^T A$, ou par la décomposition QR ou SVD de A . La décomposition LU de $A^T A$ n'est pas stable, la décomposition QR de A est moins stable que la décomposition SVD , la décomposition Cholesky de $A^T A$ donne le même résultat que la décomposition SVD de A , mais plus rapide. En conclusion, la technique par la décomposition de Cholesky de $A^T A$ est utilisée.

Pour estimer les intensités lumineuses d'un pixel de coordonnées flottantes, nous utilisons l'interpolation bilinéaire car i) elle est simple ; ii) elle est assez précise ; iii) elle concorde avec notre hypothèse de régularité locale.

4.2.7 Résultats expérimentaux

Nous avons testé cet algorithme sur une paire d'images. Nous utilisons une scène planaire car seulement dans ce cas, une homographie entre les deux images peut être établie. Cette homographie est estimée à partir des cibles appariés [Bra95]. La contrainte épipolaire et la contrainte de région d'intérêts sont aussi utilisées dans l'initialisation pour accélérer le processus et diminuer l'ambiguïté [AG92]. La validation est effectuée sur les points d'intérêt et aussi sur des points généraux aléatoirement choisis.

Dans chaque expérimentation, l'initialisation est obtenue par la méthode *ZNCC* [AG92]. Les deux raffinements sont faits par la méthode décrite dans la sec-

tion 4.2.2 et la section 4.2.3 en utilisant l'hypothèse de quatre-voisinage et huit-voisinage. Les résultats sans (*4nei* et *8nei*) et avec (*4nei2* et *8nei2*) le raffinement sont montrés.

L'erreur de l'appariement d'un pixel est définie comme la distance euclidienne entre le pixel trouvé et celui calculé par l'homographie précise. Le terme o_p représente le pourcentage de faux appariements, où un faux appariement signifie un appariement ayant une erreur supérieure à un pixel. Pour les bons appariements, nous montrons la moyenne des erreurs (m_1) et la racine carrée de la moyenne des erreurs au carré (m_2). Plus formellement, $m_1 = \frac{\sum_{i=1}^n e_i 1_{[0,1]}(e_i)}{\sum_{i=1}^n 1_{[0,1]}(e_i)}$, $m_2 = \sqrt{\frac{\sum_{i=1}^n e_i^2 1_{[0,1]}(e_i)}{\sum_{i=1}^n 1_{[0,1]}(e_i)}}$, où e_i est l'erreur de $i^{\text{ième}}$ point choisi dans le test et 1 est la fonction d'indicatrice¹. Nous avons choisi des fenêtres de dimension de 7×7 à 19×19 .

Le déplacement entre les deux images images est essentiellement une translation, avec des effets de rotations et d'échelles négligeables. Cette paire d'images est appelée *bolino* et est affichée dans la figure 4.3. Les points sont affichés dans la figure 4.4, représentés par des croix.

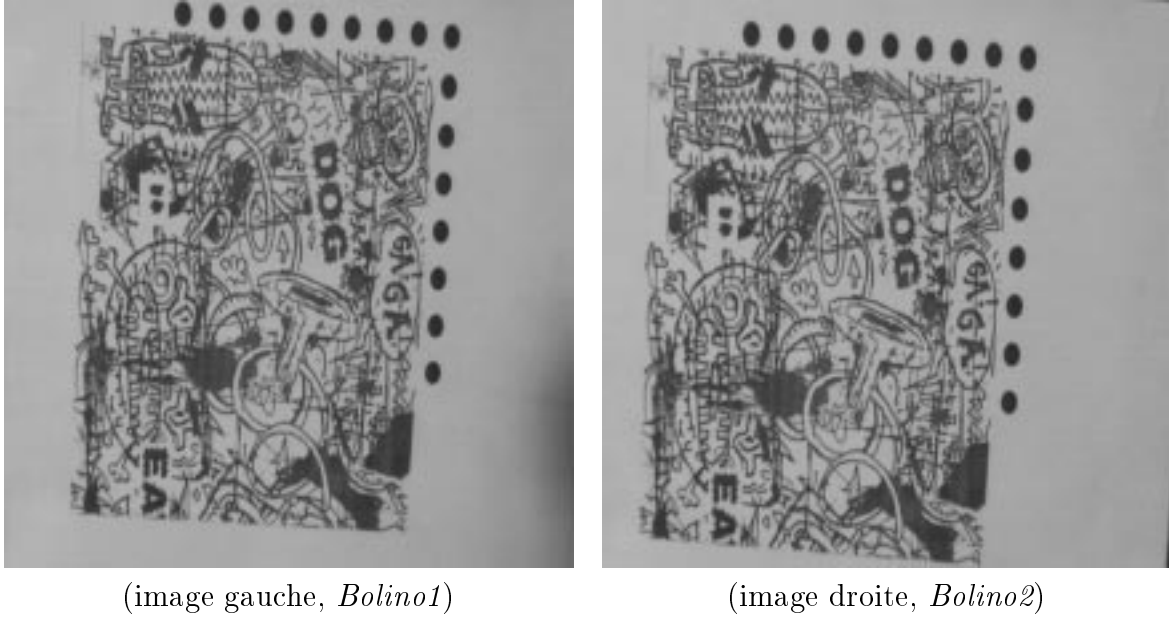


FIG. 4.3 – La paire d'images *Bolino1* et *Bolino2*.

Les résultats sont affichés dans les tableaux C.1 et C.2. Le tableau C.1 résume

$$1. 1_A(x) = \begin{cases} 0 & x \notin A \\ 1 & x \in A \end{cases}.$$

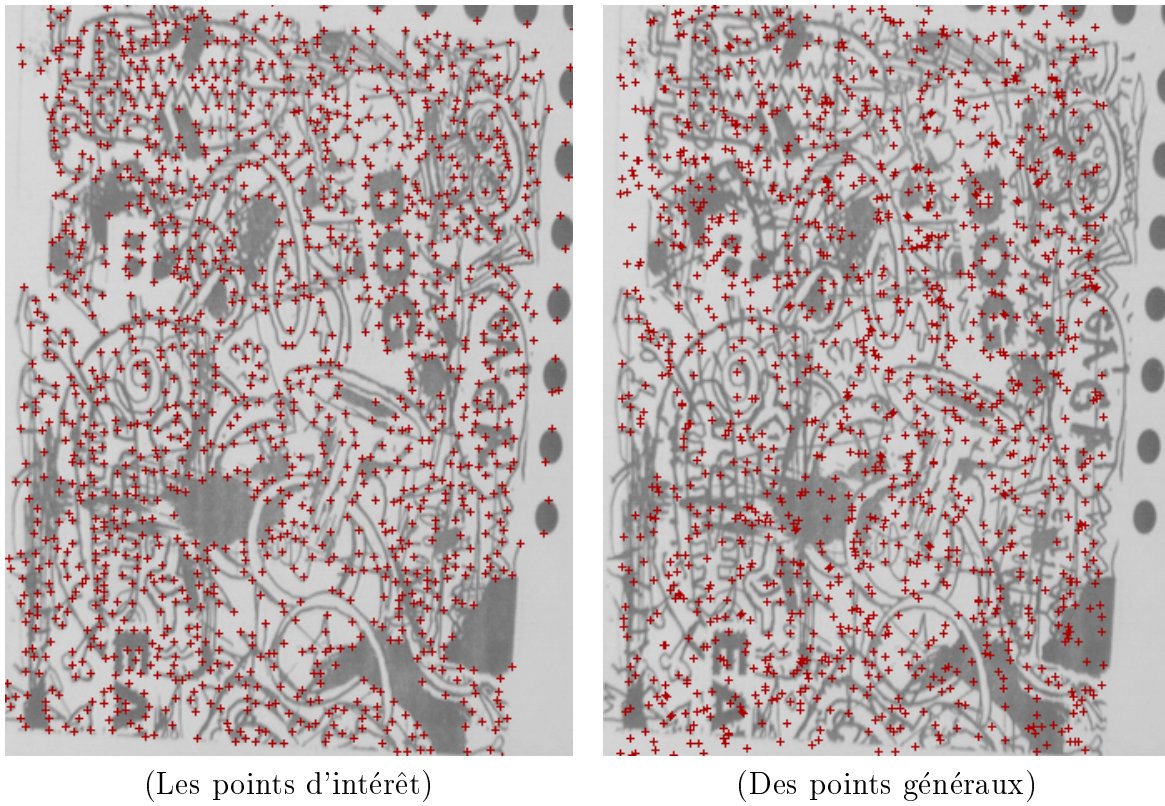


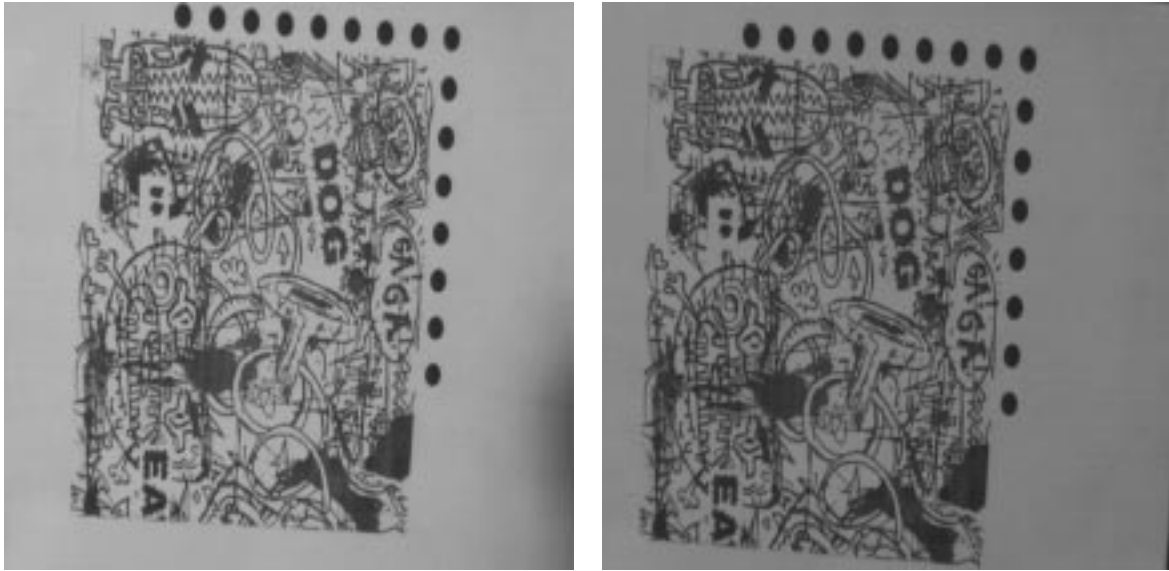
FIG. 4.4 – *Les points d'intérêt et des points généraux choisis dans l'image gauche Bolino de la figure 4.3.*

les résultats pour les points d'intérêts ; et le tableau C.2 résume ceux pour des points généraux.

Nous voyons qu'avec notre méthode, nous pouvons obtenir des résultats expérimentaux d'une précision d'ordre 0,07 pixel pour les points d'intérêt et d'ordre 0,08 pixel pour les points généraux. Les faux appariements sont exceptionnels.

Ces résultats pourront être encore améliorés si nous appliquons le post-traitement présenté dans la section 4.4. Et une précision d'ordre 0.05 sera obtenue.

Des tests sont aussi faits sur la paire d'images *Bolino1* (l'image à gauche dans la figure 4.3) et *bolino3* (figure 4.5), où le changement des niveaux de gris est plus grand.



(image gauche, *Bolino1*)

(image droite, *Bolino3*)

FIG. 4.5 – Image *Bolino1* et *Bolino3*

Bolino3 est pris au même point de vue de *Bolino2* avec seulement un changement d'ouverture d'iris qui fait l'image plus sombre. Tests sont faits sur les points d'intérêt et des points généraux montrés dans la figure 4.4. Les résultats sont montrés dans les tableaux C.3 et C.4 respectivement.

Les derniers résultats sont faits pour la paire d'image *Bolino1* (image gauche dans la figure 4.3) et *Bolino4* (figure 4.6) , avec un changement de niveaux de gris encore plus grand. Les résultats pour les points affichés dans la figure 4.4 sont montrés dans les tableaux C.5 et C.6 respectivement.

Nous voyons que les résultats sont peu influencés par le changement des ni-

(image gauche, *Bolino1*)(image droite, *Bolino4*)FIG. 4.6 – Image *Bolino1* et *Bolino4*

veaux de gris, sauf pour le pourcentage des faux appariements qui vient de mauvaises initialisations.

La complexité du temps Pour une fenêtre de dimensions 11×11 , la vitesse moyenne par point est à peu près de $0,003s$, $0,004s$, $0,005s$ and $0,007s$, respectivement pour *4nei*, *4nei2*, *8nei* and *8nei2* sur Sun SPARC.

Malheureusement, cet algorithme n'est pratiquement applicable que pour les paires d'images dont la transformation des fenêtres est quasiment une translation pure. Dans le cas contraire, la précision que nous obtenons dépasse $\frac{1}{4}$ pixel, ce qui n'est pas assez précis. Donc nous avons développé un algorithme robuste, qui est aussi plus général est précis.

4.3 Le cas général : algorithme robuste

Dans la section précédente, nous avons développé un algorithme de corrélation précise applicable au cas où la déformation des fenêtres est négligeable. Néanmoins, dans les cas réels, il y a normalement une déformation des fenêtres non négligeable, comme a indiqué par Brand [Bra95], Ackermann [Ack84] et Gruen [Gru85].

Supposons que le monde est localement planaire et que la surface est à peu près parallèle aux plans images, si la fenêtre de corrélation est assez petite, la projection parallèle est une bonne approximation, et sous ces conditions, la correspondance des fenêtres dans deux images est approximativement affine.

Les anciennes méthodes dans [Bra95, Ack84, Ack84] sont itératives et très coûteuses, qui dépendent de l'initialisation, et leur convergence n'est pas garantie théoriquement.

Dans cette section, nous développons un algorithme d'appariement robuste et précis, à partir d'une carte de disparité dense. La procédure est : étant donnée une carte de correspondance dense de points (P_1, P_2) de deux images I_1 et I_2 , nous cherchons A et t , tels que $P_2 \approx AP_1 + t$, (A, t) représente la transformation des fenêtres, puis nous sommes amenés à estimer le décalage entre $I_1(P_1)$ et $I_2(AP_1 + t)$. Cet algorithme, qui considère aussi la déformation des fenêtres d'images, est appelé *l'algorithme robuste*.

Dans cette section, d'abord nous estimons la déformation affine locale entre deux fenêtres d'images à partir d'une carte de correspondance initiale² (section 4.3.1), puis nous estimons les décalages locaux (section 4.3.2) et les fusionnons selon leurs incertitudes (section 4.3.3).

4.3.1 Obtenons la transformation affine entre les points bidimensionnels correspondants

Supposons que nous avons une correspondance initiale dense entre pixels de deux images d'une même scène. Sous l'hypothèse que la correspondance est localement affine [Bra95], nous estimons cette transformation et raffinons la correspondance initiale.

Soient les deux ensembles de points bidimensionnels P_i et P'_i ($i = 1 \dots n$), où $P_i = (x_i, y_i)^t$ et $P'_i = (x'_i, y'_i)^t$. Nous cherchons $A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$ et $t = \begin{pmatrix} t_x \\ t_y \end{pmatrix}$, tels que : $P_i \xrightarrow{A, t} P'_i$.

C'est-à-dire,

$$\begin{cases} a_{11}x_i + a_{12}y_i + t_x &= x'_i \\ a_{21}x_i + a_{22}y_i + t_y &= y'_i \end{cases}$$

ou

2. Cette carte de correspondance peut être obtenue par des méthodes globales d'appariement, comme la programmation dynamique (section 2.2).

$$\begin{pmatrix} x_i & y_i & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 0 & 1 \end{pmatrix} \begin{pmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \\ t_x \\ t_y \end{pmatrix} = \begin{pmatrix} x'_i \\ y'_i \end{pmatrix}$$

pour $i = 1 \dots n$.

Les méthodes employées doivent être robustes à cause des faux appariements éventuels. Nous introduisons deux méthodes : LMedS (page 24) et RanSaC (page 19). Ces deux méthodes trouvent une transformation affine initiale de manière robuste, détectent les faux appariements en comparant avec cette transformation et emploient une minimisation de moindres carrés sur les bonnes données pour obtenir la transformation affine finale.

4.3.1.1 LMedS

La méthode LMedS est réalisée par échantillonnage aléatoire [RL87]. Choisissons trois P_i parmi tous les points : $P_{i_1}, P_{i_2}, P_{i_3}$, nous estimons A et t en résolvant :

$$\begin{pmatrix} x_{i_1} & y_{i_1} & 0 & 0 & 1 & 0 \\ 0 & 0 & x_{i_1} & y_{i_1} & 0 & 1 \\ x_{i_2} & y_{i_2} & 0 & 0 & 1 & 0 \\ 0 & 0 & x_{i_2} & y_{i_2} & 0 & 1 \\ x_{i_3} & y_{i_3} & 0 & 0 & 1 & 0 \\ 0 & 0 & x_{i_3} & y_{i_3} & 0 & 1 \end{pmatrix} \begin{pmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \\ t_x \\ t_y \end{pmatrix} = \begin{pmatrix} x'_{i_1} \\ y'_{i_1} \\ x'_{i_2} \\ y'_{i_2} \\ x'_{i_3} \\ y'_{i_3} \end{pmatrix}.$$

Après avoir obtenu les $a_{k,l}$, $k, l = 1, 2$, et t_x, t_y , l'erreur pour le couple (P_j, P'_j) est définie comme $e_j = \sqrt{(x'_j - a_{11}x_j - a_{12}y_j - t_x)^2 + (y'_j - a_{21}x_j - a_{22}y_j - t_y)^2}$, et nous choisissons les $a_{k,l}$ et t qui minimisent la médiane des e_j^2 entre suffisamment d'essais. Notons la médiane la plus petite $m : m = \min_j \text{med } e_j^2$.

Le couple (P_j, P'_j) est considéré comme un faux appariement si et seulement si $e_j > 1.4826 \times 2.5(1 + \frac{5}{n-p})\sqrt{m}$ (page 25 et 71).

Nous sommes donc amenés au problème de la minimisation de

$$\sum_{i=1}^n W_i ||P'_i - AP_i - t||^2$$

avec $W_i \in \{0, 1\}$, $W_i = 1$ si et seulement si P_i et P'_i s'apparient.

C'est facile d'obtenir $t = \frac{\sum_{i=1}^n W_i(P'_i - AP_i)}{\sum_{i=1}^n W_i}$. Notons $\bar{P}' = \frac{\sum_{i=1}^n W_i P'_i}{\sum_{i=1}^n W_i}$ et $\bar{P} = \frac{\sum_{i=1}^n W_i P_i}{\sum_{i=1}^n W_i}$, les moyennes pondérées de P et P' . Nous avons alors $t = \bar{P}' - A\bar{P}$. Nous avons donc à minimiser $\sum_{i=1}^n W_i \|P'_i - \bar{P}' - AP_i + A\bar{P}\|^2$.

Notons $Q_i = P_i - \bar{P}$, $Q'_i = P'_i - \bar{P}'$, nous sommes donc amenés à minimiser $\sum_{i=1}^n W_i \|Q'_i - AQ_i\|^2$, qui peut être réalisé en utilisant l'estimation des moindres carrés.

Après avoir obtenu les $a_{k,l}$ ($k, l = 1, 2$), nous pouvons calculer t comme $t = \frac{\sum_{i=1}^n W_i(P'_i - AP_i)}{\sum_{i=1}^n W_i}$.

Il nous reste à déterminer la taille de l'échantillonnage aléatoire. Notons ϵ le taux de données correctes, p le nombre de paramètres à estimer et m est le nombre d'échantillonnages. la probabilité que tous les m p -uplets différents choisis contiennent au moins une erreur grossière est $p = (1 - \epsilon^p)^m$. si q est la probabilité d'erreur, alors $p < q$ donne la relation pour le nombre d'échantillonnages : $(1 - \epsilon^p)^m < q$, c'est-à-dire $m > \frac{\log q}{\log(1 - \epsilon^p)}$. pour $q = 0.01$, $p = 3$ et $\epsilon > 0,5$, si on a $m > \frac{\log 0.01}{\log(1 - 0.5^3)}$, alors $m > \frac{\log q}{\log(1 - \epsilon^p)}$ pour tous les $\epsilon > 0,5$. dans notre expérience, nous choisissons $m = 35$.

4.3.1.2 RanSaC

Bien que la méthode LMedS soit très robuste pour l'estimation de la déformation affine, elle est assez coûteuse en temps de calcul. Une méthode similaire est celle de RanSaC. Cette méthode a été présentée dans la section 1.2.2.

Le nombre nécessaire d'échantillonnages pour avoir une probabilité d'erreur assez petite est choisie d'être 35, comme la méthode de LMedS, bien que l'espérance du nombre d'échantillonnages puisse aussi être utilisée [FB81]. La tolérance d'erreur est choisie d'être 1,0, en considérant que l'initialisation doit être de précision pixellique. Si on note t la dimension minimale de l'ensemble de consensus acceptable, y la probabilité qu'une donnée est dans la tolérance d'erreur d'un modèle incorrect, nous espérons que la probabilité d'erreur y^{t-p} soit petite. Supposons $y < 0.5$, alors $t - p \geq 7$ implique que $y^{t-p} < 1\%$. Dans notre expérience, nous avons choisi pour $p + 7 = 10$ pour t .

Après avoir trouvé l'ensemble de bons points, le processus de raffinement est réalisé de la même manière que LMedS.

Si on fait toujours le nombre maximal d'échantillonnages (sauf si l'on trouve

un bon modèle pour tous les points), on obtient alors une technique de RanSaC modifiée, qui est équivalente à choisir le cardinal de l'ensemble de données comme la valeur de t . Elle est plus robuste que la méthode avec $t = 10$, mais plus lente. Cette méthode est appelée «RanSaC2».

4.3.2 Calculons le décalage sous pixelique

Une fois que nous avons trouvé la transformation affine (A, t) entre deux fenêtres d'images, nous pouvons projeter les points P_i vers P'_i , c'est-à-dire, remplacer P' par $AP + t$, appelé l'appariement corrigé. Via cette étape de correction, les appariements qui ne sont pas cohérents à son voisinage (au sens d'affinité locale) sont corrigés (si la plupart de points dans son voisinage ont de bons appariements), qui rend cet algorithme *robuste* aux faux appariements initiaux.

Considérons deux ensembles de points bidimensionnels correspondants, $P_1(i, j)$ et $P_2(i, j)$, avec une relation affine entre eux : $P_2(i, j) = AP_1(i, j) + t$. Soient I_1 et I_2 deux images, et $M_1(i, j) = I_1(P_1(i, j))$, $M_2 = I_2(P_2(i, j))$.

Soient s_{len} et t_{len} les dimensions deux fenêtres à apparier (page 100), c'est à dire pour $s - s_{len} < i < s + s_{len}$, $t - t_{len} < j < t + t_{len}$, nous avons la relation $M_1(i, j) = \sum_{(k,l) \in N} a_{k,l} M_2(i + k, j + l) + C + \epsilon$, alors nous avons :

$$\begin{aligned} I_1(P_1(i, j)) &= M_1(i, j) = \sum_{(k,l) \in N} a_{k,l} I_2(P_2(i + k, j + l)) + O + \epsilon \\ &= S \sum_{(k,l) \in N} b_{k,l} I_2(P_2(i + k, j + l)) + O + \epsilon \\ &\approx S I_2(\sum_{(k,l) \in N} b_{(k,l)} P_2(i + k, j + l)) + O + \epsilon, \end{aligned}$$

où N est un voisinage d'origine, quatre-voisinage ou huit, et $S = \sum_{(k,l) \in N} a_{k,l}$.

En reprenant la même formulation que dans la section 4.2.2, nous estimons le décalage sous-pixelique comme

$$\sum_{(k,l) \in N} b_{(k,l)} (P_2(i + k, j + l) - P_2(i, j)).$$

Il ne dépend pas de (i, j) si P_1 est une fonction affine de i, j (normalement $P_1(i, j) = (i, j)$ est affine).

Nous pouvons aussi supposer que $M_2(i, j) = \sum_{(k,l) \in N} a'_{k,l} M_1(i + k, j + l) + O' + \epsilon'$, et nous obtenons $M_2(i, j) \approx S' I_1(\sum_{(k,l) \in N} b'_{k,l} P_1(i + k, j + l) + O' + \epsilon')$.

Le décalage sous-pixelique que nous obtenons est :

$$\sum_{(k,l) \in N} b'_{k,l}(-P_1(i+k, j+l) + P_1(i, j)).$$

Pour fusionner ces deux estimations, nous avons besoin d'estimer leurs incertitudes, qui est expliqué dans la section suivante.

4.3.3 Estimation de l'incertitude de décalage sous-pixelique

Dans cette section, A prend le sens dans section 4.2.3.1, les $P_1[i]$ sont les points autour du centre de la première fenêtre, ordonnés comme montrés dans figure 4.1 et 4.2. $P_2[i]$ est l'appariement corrigé de $P_1[i]$ par la transformation affine locale estimée.

Considérons deux cas : le quatre voisinage et le huit voisinage respectivement.

Dans le cas du quatre voisinage, le décalage est estimé comme $\frac{\sum_{i=1}^5 A[i]P_2[i]}{S} = P_2[i] + \sum_{i \neq 3} \frac{A[i](P_2[i] - P_2[3])}{S}$.

Sachant $\begin{pmatrix} A \\ O \end{pmatrix} = T = \mathbb{I}_2^+ \mathbb{I}_1 = L \mathbb{I}_1$, nous avons $A[i] = L_i \mathbb{I}_1$, alors la covariance entre $A[i]$ et $A[j]$ est $L_i L_j^T \sigma^2$, où $\text{var}(M) = \sigma^2 I$.

Notons $P'_2[i] = P_2[i] - P_2[3]$, alors l'incertitude est estimée comme :

$$\frac{\sigma^2}{S^2} \sum_{i,j \neq 3} P'_2[i] P'_2[j] L_i L_j^T.$$

Dans le cas du huit voisinage, le décalage est estimé comme :

$$\frac{\sum_{i=1}^9 A[i]P_2[i]}{S} = P_2[5] + \sum_{i \neq 5} \frac{A[i](P_2[i] - P_2[5])}{S}.$$

Notons $P'_2[i] = P_2[i] - P_2[5]$, alors l'incertitude est estimée comme :

$$\frac{\sigma^2}{S^2} \sum_{i,j \neq 5} P'_2[i] P'_2[j]^T L_i L_j^T.$$

4.3.4 Résultats expérimentaux

Nous avons fait des tests sur une paire d'images d'une scène qui s'appelle *New* (figure 4.7). Il y a un effet d'échelle géométrique proche de $1 : 0,9$ entre les deux images de cette paire. Nous présentons les résultats en utilisant LMedS (section 4.3.1), les résultats avec RanSaC se trouvent dans l'appendice C.2.2.

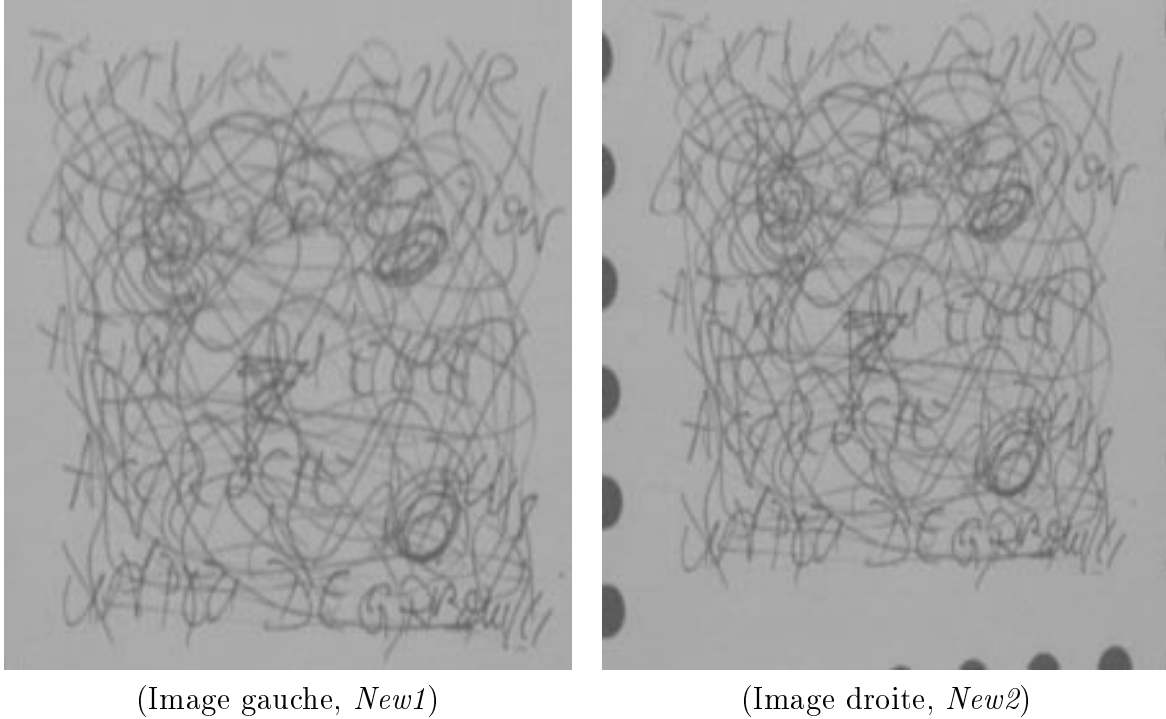
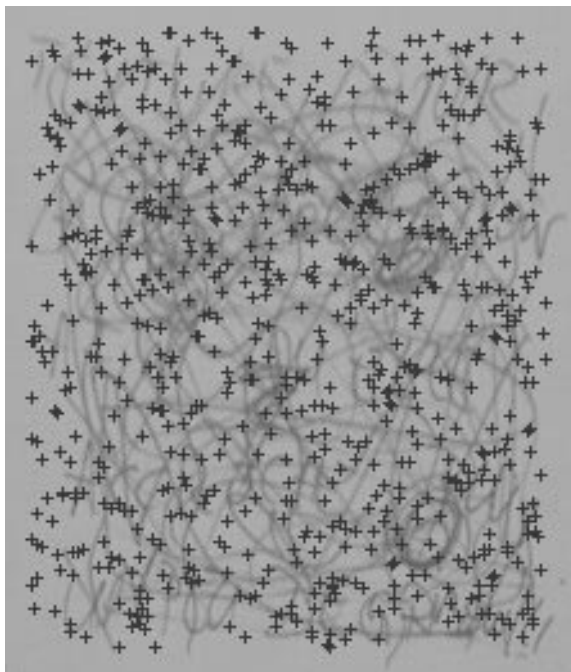


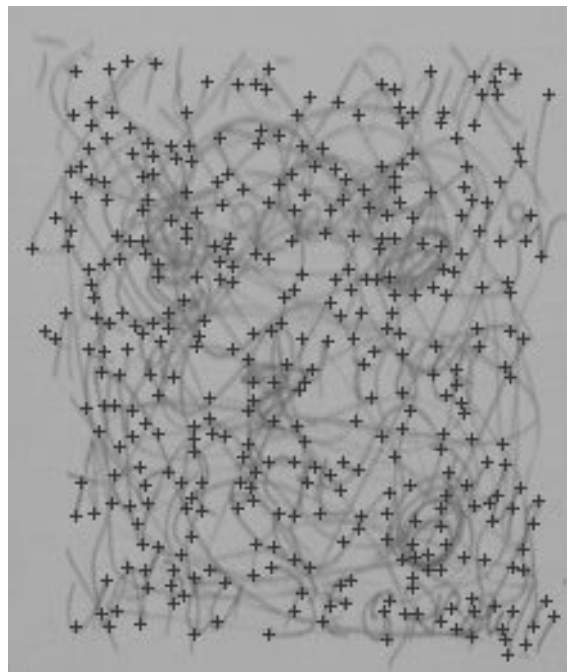
FIG. 4.7 – La paire d'image sous un effet d'échelle $1 : 0,9$.

Les points choisis sont affichés dans la figure 4.8. Voir le tableau C.7 pour les résultats obtenus pour les points d'intérêt ; et le tableau C.8 pour ceux des points généraux. Nous voyons que pour une fenêtre suffisamment grande, les faux appariements sont éliminés et que la précision est très bonne.

Maintenant regardons une paire d'images sous une rotation proche de 20 degrés, à part d'un effet d'échelle proche de $1 : 0,9$ (figure 4.9). Pour les points affichés dans la figure 4.8, le tableau affiche C.9 les résultats obtenus pour les points d'intérêt et le tableau C.10 donne ceux des points généraux. Même dans ce cas, il n'y a pas de faux appariements pour les points d'intérêt et très peu pour ceux généraux, car la plupart d'eux sont éliminés à l'aide de l'estimation robuste de la transformation des fenêtres. La précision sur les points d'intérêt est très bonne, et est bonne même pour les points généraux.



(Les points d'intérêt)



(Des points généraux aléatoirement choisis)

FIG. 4.8 – *Les points d'intérêt et des points généraux choisis dans l'image gauche de la figure 4.7.*

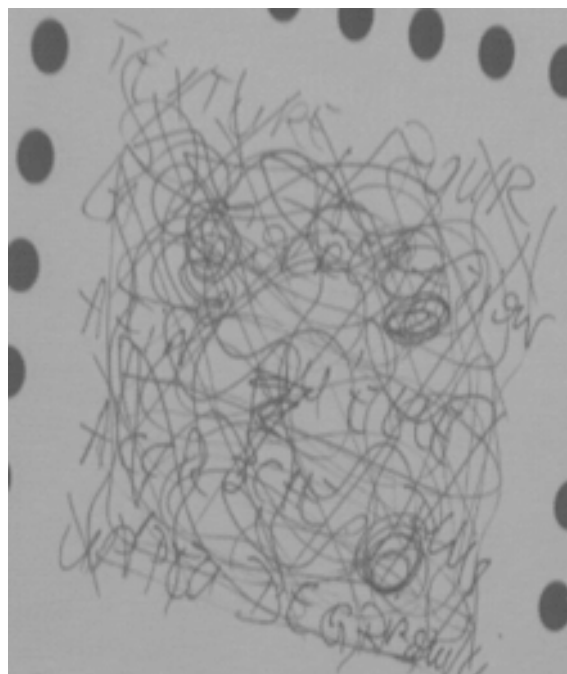
(Image gauche, *New1*)(Image droite, *New3*)

FIG. 4.9 – Paire d'image *New* sous une rotation proche de 20 degrés, à part d'un effet d'échelle de 1 : 0,9.

Donnons les résultats sur une paire d'images (figure 4.10) où il y a une grande rotation (proche de 30 degrés) entre elles, à part d'un effet d'échelle proche de $1 : 0,9$. Pour les points affichés dans la figure 4.8, le tableau C.11 affiche les résultats obtenus pour les points d'intérêt et le tableau C.12 donne ceux des points généraux. Dans ce cas assez général, il y a encore peu de faux appariements.

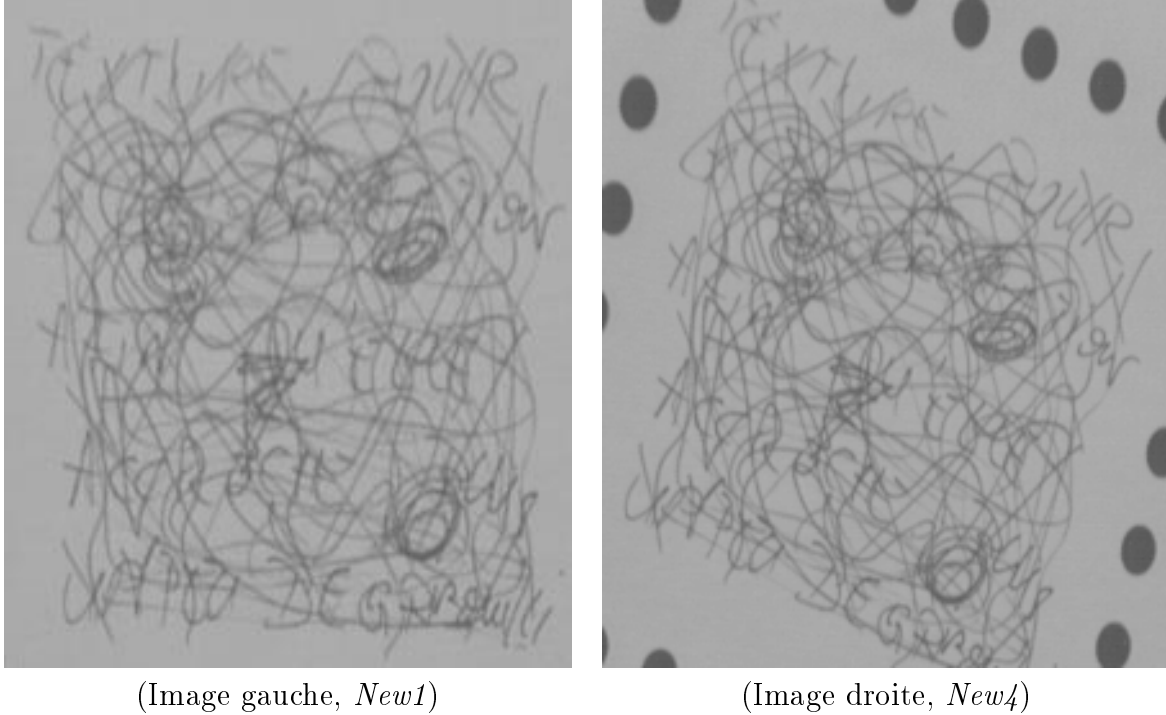


FIG. 4.10 – Paire d'image *New* sous une rotation de 30 degrés, à part d'un effet d'échelle de $1 : 0,9$.

Comme nous voyons, cette méthode est robuste face à la déformation des fenêtres, aux faux appariements, et la précision que nous obtenons est très bonne, alors que la corrélation classique est peu précise et contient de nombreuses erreurs grossières.

La complexité en temps Pour une fenêtre de dimension 15×15 , la vitesse moyenne pour un point est à peu près $0.025s$, $0.035s$, $0.029s$ and $0.041s$, respectivement pour les méthodes *4nei*, *4nei2*, *8nei* and *8nei2*.

4.4 Post traitement : incorporant la géométrie épipolaire

Dans les deux sections précédentes, nous avons présenté deux algorithmes, où la géométrie épipolaire n'a pas été incorporée. Une raison est que nous n'avons pas une géométrie épipolaire précise avant d'avoir l'appariement précis. Dans cette section, nous proposons d'incorporer la géométrie épipolaire dans le problème d'appariement précis.

D'abord, nous réestimons la géométrie épipolaire à partir des appariements précis donnés par les méthodes présentées dans les deux sections précédentes, puis pour chaque pixel dans la première image, nous calculons sa ligne épipolaire selon la géométrie épipolaire réestimée et projetons son appariement (donné par les méthodes dans les deux sections précédentes) sur cette ligne pour corriger les erreurs perpendiculaires à cette ligne.

Notons que la géométrie épipolaire doit être estimée à partir des appariements précis pour obtenir une bonne précision. Nous expliquons un peu plus ces deux étapes.

4.4.1 Réestimation de la géométrie épipolaire

Dans le cas de l'observation d'une scène rigide en trois dimensions, la géométrie épipolaire peut être réestimée selon des appariements précis. Le problème d'estimation de la géométrie épipolaire à partir des correspondances de points a été étudié ces dernières années [LF93, Fau93, ZDFL94, BM95a, Har95].

Ce problème aussi modélisé comme l'estimation de la matrice fondamentale F peut être résolue linéairement avec l'algorithme de 8 points ou non linéairement. Les algorithmes linéaires sont très simples à implémenter et à utilisés, mais ils ont deux désavantages principaux : comme la distance utilisée n'est pas normalisée. Les algorithmes non linéaires règnent dans ce domaine et montrent des meilleures propriétés par rapport aux ceux non linéaires. Considérant la possibilité de la présence des faux appariements, les méthodes robustes comme les moindres carrés médians sont utilisées pour remplacer celle des moindres carrés.

Nous utilisons une méthode composée de deux étapes : d'abord une méthode robuste linéaire pour détecter les faux appariements et obtenir l'estimation initiale, puis une méthode non linéaire utilisant les distances normalisées et la contrainte $\text{rang}(F) = 2$ pour avoir le résultat final. Nous avons choisi la méthode d'Hartley [Har95] comme celle linéaire et la méthode non linéaire de Boufama [BM95a].

La géométrie épipolaire réestimée à partir des appariements précis a une médiane d'erreurs proche de 0.04 pixels, qui montre sa haute précision.

4.4.2 Projeter sur la ligne épipolaire

Après avoir obtenu la géométrie épipolaire, la ligne épipolaire peut être calculée pour chaque point dans la première image, et son appariement dans la deuxième image est projetée sur cette ligne pour corriger l'erreur perpendiculaire à la ligne épipolaire.

Les résultats expérimentaux affichés dans les tableaux de l'appendice C.3 montrent que ce post traitement très simple comme un processus de «ajustement et modification» est efficace et donne une amélioration de 0.02 pixels pour la précision.

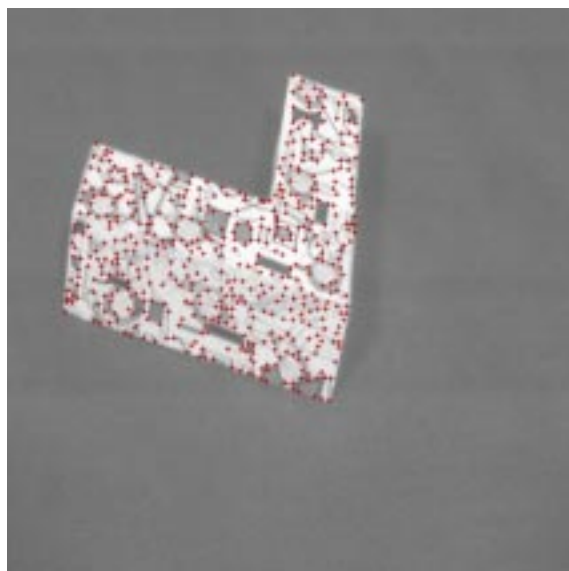
4.5 Validation via reconstruction tridimensionnelle

Cette section met en œuvre un exemple de la reconstruction tridimensionnelle de points à partir des points bidimensionnels mis en correspondance.

Étant données deux images, il est possible de reconstruire les points tridimensionnels à une transformation projective près, avec aucune information a-priori. Avec plus d'hypothèses, la reconstruction euclidienne devient possible. Elle est définie à une transformation euclidienne près, déterminée par une rotation et une translation du repère de la reconstruction. La méthode la plus couramment utilisée consiste à calibrer une tête stéréoscopique d'un robot suivi de la reconstruction de la scène dans le repère du robot [ZFD95]. D'autres méthodes sont possibles, comme celle à partir d'une reconstruction projective [BWW94]. Mais la reconstruction classique utilise un étalonnage de la caméra par la méthode de Faugeras-Toscani [FT87].

Une paire d'images a été prise dans le but d'évaluer quantitativement la précision de la reconstruction euclidienne que l'on obtient. Dans cette paire (figure 4.11), un objet est observé, qui possède une structure tridimensionnelle connue, afin de pouvoir être évalué. On appelle cette paire *L*, car l'objet dans les images a la forme d'un *L*. Une texture a été plaquée contre cet objet afin d'extraire de nombreux points d'intérêts (figure 4.12) dans les images et de faciliter la mise en correspondance entre les deux images de cette paire.

Le nombre de points de la base est 8. Leurs coordonnées euclidiennes sont

(Image gauche, L_1 .)(Image droite, L_2 .)FIG. 4.11 – *La paire d'images d'objet L .*FIG. 4.12 – *Les points d'intérêt dans L .*

connues afin de déduire la structure euclidienne de la scène. Les points sont le centre des cibles circulaires rétro-réfléchissantes posées sur l'objet. Les mesures de ces points dans l'espace euclidien ont été obtenues avec une règle. Cet objet est observé dans un volume de taille $73 \times 73 \times 23 \text{ mm}^3$. Sur cette scène, les points appartiennent à deux plans visibles parmi les trois plans d'objet. Les deux plans sont horizontal et vertical avec les équations connues.

La paire stéréo est calibrée à partir des 8 points de base. Après avoir obtenu les deux matrices de projection, les points tridimensionnels sont obtenus à partir des points bidimensionnels appariés via la méthode présentée dans [SQ95].

La méthode expérimentale proposée fournit une réponse qualitative permettant d'apprécier l'amélioration par rapport à des mises en correspondances habituelles par corrélation classique. Pour l'apprécier de façon quantitative, nous procédons selon un processus standard : estimation de dispersion de données par rapport au plan qu'elles appartiennent en classe pour éviter des biais dûs à quelques erreurs grossières.

Les points tridimensionnels construits sont affichés dans la figure 4.13, où on voit que la reconstruction est qualitativement correcte, et peu de points sont erronés.



FIG. 4.13 – *Les points tridimensionnels (représentés par des petits carrés) construits à partir des points bidimensionnels appariés selon notre méthode, projetés dans un plan horizontal.*

Pour faire une analyse quantitative, nous calculons les deux distances de chaque points aux deux plans d'objet et prenons le minimum d'entre ces deux distances comme l'erreur. La médiane des erreurs est $0.28mm$, le pourcentage des erreurs supérieures à 1 est 12%. Parmi les erreurs inférieures à $1mm$, la moyenne est $0.31mm$, et la racine de la moyenne des carrés est $0.40mm$. L'histogramme des erreurs est suivant :

0 — -1	1 — -2	2 — -3	3 — -4	4 — -5	5 — -6
88%	10.6%	0.8%	0.2%	0.2%	0.2%

De ce tableau, nous voyons que 88% de points ont une erreur inférieure à $1mm$, et 98.6% de points ont une erreur inférieure à $2mm$.

Nous montrons le résultat de la reconstruction tridimensionnelle à partir des points appariés selon la méthode classique. Les points tridimensionnels construits sont affichés dans la figure 4.14, où nous voyons qu'il y a des points très erronés.

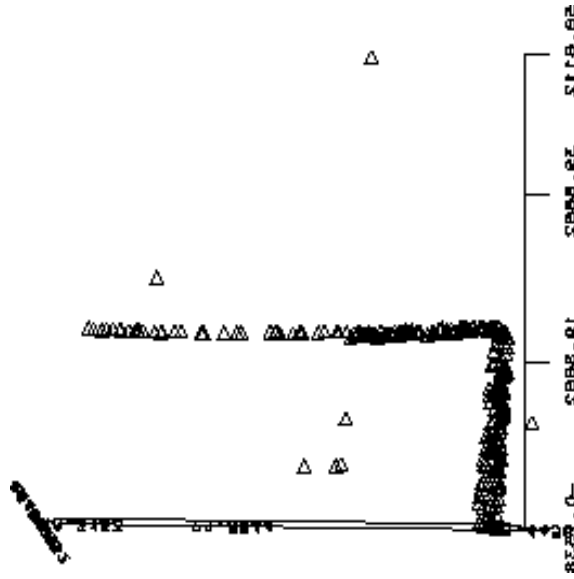


FIG. 4.14 — *Les points tridimensionnels construits (représentés par des triangles) à partir des points bidimensionnels appariés selon la méthode classique, projetés dans un plan horizontal.*

Son résultat quantitatif est suivant : la médiane des erreurs est $0.36mm$, le pourcentage des erreurs supérieures à 1 est 13.6%. Parmi les erreurs inférieures à 1, la moyenne est 0.34 , et la racine de la moyenne des carrés est 0.43 . L'histogramme

des erreurs est suivant :

0 – -1	1 – -2	2 – -3	3 – -4	4 – -5	5 – -6	6 – -7	7 – -8	8 – -9	9 – -10	10 – $-\infty$
86.4%	10.8%	0%	0%	0.40%	0.0%	0.40%	0.0%	0.0%	0.0%	2.0%

Nous voyons que le résultat à partir des appariements obtenus par la méthode classique est un peu moins précis, et contient plus des erreurs aberrantes.

Nous avons seulement fait une expérimentation de la reconstruction tridimensionnelle, plus d'expérimentations sont intéressantes à réaliser, contenant la reconstruction par plusieurs vues.

4.6 Conclusion

En partant des corrélations classiques [For82, Ack84, Gru85], nous avons proposé une méthode beaucoup plus rapide, et elle est robuste contre faux initialisations et est adaptée aux grandes déformations des fenêtres.

Quand il y a des faux appariements, comme pour la paire *New1*, *New3* *New4*, les méthodes classiques fonctionnent seulement pour les points d'intérêt, et donne plus que 20% de faux appariements. Quand les initialisations sont encore pire comme pour la paire *New1* et *New4*, les corrélations classiques ne fonctionnent plus du tout. Au contraire, notre méthode fonctionne jusqu'à pour la paire *New1* et *New4*, et donne des bons résultats.

Notre modèle est un nouvel essai pour incorporer les informations des pixels voisins dans la technique de corrélation en utilisant une masque de convolution. Elle est directe, linéaire et il atteint une précision sous-pixelique. Deux algorithmes sont présentés : un rapide et un robuste. Celui rapide considère seulement le décalage local et atteint une précision de 0.06 pixel (0.05 pour les points d'intérêt) dans le cas de translation pure. Celui robuste prend en compte aussi la déformation des fenêtres entre les deux images et donne une précision de 0.05 pixel quand les deux images se sont proches. Cet algorithme éliminent les faux appariements dans la carte de disparité initiale. Un post traitement présenté dans la section 4.4 améliore encore les résultats.

Plusieurs perspectives de ce travail sont possibles comme suit :

- Son intégration avec la corrélation partielle,
- Son application à la reconstruction tridimensionnelle,

- Son application à l'auto-calibrage et à image mosaicing.

Chapitre 5

Conclusion

Ce travail concerne le problème de l'appariement visuel. Il s'est articulé autour des thèmes de la robustesse et la précision de l'appariement. La robustesse face aux occultations partielles est abordée, ainsi que la précision face aux déformations géométriques, transformations des niveaux de gris et fausses initialisations.

La *corrélation partielle* est proposée pour résoudre le problème d'occultations partielles. Deux méthodes sont présentées : une basée sur la régression robuste (section 3.1.2.2) et une autre sur la localisation robuste (section 3.2.3.1). Les deux méthodes consistent à écarter la partie occultée, et corrélérer la partie restante.

- Corrélation partielle basée sur la régression robuste.

La corrélation partielle basée sur la régression robuste cherche d'abord la meilleure droite qui ajuste les points de coordonnées $\{I_1(X), I_2(X + \Delta X)\}$, où $I_1(I_2)$ sont les intensités lumineuses dans la première (deuxième) image, X et $X + \Delta X$ sont les coordonnées de pixels correspondants dans les deux images. Cet ajustement est trouvé de façon robuste pour éviter des occultations partielles éventuelles. Après avoir trouvé la droite d'ajustement, nous écartons les points qui n'appartiennent pas à la droite, et accomplissons la corrélation sur les restants.

- Corrélation partielle basée sur la localisation robuste.

La corrélation partielle basée sur la localisation robuste cherche d'abord à «localiser» l'ensemble des points de coordonnées $\{I_1(X), I_2(X + \Delta X)\}$, où plus précisément, une ellipse de volume minimal contenant au moins la moitié de points donnés. Après avoir trouvé cette ellipse, nous écartons les points éloignés, et estimons de manière robuste les variances de I_1 et I_2 et leur covariance. Le coefficient de corrélation est calculé comme le rapport

de leur covariance par la moyenne géométrique de leurs variances. Il peut être vu comme ZNCC robustifié.

Des expérimentations montrent que ces deux méthodes fonctionnent bien pour les points proches de contours d'occultation. Elles sont convenables pour localiser les contours d'occultation en analysant la carte de disparités. Il est possible de les intégrer avec la technique de programmation dynamique dans un processus d'appariement dense pour mieux localiser les discontinuités de disparités.

Quant à la *corrélation précise*, nous avons proposé de d'abord estimer la déformation des fenêtres de manière robuste, sur la base des appariements denses initiaux, puis d'estimer le décalage local sur la fenêtre déformée en utilisant un masque de convolution. Sur la base des appariements denses, la géométrie épipolaire peut être réestimée précisément, et elle peut servir à corriger les erreurs perpendiculaires à la ligne épipolaire de chaque appariement. La méthode consiste en quatre étapes :

- Estimation initiale de l'appariement dense.
Elle peut être obtenue par une méthode d'appariement dense, comme la programmation dynamique.
- Estimation robuste de la déformation des fenêtres.
La méthode des moindres carrés médians et celle de RanSaC ont été utilisées pour estimer la déformation des fenêtres et corriger les fausses initialisations à l'aide de la cohérence de disparités locales.
- Estimation du décalage local.
En supposant la régularité locale des intensités lumineuses, le décalage local est obtenu en estimant un masque de convolution. Cette estimation est linéaire et directe.
- Correction via la géométrie épipolaire précise.
Nous réestimons la géométrie épipolaire à partir des appariements précis, puis calculons la droite épipolaire pour chaque point à apparier dans la première image, et projeter son correspondant obtenu sur cette droite pour corriger les erreurs perpendiculaires à cette droite.

Grâce à l'estimation robuste de la transformation des fenêtres, cette méthode est aussi robuste aux fausses initialisations et grandes déformations des fenêtres. Les résultats expérimentaux montrent qu'une précision d'ordre 0,05 pixel est obtenue quand la déformation des fenêtres est petite, et elle atteint 0,08 pixel

quand la déformation est large. Le pourcentage des faux appariements est très petit (inférieur à 5% pour tous les cas).

Deux autres chapitres sont consacrés à la synthèse de travaux antérieurs, bases de ces développements. Un chapitre résume les différentes méthodes robustes, des plus anciennes comme la transformation de Hough et la technique de RanSaC, aux plus nouvelles comme MINPRAN, MUSE, etc ; l'autre chapitre résume des techniques d'appariements visuels comme la corrélation, la distance de Hausdorff, la technique d'invariants locaux et la technique de programmation dynamique.

Pour les méthodes robustes, nous avons discuté de deux «anciennes» méthodes : la transformation de Hough et la technique de RanSaC qui cherchent l'ajustement qui a l'ensemble de points cohérents de cardinal maximal. La transformation de Hough cherche l'ajustement dans l'espace de paramètres de manière exhaustive et elle est très coûteuse, tandis que la technique de RanSaC prend plusieurs tirages aléatoires, construit l'ajustement à partir de ces tirages, et choisit le meilleur parmi eux : elle est donc plus rapide. La méthode LMedS est similaire à ces deux méthodes, mais elle estime l'écart-type du bruit lors de la recherche de l'ajustement, elle est donc plus autonome car l'information de l'écart-type du bruit n'est pas requise.

D'autres méthodes d'estimation robustes plus récentes comme MINPRAN, RESC, MUSE essaient de dépasser le seuil de rupture 50% en ajoutant des hypothèses raisonnables. Ces méthodes, au lieu de prendre la médiane des résidus comme la méthode LMedS, trient les résidus en ordre croissant et les analysent en plus de détails. Grâce à ces traitements plus sophistiqués, elles donnent de meilleurs résultats que LMedS quand plus de la moitié de données sont fausses.

Les techniques d'appariement peuvent être regroupées en :

- Les technique locales.

La corrélation, la technique de la distance de Hausdorff et la technique d'invariants locaux sont des méthodes d'appariements de pixel qui comparent la ressemblance (où la différence) entre le modèle et le candidat et prennent le meilleur. La corrélation prend tous les pixels dans la fenêtre et compose un vecteur caractéristique de dimension large, tandis que la technique d'invariants locaux prend les invariants différentiels d'intensités lumineuses et donne un vecteur caractéristique de dimension relativement petite. Donc, la corrélation est plus discriminante que la technique d'invariants locaux et elle est applicable pour les points texturés, alors que la technique d'invariants locaux est applicable seulement pour les points d'intérêts, mais elle n'est pas sensible à la déformation perspective des images. La technique de distance de Hausdorff compare la ressemblance de deux ensembles de points, autrement dit, deux images binaires. Sa version classique prend le

maximum des distances entre les points d'un ensemble et l'autre ensemble ; elle peut être naturellement étendue à la version généralisée, qui prend le $k^{\text{ième}}$ plus grande distance au lieu d'en prendre le maximum. La version généralisée est robuste aux occultations partielles. Les désavantages de cette technique résident sur sa complexité en temps et le fait qu'elle est difficile à s'employer au cas où il y a des déformations perspectives non négligeables.

- Les techniques globales comme la programmation dynamique.

La technique de programmation dynamique est une méthode globale qui peut se servir à apparier les pixels non texturés dans l'image à l'aide des contraintes globales comme la contrainte d'unicité, la contrainte d'ordre, etc. Elle peut aussi prendre en compte l'occultation en mettant un coût pour chaque occultation. Il est possible d'obtenir des appariements denses en intégrant cette technique avec la technique de corrélation. D'autres techniques globales comme la relaxation existent ; elles n'ont pas été abordées dans ce mémoire.

Les travaux abordés dans ce mémoire suscitent quelques développements qu'il serait intéressant de réaliser :

- La détection d'occultations partielles.

La détection d'occultation partielle peut être intégrée dans la corrélation précise ou la localisation de cibles modélisables. L'idée est de chercher une relation affine pour l'ensemble de couples d'intensités lumineuses, d'écarter les couples qui ne vérifient pas cette relation, et d'estimer le décalage local selon la partie restante.

- Localisation de cibles.

Le travail pourrait être étendu à la localisation de formes géométriques comme les cibles circulaires etc. L'idée est d'utiliser une technique robuste d'ajustement de points pour trouver la forme et d'estimer le décalage local à partir de la forme obtenue.

- Reconstruction de surfaces tridimensionnelles.

Il serait intéressant de réaliser des expérimentations de la reconstruction tridimensionnelle à partir de points bidimensionnels appariés selon notre méthode, non seulement la reconstruction de points, mais aussi celle des surfaces est possible. La reconstruction robuste des surfaces tridimensionnelles à partir des points tridimensionnels est une perspective attirante pour la suite.

- Application de la corrélation à l'auto-calibrage et à image mosaicing.

Beaucoup de processus visuels requièrent l'appariement précis, comme l'auto-calibrage et image mosaicing. Il serait intéressant d'évaluer les influences de notre corrélation précise sur ces applications.

Étant un sujet majeur de la vision par ordinateur, l'appariement visuel semble encore difficile et loin d'être résolu de façon définitive.

Annexe A

Preuve de l'efficacité relative

Dans cette section, nous allons montrer que dans le cas d'une distribution gaussienne, la moyenne d'échantillon atteint la BVM et la médiane d'échantillon a une efficacité relative de seulement $\frac{2}{\pi}$.

La preuve consiste en quatre étapes :

1. Dédution de la condition nécessaire et suffisante pour atteindre la bande de variance minimale dans le cas général.
2. Calcul de la variance de la moyenne d'échantillon dans le cas gaussien.
3. Dédution de la fonction de distribution de la médiane d'échantillon dans le cas général.
4. Calcul de la variance asymptote de la médiane d'échantillon dans le cas gaussien.

Étant donnés n nombres réels x_i , considérons un estimateur $t(x_i)$ d'un paramètre θ inconnu. Soit $L(x_i; \theta)$ la fonction de densité de probabilité. Posons

$$\tau(\theta) = E(t) = \int tL(x_i; \theta)d^n x,$$

où E désigne l'espérance. Nous obtenons

$$\tau'(\theta) = \int t \frac{\partial L}{\partial \theta} d^n x$$

et évidemment

$$\int L(x_i; \theta) d^n x = 1,$$

car L est une fonction de densité.

Donc

$$\frac{\partial}{\partial \theta} \int L(x_i; \theta) d^n x = 0.$$

Nous avons

$$\begin{aligned} \int (t - \tau) \frac{\partial \ln L}{\partial \theta} L d^n x &= \int (t - \tau) \frac{\partial L}{\partial \theta} d^n x \\ &= \int t \frac{\partial L}{\partial \theta} d^n x - \tau \int \frac{\partial L}{\partial \theta} d^n x \\ &= \tau'(\theta) - \tau \frac{\partial}{\partial \theta} \int L(x_i; \theta) d^n x \\ &= \tau'(\theta). \end{aligned}$$

L'inégalité de Cauchy donne

$$\tau'(\theta)^2 \leq \int (t - \tau)^2 L d^n x \int \left(\frac{\partial \ln L}{\partial \theta} \right)^2 L d^n x.$$

Posons $V(t) = \int (t - \tau)^2 L d^n x$, nous obtenons

$$\tau'^2 \leq V(t) \int \left(\frac{\partial \ln L}{\partial \theta} \right)^2 L d^n x.$$

Cette relation est appelée l'inégalité de *Cramer-Rao*.

Dans le cas $\tau(\theta) = \theta$, l'inégalité devient

$$V(t) \geq \frac{1}{\int \left(\frac{\partial L}{\partial \theta} \right)^2 L d^n x}.$$

Elle est appelée BVM : *bande de variance minimale*.

Pour l'atteindre, nous exigeons qu'il existe $A(\theta)$, tel que :

$$A(\theta)(t - \tau) = \frac{\partial \ln L}{\partial \theta}. \quad (\text{A.1})$$

Dans ce cas,

$$V(t) = E(t - \tau)^2 = \frac{E((t - \tau) \frac{\partial \ln L}{\partial \theta})}{A(\theta)} = \frac{\tau'(\theta)}{A(\theta)} = \frac{1}{A(\theta)}.$$

Nous obtenons donc la condition nécessaire et suffisante pour atteindre la BVM (la *Bande de Variance Minimale*). Dans le cas gaussien, nous avons :

$$L = Ce^{-\frac{\sum_i (x_i - \theta)^2}{\sigma^2}}$$

(C est constant) et

$$\frac{\partial \ln L}{\partial \theta} = \frac{\sum_i (x_i - \theta)}{\sigma^2}.$$

Si nous choisissons $t = \frac{\sum_i x_i}{n}$ et $\tau(\theta) = \theta$, nous obtenons :

$$\frac{\frac{\partial \ln L}{\partial \theta}}{t - \tau} = \frac{\frac{\sum_i (x_i - \theta)}{\sigma^2}}{\frac{\sum_i (x_i - \theta)}{n}} = \frac{n}{\sigma^2}.$$

La condition (A.1) est vérifiée avec $A(\theta) = \frac{n}{\sigma^2}$. La variance minimale est $\frac{1}{A(\theta)} = \frac{\sigma^2}{n}$, qui est atteinte par la moyenne.

Considérons maintenant la médiane d'un échantillon de taille $n = 2v + 1$ (le cas $n = 2v$ est similaire). Supposons que la fonction de distribution de la population est F (donc la fonction de densité de probabilité est $f = F'$). Notons g la fonction de densité de probabilité de la médiane de $(x_i)_{i=1}^n$, nous avons alors :

$$g(x) = C_n^1 C_{2v}^v F(x)^v (1 - F(x))^v f(x).$$

Soit ξ la médiane de la distribution, c'est-à-dire : $F(\xi) = \frac{1}{2}$ et nous avons

$$F(x) = F(\xi) + (x - \xi)f(\xi) + o(x - \xi)$$

où $\lim_{x \rightarrow \xi} \frac{o(x - \xi)}{x - \xi} = 0$.

Notons $C = C_n^1 C_{2v}^v$, alors :

$$\begin{aligned} g(x) &= C \left(\frac{1}{2} + (x - \xi)f(\xi) + o(x - \xi) \right)^v \left(\frac{1}{2} - (x - \xi)f(\xi) + o(x - \xi) \right)^v f(x) \\ &\approx \frac{C}{4^v} (1 - 4((x - \xi)f(\xi))^2)^v f(x). \end{aligned}$$

Si $v \rightarrow \infty$, $g(x)$ est proportionnel à

$$(1 - (2(x - \xi)f(\xi))^2)^v \sim e^{-v(2f(\xi)(x-\xi))^2} \sim e^{-\frac{v}{2}(2f(\xi)(x-\xi))^2}$$

où $\sim$ note la relation proportionnelle.

Sa distribution est asymptotement $N(\xi, \frac{1}{4f^2(\xi)n})$.

Dans le cas gaussien où $f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\theta)^2}{2\sigma^2}}$, la médiane de F est $\xi = \theta$, donc $f(\xi) = \frac{1}{\sqrt{2\pi}\sigma}$, et la variance de la médiane d'échantillon devient $\frac{1}{4f^2(\xi)n} = \frac{2\pi\sigma^2}{4n} = \frac{\pi\sigma^2}{2n}$. Son efficacité relative est donc $\frac{\frac{\sigma^2}{n}}{\frac{\pi\sigma^2}{2n}} = \frac{2}{\pi}$.

Annexe B

Une analyse statistique de SSD et ZSSD

Dans cette appendice, nous allons donner une analyse de SSD et ZSSD, qui montre la possibilité de déduire l'intervalle de confiance et le rapport signal bruit sous des hypothèses raisonnables.

Soient $(a)_{i=1}^n$ et $(b)_{i=1}^n$ deux signaux bruités venant d'un même signal non bruité s (le signal sous-jacent) :

$$\begin{aligned} a_i &= s_i + n_i, \\ b_i &= s_i + m_i \end{aligned} \tag{B.1}$$

où s_i est le signal sous-jacent, n_i et m_i sont du bruit.

Supposons que n_i et m_i sont non corrélés et de distribution identique, d'où nous avons $En_i^2 = Em_i^2 = \sigma^2$, $En_i m_i = 0$.

Pour analyser SSD, nous supposons de plus que n_i et m_i sont centrés : $En_i = Em_i = 0$, alors :

$$\frac{E \sum_{i=1}^n (a_i - b_i)^2}{2n} = \frac{E \sum_{i=1}^n (n_i - m_i)^2}{2n} = \sigma^2. \tag{B.2}$$

Si nous connaissons la distribution de n_i et m_i , nous pouvons savoir la probabilité que l'hypothèse (B.1) est vérifiée. Par exemple : si leur distribution est gaussienne, il suffit d'effectuer un test de χ^2 de degré $2n$ sur $\frac{\sum_{i=1}^n (a_i - b_i)^2}{2n}$.

Si nous supposons de plus que les s_i sont de distribution identique, nous

pouvons estimer la moyenne de s_i comme la moyenne de tous les a_i et b_j :

$$\frac{\sum_{i=1}^n (a_i + b_i)}{2n} \quad (\text{B.3})$$

Nous pouvons aussi estimer la variance de s à partir de la variance de population de a et b , et on pouvons utiliser un test du rapport signal par bruit en combinant les estimations (B.2) et (B.3).

Pour analyser ZSSD, nous avons besoin des conditions suivantes : les s_i sont de variance σ_s^2 , non corrélés avec les bruits, et non corrélés entre eux.

Nous pouvons obtenir directement :

$$E \sum_{i=1}^n (a_i - \bar{a} - b_i + \bar{b})^2 = 2(n-1)\sigma_n^2, \quad (\text{B.4})$$

$$E \left(\sum_{i=1}^n (a_i - \bar{a})^2 + \sum_{i=1}^n (b_i - \bar{b})^2 \right) = 2(n-1)(\sigma_s^2 + \sigma_n^2). \quad (\text{B.5})$$

Nous pouvons donc utiliser l'intervalle de confiance avec l'estimation (B.4) comme dans le cas SSD. Et si nous rajoutons l'hypothèse que les s_i sont de distribution identique, nous pouvons estimer la moyenne des s_i , et en combinant (B.4) et (B.5), nous pouvons utiliser un test de rapport signal par bruit.

C'est seulement une analyse statistique grossière de SSD et ZSSD.

Annexe C

Résultats expérimentaux de la corrélation précise

Dans cette appendice, nous montrons les résultats expérimentaux de la corrélation précise proposée dans le chapitre 4. La section C.1 montre les résultats du cas translationnel dont la méthode est proposée dans la section 4.2, la section C.2 présente les résultats du cas général dont la méthode est expliquée dans la section 4.3, et la section C.3 montre les résultats après le post-traitement en incorporant la géométrie épipolaire dont la méthode est présentée dans la section 4.4.

C.1 Résultats expérimentaux de la corrélation précise : cas translationnel

Cette section montre les résultats de la corrélation précise sur trois paires d'images où le déplacement entre les deux images est essentiellement une translation. Ces trois paires d'images sont appelées la paire *Bolino1* et *Bolino2*, la paire *Bolino1* et *Bolino3* et la paire *Bolino1* et *Bolino4* et elles sont affichées dans la figure 4.3 à la page 108, figure 4.5 à la page 110, et figure 4.6 à la page 111 respectivement.

L'ouverture d'iris de l'objectif reste inchangé entre *Bolino1* et *Bolino2* et il y a peu de changement de luminosités. Selon l'ouverture d'iris, entre *Bolino1* et *Bolino3*, le changement de luminosités est non négligeable, et entre *Bolino1* et *Bolino4*, le changement est encore plus grand, Le niveau de gris moyen de la partie texturée de ces images est de 125 pour *Bolino1*, 126 pour *Bolino2*, 95 pour *Bolino3* et 78 pour *Bolino4*.

Pour chaque paire, les résultats pour les points d'intérêts et les points aléa-

toirement choisis (figure 4.4) sont montrés.

Dans chaque tableau, le taux de faux appariements (o_p), la moyenne des erreurs (m_1) et la racine carrée de la moyenne des erreurs au carré (m_2) sont affichées, pour les différentes tailles de la fenêtre de corrélation (qui varient de 7 à 19).

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	0,07	0,00	0,00	0,00	0,00	0,00	0,00
	m_1	0,09	0,08	0,07	0,07	0,07	0,08	0,08
	m_2	0,11	0,09	0,09	0,09	0,09	0,09	0,09
4nei2	$o_p\%$	0,07	0,00	0,00	0,00	0,00	0,00	0,00
	m_1	0,08	0,07	0,07	0,07	0,07	0,07	0,07
	m_2	0,10	0,09	0,08	0,08	0,08	0,09	0,09
8nei	$o_p\%$	0,07	0,00	0,00	0,00	0,00	0,00	0,00
	m_1	0,11	0,09	0,08	0,08	0,08	0,08	0,08
	m_2	0,13	0,10	0,09	0,09	0,09	0,09	0,09
8nei2	$o_p\%$	0,07	0,00	0,00	0,00	0,00	0,00	0,00
	m_1	0,08	0,08	0,07	0,07	0,07	0,07	0,07
	m_2	0,10	0,09	0,08	0,08	0,08	0,08	0,08

TAB. C.1 – Résultats pour les points d'intérêt de la paire Bolino1 et Bolino2 (figure 4.3) : le déplacement entre ces deux images est essentiellement une translation.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	4	2	0,9	0,7	0,4	0,3	0,07
	m_1	0,16	0,11	0,10	0,09	0,09	0,08	0,09
	m_2	0,22	0,15	0,12	0,11	0,11	0,10	0,11
4nei2	$o_p\%$	4	2	0,9	0,7	0,4	0,4	0,07
	m_1	0,17	0,12	0,10	0,09	0,08	0,08	0,09
	m_2	0,23	0,17	0,14	0,12	0,11	0,11	0,11
8nei	$o_p\%$	5	2	0,9	0,7	0,4	0,4	0,07
	m_1	0,20	0,14	0,11	0,10	0,09	0,09	0,09
	m_2	0,27	0,18	0,14	0,12	0,12	0,11	0,11
8nei2	$o_p\%$	4	2	0,9	0,7	0,4	0,4	0,07
	m_1	0,18	0,12	0,10	0,09	0,08	0,08	0,08
	m_2	0,26	0,18	0,14	0,12	0,11	0,10	0,11

TAB. C.2 – Résultats pour des points généraux de la paire Bolino1 et Bolino2 (figure 4.3) : le déplacement entre ces deux images est essentiellement une translation.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	1,37	1,16	1,09	0,96	0,96	0,96	0,96
	m_1	0,10	0,08	0,08	0,08	0,08	0,08	0,08
	m_2	0,12	0,10	0,09	0,09	0,09	0,10	0,10
4nei2	$o_p\%$	1,37	1,16	1,09	0,89	0,89	0,89	0,89
	m_1	0,09	0,08	0,07	0,07	0,07	0,08	0,08
	m_2	0,11	0,09	0,09	0,09	0,09	0,09	0,09
8nei	$o_p\%$	1,37	1,16	1,09	0,96	0,96	0,96	0,96
	m_1	0,12	0,09	0,08	0,08	0,08	0,08	0,08
	m_2	0,14	0,11	0,10	0,10	0,10	0,10	0,10
8nei2	$o_p\%$	1,30	1,16	1,09	0,89	0,89	0,89	0,89
	m_1	0,10	0,08	0,07	0,07	0,07	0,08	0,08
	m_2	0,12	0,09	0,09	0,09	0,09	0,09	0,09

TAB. C.3 – Résultats pour les points d'intérêt pour la paire Bolino1 et Bolino3 (figure 4.5) : le déplacement entre ces deux images est essentiellement une translation.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	6,71	3,64	2,72	2,36	2,15	2,08	1,65
	m_1	0,18	0,12	0,10	0,09	0,09	0,09	0,09
	m_2	0,25	0,17	0,14	0,11	0,11	0,11	0,11
4nei2	$o_p\%$	7,14	3,86	2,86	2,29	2,15	2,08	1,72
	m_1	0,19	0,13	0,10	0,09	0,09	0,08	0,09
	m_2	0,26	0,19	0,15	0,12	0,11	0,11	0,11
8nei	$o_p\%$	7,79	4,29	2,86	2,15	2,07	2,01	1,65
	m_1	0,22	0,15	0,12	0,10	0,09	0,09	0,09
	m_2	0,28	0,20	0,16	0,13	0,12	0,12	0,12
8nei2	$o_p\%$	7,50	4,07	2,93	2,15	1,93	1,93	1,58
	m_1	0,19	0,13	0,10	0,09	0,09	0,08	0,09
	m_2	0,26	0,19	0,15	0,12	0,12	0,11	0,11

TAB. C.4 – Résultats pour des points généraux pour la paire Bolino1 et Bolino3 (figure 4.5) : le déplacement entre ces deux images est essentiellement une translation.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	3.96	3.69	3.14	3.01	2.75	2.68	2.54
	m_1	0.11	0.09	0.09	0.09	0.09	0.09	0.09
	m_2	0.14	0.11	0.11	0.11	0.11	0.12	0.13
4nei2	$o_p\%$	3.96	3.69	3.28	3.01	2.75	2.61	2.68
	m_1	0.10	0.09	0.08	0.08	0.08	0.08	0.08
	m_2	0.12	0.10	0.10	0.10	0.10	0.11	0.11
8nei	$o_p\%$	3.89	3.55	3.14	2.95	2.54	2.47	2.61
	m_1	0.14	0.11	0.10	0.09	0.09	0.10	0.09
	m_2	0.17	0.13	0.12	0.11	0.12	0.12	0.12
8nei2	$o_p\%$	3.83	3.42	3.08	2.81	2.40	2.33	2.33
	m_1	0.11	0.09	0.08	0.08	0.08	0.08	0.08
	m_2	0.13	0.10	0.10	0.10	0.10	0.10	0.10

TAB. C.5 – Résultats pour les points d'intérêt pour la paire Bolino1 et Bolino4 (figure 4.6) : le déplacement entre ces deux images est essentiellement une translation.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	6.71	3.64	2.72	2.36	2.15	2.08	1.65
	m_1	0.18	0.12	0.10	0.09	0.09	0.09	0.09
	m_2	0.25	0.17	0.14	0.11	0.11	0.11	0.11
4nei2	$o_p\%$	7.14	3.86	2.86	2.29	2.15	2.08	1.72
	m_1	0.19	0.13	0.10	0.09	0.09	0.08	0.09
	m_2	0.26	0.19	0.15	0.12	0.11	0.11	0.11
8nei	$o_p\%$	7.79	4.29	2.86	2.15	2.07	2.01	1.65
	m_1	0.22	0.15	0.12	0.10	0.09	0.09	0.09
	m_2	0.28	0.20	0.16	0.13	0.12	0.12	0.12
8nei2	$o_p\%$	7.50	4.07	2.93	2.15	1.93	1.93	1.58
	m_1	0.19	0.13	0.10	0.09	0.09	0.08	0.09
	m_2	0.26	0.19	0.15	0.12	0.12	0.11	0.11

TAB. C.6 – Résultats pour des points généraux pour la paire Bolino1 et Bolino4 (figure 4.6) : le déplacement entre ces deux images est essentiellement une translation.

C.2 Résultats expérimentaux de la corrélation précise : cas général

Cette section montre les résultats de la corrélation précise sur trois paires d'images où le déplacement est affine. Ces trois paires d'images sont appelées la paire *New1* et *New2*, la paire *New1* et *New3* et la paire *New1* et *New4* et elles sont affichées dans la figure 4.7 à la page 117, la figure 4.9 à la page 119 et la figure 4.10 à la page 120.

Entre *New1* et *New2*, il y a un changement d'échelle géométrique proche de $1 : 0,9$, et il n'y a peu de rotations ; entre *New1* et *New3*, il y a une rotation proche de 20 degrés et un changement d'échelle géométrique proche de $1 : 0,9$; entre *New1* et *New4*, il y a une rotation proche de 30 degrés et un effet d'échelle proche de $1 : 0,9$.

Pour chaque paire, les résultats pour les points d'intérêts et les points aléatoirement choisis (figure 4.8) sont montrés.

Dans chaque tableau, le taux de faux appariements (o_p), la moyenne des erreurs (m_1) et la racine carrée de la moyenne des erreurs au carré (m_2) sont affichées, pour les différentes tailles de la fenêtre de corrélation (qui varient de 7 à 29).

La subsection C.2.1 montre les résultats en utilisant la méthode des moindres carrés médians pour l'estimation de la transformation des fenêtres, et la subsection C.2.2 présente les résultats avec la technique de RanSaC, et aussi sa variation «RanSaC2». Ces techniques de l'estimation de la transformation des fenêtres sont présentés dans la section 4.3.1 à la page 112.

En résumé, parmi ces trois méthodes d'estimation, la méthode des moindres carrés donne des meilleurs résultats mais elle est la plus lente, la technique de RanSaC est de moins bons résultats et elle est plus rapide, RanSaC2 donne des résultats de qualité intermédiaire et elle prend un temps d'exécution intermédiaire.

C.2.1 Résultats par la méthode des moindres carrés médians

Cette subsection présente les résultats expérimentaux de la corrélation précise en utilisant la méthode des moindres carrés médians. À la page 145 sont donnés plus d'explication.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,16	0,14	0,11	0,10	0,08	0,07	0,06	0,05	0,05	0,05	0,05	0,04
	m_2	0,18	0,16	0,14	0,12	0,10	0,08	0,07	0,07	0,07	0,06	0,06	0,06
4nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,14	0,13	0,12	0,10	0,09	0,07	0,07	0,06	0,06	0,06	0,06	0,05
	m_2	0,16	0,15	0,13	0,12	0,11	0,09	0,08	0,07	0,08	0,07	0,07	0,07
8nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,21	0,16	0,13	0,11	0,09	0,08	0,07	0,06	0,06	0,06	0,06	0,06
	m_2	0,24	0,18	0,15	0,13	0,11	0,09	0,08	0,07	0,07	0,07	0,07	0,07
8nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,14	0,13	0,11	0,10	0,09	0,07	0,06	0,06	0,06	0,05	0,05	0,05
	m_2	0,16	0,15	0,13	0,12	0,10	0,09	0,08	0,07	0,07	0,07	0,07	0,07

TAB. C.7 – Les points d'intérêt pour New : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.7).

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	4,57	2,00	1,29	0,57	0,29	0,14	0	0	0	0	0	0
	m_1	0,26	0,20	0,17	0,15	0,13	0,11	0,09	0,09	0,08	0,07	0,06	0,06
	m_2	0,32	0,26	0,22	0,20	0,18	0,15	0,14	0,13	0,12	0,11	0,10	0,10
4nei2	$o_p\%$	4,14	1,71	1,14	0,43	0,71	0,29	0	0,14	0	0	0	0
	m_1	0,24	0,20	0,17	0,15	0,13	0,11	0,10	0,09	0,09	0,08	0,07	0,07
	m_2	0,31	0,26	0,22	0,20	0,18	0,16	0,15	0,14	0,14	0,12	0,11	0,11
8nei	$o_p\%$	7,00	2,43	1,43	1,29	0,29	0,14	0	0	0	0	0	0
	m_1	0,30	0,22	0,19	0,16	0,14	0,12	0,11	0,10	0,09	0,08	0,08	0,08
	m_2	0,36	0,28	0,24	0,21	0,19	0,17	0,15	0,14	0,13	0,12	0,11	0,11
8nei2	$o_p\%$	6,00	2,00	1,71	0,86	0,29	0,29	0,29	0,43	0,14	0	0	0
	m_1	0,24	0,20	0,18	0,15	0,13	0,12	0,10	0,10	0,09	0,08	0,08	0,07
	m_2	0,32	0,27	0,24	0,22	0,20	0,18	0,16	0,15	0,14	0,13	0,12	0,11

TAB. C.8 – Des points généraux pour New : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.7).

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,16	0,12	0,10	0,09	0,09	0,08	0,08	0,07	0,07	0,07	0,06	0,06
	m_2	0,20	0,14	0,13	0,12	0,11	0,11	0,10	0,10	0,10	0,10	0,10	0,10
4nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,16	0,13	0,11	0,10	0,10	0,09	0,09	0,08	0,08	0,08	0,07	0,07
	m_2	0,19	0,15	0,14	0,13	0,12	0,12	0,11	0,11	0,11	0,11	0,10	0,10
8nei	$o_p\%$	0,27	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,22	0,14	0,12	0,11	0,10	0,09	0,09	0,08	0,07	0,07	0,07	0,07
	m_2	0,26	0,17	0,14	0,13	0,12	0,11	0,11	0,10	0,10	0,10	0,10	0,10
8nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,16	0,13	0,11	0,10	0,10	0,09	0,08	0,08	0,08	0,08	0,07	0,07
	m_2	0,20	0,15	0,14	0,13	0,12	0,11	0,11	0,11	0,10	0,11	0,10	0,10

TAB. C.9 – Les points d'intérêt pour New : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.9).

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	7,29	4,00	2,29	2,00	1,71	1,71	1,57	1,00	1,71	1,43	2,29	2,00
	m_1	0,26	0,19	0,16	0,14	0,12	0,12	0,11	0,11	0,10	0,10	0,09	0,09
	m_2	0,33	0,25	0,22	0,20	0,17	0,17	0,17	0,18	0,17	0,17	0,15	0,16
4nei2	$o_p\%$	6,57	3,57	2,29	2,00	1,57	1,43	1,29	1,00	1,43	1,57	2,14	1,71
	m_1	0,26	0,20	0,17	0,15	0,14	0,13	0,13	0,12	0,12	0,11	0,11	0,11
	m_2	0,33	0,27	0,23	0,20	0,19	0,19	0,19	0,19	0,19	0,17	0,16	0,17
8nei	$o_p\%$	9,00	4,57	3,14	2,00	1,43	1,71	1,43	1,14	1,71	1,71	2,14	2,00
	m_1	0,31	0,21	0,17	0,16	0,14	0,13	0,13	0,12	0,11	0,11	0,11	0,10
	m_2	0,38	0,27	0,23	0,22	0,20	0,18	0,18	0,18	0,17	0,17	0,17	0,16
8nei2	$o_p\%$	7,71	3,71	2,43	2,14	1,71	1,71	1,57	1,43	1,71	1,71	2,00	2,00
	m_1	0,26	0,20	0,17	0,15	0,13	0,13	0,12	0,12	0,12	0,11	0,11	0,11
	m_2	0,34	0,26	0,23	0,21	0,19	0,18	0,18	0,18	0,18	0,17	0,17	0,17

TAB. C.10 – Des points généraux pour New : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.9).

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0,27
	m_1	0,19	0,16	0,14	0,12	0,11	0,10	0,10	0,09	0,09	0,08	0,08	0,07
	m_2	0,22	0,19	0,17	0,15	0,14	0,13	0,12	0,12	0,11	0,11	0,11	0,10
4nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0,27	0,27
	m_1	0,18	0,16	0,15	0,13	0,12	0,12	0,11	0,11	0,10	0,10	0,10	0,09
	m_2	0,21	0,19	0,18	0,16	0,15	0,14	0,13	0,13	0,13	0,13	0,12	0,12
8nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0,00	0,27
	m_1	0,26	0,19	0,16	0,15	0,13	0,12	0,11	0,11	0,10	0,09	0,09	0,08
	m_2	0,31	0,23	0,20	0,18	0,16	0,15	0,14	0,13	0,12	0,12	0,12	0,11
8nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0,27	0,27
	m_1	0,19	0,17	0,15	0,14	0,13	0,12	0,11	0,11	0,10	0,10	0,10	0,09
	m_2	0,22	0,20	0,18	0,17	0,15	0,15	0,14	0,14	0,13	0,13	0,12	0,12

TAB. C.11 – Les points d'intérêt pour New : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images. (figure 4.10).

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	5,29	3,43	2,14	1,57	1,29	1,29	1,14	1,14	1,43	1,57	1,14	1,71
	m_1	0,26	0,22	0,18	0,17	0,15	0,14	0,13	0,12	0,12	0,11	0,11	0,11
	m_2	0,33	0,28	0,23	0,22	0,20	0,19	0,18	0,18	0,17	0,18	0,18	0,17
4nei2	$o_p\%$	5,14	3,14	1,71	1,57	1,43	1,57	1,43	1,43	1,43	1,57	1,14	1,71
	m_1	0,27	0,23	0,19	0,18	0,17	0,16	0,15	0,14	0,14	0,13	0,13	0,13
	m_2	0,33	0,29	0,25	0,23	0,22	0,21	0,20	0,20	0,19	0,19	0,19	0,18
8nei	$o_p\%$	8,00	3,71	1,71	2,00	1,00	1,14	1,14	1,14	1,43	1,71	1,29	1,71
	m_1	0,33	0,26	0,21	0,19	0,18	0,17	0,15	0,14	0,13	0,13	0,12	0,12
	m_2	0,40	0,31	0,27	0,24	0,24	0,22	0,20	0,20	0,19	0,19	0,19	0,18
8nei2	$o_p\%$	6,43	3,71	2,29	2,29	1,71	1,71	1,57	1,71	1,43	1,71	1,14	1,71
	m_1	0,29	0,24	0,21	0,19	0,18	0,17	0,15	0,15	0,14	0,14	0,13	0,13
	m_2	0,36	0,30	0,26	0,24	0,23	0,22	0,21	0,21	0,20	0,20	0,20	0,19

TAB. C.12 – Des points généraux pour New : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.10).

C.2.2 Résultats par la technique de RanSaC

Cette subsection présente les résultats expérimentaux de la corrélation précise en utilisant la technique de RanSaC (subsection C.2.2.1) et sa variation RanSaC2 (subsection C.2.2.1). À la page 145 sont données plus d'explication.

C.2.2.1 RanSaC

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,16	0,13	0,11	0,10	0,08	0,08	0,07	0,06	0,06	0,06	0,05	0,05
	m_2	0,18	0,15	0,13	0,11	0,10	0,10	0,08	0,08	0,08	0,07	0,07	0,07
4nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,14	0,12	0,11	0,10	0,09	0,08	0,07	0,07	0,07	0,06	0,06	0,06
	m_2	0,16	0,14	0,13	0,12	0,10	0,10	0,09	0,09	0,08	0,08	0,08	0,07
8nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,21	0,15	0,13	0,11	0,10	0,09	0,08	0,08	0,07	0,07	0,06	0,06
	m_2	0,24	0,18	0,15	0,13	0,12	0,10	0,10	0,09	0,09	0,08	0,08	0,08
8nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,14	0,12	0,11	0,10	0,09	0,08	0,07	0,07	0,07	0,06	0,06	0,06
	m_2	0,16	0,14	0,13	0,12	0,11	0,10	0,09	0,09	0,09	0,08	0,08	0,07

TAB. C.13 – Les points d'intérêt pour New : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'image (figure 4.7), méthode RanSaC.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	3,86	2	1,43	0,43	0,71	0,29	0	0	0	0	0	0
	m_1	0,26	0,20	0,16	0,15	0,13	0,11	0,10	0,09	0,09	0,08	0,08	0,07
	m_2	0,32	0,25	0,22	0,20	0,17	0,16	0,16	0,14	0,13	0,12	0,13	0,12
4nei2	$o_p\%$	3,43	1,43	1,14	0,43	1,14	0,43	0	0,14	0	0	0	0
	m_1	0,24	0,19	0,17	0,15	0,13	0,12	0,11	0,10	0,09	0,09	0,09	0,09
	m_2	0,31	0,25	0,22	0,21	0,18	0,17	0,16	0,15	0,14	0,13	0,14	0,13
8nei	$o_p\%$	5,86	2	1,71	0,57	0,71	0,29	0,14	0,14	0	0	1,43	0
	m_1	0,31	0,23	0,19	0,17	0,14	0,13	0,12	0,11	0,10	0,09	0,09	0,09
	m_2	0,38	0,28	0,24	0,23	0,19	0,17	0,16	0,15	0,14	0,13	0,13	0,13
8nei2	$o_p\%$	5,29	2,14	1,71	0,86	0,71	0,71	0,29	0,29	0,29	0	0,14	0
	m_1	0,25	0,20	0,17	0,15	0,14	0,12	0,11	0,10	0,09	0,09	0,09	0,09
	m_2	0,33	0,27	0,23	0,22	0,20	0,17	0,17	0,15	0,14	0,14	0,14	0,14

TAB. C.14 – Des points généraux pour New : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.7), méthode RanSaC.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0,27	0,27	0	0	0	0	0,27	0	0,27	0,27
	m_1	0,18	0,13	0,11	0,10	0,10	0,10	0,09	0,08	0,08	0,08	0,08	0,07
	m_2	0,21	0,16	0,14	0,13	0,12	0,13	0,13	0,11	0,11	0,12	0,11	0,11
4nei2	$o_p\%$	0	0	0	0	0	0	0	0	0,27	0	0,27	0
	m_1	0,16	0,14	0,12	0,11	0,10	0,10	0,10	0,09	0,09	0,09	0,08	0,08
	m_2	0,20	0,16	0,15	0,15	0,13	0,13	0,13	0,11	0,12	0,12	0,11	0,13
8nei	$o_p\%$	0,27	0	0	0	0	0	0	0	0	0	0,27	0
	m_1	0,23	0,16	0,14	0,13	0,11	0,11	0,11	0,09	0,09	0,09	0,09	0,09
	m_2	0,28	0,19	0,17	0,16	0,14	0,14	0,13	0,11	0,13	0,12	0,12	0,13
8nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,17	0,13	0,12	0,11	0,10	0,10	0,10	0,08	0,09	0,09	0,09	0,08
	m_2	0,21	0,16	0,14	0,16	0,14	0,14	0,13	0,11	0,13	0,12	0,13	0,13

TAB. C.15 – Les points d'intérêt pour New : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.9), méthode RanSaC.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	7, 14	4, 86	3, 29	2, 43	2, 29	2, 71	2, 57	2, 14	3	3, 14	2, 57	2, 43
	m_1	0, 27	0, 20	0, 17	0, 15	0, 14	0, 13	0, 13	0, 12	0, 11	0, 11	0, 11	0, 11
	m_2	0, 35	0, 27	0, 23	0, 22	0, 20	0, 19	0, 19	0, 19	0, 18	0, 17	0, 17	0, 19
4nei2	$o_p\%$	7, 71	4, 43	3, 43	2, 57	2, 43	2, 43	3	2, 14	2, 86	3, 14	2, 43	2, 57
	m_1	0, 26	0, 21	0, 18	0, 16	0, 15	0, 14	0, 13	0, 13	0, 12	0, 12	0, 12	0, 12
	m_2	0, 32	0, 28	0, 24	0, 22	0, 21	0, 20	0, 19	0, 20	0, 20	0, 18	0, 18	0, 19
8nei	$o_p\%$	9, 29	5, 14	3, 86	2, 43	2, 43	2, 57	2, 43	2, 14	3	3	2, 57	2, 14
	m_1	0, 33	0, 24	0, 19	0, 17	0, 16	0, 15	0, 14	0, 13	0, 13	0, 12	0, 12	0, 13
	m_2	0, 40	0, 30	0, 24	0, 24	0, 22	0, 20	0, 21	0, 20	0, 19	0, 18	0, 18	0, 20
8nei2	$o_p\%$	8, 57	4, 57	3, 43	3	2, 43	2, 14	2, 57	2	3, 29	3, 14	2, 43	2, 71
	m_1	0, 26	0, 21	0, 18	0, 15	0, 15	0, 14	0, 13	0, 13	0, 12	0, 12	0, 12	0, 12
	m_2	0, 34	0, 28	0, 24	0, 21	0, 21	0, 20	0, 20	0, 20	0, 17	0, 18	0, 18	0, 19

TAB. C.16 – Des points generaux pour New : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.9) , méthode RanSaC.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0	0	0	0	0	0, 53	0, 27	0	0, 27	0, 27
	m_1	0, 20	0, 17	0, 15	0, 15	0, 13	0, 12	0, 12	0, 11	0, 10	0, 10	0, 10	0, 09
	m_2	0, 24	0, 20	0, 19	0, 18	0, 16	0, 15	0, 15	0, 14	0, 13	0, 13	0, 13	0, 13
4nei2	$o_p\%$	0	0	0	0	0	0	0	0, 53	0, 27	0	0	0
	m_1	0, 19	0, 17	0, 15	0, 15	0, 14	0, 13	0, 12	0, 12	0, 12	0, 11	0, 12	0, 10
	m_2	0, 23	0, 20	0, 19	0, 18	0, 17	0, 16	0, 16	0, 15	0, 14	0, 13	0, 15	0, 14
8nei	$o_p\%$	0, 27	0	0	0	0	0	0	0, 53	0, 27	0	0, 27	0, 27
	m_1	0, 28	0, 21	0, 18	0, 17	0, 15	0, 14	0, 13	0, 13	0, 12	0, 11	0, 11	0, 10
	m_2	0, 33	0, 25	0, 21	0, 21	0, 18	0, 18	0, 17	0, 16	0, 15	0, 14	0, 14	0, 13
8nei2	$o_p\%$	0	0	0	0	0	0	0	0, 53	0, 27	0	0, 27	0, 27
	m_1	0, 21	0, 19	0, 17	0, 16	0, 14	0, 14	0, 13	0, 13	0, 12	0, 12	0, 12	0, 11
	m_2	0, 24	0, 22	0, 20	0, 20	0, 17	0, 17	0, 17	0, 16	0, 15	0, 14	0, 15	0, 14

TAB. C.17 – Les points d'intérêt pour New : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.10) , méthode RanSaC.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	3, 70	3, 43	2, 71	2, 29	1, 86	2, 00	2, 57	1, 86	1, 71	1, 71	2, 14	2, 14
	m_1	0, 29	0, 24	0, 20	0, 18	0, 17	0, 16	0, 15	0, 15	0, 14	0, 13	0, 13	0, 13
	m_2	0, 35	0, 30	0, 25	0, 23	0, 22	0, 21	0, 20	0, 20	0, 20	0, 19	0, 20	0, 19
4nei2	$o_p\%$	3, 70	3, 14	2, 00	2, 29	1, 86	2	3, 14	2, 14	1, 71	1, 71	2	2, 29
	m_1	0, 27	0, 24	0, 21	0, 19	0, 18	0, 18	0, 16	0, 16	0, 15	0, 15	0, 15	0, 14
	m_2	0, 32	0, 30	0, 27	0, 25	0, 24	0, 23	0, 21	0, 21	0, 22	0, 20	0, 21	0, 20
8nei	$o_p\%$	7, 41	4, 71	3	2, 57	2, 00	2, 29	2, 57	1, 86	1, 57	1, 71	2, 00	2, 14
	m_1	0, 35	0, 27	0, 23	0, 21	0, 20	0, 18	0, 17	0, 17	0, 16	0, 15	0, 15	0, 15
	m_2	0, 40	0, 33	0, 28	0, 26	0, 25	0, 24	0, 23	0, 22	0, 22	0, 21	0, 22	0, 21
8nei2	$o_p\%$	5, 56	3, 57	2, 14	2, 43	2, 00	2, 29	3, 00	2, 29	2, 29	2, 14	2, 14	2, 43
	m_1	0, 30	0, 26	0, 23	0, 21	0, 20	0, 19	0, 17	0, 17	0, 16	0, 15	0, 16	0, 15
	m_2	0, 34	0, 32	0, 29	0, 26	0, 26	0, 24	0, 23	0, 23	0, 22	0, 20	0, 22	0, 20

TAB. C.18 – Des points generaux pour New : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.10) , méthode RanSaC.

C.2.2.2 RanSaC2

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,15	0,13	0,11	0,09	0,07	0,06	0,06	0,06	0,05	0,05	0,05	0,05
	m_2	0,18	0,15	0,13	0,11	0,09	0,08	0,07	0,07	0,07	0,06	0,06	0,06
4nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,14	0,12	0,11	0,09	0,08	0,07	0,07	0,06	0,06	0,06	0,06	0,06
	m_2	0,16	0,14	0,12	0,11	0,09	0,08	0,08	0,08	0,08	0,07	0,07	0,07
8nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,20	0,15	0,12	0,10	0,09	0,07	0,07	0,07	0,07	0,06	0,06	0,06
	m_2	0,24	0,17	0,14	0,12	0,10	0,09	0,08	0,08	0,08	0,07	0,07	0,07
8nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,14	0,12	0,11	0,09	0,08	0,07	0,06	0,06	0,06	0,06	0,06	0,05
	m_2	0,16	0,14	0,12	0,11	0,09	0,08	0,08	0,08	0,08	0,07	0,07	0,07

TAB. C.19 – Les points d'intérêt pour New : il y un petit effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.7), méthode RanSaC2.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	3,57	1,14	1	0,57	0,43	0,14	0	0	0	0	0	0
	m_1	0,27	0,20	0,17	0,14	0,12	0,10	0,09	0,08	0,08	0,07	0,07	0,07
	m_2	0,33	0,26	0,22	0,20	0,17	0,14	0,14	0,13	0,12	0,11	0,10	0,10
4nei2	$o_p\%$	4	1,29	1,14	2,86	0,57	0,14	0	0,14	0,14	0	0	0
	m_1	0,24	0,19	0,16	0,15	0,12	0,11	0,10	0,09	0,09	0,08	0,08	0,08
	m_2	0,31	0,26	0,22	0,21	0,18	0,16	0,15	0,15	0,13	0,12	0,11	0,11
8nei	$o_p\%$	6,14	2,14	1,43	1,14	0,43	0,14	0	0	0	0	0	0
	m_1	0,31	0,23	0,19	0,16	0,13	0,11	0,11	0,10	0,09	0,08	0,08	0,08
	m_2	0,38	0,29	0,24	0,21	0,18	0,16	0,15	0,14	0,13	0,12	0,11	0,12
8nei2	$o_p\%$	5,71	1,71	1,43	0,86	0,43	0,43	0,29	0,43	0,14	0	0	0
	m_1	0,25	0,21	0,17	0,15	0,13	0,11	0,10	0,09	0,09	0,08	0,08	0,08
	m_2	0,33	0,28	0,24	0,21	0,19	0,18	0,16	0,14	0,14	0,13	0,12	0,12

TAB. C.20 – Des points généraux pour New : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.7), méthode RanSaC2.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,16	0,12	0,10	0,09	0,09	0,09	0,08	0,08	0,07	0,07	0,07	0,07
	m_2	0,19	0,14	0,13	0,12	0,11	0,11	0,11	0,10	0,11	0,10	0,10	0,10
4nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,15	0,13	0,11	0,10	0,10	0,09	0,09	0,08	0,08	0,08	0,07	0,07
	m_2	0,19	0,15	0,14	0,13	0,12	0,12	0,11	0,11	0,11	0,11	0,11	0,11
8nei	$o_p\%$	0,27	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,21	0,15	0,12	0,11	0,10	0,08	0,09	0,09	0,08	0,08	0,08	0,08
	m_2	0,25	0,17	0,14	0,13	0,12	0,12	0,11	0,11	0,11	0,11	0,11	0,11
8nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,16	0,12	0,11	0,10	0,09	0,09	0,09	0,08	0,08	0,08	0,07	0,07
	m_2	0,19	0,15	0,14	0,13	0,12	0,12	0,11	0,11	0,11	0,11	0,10	0,10

TAB. C.21 – Les points d'intérêt pour New : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.9), méthode RanSaC2.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	6,71	4,14	2,71	2	1,57	2	1,29	1,43	1	1,86	1,43	1,71
	m_1	0,26	0,19	0,16	0,14	0,13	0,12	0,12	0,11	0,11	0,10	0,10	0,10
	m_2	0,34	0,25	0,22	0,20	0,18	0,17	0,18	0,18	0,18	0,17	0,16	0,17
4nei2	$o_p\%$	6,57	3,43	2,29	2,14	1,86	2	1,43	1,43	1	1,86	1,29	1,71
	m_1	0,25	0,20	0,17	0,15	0,14	0,13	0,13	0,12	0,12	0,11	0,11	0,11
	m_2	0,33	0,27	0,24	0,20	0,20	0,18	0,19	0,19	0,19	0,18	0,17	0,17
8nei	$o_p\%$	9	4,14	3,29	2,29	1,29	2	1,43	1,43	1,14	1,71	1,29	1,57
	m_1	0,31	0,21	0,17	0,16	0,15	0,13	0,13	0,12	0,12	0,12	0,12	0,11
	m_2	0,38	0,28	0,23	0,22	0,21	0,18	0,19	0,19	0,18	0,18	0,18	0,18
8nei2	$o_p\%$	7,71	3,86	2,29	1,71	1,71	1,86	1,43	1,57	1	1,71	1,14	1,57
	m_1	0,26	0,20	0,17	0,15	0,13	0,13	0,13	0,12	0,12	0,11	0,11	0,11
	m_2	0,34	0,27	0,23	0,21	0,19	0,18	0,19	0,19	0,19	0,18	0,18	0,18

TAB. C.22 – Des points generaux pour New : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.9), méthode RanSaC2.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0,27	0,27	0,27
	m_1	0,19	0,16	0,14	0,13	0,11	0,11	0,10	0,10	0,09	0,09	0,08	0,08
	m_2	0,23	0,19	0,17	0,16	0,14	0,13	0,13	0,13	0,13	0,11	0,11	0,12
4nei2	$o_p\%$	0	0	0	0	0	0	0	0	0,27	0,27	0,27	0,53
	m_1	0,19	0,16	0,15	0,14	0,12	0,12	0,11	0,11	0,11	0,10	0,10	0,10
	m_2	0,22	0,19	0,18	0,16	0,15	0,15	0,14	0,15	0,13	0,12	0,12	0,12
8nei	$o_p\%$	0,53	0	0	0	0	0	0	0	0	0,27	0,27	0,27
	m_1	0,26	0,20	0,17	0,15	0,13	0,13	0,12	0,11	0,11	0,09	0,09	0,09
	m_2	0,30	0,24	0,20	0,18	0,16	0,16	0,15	0,14	0,14	0,12	0,12	0,12
8nei2	$o_p\%$	0	0	0	0	0	0	0	0	0,27	0,27	0,27	0,53
	m_1	0,20	0,18	0,16	0,14	0,13	0,12	0,12	0,12	0,11	0,10	0,10	0,10
	m_2	0,23	0,21	0,19	0,18	0,16	0,15	0,15	0,15	0,14	0,13	0,12	0,12

TAB. C.23 – Les points d'intérêt pour New : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.10), méthode RanSaC2.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	5,43	3,28	1,86	1,57	0,71	1,29	1,14	1,14	0,86	1	0,29	0,71
	m_1	0,27	0,22	0,19	0,17	0,16	0,15	0,14	0,13	0,12	0,12	0,13	0,12
	m_2	0,33	0,29	0,24	0,22	0,21	0,20	0,19	0,19	0,18	0,19	0,20	0,19
4nei2	$o_p\%$	5,29	3,14	1,86	1	0,57	1	1,71	1,57	1	1	0,29	0,71
	m_1	0,27	0,23	0,20	0,19	0,18	0,17	0,15	0,15	0,14	0,14	0,15	0,14
	m_2	0,33	0,29	0,25	0,24	0,23	0,23	0,20	0,20	0,20	0,20	0,21	0,20
8nei	$o_p\%$	7,72	3,57	2,29	1,71	0,71	1,43	1,43	1,14	0,71	0,71	0,43	0,86
	m_1	0,34	0,26	0,21	0,20	0,19	0,17	0,15	0,15	0,14	0,14	0,14	0,13
	m_2	0,41	0,32	0,26	0,25	0,24	0,23	0,20	0,21	0,20	0,21	0,21	0,20
8nei2	$o_p\%$	6,00	3,71	2	1,86	1	1,57	1,57	1,86	1,14	1,29	0,43	1
	m_1	0,30	0,25	0,21	0,20	0,18	0,18	0,16	0,15	0,15	0,14	0,15	0,14
	m_2	0,36	0,31	0,26	0,25	0,24	0,23	0,22	0,21	0,21	0,21	0,22	0,20

TAB. C.24 – Des points generaux pour New : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.10). méthode RanSaC2,

C.3 Résultats expérimentaux de la corrélation précise après le post traitement

Cette section montre les résultats expérimentaux de la corrélation précise après le post-traitement (section 4.4). Les deux groupes d'images *Bolino* (figure 4.3 à la page 108, figure 4.5 à la page 110 et figure 4.6 à la page 111) et *New* (figure 4.7 à la page 117, figure 4.9 à la page 119 et figure 4.10 à la page 120) sont utilisés. Nous présentons les résultats sur les points d'intérêt et les points aléatoirement choisis affichés à la figure 4.4 et 4.8.

Pour chaque paire, les résultats pour les points d'intérêts et les points aléatoirement choisis (figure 4.8 et 4.4) sont montrés.

Dans chaque tableau, le taux de faux appariements (o_p), la moyenne des erreurs (m_1) et la racine carrée de la moyenne des erreurs au carré (m_2) sont affichées, pour les différentes tailles de la fenêtre de corrélation (qui varient de 7 à 19 pour les images *Bolino* et de 7 à 29 pour les images *New*).

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	0,07	0,00	0,00	0,00	0,00	0,00	0,00
	m_1	0,08	0,06	0,06	0,06	0,05	0,05	0,05
	m_2	0,09	0,07	0,07	0,07	0,06	0,06	0,07
4nei2	$o_p\%$	0,07	0,00	0,00	0,00	0,00	0,00	0,00
	m_1	0,06	0,05	0,06	0,06	0,06	0,05	0,06
	m_2	0,08	0,06	0,07	0,07	0,07	0,06	0,07
8nei	$o_p\%$	0,07	0,00	0,00	0,00	0,00	0,00	0,00
	m_1	0,08	0,06	0,06	0,06	0,06	0,06	0,05
	m_2	0,10	0,08	0,07	0,07	0,07	0,07	0,07
8nei2	$o_p\%$	0,07	0,00	0,00	0,00	0,00	0,00	0,00
	m_1	0,06	0,05	0,06	0,06	0,05	0,05	0,05
	m_2	0,07	0,06	0,07	0,07	0,06	0,06	0,06

TAB. C.25 – Les points d'intérêt pour la paire Bolino1 et Bolino2 (figure 4.3) : le déplacement entre les deux images est essentiellement une translation, résultats après le post traitement.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	3,14	1,43	0,86	0,72	0,36	0,27	0,07
	m_1	0,12	0,07	0,07	0,06	0,06	0,06	0,06
	m_2	0,18	0,11	0,10	0,08	0,08	0,08	0,08
4nei2	$o_p\%$	3,14	1,43	0,93	0,72	0,36	0,29	0,07
	m_1	0,15	0,10	0,07	0,06	0,06	0,06	0,06
	m_2	0,21	0,13	0,10	0,08	0,08	0,08	0,08
8nei	$o_p\%$	3,93	1,50	0,86	0,72	0,36	0,29	0,07
	m_1	0,18	0,11	0,08	0,07	0,07	0,07	0,07
	m_2	0,24	0,15	0,11	0,09	0,09	0,09	0,09
8nei2	$o_p\%$	3,14	1,50	0,86	0,72	0,36	0,29	0,07
	m_1	0,14	0,08	0,07	0,06	0,05	0,06	0,06
	m_2	0,21	0,12	0,10	0,08	0,07	0,08	0,08

TAB. C.26 – Des points généraux pour la paire Bolino1 et Bolino2 (figure 4.3) : le déplacement entre les deux images est essentiellement une translation, résultats après le post traitement.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	1,23	1,16	1,09	0,82	0,69	0,27	0,96
	m_1	0,08	0,06	0,05	0,05	0,05	0,06	0,07
	m_2	0,10	0,08	0,07	0,07	0,07	0,08	0,09
4nei2	$o_p\%$	1,37	1,16	0,89	0,89	0,89	0,55	0,75
	m_1	0,06	0,06	0,05	0,06	0,06	0,05	0,06
	m_2	0,08	0,07	0,07	0,08	0,07	0,08	0,08
8nei	$o_p\%$	1,23	1,02	0,97	0,89	0,82	0,62	0,62
	m_1	0,08	0,07	0,06	0,06	0,05	0,06	0,06
	m_2	0,11	0,09	0,08	0,07	0,07	0,08	0,08
8nei2	$o_p\%$	1,30	1,16	1,09	0,82	0,69	0,55	0,89
	m_1	0,07	0,06	0,05	0,05	0,05	0,05	0,05
	m_2	0,09	0,07	0,06	0,07	0,07	0,07	0,07

TAB. C.27 – Les points d'intérêt pour la paire Bolino1 et Bolino3 (figure 4.5) : le déplacement entre les deux images est essentiellement une translation, résultats après le post traitement.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	5,93	3,43	2,50	1,93	2,07	1,50	0,79
	m_1	0,14	0,09	0,08	0,07	0,06	0,06	0,07
	m_2	0,20	0,15	0,12	0,10	0,09	0,09	0,10
4nei2	$o_p\%$	6,57	3,57	2,50	2,15	2,00	1,86	1,58
	m_1	0,15	0,11	0,08	0,07	0,07	0,06	0,07
	m_2	0,22	0,17	0,12	0,10	0,09	0,09	0,09
8nei	$o_p\%$	6,57	4,07	2,50	2,00	1,93	1,79	1,36
	m_1	0,19	0,11	0,09	0,07	0,07	0,08	0,08
	m_2	0,24	0,16	0,13	0,10	0,10	0,10	0,10
8nei2	$o_p\%$	5,79	3,50	2,50	1,86	1,86	1,36	0,79
	m_1	0,14	0,11	0,08	0,07	0,06	0,06	0,06
	m_2	0,22	0,17	0,12	0,10	0,09	0,09	0,09

TAB. C.28 – Des points généraux pour la paire Bolino1 et Bolino3 (figure 4.5) : le déplacement entre les deux images est essentiellement une translation, résultats après le post traitement.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	3,83	3,48	3,08	3,08	2,81	2,54	2,33
	m_1	0,10	0,07	0,06	0,06	0,06	0,06	0,06
	m_2	0,12	0,09	0,09	0,08	0,09	0,09	0,09
4nei2	$o_p\%$	3,69	3,48	3,14	3,14	2,88	2,61	2,61
	m_1	0,09	0,07	0,06	0,06	0,06	0,06	0,06
	m_2	0,11	0,09	0,08	0,08	0,09	0,09	0,09
8nei	$o_p\%$	3,76	3,55	3,14	3,14	2,74	2,54	2,26
	m_1	0,11	0,08	0,07	0,07	0,06	0,07	0,06
	m_2	0,13	0,10	0,09	0,09	0,09	0,09	0,10
8nei2	$o_p\%$	3,42	3,42	3,08	3,08	2,74	2,40	2,26
	m_1	0,08	0,07	0,06	0,06	0,06	0,06	0,06
	m_2	0,11	0,08	0,08	0,08	0,08	0,08	0,08

TAB. C.29 – Les points d'intérêt pour la paire Bolino1 et Bolino4 (figure 4.6) : le déplacement entre les deux images est essentiellement une translation, résultats après le post traitement.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19
4nei	$o_p\%$	7,43	5,71	4,15	3,65	3,43	3,15	2,80
	m_1	0,15	0,10	0,09	0,08	0,08	0,08	0,08
	m_2	0,22	0,14	0,13	0,11	0,11	0,11	0,11
4nei2	$o_p\%$	8,07	5,29	4,15	3,58	3,51	3,22	2,65
	m_1	0,16	0,10	0,09	0,08	0,07	0,07	0,08
	m_2	0,22	0,15	0,13	0,11	0,11	0,10	0,11
8nei	$o_p\%$	9,07	5,57	4,15	3,15	3,43	3,15	2,37
	m_1	0,18	0,11	0,10	0,08	0,08	0,09	0,10
	m_2	0,25	0,16	0,14	0,12	0,11	0,12	0,13
8nei2	$o_p\%$	8,29	5,50	4,00	3,51	3,43	3,01	2,58
	m_1	0,16	0,10	0,09	0,07	0,07	0,07	0,07
	m_2	0,24	0,16	0,13	0,11	0,10	0,10	0,10

TAB. C.30 – Des points généraux pour la paire Bolino1 et Bolino4 (figure 4.6) : le déplacement entre les deux images est essentiellement une translation, résultats après le post traitement.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,11	0,11	0,09	0,08	0,06	0,06	0,05	0,04	0,04	0,04	0,04	0,04
	m_2	0,13	0,13	0,11	0,10	0,08	0,07	0,06	0,05	0,06	0,06	0,05	0,05
4nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,10	0,10	0,09	0,09	0,07	0,06	0,06	0,06	0,06	0,06	0,06	0,05
	m_2	0,13	0,11	0,11	0,11	0,08	0,08	0,07	0,07	0,07	0,08	0,07	0,06
8nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,15	0,11	0,09	0,08	0,07	0,06	0,05	0,05	0,05	0,05	0,05	0,05
	m_2	0,19	0,14	0,11	0,10	0,08	0,07	0,06	0,06	0,06	0,06	0,06	0,07
8nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,10	0,09	0,09	0,08	0,07	0,06	0,05	0,06	0,05	0,05	0,05	0,05
	m_2	0,13	0,11	0,11	0,09	0,09	0,07	0,06	0,07	0,06	0,06	0,07	0,06

TAB. C.31 – Les points d'intérêt pour New : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.7), résultats après le post traitement.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	3,29	1,14	0,43	0,29	0,43	0,14	0	0	0	0	0	0
	m_1	0,26	0,15	0,12	0,11	0,09	0,08	0,07	0,07	0,06	0,05	0,05	0,06
	m_2	0,31	0,19	0,17	0,16	0,13	0,11	0,12	0,11	0,10	0,09	0,09	0,11
4nei2	$o_p\%$	2,29	0,57	0,43	0,29	0,43	0,14	0	0,14	0	0	0	0
	m_1	0,22	0,15	0,12	0,10	0,09	0,09	0,08	0,09	0,08	0,07	0,07	0,07
	m_2	0,28	0,20	0,18	0,14	0,13	0,13	0,12	0,13	0,12	0,10	0,10	0,09
8nei	$o_p\%$	2,86	1,00	0,71	0,57	0,43	0,14	0	0	0	0	0	0
	m_1	0,22	0,16	0,15	0,12	0,10	0,09	0,08	0,08	0,08	0,07	0,07	0,07
	m_2	0,30	0,21	0,19	0,18	0,14	0,12	0,12	0,12	0,11	0,10	0,10	0,10
8nei2	$o_p\%$	2,86	1,29	0,57	0,57	0,14	0,29	0,29	0,29	0,14	0	0	0
	m_1	0,20	0,13	0,13	0,11	0,10	0,09	0,09	0,08	0,07	0,06	0,07	0,06
	m_2	0,27	0,19	0,19	0,16	0,15	0,13	0,12	0,12	0,11	0,09	0,11	0,09

TAB. C.32 – Des points généraux pour New : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.7), résultats après le post traitement.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,12	0,11	0,10	0,08	0,08	0,08	0,06	0,06	0,06	0,06	0,06	0,05
	m_2	0,15	0,13	0,12	0,10	0,10	0,10	0,09	0,08	0,09	0,09	0,09	0,09
4nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,13	0,13	0,09	0,08	0,09	0,08	0,08	0,07	0,07	0,06	0,06	0,06
	m_2	0,15	0,15	0,11	0,10	0,11	0,10	0,10	0,09	0,09	0,09	0,09	0,08
8nei	$o_p\%$	0,27	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,16	0,12	0,09	0,10	0,10	0,09	0,07	0,06	0,06	0,07	0,06	0,06
	m_2	0,21	0,15	0,11	0,11	0,11	0,11	0,09	0,09	0,07	0,08	0,08	0,08
8nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,12	0,10	0,10	0,09	0,08	0,07	0,06	0,06	0,06	0,06	0,07	0,05
	m_2	0,15	0,12	0,12	0,11	0,10	0,09	0,09	0,09	0,08	0,09	0,09	0,07

TAB. C.33 – Les points d'intérêt pour New : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.9), résultats après le post traitement.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	4,29	2,86	1,86	1,14	0,86	1,57	1,43	1,00	1,43	1,57	2,29	1,71
	m_1	0,21	0,17	0,14	0,11	0,10	0,10	0,10	0,09	0,09	0,09	0,09	0,08
	m_2	0,28	0,22	0,19	0,16	0,15	0,16	0,16	0,15	0,15	0,15	0,15	0,16
4nei2	$o_p\%$	4,00	2,43	1,57	1,43	1,43	1,00	1,43	0,86	1,29	1,43	1,86	1,71
	m_1	0,22	0,19	0,13	0,12	0,11	0,11	0,11	0,10	0,11	0,10	0,09	0,09
	m_2	0,28	0,24	0,18	0,16	0,16	0,15	0,17	0,15	0,18	0,15	0,15	0,15
8nei	$o_p\%$	5,43	2,43	2,00	1,57	1,00	0,86	1,14	0,86	1,29	1,00	1,86	1,57
	m_1	0,26	0,17	0,14	0,13	0,11	0,10	0,09	0,11	0,09	0,09	0,08	0,12
	m_2	0,33	0,23	0,19	0,19	0,15	0,14	0,15	0,16	0,15	0,15	0,14	0,16
8nei2	$o_p\%$	4,14	2,86	1,43	1,29	1,29	1,00	1,00	1,29	1,14	1,43	1,86	2,00
	m_1	0,22	0,15	0,13	0,12	0,11	0,11	0,10	0,10	0,09	0,10	0,09	0,09
	m_2	0,30	0,21	0,19	0,18	0,16	0,16	0,16	0,15	0,16	0,16	0,16	0,14

TAB. C.34 – Des points généraux pour New : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.9) , résultats après le post traitement.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,14	0,13	0,10	0,10	0,08	0,09	0,07	0,08	0,06	0,07	0,06	0,08
	m_2	0,18	0,16	0,14	0,13	0,10	0,12	0,09	0,10	0,09	0,09	0,09	0,10
4nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0,27
	m_1	0,15	0,13	0,13	0,10	0,09	0,09	0,09	0,09	0,07	0,07	0,08	0,07
	m_2	0,18	0,16	0,15	0,13	0,11	0,11	0,11	0,11	0,10	0,10	0,11	0,10
8nei	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0,27	0,27
	m_1	0,20	0,14	0,13	0,12	0,11	0,10	0,08	0,08	0,09	0,07	0,06	0,05
	m_2	0,25	0,18	0,16	0,15	0,13	0,12	0,10	0,11	0,11	0,10	0,09	0,08
8nei2	$o_p\%$	0	0	0	0	0	0	0	0	0	0	0	0
	m_1	0,20	0,16	0,14	0,17	0,13	0,10	0,10	0,09	0,13	0,10	0,10	0,08
	m_2	0,24	0,19	0,18	0,21	0,15	0,13	0,13	0,12	0,16	0,12	0,13	0,10

TAB. C.35 – Les points d'intérêt pour New : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.10) , résultats après le post traitement.

		7×7	9×9	11×11	13×13	15×15	17×17	19×19	21×21	23×23	25×25	27×27	29×29
4nei	$o_p\%$	2,15	2,00	0,71	0,29	0,86	0,29	0,00	0,29	1,14	1,00	0,00	0,29
	m_1	0,22	0,15	0,13	0,12	0,11	0,09	0,09	0,08	0,10	0,09	0,08	0,10
	m_2	0,27	0,22	0,18	0,16	0,15	0,13	0,13	0,13	0,16	0,17	0,14	0,15
4nei2	$o_p\%$	2,58	1,43	1,00	0,86	0,71	0,14	0,14	0,29	0,00	0,00	0,00	0,00
	m_1	0,23	0,18	0,14	0,15	0,13	0,14	0,11	0,10	0,10	0,09	0,10	0,09
	m_2	0,29	0,24	0,19	0,19	0,18	0,18	0,15	0,15	0,14	0,13	0,16	0,13
8nei	$o_p\%$	5,44	2,14	1,14	0,86	0,29	0,29	0,43	0,00	0,14	1,00	0,86	0,86
	m_1	0,25	0,21	0,19	0,17	0,15	0,12	0,12	0,09	0,10	0,10	0,09	0,09
	m_2	0,33	0,26	0,23	0,21	0,19	0,17	0,17	0,14	0,15	0,17	0,17	0,17
8nei2	$o_p\%$	2,00	2,57	1,00	0,29	0,43	0,14	0,14	0,00	0,00	0,00	0,00	0,00
	m_1	0,26	0,22	0,16	0,14	0,14	0,14	0,12	0,10	0,10	0,09	0,09	0,10
	m_2	0,32	0,26	0,21	0,19	0,19	0,18	0,16	0,15	0,15	0,13	0,13	0,14

TAB. C.36 – Des points généraux pour New : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.10) , résultats après le post traitement.

Liste des tableaux

2.1	Définition des mesures de ressemblance. $R(u, v)$ définit le modèle de niveau de gris, $S(x, y)$ définit un signal de niveau de gris dans une fenêtre contenant dans la région d'intérêt (RDI, une partie d'image) d'une autre image, $ulen + 1$ et $vlen + 1$ sont le hauteur et le largeur de R et S , $n = (vlen + 1)(ulen + 1)$	44
2.2	Invariance des corrélations classiques par la transformation de signal: I: invariant, V: variant.	47
3.1	Le premier exemple sur l'image <i>benetton</i>	80
3.2	Le deuxième exemple sur l'image <i>benetton</i>	80
3.3	Le premier exemple sur l'image <i>world</i>	80
3.4	Le deuxième exemple sur l'image <i>world</i>	80
3.5	Exemple pour le cas général sur l'image <i>Benetton</i>	81
3.6	Exemple pour le cas général sur l'image <i>world</i>	81
3.7	Le résultat de MVE pour le test du tableau 3.1.	93
3.8	Le résultat de MVE pour le test du tableau 3.2.	93
3.9	Le résultat de MVE pour le premier test du tableau 3.3.	94
3.10	Le résultat de MVE pour le deuxième test du tableau 3.4.	94
3.11	Le résultat de MVE pour le test du tableau 3.5.	94
3.12	Le résultat de MVE pour le test du tableau 3.6.	94
C.1	Résultats pour les points d'intérêt de la paire <i>Bolino1</i> et <i>Bolino2</i> (figure 4.3): le déplacement entre ces deux images est essentiellement une translation.	142

C.2	Résultats pour des points généraux de la paire <i>Bolino1</i> et <i>Bolino2</i> (figure 4.3) : le déplacement entre ces deux images est essentiellement une translation.	142
C.3	Résultats pour les points d'intérêt pour la paire <i>Bolino1</i> et <i>Bolino3</i> (figure 4.5) : le déplacement entre ces deux images est essentiellement une translation.	143
C.4	Résultats pour des points généraux pour la paire <i>Bolino1</i> et <i>Bolino3</i> (figure 4.5) : le déplacement entre ces deux images est essentiellement une translation.	143
C.5	Résultats pour les points d'intérêt pour la paire <i>Bolino1</i> et <i>Bolino4</i> (figure 4.6) : le déplacement entre ces deux images est essentiellement une translation.	144
C.6	Résultats pour des points généraux pour la paire <i>Bolino1</i> et <i>Bolino4</i> (figure 4.6) : le déplacement entre ces deux images est essentiellement une translation.	144
C.7	Les points d'intérêt pour <i>New</i> : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.7).	146
C.8	Des points généraux pour <i>New</i> : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.7).	146
C.9	Les points d'intérêt pour <i>New</i> : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.9).	146
C.10	Des points généraux pour <i>New</i> : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.9).	147
C.11	Les points d'intérêt pour <i>New</i> : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images. (figure 4.10).	147
C.12	Des points généraux pour <i>New</i> : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.10).	148
C.13	Les points d'intérêt pour <i>New</i> : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'image (figure 4.7), méthode RanSaC.	148

C.14 Des points généraux pour <i>New</i> : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.7), méthode RanSaC.	149
C.15 Les points d'intérêt pour <i>New</i> : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.9), méthode RanSaC.	149
C.16 Des points généraux pour <i>New</i> : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.9) , méthode RanSaC.	150
C.17 Les points d'intérêt pour <i>New</i> : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.10) , méthode RanSaC.	150
C.18 Des points généraux pour <i>New</i> : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.10) , méthode RanSaC.	150
C.19 Les points d'intérêt pour <i>New</i> : il y un petit effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.7), méthode RanSaC2.	151
C.20 Des points généraux pour <i>New</i> : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.7), méthode RanSaC2.	151
C.21 Les points d'intérêt pour <i>New</i> : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.9), méthode RanSaC2.	152
C.22 Des points généraux pour <i>New</i> : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.9), méthode RanSaC2.	152
C.23 Les points d'intérêt pour <i>New</i> : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.10), méthode RanSaC2.	153
C.24 Des points généraux pour <i>New</i> : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux fenêtres d'images (figure 4.10). méthode RanSaC2,	153
C.25 Les points d'intérêt pour la paire <i>Bolino1</i> et <i>Bolino2</i> (figure 4.3): le déplacement entre les deux images est essentiellement une translation, résultats après le post traitement.	154

C.26 Des points généraux pour la paire <i>Bolino1</i> et <i>Bolino2</i> (figure 4.3) : le déplacement entre les deux images est essentiellement une trans- lation, résultats après le post traitement.	155
C.27 Les points d'intérêt pour la paire <i>Bolino1</i> et <i>Bolino3</i> (figure 4.5) : le déplacement entre les deux images est essentiellement une trans- lation, résultats après le post traitement.	155
C.28 Des points généraux pour la paire <i>Bolino1</i> et <i>Bolino3</i> (figure 4.5) : le déplacement entre les deux images est essentiellement une trans- lation, résultats après le post traitement.	156
C.29 Les points d'intérêt pour la paire <i>Bolino1</i> et <i>Bolino4</i> (figure 4.6) : le déplacement entre les deux images est essentiellement une trans- lation, résultats après le post traitement.	156
C.30 Des points généraux pour la paire <i>Bolino1</i> et <i>Bolino4</i> (figure 4.6) : le déplacement entre les deux images est essentiellement une trans- lation, résultats après le post traitement.	157
C.31 Les points d'intérêt pour <i>New</i> : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.7), résultats après le post traitement.	157
C.32 Des points généraux pour <i>New</i> : il y a un petit effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.7), résultats après le post traitement.	158
C.33 Les points d'intérêt pour <i>New</i> : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.9), résultats après le post traitement.	158
C.34 Des points généraux pour <i>New</i> : il y a une rotation (proche de 20 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.9) , résultats après le post traitement.	159
C.35 Les points d'intérêt pour <i>New</i> : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.10) , résultats après le post traitement.	159
C.36 Des points généraux pour <i>New</i> : il y a une rotation (proche de 30 degrés) à part d'un effet d'échelle (proche de 1 : 0,9) entre les deux images (figure 4.10) , résultats après le post traitement.	160

Table des figures

1.1	(a) Données originales avec 5 points et la droite obtenue par régression des moindres carrés. (b) Mêmes données mais avec une donnée aberrante dans la direction y	16
1.2	(a): Les points dont un aberrant; (b): La droite obtenue par estimation aux moindres carrés; (c): Les coordonnées du point d'accumulation (5, 100) donnent la droite ajustée aux points, $y = 5x + 100$. Pour ce point, la valeur de l'accumulation est 10. . . .	18
1.3	(a) Données originales avec cinq points et la droite obtenue par régression des moindres carrés médians. (b) Mêmes données mais avec une erreur grossière dans la direction y	25
1.4	Si les données sont uniformément distribuées de Z_{min} à Z_{max} , alors la probabilité que plus de k points se trouvent dans $\phi \pm r$ est donnée par la somme binomiale.	30
2.1	Définitions pour la corrélation.	43
2.2	Sens géométrique de SSD et NCC.	45
2.3	L'espace d'appariement.	62
3.1	Occultation. Gauche: I_1 , droite: I_2	73
3.2	Régression robuste entre I_1 and I_2 , les petits cercles sont des couples de points non appariés.	74
3.3	Paolo. Deux tests d'appariements: l'un sur le doigt et l'autre sur l'oreille. Les petits carrés dans l'image à gauche sont des modèles.	76
3.4	En blanc sont les parties non appariées pour les deux exemples de la figure 3.3, l'image (a) correspond au doigt où retrouve le bord du cube occultant, l'image (b) correspond à l'oreille.	76

3.5	La paire stéréo 1: <i>benetton</i>	77
3.6	La paire stéréo 2: <i>world</i>	78
3.7	Fond <i>benetton</i> . Points choisis (noirs) dans la Figure 3.5 (a) pour test 1 et 2. Gauche: le test 1 correspondant au tableau 3.1; Droite: le test 2 correspondant au tableau 3.2	78
3.8	Fond <i>world</i> . Points choisis (noirs) dans la Figure 3.6 (a) pour test 3 et 4. Gauche: le test 3 correspondant au tableau 3.3; Droite: le test 4 correspondant au tableau 3.4	79
3.9	La translation du signal. La partie à gauche est trouvée fausse et rejetée et la partie droite appariée très bien. S'il n'y a pas de translation, nous ne trouverons pas de fausse partie et les fenêtres entières s'apparient moins bien qu'après la translation.	81
3.10	Contours d'occultations à partir des techniques de corrélation différentes. (a): zssd; (b): rang.	83
3.11	Contours d'occultations à partir des techniques de corrélation différentes. (a): census; (b): rzssdp.	83
3.12	Localisation robuste par l'estimateur MVE de (I_1, I_2) , les points dehors de l'ellipse représentent des couples de points non appariés.	93
4.1	Vecteur A dans le cas de quatre voisinage.	103
4.2	Vecteur A dans le cas de huit voisinage.	104
4.3	La paire d'images Bolino1 et Bolino2.	108
4.4	Les points d'intérêt et des points généraux choisis dans l'image gauche <i>Bolino</i> de la figure 4.3.	109
4.5	Image Bolino1 et Bolino3	110
4.6	Image Bolino1 et Bolino4	111
4.7	La paire d'image sous un effet d'échelle 1 : 0,9.	117
4.8	Les points d'intérêt et des points généraux choisis dans l'image gauche de la figure 4.7.	118
4.9	Paire d'image New sous une rotation proche de 20 degrés, à part d'un effet d'échelle de 1 : 0,9.	119
4.10	Paire d'image New sous une rotation de 30 degrés, à part d'un effet d'échelle de 1 : 0,9.	120

4.11 La paire d'images d'objet L	123
4.12 Les points d'intérêt dans L	123
4.13 Les points tridimensionnels (représentés par des petits carrés) construits à partir des points bidimensionnels appariés selon notre méthode, projetés dans un plan horizontal.	124
4.14 Les points tridimensionnels construits (représentés par des tri- angles) à partir des points bidimensionnels appariés selon la mé- thode classique, projetés dans un plan horizontal.	125

Bibliographie

- [Ack84] F. Ackermann. Digital image correlation: performance and potential application in photogrammetry. *Photogrammetric Record*, 64(11):429–439, October 1984.
- [AG92] P. Aschwenden and W. Guggenbühl. Experimental results from a comparative study on correlation-type registration algorithms. In Förstner and Ruwiedel, editors, *Robust Computer Vision*, pages 268–282. Wichmann, 1992.
- [Aya90] N. Ayache. *Stereovision and Sensor Fusion*. MIT-Press, 1990.
- [BBW88] P.J. Besl, J.B. Birch, and L.T. Watson. Robust window operators. In *Proceedings of the 2nd International Conference on Computer Vision, Tampa, Florida, USA*, pages 591–600, 1988.
- [BLM96] J. Blanc, S. Livatino, and R. Mohr. Fast and realistic image synthesis for telemanipulation purposes. In *European Workshop on Hazardous Robotics 1996 – Barcelona, Spain.*, pages 77–83, November 1996.
- [BM95a] B. Boufama and R. Mohr. Epipole and fundamental matrix estimation using the virtual parallax property. In *Proceedings of the 5th International Conference on Computer Vision, Cambridge, Massachusetts, USA*, pages 1030–1036, June 1995.
- [BM95b] R. Brunelli and S. Messelodi. Robust estimation of correlation with applications to computer vision. *Pattern Recognition*, 28(6):833–841, 1995.
- [BN96] D. N. Bhat and S. K. Nayar. Ordinal measure for visual correspondence. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, San Francisco, California, USA*, pages 351 – 357, San Francisco, California, June 1996.
- [BR94] M. J. Black and A. Rangarajan. The outlier processus: Unifying processes and robust statistics. In *Proceedings of the Conference*

- on Computer Vision and Pattern Recognition, Seattle, Washington, USA*, pages 15–22, 1994.
- [Bra95] P. Brand. *Reconstruction Tridimensionnelle d'une Scène à partir d'une Caméra en Mouvement: de l'Influence de la Précision*. PhD thesis, Université Claude Bernard, Lyon I, October 1995.
- [BWW94] B. Boufama, D. Weinshall, and M. Werman. Shape from motion algorithms: a comparative analysis of scaled orthography and perspective. In J.O. Eklundh, editor, *Proceedings of the 3rd European Conference on Computer Vision, Stockholm, Sweden*, pages 199–204. Springer-Verlag, May 1994.
- [BZ87] A. Blake and A. Zisserman. *Visual Reconstruction*. The MIT Press, Cambridge, Massachusetts, 1987.
- [Can86] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
- [Fau92] O. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig? In G. Sandini, editor, *Proceedings of the 2nd European Conference on Computer Vision, Santa Margherita Ligure, Italy*, pages 563–578. Springer-Verlag, May 1992.
- [Fau93] O. Faugeras. *Three-Dimensional Computer Vision - A Geometric Viewpoint*. Artificial intelligence. M.I.T. Press, Cambridge, MA, 1993.
- [FB80] M.A. Fischler and R.C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis. Technical report, SRI International, Menlo Park, CA, 1980.
- [FB81] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Graphics and Image Processing*, 24(6):381 – 395, June 1981.
- [Fel86] W. Feller. *An Introduction to Probability Theory and Its Applications*. New York: Wiley, New York, 1986.
- [FFH⁺92] O. Faugeras, P. Fua, B. Hotz, R. Ma, L. Robert, M. Thonnat, and Z. Zhang. Quantitative and qualitative comparisons of some area and feature-based stereo algorithms. In W. Forstner, R. Haralick, and B. Radig, editors, *Robust Computer Vision*, pages 1–26, Inria Sophia-Antipolis, 2004 Route des Lucioles, 06565 Valbonne Cedex, m2 1992. Comparisons of area and feature-based stereo algorithms.

- [Flo93] L. Florack. *The Syntactical Structure of Scalar Images*. PhD thesis, Universiteit Utrecht, m11 1993.
- [For82] W. Forstner. Mathematical models, accuracy aspects and quality control. *International Archives of Photogrammetry*, 24(3):176–189, June 1982.
- [For84] W. Forstner. Quality assessment of object location and point transfer using digital image correlation techniques. *International society for photogrammetry and remote sensing*, pages 1–23, 1984.
- [FT87] O.D. Faugeras and G. Toscani. Camera calibration for 3D computer vision. In *Proceedings of International Workshop on Machine Vision and Machine Intelligence, Tokyo, Japan*, 1987.
- [GG91] D. Geiger and F. Girosi. Parallel and deterministic algorithms for mrfs: surface reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):401–412, May 1991.
- [GH87] N. M. Grzywacz and E. C. Hildreth. Incremental rigidity scheme for recovering structure from motion: Position-based versus velocity-based formulations. *J. Opt. Soc. Am. A*, 4:503–518, March 1987.
- [GLY92] D. Geiger, B. Ladendorf, and A. Yuille. Occlusions and binocular stereo. In G. Sandini, editor, *Proceedings of the 2nd European Conference on Computer Vision, Santa Margherita Ligure, Italy*, pages 425–433. Springer Verlag, 1992.
- [GQ92] P. Gros and L. Quan. Projective invariants for vision. Technical Report RT 90 IMAG - 15 LIFIA, LIFIA–IRIMAG, Grenoble, France, December 1992.
- [Gru85] A.W. Gruen. Adaptive least squares correlation: a powerful image matching technique. *S. Afr. Journal of Photogrammetry, Remote Sensing and Cartography*, 14(3):175–187, 1985.
- [GS91] A. Gruen and D. Stallmann. High accuracy edge matching with an extension of the MPGC-matching algorithm. In *Industrial vision metrology, Winnipeg*, pages 42–55. SPIE 1526, 1991.
- [HA89] W. Hoff and N. Ahuja. Surfaces from stereo: Integrating feature matching, disparity estimating, and contour detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(2):121–136, February 1989.
- [Har86] R. M. Haralick. Computer vision theory: The lack thereof. *Computer Vision, Graphics and Image Processing*, 36:372–386, 1986.

- [Har95] R. Hartley. In defence of the 8-point algorithm. In *Proceedings of the 5th International Conference on Computer Vision, Cambridge, Massachusetts, USA*, pages 1064–1070, June 1995.
- [HB95] M. Hahn and C. Brenner. Area based matching of colour images. In *From Image Sequence to Virtual Reality*, pages 227–236, 1995.
- [Hei81] S. Heiler. Robust estimates in linear regression—a simulation approach. *Computational Statistics*, pages 115–136, 1981.
- [HGC92] R. Hartley, R. Gupta, and T. Chang. Stereo from uncalibrated cameras. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, Urbana-Champaign, Illinois, USA*, pages 761–764, 1992.
- [HJ95] D.P. Huttenlocher and E.W. Jaquith. Computing visual correspondence: Incorporating the probability of a false match. In *Proceedings of the 5th International Conference on Computer Vision, Cambridge, Massachusetts, USA*, pages 515–522, 1995.
- [HJL⁺89] R. M. Haralick, H. Joo, C. Lee, X. Zhuang, V.G. Vaidya, and M. B. Kim. Pose estimation from corresponding point data. *IEEE Transactions on Systems, Man and Cybernetics*, 6(19):1426–1446, November/December 1989.
- [HM93] R. Horaud and O. Monga. *Vision par ordinateur: outils fondamentaux*. Editions Hermès, Paris, 1993.
- [HN94] T.S. Huang and A.N. Netravali. Motion and structure from feature correspondences: A review. *Proceedings of the IEEE*, 82(2):252–268, February 1994.
- [Hou62] P. V. C. Hough. A method and means for recognition complex patterns. *U. S. Patent*, 1962.
- [HR93] D.P. Huttenlocher and W.J. Rucklidge. A multi-resolution technique for comparing images using the Hausdorff distance. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, New York, USA*, pages 705–706, 1993.
- [HRRS86] F.R. Hampel, E.M. Ronchetti, P.J. Rousseeuw, and W.A. Stahel. *Robust Statistics: the Approach Based on Influence Functions*. Wiley series in probability and mathematical. John Wiley and Sons, New York, 1986.
- [HS80] B. Horn and B. Schunk. Determining Optical Flow. Ai-memo 572, MIT, 1980.

- [Hub81] P.J. Huber. *Robust Statistics*, volume IX of *Wiley*. John Wiley, New York, 1981.
- [IB94] S.S. Intille and A.F. Bobick. Disparity-space images and large occlusion stereo. In *Proceedings of the 3rd European Conference on Computer Vision, Stockholm, Sweden*, pages 179–186. Springer-Verlag, 1994.
- [IK88] J. Illingworth and J. Kittler. A survey of the Hough transform. *Computer Vision, Graphics and Image Processing*, 44:87–116, 1988.
- [JMB91] J.M. Jolion, P. Meer, and S. Bataouche. Robust clustering with applications in computer vision. *PAMI*, 13(8):791–802, August 1991.
- [KBS75] C. Kimme, D. H. Ballard, and J. Sklansky. Find circles by an array of accumulators. *CACM*, 18:120–122, 1975.
- [KvD87] J. J. Konderink and A. J. van Doorn. Representation of local geometry in the visual system. *Biological Cybernetics*, 55:367–375, 1987.
- [LF93] Q.T. Luong and O.D. Faugeras. Determining the fundamental matrix with planes: Instability and new algorithms. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, New York, USA*, pages 489–494, June 1993.
- [LG94] J.L. Lotti and G. Giraudon. Adaptive window algorithm for aerial image stereo. In *Proceedings of the 12th International Conference on Pattern Recognition, Jerusalem, Israel*, pages 701–703, October 1994.
- [LI96] S. Z. LI. Robustizing robust m-estimation using deterministe annealing. *Pattern Recognition*, 29(1):159–166, January 1996.
- [LM96] Z.D. Lan and R. Mohr. Precise matching by robust estimation of deformation and local coherence, 1996. Submitted to Cvpr97.
- [LM97] Z.D. Lan and R. Mohr. Robust location based partial correlation. Inria Rhone Alpes, January 1997. accepted by CAIP97.
- [LMP96] K-M. Lee, P. Meer, and R-H. Park. Robust adaptive segmentation of range images. Technical report, Sogang University, Korea and Rutgers University, Piscataway, June 1996.
- [LMR95] Z. D. Lan, R. Mohr, and P. Remagnino. Robust matching by partial correlation. In *Proceedings of the sixth British Machine Vision Conference, Birmingham, England*, pages 651–660, September 1995.

- [MA90] R. Mohr and E. Arbogast. It can be done without camera calibration. Technical Report RR 805-I- IMAG 106 LIFIA, LIFIA-IMAG, February 1990.
- [MGB63] A. M. Mood, F. A. Graybill, and D. C. Boes. *Introuduction to the Theory of Statistics*. McGRAW-HILL, 1963. LM12402 in media-theque.
- [Min94] D. Mintz. Robust consensus based edge detection. *Computer Vision, Graphics and Image Processing: Image Understanding*, 59(2):137–153, March 1994.
- [MKM⁺92] J. Mundy, D. Kapur, S. Maybank, P. Gros, and L. Quan. Geometric interpretation of joint conic invariants. In J. Mundy and A. Zisserman, editors, *Geometric Invariance in Computer Vision*, pages 77–86. MIT Press, 1992.
- [MMR90] D. Mintz, P. Meer, and A. Rosenfeld. Consensus by decomposition : a paradigm for fast high breakdown point robust estimation. In *Proceedings of DARPA Image Understanding Workshop, Pittsburgh, Pennsylvania, USA*, December 1990.
- [MMRK91] P. Meer, D. Mintz, A. Rosenfeld, and D.Y. Kim. Robust regression methods for computer vision: a review. *International Journal of Computer Vision*, 6(1):59–70, 1991.
- [MNMR90] P. Meer, D. Nintz, A. Montanvert, and A. Rosenfeld. Consensus vision. *AAAI-90*, July 1990.
- [MS96] J. V. Miller and C. V. Stewart. Muse: Robust surface fitting using unbiased scale estimates. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, San Francisco, California, USA*, pages 300 – 306, San Francisco, California, June 1996.
- [Nis84] H.K. Nishihara. PRISM, a pratical real-time imaging stereo matcher. Technical Report Technical Report A.I. Memo 780, Massachusetts Institute of Technology, 1984.
- [NW94] N. S. Netanyahu and I. Weiss. Anylitic outlier removal in line fitting. In *Proceedings of IAPR Workshop on Computer Vision*, pages 406–408, October 1994.
- [OK92] M. Okutomi and T. Kanade. A locally adaptive window for signal matching. *International Journal of Computer Vision*, 7(2):143–162, 1992.

- [RBM94] P. Remagnino, P. Brand, and R. Mohr. Correlation techniques in adaptative template matching with uncalibrated cameras. In *Vision Geometry III, SPIE's international symposium on photonic sensors & control for commercial applications*, volume 2356, pages 252–253, October 1994.
- [RL87] P.J. Rousseeuw and A.M. Leroy. *Robust regression and outlier detection*, volume XIV of *Wiley*. J.Wiley and Sons, New York, 1987.
- [RL93] G. Roth and M. D. Levine. Extracting geometric primitives. *Computer Vision, Graphics and Image Processing*, 58(1):1 – 22, July 1993.
- [Rou84] P. J. Rousseeuw. Least median of squares regression. *Journal of American Statistics Association*, 79:871–880, 1984. The first paper of LMedS.
- [Ruc95] W.J. Rucklidge. Locating objects using the Hausdorff distance. In *Proceedings of the 5th International Conference on Computer Vision, Cambridge, Massachusetts, USA*, pages 457–464, 1995.
- [Sch86] B. G. Schunck. The image flow constraint equation. *Computer Vision, Graphics and Image Processing*, 35:20–46, 1986.
- [Sch89] B. G. Schunck. Image flow segmentation and estimation by constraint line clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:1010–1027, October 1989.
- [Sch90] B. G. Schunck. Robust computational vision. In *Proceedings of the International Workshop on Robust Computer Vision, Seattle, Washington, USA*, pages 1–19, October 1990.
- [Sch96] C. Schmid. *Appariement d'images par invariants locaux de niveaux de gris*. PhD thesis, Institut National Polytechnique de Grenoble, July 1996.
- [Sin90] S. S. Sinha. *Discontinuity Preserving Surface Reconstruction in Vision Processing*. PhD thesis, University of Michigan, Michigan, USA, 1990.
- [SM95] C. Schmid and R. Mohr. Matching by local invariants. Technical report, INRIA, August 1995.
- [Sor70] H. W. Sorensen. Least-squares estimation: from gauss to kalman. *IEEE spectrum*, pages 63–68, June 1970.

- [SQ95] P. Sturm and L. Quan. Affine stereo calibration. In *6th International Conference CAIP'95, Computer Analysis of Images and Patterns*, pages 838–843, September 1995.
- [SS90] S. S. Sinha and B. G. Schunck. A robust method for surface reconstruction. In *Proceedings of the International Workshop on Robust Computer Vision, Seattle, Washington, USA*, pages 183–200, Seattle, WA, USA, October 1990.
- [Ste93] C. V. Stewart. A new robust operator for computer vision: Theoretical analysis. Technical report, Departement of Computer Science, Rensselaer Polytechnic Institute, August 1993.
- [Ter88] D. Terzopoulos. The computation of visible-surface representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):417–438, July 1988.
- [TMV85] J.L. Turney, T. N. Mudge, and R. A. Volz. Recognizing partially occluded parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-7(4):410–421, 1985.
- [TZ96] P.H.S. Torr and A. Zisserman. Robust parameterization and computation of the trifocal tensor. In R.B. Fisher and E. Trucco, editors, *Proceedings of the seventh British Machine Vision Conference, Edinburgh, Scotland*, volume 2, pages 655–664. British Machine Vision Association, September 1996.
- [Ull84] S. Ullman. Maximizing rigidity: The incremental recovery of 3-d structure from rigid and nonrigid motion. *perception*, 13:255–274, 1984.
- [YBK94] X. Yu, T. D. Bui, and A. Krzyzak. Robust estimation for range image segmentation and reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):530–538, May 1994.
- [Zab94] R. Zabih. *Individuating Unknown Objects by Combining Motion and Stereo*. PhD thesis, Stanford University, Departement of Computer Science, 5161 Upson Hall, Cornell University, Ithaca Ny 14853-7501, rdz@cs.cornell.edu, August 1994. non parametric correlation.
- [ZDFL94] Z. Zhang, R. Deriche, O. Faugeras, and Q.T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. Rapport de recherche 2273, INRIA, May 1994.

- [ZFD95] Z. Zhang, O. Faugeras, and R. Deriche. Calibrating a binocular stereo through projective reconstruction using both a calibration object and the environment. In *Europe-China Workshop on Geometrical Modelling and Invariants for Computer Vision, Xian, China*, pages 253–260, April 1995.
- [ZH94] X. Zhuang and Y. Huang. Robust 3-D – 3-D estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8):818–824, August 1994.
- [Zha93] C.S. Zhao. *Reconstruction de surfaces tridimensionnelles en vision par ordinateur*. PhD thesis, Institut National Polytechnique de Grenoble, December 1993.
- [ZW94] R. Zabih and J. Woodfill. Non-parametric local transforms for computing visual correspondance. In *Proceedings of the 3rd European Conference on Computer Vision, Stockholm, Sweden*, pages 151–158. Springer-Verlag, May 1994.
- [ZWZ92] X. Zhuang, T. Wang, and P. Zhang. A highly robust estimator through partially likelihood function modeling and its applications in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(1):19–15, January 1992.

Index

- échantillonnage, 40
- évaluation, 22
 - complexité en temps, 22
 - efficacité relative, 22
 - point de rupture
 - régression, 22
 - seuil de rupture
 - localisation, 85
 - régression, 22
- appariement, 40
- BVM, 135, 136
- calibrage faible, 40
- centralisation, 45
- centre, 85
- contrainte de la fenêtre principale
 - MVE, 91
 - rzssdp, 73
- contrainte des points principaux
 - MVE, 91
- contrainte du point principal
 - rzssdp, 72
- convolution, 98, 99
- corrélation, 42, 65, 66
 - classique, 44
 - Brunelli, 47
 - LSAD, 45
 - LSSD, 45
 - NCC, 45
 - SAD, 45, 68
 - SES1, 46
 - SES2, 46
 - SSD, 45, 68
 - ZNCC, 45, 68, 75, 82
 - ZSAD, 45, 68
 - ZSSD, 45, 67, 75
- non paramétrique, 48
 - Bhat, 51
 - census, 49
 - rang, 49
 - Zabih, 48
- partielle, 65
 - MVE, 92
 - rzssdp, 70, 77
- précise, 97
 - algorithme rapide, 97, 102
 - algorithme robuste, 97, 112
- décalage, 100
- Dirac, 98
- disparité, 40
- dispersion, 82, 85
- distorsion perspective, 41
- droite épipolaire, 40
- estimateurs robustes, 22
 - ALKS, 14, 32
 - CBD, 27
 - Hough, 17
 - L-estimateurs, 23
 - LMedS estimateur, 14, 24, 113
 - M-estimateurs, 14, 23
 - MF, 14, 28
 - MINPRAN, 14, 29
 - MOM, 14, 34
 - MUSE, 14, 33
 - R-estimateurs, 23
 - RanSaC, 19, 114
 - paramètres, 19–21
 - RESC, 14, 31

- estimation robuste, 14
- factionalisme, 48
- fenêtre principale
 - MVE, 91
 - rzssdp, 72
- Hausdorff, 56
- homologue, 40
- huit-voisinage, 104, 116
 - 8nei, 105
 - 8nei2, 105
- images planaires, 73, 77
- invariance
 - corrélation, 47
 - localisation, 85, 87
- invariant, 54
- localisation, 82, 86
 - MVE, 87
- médiane, 22
- Mahalanobis, 55
- Minkowski, 49
- mise en correspondance, 40
- moindres carrés, 16
- normalisation, 45
- occultation partielle, 48, 65, 73, 74
- point principal
 - rzssdp, 72
- points de contrôle, 63
- points principaux
 - MVE, 91
- programmation dynamique, 58
- quantification, 40
- quatre-voisinage, 103, 116
 - 4nei, 105
 - 4nei2, 105
- régions d'intérêts, 75
- régression, 16, 66, 68
 - régression robuste, 74
 - régularité de signal, 99
 - rang, 15
 - rapport bruit par signal, 67
 - ressemblance, 43
- stéréo, 40
- variation photométrique, 41