Ministère de l'enseignement supérieur et de la recherche scientifique
Université de Sciences et de la Technologie d'Oran – Mohamed BOUDIAF
Faculté de Génie Electrique
Département d'électronique

Thèse de Doctorat en Sciences

Spécialité : Electronique
Option : Vision et Reconnaissance de Formes

Thème

Approches robustes de recalage et de mosaïquage automatiques d'images de télédétection sous contraintes géométriques et photométriques

Présentée par
Mme MEDDEBER Lila, épouse ZOUAGUI

Soutenue le xx/xx/2014, devant le jury composé de :

Dr. BOUGHANMI Nabil	Pr., USTO	Président
Dr. BERRACHED Nasreddine	Pr., USTO	Rapporteur
Dr. TALEB-AHMED Abdelmalik	Pr., Univ. Valenciennes	Examinateur
Dr. BEREKSI-REGUIG Fethi	Pr., Univ. Tlemcen	Examinateur
Dr. KAHLOUCHE Salem	Dr., CTS Arzew	Examinateur
Dr. LOUKIL Abdelhamid	MC-A, USTO	Examinateur

REMERCIEMENTS

Ce travail de thèse a été réalisé au sein du Laboratoire de Recherche en Systèmes Intelligents (LARESI) du département d'électronique de la faculté de Génie Electrique de l'USTOMB.

Je souhaite tout d'abord exprimer mes remerciements à mon directeur de thèse, M. le professeur Nassreddine Berrached, qui m'a guidé pendant ces années de travail. D'abord, pour m'avoir initié à la recherche en vision par ordinateur pendant mon magister et proposé de faire une thèse, puis pour le soutien et les conseils qui m'ont permis de mener à bien ce projet.

Je tiens à remercier M. Abdelmalik Taleb-Ahmed, Professeur à l'université de Valenciennes. Qu'il soit assuré de ma profonde reconnaissance pour m'avoir accueilli plusieurs fois au sein de son laboratoire LAMIH. Je le remercie aussi pour ses précieux conseils et toutes les corrections qu'il a apportées à mon rapport de thèse.

Je remercie particulièrement M. Fethi Bereksi-Reguig, Professeur de l'université de Tlemcen, M. Salem Kahlouche, Directeur de recherche au CTS d'Arzew, M. Abdelhamid Loukil, Maitre de conférence à l'université des Sciences et de la Technologie d'Oran, qui ont accepté d'examiner cette thèse.

Je remercie également M. Nabil Boughanmi, professeur de l'université des Sciences et de la Technologie d'Oran pour l'honneur qu'il me fait de présider le jury de thèse.

Merci enfin à mes chers parents, à mon cher mari, à mes adorables enfants, à ma sœur et mes frères, pour m'avoir soutenu si fortement pendant toute la durée de ma thèse. Merci à toute ma famille.

TABLE DES MATIERES

INTRODUCTION GENERALE 1

Chapitre 1: Recalage d'images – cas des images de télédétection de modalités différentes

1.1. Présentation générale du processus de recalage d'images 7
1.2. Restriction du problème au cas d'images de télédétection de modalités différentes….. 13
1.2.1. Choix d'un modèle de transformation géométrique 13
1.3. Principe du recalage géométrique d'images 18
1.4. Définition mathématique du recalage d'images 19
1.4.1. Espace de recherche (types de transformation) 19
1.4.2. Interpolation des niveaux de gris des images transformées 24
1.5. Recalage basé sur les niveaux de gris des images 28
1.6. Recalage basé sur des primitives images 35
1.6.1. Détection des primitives images 35
1.6.2. Recalage des primitives images 41
1.7. Conclusion 42

Chapitre 2: Mise en correspondance de points d'intérêt sous contraintes photométriques

2.1. Le schéma habituel de la mise en correspondance 43
2.2. Points d'intérêt et descripteurs locaux 45
2.2.1. Détecteur de Harris 46
2.2.1.1. Principe 46
2.2.1.2. Propriétés 47
2.2.2. Le détecteur Harris-Laplacien 48
2.2.2.1. Principe 48
2.2.3. Critères d'évaluation 50
2.2.4. Caractéristiques souhaitables d'un détecteur 51
2.2.5. Détecteur Sift 51
2.2.6. Les descripteurs de points clés 53
2.3. Difficultés du problème de mise en correspondance sous contraintes photométriques 55
2.3.1. Recherche des correspondances SIFT (Scale Invariant Feature Transform) 56
2.3.2. Approche de mise en correspondance proposée 58
2.3.2.1. Propriétés fondamentales 58
2.3.2.2. Algorithme proposé 59
2.3.2.3. Résultats de simulation 65
2.4. Conclusion 67

Chapitre 3: Mise en correspondance de points d'intérêt sous contraintes géométriques

3.1. Filtrage robuste à l'aide de la contrainte géométrique 69

3.2. Modélisation de la caméra 69

3.2.1. Projection d'un point 3D sur l'image 71

3.2.1.1. Calcul de la matrice M 71

3.2.1.2. Calcul de la matrice K 73

3.3. Mise en correspondance de deux images 74

3.3.1. Stratégies de mise en correspondance 76

3.4. Calcul homographique 76

3.4.1. Estimation de l'homographie 78

3.4.1.1. L'algorithme DLT (Direct Linear Transform) 78

3.4.1.2. Normalisation des pixels 79

3.4.1.3. Limites de l'estimation aux moindres carrés 80

3.4.2. Optimisation de l'estimation de l'homographie par RANSAC 82

3.4.2.1. Calcul du nombre d'itérations N 82

3.4.3. Évolutions de RANSAC (RANdom SAmple Consensus) 84

3.4.4. Modifier la technique d'échantillonnage par l'algorithme de PROSAC (PROgressive SAmple Consensus)..... 84

3.4.4.1. Stratégie d'échantillonnage 85

3.5. Conclusion 88

Chapitre 4: Implémentations et résultats expérimentaux

4.1. Motivations et objectifs 89

4.2. Autres méthodes d'extraction de points d'intérêts 91

4.2.1. Algorithme basé sur le détecteur « Harris-Laplace » 91

4.2.1.1. Localisation de points caractéristiques 91

4.2.1.2. Sélection d'une échelle adaptée 92

4.2.1.3. Descripteurs 93

4.2.1.4. Propriétés du détecteur de points d'intérêt 94

4.2.1.4.1. Stabilité géométrique (Robustesse) 94

4.2.1.5. Algorithme utilisé 96

4.2.1.6. Paramétrisation 97

4.2.1.6.1. Espace multi-échelle, facteur d'échelle 97

4.2.1.6.2. Détermination des seuils 98

4.2.2. Algorithme basé sur l'aspect géométrique du point d'intérêt 102

4.2.2.1. Problématique 102

4.2.2.2. Objectif de l'algorithme 102

4.2.2.3. Formalisation du problème 103

4.2.2.4. Restriction du voisinage étudié: 3 critères 104

4.2.2.4.1. Premier critère: BrQ 105

4.2.2.4.2. Deuxième critère: DirQ 106
4.2.2.4.3. Troisième critère: AngQ 107
4.2.2.4.4. Réunion des 3 critères 108
4.2.2.4.5. Décision finale: quels sont les coins ? 108
4.2.2.4.5.1. Estimation des moments μ_φ et $\sigma_\varphi(X)$ 109
4.2.2.4.5.2. Calcul de l'Apparence et de la fonction de réponse de coin 110
4.2.2.6. Algorithme proposé 110
4.2.2.7. Paramétrisation et tests comparatifs de l'algorithme proposé 111
4.2.2.7.1. Images de synthèses 111
4.2.2.7.2. Images réelles 116
4.2.2.8. Discussion 118
4.3. Fusion d'images « Blending » 118
4.3.1 Fusion par pondération 118
4.3.2 Fusion par pyramides laplaciennes 119
4.4. Résultats des différentes étapes de la méthode de mosaïquage proposée 124
4.4.1. Détection et mise en correspondance de primitives 125
4.4.2. Estimation de la transformation 126
4.5. Evaluation des résultats 127
4.5.1. Moyenne des distances entre les intensités 127
4.5.2. Moyenne des distances géométriques 127
4.5.3. Moyenne carrée des erreurs 127
4.6. Recalage et construction panoramique de N images 131
4.7. Conclusion 136
CONCLUSION GENERALE ET PERSPECTIVES 137
BIBLIOGRAPHIE 140

TABLE DES FIGURES

FIGURE 1: Schéma fonctionnel du système proposé pour la construction panoramique des images de télédétection ... 5

FIGURE 1.1: Principe du recalage ... 7

FIGURE 1.2: Baie de Tokyo acquise par le capteur multispectral Landsat (à g.) et par le capteur radar AMI-ERS2 (à d.) ... 8

FIGURE 1.3: Organisation modulaire classique d'un algorithme de recalage ... 10

FIGURE 1.4: Transformation forward (à g.), et backward (à d.) ... 12

FIGURE 1.5: Capteur whiskbroom (à g.) et pushbroom (à d.) ... 14

FIGURE 1.6: Déformations causées par les changements de paramètres de vol des plates-formes ... 15

FIGURE 1.7: Géométrie d'une photographie aérienne ... 16

FIGURE 1.8: Différence entre arrangements de pixels en distance-sol (à g.). Effets de relief lors de la conversion en distance-sol (à d.) ... 17

FIGURE 1.9: Représentation simplifiée de l'espace T de transformation 2D–2D pour le recalage entre deux images d'une même scène prises pour deux points de vue différents. T représente un changement de perspective ... 19

FIGURE 1.10: Déformations issues d'une transformation affine (d'après [HAR04]). (a) Rotation d'angle θ. (b) Déformation **R**(-φ)**DR**(φ). Il faut noter que les directions des changements d'échelle (λ_1 et λ_2) sont orthogonales ... 22

FIGURE 1.11: Illustrations des deux approches possibles pour transformer une image : directe (a) et inverse (b). I_R est l'image originale et I_{Tr} est l'image transformée. T est la transformation appliquée aux coordonnées des images. La transformation prise comme exemple ici est une similarité qui va multiplier par deux la taille de l'image (s = 2). (a) Transformation de l'image en appliquant la méthode directe. (b) Transformation par la méthode inverse avec une interpolation par le plus proche voisin ... 25

FIGURE 1.12: Interpolation bilinéaire ... 26

FIGURE 1.13: Principe de l'Interpolation bicubique ... 27

FIGURE 1.14: Résultat des différentes méthodes d'interpolation pour une transformation constituée d'un changement d'échelle d'un facteur 6. (a) Image originale. (b) Interpolation de (a) par le plus proche voisin. (c) Interpolation bilinéaire de (a). (d) Interpolation bicubique de (a) ... 27

FIGURE 1.15: Ré-échantillonage d'images en coordonnées polaires. (a) Image originale. (b) Image après une rotation de -45°. (c) Ré-échantillonage de l'image originale en coordonnées polaires. (d) Image avec rotation en coordonnées polaires. Le décalage sur l'axe θ entre les images (c) et (d) est visible. Ce déplacement peut être obtenu par corrélation croisée en utilisant la FFT ... 31

FIGURE 1.16: Ré-échantillonage d'images en coordonnées log-polaires. Les zones en noir représentent les couples (ρ, θ) non contenus dans les images (a) et (b). (a) Image originale. (b) Image après une rotation de -45°. (c) Un grossissement de l'image (a) avec un facteur d'échelle de 1,5. (d) Image (a) après rotation de -45° et changement d'échelle de facteur 1,5. (e) Image originale rééchantillonnée en coordonnées log-polaires. (f) Image en coordonnées log-polaires de l'image (b). Il existe un déplacement sur l'axe θ. (g) Un déplacement sur l'axe log ρ (variation de l'amplitude) est visible lors du ré-échantillonage de l'image (c) en coordonnées log-polaires. (h) Déplacements sur l'axe θ et log(ρ) pour l'image avec rotation et changement d'échelle. Ces déplacements peuvent être obtenus par corrélation croisée en utilisant la FFT ... 32

FIGURE 1.17: Organigramme de la transformée rapide de Fourier [LIU11] ... 32

FIGURE 1.18: Histogramme joint de deux images IRM. (a) Deux images IRM décalées l'une par rapport à l'autre (translation sur l'axe horizontal). (b) Histogramme joint pour un déplacement de 4 pixels. (c) Histogramme joint pour un déplacement de 2 pixels. (d) Histogramme lorsque les images sont recalées 33

FIGURE 1.19: Information mutuelle: Critère de MI (a) calculé au voisinage du point P entre la nouvelle et l'ancienne photographie de la mosaïque (b). Le maximum de MI montre la position appariée correcte (point A). Le point B indique la fausse position appariée sélectionnée précédemment par l'opérateur humain. L'erreur a été provoquée par la qualité et la nature complexe de l'image 34

FIGURE 1.20: Représentation du profil d'un changement d'intensité (en noir). En gris foncé, la première dérivée. En gris clair, la dérivée seconde 37

FIGURE 1.21: Organigramme de recalage basé sur les ondelettes 40

FIGURE 2.1: Panorama réalisé à l'aide de mise en correspondance de points d'intérêt 44

FIGURE 2.2: Influence des valeurs propres sur la rapidité des transitions 47

FIGURE 2.3: Influence des valeurs propres sur la nature des transitions 47

FIGURE 2.4: Localisation de l'invariant (coin) et détermination de son échelle (cercles rouges) sur deux images ayant une scène commune à deux échelles différentes 48

FIGURE 2.5: Détermination de l'échelle de l'invariant (au maximum de l'amplitude). L'image filtrée à droite est la même que celle de gauche à une échelle $\frac{1}{2}$ près 49

FIGURE 2.6: L'espace d'´echelle dans le cas de l'opérateur Harris-Laplacien : en ordonnée et en abscisse l'opérateur de Harris (coordonnées du point), en z l'échelle du point donnée par le Laplacien 49

FIGURE 2.7: – A gauche – Points d'intérêt détectés. Les rayons des cercles sont proportionnels aux échelles calculées, et l'orientation trouvée est matérialisée par le rayon. A droite – Descripteurs pour une sélection de points. Les grilles affichées contiennent les statistiques d'orientation du gradient dans un voisinage du point dont la taille est proportionnelle à l'échelle détectée 51

FIGURE 2.8: Espace échelle gaussien. On recherche les extrema des différences de gaussienne afin de déterminer la position des points d'intérêt. Une octave correspond au doublement de l'échelle du flou gaussienσ 52

FIGURE 2.9: Points invariants sur une image aérienne: la taille du cercle correspond à l'échelle à laquelle a été détecté le point au centre 53

FIGURE 2.10: Descripteur d'un point invariant: la fenêtre de gauche représente un ensemble de 16×16 échantillons portant chacun un vecteur issu d'un filtrage par une gaussienne de variance à l'échelle du point invariant considéré. Celle de droite représente le descripteur calculé: celui-ci est divisé en 4 parties contenant chacune un histogramme d'orientations. Chaque orientation est calculée en fonction des vecteurs de la fenêtre de gauche 54

FIGURE 2.11: Points d'intérêt issus de la figure 2.7, mis en correspondance par le critère du rapport de distances. Notez les quelques correspondances aberrantes, pour lesquelles les points d'intérêt appariés ne correspondent pas aux mêmes points physiques 55

FIGURE 2.12: Relations entre coefficients approximatifs aux différentes échelles 61

FIGURE 2.13: Ordre de recherche 63

FIGURE 2.14: Organigramme du procédé de contrôle 64

FIGURE 3.1: Modélisation de la caméra et projection d'un point 3D 70

FIGURE 3.2: Angle de vue de la caméra 70

FIGURE 3.3: Projection perspective d'un point du monde par changement de repère 71

FIGURE 3.4: Projection d'un point P sur le plan image 73

FIGURE 3.5: Mise en correspondance de deux points images 75

FIGURE 3.6: Résolution aux moindres carrés avec des données aberrantes: recherche d'un modèle affine f minimisant $\sum_{i=1}^{n}|y_i - f(x_i)|^2$ où les (x_i, y_i) sont les données; la solution trouvée est beaucoup plus attirée par le point éloigné que par les quatre points suivant le modèle 81

FIGURE 4.1: Schéma fonctionnel du système proposé pour la construction panoramique des images de télédétection 89

FIGURE 4.2: Schéma Synoptique du 1er module de travail « Module de Recalage » 90

FIGURE 4.3: Détecteur Harris sous un espace Multi-Echelles 93

FIGURE 4.4: Robustesse du détecteur de Harris-Laplace et du détecteur SIFT pour des changements d'échelle 95

FIGURE 4.5: Localisation de points d'intérêt pour une échelle à partir d'une carte de Harris 97

FIGURE 4.6: Echelles caractéristiques obtenues pour quatre points d'intérêt dans un espace multi-échelle utilisant un facteur d'échelle de $\sqrt{2}$ 98

FIGURE 4.7: Délocalisation des points caractéristiques avec l'échelle 99

FIGURE 4.8: Elimination de points caractéristiques par sélection des échelles adaptées 99

FIGURE 4.9: Extraction des points caractéristiques saillants 100

FIGURE 4.10: Points caractéristiques obtenus respectivement avec les réponses (H2) et (H3) 101

FIGURE 4.11: Points de Harris-Laplace en l'absence de seuil 101

FIGURE 4.12: Points d'intérêt de Harris-Laplacien de Gaussienne 101

FIGURE 4.13: Analyse du point $\mathbf{Q_1}$, seule une partie du voisinage est intéressante 102

FIGURE 4.14: Caractérisation d'un coin 103

FIGURE 4.15: Représentation simplifiée de la probabilité $\mathbf{P_{SG}(X)}$ 104

FIGURE 4.16: Restriction de voisinage, (a)Distance d(X) entre une courbe et un point X, (b) Fonction de lignes des niveaux d'un coin 105

FIGURE 4.17: Restriction par le critère **BrQ** à l'aide la probabilité conditionnelle $\mathbf{P}(\mathbf{BrQ}|\Delta\mathbf{b}(\mathbf{X}))$ 106

FIGURE 4.18: Restriction par les critères **BrQ** et **DirQ** à l'aide la probabilité $\mathbf{P}(\mathbf{BrQ}|\Delta(\mathbf{X})).\mathbf{P}(\mathbf{DirQ}|\mathbf{h}(\mathbf{X}))$ les points **X** qui n'appartiennent pas à un segment [de ligne de niveau] pointant vers **Q** sont supprimés (par rapport à la figure 4.17) 107

FIGURE 4.19: Restriction par les critères **BrQ** et **DirQ** à l'aide la probabilité $\mathbf{P}(\mathbf{BrQ}|\Delta(\mathbf{X})).\mathbf{P}(\mathbf{DirQ}|\mathbf{h}(\mathbf{X})).\mathbf{P}(\mathbf{AngQ}|\Delta(\boldsymbol{\varphi}(\mathbf{X})))$ 108

FIGURE 4.20: Vecteurs $\boldsymbol{\varphi}(\mathbf{X})$ (directions du gradient) pondérés par $\mathbf{w(X)}$ pour le calcul de $\boldsymbol{\mu_\varphi}$ et $\boldsymbol{\sigma_\varphi}(\mathbf{X})$: dépendance de la distance et de la valeur de $\mathbf{P_{SG}(X)}$ 109

FIGURE 4.21: Image synthétique de base et la solution de référence (60 coins) 111

FIGURE 4.22: Tests sur images brutes - Dans l'ordre : Solution de référence (60 coins réels), Détecteur proposé, SIFT et Harris 112

FIGURE 4.23: Tests sur image bruitée - Dans l'ordre : Solution de référence, Détecteur proposé, SIFT et Harris 113

FIGURE 4.24: Tests sur image avec flou - Dans l'ordre : Solution de référence, Détecteur proposé, SIFT et Harris 114

FIGURE 4.25: Tests sur image avec rotation-Dans l'ordre: Solution de référence, Détecteur proposé, SIFT et Harris 115

FIGURE 4.26: Résultats des tests présentés sous forme d'histogrammes 115

FIGURE 4.27: Tests sur image aérienne en conditions idéales-(a) Image brute, (b) Détecteur proposé,(c) Harris, (d) SIFT avec ses descripteurs 117

FIGURE 4.28: Tests sur une image satellite de Bruxelles en conditions idéales-(a) Image brute, (b) Détecteur proposé, (c) Harris, (d) SIFT avec ses descripteurs 118

FIGURE 4.29: Fusion d'images (Blending) basée sur les distances 119

FIGURE 4.30: Différentes étapes de génération de la pyramide Laplacienne 120

FIGURE 4.31: Pyramide finale de reconstruction. (a) Masques de reconstruction des images A et B (LA_n: pyramide gauche, LB_n: pyramide droite), (b) pyramide de reconstruction de l'image mosaïquée finale 121

FIGURE 4.32: Génération et reconstruction de la pyramide laplacienne 122

FIGURE 4.33: Pyramide des différents masques de reconstruction de la figure 4.32 123

FIGURE 4.34: Fusion d'images (blending) à plusieurs bandes, (a). Images originales et l'image finale mosaïquée, (b). images laplaciennes + les masques sur la bande 1 (échelle 0 àσ), (c). images laplaciennes + les masques sur la bande 2 (échelle σ à 2σ), (d). images laplaciennes + les masques sur la bande 3 (échelle <2σ) 123

FIGURE 4.35: Schéma fonctionnel du système proposé pour la construction panoramique des images de télédétection 124

FIGURE 4.36: Images de tests, «Images satellitaires d'IKONOS à haute résolution de Ponte Golden Gate, San Francisco acquises à deux instants séparés» 124

FIGURE 4.37: Différentes étapes de l'algorithme de recalage. (a) détection des primitives, (b) appariement avec un seuil de 0.5. Nombre de points détectés : Image1 : 800, Image2 : 756 ; Nombre de points en correspondance : 200 125

FIGURE 4.38: Résultat de fusionnement par interpolation bilinéaire après application de la fonction de transformation affine sur les deux images (a) et (b) 126

FIGURE 4.39: Fusionnement des deux images (a) et (b) par distances pondérées 127

FIGURE 4.40: Fusionnement d'une image satellite et cartographiques de la ville de sidi Bel-Abbes. (a)image satellite / bâtis, (b) image satellite/ réseau routier 127

FIGURE 4.41: Mosaïquage de 8 images satellitaires avec une projection plane 131

FIGURE 4.42: Mosaïquage de 9 images satellitaires avec une transformation affine 132

FIGURE 4.43: Mosaïquage de 9 images aériennes avec une transformation affine 133

FIGURE 4.44: Mosaïquage de 18 images aériennes avec une transformation projective 135

LISTE DES TABLEAUX

TABLEAUX 1.1: Exemples de fonctions radiales [ZIT03] 12

TABLEAUX 1.2: Modèles de transformations géométriques linéaires en 2D. Toutes les transformations comprennent les paramètres de la transformation précédente à laquelle nous rajoutons de nouveaux paramètres. Les paramètres t_x, t_y, s, θ sont expliqués dans le texte 23

TABLEAUX 2.1: Comparaison de la durée de classification 65

TABLEAUX 2.2: Complexité arithmétique pour des vecteurs de caractéristiques à 8-dimensions 66

TABLEAUX 2.3: Complexité arithmétique pour des vecteurs de caractéristiques à 13-dimensions 66

TABLEAUX 2.4: Complexité arithmétique pour des vecteurs de caractéristiques à 16-dimensions 67

TABLEAUX 4.1: Entropie des descripteurs calculée pour des points d'intérêt issus de deux détecteurs : le détecteur de Harris-Laplace et le détecteur «SIFT »..... 96

TABLEAUX 4.2: Seuils de Harris et de Laplacien de Gaussienne 100

TABLEAUX 4.3: Valeurs de RMSE et ϵ_{est} pour des images de tests 130

TABLEAUX 4.4: Différentes valeurs d' RMSE pour les images des figures 4.41, 4.42, 4.43, 4.44 135

Résumé

L'objectif de notre thèse s'inscrit dans le cadre de la construction panoramique des images de télédétection (Images satellitaires, Radars, Aériennes et Cartographiques) de modalités différentes, qui chacune apporte un type d'information particulier à propos d'une scène donnée. Ces images ne sont généralement pas situées dans un même référentiel spatial, ce qui empêche leur exploitation simultanée. Pour reconstituer un panorama à partir d'un grand nombre d'images, nous nous sommes basés sur deux grands axes: une part importante de notre travail a été consacrée au recalage automatique des images de télédétection en passant par les étapes suivantes: 1) L'extraction de points d'intérêt, 2) La mise en correspondance des points d'intérêt entre les images à l'aide de descripteurs photométriques des voisinages de point, 3) Le filtrage des appariements produits à l'étape précédente afin de ne conserver que ceux compatibles avec une contrainte géométrique, dont on peut alors calculer les paramètres. Cependant, la ressemblance photométrique seule utilisée en deuxième étape ne suffit pas quand plusieurs points ont la même apparence. Nous proposons également différents traitements permettant d'améliorer sensiblement le rendu et de corriger la plupart des défauts liés à l'assemblage et à la projection cylindrique d'une collection d'images acquises avec des paramètres de prises de vue différents. Notre première contribution a consisté en l'élaboration d'une nouvelle approche d'extraction de points d'intérêt (coins) à l'aide de critères photométriques et géométriques. Notre seconde contribution étend ce résultat aux forts changements de point de vue, en améliorant la partie de mise en correspondance de la méthode SIFT et RANSAC. Elles permettent d'obtenir des correspondances plus nombreuses et plus densément réparties, dans des scènes difficiles contenant des motifs répétés observés sous des angles très différents.

Mots-clés : points d'intérêt, mise en correspondance, descripteurs SIFT, ondelettes de HAAR, PROSAC, fusion d'images, forts changements de point de vue, mosaïquage.

Abstract

The objective of this thesis concerns the construction of panoramic images of remote sensing (satellite, Radar, aerial and Cartographic images) from different modalities, each of them provides a particular type of information about a given scene. These images are usually not located in the same spatial reference, which prevents simultaneous operation. To reconstruct a panorama from a large number of images, we relied on two main areas: an important part of our work was devoted to the automatic registration of remote sensing images through the following steps: 1) Extracting interest points, 2) Matching interest points using photometric descriptors computed on point neighborhoods, 3) Filtering previous matches so as to retain only those compatible with a geometric constraint, whose parameters can then be computed. However, the only photometric similarity used in the second step is not enough when several points have the same appearance. We also propose various processing techniques which appreciably improve the rendering of the mosaic image and correct the majority of the defaults related to the cylindrical projection and the assembly of a collection of images acquired with differing image capture parameters. Our first contribution was the elaboration of a new extracting interest points approach (corners) using photometric and geometric criteria. Our second contribution extends that result to strong viewpoint changes, improving the matching part of the SIFT and RANSAC method. The correspondences obtained are both more numerous and more densely distributed, in scenes containing many repeated patterns seen from very different viewpoints.

Keywords: points of interest, matching, descriptors SIFT, wavelet of HAAR, Prosac, blending, multiview point, mosaicking.

INTRODUCTION GENERALE

Le traitement d'images joue désormais un rôle prépondérant dans un grand nombre de domaines tels que l'imagerie satellitaire, optique ou médicale. La quantité des données, souvent importante, peut être allégée grâce à une automatisation de l'analyse et de l'interprétation des images.

Une des problématiques majeures en traitement d'images est de pouvoir analyser et traiter plusieurs images dans un référentiel commun pour regrouper ou comparer leurs informations respectives. Ce problème, connu sous le nom de recalage (ou alignement), est nécessaire dès qu'il s'agit de confronter plusieurs sources d'informations acquises à des temps différents, par des capteurs différents ou à partir de vue différentes. Le recalage d'images, fait référence à la procédure ayant pour but de trouver une transformation géométrique reliant deux ou plusieurs images. Le but est de situer les images dans un même repère géométrique. C'est un domaine de recherche aux multiples applications et impliquant un vaste nombre de techniques d'imagerie numérique. Cette multiplicité provient essentiellement de la notion assez générale de lien entre images: les méthodes de recalage interviennent en fait à chaque fois qu'une correspondance spatiale entre images est requise. Les applications qui s'en déduisent sont aussi nombreuses que variées.

L'expression du besoin de performance en systèmes spatiaux d'observation de la terre s'oriente depuis plusieurs années vers l'amélioration de la résolution spatiale. Les nouvelles et futures générations de systèmes optiques commerciaux à haute résolution spatiale ouvrent le champ à de nouvelles applications comme la gestion des risques liés à l'environnement ou la cartographie urbaine 3D, à condition de conserver une fauchée conséquente.

Or, pour la technologie actuelle des instruments optiques, on constate qu'un accroissement de la résolution implique une diminution de la fauchée des zones imagées: 60km pour SPOT contre 11km pour IKONOS. Par ailleurs, les plates formes des satellites d'observation à haute résolution spatiale présentent de grandes capacités de manœuvrabilité. Cette agilité permet d'envisager l'acquisition, lors d'un même passage orbital, de plusieurs scènes élémentaires de faibles fauchées qui, mises bout à bout par des techniques de mosaïquage, couvrent une zone de grande surface.

L'utilisation du mosaïquage permet donc, au niveau système, de réduire les contraintes de dimensionnement du champ de l'instrument et donc la création de produits images mosaïquées haute résolution spatiale suivant un champ relativement important.

La problématique

L'objectif du mosaïquage est de créer, à partir de plusieurs images contiguës ou non, une seule image cohérente et homogène sur les plans géométrique et radiométrique. L'image mosaïquée résultante ne doit pas présenter de discontinuités rédhibitoires en termes de qualité géométrique intrinsèque, radiométrique ou encore de restitution de contraste entre les différentes images la constituant, notamment au niveau de leurs frontières.

Les acquisitions suivant le même passage orbital des différentes scènes élémentaires sont quasi-simultanées: elles permettent donc de s'affranchir des problèmes de différence d'angle d'éclairement et de changements importants du paysage, à l'exclusion des objets mobiles comme les véhicules et les nuages. La contrepartie est que les images successives présentent d'importantes discontinuités d'angle d'acquisition, notamment dues au basculement en tangage entre les prises de vue.

Ces différences importantes d'angles impliquent des distorsions géométriques (effet de parallaxe, variation de la taille du pixel au sol) ainsi que des différences radiométriques locales lorsque les éléments du paysage ne présentent pas des caractéristiques purement lambertiennes (effet non négligeable du caractère bidirectionnel de la réflectance de certains éléments du paysage). Des traitements au sol sont donc à prévoir afin de corriger ou d'atténuer ces différences pour la constitution de mosaïques cohérentes et homogènes.

L'objectif de notre travail s'inscrit dans le cadre de la construction panoramique des images de télédétection (Images satellitaires, Radars, Aériennes et Cartographiques) de modalités différentes, qui chacune apporte un type d'information particulier à propos d'une scène donnée. Ces images ne sont généralement pas situées dans un même référentiel spatial, ce qui empêche leur exploitation simultanée. Comme on peut avoir des cas de construction panoramique entre des images de télédétection réelles et des modèles synthétiques.

L'utilisation d'un champ visuel large peut être une solution partielle, mais les images obtenues avec un tel champ visuel ont des distorsions substantielles. La construction d'une mosaïque consiste à recoller toutes les images de référence dans le même repère, de façon à reconstituer une seule vue d'ensemble de la scène.

Pour reconstituer un panorama à partir d'un grand nombre d'images, nous nous sommes basés sur deux grands axes (voir figure 1):

- Une part importante de notre travail a été consacrée au recalage automatique des images de télédétection et à l'estimation du mouvement de la caméra.

- Nous proposons également différents traitements permettant d'améliorer sensiblement le rendu et de corriger la plupart des défauts liés à l'assemblage et à la projection cylindrique ou plane d'une collection d'images acquises avec des paramètres de prises de vue différents.

Le mouvement de la caméra entre deux positions est estimé en prenant en compte les contraintes photométriques et géométriques, et en observant le déplacement des points considérés dans les images en tenant compte du phénomène de parallaxe. Le procédé utilisé repose sur les étapes suivantes.

- Détection des points d'intérêt.
- La mise en correspondance.
- Approximation de l'homographie correspondante.

L'extraction des points d'intérêt doit se faire sans perte d'informations utiles, d'où la nécessité de certains critères de choix auxquels doivent répondre ces points tels que: la robustesse au bruit et aux changements de luminosité; la précision de leur position dans l'image ou dans l'objet; l'invariance aux transformations de rotation, d'échelle, de translation,…; la localité qui permet à un point d'être plus robuste aux occlusions; la distinctivité qui permet de mettre correctement en correspondance un point d'intérêt avec un autre point d'intérêt sur une autre image.

Plusieurs types de points d'intérêt sont disponibles dans la littérature mais qui ne vérifient pas toutes les contraintes citées ci-dessus. Dans notre travail de thèse, nous nous sommes intéressés plus particulièrement à développer deux nouvelles approches différentes se basant sur les points d'intérêt et qui respectent un grand nombre de contraintes géométriques et photométriques.

Ensuite la recherche d'appariements, ou la mise en correspondance de points d'intérêt d'une image à l'autre d'une même scène observée avec un point de vue différent, est effectuée à l'aide des descripteurs photométriques.

Puis, on détermine les paramètres de la matrice de mouvement, soit dans le cas de la géométrie épipolaire, où sont calculés les paramètres de la matrice fondamentale encodant le mouvement de translation et de rotation séparant les deux vues à partir des coordonnées des points d'intérêt mis en correspondance, soit en restreignant la mise en correspondance à deux plans, la relation à estimer étant alors une homographie.

Il est en même temps nécessaire de réaliser un filtrage des correspondances. Les appariements extraits par les méthodes de mise en correspondance ne sont pas tous exacts. L'utilisation de procédés capables de traiter la présence de données aberrantes, tel RANSAC, permet de filtrer les données afin de ne prendre en compte dans l'évaluation du mouvement que celles jugées fiables, et ainsi de rendre robuste l'estimation statistique subséquente. RANSAC nécessite la fixation de deux seuils qui ont une forte influence sur le résultat, l'un pour la précision des appariements retenus, et l'autre pour la taille minimale de l'ensemble retenu.
Enfin, à partir de ces données filtrées, les informations concernant le positionnement de la caméra sont obtenues.

Pour les contributions que nous avons apportées :

Plusieurs difficultés majeures rendent le processus de recalage difficile à réaliser automatiquement, surtout au niveau de la mise en correspondance des points d'intérêt. Tout d'abord, la représentation de l'information n'est pas la même d'une modalité à l'autre ; de plus, les éléments d'une même scène n'ont pas forcément les mêmes expressions. C'est pourquoi le recalage automatique d'images de télédétection de modalités différentes reste à l'heure actuelle un domaine de recherche très actif. Dans ce travail de thèse, nous avons proposé donc d'apporter un ensemble de contributions au problème de la mise en correspondance de points d'intérêt et de l'estimation du mouvement. Pour cela, nous avons présenté plusieurs approches, que nous résumons.

Notre 1ère contribution est basée sur la détection des coins elle consiste à restreindre le voisinage du coin recherché car la plupart des méthodes actuelles travaillent sur tout le voisinage d'un coin (Q) afin de déterminer à l'aide d'une fonction de décision prédéfinie si Q est un coin ou non. Nous avons remarqué que la majorité des détecteurs utilisé et connus, se base que sur les valeurs radiométriques de l'image et plus particulièrement sur celles prises par le gradient ce qui va introduire des imprécisions non souhaitables dans nos applications. Notre contribution consistait à introduire d'autres facteurs géométriques de sélection des points d'intérêt en plus des facteurs de radiométries.

Notre 2ème apport consiste en l'amélioration d'une méthode de mise en correspondance de points invariante aux fortes déformations utilisées par l'algorithme SIFT, qui génère les points d'intérêt et les descripteurs associés, et permet la mise en correspondance pour de forts changements de point de vue. Notre extension remplace le schéma de mise en correspondance KD-tree et BBF par une autre méthode plus robuste et rapide et qui se base sur la transformée en ondelettes de Haar et améliore la sélection des paires d'images simulées pour fournir des correspondances plus nombreuses et plus densément réparties. Elle permet en outre de mettre en correspondance des scènes contenant de nombreux motifs répétés, par exemple des scènes d'intérieur ou d'extérieur en milieu urbain, dans lesquelles de telles zones peuvent être observées sous des angles très différents.

Notre dernier apport concerne l'étape de filtrage des correspondances et plus précisément de remplacer la technique d'échantillonnage de l'algorithme RANSAC par celui de PROSAC *(Progressive Sample Consensus).* Plutôt que de traiter tous les appariements de la même manière (RANSAC), on sélectionne ainsi des échantillons sur un ensemble de correspondances les mieux appariées. Cet ensemble a comme particularité de s'élargir petit à petit au fil des itérations, ce qui permet une recherche plus rapide du consensus en utilisant une fonction d'évaluation basée sur la probabilité de fausses alarmes.

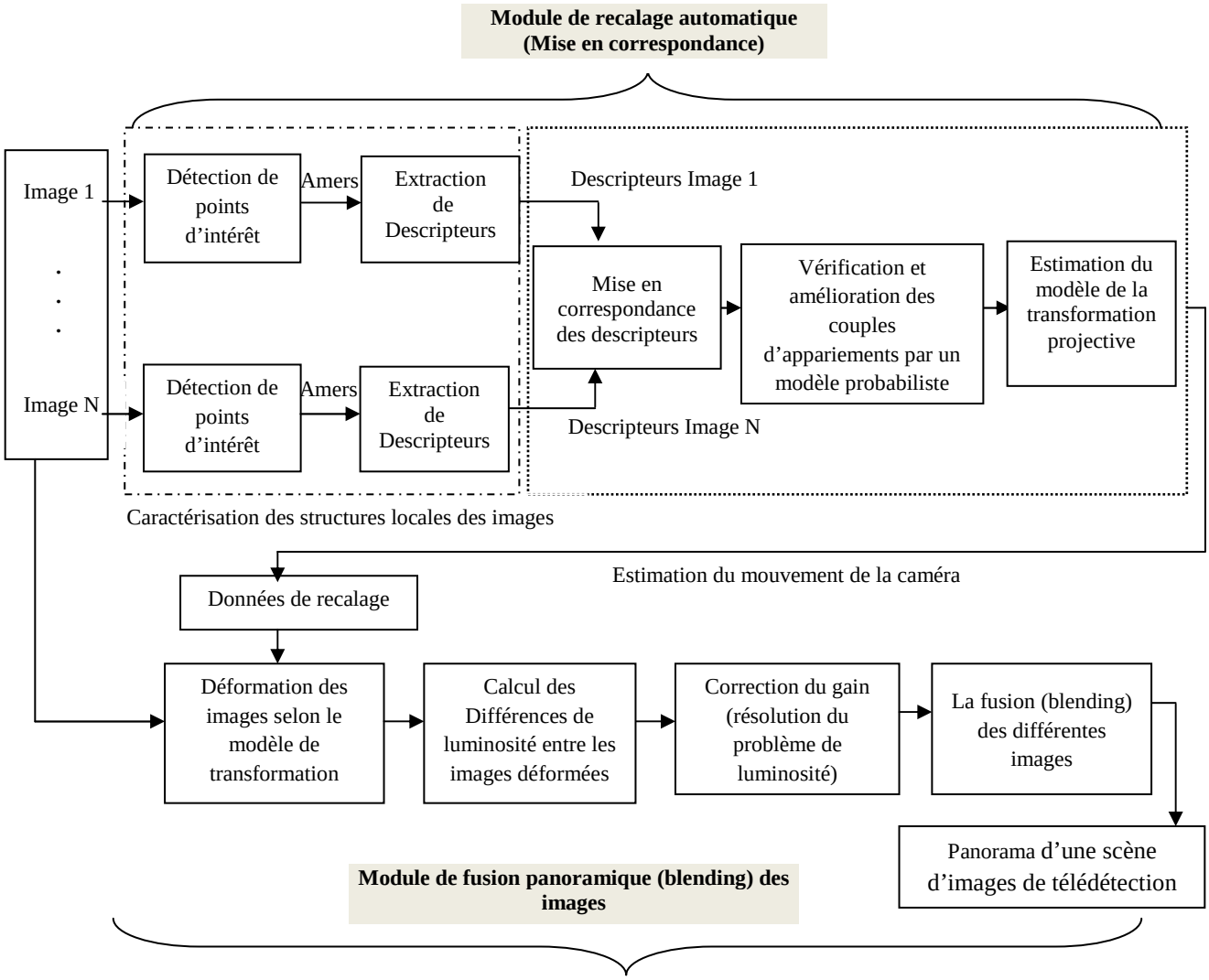


Figure 1. *Schéma fonctionnel du système proposé pour la construction panoramique des images de télédétection*

Ce manuscrit est divisé en deux parties : la première partie (chapitres 1) porte sur certaines considérations méthodologiques générales, qui aboutissent à la définition exacte du problème à traiter; dans cette même partie, un « état de l'art » nous permet de présenter les méthodes et les approches qui ont été proposées dans l'étude de ce problème. La seconde partie (chapitres 2 à 4) est consacrée à l'approche originale qui a été développée durant cette thèse, ainsi qu'à la présentation de résultats obtenus sur des données synthétiques et réelles.

Le premier chapitre traite le problème de recalage d'un point de vue général. Nous présentons les différentes étapes qui composent le recalage d'images et nous abordons les deux approches principales pour recaler une paire d'images. Ces approches sont les méthodes basées sur l'utilisation directe des niveaux d'intensité des images et les méthodes basées sur la détection de primitives images. Nous étudions ainsi dans ce chapitre les mesures de similarité et les méthodes d'optimisation pouvant être utilisées pour le recalage d'images.

Le but du Chapitre 2 est de proposer une contribution au problème de mise en correspondance sous contraintes photométriques des images satellitaires multi-temporelles. L'originalité de cette contribution repose sur l'utilisation de nouveaux détecteurs de points d'intérêt à base de l'algorithme SIFT et RANSAC pour l'extraction de points d'appui fiables des images satellitaires à recaler. Les points d'appui sont, par la suite, mis en correspondance par une mesure de similarité basée sur l'algorithme KD-tree et l'ondelette de HAAR. En partant des points homologues, les paramètres du modèle de transformation peuvent ainsi être estimés dans le chapitre 3. Nous détaillons les différentes méthodes pour créer ces mosaïques et les méthodes d'estimation des homographies qui constituent la phase la plus difficile dans notre travail. Cette stratégie de mise en correspondance sous contraintes géométriques a montré la capacité de recaler réellement des images multi-temporelles avec précision sous-pixelique. Après avoir évoqué l'importance du recalage dans le domaine du traitement d'images en général et en imagerie satellitaire en particulier, nous présentons dans le chapitre 4 l'ensemble des développements méthodologiques relatifs aux algorithmes proposés. Les résultats expérimentaux sont alors exposés.

Nous terminons ce document par une conclusion qui met en avant les contributions et les perspectives de nos travaux.

CHAPITRE 1

RECALAGE D'IMAGES – CAS DES IMAGES DE TELEDETECTION DE MODALITES DIFFERENTES

1.1. Présentation générale du processus de recalage d'images

On dispose de deux ensembles de données, deux images, qui ont été obtenus à partir de la même scène 3D mais dans des conditions différentes: modes d'acquisition différents et/ou depuis des points de vue différents et/ou à des instants différents. On considère que la scène a subi des modifications mineures entre les deux acquisitions, de sorte que les positions spatiales, les propriétés géométriques, ou encore les propriétés physiques d'une forte majorité d'éléments de la scène sont restées quasiment les mêmes. Une acquisition consiste en la projection sur un plan 2D d'informations liées à une scène en 3D. L'objectif du recalage est de trouver une transformation géométrique, issue d'un modèle particulier, qui permette de placer sur un même point du plan deux projections qui se rapportent aux mêmes éléments de la scène (voir figure 1.1).

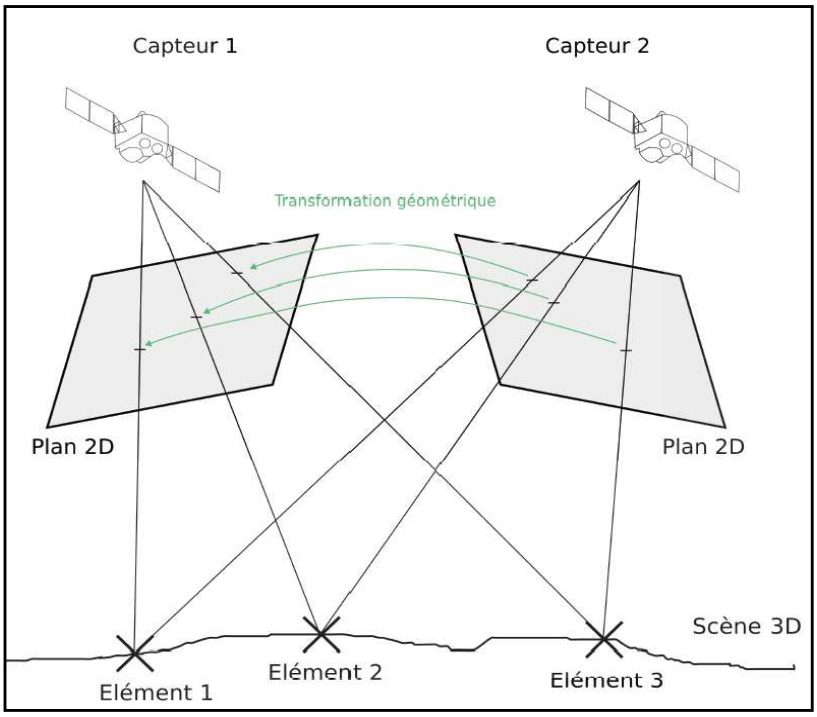


Figure 1.1. *Principe du recalage*

En réalité, les images à recaler ne sont pas définies comme deux plans 2D continus, mais comme deux tableaux 2D de pixels. Chacun d'entre eux est lié à une cellule de résolution au sol d'une certaine aire : de ce fait, plusieurs éléments de la scène peuvent être intégrés au sein d'un même pixel, ou au contraire, une cible peut appartenir à plusieurs pixels à la fois. De plus, la nature des capteurs utilisés agit comme un filtre qui sélectionne l'information : dans le

cas de modes d'acquisition différents, l'expression des éléments de la scène peut changer d'une image à l'autre.
Au final, la transformation géométrique recherchée est donc la fonction qui superpose géométriquement les pixels de chacune des deux images dont il est possible de décider s'ils se rapportent aux mêmes groupes d'éléments de la scène.

Figure 1.2. *Baie de Tokyo acquise par le capteur multispectral Landsat (à g.) et par le capteur radar AMI-ERS2 (à d.).*

Par exemple, deux images de modalités différentes de la baie de Tokyo sont représentées dans la figure 1.2. A gauche se trouve une image multispectrale (Landsat 7), et à droite se trouve une image radar (AMI-ERS2); les deux images possèdent la même résolution, soit 30m. Entre ces deux images, il existe de nombreuses structures que l'on peut reconnaître, comme la forme de la baie, le tracé des cours d'eau ou encore la ligne de côte. En revanche, de nombreux autres éléments de la scène ont des expressions différentes comme la végétation ou encore le milieu maritime. Ces deux images ne sont pas recalées, alors que l'application d'une simple transformation rigide (composition d'une rotation et d'une translation dans le plan) permettrait de superposer entre eux une large majorité d'éléments.

Le recalage d'images ne concerne pas uniquement la télédétection, mais aussi le domaine médical ainsi que celui de la robotique, de l'automatisation... et de la vision par ordinateur en général. Par exemple dans le cas médical, il est possible d'utiliser comme sources de données complémentaires, la Tomodensitométrie, sensible à la présence de structures solides comme les os, et la Tomographie d'Émission Monophotonique plutôt sensible à l'activité métabolique et qui ne peut détecter les structures solides que de manière indirecte. Comme les images acquises par ces deux procédés ne sont très probablement pas de même résolution, et prises depuis des points de vue et à des instants différents, cet exemple donne toute la mesure de la difficulté du recalage d'images médicales. Ce type d'approche « multi-source » et/ou « multi-temporel » est d'ailleurs une pratique très répandue en médecine: il peut s'agir d'examiner le

développement d'une tumeur, de contrôler et suivre les effets d'un traitement, de comparer l'anatomie du patient avec des atlas, etc...

Dans le cas plus général de la vision par ordinateur, nous pouvons par exemple citer des applications comme la segmentation d'image, la reconnaissance de formes, la reconstruction 3D d'objets, leurs localisations dans l'espace, le tracking, la vision stéréoscopique, etc... Il est important de noter que ces trois domaines d'application du recalage, télédétection, imagerie médicale, et vision par ordinateur, ne sont pas clivés, et que bien au contraire, les recherches dans les trois domaines sont interdépendantes.

D'une façon générale, la majorité des algorithmes utilisés en recalage d'images sont décomposables en plusieurs sous-étapes; la modularité de ces algorithmes permet alors de replacer ceux-ci dans un contexte unifié qui facilite leur comparaison. Les algorithmes de recalage sont communément divisés de la manière suivante :

- ***Étape de mise en correspondance*** : il s'agit de l'étape la plus importante du processus de recalage. La mise en correspondance consiste à repérer dans les images des occurrences qui sont associées aux mêmes éléments de la scène. Pour cela, il existe deux approches: la mise en correspondance peut reposer soit sur le contenu pixelique des images (méthodes « area-based »), soit sur l'utilisation de caractéristiques d'éléments remarquables qui ont généralement en télédétection, un support géographique effectif (méthodes « feature-based »). L'emploi d'une méthode feature-based nécessite alors une étape préalable d'extraction de primitives.

- ***Étape d'estimation de la transformation géométrique:*** il s'agit de choisir un modèle de transformation géométrique, puis d'en estimer les paramètres à partir des appariements $\{p_i,q_i\}_{i=1..N}$ établis lors de l'étape de mise en correspondance. L'instance particulière du modèle ainsi identifiée permet alors le recalage des images.

- ***Étape de transformation et rééchantillonnage***: l'image esclave, qui est celle devant être recalée sur l'image maître, est transformée (ie. « Déplacée ») en fonction de l'instance particulière du modèle qui a été estimée lors de l'étape précédente. Après transformation, les nouvelles coordonnées des pixels de l'image esclave sont généralement non-entières; il est alors nécessaire de rééchantillonner cette image sur une grille 2D régulière.

Ces étapes sont résumées dans la figure 1.3 suivante:

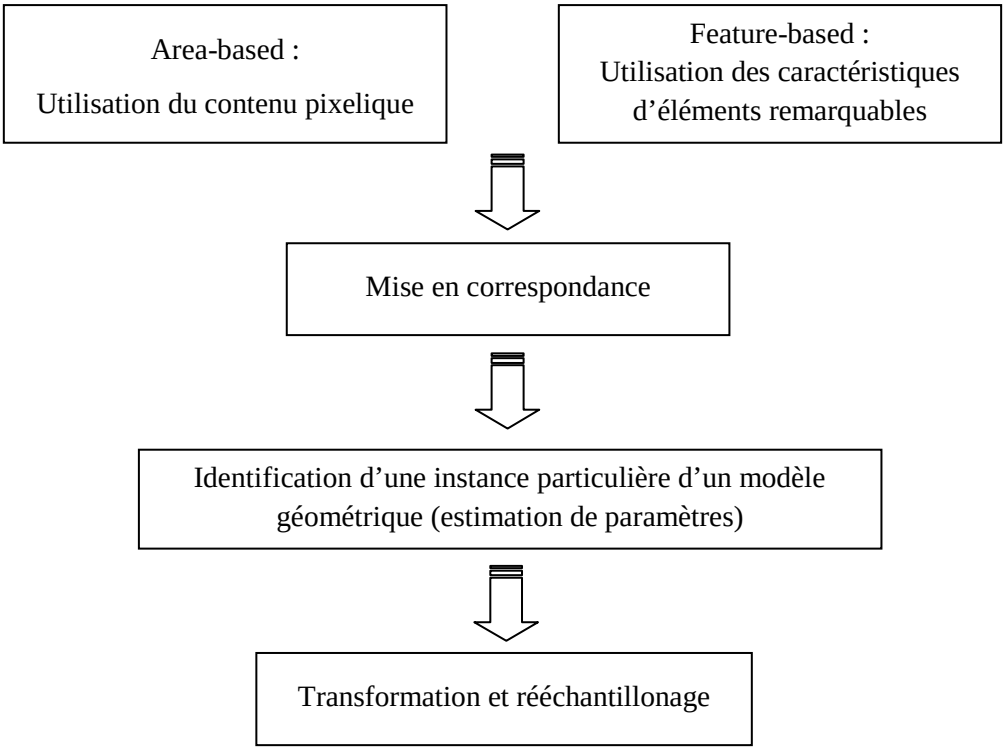


Figure 1.3. *Organisation modulaire classique d'un algorithme de recalage.*

Les méthodes area-based consistent à définir une mesure de similarité qui permet d'évaluer à quel point deux sous-images (correspondant à deux fenêtres dans chaque image) se ressemblent. Il s'agit donc de tester plusieurs paires de fenêtres et de ne retenir que les paires les plus pertinentes en fonction de leurs niveaux de « ressemblance » : les méthodes area-based reposent alors sur la définition de métriques qui utilisent les intensités de voisinages de pixels. Qu'il s'agisse de calculer directement la distance entre intensités, de calculer une dépendance statistique ou encore une information spectrale locale (ondelettes), toutes les méthodes area-based ont pour point commun l'utilisation de l'information supportée par les pixels eux-mêmes. Par ailleurs, l'utilisation de métriques sur des pixels ne se limite pas seulement à l'emploi de voisinages. Ainsi, il est aussi possible d'utiliser les deux images dans leur globalité, et d'optimiser la mesure de similarité en fonction des paramètres de la transformation géométrique recherchée : lorsqu'un maximum est atteint, alors les deux images sont considérées comme recalées et l'algorithme donne en sortie les paramètres qui correspondent au maximum atteint. De plus, il existe des méthodes non paramétriques qui consistent à optimiser un champ de déplacements en étant guidé par une mesure de similarité, avec contraintes sur la régularité du champ (méthodes dites « variationnelles » dans lesquelles on optimise une fonctionnelle). L'une des mesures de similarité parmi les plus connues est « l'Information Mutuelle » [VIO97] : elle repose sur l'idée de mesurer une dépendance statistique entre deux signaux.

Les méthodes feature-based reposent quant à elles sur l'utilisation de singularités spatiales comme des points, des croisements, ou des structures plus complexes. Dans ce type d'approches, se pose la question de l'extraction de ces singularités, de leur bonne représentation dans les images, et plus encore de leur caractère « général », *ie.* de la possibilité

de pouvoir les extraire quelles que soient la situation et les particularités de la scène étudiée. Nous reviendrons sur les méthodes feature-based de manière plus approfondie dans le prochain chapitre de cette thèse. Ajoutons simplement que les méthodes feature-based peuvent être principalement répertoriées en deux classes: les méthodes qui utilisent les relations géométriques entre les singularités d'une part, et celles qui utilisent des relations structurelles d'autre part.

Le choix d'un modèle de transformation géométrique est important, puisque celui-ci est étroitement lié aux déformations que subissent les images. Les modèles de transformation se distinguent en deux catégories: d'une part les modèles globaux qui s'appliquent à la totalité de l'image esclave, et d'autre part les modèles locaux qui s'appliquent sur des morceaux de l'image esclave, pour lesquels les paramètres de la transformation varient à chaque fois. Par ailleurs, des modèles de transformation mixtes existent, où une partie seulement des paramètres sont fixés pour la totalité de l'image esclave. Parmi les modèles qu'il est courant d'utiliser en recalage d'images, citons:

- Les similarités qui consistent en la composition d'une translation dans le plan (t_x,t_y), d'une rotation d'angle et d'un changement d'échelle s (quatre paramètres),
- Les transformations affines qui forment un autre modèle de transformation légèrement plus général que les similarités (six paramètres),
- Le modèle de projection en perspective qui est utile dans le cas où le point de visée d'un capteur n'est pas au nadir « point situé à la verticale du capteur » (huit paramètres),
- Les modèles à base de fonctions radiales,
- etc...

Les modèles à base de fonctions radiales forment eux des exemples de modèles mixtes. Ils sont définis comme suit:

$$\begin{cases} x' = a_0 + a_1.x + a_2.y + \sum_{i=1}^{N} c_i^1 . g(r_i) \\ y' = b_0 + b_1.x + b_2.y + \sum_{i=1}^{N} c_i^2 . g(r_i) \end{cases} \quad (1.1)$$

avec g une fonction radiale et avec r_i égal à la distance entre la position courante (x,y) et la position du $i^{ème}$ point apparié de l'image maître (x_i,y_i); ce sont les fonctions g qui permettent des transformations géométriques locales, par oppositions aux paramètres (a_0,a_1,a_2) et (b_0,b_1,b_2). Plusieurs choix sont possibles pour la fonction g, dont certains sont résumés dans le tableau 1.1 ci-dessous:

Quadratique	Quadratique inverse	Gaussien	Thin plate spline
$(x\text{-}x_i)^2 + (y\text{-}y_i)^2$	$\frac{1}{(x-x_i)^2+(y-y_i)^2}$	$e^{\frac{(x-x_i)^2+(y-y_i)^2}{-2\sigma^2}}$	$[(x-x_i)^2+(y-y_i)^2]$. $\mathrm{Log}[(x-x_i)^2+(y-y_i)^2]$

Tableau 1.1. *Exemples de fonctions radiales [ZIT03].*

Les modèles à base de fonctions radiales permettent, de part leurs capacités à transformer les images aussi bien globalement que localement, d'obtenir des recalages généralement de bonnes précisions ; leur principal défaut réside dans ce que l'estimation des paramètres, au nombre de 2N+6, est très coûteuse en temps de calcul.

La transformation et le rééchantillonnage de l'image esclave constituent l'étape finale du processus de recalage. La transformation de l'image esclave peut être effectuée de deux façons différentes (voir figure 1.4):

- Transformation « forward » : Elle consiste à appliquer à chaque position de l'image esclave la ou les transformation(s) géométrique(s) particulière(s) (dans le cas de transformations par morceaux).

- Transformation « backward » : la ou les transformations inverses sont appliquées sur chaque position de l'image maître pour extraire la valeur des pixels correspondant de l'image esclave.

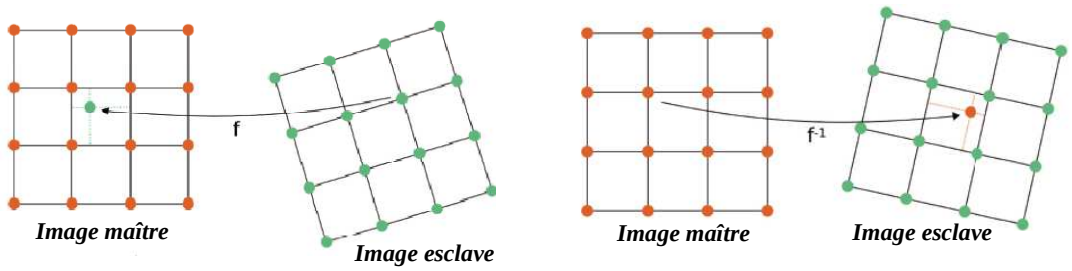


Figure 1.4. *Transformation forward (à g.), et backward (à d.).*

Le rééchantillonnage est nécessaire puisque dans un cas comme dans l'autre, les positions obtenues ne seront vraisemblablement pas de coordonnées entières (voir figure 1.4). Il s'agit alors d'interpoler l'image esclave transformée sur une grille régulière 2D ; pour cela, plusieurs fonctions peuvent être utilisées : plus proche voisin, interpolation bilinéaire, bicubique, spline, B-spline, gaussienne, etc...

1.2. Restriction du problème au cas d'images de télédétection de modalités différentes

Dans cette seconde section, notre but va être de circonscrire le problème de recalage que nous souhaitons traiter dans cette thèse. Pour cela, nous allons utiliser certaines connaissances liées au domaine de la télédétection qui vont nous permettre de restreindre notre champ d'investigation; ces restrictions sont au nombre de deux : l'une concerne le choix d'un modèle de transformation géométrique utile au recalage, tandis que l'autre concerne la nature de ce qui peut constituer une information commune aux images de différentes modalités. Celles-ci sont argumentées dans les deux sous-sections suivantes.

1.2.1. Choix d'un modèle de transformation géométrique

Le choix d'un modèle de transformations géométriques est important puisqu'il est étroitement lié aux déformations géométriques que subissent les images. En effet, l'ensemble des distorsions dans les images demandent autant de traitements particuliers, afin que les pixels appartenant aux régions déformées puissent être recalés. Par conséquent, des images ayant subi de nombreuses déformations géométriques réclament, pour être recalées entre elles, l'utilisation de modèles de transformations géométriques ayant de nombreux degrés de liberté: plus les déformations géométriques sont complexes, plus un modèle doit permettre des déformations locales, et plus le nombre de paramètres du modèle est important. Or lorsque l'espace de recherche (nombre de paramètres) devient grand, le problème du recalage se complexifie par la même occasion. Nous avons donc tout intérêt à utiliser des modèles de transformations géométriques les plus simples possibles, et donc à ce que les images soient au mieux corrigées de leurs déformations géométriques.

Les déformations que subissent les images sont très diverses, et dépendantes des modalités d'acquisition; néanmoins, nous pouvons les classer de la manière suivante:

- Déformations prévisibles: ce sont les déformations dites « systématiques », liées principalement au fonctionnement des capteurs.

- Déformations imprévisibles: ces déformations résultent de paramètres exogènes au système d'acquisition.

Étant donné que les déformations géométriques que subissent les images ne sont pas les mêmes d'un type de capteur à l'autre, il nous paraît important de donner quelques éléments supplémentaires quant à leurs fonctionnements. En plus des capteurs radars qui sont de type actif, il existe trois types de capteurs passifs (optiques) : « whiskbroom », « pushbroom » (voir figure 1.5) et les photographies aériennes.

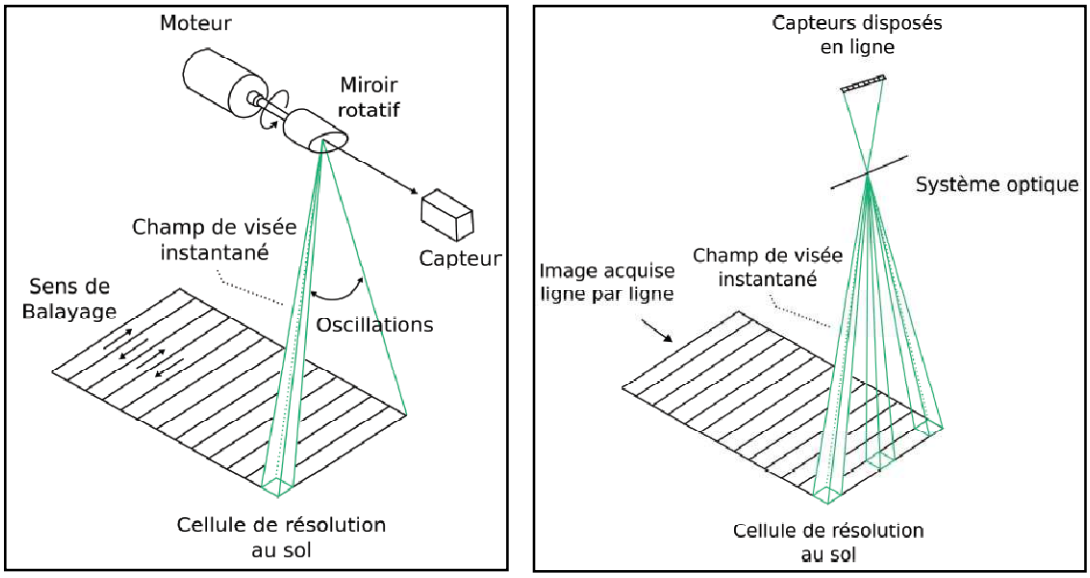


Figure 1.5. *Capteur whiskbroom (à g.) et pushbroom (à d.).*

Dans le cas d'un système whiskbroom [CHR06] (aussi appelé « across-track scanner »), un miroir rotatif permet à un seul capteur de scanner la surface du sol, par un mouvement qui oscille dans une direction perpendiculaire à la direction de déplacement de la plate-forme ; dans ce type d'acquisition, un seul pixel est enregistré à la fois. Dans le cas d'un système pushbroom [CHR06] (aussi appelé « along-track scanner »), ce n'est pas un capteur qui est utilisé, mais un ensemble de capteurs qui sont disposés de façon linéaire et perpendiculairement à la direction de déplacement de la plateforme; dans ce type d'acquisition, les images sont acquises ligne par ligne. Les systèmes pushbroom ont l'avantage d'être moins affectés par des déformations géométriques que les systèmes whiskbroom, mais nécessitent que les capteurs soient tous correctement calibrés en même temps. Avec leur unique capteur, les systèmes whiskbroom n'ont pas cet inconvénient; en revanche, ils sont d'une résolution radiométrique moindre à cause du peu de temps disponible durant le balayage pour effectuer chaque mesure. Dans le cas des photographies aériennes enfin, tous les pixels sont obtenus simultanément.

Nous allons maintenant porter notre attention sur les déformations prévisibles; sans être exhaustifs, notre but va être simplement de donner quelques exemples de telles déformations. Parmi les déformations prévisibles, celles qui sont les plus marquantes et qui concernent aussi bien les systèmes whiskbroom, pushbroom ou radar sont liées au caractère aéroporté des plates-formes qui les convoient. En effet, de nombreuses déformations géométriques proviennent du fait que les paramètres du vol peuvent changer durant les acquisitions, comme l'attitude (tangage, roulis, lacet), l'altitude ou la vitesse de déplacement. Dans la figure 1.6 ci-dessous, nous représentons les effets de telles variations sur des images non-corrigées.

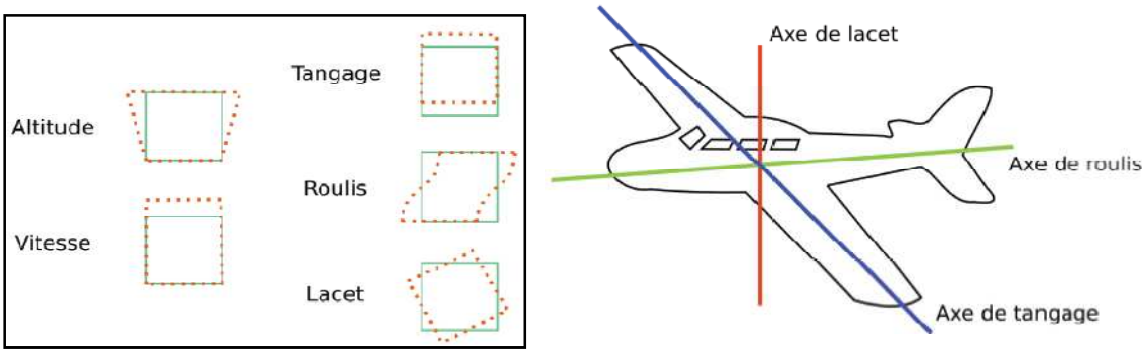


Figure 1.6. *Déformations causées par les changements de paramètres de vol des plates-formes*

On peut notamment remarquer les déformations importantes provoquées par le roulis de la plateforme, qui donnent une allure ondulée aux images. Parmi les autres exemples de déformations prévisibles, nous avons:

- La « distorsion tangentielle d'échelle » qui concerne les systèmes whiskbroom, et qui est causée par la vitesse de rotation constante du miroir : le champ de visée instantané du capteur se déplace alors plus rapidement lorsqu'il atteint les bords de l'image. Il en résulte que les cellules de résolution au sol ne sont pas distribuées uniformément dans la direction perpendiculaire au déplacement de la plate-forme. Cela provoque alors un effet de compression sur les bords de l'image, puisque les pixels sont, eux, positionnés uniformément.

- La déviation oblique des cellules de résolution au sol due au temps fini pour que les systèmes whiskbroom acquièrent une ligne de l'image alors que la plate-forme continue sa progression.

- Les effets de compression : les cellules de résolution au sol des systèmes dont la visée est large (systèmes optiques) sont d'une surface supérieure sur les bords de l'image, provoquant un effet de compression dans les images non-corrigées. Cet effet peut-être accentué par la courbure terrestre (systèmes spatioportés).

- La rotation terrestre (systèmes spatioportés), provoquant un déplacement des pixels vers l'Est.

- Etc...

Le point important à retenir est que toutes ces déformations peuvent être systématiquement corrigées, y compris les déformations complexes dues aux changements d'attitudes, de vitesse ou d'altitude des plates-formes aéroportées ; en effet, grâce aux INCS (« Inertial Navigation and Control System »), les changements des paramètres du vol sont enregistrés et peuvent être pris en compte lors du processus de correction des images [LIL04]. Nous considérerons donc

dans notre travail que les déformations prévisibles ont été éliminées des images devant être recalées.

Les déformations imprévisibles résultent de paramètres exogènes au système d'acquisition, principalement des effets de relief; en effet, ceux-ci provoquent des déformations parfois importantes et sans cohérence d'un type d'image à l'autre, modifiant angles et orientations par rapport à la réalité de la scène. La figure 1.7 illustre, dans le cas des photographies aériennes par exemple, comment se projette dans le plan-image un point P situé à une altitude h.

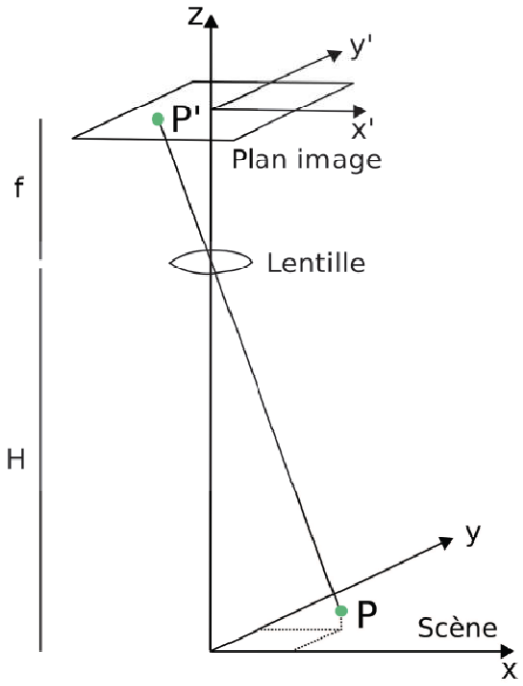


Figure 1.7. *Géométrie d'une photographie aérienne.*

Le centre optique est situé en (0,0,H), le point P est situé en (x,y,h) et apparaît dans l'image au point P' de coordonnées (x',y',H+f), où f est la distance focale. Nous avons les relations suivantes:

$$x' = \frac{-f.x}{H-h} \quad et \quad y' = \frac{-f.y}{H-h} \tag{1.2}$$

Ainsi, si le sommet d'une cible est situé à une altitude h quelconque et que sa base est située à une hauteur h=0, nous obtenons :

$$x'_{sommet} = \frac{-f.x}{H-h} \quad et \quad y'_{sommet} = \frac{-f.y}{H-h} \tag{1.3a}$$

$$x'_{base} = \frac{-f.x}{H} \quad et \quad y'_{base} = \frac{-f.y}{H} \tag{1.3b}$$

Base et sommet ne sont donc pas situés sur les mêmes pixels, le sommet ayant subi une homothétie dans le plan l'éloignant du point principal de l'image. Plus généralement, dans les photographies aériennes, les cibles situées au-dessus de l'altitude moyenne de la scène

subissent un déplacement radial qui les éloigne du point principal de l'image, tandis que les cibles situées en dessous subissent un déplacement radial dans la direction du point principal. Dans le cas des images whiskbroom/pushbroom, les déformations sont presque similaires, sauf que chaque ligne de l'image finale peut être considérée comme une « image » à part entière ; il n'y a donc pas de point principal unique dans l'image finale, mais un ensemble de points principaux qui s'organisent en ligne parallèlement à la direction de déplacement de la plate-forme. Par conséquent, dans les images whiskbroom/pushbroom, les déformations dues au relief sont unidimensionnelles et dans la direction perpendiculaire à la direction de déplacement de la plate-forme.

Les systèmes radars connaissent des déformations de relief différentes des systèmes optiques. Les radars enregistrent les distances entre les cibles et l'antenne en mesurant les temps de propagation des ondes réfléchies : on parle alors de « distance-temps ». La distance entre les cibles, la « distance-sol », ne peut être connue qu'indirectement par trigonométrie. La partie gauche de la figure 1.8 illustre la différence entre un arrangement des pixels en distance-temps et un arrangement des pixels en « distance-sol ».

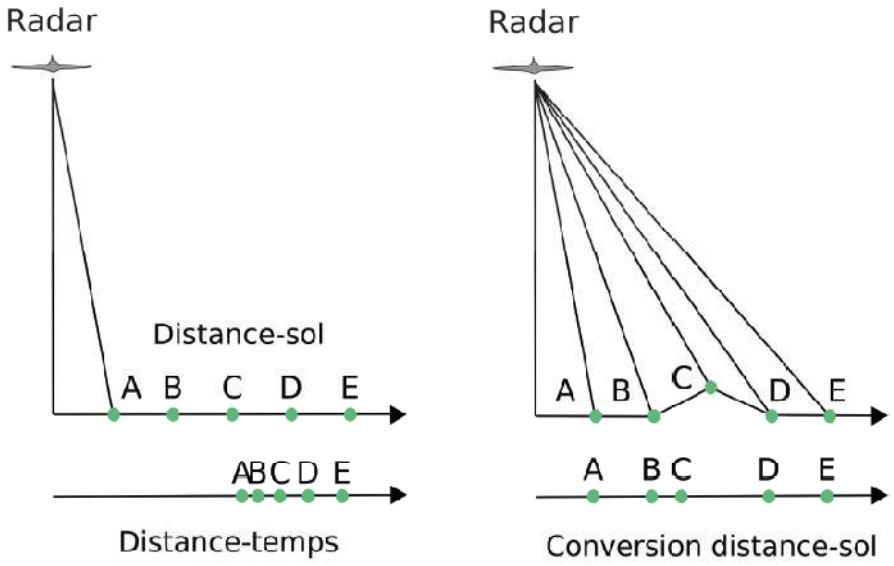


Figure 1.8. *Différence entre arrangements de pixels en distance-sol (à g.). Effets de relief lors de la conversion en distance-sol (à d.).*

Les effets de relief apparaissent alors lors de la conversion en distance-sol des images radars. La partie droite de la figure 1.8 illustre la situation pour un point situé en altitude: le point C est situé à une altitude légèrement supérieure à celles des autres points. Du fait de son élévation, la distance entre le point C et le capteur radar est plus court: lorsque l'arrangement des pixels d'une image radar est converti en distance-sol, le point C est alors bien plus proche du point B qu'il n'aurait dû l'être. C'est ainsi qu'en imagerie radar, les points surélevés subissent un déplacement apparent dans les images en direction du capteur, contrairement aux images optiques où ce déplacement se fait en sens opposé.

En définitive, toutes les déformations de relief pourraient être prévisibles (et donc corrigibles) si les différentes altitudes de la scène étaient connues. Ce type d'information pourrait être apporté par des Modèles d'Elévation Numérique (MEN) par exemple. Mais d'une part ces modèles ne sont pas toujours disponibles, et d'autre part cela nécessiterait le recalage des images avec les MEN relatifs aux scènes étudiées. Cela suggère qu'à défaut d'avoir les altitudes de la scène, il faut utiliser des modèles de transformations géométriques complexes pour faire face à n'importe quelle situation. Or un problème de recalage d'images peut tout à fait se traiter de manière hiérarchique. En effet, dans un premier temps, les images (ou leurs sous-parties) peuvent être recalées de façon globale à l'aide d'un modèle de transformations géométriques simple à faible nombre de paramètres, tandis que les régions les plus déformées peuvent être traitées dans un second temps (si cela est nécessaire) par des modèles ayant plus de degrés de liberté : la transformation finale sera alors formée par la composition des transformations. Mais au-delà d'un simple mécanisme analytique (composition de fonctions), décomposer un problème de recalage en l'estimation de transformations qui s'appliquent à des échelles différentes est favorable à la simplification du problème initial: en effet, même si une transformation globale ne peut recaler tous les éléments entre eux, elle fournit en revanche de précieux indices quant à la proximité des éléments restants à recaler, facilitant l'estimation des transformations plus fines utilisées en aval. La caractérisation d'une transformation géométrique globale se révèle donc une étape primordiale dans le processus de recalage. C'est sur cette première transformation, à la racine du processus de recalage, que nous allons nous concentrer dans cette thèse, en nous efforçant d'en donner une estimation sans que celle-ci soit perturbée par les déformations restantes.

Ainsi nous choisissons comme modèle de transformation géométrique celui de la transformation rigide, qui consiste en la composition d'une rotation et d'une translation dans le plan ; dans le cas de changements de résolution entre les capteurs, il sera tenu compte en plus d'une homothétie: le modèle de transformation géométrique correspondra alors à une similarité. Ce modèle, simple, nous permettra de faire face à la plupart des situations de recalage en télédétection. Par exemple, ce choix a été adopté dans les travaux suivants: [LI95], [HSI97], [WAN97], [ZHA04], [CUI07], [SHA07], [LI07], [CUI09].

1.3. Principe du recalage géométrique d'images

Les algorithmes de recalage d'images, déterminent la relation géométrique (transformation spatiale) liant deux images contenant des informations communes. En d'autres termes, le recalage d'images est l'opération qui situe deux images ou plus contenant des informations homologues, dans un même repère géométrique. Ces images peuvent être représentées en 2D (le type d'image d'intérêt dans cette thèse) ou en 3D. La figure 1.9 montre une représentation simplifiée du lien entre deux images 2D issues d'une même scène 3D.

Si les images à recaler sont issues de capteurs différents (par exemple des images IRM et des images ultrasonographiques), alors le recalage est dit multimodal. En revanche, des images issues d'un même type de capteur font appel à du recalage monomodal.

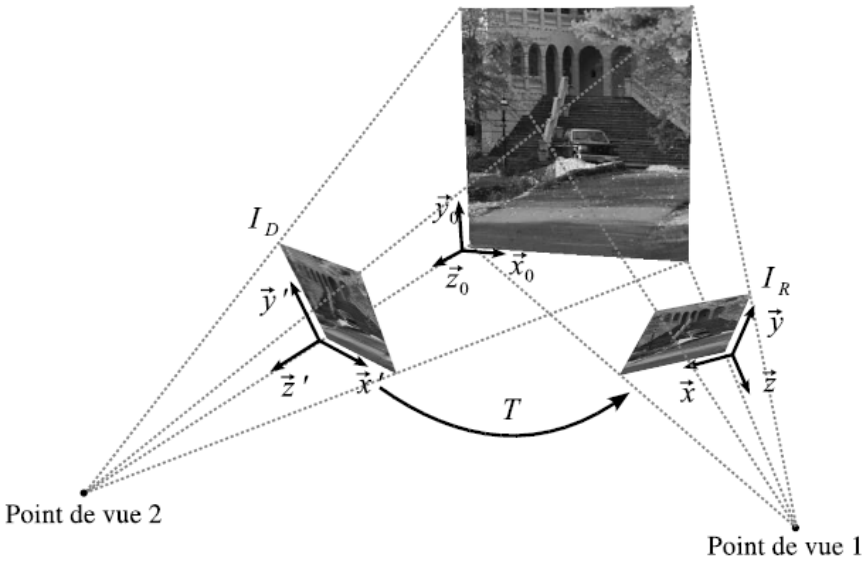


Figure 1.9. *Représentation simplifiée de l'espace T de transformation 2D–2D pour le recalage entre deux images d'une même scène prises pour deux points de vue différents. T représente un changement de perspective.*

Les applications du recalage d'images sont très vastes. Il est utilisé dans la vision stéréoscopique, la reconnaissance de formes, dans le suivi d'objets, le mosaïquage d'images, l'indexation de bases d'images, etc. Étant donné le très vaste champ applicatif du recalage d'images, nous allons aborder seulement quelques exemples significatifs (pour notre travail) du recalage d'images dans le domaine médical. Des aperçus très complets sur les techniques de recalage dans des divers domaines peuvent être trouvées dans [BRO92] et [ZIT03], et dans [PLU03] pour des applications dans le domaine médical.

1.4. Définition mathématique du recalage d'images

Le processus de recalage de deux images peut être formulé mathématiquement comme suit :

$$\tilde{T} = \underset{T}{argopt} \sum_X sim[g(I_R(X)), \Omega(g(I_D(T(X))))] \qquad (1.4)$$

Dans l'équation (1.4), $\boldsymbol{I_D}$ est l'image à recaler avec $\boldsymbol{I_R}$ (respectivement, image à transformer ou source et image de référence ou cible). $\tilde{T}$ est la transformation qui va amener les coordonnées des pixels de l'image I_D dans le système de coordonnées de l'image I_R. Les valeurs des paramètres de cette transformation sont actualisées par une méthode d'optimisation, ***argopt***, qui cherche à maximiser ou minimiser une mesure de similarité, ***sim***. Cette dernière est calculée soit à partir de primitives images détectées par une fonction g, soit directement avec les niveaux de gris des images. Dans ce dernier cas, aucune fonction $\boldsymbol{g}$ n'est appliquée. Cet algorithme d'optimisation cherche les valeurs des paramètres de la transformation dans un espace de recherche donné par $\boldsymbol{T}$. Ω est la méthode d'interpolation associée à la transformation $\boldsymbol{T}$ qui sera appliquée à l'image $\boldsymbol{I_D}$. $\tilde{T}$ est la transformation qui optimise ***sim*** et qui superpose au mieux les parties communes des images.

Pour notre étude, nous avons séparé les méthodes de recalage en deux classes: les méthodes qui n'utilisent pas la fonction g, que nous allons appeler méthodes de recalage basées sur les niveaux de gris des images, et les méthodes qui en font usage, appelées dans notre document méthodes basées sur la détection de primitives images.

Même s'il est possible de reconnaître tous les éléments mathématiques dans les deux approches de recalage d'images, certains de ces éléments sont utilisés de façon différente. C'est pour cette raison que nous allons d'abord présenter les éléments du recalage d'images qui sont communs aux deux approches: l'espace de recherche et l'interpolation des images transformées, pour ensuite présenter l'utilisation que chaque approche fait du reste des éléments.

1.4.1. Espace de recherche (types de transformation)

Le choix d'un modèle de transformation géométrique est important, puisque celui-ci est étroitement lié aux déformations que subissent les images. Les modèles de transformation se distinguent en deux catégories: d'une part les modèles globaux qui s'appliquent à la totalité de l'image cible, et d'autre part les modèles locaux qui s'appliquent sur des morceaux de l'image cible, pour lesquels les paramètres de la transformation varient à chaque fois. Par ailleurs, des modèles de transformation mixtes existent, où une partie seulement des paramètres sont fixés pour la totalité de l'image cible. Parmi les modèles qu'il est courant d'utiliser en recalage d'images, citons la *transformation projective* à 8 degrés de liberté (1 axe de rotation dans le plan image, 2 directions de cisaillement (shearing), 2 rotations en perspective dites hors plan, 2 translations et 1 changement d'échelle isotrope). Néanmoins, pour des mouvements particuliers (plus limités) de la caméra, la mise en correspondance entre les vues de la scène peut être effectuée en utilisant des transformations plus simples. Ces transformations ont moins de degrés de liberté et par conséquent, sont représentées par moins de paramètres. Les transformations géométriques élémentaires (translations, rotations et changements d'échelle) peuvent être combinées pour obtenir des transformations plus ou moins complexes avec différents degrés de liberté.

Pour le texte suivant, les coordonnées homogènes sont représentées par le symbole ˜ (par exemple $\tilde{x}$). Les coordonnées standards n'ont pas de˜ (tilde).

- ***Translations:*** Si le mouvement de la caméra consiste seulement en des translations parallèles au plan d'acquisition, le rapport entre les images peut être représenté par une transformation à 2 degrés de liberté. Ce modèle est le plus simple. Ces translations (x'= x + **t**) peuvent être exprimées sous forme matricielle dans un espace projectif :

$$X' = [I \quad t]\,\tilde{X} \tag{1.5}$$

où ***I*** est la matrice unité de taille 2×2, t = $(t_x, t_y)^T$ est le vecteur des translations et $\tilde{X}=(x, y, 1)^T$ est un vecteur 2D en coordonnées homogènes.

- ***Transformation rigide:*** Un mouvement de la caméra entraînant des translations parallèles au plan de la scène et une rotation autour de l'axe perpendiculaire à ce plan, peut être modélisé par une matrice de transformation rigide ou euclidienne (ainsi nommée car les distances euclidiennes, le parallélisme et les angles sont préservés) à 3 degrés de liberté. Elle peut être exprimée sous la forme x'= **R**x + **t** ou de façon matricielle :

$$X' = [\mathrm{R} \quad t]\,\tilde{X} \tag{1.6}$$

Où $$R = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

$\boldsymbol{\theta}$ est l'angle de rotation dans le plan. R est une matrice orthonormale de rotation avec $RR^T = I$ et $|R| = 1$.

- ***Similitude:*** Si des translations le long de l'axe perpendiculaire au plan de la scène (qui se traduisent par des changements d'échelle de l'image) sont ajoutées à la transformation euclidienne, elle devient une similitude telle que x'= **sR**x + **t**, où **s** est un facteur d'échelle. Elle peut aussi être écrite comme suit

$$X' = [\mathbf{s}\mathrm{R} \quad \mathbf{t}]\,\tilde{X} = \begin{bmatrix} a & -b & t_x \\ b & a & t_y \end{bmatrix} \tag{1.7}$$

où $a = s\cos(\theta)$ et $b = s\sin(\theta)$.

Cette transformation à 4 degrés de liberté (2 translations, 1 axe de rotation et 1 changement d'échelle isotrope) préserve les angles et le parallélisme entre les lignes. Des applications dans le filigranage d'images [LIN01], dans l'imagerie satellitaire [CHE94] et dans la photographie aérienne [XIE03] ont été développées à l'aide de ce type de transformation.

- ***Transformation affine:*** Pour cette transformation, le changement d'échelle est asymétrique et donc, un effet de « cisaillement »apparaît. La transformation est définie par la matrice (de taille 2×3) à 6 degrés de liberté :

$$X' = [\mathbf{A} \quad \mathbf{t}]\,\tilde{X} = \begin{bmatrix} a_{00} & a_{01} & t_x \\ a_{10} & a_{11} & t_y \end{bmatrix} \tilde{X} \tag{1.8}$$

La matrice A peut être représentée par une combinaison de matrices de rotation et une matrice de facteurs d'échelle asymétriques [HAR04] :

$$A = R(\theta)R(-\varphi)\mathrm{DR}(\varphi) \tag{1.9}$$

avec θ l'angle de rotation de l'image, φ l'angle de déformation (Figure 1.10) et

$$\mathrm{D} = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \tag{1.10}$$

où λ_1 et λ_2 sont les facteurs d'échelle asymétriques.

Cette transformation préserve le parallélisme et les rapports de longueurs entre les lignes ainsi que les rapports de surfaces.

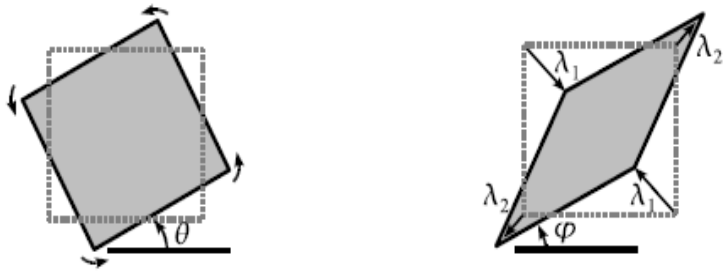


Figure 1.10. *Déformations issues d'une transformation affine (d'après [HAR04]). (a) Rotation d'angle θ. (b) Déformation* ***R****(-φ)****DR****(φ). Il faut noter que les directions des changements d'échelle (*λ_1 *et* λ_2*) sont orthogonales.*

Des exemples d'application de ce modèle de transformation sont le recalage de contours et d'images échocardiographiques [DYD03], le recalage d'images CT–T2 [ZHA04] ou le recalage de documents photographiés à la main avec un appareil photo [SAT04].

- ***Transformation projective:*** Cette transformation est aussi appelée transformation perspective ou homographie. C'est un modèle de transformation à 8 degrés de liberté. Ceci implique des mouvements de translation, de rotations, de changements d'échelle, ainsi que des changements de perspective de la caméra (rotations hors plan). Elle peut être exprimée comme suit: $\widetilde{X}' \sim \widetilde{H}\,\widetilde{X}$, où $\widetilde{H}$ est une matrice de taille 3×3. Celle-ci peut être écrite aussi comme suit :

$$\tilde{X} = \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & h_{22} \end{bmatrix} \tilde{X} \qquad (1.11)$$

La matrice de transformation projective **H** peut être décomposée en une série de transformations [HAR04] :

$$\mathrm{H} = \mathrm{H_S H_A H_P} = \begin{bmatrix} \mathrm{sR} & \mathrm{t} \\ 0^{\mathrm{T}} & 1 \end{bmatrix} \begin{bmatrix} \mathrm{K} & 0 \\ 0^{\mathrm{T}} & 1 \end{bmatrix} \begin{bmatrix} \mathrm{I} & 0 \\ \mathrm{v}^{\mathrm{T}} & \mathrm{v} \end{bmatrix} = \begin{bmatrix} \mathrm{A} & \mathrm{t} \\ \mathrm{v}^{\mathrm{T}} & \mathrm{v} \end{bmatrix} \qquad (1.12)$$

avec **A** une matrice non-singulière donnée par **A** = **sRK** + **tv**$^{\mathbf{T}}$ et **K** une matrice triangulaire supérieure normalisée de telle façon que det(**K**) = 1. **R** est encore une matrice de rotation dans le plan, s est un facteur d'échelle, t est un vecteur translation et $v = (v_1, v_2)^{\mathrm{T}}$ est un vecteur contenant les deux premiers éléments de normalisation de la matrice de transformation perspective (h_{20}, h_{21}). Cette composition est valide pour $v \neq 0$ et elle est unique si ***s*** est positif.

Les coordonnées homogènes résultantes doivent être normalisées pour obtenir des coordonnées « inhomogènes »:

$$x' = \frac{h_{00}\,x + h_{01}y + h_{02}}{h_{20}\,x + h_{21}\,y + h_{22}} \quad \text{et} \quad y' = \frac{h_{10}\,x + h_{11}y + h_{12}}{h_{20}\,x + h_{21}\,y + h_{22}} \tag{1.13}$$

Les transformations projectives préservent les lignes droites. Ce modèle est utilisé lorsque la caméra effectue des mouvements libres dans un repère tridimensionnel, comme par exemple lors de la prise d'une vidéo [IRA96], dans le cas du mosaïquage d'images issues d'exploration sous-marine [GRA05] ou pour des photographies prises avec un appareil photo déplacé manuellement [ZOM00].

Le tableau ci-dessous donne un résumé des modèles de transformation décrits précédemment.

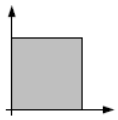
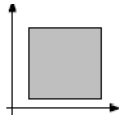
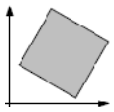
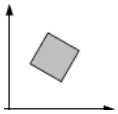
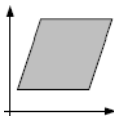
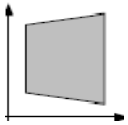
Transformation	Matrice	Effet	Invariants
Identité 0 degré de liberté	$\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}$		
Translation 2 degrés de liberté	$\begin{bmatrix}1 & 0 & t_x\\ 0 & 1 & t_y\end{bmatrix}$		Orientation, distances, angles, parallélisme, lignes droites
Rigide 3 degrés de liberté	$\begin{bmatrix}\cos\theta & -sin\theta\\ sin\theta & cos\theta\end{bmatrix}$		Distances, angles, parallélisme, lignes droites
Similitude 4 degrés de liberté	$\begin{bmatrix}\boldsymbol{s}\cos\theta & -\boldsymbol{s}\,sin\theta\\ \boldsymbol{s}\,sin\theta & \boldsymbol{s}\,cos\theta\end{bmatrix}$		Angles, parallélisme, lignes droites
Affine 6 degrés de liberté	$\begin{bmatrix}a_{00} & a_{01} & a_{02}\\ a_{10} & a_{11} & a_{12}\end{bmatrix}$		Parallélisme, lignes droites
Projective 8 degrés de liberté	$\begin{bmatrix}h_{00} & h_{01} & h_{02}\\ h_{10} & h_{11} & h_{12}\\ h_{20} & h_{21} & h_{22}\end{bmatrix}$		Lignes droites

Tableau 1.2. *Modèles de transformations géométriques linéaires en 2D. Toutes les transformations comprennent les paramètres de la transformation précédente à laquelle nous rajoutons de nouveaux paramètres. Les paramètres t_x, t_y, s, θ sont expliqués dans le texte.*

1.4.2. Interpolation des niveaux de gris des images transformées

Un pixel dont les coordonnées sont entières dans l'image à transformer, ne sera pas forcément placé sur des coordonnées entières dans l'image transformée. Par ailleurs, un pixel de l'image à transformer ne correspond pas à un seul pixel de l'image transformée (et inversement). Pour ces raisons, il n'est pas suffisant de recopier les niveaux de gris de l'image à transformer vers l'image transformée.

Lorsqu'une transformation est appliquée à une image, deux méthodes peuvent être utilisées pour déterminer les valeurs d'intensité des pixels de l'image transformée: la méthode directe et la méthode inverse. Pour les deux méthodes, la première étape consiste à appliquer la transformation aux coordonnées minimum et maximum de l'image (les quatre coins). De cette façon, la surface du polygone à quatre côtes (bounding box) sur laquelle l'image transformée va être « reproduite » est connue. Si la méthode directe est utilisée, la transformation est appliquée à toutes les coordonnées des pixels de l'image originale (image source) et le niveau de gris des pixels originaux est copié dans l'image transformée sur les pixels dont les coordonnées sont les entiers les plus proches des coordonnées calculées. Cette opération est illustrée dans la figure 1.11(a). Il existe deux problèmes associés à l'utilisation de cette technique. D'une part, lorsque la transformation est appliquée à l'image originale, certains pixels de l'image transformée peuvent ne correspondre à aucun pixel dans l'image originale. Cela fait apparaître des trous de niveaux de gris dans l'image transformée. D'autre part, un pixel dans l'image transformée peut correspondre à plusieurs pixels dans l'image originale. Donc, les valeurs d'intensité des pixels vont se superposer à une même position dans l'image transformée. La méthode directe ne permet pas d'affecter une et une seule valeur de niveaux d'intensité à la totalité des pixels de l'image transformée.

En raison des inconvénients liés à l'utilisation de la méthode directe, la méthode inverse est généralement préférée. Pour cette méthode, une fois l'extension de l'image transformée connue, la transformation inverse $\boldsymbol{T^{-1}}$ est appliquée à toutes les coordonnées entières des pixels de l'image transformée pour calculer, pour chaque pixel, le niveau de gris correspondant à cette position dans l'image originale. Comme il est probable que cette position ne soit pas entière, il faut alors calculer la valeur de ce pixel par interpolation des valeurs des pixels voisins. Le principe de la méthode inverse est montré dans la figure 1.11(b). Cette méthode garantit l'absence de trous dans l'image transformée.

Pour calculer la valeur du pixel dans l'image transformée, une méthode d'interpolation qui prend en compte les valeurs d'intensité des pixels dans le voisinage du pixel correspondant dans l'image source est utilisée, les plus connues étant les interpolations *par le plus proche voisin*, *bilinéaire* et *bicubique.*

- ***Interpolation par le plus proche voisin:*** Cette méthode d'interpolation est la plus simple. Ici, la valeur du pixel cherchée est obtenue, comme son nom l'indique, en prenant la valeur du pixel qui sera le plus proche de la position calculée sur l'image originale.

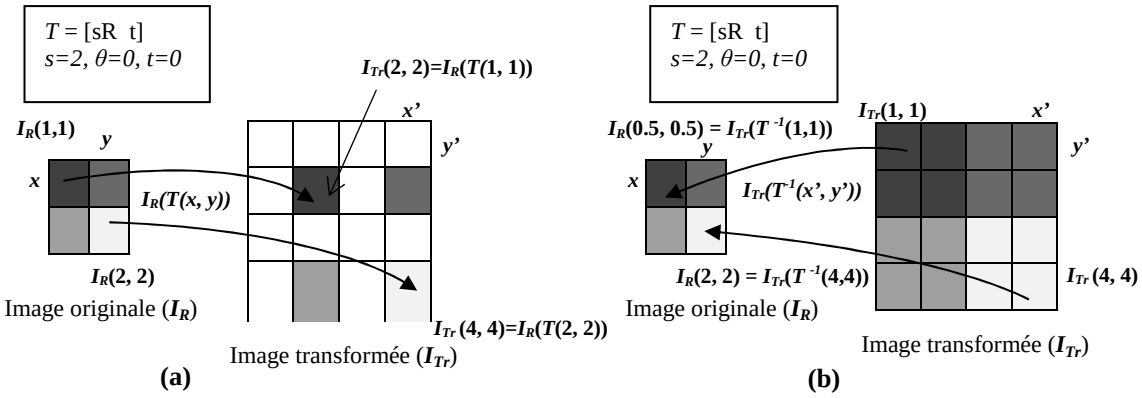


Figure 1.11. *Illustrations des deux approches possibles pour transformer une image : directe (a) et inverse (b). I_R est l'image originale et I_{Tr} est l'image transformée. T est la transformation appliquée aux coordonnées des images. La transformation prise comme exemple ici est une similarité qui va multiplier par deux la taille de l'image (s = 2). (a) Transformation de l'image en appliquant la méthode directe. (b) Transformation par la méthode inverse avec une interpolation par le plus proche voisin.*

- ***Interpolation bilinéaire***: Cette méthode considère un voisinage de taille 2 × 2 pixels autour de la coordonnée d'intérêt calculée (figure 1.12). Une moyenne pondérée est donc calculée avec les valeurs d'intensité de ces 4 pixels. C'est une extension de l'interpolation linéaire au cas bi-dimensionnel. La méthode consiste à appliquer une interpolation linéaire d'abord dans une direction et ensuite dans l'autre. Si les quatre valeurs des pixels les plus proches autour de la position d'intérêt sont connues, la valeur du pixel de coordonnées (p_x, p_y) dans l'image transformée peut être calculée comme suit. Soit (x, y) les coordonnées entières du pixel situé immédiatement en haut et à gauche du pixel de coordonnées non entières ramené de l'image transformée dans l'image à transformer avec T^{-1}(figure 1.12). L'interpolation selon l'axe $\vec{x}$ est effectuée avec la relation (1.14).

$$I_R(p_x, y) = (1-a)I_R(x, y) + aI_R(x+1, y)$$
$$I_R(p_x, y+1) = (1-a)I_R(x, y+1) + aI_R(x+1, y+1) \quad (1.14)$$

où *a* est la distance, dans la direction $\vec{x}$, du point dont l'intensité doit être calculée à la position du pixel pris comme référence (*x*, *y*). Avec ces deux valeurs, il est maintenant possible de calculer le niveau de gris $I_R(p_x, p_y)$ du point de coordonnées entières (p_x, p_y) dans l'image transformée en interpolant les niveaux de gris selon $\vec{y}$:

$$I_R(p_x, p_y) = (1-b)I_R(p_x, y) + bI_R(p_x, y+1) \quad (1.15)$$

où b est la distance, dans la direction $\vec{y}$.

Le résultat de cette méthode est visuellement beaucoup plus lisse (moins pixelisé) que celui obtenu par la méthode du plus proche voisin.

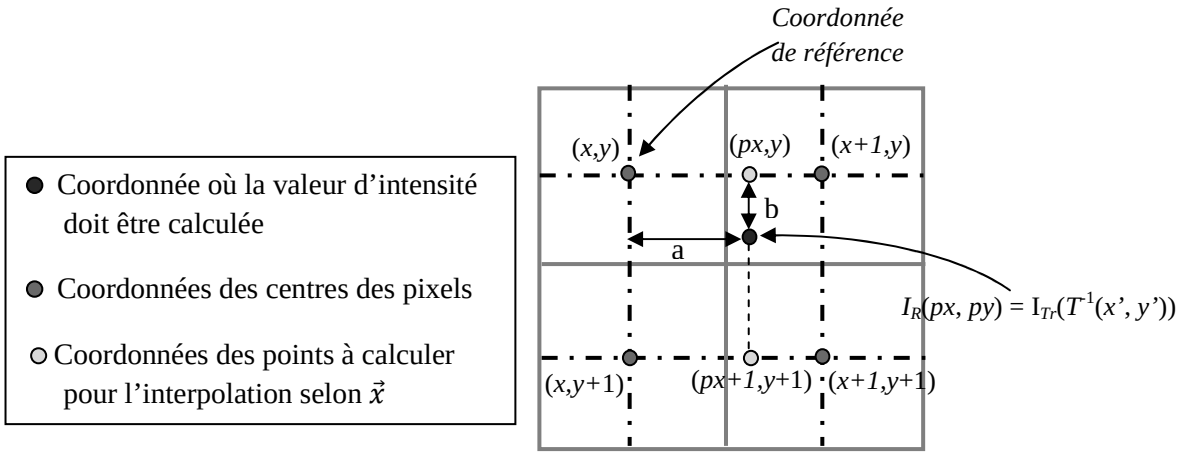


Figure 1.12. *Interpolation bilinéaire.*

- ***Interpolation bicubique***: Cette méthode prend en compte un voisinage de taille 4×4. Les seize pixels les plus proches du pixel dont l'intensité doit être calculée définissent ce voisinage. Comme montré dans la figure 1.13, une moyenne pondérée des seize niveaux de gris est calculée en utilisant un polynôme du 3ème degré. Comme pour l'interpolation bilinéaire, une première interpolation cubique est appliquée d'abord selon une première direction et ensuite selon l'autre en considérant cette fois-ci quatre pixels par ligne. Les coordonnées entières (x, y) sont encore celles du pixel situé immédiatement en haut et à gauche du pixel de coordonnées (p_x, p_y) dont le niveau de gris $I_R(p_x, p_y)$ doit être calculé, a est la distance selon $\vec{x}$ entre les points de coordonnées (x, y) et (p_x, p_y). L'interpolation selon $\vec{x}$ est calculée comme suit :

$$I_R(p_x, y) = -a(1-a)I_R(x-1, y) + (1-2a^2+a^3)I_R(x, y) + a(1+a-a^2)I_R(x+1, y) - a^2(1-a)I_R(x+2, y)$$

$$I_R(p_x, y-1) = -a(1-a)I_R(x-1, y-1) + (1-2a^2+a^3)I_R(x, y-1) + a(1+a-a^2)I_R(x+1, y-1) - a^2(1-a)I_R(x+2, y-1) \quad (1.16)$$

$$I_R(p_x, y+1) = -a(1-a)I_R(x-1, y+1) + (1-2a^2+a^3)I_R(x, y+1) + a(1+a-a^2)I_R(x+1, y+1) - a^2(1-a)I_R(x+2, y+1)$$

$$I_R(p_x, y+2) = -a(1-a)I_R(x-1, y+2) + (1-2a^2+a^3)I_R(x, y+2) + a(1+a-a^2)I_R(x+1, y+2) - a^2(1-a)I_R(x+2, y+2)$$

Avec les quatre valeurs obtenues, la valeur de $I_R(p_x, p_y)$ peut être calculée comme suit:

$$I_R(p_x, p_y) = -b(1-b)I_R(p_x, y-1) + (1-2b^2+b^3)I_R(p_x, y) + b(1+b-b^2)I_R(p_x, y+1) - b^2(1-b)I_R(p_x, y+2) \quad (1.17)$$

La distance b est définie sur la figure 1.13 (ci-dessous).

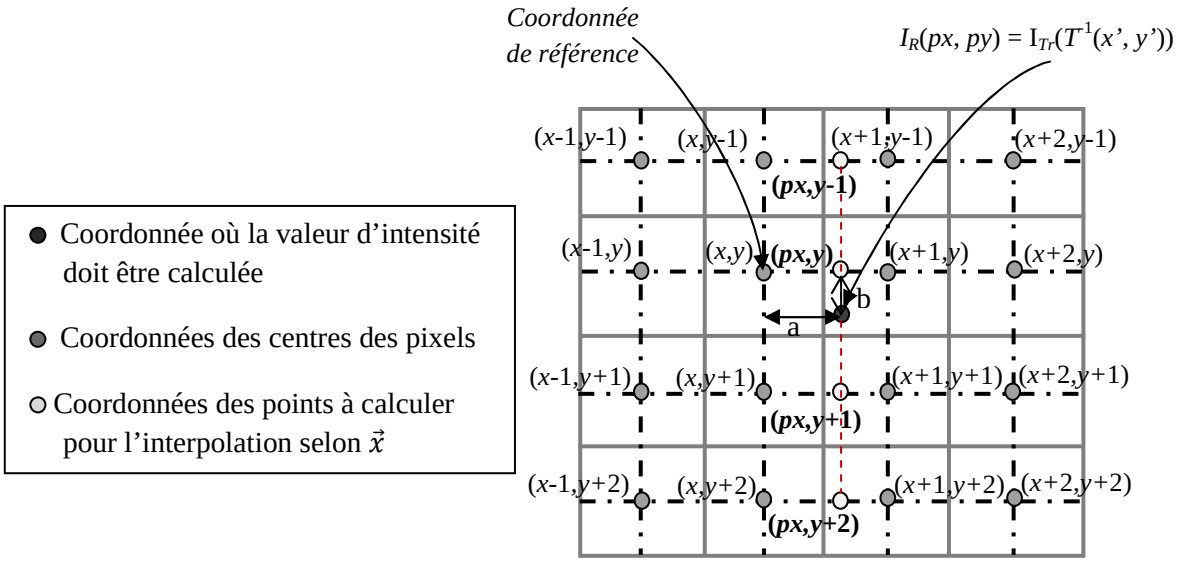


Figure 1.13. *Principe de l'Interpolation bicubique.*

La figure 1.14 montre un exemple du résultat de l'application des méthodes d'interpolation décrites ci-dessus. L'interpolation bicubique fournit un rendu visuel légèrement amélioré par rapport à l'interpolation bilinéaire. Cependant, elle requiert beaucoup plus de temps de calcul que cette dernière. Pour la plupart des applications de recalage d'images, l'interpolation bilinéaire offre le meilleur compromis entre rendu visuel et temps de calcul. L'interpolation par le plus proche voisin peut être préférée si les valeurs d'intensité originales doivent absolument être préservées.

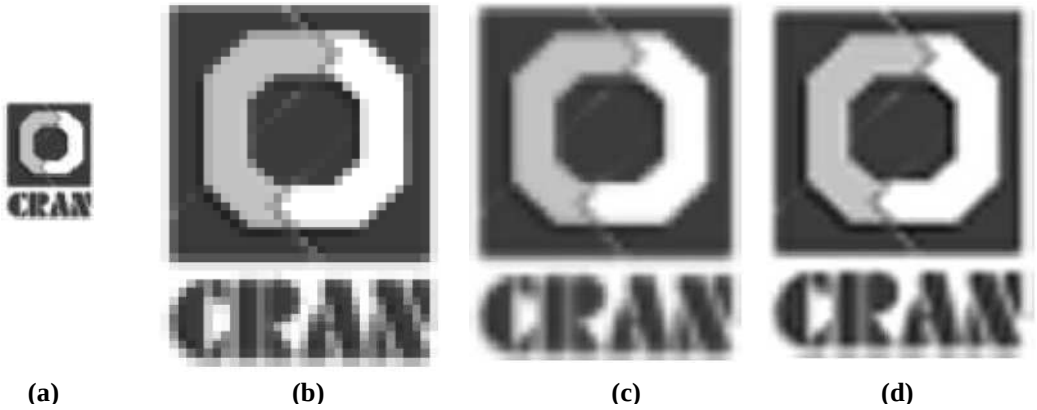


Figure 1.14. *Résultat des différentes méthodes d'interpolation pour une transformation constituée d'un changement d'échelle d'un facteur 6. (a) Image originale. (b) Interpolation de (a) par le plus proche voisin. (c) Interpolation bilinéaire de (a). (d) Interpolation bicubique de (a).*

1.5. Recalage basé sur les niveaux de gris des images

Pour définir une méthode de recalage basée sur les niveaux de gris des images, il faut d'abord choisir une mesure de similarité permettant effectivement de comparer les images. Cette mesure sera sélectionnée en tenant compte des caractéristiques des images (mono ou multimodalités, bruits présents sur les images, déformations géométriques, etc). Une fois la mesure de similarité choisie, une technique de recherche de la similarité optimale doit être mise en place. Pour mesurer le degré de similarité de deux images, des méthodes basées sur la comparaison directe des niveaux d'intensité, la corrélation des images, la transformée de fourier ou des fonctions statistiques ont été proposées. Ces techniques sont décrites ci-dessous.

1.5.1. Mesures de similarité

a. Différences de niveau de gris: La façon la plus simple de recaler deux images est de comparer directement leurs niveaux de gris pixel par pixel. Soit I_R(x) l'image de référence et I_D(x) l'image à recaler sur I_R, avec x = $(x, y)^T$. Pour effectuer le recalage, nous pouvons calculer le minimum de la Somme des Différences au Carré, (SDC), entre les images,

$$SDC = \sum_X [I_D(T(X)) - I_R(X)]^2 \tag{1.18}$$

où T est la transformation appliquée aux coordonnées de l'image source I_D(x). Ce genre de mesure est bien adapté à des applications où les images sont issues d'un même capteur (monomodalité) avec des conditions d'éclairage très proches entre les images à recaler.

b. Corrélation croisée: Une alternative à la différence d'intensité entre les images est l'utilisation d'une corrélation croisée qui permettra de maximiser pixel à pixel le produit de deux images,

$$CC = \sum_X I_R(\mathrm{x})\, I_D(T(\mathrm{x})) \tag{1.19}$$

Un problème lié à la corrélation est que si une région dans l'image I_D a des valeurs d'intensité très supérieurs au reste de l'image, il est très probable que le maximum de la fonction de corrélation sera faussé par cette partie de l'image. C'est pour cela, qu'en général la corrélation croisée normalisée est préférée:

$$CCN = \frac{\sum_{\mathrm{X}}[I_R(\mathrm{x}) - \hat{I}_R][I_R(T(\mathrm{x})) - \hat{I}_D]}{\sqrt{\sum_{\mathrm{X}}[I_R(\mathrm{x}) - \hat{I}_R]^2 [I_D(T(\mathrm{x})) - \hat{I}_D]^2}} \tag{1.20}$$

où

$$\hat{I}_R = \frac{1}{N}\sum_{\mathrm{x}} I_R(\mathrm{x}) \quad \text{et} \quad \hat{I}_D = \frac{1}{N}\sum_{\mathrm{x}} I_D(T(\mathrm{x})) \tag{1.21}$$

sont les moyennes des images et *N* est le nombre de pixels dans les images. Le résultat de la corrélation croisée normalisée est toujours compris entre -1 et 1. Cependant, cette mesure est non définie si l'une des images a une variance égale à zéro. Cette mesure de similarité est souvent utilisée pour aligner des images dont le lien est décrit par des translations 2D, mais peut également être utilisée s'il existe des variations légères de rotation et de changement d'échelle. Cette mesure de similarité est utilisée lors du recalage d'images monomodalité. La CCN est moins sensible aux changements d'illumination que la SDC.

c. Recalage basé sur la transformée de Fourier*:** Lorsque entre deux images, les composantes dominantes de la transformation sont des translations, une approche basée sur la transformée de Fourier peut être suffisante. Le principe de cette technique repose sur la propriété de translation de la transformée de Fourier. En effet, la transformée de Fourier ***F d'une image $\boldsymbol{I_D(x + t)}$ ayant subie une translation $\mathbf{t= (}t_x, t_y\mathbf{)}$, aura la même image des modules que celle de l'image originale $I_D(x)$, mais l'image des phases variera proportionnellement avec le déplacement,

$$\mathcal{F}\{I_D(x+t)\} = F\{I_D(x)\}e^{-2\pi jtf} = \mathfrak{T}_D(f)e^{-2\pi jtf} \qquad (1.22)$$

où $f = (u, v)^T$ est le vecteur des fréquences spatiales de l'image et $\mathfrak{T}_D(f)$ dénote la transformée de Fourier de l'image $I_D(x)$.

Une autre propriété utile de la transformée de Fourier est qu'une convolution dans le domaine spatial correspond à une multiplication dans le domaine fréquentiel (théorème de Plancherel). Donc, la transformée de Fourier de la corrélation croisée des images peut être exprimée comme suit :

$$\mathcal{F}\{\sum_{x} I_R(x)\, I_D(x+t)\} = \mathcal{F}\{I_R(t) \otimes I_D(t)\} = \mathfrak{T}_R(f)\mathfrak{T}_D{}^*(f) \qquad (1.23)$$

où ⊗ dénote l'opération de convolution et $\mathfrak{T}_D{}^*(f)$ est le conjugué complexe de $\mathfrak{T}_D(f)$. En résumé, pour évaluer de façon efficace la corrélation croisée entre les images sur toute la gamme de valeurs possibles de t, la transformée de Fourier des images $I_R(x)$ et $I_D(x+t)$ est calculée, pour ensuite, effectuer le produit de la transformée de la première image par le conjugué de la deuxième image. Finalement, la transformée de Fourier inverse est calculée. L'algorithme de la transformée rapide de Fourier (FFT, Fast Fourier Tansform) permet de calculer la transformée d'une image de taille N × M en O(*NM log NM*) opérations (N et M étant des puissances de 2). Ceci est sensiblement plus rapide que les $O(N^2M^2)$ opérations requises par une recherche exhaustive dans les parties communes des images à recaler.

d. Corrélation des phases: Une variante de la corrélation croisée dans le domaine de Fourier, aussi utilisée pour l'estimation du déplacement entre images, est la corrélation des phases (CC_{CP}). Dans cette méthode, le spectre des deux images est « nettoyé » en divisant le produit des transformées (équation 1.23) par les modules des transformées avant d'appliquer la transformée inverse de Fourier,

$$CC_{CP}(\mathrm{t}) = \mathrm{F}^{-1}\left\{\frac{\mathfrak{T}_R(f)\mathfrak{T}_D{}^*}{\|\mathfrak{T}_R(f)\|\|\mathfrak{T}_D(f)\|}\right\} \tag{1.24}$$

Dans l'idéal, le résultat est une impulsion unique localisée à la valeur correcte de t. Cette caractéristique simplifierait l'estimation du déplacement.

e. Transformée de Fourier-Mellin: Même si le recalage basé sur la transformée de Fourier est utilisé pour estimer des déplacements entre les images, il peut être utilisé aussi, sous certaines conditions, pour calculer des rotations et des changements d'échelle isotropes sans aucun déplacement. D'abord, il faut estimer uniquement la rotation entre les images. Si le lien géométrique entre deux images est une rotation $R(\hat{\theta})$ où $\hat{\theta}$ est le déplacement angulaire, tel que:

$$I_D(R\mathrm{x}) = I_R(\mathrm{x}) \tag{1.25}$$

alors ces images peuvent être ré-échantillonnées en coordonnées polaires en prenant comme origine le centre des images (figure 1.15),

$$\bar{I}_R(\rho,\theta) = I_R(\rho\cos\theta, \rho sin\theta) \text{ et } \bar{I}_D(\rho,\theta) = I_D(\rho\cos\theta, \rho sin\theta) \tag{1.26}$$

où

$$\rho = \sqrt{(x-x_c)^2 + (y-y_c)^2} \quad \text{et} \quad \theta = tan^{-1}\left(\frac{y-y_c}{x-x_c}\right) \tag{1.27}$$

avec x_c et y_c les coordonnées du centre des images. De ces équations il peut être déduit que:

$$\bar{I}_D\left(\rho,\theta+\hat{\theta}\right) = \bar{I}_R(\rho,\theta) \tag{1.28}$$

où le symbole $\bar{}$ représente les images ré-échantillonnées.

Il est alors possible de calculer ce décalage angulaire en utilisant la corrélation basée sur la transformée de Fourier (équation 1.23) ou sur la corrélation des phases (équation 1.24).

Si entre les deux images, il existe aussi un facteur d'échelle $\hat{s}$, ce changement peut être représenté en utilisant des coordonnées logarithmiques sur l'axe ρ des coordonnées polaires:

$$I_D\left(e^{\hat{s}}R\mathrm{x}\right) = I_R(\mathrm{x}) \tag{1.29}$$

avec $\hat{s} = \log(\hat{\rho})$ le déplacement sur l'axe $\log(\rho)$ qui représente le changement d'échelle. En faisant le ré-échantillonage des images en coordonnées logarithmiques-polaires (figure 1.16), nous avons

$$\bar{I}_R(s,\theta) = I_R(e^s\cos\theta, e^s sin\theta) \text{ et } \bar{I}_D(s,\theta) = I_D(e^s\cos\theta, e^s sin\theta) \tag{1.30}$$

avec $s = \log(\rho)$, $\bar{I}_R(s,\theta), \bar{I}_D(s,\theta)$ les images ré-échantillonnées en coordonnées logarithmiques-polaires, telles que

$$\bar{I}_D\left(s+\hat{s},\theta+\hat{\theta}\right) = \bar{I}_R(s,\theta) \tag{1.31}$$

Il est maintenant possible d'appliquer la corrélation basée sur la transformée de Fourier pour trouver les déplacements angulaire et en échelle entre les images.

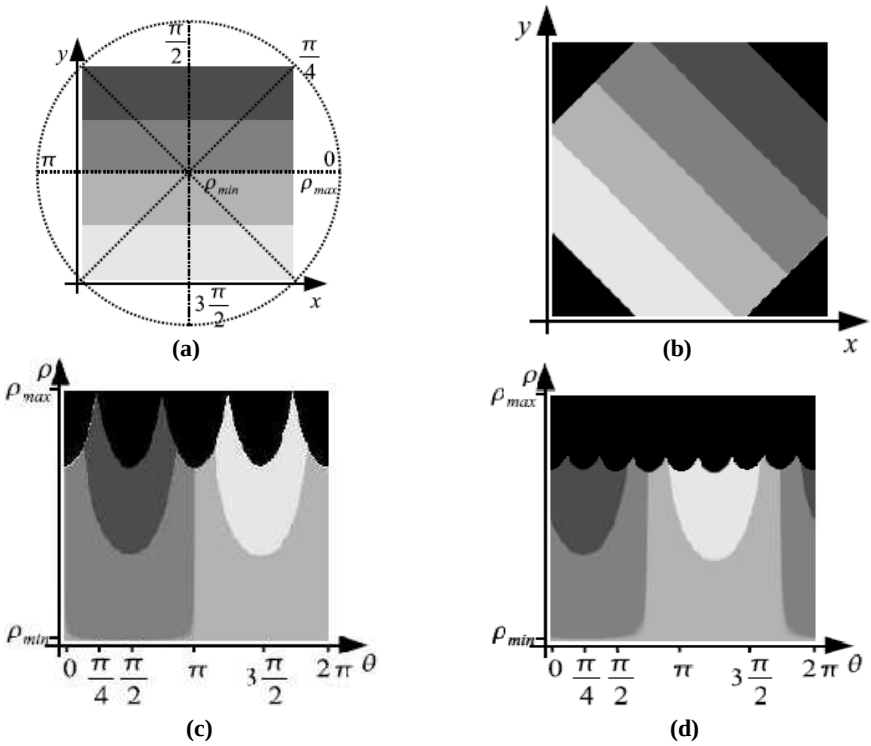


Figure 1.15. *Ré-échantillonage d'images en coordonnées polaires. (a) Image originale. (b) Image après une rotation de -45°. (c) Ré-échantillonage de l'image originale en coordonnées polaires. (d) Image avec rotation en coordonnées polaires. Le décalage sur l'axe θ entre les images (c) et (d) est visible. Ce déplacement peut être obtenu par corrélation croisée en utilisant la FFT.*

Pour le cas où une translation existe aussi entre les images (pour former un modèle de transformation de similitude),

$$I_D\left(e^{\hat{s}}R\mathrm{x}+\mathrm{t}\right) = I_R(\mathrm{x}) \tag{1.32}$$

il est aussi possible de récupérer cette translation mais il faut appliquer une procédure spécialement adapté à ce cas. Sur la base d'un travail de [CAS87], [RED96] ont développé un algorithme pour estimer les paramètres d'une similitude. D'abord, il faut obtenir la transformée de Fourier des images, pour ne retenir que les amplitudes des spectres. Cette amplitude est censée être invariante aux déplacements dans le domaine spatial (si les spectres sont cycliques). Lorsque l'échelle et la rotation sont connues, une image peut être obtenue avec l'inverse de ces paramètres appliqués à l'image à transformer pour ensuite estimer le déplacement par la corrélation basée sur la transformée de Fourier.

Pour que cet algorithme fonctionne, il est nécessaire que le déplacement entre les images soit petit (quelques pixels seulement) et que le contenu informatif dans les images soit important (figure 1.17).

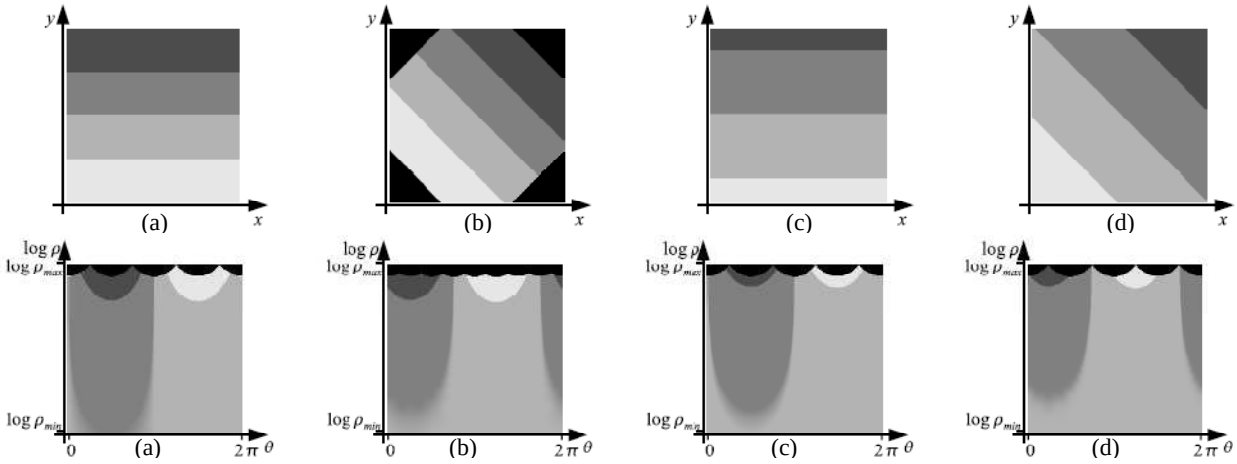


Figure 1.16. *Ré-échantillonage d'images en coordonnées log-polaires. Les zones en noir représentent les couples (ρ, θ) non contenus dans les images (a) et (b). (a) Image originale. (b) Image après une rotation de -45°. (c) Un grossissement de l'image (a) avec un facteur d'échelle de 1,5. (d) Image (a) après rotation de -45° et changement d'échelle de facteur 1,5. (e) Image originale rééchantillonnée en coordonnées log-polaires. (f) Image en coordonnées log-polaires de l'image (b). Il existe un déplacement sur l'axe θ. (g) Un déplacement sur l'axe log ρ (variation de l'amplitude) est visible lors du ré-échantillonage de l'image (c) en coordonnées log-polaires. (h) Déplacements sur l'axe θ et log(ρ) pour l'image avec rotation et changement d'échelle. Ces déplacements peuvent être obtenus par corrélation croisée en utilisant la FFT.*

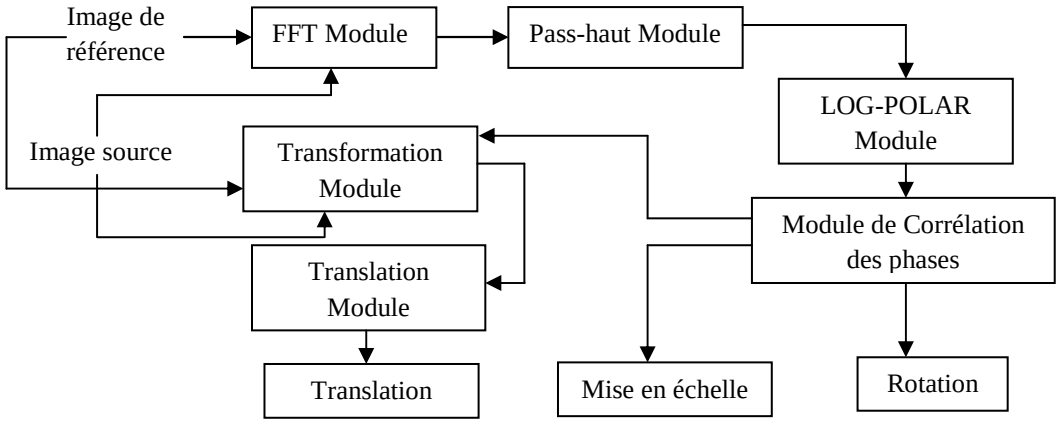


Figure 1.17. *Organigramme de la transformée rapide de Fourier* [LIU11]

f. Critère de Woods: Le recalage d'images multimodales (images issues de capteurs basés sur des principes physiques différents) est très souvent nécessaire, spécialement dans le domaine médical, mais compliqué du fait des différences de nature, de taille ou de résolution de ces images.

Par exemple, la comparaison entre des images anatomiques et fonctionnelles d'un patient permet d'améliorer et/ou de préciser le diagnostic. Dans ce cas, une approche basée sur une mesure statistique de la distribution des pixels dans l'image est préférée. Le critère de Woods

[WOO93] est une mesure de similarité initialement créée pour le recalage IRM-TEP. Ce critère suppose que les pixels ayant une intensité d dans l'image I_D correspondent tous à des pixels d'intensité r dans l'image I_R si les deux images sont recalées. De façon simplifiée, le critère de Woods indique que l'histogramme conjoint des images sera moins étalé lorsque les images seront recalées. Ce comportement est illustré dans la figure 1.18. Ce critère peut être calculé de la façon suivante:

$$CW(I_R|I_D) = \sum_R \frac{\sigma_{I_{D|R}}}{m_{I_{D|R}}} p_{RD} \tag{1.32}$$

Avec

$$m_{I_{D|R}} = \frac{1}{p_R} \sum_D \mathrm{D} p_{RD} \quad \text{et } \sigma_{I_{D|R}} = \sqrt{\frac{1}{p_R} \sum_D (D - m_{I_{D|R}})} \tag{1.33}$$

où R et D sont respectivement, les niveaux de gris des images I_R et I_D, $m_{I_{D|R}}$ et $\sigma_{I_{D|R}}$ sont respectivement, une moyenne et une variance conditionnelles, c'est-à-dire, la moyenne et la variance des niveaux de gris D de l'image I_D correspondants tous à un même niveau de gris R dans l'image I_R. p_R est la probabilité d'un niveau d'intensité R dans l'image I_R. p_{RD} est la probabilité jointe des niveaux d'intensité R et D. $CW(I_R|I_D)$ est minimal lorsque les images sont recalées.

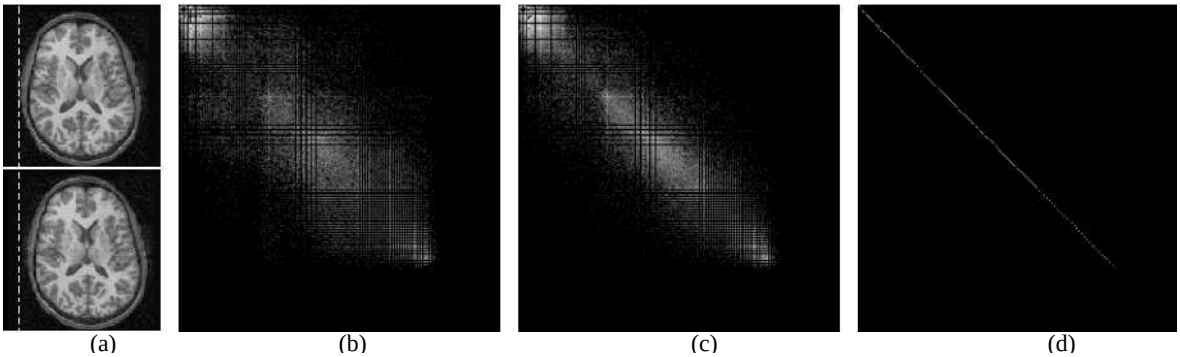


Figure 1.18. *Histogramme joint de deux images IRM. (a) Deux images IRM décalées l'une par rapport à l'autre (translation sur l'axe horizontal). (b) Histogramme joint pour un déplacement de 4 pixels. (c) Histogramme joint pour un déplacement de 2 pixels. (d) Histogramme lorsque les images sont recalées.*

Information mutuelle. L'information mutuelle prend son origine dans la théorie de l'information.

C'est une mesure de la dépendance statistique entre deux groupes de données. L'information mutuelle entre deux variables aléatoires I_R et I_D (les images à recaler) est donnée par

$$IM(I_R, I_D) = H(I_R) - \mathrm{H}(I_R|I_D) = H(I_R) + H(I_D) - H(I_R, I_D) \tag{1.34}$$

avec

$$H(I_R) = -\sum_R p_R \ln(p_R) \quad et \quad H(I_D) = -\sum_D p_D \ln(p_D) \tag{1.35}$$

où $H(I_R)$ et $H(I_D)$ sont les entropies des niveaux de gris des images I_R et I_D respectivement et $H(I_R|I_D)$ est l'entropie conditionnelle. Les entropies $H(I_R)$ et $H(I_D)$ prises de façon séparée,

mesurent la complexité des images, tandis que l'entropie jointe $H(I_R, I_D)$ mesure la quantité d'information que les images I_R et I_D apportent en même temps. p_R et p_D sont des fonctions de densité de probabilité des niveaux de gris R et D. p_{RD} est une fonction de probabilité jointe. Si les images sont proches, une image fournit la même information que l'autre et l'entropie jointe est minimale.

Pour des applications pour lesquelles l'information mutuelle est effectivement adaptée, l'information mutuelle est maximale si les images sont recalées. Dans ce cas, la maximisation de l'information mutuelle conduit à la superposition des images. L'un des premiers travaux concernant l'utilisation de l'information mutuelle est celui développé par [VIO97]. Ils décrivent l'utilisation de cette technique pour recaler des images IRM et pour recaler des objets 3D à une scène réelle. Les auteurs utilisent la fenêtre de Parzen pour calculer les fonctions de densité de probabilité. En utilisant une méthode similaire, construisent des images panoramiques à partir d'images endoscopiques.

Actuellement, le recalage basé sur l'information mutuelle est une technique très utilisée pour le recalage d'images multimodales. L'inconvénient de cette mesure est le coût en terme de calcul, car la détermination de l'information mutuelle est longue par rapport aux mesures de similarité présentées précédemment.

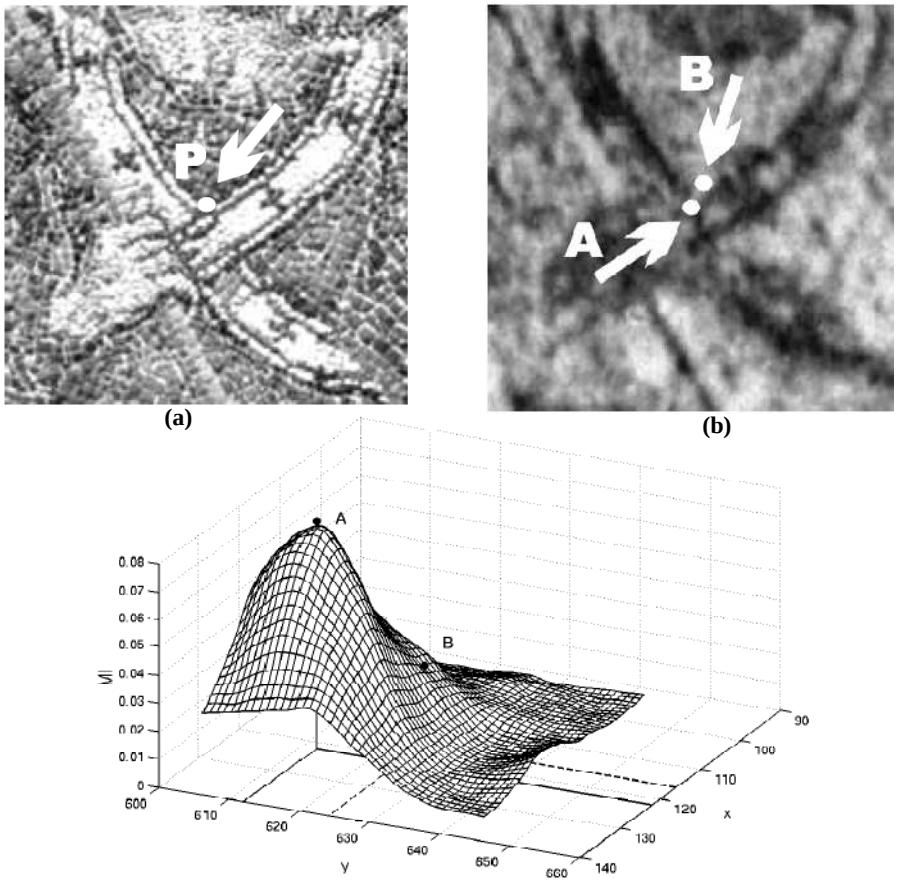


Figure 1.19. *Information mutuelle: Critère de MI (a) calculé au voisinage du point P entre la nouvelle et l'ancienne photographie de la mosaïque (b). Le maximum de MI montre la position appariée correcte (point A). Le point B indique la fausse position appariée sélectionnée précédemment par l'opérateur humain. L'erreur a été provoquée par la qualité et la nature complexe de l'image.*

1.6. Recalage basé sur des primitives images

La deuxième approche pour recaler les images est basée sur la détection de primitives images. Ces méthodes utilisent des techniques spécifiques de segmentation adaptées à l'application. Les algorithmes de segmentation sont représentés par g dans la définition mathématique du recalage d'images donnée par l'équation 1.4. Ces primitives images à extraire peuvent être des coins, des contours, des régions, etc. Il est possible d'extraire des primitives images des deux images à recaler, mais il est aussi possible d'en extraire uniquement dans une seule image pour les « comparer » directement aux niveaux d'intensité de l'autre image. Dans le cas plus classique où deux images sont segmentées, une distance entre les primitives est souvent calculée pour mesurer leur degré de superposition (distance euclidienne, distance de Hausdorff, carte de distances, etc).

Les paramètres de l'espace de recherche sont à nouveau donnés par la matrice de la transformation géométrique liant les images. Cette méthode de recalage a été utilisée depuis longtemps dans la vision stéréo et est de plus en plus utilisée dans le mosaïquage d'images [BRO03] [HEI05].

La première étape dans cette approche est l'extraction des primitives images.

1.6.1 Détection des primitives images

Le choix d'un type de primitives images à détecter dépend de plusieurs facteurs. Ainsi, les caractéristiques du capteur sont à prendre en compte (choix d'une primitive la moins sensible possible au bruit caractéristique du capteur). Le choix de la mesure de similarité est à faire en fonction du type et de la quantité de structures présentes dans l'image (lignes, cercles, etc).

Souvent, les méthodes de détection de primitives image cherchent sur toute l'image des « candidats » pour ensuite décider si l'information détectée est une primitive image du type souhaité ou pas. Les primitives résultantes sont des régions de l'image qui forment des points isolés, des courbes continues (point connectés) ou des régions connectées. La caractéristique la plus importante d'un détecteur de primitives images est la *répétabilité*, c'est-à-dire, sa capacité pour détecter la même primitive dans plusieurs images de la même scène. Un grand nombre d'algorithmes de détection a été développé. Ces détecteurs varient selon le type de primitive à détecter, la complexité de calcul et la répétabilité. Au final, la qualité du recalage dépend des performances du détecteur de primitives.
Nous présentons, ci-dessous quelques primitives images communément utilisées dans la littérature pour recaler des images.

***a. Contours*:** Les contours sont des régions d'une image correspondant à une forte variation d'intensité. En 1D, le profil d'un tel changement d'intensité aurait de façon idéale la forme d'un échelon, ou pour être plus proche de la réalité, d'une rampe plus ou moins étalée (figure 1.20). Une façon évidente de détecter les variations de niveaux de gris est de dériver le profil 1D. Dans les images, la dérivée doit être remplacée par un gradient qui peut être calculé à partir de deux dérivées (composantes orthogonales, typiquement selon les lignes et colonnes

des images). Deux informations importantes qui peuvent être calculées à partir des composantes $\frac{\partial I(\mathrm{x})}{\partial x}$ et $\frac{\partial I(\mathrm{x})}{\partial y}$ du gradient sont l'amplitude et l'orientation du gradient. Étant donnée une image $I(\mathrm{x})$, le module de son gradient s'exprime comme suit :

$$|\nabla I(\mathrm{x})| = \sqrt{\left(\frac{\partial I(\mathrm{x})}{\partial x}\right)^2 + \left(\frac{\partial I(\mathrm{x})}{\partial y}\right)^2} \tag{1.36}$$

et l'orientation est donnée par :

$$\angle \nabla I(\mathrm{x}) = \arctan\left(\frac{\frac{\partial I(\mathrm{x})}{\partial x}}{\frac{\partial I(\mathrm{x})}{\partial y}}\right) \tag{1.37}$$

Un maximum local de l'amplitude du gradient va représenter un point du contour dans l'image. Quand la dérivée première représente un maximum, la dérivée seconde doit avoir une valeur nulle. Ceci, est illustré dans la figure 1.20. C'est pour cette raison qu'une approche alternative dans la détection de contours est de trouver les zéros dans les images des dérivées secondes. L'opérateur différentiel utilisé pour effectuer cette opération est le *Laplacien*:

$$\nabla^2 I(\mathrm{x}) = \frac{\partial^2 I(\mathrm{x})}{\partial x^2} + \frac{\partial^2 I(\mathrm{x})}{\partial y^2} \tag{1.38}$$

En pratique, des approximations des dérivées des images sont obtenues par la convolution des images avec des filtres. Les filtres approximant des gradients sont aussi appelés opérateurs de gradient. Parmi les plus utilisés, nous pouvons citer l'opérateur de Roberts, de Prewitt, de Sobel, de Kirch et de Robinson.

Une amélioration de la détection des contours par la méthode du gradient a été proposée par Canny [CAN86]. Cette méthode vise à développer un filtre optimal pour détecter un échelon affecté par du bruit blanc. Dans l'article original, Canny propose une série de critères qu'un détecteur de contours optimal doit remplir. Le premier critère, est d'avoir après la détection, un rapport signal (gradient au niveau des contours) à bruit le plus élevé possible (distorsion contour/bruit). Le deuxième critère est que la distance entre les points du contour détecté et les points correspondants dans l'image (contour réel) doit être minimale. Le troisième critère est d'avoir une seule réponse pour chaque contour. Canny a démontré que la dérivée première est le filtre simple qui approxime au mieux le filtre optimal.

Le travail de [MÄK01] est un exemple de l'application du filtre de Canny à l'extraction des contours d'images IRM qui sont recalés avec des images TEP via une transformation non linéaire. La superposition de ces données doit faciliter le diagnostic de maladies coronaires. [GAY06] utilisent le détecteur de Canny pour extraire les contours d'une image 2D qui sera recalée avec un modèle 3D déformable en minimisant la différence d'intensité entre les contours et le modèle 3D.

Représenter une image par ses contours, a l'avantage de réduire considérablement la quantité de données tandis que la plupart des informations importantes est retenue. De cette façon il est possible de diminuer le temps requis pour recaler une paire d'images, mais il est aussi possible que la qualité du recalage en soit affectée.

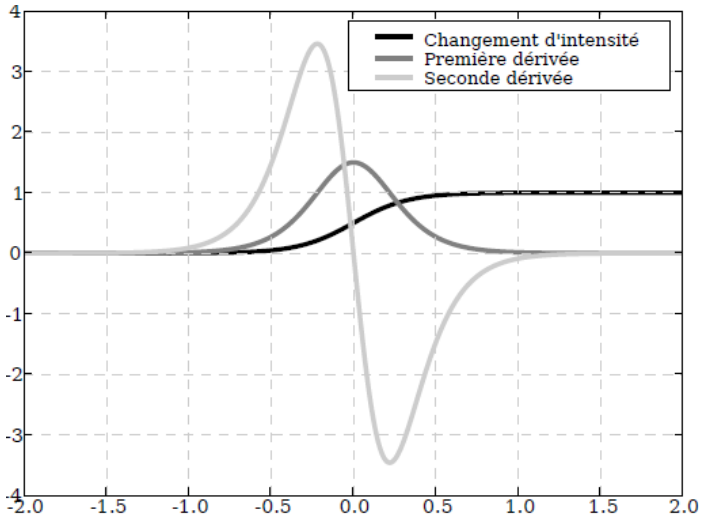


Figure 1.20. *Représentation du profil d'un changement d'intensité (en noir). En gris foncé, la première dérivée. En gris clair, la dérivée seconde.*

b. Coin: Un coin peut être vu comme des variations de niveaux de gris pour lesquelles il y a deux directions dominantes différentes dans un voisinage local centré sur le point d'intérêt. Le terme coin est un terme générique. D'ailleurs, un coin peut aussi bien faire partie d'une intersection de droites que de la rencontre de deux segments de droite ou bien encore correspondre à l'intensité maximum dans une courbe.

Une des méthodes les plus utilisées pour la détection des coins est le détecteur de Harris [HAR88]. C'est une méthode basée sur le travail de Moravec [MOR79]. Le détecteur de Harris utilise l'auto-corrélation locale d'une image. Cette auto- corrélation locale, mesure les changements locaux des niveaux de gris au travers d'une petite fenêtre qui est déplacée pour parcourir toute l'image. Soit $I(\mathrm{x})$ le pixel superposé par le centre de la fenêtre. La fonction d'auto-corrélation est calculée avec les niveaux de gris de l'ensemble des pixels de la fenêtre W. Le vecteur $\mathrm{u} = (\Delta x,\Delta y)^T$ permet de couvrir l'ensemble de la fenêtre W et représente des déplacements relatifs à $I(\mathrm{x})$ avec $\mathrm{x} = (x, y)^T$. Une valeur de corrélation est calculée pour tout $I(\mathrm{x})$ comme suit :

$$C(\mathrm{x}) = \sum_{\mathrm{W}} [I(\mathrm{x} - u) - I(\mathrm{x})]^2 \qquad (1.39)$$

Les niveaux de gris de W sont en général pondérés par une Gaussienne centrée sur $I(\mathrm{x})$. En faisant un développement limité du premier ordre en séries de Taylor de l'expression sur u nous obtenons : $\mathrm{u}^T\mathrm{M_H u}$, avec $\mathrm{M_H}$ une matrice des dérivées secondes de la région analysée.

$$M_H = \begin{bmatrix} \sum_w \frac{\partial^2 I(\mathrm{x})}{\partial x^2} & \sum_w \frac{\partial I(\mathrm{x})}{\partial x}\frac{\partial I(\mathrm{x})}{\partial y} \\ \sum_w \frac{\partial I(\mathrm{x})}{\partial x}\frac{\partial I(\mathrm{x})}{\partial y} & \sum_w \frac{\partial^2 I(\mathrm{x})}{\partial y^2} \end{bmatrix} \quad (1.40)$$

Ce détecteur utilise les valeurs propres λ_0 et λ_1 de la matrice M_H pour vérifier si W recouvre un coin. Si $\lambda_0 \approx 0$ et $\lambda_1 \approx 0$, il n'y a pas d'intersection sur la coordonnée x. Si $\lambda_0 \approx 0$ et $\lambda_1 > 0$, cela veut dire qu'un contour a été trouvé. Si $\lambda_0 > 0$ et $\lambda_1 > 0$ et les deux valeurs sont différentes, une intersection a été trouvée. Une approche plus simple, qui évite de calculer les valeurs propres, pour indiquer quels points seraient potentiellement des coins a été proposée aussi par [HAR88]. Les auteurs proposent d'utiliser M_H comme suit:

$$R = \det(M_H) - \alpha\, trace(M_H)^2 = \lambda_0\lambda_1 - \alpha(\lambda_0 + \lambda_1)^2 \quad (1.41)$$

où α es une valeur déterminée de façon empirique. Si $R >> 0$, il existe un coin; si $R << 0$, il existe un bord ; si $|R| < 0$ c'est une région où il n'y a pas de changements d'intensité. [HAR88] proposent α = 0.06 pour une application du suivi d'objets. [SHI94] proposent $\min(\lambda_0 , \lambda_1)$ pour améliorer la détection des coins dans le cadre du suivi du mouvement à partir d'une séquence d'images.

Les points détectés par l'algorithme de Harris ne sont pas invariants aux changements d'échelle. [LIN98] modifie le détecteur de Harris pour le rendre invariant aux changements d'échelle. D'autres algorithmes développés pour détecter des coins sont: SUSAN (Smallest Univalue Segment Assymilating Nucleus) [SMI97] développé dans le cadre d'une application au suivi d'objets, l'algorithme FAST (Features from Accelerated Segment Test) [ROS05] développé pour la modélisation 3D en temps réel et le détecteur de [WAN95] utilisé pour l'extraction de mouvement à partir d'une séquence d'images. Un résumé, ainsi qu'une comparaison expérimentale des détecteurs d'intersections ont été effectués par [SCH00].

c. Régions: Les régions sont des primitives images très utilisées dans le recalage d'images médicales. Elles sont obtenues par des algorithmes de segmentation dont le but est de détecter des structures anatomiques particulières. Les snakes (contours actifs) sont un exemple de modèle déformable très utilisé. Un snake est un contour qui est attiré par la primitive à détecter. Par exemple, s'il faut délimiter le pourtour d'une région, le gradient de l'image peut être vu comme une énergie qui attire le contour (énergie externe au contour). Une énergie interne au contour (liée à l'allongement et à la déformation du contour) est définie selon l'application pour introduire des connaissances a priori de la forme sur la région. La minimisation de la somme des énergies (internes et externes) fait converger le snake vers les bords de la région. Par exemple, [DAV96] utilise un contour déformable pour segmenter des images IRM du cerveau qui seront ensuite recalées entre elles pour comparer les différences entre un individu sain et un malade. Une autre application des modèles déformables est celle de [THO96], qui utilise ces modèles pour segmenter des surfaces dans des images IRM qui sont recalées entre elles. Ce recalage est utilisé dans le cadre de l'étude de la maladie d'Alzheimer.

Une autre approche de segmentation en régions totalement différente de la précédente repose sur les descripteurs invariants. Les points segmentés par les détecteurs de contours et les détecteurs de coins ne représentent qu'une partie très petite de l'information par rapport à la taille de l'image. Cette information peut être complétée en prenant en compte la région autour des points détectés par ces détecteurs. Les approches basées sur les descripteurs invariants utilisent en général un détecteur de coins pour trouver des points de départ pour ensuite effectuer des statistiques sur les niveaux de gris des régions englobant les coins détectés. Ces statistiques conduisent à des descripteurs de régions qui sont invariants aux changements d'échelle et aux transformations affines. Cette invariance peut s'avérer très utile lors du recalage d'images caractérisées par des différences en termes d'échelle et même de perspective.

Une façon simple d'obtenir une invariance aux changements d'échelle est de chercher des maxima des niveaux de gris (comparer) à des échelles différentes. Ces différences d'échelles peuvent être représentées dans une pyramide de Laplace où chaque niveau est une image correspondant à la différence de deux images filtrées par des gaussiennes (DoG) [LOW04]. Un exemple de descripteur invariant aux transformations affines (et même à quelques transformations en perspective) est le détecteur SIFT (Scale Invariant Feature Transform) proposé par [LOW04].

D'autres algorithmes développés récemment et qui conduisent à une invariance aux transformations affines sont GLOH (Gradient Location and Orientation Histogram) [MIk04] et SURF (Speeded Up Robust Features) [BAY06].

***d. Approches à base d'ondelettes*:** Les ondelettes ont aussi été appliquées à la détection de points d'intérêt. Dans [CHA91], l'image d'entrée est décomposée par les ondelettes B-splines suivant plusieurs échelles. La somme des composantes fréquentielles des sous-bandes bas-haut, haut-bas, et haut-haut est seuillée pour obtenir la carte contours. Un point d'intérêt est alors détecté si la composante haut-haut est plus grande que le seuil et appartient à la carte contours. Cependant, Fransson [FRA92] prétend que les sous-bandes haut-haut sont pleines de bruit en pratique. Il utilise donc les sous-bandes bas-haut et haut-bas pour détecter les coins. Chen et al. [CHE95] ont proposé un nouveau détecteur de contours et de coins basé sur le ratio des modules de la transformée en ondelettes inter-échelle (figure 1.21). Ce détecteur de contours peut correctement détecter les contours aux positions des coins, permettant ainsi la détection précise des coins. Pour réduire le nombre de points à traiter, ils appliquent la méthode de suppression des nonminima sur l'image contours et extraient ainsi l'image minima. Se basant sur la variance d'orientation, les points contours qui ne correspondent pas à des coins sont éliminés. Afin de localiser les points d'intérêt, ils ont proposé un nouvel indicateur de coins fondé sur la propriété de l'invariance par échelle des orientations des coins. Dans [MAN96], les coins d'objets ont été détectés en appliquant un modèle d'interaction d'échelle (scale-interaction model), où deux filtres directionnels passe-bande avec deux bandes passantes différentes (c'est-à-dire à deux échelles différentes) ont été appliqués à l'image originale.

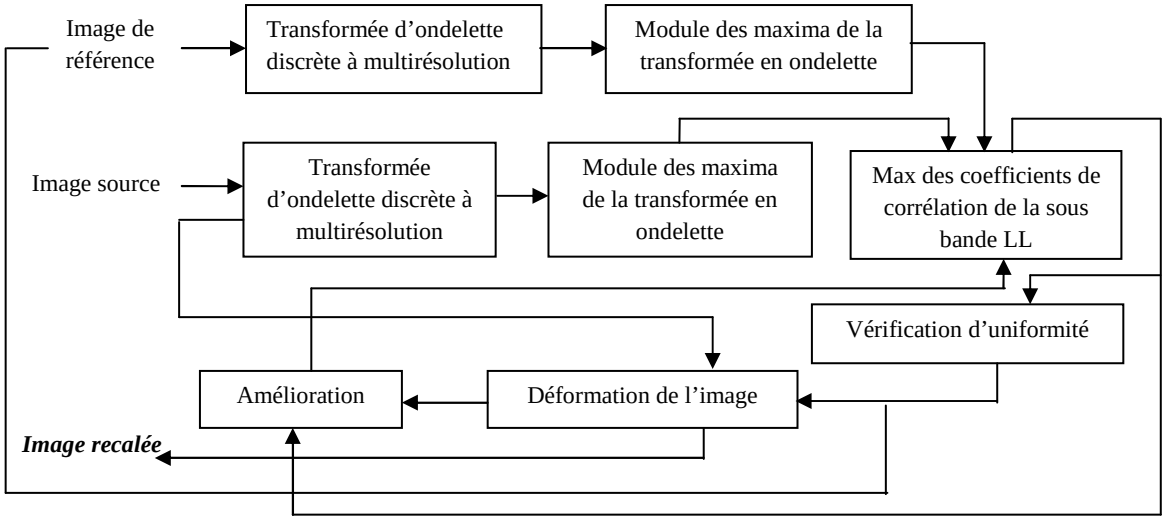


Figure 1.21. *Organigramme de recalage basé sur les ondelettes.*

La différence des deux sorties filtrées est ensuite calculée selon plusieurs orientations spécifiques, suivie d'une transformation non linéaire. La formulation de ce modèle de détection de points d'intérêt est basée sur l'utilisation des ondelettes de Gabor. Dans [QUD98], il a été proposé de remplacer les filtres passe-bande par des filtres passe-bas, et l'orientation désirée par la moyenne de toutes les orientations. De bons résultats ont été obtenus dans les deux cas, et des interprétations ont été données.

Récemment, Peter Kovesi [KOV03], a présenté un nouveau détecteur de coins et de contours fondé sur l'étude de la congruence de phase. Cette fonction a été largement décrite dans la littérature comme étant un outil efficace pour la recherche de points dont les composantes de phase dérivant d'une analyse échelle-fréquence sont alignées, c'est-à-dire les points où les composantes fréquentielles sont maximalement en phase. Le détecteur de Kovesi utilise les moments principaux de la fonction de congruence de phase pour déterminer les coins et les contours. Ces moments sont donnés par les équations suivantes :

$$\begin{aligned} a &= \sum (PC(\theta)\cos(\theta))^2 \\ b &= \sum (PC(\theta)\cos(\theta)).\left(PC(\theta)sin(\theta)\right) \\ c &= \sum (PC(\theta)\sin(\theta))^2 \end{aligned} \qquad (1.42)$$

où $PC(\theta)$ est la valeur de la congruence de phase déterminée dans la direction θ, et la somme est évaluée sur l'ensemble discret des orientations ou directions utilisées. Les moments maximum est minimum, M et m respectivement, sont donnés par :

$$M = \frac{1}{2}\left(c + a + \sqrt{b^2 + (a-c)^2}\right) \quad \text{et} \quad m = \frac{1}{2}\left(c + a - \sqrt{b^2 + (a-c)^2}\right) \qquad (1.43)$$

Cette mesure est basée sur les ondelettes 2D de Gabor orientées. L'opérateur de détection de coins et de contours résultant est bien localisé et sa réponse est invariante aux variations de

l'amplitude du contraste dans une image. Une autre propriété intéressante de cet opérateur réside dans le fait que la carte de coins est strictement un sous-ensemble de la carte de contours. Ceci facilite l'utilisation coopérative de l'information de coins et de contours. Cependant, cette méthode nécessite une estimation du bruit à petite échelle. De plus, la congruence de phase est très coûteuse en temps de calcul.

Plus récemment, deux algorithmes multiéchelles de détection de coins dans les images en niveaux de gris basés sur la transformée en ondelettes Log-Gabor ont été proposés [GAO07]. L'image d'entrée est décomposée par les ondelettes Log-Gabor à plusieurs échelles et selon plusieurs orientations. Dans le premier algorithme, la magnitude le long de la direction orthogonale à l'orientation du gradient est prise comme une mesure de point d'intérêt. Dans la deuxième méthode, les composantes à différentes échelles et orientations sont projetées sur les axes x et y et formulées dans la matrice des moments de second ordre. Enfin, la valeur propre la plus petite de cette matrice est utilisée pour détecter les coins. Comparés aux détecteurs de Harris et de Kovesi, il a été démontré que les algorithmes proposés sont plus performants.

1.6.2. Recalage des primitives images

Comme pour les méthodes basées sur les niveaux d'intensité des images, il faut définir une mesure de similarité qui représente un degré de ressemblance de deux primitives images. Mesurer une distance entre primitives images permet par exemple d'estimer le degré de superposition (donc de ressemblance) de ces primitives. La norme euclidienne est classiquement utilisée pour mesurer la distance entre deux points. Très peu gourmande en ressources, elle est très utilisée pour des applications à la vision par ordinateur [BRO92].

Pour le recalage de contours et des surfaces, il existe d'autres mesures de distance. Une approche classique repose sur l'utilisation de cartes de distances. Dans une carte de distances, les niveaux de gris des pixels superposés par le contour valent zéro. Les niveaux de gris des autres pixels sont directement liés à la plus courte distance du pixel au point de contour le plus proche.

Ces distances peuvent être calculées rapidement via du filtrage récursif. Ces cartes permettent de calculer de façon très efficace la distance entre deux contours, la carte étant calculée avec un contour alors que le deuxième contour est superposé à la carte. La moyenne des niveaux de gris superposés par ce contour donne une distance moyenne entre les deux contours. [FIT01] propose l'utilisation d'une carte de distances codant des distances euclidiennes pour recaler des nuages de points 2D et 3D.

Une autre mesure très utilisée pour mesurer la distance entre contours ou surfaces est la distance de Hausdorff. Pour deux ensembles de points A = $\{a_1, \ldots a_n\}$ et B = $\{b_1, \ldots b_n\}$ représentant 2 surfaces ou contours, la distance de Hausdorff est définie par :

$$D_H(A,B) = \max\big(d_H(A,B), d_H(B,A)\big) \tag{1.44}$$

avec

$$d_H(A,B) = \max_{a\epsilon A} \min_{b\epsilon B} \|a-b\| \quad (1.45)$$

où $d_H(A, B)$ est une distance de Hausdorff *dirigée*. [CHM03] utilisent cette distance pour recaler des images portale et de simulation en imagerie à rayons X. Récemment [POS07] proposent une méthode basée sur la minimisation de la distance de Hausdorff pour le recalage d'images 3D (point à la surface du visage) et de données CT, dans le cadre du positionnement du patient lors de sessions de radiothérapie intracrânienne (point sur la tête du patient).

1.7. Conclusion

Dans le cas de séquences vidéo, le mouvement local prédominant des primitives images est souvent la translation (déplacement entre deux images consécutives) [SHI94]. Si les primitives choisies sont des coins, il est possible d'utiliser une mesure de similarité comme celles présentées dans la section 1.5.1. En effet, la *différence d'intensité* ou la *corrélation croisée normalisée* peuvent par exemple être utilisées de façon directe pour comparer les intensités des pixels situés dans des petites régions au voisinage des primitives images. Si les points à recaler se trouvent dans des images non consécutives d'une séquence, leur apparence peut varier fortement. Dans ce cas, il pourra s'avérer nécessaire d'utiliser un modèle de transformation plus complexe, comme un modèle de transformation affine pour lequel les descripteurs invariants deviennent indispensables.

Comme pour le recalage utilisant tous les niveaux d'intensité des images, il faut, après avoir établi une mesure de similarité reliant les primitives images, choisir la transformation géométrique adaptée à l'application. Ces transformations seront optimisées pour réduire la distance entre les primitives images. À la fin du processus de recalage, la méthode d'optimisation va délivrer la transformation globale existant entre les images.

Si beaucoup de primitives sont détectées, la manière la plus simple de trouver tous les points correspondants entre deux images est de comparer toutes les primitives trouvées dans une image avec toutes les primitives de l'autre image. Cette méthode requiert des temps de calcul trop importants. Le temps de calcul peut être réduit en utilisant des approches indexées comme les kd trees, utilisés dans [BRO03]. Une fois l'ensemble des correspondances potentielles calculées, il faut trier celles-ci pour garder uniquement celles qui vont produire le recalage le plus précis. Dans cette optique, l'approche la plus utilisée (surtout dans le recalage d'images monomodalité 2D–2D [ZIT03]) est appelée RANSAC (RANdom SAmple Consensus).

Le problème le plus important dans l'implémentation de ces algorithmes est la détection précise des primitives images et la mise en correspondance des points communs dans les images à recaler.

CHAPITRE 2

MISE EN CORRESPONDANCE DE POINTS D'INTERET SOUS CONTRAINTES PHOTOMETRIQUES

La problématique de cette partie est liée à la recherche de similarités entre images. Pour se faire, il n'est pas efficace de comparer directement les valeurs des pixels entre des images. Cette méthode ne serait par exemple pas robuste à des déformations comme celles liées à des changements d'illumination, à l'échelle, aux rotations, aux changements de points de vue. Il est toutefois possible d'essayer de trouver des corrélations entre des fenêtres de pixels. Les méthodes récentes, plus efficaces, consistent à décrire chaque image avec un ou plusieurs descripteurs. L'objectif est alors de construire un descripteur à partir des valeurs pixeliques, qui sera invariant selon les critères souhaités (rotation, échelle...). Ces descripteurs sont alors des points dans un espace de grande dimension. Pour trouver des similarités entre plusieurs images, on se limite à trouver des descripteurs proches selon une distance choisie.

Notre première contribution dans cette thèse consiste à modifier la partie de sélection des plus proches descripteurs utilisée dans l'algorithme SIFT classique. Pour cela, Nous introduisons un algorithme basé sur les ondelettes de Haar afin de diminuer le nombre de faux appariements des différents descripteurs locaux SIFT ainsi que le temps de recherche.

2.1. Le schéma habituel de la mise en correspondance

Afin de déterminer l'orientation relative de deux caméras à partir d'appariements de points entre ces vues, une partie importante de la littérature en vision par ordinateur repose sur la mise en correspondance de points d'intérêt entre plusieurs vues.

La *Mise en Correspondance* vise à identifier les points d'intérêt détectés parmi plusieurs images (figure 2.1) qui sont les projetés du même point en 3 dimensions. Cette action est souvent réalisée en prenant en compte des descripteurs locaux, c'est-à-dire un encodage des valeurs en niveaux de gris du voisinage d'un point d'intérêt.

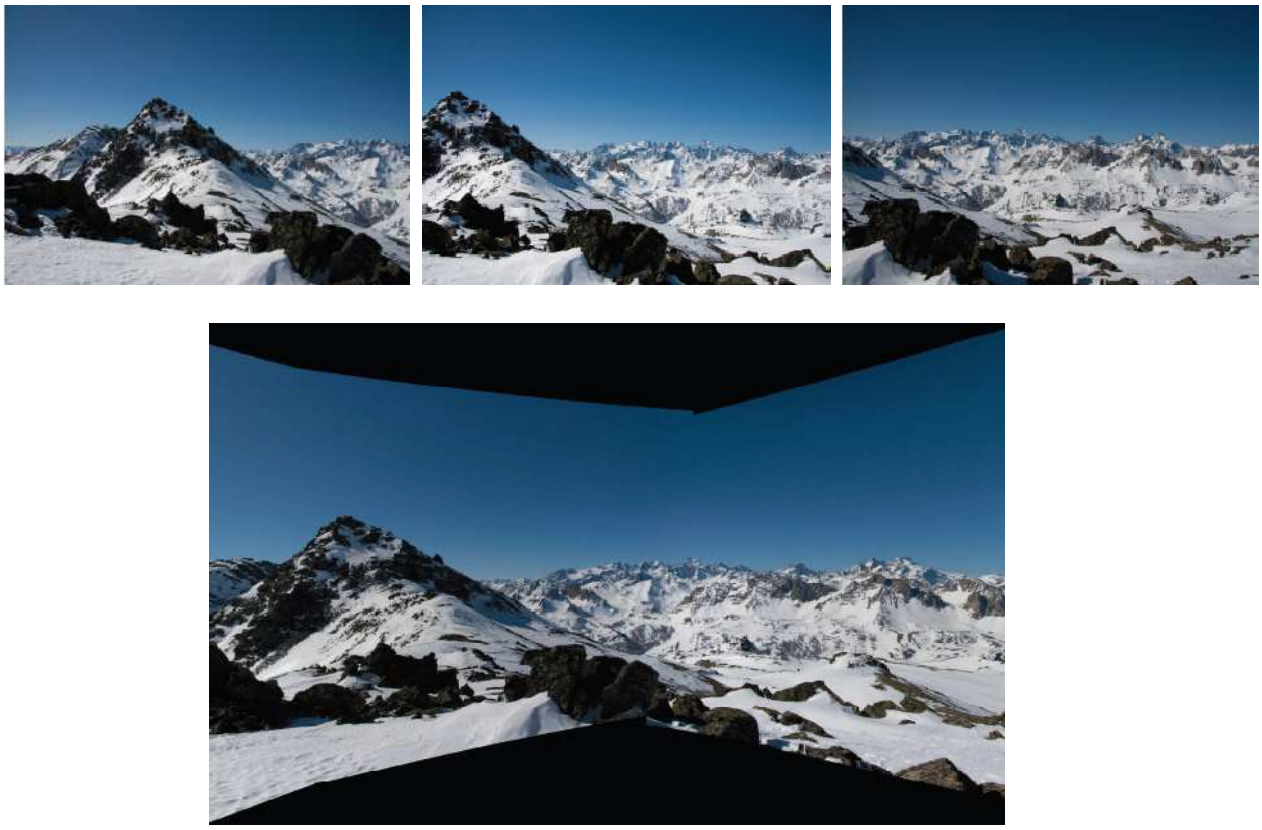


Figure 2.1. *Panorama réalisé à l'aide de mise en correspondance de points d'intérêt.*

Nous allons maintenant décrire le schéma de mise en correspondance entre deux images distinctes d'une même scène 3D statique prise par une caméra en mouvement. Une méthode est la suivante:

1. Dans chaque vue, extraire des points d'intérêt ainsi que leurs descripteurs associés représentant la photométrie au voisinage, dans une fenêtre de taille limitée centrée sur le point.
2. Les mettre en correspondance à l'aide d'une mesure de (dis-)similarité entre les descripteurs.

   Une première façon de procéder est de comparer des fenêtres carrées centrées sur chaque point par corrélation.

   Définition: ***Corrélation*** *entre deux fenêtres d'images A et B de même taille de n pixels de côté. La mesure s'écrit :*

$$\sum_{i,j=1}^{n} A_{i,j} B_{i,j} \tag{2.1}$$

   Cependant, une telle mesure ne prend pas en compte les déformations liées aux changements de points de vue. De plus, la corrélation ainsi définie n'est pas invariante

par changement de contraste entre les images. Il existe des versions plus élaborées centrées et normalisées, avec des performances légèrement supérieures, mais elles ne suffisent pas à décrire le voisinage d'un point de façon invariante à tout type de transformation. La version centrée et normalisée est donnée par l'équation 2.2.

$$Corr(A,B) = \frac{\sum_{i,j=1}^{n}\left(A_{i,j} - \bar{A}\right)\left(B_{i,j} - \bar{B}\right)}{\sqrt{\sum_{i,j=1}^{n}\left(A_{i,j} - \bar{A}\right)^2 \sum_{i,j=1}^{n}\left(B_{i,j} - \bar{B}\right)^2}} \qquad (2.2)$$

Avec $\bar{A}$, $\bar{B}$ sont les intensités moyennes de A et B.

Par exemple, sans information a priori, il est impossible de savoir si les deux points sont observés à la même *échelle*, nécessaire pour définir la taille du voisinage que l'on considère. Ces observations mettent en évidence la nécessité de disposer de représentations invariantes du voisinage d'un point.

3. Filtrer les appariements en sélectionnant les plus cohérents avec une transformation géométrique imposée par un mouvement de caméra plausible.

Combiner toutes ces étapes est un travail difficile. Ceci nécessite en effet de fixer de nombreux paramètres, et un mauvais choix pour l'un d'eux risque de corrompre le résultat du procédé dans son ensemble.

Nous allons maintenant détailler les points cruciaux des étapes de mise en correspondance dans les prochaines sections, afin de mettre en évidence les points qui ont retenu notre attention dans cette thèse.

2.2. Points d'intérêt et descripteurs locaux

Dans cette section, nous présentons l'extraction de points d'intérêt, la première étape du schéma de mosaïquage: d'abord, la partie détection, afin d'exhiber des points faciles à localiser d'une image à l'autre; ensuite la partie descripteur, vecteur de caractéristiques invariantes utilisées pour la mise en correspondance.

Les points d'intérêt sont des endroits saillants, faciles à repérer. Depuis Moravec [MOR79] et ses premiers travaux sur la corrélation de zones d'intérêt, la littérature traitant la détection de primitives bas-niveau s'est enrichie de nombreuses méthodes reposant sur différents outils: détection de contour via des filtres, calculs de gradient, de courbure.

2.2.1. Détecteur de Harris

Le détecteur de Harris a été le point de départ des techniques de détection des points invariants. Il a la propriété de détecter les coins sur les images et ce indépendamment de la direction dans laquelle a été prise l'image. Il observe la rapidité des transitions en intensité sur l'image, un coin étant caractérisé par une transition isotrope et rapide.

Mais cet opérateur est lié à l'échelle de l'image (si par exemple un arbre a été photographié à 50 mètres de distance ou à 200 mètres, les points détectés ne seront pas les même). Le détecteur de Harris ne peut donc pas être considéré comme utilisable dans la création de panoramas car les images d'une même scène peuvent être prises à des distances très variables.
Les détecteurs présentés par la suite reposent sur le même principe que l'opérateur de Harris, à savoir localiser sur l'image les coins indépendamment de l'orientation de l'image, mais sont eux invariants aux changements d'échelle.

2.2.1.1. Principe

L'idée est de trouver un opérateur mettant en évidence les coins dans les images. Pour se faire, Harris tente de détecter les changements bruts d'intensité sur l'image (ce qui correspond à un contour): d'un pixel (x, y) à un pixel (x+u, y+v) l'intensité passe de I(x, y) à I(x+u, y+v). On introduit alors une fonction calculant les changements d'intensité:

$$E(u,v) = \sum_{x,y} w(x,y) \times [I(x+u, y+v) - I(x,y)]^2 \qquad (2.3)$$

où $w(x, y)$ est la gaussienne à deux dimensions :

$$w(x,y) = \frac{1}{2\pi\sigma^2} e^{-(\frac{x^2+y^2}{2\sigma^2})^2} \qquad (2.4)$$

L'équation (2.3) peut être approchée par :

$$E(u,v) \cong [u,v] \times M \times \begin{bmatrix} u \\ v \end{bmatrix} \qquad (2.5)$$

où

$$M = \sum_{x,y} w(x,y) \times \begin{bmatrix} I(x,x) & I(x,y) \\ I(x,y) & I(y,y) \end{bmatrix} \qquad (2.6)$$

Cette matrice est symétrique et donc diagonalisable. Considérons λ_1, λ_2 ses valeurs propres:
L'équation (2.4) donne

$$E(u,v) \cong \lambda_1 u^2 + \lambda_2 v^2 \qquad (2.7)$$

Un point peut se caractériser par un changement d'intensité identique dans toutes les directions, on s'intéresse donc à $E(u, v) = constante$ (ce qui nous donne une ellipse: voir

figure 2.2). Les vecteurs propres sont caractéristiques de la nature de la transition: le vecteur portant λ_{max} donne la direction de changement rapide, λ_{min} celle de changement plus lent. On peut considérer qu'une transition est un point si λ_{max} et λ_{min} sont tous les deux très grands.

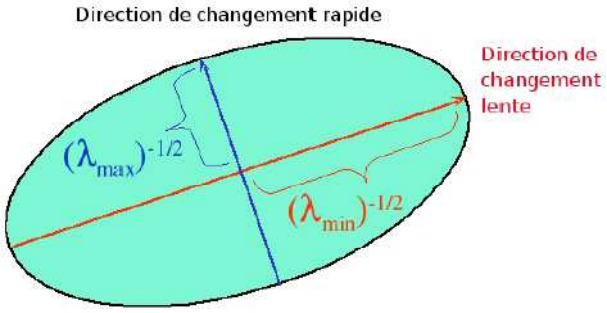


Figure 2.2. *Influence des valeurs propres sur la rapidité des transitions.*

Plutôt que d'utiliser à la fois ces deux variables, on préfère n'en utiliser qu'une seule :

$$R = det(M) - k(trace(M))^2 \qquad (2.8)$$

où

$$det(M) = \lambda_1\lambda_2 \qquad \text{et} \qquad trace(M) = \lambda_1+\lambda_2 \qquad (2.9)$$

Les valeurs de R nous renseignent sur la nature des transitions (voir figure 2.3). Ainsi si $R > 0$ la transition est possiblement un point. Pour avoir des résultats plus stables on peut fixer un seuil positif et n'accepter que les valeurs supérieures à ce seuil. Les points considérés comme des coins sont alors les maxima locaux de R (fonction qui dépend de x et de y).

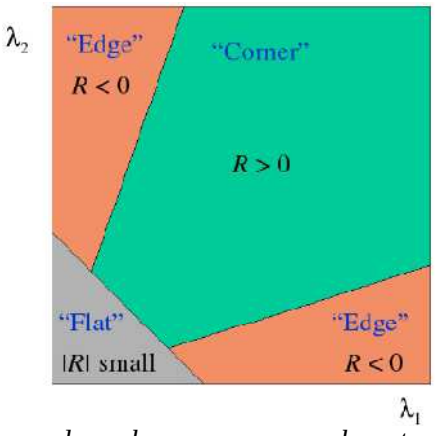


Figure 2.3. *Influence des valeurs propres sur la nature des transitions.*

2.2.1.2. Propriétés

Cette méthode est invariante par rotation: en effet, l'ellipse change de direction, mais pas la valeur de λ_{max} et λ_{min}.

De plus le détecteur de Harris est invariant aux variations affines (c'est à dire à un changement global et uniforme des intensités sur l'image) tant que les maxima ne sont pas en deçà du seuil. Dans le cas contraire les points sont ignorés, ce qui peut être relativement gênant. Mais le problème majeur est que le détecteur de Harris n'est pas indépendant de l'échelle de l'image. Ainsi si l'on représente la même scène mais à deux échelles différentes, il y aura une différence de points détectés.

2.2.2. Le détecteur Harris-Laplacien

2.2.2.1. Principe

Le but est de déterminer la localisation des points invariants et quelle que soit leur échelle (Voir figure 2.4). Il faut aussi faire en sorte que la méthode créée soit invariante par rotation.

L'idée utilisée dans le détecteur Harris-Laplacien est d'abord de détecter les points invariants sur l'image par l'opérateur de Harris, puis d'utiliser comme filtre le Laplacien $L(x, y, \sigma)$ pour chaque point :

$$L(x, y, \sigma) = \sigma^2 \nabla^2 G(x, y, \sigma) \tag{2.10}$$

Où σ^2 est la variance de G et ∇^2 est l'opérateur Laplacien. De plus,

$$\mathrm{G}(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \mathrm{e}^{-\frac{x^2+y^2}{2\sigma^2}} \tag{2.11}$$

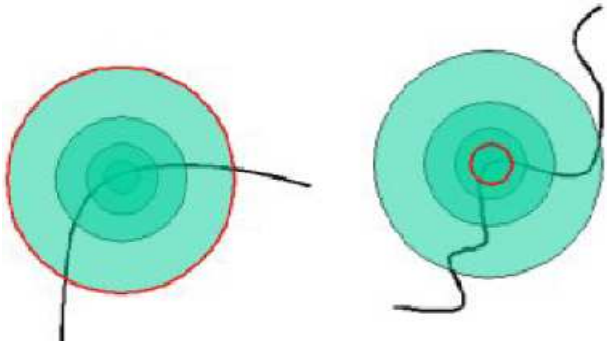


Figure 2.4. *Localisation de l'invariant (coin) et détermination de son échelle (cercles rouges) sur deux images ayant une scène commune à deux échelles différentes.*

Le Laplacien a pour effet de mettre en évidence les fortes transitions (contours et points). De plus, comme l'a montré Lindeberg [LIN98], σ^2 est un facteur qui garantit à l'opérateur $L(x, y, \sigma)$ l'invariance par rapport à l'échelle. Pour calculer l'échelle au point considéré, il suffit donc d'observer l'amplitude obtenue en fonction de σ que l'on fait varier, l'extremum local nous donnant la solution (voir figure 2.5).

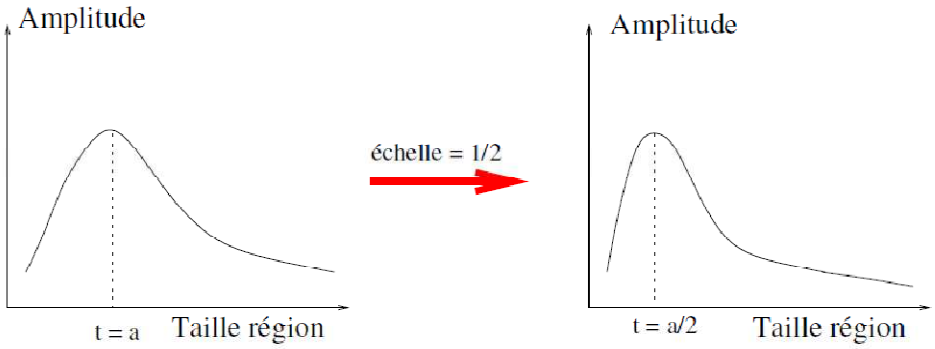


Figure 2.5. *Détermination de l'échelle de l'invariant (au maximum de l'amplitude). L'image filtrée à droite est la même que celle de gauche à une échelle* $\frac{1}{2}$ *près.*

Cette méthode utilise la notion dite d'*espace d'échelle* ou *scale space* (voir figure 2.6).

Notons d'abord que le Laplacien est aussi invariant par changement de rotation. Ceci nous donne donc une méthode invariante par rotation et par changement d'´echelle.

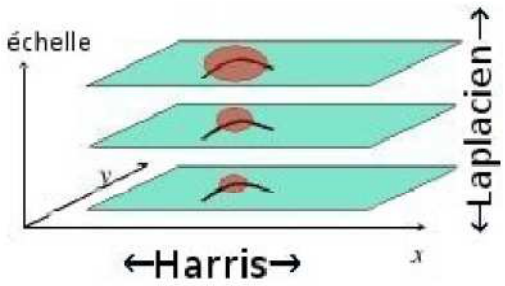


Figure 2.6. *L'espace d'´echelle dans le cas de l'opérateur Harris-Laplacien : en ordonnée et en abscisse l'opérateur de Harris (coordonnées du point), en z l'échelle du point donnée par le Laplacien.*

Tuytelaars et Mikolajcszyk [TUY08] définissent les points d'intérêt comme un motif dans l'image qui diffère de son voisinage immédiat, pour des propriétés comme l'intensité, la couleur et la texture. Une telle définition couvre les coins, les jonctions en T, en L, les motifs de texture, et certaines parties de contours. Leurs caractéristiques attendues varient suivant l'utilisation que l'on fait des points d'intérêt. Lorsqu'on vise un positionnement précis, il est nécessaire de distinguer les objets très similaires, voire identiques, mais positionnés différemment. Pour leurs différentes utilisations, on leur associe un descripteur regroupant des mesures prises sur une région locale centrée sur le point détecté.

Ces méthodes, détectant position et échelle de concert, si elles permettent une meilleure détection de points, n'aident que peu leur appariement, qui reste limité par les méthodes de corrélation. A partir des travaux fondateurs de Schmid et Mohr [SCH97], qui associent aux points d'intérêt des descripteurs photométriques locaux, invariants aux rotations et changements d'échelles, une grande variété de méthodes ont été proposées afin d'associer à chaque point d'intérêt un descripteur photométrique local.

2.2.3. Critères d'évaluation

Afin d'évaluer la performance de ces points, on définit la *répétabilité* d'un point détecté dans une scène comme la capacité à le repérer, confronté à différentes transformations 2D d'une image. Ces transformations sont par exemple: zoom, rotation, échantillonnage colorimétrique de l'image, flou...

La *répétabilité* est forte lorsque, pour deux conditions différentes d'observation d'une même scène, un grand pourcentage des points détectés dans la première image est aussi visible dans la seconde. Elle est calculée pour une précision δ fixant la distance maximale entre les coordonnées des points détectés dans la première image et projetés dans la seconde suivant la transformation expliquant leur déplacement, et celles mesurées dans la seconde, entre une image de référence et sa variante soumise à une transformation 2D quelconque. Les articles d'évaluation de Mikolajczyk proposent une analyse de la répétabilité des points d'intérêt [MIK04].

Nous distinguons la *répétabilité* à des transformations 2D de la capacité à mettre en correspondance des points dans le cas d'un changement de point de vue dû à un mouvement de la caméra comprenant une translation du centre optique (pas de rotation pure autour de l'axe optique): dans ce cas, seule la connaissance complète de la scène nous permettrait de le mesurer. Moreels et Perona [MOR07] présentent une évaluation des détecteurs prenant en compte cette limitation, via l'utilisation d'une table robotisée tournante et d'un système de validation à trois vues. A partir de la géométrie calculée entre chacune des paires de vues, qui donne une contrainte point / droite, il est possible d'en déduire une contrainte point / point via l'intersection de deux droites épipolaires dans la troisième vue, ce qui permet un calcul automatisé de la répétabilité pour toute forme d'objet. Cela leur permet d'établir que les combinaisons de détecteurs et descripteurs Hessian-affine et SIFT sont les plus robustes vis à vis des changements de points de vue, et que pour les changements d'éclairage et de focale des caméras les Harris-affine avec SIFT, et les Hessian-affine avec des « shape context » donnent les meilleurs résultats. Ils montrent aussi qu'aucune des combinaisons testées ne donne des bons résultats pour des changements de points de vue de plus de 25-30°.

Obtenir une répétabilité forte à une certaine classe de transformations ne nous assure cependant pas du caractère *distinctif* de l'information apportée par le point. Ce caractère distinctif, qui signifie empiriquement que le descripteur du point contient suffisamment d'information pour mettre en correspondance sans ambiguïté, varie en fonction du contexte. Lorsqu'une scène contient de nombreux motifs répétés, l'ambiguïté reste entière si l'on utilise uniquement l'information photométrique.

2.2.4. Caractéristiques souhaitables d'un détecteur

Nous indiquons les qualités recherchées pour un point d'intérêt utilisé dans le cadre de l'estimation de mouvement. On souhaite disposer de points d'intérêt munis de descripteurs, ayant les qualités suivantes :

- une précision de localisation élevée;
- une bonne répétabilité vis à vis des transformations de l'image;
- un caractère local pour une robustesse aux occultations;
- une description invariante:
  - à certains changements de points de vue (similitudes, transformations affines, ou homographies);
  - aux différences de contraste (à défaut de la prise en compte des variations d'éclairage).

Un des algorithmes les plus populaires, qui satisfait raisonnablement les caractéristiques ci-dessus, est probablement SIFT de David Lowe [LOW04]. Nous utilisons dans nos expériences cette approche, et allons donc le présenter plus en détails. La figure 2.7 présente un exemple de détections.

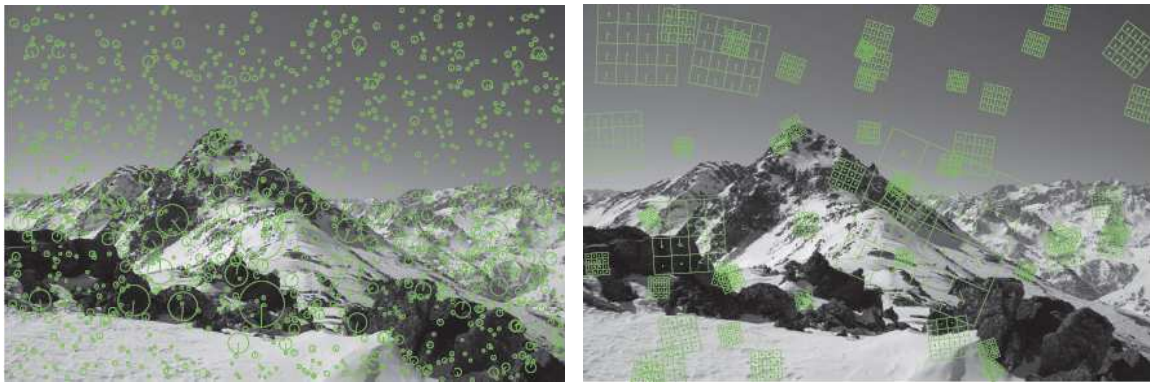


Figure 2.7. *– A gauche – Points d'intérêt détectés. Les rayons des cercles sont proportionnels aux échelles calculées, et l'orientation trouvée est matérialisée par le rayon. A droite – Descripteurs pour une sélection de points. Les grilles affichées contiennent les statistiques d'orientation du gradient dans un voisinage du point dont la taille est proportionnelle à l'échelle détectée.*

2.2.5. Détecteur Sift

Le détecteur SIFT, proposé par [LOW04] extrait des points remarquables d'une image. L'extraction par ce procédé est stable dans le sens où une autre image raisonnablement éloignée d'une même scène permettra d'obtenir des points correspondants pour la plupart aux

mêmes coordonnées 3D. De tels points sont invariants aux modifications d'échelle et aux rotations, et partiellement invariants aux changements de luminosité et de point de vue 3D.

Grandes étapes: Les étapes principales de ce procédé sont les suivantes :

1. **Détection d'extrema dans l'espace-échelle** gaussien, via le calcul des extrema des différences de gaussienne convoluées avec l'image. L'espace-échelle (voir la figure 2.8) est défini comme la fonction $L(x, y, \sigma)$, produit de convolution entre la gaussienne $G(x, y, \sigma)$ et l'image $I(x, y)$:

$$L(x, y, \sigma) = G(x, y, \sigma) \otimes I(x, y) \tag{2.12}$$

avec $\otimes$ le produit de convolution et

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \tag{2.13}$$

La fonction différence de gaussienne convoluée avec l'image s'écrit alors :

$$DoG(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{2.14}$$

Le point d'intérêt est alors défini comme extrema des différences de gaussiennes.

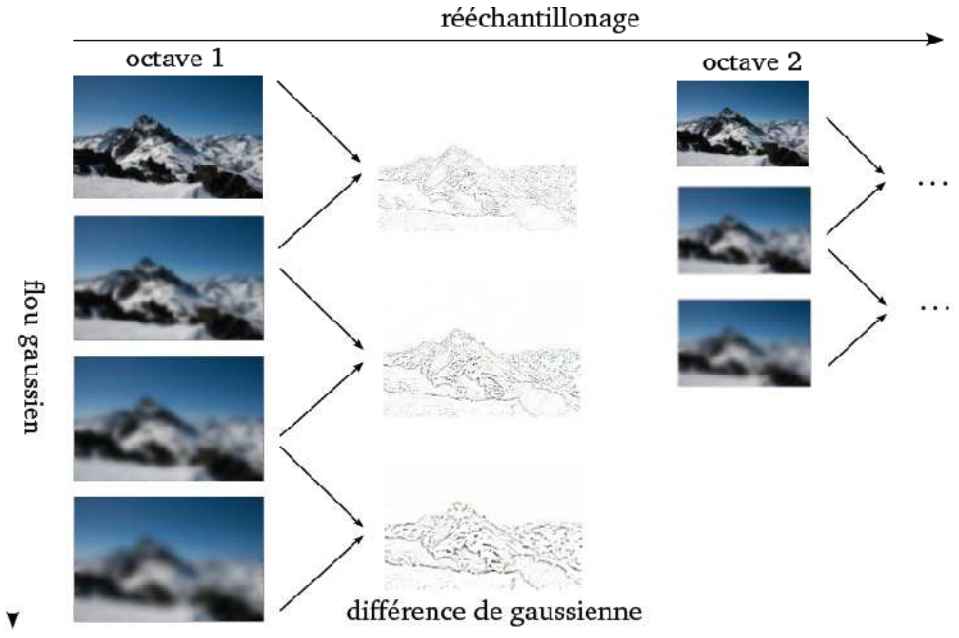


Figure 2.8. *Espace échelle gaussien. On recherche les extrema des différences de gaussienne afin de déterminer la position des points d'intérêt. Une octave correspond au doublement de l'échelle du flou gaussien σ.*

2. **Localisation de points clés :** pour chaque candidat, une optimisation sous-pixelique des coordonnées *x*, *y* et de l'échelle *s* est réalisée : si la position obtenue est stable (les coordonnées trouvées après l'optimisation décrivent un point suffisamment proche du point détecté initialement), il est déclaré point clé. Il faut cependant rester prudent quant aux points détectés: certains peuvent soit provenir d'un bruit important (créant de rapides transitions mais non désirables dans notre cas) soit des contours mis en évidence (on ne s'intéresse pas aux contours mais aux coins). Ces points gênants peuvent être éliminés par un seuillage et les contours par le calcul des valeurs propres de la matrice

$$H = \begin{bmatrix} DoG_{xx} & DoG_{xy} \\ DoG_{xy} & DoG_{yy} \end{bmatrix} \tag{2.15}$$

La figure 2.9 obtenue par cette méthode donne un exemple du nombre de points invariants qui peuvent être obtenus.

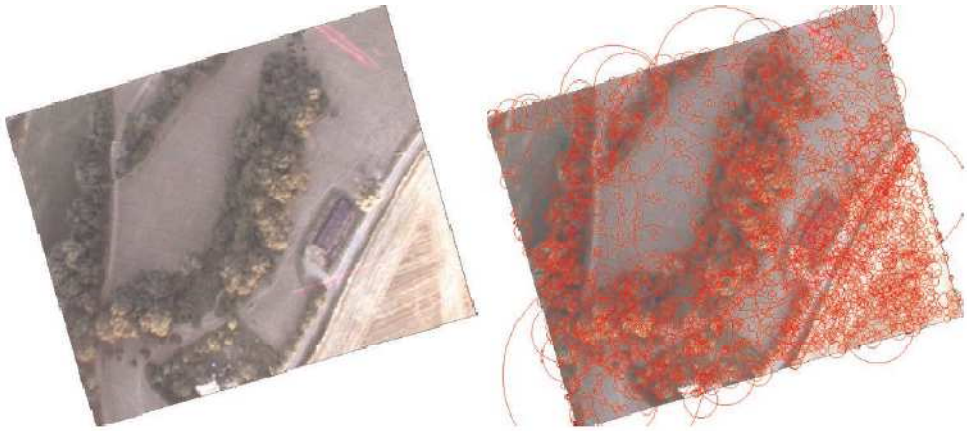


Figure 2.9. *Points invariants sur une image aérienne: la taille du cercle correspond à l'échelle à laquelle a été détecté le point au centre.*

3. **Affectation d'une orientation:** Afin de compléter la description d'un point, il est utile de déterminer sa direction dominante du gradient au voisinage privilégiée définie comme suit: pour chaque point, à l'échelle trouvée, on calcule un vecteur d'amplitude *m*(*x*, *y*) et de direction $\theta(x, y)$:

$$m(x,y) = \sqrt{\big(L(x+1,y) - L(x-1,y)\big)^2 + \big(L(x,y+1) - L(x,y-1)\big)^2}$$

$$\theta(x,y) = tan^{-1}\left[\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}\right] \tag{2.16}$$

2.2.6. Les descripteurs de points clés

Une fois qu'un point d'intérêt a été obtenu et est muni d'une position, d'une échelle et d'une orientation, il est possible de calculer un descripteur.

Calcul du descripteur pour SIFT: Le descripteur est construit à partir des gradients pré-calculés pour tous les niveaux de lissage à l'étape d'extraction. La fenêtre de la zone d'intérêt est divisée en n × n blocs. Dans chaque bloc, un histogramme de *r* valeurs est construit, les amplitudes des gradients locaux sont enregistrées. En général, un histogramme de 8 orientations ($r = 8$) est utilisé. Pour éviter les effets de bords, une interpolation trilinéaire est utilisée pour propager un gradient local dans les cases voisines. Les gradients locaux sont aussi pondérés par une gaussienne qui diminue le poids des gradients les plus éloignés de la position du point d'intérêt. La figure 2.10 illustre la création d'un descripteur SIFT avec une grille de 2x2 histogrammes, chacun ayant 8 orientations possibles.

Pour être robuste à un changement global de contraste, le descripteur est normalisé. Un changement de contraste multiplie les gradients par un coefficient constant. La normalisation du descripteur le rend donc robuste à cette transformation. Il faut noter que ce descripteur est robuste à un changement global de luminosité puisqu'il est calculé à partir de différences de pixel (pour calculer les gradients locaux). Pour être plus robuste aux changements d'illumination non linéaires (par exemple saturation des capteurs ou aux changements d'illuminations sur un objet 3D non plat), les coordonnées du descripteur normalisé sont bornées à 0.2 et le descripteur est ensuite normalisé une seconde fois. La valeur 0.2 a été déterminée expérimentalement dans [LOW04].

Les voisinages des points d'intérêt, ayant des propriétés d'invariance aux changements d'échelle et d'orientation sont définis à l'aide de l'échelle donnée par les extrema des différences de gaussiennes, et de la direction principale du gradient à l'intérieur d'un cercle dont le rayon est proportionnel à cette échelle. Le descripteur du point est constitué d'histogrammes qui regroupent les statistiques sur les directions du gradient dans ce voisinage.

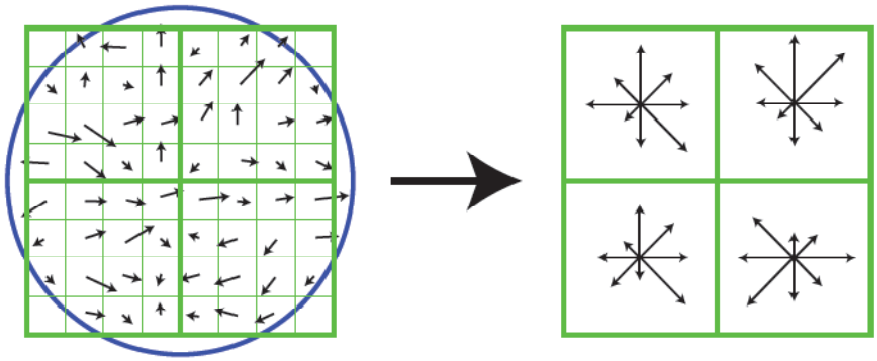


Figure 2.10. *Descripteur d'un point invariant: la fenêtre de gauche représente un ensemble de 16×16 échantillons portant chacun un vecteur issu d'un filtrage par une gaussienne de variance à l'échelle du point invariant considéré. Celle de droite représente le descripteur calculé: celui-ci est divisé en 4 parties contenant chacune un histogramme d'orientations. Chaque orientation est calculée en fonction des vecteurs de la fenêtre de gauche.*

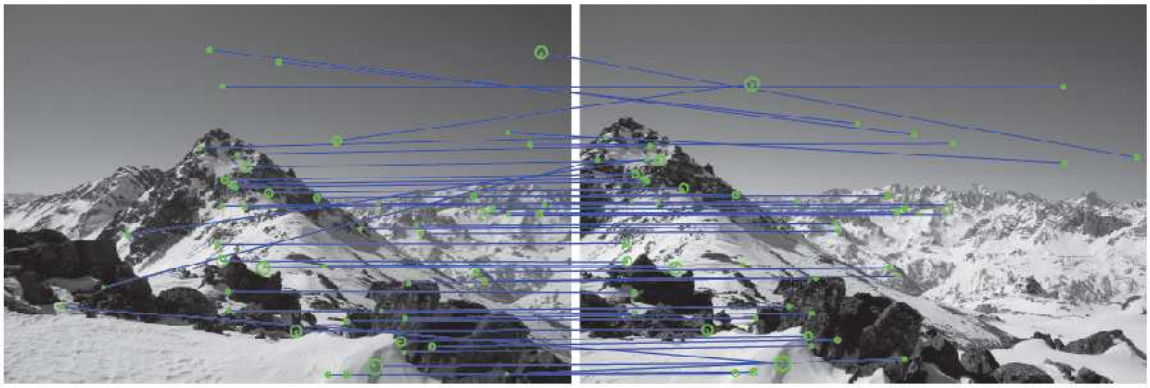


Figure 2.11. *Points d'intérêt issus de la figure 2.7, mis en correspondance par le critère du rapport de distances. Notez les quelques correspondances aberrantes, pour lesquelles les points d'intérêt appariés ne correspondent pas aux mêmes points physiques.*

Remarquons que la direction du gradient est invariante aux changements de contraste, au contraire de la norme du gradient. Le vecteur construit est la concaténation d'histogrammes des directions du gradient. Les votes contribuant à chaque classe sont pondérés par la norme du gradient. Afin de réduire les effets des changements d'éclairage, une normalisation du vecteur ainsi formé est finalement réalisée.

2.3. Difficultés du problème de mise en correspondance sous contraintes photométriques

Les deux grandes caractéristiques du problème de la recherche des plus proches voisins pour les descripteurs locaux SIFT sont la taille des nuages de points et le nombre de dimensions de l'espace dans lequel sont ces descripteurs. Du fait de ces deux caractéristiques, le temps nécessaire pour trouver les plus proches voisins d'un point peut être un frein pour certaines applications.

La seconde caractéristique de ce problème est le nombre de dimensions des espaces étudiés. Dans nos travaux, nous nous sommes concentrés sur les descripteurs locaux SIFT qui génèrent des points dans un espace à 128 dimensions.

Les difficultés liées au grand nombre de dimensions de l'espace étudié sont classiquement regroupées sous le terme « malédiction de la dimension ».

Malédiction de la dimension Dans le domaine de la vision par ordinateur, les espaces utilisés sont souvent le plan ou un espace à 3 dimensions. Il s'avère que les algorithmes utilisés pour ces espaces deviennent inefficaces lorsque le nombre de dimensions augmente (e.g. à partir d'une dimension égale à 10). Cette perte d'efficacité des algorithmes, liée à la grande dimension des espaces est une composante de cette « malédiction de la dimension ».

Pour mieux appréhender les difficultés possibles liées à des espaces de grande dimension, nous en exposons ici quelques particularités (d'autres exemples sont illustrés dans [BÖH01] ou [KOP00]):

- Pour se représenter des propriétés dans un espace de grande dimension, nous avons souvent tendance à imaginer un espace 3D et à extrapoler les conclusions obtenues. Toutefois, cette approche peut être trompeuse. Par exemple, considérons un hypercube dans un espace à d dimensions: $[0,1]^d$. On définit le centre du cube: c = (0.5, …, 0.5). Le lemme « toute sphère qui touche ou intersecte toutes les frontières (de dimension d-1) du cube de dimension d contient le centre c » est vrai en dimension 2 et 3.
  Toutefois, il ne l'est plus en dimension 16. On montre cela par un contre exemple. Considérons le point p = (0.3, …, 0.3). La distance euclidienne entre ce point et le centre est de $\sqrt{16 \times 0.2^2} = 0.8$. On définit la sphère S centrée en *p*, de rayon 0.7. Cette sphère touche ou intersecte tous les bords du cube (surface de dimension 15) mais ne contient pas le point central, et donc ne vérifie pas le lemme.

- Dans [KOP00], l'auteur trace le rapport entre le volume d'une hypersphère de rayon 1 et le volume d'un hypercube de coté 2 pour des dimensions d'espace variables. On voit qu'à partir de 10 dimensions, le volume de l'hypersphère est négligeable par rapport au volume de l'hypercube. Cela a un impact direct sur la conception de structures de recherche pour des espaces de grande dimension.

Ces exemples montrent que le nombre de dimensions est un facteur essentiel du problème. Un algorithme de recherche des plus proches voisins performant, pour un espace à trois dimensions, ne le sera pas forcément pour un espace à 128 dimensions comme celui des SIFT.

2.3.1. Recherche des correspondances SIFT

Dans cette étape, on cherche à trouver une mesure de similarité photométrique entre les descripteurs SIFT, qui établit leur ressemblance ou dissemblance. Définir cette métrique est difficile. Une approche naïve est alors de définir comme appariements ceux dont la distance suivant cette mesure de similarité est en dessous d'un certain seuil. Cependant, mettre un simple seuil sur la distance euclidienne entre les descripteurs afin de définir des appariements ne fonctionne pas. Cette approche, dite du *seuil global*, n'est pas concluante selon [LOW04].

Notons $(f_i^s)_{i\in 1..n_1}$ les descripteurs SIFT de l'image d'entrée Im_s et $(f_i^{ex})_{i\in 1..n_2}$ les descripteurs SIFT de toutes les images existantes. n_2 peut avoir une valeur très élevée. Pour chaque descripteur de Im_s, ses voisins sont cherchés dans $(f_i^{ex})_{i\in 1..n_2}$. Pour rechercher les correspondances avec un descripteur f_i^s de Im_s, il existe trois méthodes [BÖH01] :

1. *Méthode des ratios de distances entre plus proches voisins*

   Pour un descripteur f_i^s, on cherche f_u^{ex} son plus proche voisin et f_v^{ex} le second plus proche voisin:

   $$u = argmin_{u\in 1..n_2} d(f_i^s, f_u^{ex}) \quad (2.17)$$

   et

   $$v = argmin_{v\in 1..n_{2,v\neq j}} d(f_j^s, f_v^{ex}) \quad (2.18)$$

La correspondance entre f_i^set f_u^{ex} *e*st acceptée si le ratio $\frac{d(f_i^s, f_u^{ex})}{d(f_i^s, f_v^{ex})}$ est inférieur à un seuil. Cela met en avant que f_i^s doit ressembler beaucoup à f_u^{ex} mais peu à tous les autres descripteurs de la base. Avec cet algorithme, un descripteur de l'image d'entrée est en correspondance avec un seul descripteur de la base.

2. k plus proches voisins (méthode notée k-NN)

On cherche les k descripteurs dans $(f_j^{ex})_{j\in 1..n_2}$ qui sont les plus proches de f_i^s. Si l'on note $G = \{g_c, c \in [1..k]\}$ les indices de ces descripteurs :

$$\forall j \in G \; et \; \forall l \in 1..n_2, l \notin G, d(f_i^s, f_j^{ex}) < \; d(f_i^s, f_l^{ex}) \qquad (2.19)$$

Dans cette méthode, le nombre de correspondances pour un descripteur de Im_s est *k* (dans cette formulation, les *k* voisins ne sont pas classés selon leur distance au point d'entrée).

3. voisins selon un seuil (méthode notée r-NN)

Tous les points de $(f_j^{ex})_{j\in 1..n_2}$ qui sont à une distance avec f_i^s inférieure à un seuil *r* sont acceptés comme correspondances. Si l'on note $G = \{g_c, c \in [1..k]\}$ les indices de ces descripteurs:

$$\forall j \in G, d(f_i^s, f_j^{ex}) < \; r \; et \; \forall j \notin G, d(f_i^s, f_j^{ex}) \geq \; r \qquad (2.20)$$

Dans l'article qui introduit les SIFT, il est montré que la méthode des ratios qui utilise la distance euclidienne, donne des résultats convenables mais encore pollués par des correspondances aberrantes (la figure 2.11 montre le résultat de cette étape). Une telle approche a pour vertu de supprimer les correspondances pour des points munis de descripteurs peu discriminants, qui ont une distance faible avec de nombreux descripteurs. Supprimer ces points mal caractérisés permet d'éliminer des appariements qui risquent forts d'être faux. Néanmoins, si deux zones contiennent des motifs identiques et sont munies de points d'intérêt, ceux-ci seront aussi éliminés par ce critère, ce qui est alors dommageable pour la suite du processus de notre travail. On a un problème similaire avec l'approche *k*-NN. Comment choisir le paramètre *k*? Si une zone de recouvrement apparait dans un grand nombre d'images, il faudra choisir k grand pour avoir des correspondances sur cette zone dans toutes les images qui la comportent. Mais en contrepartie, avec *k* grand, beaucoup de fausses correspondances vont être acceptées. Enfin, pour l'algorithme r-NN, il commence tout d'abord par établir ses correspondances entre descripteurs voisins, puis, il chercher à décider si des images sont similaires ou non par une étape de vérification affine.

Dans notre thèse, nous allons donc utiliser une nouvelle approche de mise en correspondance photométrique basée sur les ondelettes de Haar. Une fois ces correspondances entre descripteurs voisins établies, on va chercher à décider si des images contiennent des zones de recouvrement similaires ou non.

2.3.2. Approche de mise en correspondance proposée

Plusieurs algorithmes rapides ont été proposés pour réduire la complexité informatique des algorithmes de recherche des *K*-plus proches voisins. Dans notre thèse nous proposons de remplacer l'algorithme Best-Bin-First de SIFT par un autre algorithme plus rapide, plus robuste et de même famille mais basé sur la transformée en ondelettes. La technique exploite l'information importante des coefficients d'approximation du vecteur de coefficient de la transformée, à partir duquel nous obtenons deux inégalités cruciales qui peuvent être employées pour rejeter ces vecteurs pour lesquels il est impossible à être les *k* plus proches voisins. La complexité informatique pour rechercher les *k* plus proches voisins peut être en grande partie réduite. Les résultats expérimentaux sur la classification de texture vérifient l'efficacité de cet algorithme.

2.3.2.1. Propriétés fondamentales:

Nous pouvons utiliser l'ondelette de Haar pour décomposer et reconstruire itérativement un vecteur ***x***, comme suit,

$$\begin{cases} X_{S_{j,r}} = 1/\sqrt{2}\left[X_{S_{j-1,2r}} + X_{S_{j-1,2r+1}}\right] \\ X_{d_{j,r}} = 1/\sqrt{2}\left[X_{S_{j-1,2r}} - X_{S_{j-1,2r+1}}\right] \end{cases}$$

$$\begin{cases} X_{S_{j-1,2r}} = 1/\sqrt{2}\left[X_{S_{j,r}} + X_{d_{j,r}}\right] \\ X_{S_{j-1,2r+1}} = 1/\sqrt{2}\left[X_{S_{j,r}} - X_{d_{j,r}}\right] \end{cases} \quad (2.21)$$

Où $X_{S_{j,r}}$ et $X_{d_{j,r}}$ sont les coefficients d'approximations et de détails à l'échelle *j*, respectivement. Le vecteur spatial original ***x*** est noté $\boldsymbol{X_{S_0}}$ à l'échelle 0. Quand la dimension de ***x***, N, satisfait à $2^{L-1} < N \leq 2^L$, ***x*** peut être entièrement décomposé après *L* niveau de la transformée en ondelettes, c.-à-d., il y a seulement un coefficient approximatif noté par $X_{S_{L,0}}$. En raison de l'orthogonalité de la transformée en ondelettes de Haar, il y a $d(x, y) = d(X, Y)$ pour deux vecteurs ***x*** et ***y***. ***X*** et ***Y*** sont les vecteurs des coefficients de la transformée de ***x*** et ***y***, respectivement. $d(x,y) = \sqrt{\sum_{i=0}^{N-1}(x_i - y_i)^2}$ est la distance euclidienne entre ***x*** et ***y***. Ainsi, les *K* plus proches voisins, recherchés dans le domaine spatial, sont les mêmes que ceux dans le domaine d'ondelette.

Supposons que les *K* plus proches voisins du vecteur d'entrée ***Q*** sont Y^{j_1}, Y^{j_2}, …, Y^{j_k} dans les vecteurs vérifiés de $\widetilde{\Omega} = \{Y^1, Y^2, \ldots, Y^n\}$ (***Q***, Y^1,…, Y^{n-1} et Y^n sont les vecteurs des coefficients de la transformée de ***q***, y^1,…, y^{n-1} et y^n, respectivement), et les *k* distances correspondantes sont $d(Q, Y^{j_1}) \leq d(Q, Y^{j_2}) \leq \ldots \leq d(Q, Y^{j_k})$. Pour un vecteur $Y^i \in \widetilde{\Omega}$, afin d'obtenir la relation entre $d(Q, Y^i)$ et $d(Q, Y^{j_k})$, WKPDS [HWA98] utilise la méthode suivante:

$$\underline{\text{Si}}\ \ \sum_{r=0}^{p}(Q_r - Y_r^i)^2 \geq d^2(\boldsymbol{Q}, \boldsymbol{Y}^{j_k}), p \leq N' - 1\ \underline{\text{alors}}\ \ d(\boldsymbol{Q}, Y^i) \geq d(\boldsymbol{Q}, Y^{j_k})$$

Où N'est la dimension du vecteur de coefficients de la transformée. Pan et al. [PAN04] ont adopté une condition supplémentaire d'arrêt, si

$$Y_{S_{L,0}}^i \leq Q_{S_{L,0}} - \sqrt{\frac{N}{2^L}}\, d(\boldsymbol{Q}, \boldsymbol{Y}^{j_k})\ \ ou\ \ Y_{S_{L,0}}^i \geq Q_{S_{L,0}} + \sqrt{\frac{N}{2^L}}\, d(\boldsymbol{Q}, \boldsymbol{Y}^{j_k}) \qquad (2.22)$$

Où $Q_{S_{L,0}}$ et $Y_{S_{L,0}}^i$ sont les coefficients approximatifs de $\boldsymbol{q}$ et $\boldsymbol{y}^i$ à l'échelle *L*, respectivement.

Une fois que $d(\boldsymbol{Q}, \boldsymbol{Y}^i) < d(\boldsymbol{Q}, \boldsymbol{Y}^{j_k})$, remplacer $\boldsymbol{Y}^{j_k}$ avec $\boldsymbol{Y}^i$, puis réorganiser les nouveaux *k* plus proches voisins actuels et les *K* distances correspondant à [HWA98].
Lorsque tous les vecteurs de l'ensemble des données sont vérifiés, les k vecteurs les plus courants sont les k plus proches voisins finaux.

2.3.2.2. Algorithme proposé

Ici nous présentons l'algorithme utilisé dans notre travail et qui remplace l'algorithme d'appariement Best-Bin-First de SIFT. Si le vecteur *x* est transformé par l'exécution (L-1) niveau de décomposition par l'ondelette de Haar.

Dans ce qui suit, nous allons détailler le noyau de l'algorithme rapide proposé.

Proposition : Pour deux vecteurs $\boldsymbol{x}$, $\boldsymbol{y}$ de *N* dimensions ($2^{L-1} < N \leq 2^L$), après une décomposition avec l'ondelette de Haar, on obtient:

$$\begin{aligned} \left|X_{S_{L-1,0}} - Y_{S_{L-1,0}}\right| &\leq \sqrt{\frac{2^{L-1}}{2^{L-1}}}\, d(\boldsymbol{X}', \boldsymbol{Y}') \leq d(\boldsymbol{X}, \boldsymbol{Y}) = d(\boldsymbol{x}, \boldsymbol{y}) \\ \left|X_{S_{L-1,1}} - Y_{S_{L-1,1}}\right| &\leq \sqrt{\frac{N - 2^{L-1}}{2^{L-1}}}\, d(\boldsymbol{X}^{"}, \boldsymbol{Y}^{"}) \\ &\leq \sqrt{\frac{N - 2^{L-1}}{2^{L-1}}}\, d(\boldsymbol{X}, \boldsymbol{Y}) = \sqrt{\frac{N - 2^{L-1}}{2^{L-1}}}\, d(\boldsymbol{x}, \boldsymbol{y}) \end{aligned} \qquad (2.23)$$

Où :

$X_{S_{L-1,r}}$ et $Y_{S_{L-1,r}}$ sont les coefficients d'approximations à l'échelle *L-1*, *r* =0,1.

$\boldsymbol{X'}$ et $\boldsymbol{X}^{"}$ sont les vecteurs de coefficients de la transformée de $\boldsymbol{x}'$ et $\boldsymbol{x}$", respectivement.

x' dénote un sous-vecteur constitué des 2^{L-1} premiers éléments de $\boldsymbol{x}$.

$\boldsymbol{x}$" comprend les $N\text{-}2^{L-1}$ derniers éléments de $\boldsymbol{x}$, constituant le deuxième sous-vecteur.

Pour $\boldsymbol{Y}'$ et $\boldsymbol{Y}^{"}$, le principe est le même que celui de $\boldsymbol{X'}$ et $\boldsymbol{X}^{"}$, respectivement.

Démonstration: Soit $\boldsymbol{x} = (x_0, x_1, \dots, x_{N-1})$, $\boldsymbol{x'} = (x_0, x_1, \dots, x_{2^{L-1}-1})$ et $\boldsymbol{x''} = (x_{2^{L-1}}, x_{2^{L-1}+1}, \dots, x_{N-1})$. Les valeurs moyennes de $\boldsymbol{x}$, $\boldsymbol{x'}$, $\boldsymbol{x''}$ sont données par $M_x = \frac{1}{N}\sum_{i=0}^{N-1} x_i$, $M_{x'} = \frac{1}{2^{L-1}}\sum_{i=0}^{2^{L-1}-1} x_i$ et $M_{x''} = \frac{1}{N-2^{L-1}}\sum_{i=2^{L-1}}^{N-1} x_i$ respectivement. A partir de la transformée en ondelettes de Haar d'équation (2.21), on peut voir que le coefficient approximatif $X_{S_{j,r}}$ à l'échelle j a et seulement a une relation avec les deux coefficients approximatifs $X_{S_{j-1,2r}}$ et $X_{S_{j-1,2r+1}}$ à l'échelle *j-1*, qui est montrée dans la figure 2.12. Ainsi,

$$X_{S_{L-1,0}} = 1/\sqrt{2}\,\left(X_{S_{L-2,0}} + X_{S_{L-2,1}}\right)$$
$$= 1/\sqrt{2}\,\left(1/\sqrt{2}\left(X_{S_{L-3,0}} + X_{S_{L-3,1}}\right) + 1/\sqrt{2}\left(X_{S_{L-3,0}} + X_{S_{L-3,1}}\right)\right)$$

$$= \cdots = \frac{1}{\sqrt{2^{L-1-j}}}\sum_{i=0}^{2^{L-1-j}-1} X_{S_{j,i}} = \frac{1}{\sqrt{2^{L-j}}}\sum_{i=0}^{2^{L-j}-1} X_{S_{j-1,i}} = \dots$$

$$= \cdots = \frac{1}{\sqrt{2^{L-1}}}\sum_{i=0}^{2^{L-1}-1} X_{S_{0,i}} = \frac{1}{\sqrt{2^{L-1}}}\sum_{i=0}^{2^{L-1}-1} X_i = \frac{2^{L-1}}{\sqrt{2^{L-1}}}\,M_{x'}$$

On peut considérer que $X_{S_{L,0}}$ est équivalent à $N.M_x/\sqrt{2^L}$ [PAN04].

Comme $X_{S_{L,0}} = 1/\sqrt{2}\,\left(X_{S_{L-1},0} + X_{S_{L-1},1}\right)$, il y a $X_{S_{L-1,1}} = \sqrt{2}X_{S_{L,0}} - X_{S_{L-1,0}}$

$$= N.M_x/\sqrt{2^{L-1}} - 2^{L-1}\,M_{x'}\Big/\sqrt{2^{L-1}} = \frac{1}{\sqrt{2^{L-1}}}\sum_{i=2^{L-1}}^{N-1} x_i = \frac{N-2^{L-1}}{\sqrt{2^{L-1}}}\,M_{x''}$$

De la même manière, on peut obtenir:

$$Y_{S_{L-1,0}} = 2^{L-1}.M_{y'}/\sqrt{2^{L-1}} \text{ et } Y_{S_{L-1,1}} = (N-2^{L-1}).M_{y''}/\sqrt{2^{L-1}}$$

Et en combinant l'inégalité suivante [GUA92]: $d(x,y) \geq \sqrt{N}\,\left|M_x - M_y\right|$ et les formules ci-dessus, on peut facilement obtenir:

$$\left|X_{S_{L-1,0}} - Y_{S_{L-1,0}}\right| = \left|2^{L-1}.M_{x'}/\sqrt{2^{L-1}} - 2^{L-1}.M_{y'}/\sqrt{2^{L-1}}\right|$$
$$\leq \sqrt{2^{L-1}/2^{L-1}}\;d(\boldsymbol{x'},\boldsymbol{y'}) = \sqrt{2^{L-1}/2^{L-1}}\;d(\boldsymbol{X'},\boldsymbol{Y'})$$
$$\leq d(\boldsymbol{X},\boldsymbol{Y}) = d(\boldsymbol{x},\boldsymbol{y})$$

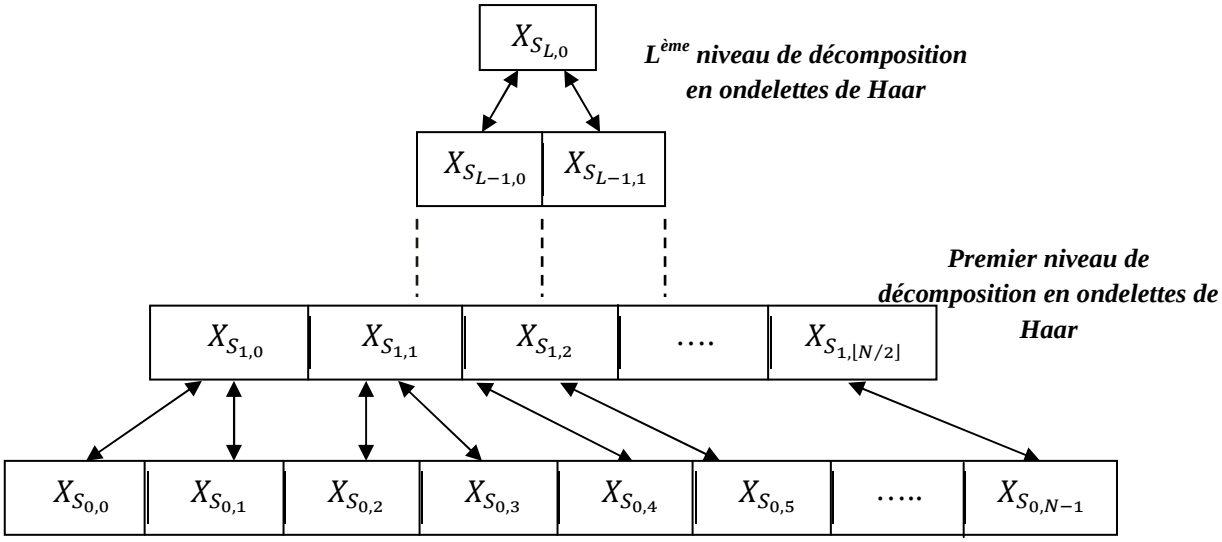


Figure 2.12. *Relations entre coefficients approximatifs aux différentes échelles.*

Noter que, **si** $Y^i_{S_{L-1,0}} \leq Q_{S_{L-1,0}} - d(\boldsymbol{Q}, \boldsymbol{Y^{j_k}})$ ou $Y^i_{S_{L-1,0}} \geq Q_{S_{L-1,0}} + d(\boldsymbol{Q}, \boldsymbol{Y^{j_k}})$,

$$\text{ou } Y^i_{S_{L-1,1}} \leq Q_{S_{L-1,1}} - \sqrt{(N - 2^{L-1})/2^{L-1}}\, d(\boldsymbol{Q}, \boldsymbol{Y^{j_k}}) \qquad (2.24)$$

$$\text{ou } Y^i_{S_{L-1,1}} \geq Q_{S_{L-1,1}} + \sqrt{(N - 2^{L-1})/2^{L-1}}\, d(\boldsymbol{Q}, \boldsymbol{Y^{j_k}}),$$

alors $d(\boldsymbol{Q}, \boldsymbol{Y^i}) \geq d(\boldsymbol{Q}, \boldsymbol{Y^{j_k}})$.

Avec les résultats ci-dessus, on peut déterminer si $\boldsymbol{Y^i}$ est plus proche que $\boldsymbol{Y^{j_k}}$ de $\boldsymbol{Q}$ avec trois étapes. Premièrement, il faut vérifier le vecteur $\boldsymbol{Y^i}$ avec l'équation (2.22) [PAN04]. Si on ne peut pas obtenir de conclusion, l'équation (2.24) est utilisée. Une fois que n'importe quelle inégalité d'équation (2.24) est vraie, il y aura $d(\boldsymbol{Q}, \boldsymbol{Y^i}) \geq d(\boldsymbol{Q}, \boldsymbol{Y^{j_k}})$. Pendant la phase de validation avec l'équation (2.24), $Y^i_{S_{L-1,0}}$ et $Y^i_{S_{L-1,1}}$ peuvent être obtenue de $Y^i_{S_{L,0}}$ et $Y^i_{S_{L,1}}$ avec la transformée inverse en ondelettes de Haar. Ainsi, il n'a pas besoin d'espace de stockage supplémentaire pour les conserver avant de chercher les *k*-plus proches voisins. Cependant, nous remarquons qu'il ya deux opérateurs de multiplication dans la transformée inverse de l'équation (2.21), qui consomment beaucoup de temps au cours du processus de recherche.

Donc, les modifications sont comme suit:

$$\begin{cases} X^{mod}_{S_{L-1,0}} = X_{S_{L,0}} + X_{d_{L,0}} \\ X^{mod}_{S_{L-1,1}} = X_{S_{L,0}} - X_{d_{L,0}} \end{cases}$$

L'équation (2.23) devient:

$$\left| X^{mod}_{S_{L-1,0}} - Y^{mod}_{S_{L-1,0}} \right| \leq \sqrt{\frac{2^L}{2^{L-1}}}\, d(\boldsymbol{X'}, \boldsymbol{Y'}) \leq \sqrt{2}\, d(\boldsymbol{X}, \boldsymbol{Y})$$

$$\left|X_{S_{L-1,1}}^{mod} - Y_{S_{L-1,1}}^{mod}\right| \leq \sqrt{\frac{2N-2^L}{2^{L-1}}}\, d(\boldsymbol{X}", \boldsymbol{Y}") \leq \sqrt{\frac{2N-2^L}{2^{L-1}}}\, d(\boldsymbol{X}, \boldsymbol{Y})$$

En conséquence, l'équation (2.24) est ajustée comme,

si $(Y_{S_{L-1,0}}^{i})^{mod} \leq (Q_{S_{L-1,0}})^{mod} - \sqrt{2}d(\boldsymbol{Q}, \boldsymbol{Y^{j_k}})$ <u>ou</u> $(Y_{S_{L-1,0}}^{i})^{mod} \geq (Q_{S_{L-1,0}})^{mod} + \sqrt{2}d(\boldsymbol{Q}, \boldsymbol{Y^{j_k}})$

$$\underline{\text{ou}}\ (Y_{S_{L-1,1}}^{i})^{mod} \leq (Q_{S_{L-1,1}})^{mod} - \sqrt{(2N - 2^L)/2^{L-1}}\, d\left(\boldsymbol{Q}, \boldsymbol{Y^{j_k}}\right) \qquad (2.25)$$

$$\underline{\text{ou}}\ (Y_{S_{L-1,1}}^{i})^{mod} \geq (Q_{S_{L-1,1}})^{mod} + \sqrt{(2N - 2^L)/2^{L-1}}\, d(\boldsymbol{Q}, \boldsymbol{Y^{j_k}}),$$

alors $d(\boldsymbol{Q}, \boldsymbol{Y^{i}}) \geq d(\boldsymbol{Q}, \boldsymbol{Y^{j_k}})$.

Cette version modifiée peut conserver de nombreuses opérations de multiplication. Ainsi, l'équation (2.25) remplace l'équation (2.24) et sert comme une deuxième condition de validation. Maintenant, le nouvel algorithme proposé se résume comme suit:

Etape 0: Transformer tous les vecteurs de l'ensemble de vecteurs Ω et les trier par ordre croissant en fonction de leurs coefficients approximatifs. Supposons que l'ensemble de vecteurs prétraité est $\widetilde{\Omega} = \{\boldsymbol{Y^1}, \boldsymbol{Y^2}, \ldots, \boldsymbol{Y^n}\}$. Où $\boldsymbol{Y^i}$ est le vecteur de coefficient de transformation du vecteur $\boldsymbol{y}^{\text{i}}$, i = 1, …, n. Par souci de simplification de l'expression, on suppose implicitement que l'ordre de l'ensemble de vecteurs d'origine est le même que celui de l'ensemble de vecteurs de coefficients de transformation réorganisés.

Etape 1: Pour un vecteur requête $\boldsymbol{q}$, décomposer le entièrement avec l'ondelette de Haar et obtenir le vecteur de coefficient de transformation $\boldsymbol{Q}$. Au cours de la décomposition, stocker $(Q_{S_{L-1,0}})^{mod}$et $(Q_{S_{L-1,1}})^{mod}$

Etape 2: Calculer $J = \underset{j}{argmin}\left|Q_{S_{L,0}} - Y_{S_{L,0}}^{j}\right|$ (J est l'argument qui minimise le terme $\left|Q_{S_{L,0}} - Y_{S_{L,0}}^{j}\right|$). Initialiser les k plus proches vecteurs actuels et à petites distances avec les k plus proches vecteurs de $\boldsymbol{Y^J}$. Cette méthode adoptée est indiqué comme suit, $d_i = d(\boldsymbol{X}, \boldsymbol{Y}^{j_i})$, i = 1, 2, …, k, $j_i = J - \lfloor k/2 \rfloor + i$. Lorsque $J - \lfloor k/2 \rfloor < 1$, alors mettre $j_i = i$. Si $J - \lfloor k/2 \rfloor < n - k + 1$, mettre $j_i = n - k + i$. Soit $I_Haut = j_k$ et $I_Bas = j_1$. Ensuite, réorganiser les k distances tel que $d_1 \leq d_2 \leq \ldots \leq d_k$ et les k plus proches vecteurs correspondant sont également réorganisées. Mettez $l = I_Haut$ +1. L'ordre de recherche est effectué en haut et en bas comme il est montré sur la figure 2.13.

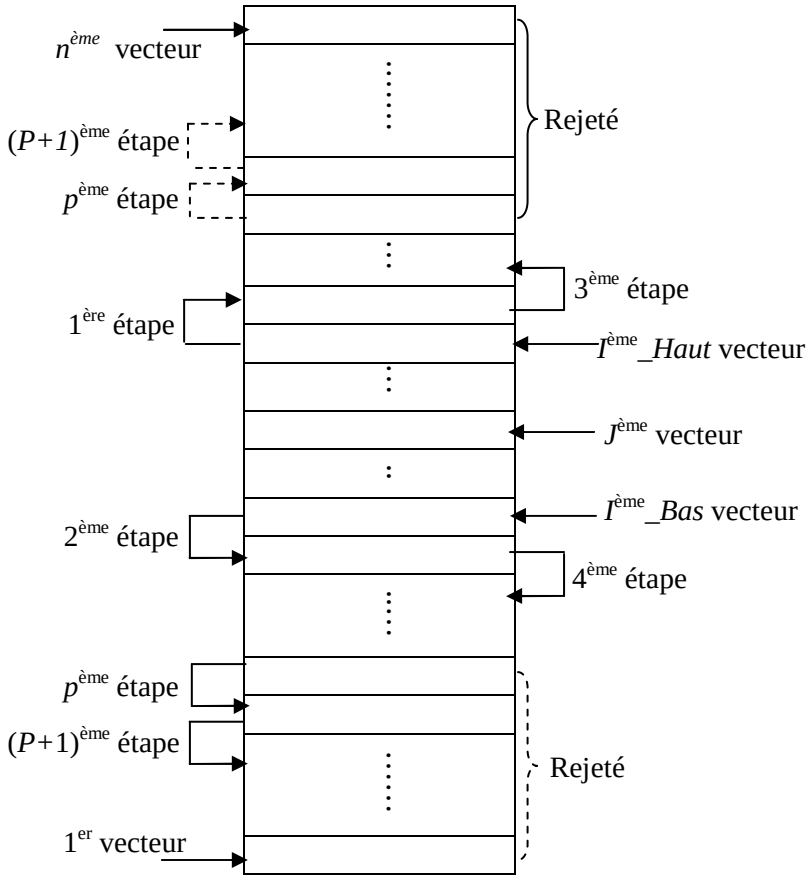


Figure 2.13. *Ordre de recherche.*

Etape 3: Vérifier l'arrêt de ce programme (Il n'y a aucun vecteur à tester). Testez l'équation (2.22) sur le vecteur $\boldsymbol{Y^l}$, comme suit: Lorsque $1 \leq l <$*I_Bas*, effectuer l'étape 1). Sinon, passez à l'étape 2).

I) Si la première inégalité est satisfaite, alors vecteurs $\boldsymbol{Y^j}$, $1 \leq j \leq l$, seront rejetés et choisissez $\boldsymbol{Y^l}$ à partir de $\widetilde{\Omega}$ selon l'ordre de recherche montré sur la figure 2.13, passez à l'étape 3. Sinon, passez à l'étape 4.

II) Si la deuxième inégalité est satisfaite, alors les vecteurs $\boldsymbol{Y^j}$, $l \leq j \leq n$ seront éliminés et choisissez $\boldsymbol{Y^l}$ à partir de $\widetilde{\Omega}$ selon l'ordre de recherche montré sur la figure 2.13, passez à l'étape 3. Sinon, passez à l'étape 4.

Etape 4: Calculer $(Y^l_{S_{L-1,0}})^{mod}$ et $(Y^l_{S_{L-1,1}})^{mod}$, puis vérifiez l'équation (2.25) sur $\boldsymbol{Y^l}$. Si l'une des inégalités est vraie, nous pouvons rejeter $\boldsymbol{Y^l}$ car il est impossible pour qu'il soit l'un des k plus proches voisins. Ensuite, choisissez $\boldsymbol{Y^l}$ à partir de $\widetilde{\Omega}$ selon l'ordre de recherche montré sur la figure 2.13, passez à l'étape 3. Sinon, passez à l'étape suivante.

Etape 5: Déterminer la relation entre $d(\boldsymbol{Q}, \boldsymbol{Y^l})$ et d_k. Si $d(\boldsymbol{Q}, \boldsymbol{Y^l}) < d_k$, $\boldsymbol{Y}^{j_k}$ sera remplacé par $\boldsymbol{Y^l}$, soit $d_k = d(\boldsymbol{Q}, \boldsymbol{Y^l})$, puis réorganiser les k distances tel que $d_1 \leq d_2 \leq \ldots \leq d_k$ et les k plus proches voisins correspondants sont également réorganisés. Choisissez $\boldsymbol{Y^l}$ à partir de $\widetilde{\Omega}$ selon l'ordre de recherche montré sur la figure 2.13, passez à l'étape 3.

Comme le montre la figure 2.13, la recherche sera effectuée de manière haut-bas. Si l'équation (2.22) sera satisfaite dans les deux sens, la recherche va rejeter tous les vecteurs non contrôlés dans cette direction et continuer dans une autre direction. Par exemple, les vecteurs de la direction supérieure ont été rejetés à la p-1 étape, puis p et p+1 étape, ce qui va permettre de vérifier seulement les vecteurs dans le sens de la descente. Lorsque l'algorithme est terminé, les *k* plus proches vecteurs finaux "jusqu'ici" sont les *k* plus proches voisins du vecteur requête $\boldsymbol{Q}$.

L'algorithme rapide proposé présente les avantages suivants. Tout d'abord, il recherche les *k* plus proches voisins de façon plus pertinente. Deuxièmement, comparé par rapport à WKPDS [HWA98] et l'algorithme de Pan [PAN04], l'ensemble de vecteurs prétraité n'a pas besoin d'espace de stockage supplémentaire. Enfin, il est beaucoup plus rapide que les deux techniques mentionnées ci-dessus.

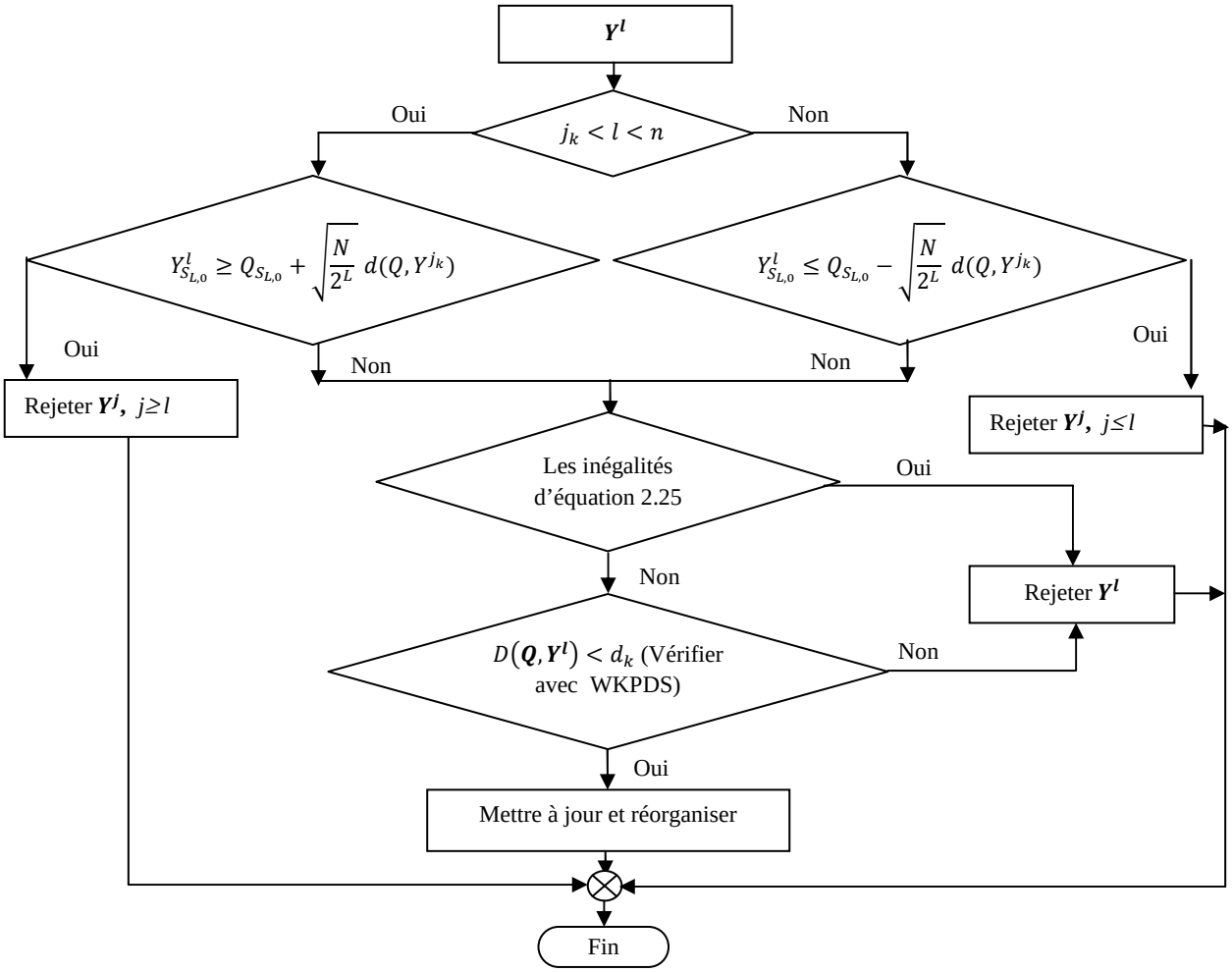


Figure 2.14. *Organigramme du procédé de contrôle.*

2.3.2.3. Résultats de simulation

Cette partie présente quelques résultats de simulation pour la classification d'images de texture. 30 Brodatz images de texture complexes [BRO06] sont utilisées, chaque image est de taille 512 × 512 pixels avec 256 niveaux de gris.

Dimension	Algorithme	K=5	K=3	K=1
		Durée	Durée	Durée
8	**WKPDS**	20.34	19.95	18.60
	BBF	4.04	3.70	2.13
	Algorithme proposé	1.10	0.84	0.48
13	**WKPDS**	28.55	27.73	24.59
	BBF	5.28	4.64	3.50
	Algorithme proposé	2.58	2.08	1.23
16	**WKPDS**	26.64	25.67	23.13
	BBF	7.95	6.16	4.78
	Algorithme proposé	3.84	3.17	2.05
128	**WKPDS**	35.08	33.71	31.89
	BBF	9.01	8.70	7.33
	Algorithme proposé	4.25	3.59	3.12

Tableau 2.1. *Comparaison de la durée de classification.*

Pour chaque image, 400 échantillons d'images de taille 512 × 512 sont choisis au hasard à partir de l'image originale, 300 échantillons d'images sont utilisés comme échantillons d'apprentissage, et 100 échantillons sont utilisés pour tester l'algorithme. Dans cet algorithme, l'extrema des densités de coefficients d'ondelettes sont utilisés comme vecteur de caractéristiques [PAN99] pour classifier les images de texture.

Le tableau 2.1 présente le temps de calcul, exprimé en secondes, de l'ensemble des méthodes et implémentations listées précédemment. Pour cette étude, les ensembles de points que sont les références et les requêtes sont de même taille et contiennent des données différentes. Dans le cas où les deux ensembles seraient identiques, le problème serait plus simple; nous considérons le problème le plus dur. Le temps de calcul dépend de la dimension de ses points et du paramètre *k* correspondant au nombre de voisins à considérer (*k*-PPV).

Lorsque l'on compare notre algorithme avec celui de BBF, la réduction moyenne du temps de classification est d'environ 36.7%, 42.0% et 48.3% quand k = 5, 3 et 1, respectivement. La plus forte réduction du temps de classification est d'environ 57.9%. En comparant avec l'algorithme WKPDS, la plus petite réduction est d'environ 86%. A partir de ces résultats, nous pouvons dire que l'algorithme proposé permet de réduire largement la charge calculatoire de la recherche des *k*-plus proches voisins quelle que soit la dimension des vecteurs d'entrées.

Le temps de calcul augmente linéairement avec la dimension quelle que soit la méthode utilisée. Ce résultat semble logique dans la mesure où le nombre d'opérations nécessaires pour calculer la distance entre deux vecteurs dépend linéairement de la dimension. Bien que linéaire, l'influence de la dimension n'est pas égale entre les différentes méthodes et implémentations. Cette influence est exprimée par la pente de la fonction linéaire représentant le temps de calcul en fonction de la dimension. Nous pouvons constater que l'approche proposée est moins sensible à la dimension en comparaison des autres méthodes. En d'autres termes, plus la dimension est importante, plus l'utilisation de la méthode proposée est intéressante. Cette particularité est très intéressante pour des applications où les dimensions utilisées sont élevées comme dans notre cas de travail.

Afin d'évaluer la complexité calculatoire de la méthode proposée, les résultats de comparaison au sujet de l'addition (Add), la multiplication (MULT) et de la comparaison (Comp) par vecteur pour trouver le plus proche voisin sont présentés dans les tableaux 2.2 à 2.4. « Num » désigne le nombre moyen de vecteurs dans l'ensemble de conception sur lequel WKPDS sera exécuté pour un vecteur de test. L'algorithme proposé utilise les équations (2.22) et (2.25) pour rejeter ces vecteurs impossibles d'être les *k*-plus proches voisins. Ces vecteurs qui ne peuvent pas être rejetés par les deux équations doivent être déterminés par WKPDS, qui est la contribution dominante de la complexité des calculs. La différence entre l'algorithme proposé et celui de BBF est que l'équation (2.25) est adoptée. Ainsi le terme, « Num », peut être utilisé pour indiquer combien de vecteurs ont été rejetés par la nouvelle équation (2.25).

Algorithme	Mult	Add	Comp	Num
KNN	72000	135000	8999	//
WKPDS	13125.0	35249.3	22103.0	9000
BBF	1196.4	2379.7	1696.4	495.6
Algorithme proposé	398.0	1776.2	2661.8	90.9

Tableau 2.2. *Complexité arithmétique pour des vecteurs de caractéristiques à 8-dimensions.*

Algorithme	Mult	Add	Comp	Num
KNN	117000	225000	8999	//
WKPDS	19608.0	48202.4	28565.0	9000
BBF	2628.4	5229.8	3589.9	978.5
Algorithme proposé	1072.8	4077.7	5649.6	205.4

Tableau 2.3. *Complexité arithmétique pour des vecteurs de caractéristiques à 13-dimensions.*

Algorithme	Mult	Add	Comp	Num
KNN	144000	279000	8999	//
WKPDS	17920.3	44822.5	26874.9	9000
BBF	3878.7	7728.3	5337.9	1479.5
Algorithme proposé	1726.9	6385.9	8888.5	397.3

Tableau 2.4. *Complexité arithmétique pour des vecteurs de caractéristiques à 16-dimensions.*

A partir de l'analyse de complexité arithmétique des tables ci-dessus, nous pouvons voir que l'algorithme proposé réduit une grande quantité d'opérations de multiplication et d'addition au prix d'une augmentation des opérations de comparaison, de sorte que la charge de calcul global est réduite. La raison réside dans le fait que le coût de calcul des opérations de multiplication et d'addition est beaucoup plus lourd que celui de l'opération de comparaison. Le terme, « Num », indique que l'équation (2.25) proposée rejette beaucoup de vecteurs impossibles d'être le plus proche voisin, ce qui réduit largement la complexité de calcul.

2.4. Conclusion:

L'objectif de ce chapitre était d'abord d'introduire les principales approches de la description locale d'une image. Cette description repose sur les étapes de détection de points d'intérêt et sur la construction de vecteurs de description avec différentes méthodes, en particulier, celle de SIFT. De manière générale, la description locale permet de mettre en correspondance des points dans des images de textures très complexes (images satellitaires ou aériennes) et qui présentent des zones de recouvrement avec différents pourcentage. Ces correspondances sont ensuite utilisées pour de nombreuses applications telles que le recalage ou le mosaïquage d'images traitées dans notre travail de thèse.

La mise en correspondance est un des points les plus problématiques de la méthode. Pour réaliser l'appariement de points d'intérêt, il est nécessaire de se munir d'une mesure de similarité photométrique entre les descripteurs, qui établit leur ressemblance ou dissemblance. On a vu que définir cette métrique est difficile. Une approche simple est alors de définir comme appariements ceux dont la distance suivant cette mesure de similarité est en dessous d'un certain seuil. Cependant, mettre un simple seuil sur la distance euclidienne entre les descripteurs afin de définir des appariements ne fonctionne pas dans le cas des images contenant beaucoup d'informations tel que les images satellites ou aériennes.

Une méthode habituelle utilisée dans l'algorithme SIFT, montrée comme plus robuste, pour définir un ensemble d'appariements est de garder le plus proche voisin, au sens de la distance entre descripteurs, pourvu que le rapport de distances entre le plus proche et le second plus proche voisin soit inférieur à un seuil fixé. Le plus proche voisin est d'autant plus fiable que le rapport des distances est faible. Cette approche, qui utilise la distance

euclidienne, donne des résultats convenables mais encore pollués par des correspondances aberrantes. Une telle approche a pour vertu de supprimer les correspondances pour des points munis de descripteurs peu discriminants, qui ont une distance faible avec de nombreux descripteurs. Supprimer ces points mal caractérisés permet d'éliminer des appariements qui risquent forts d'être faux. Néanmoins, si deux zones contiennent des motifs identiques et sont munies de points d'intérêt, ceux-ci seront aussi éliminés par ce critère, ce qui est alors dommageable pour la suite du processus de mise en correspondance géométrique.

Pour pallier ces problèmes, nous avons proposé dans notre thèse une nouvelle approche améliorée des *K*-plus proches voisins basée sur les ondelettes de Haar. Cette dernière exploite l'information importante des coefficients d'approximation de la transformée, à partir de laquelle nous obtenons deux inégalités cruciales qui peuvent être employées pour rejeter vecteurs dont il est impossible d'être les *k* plus proches voisins. Nous avons pu constater que cette approche a réduit en grande partie la charge calculatoire de la recherche tout en améliorant la qualité d'appariements, ce qui a était confirmé par les résultats expérimentaux. Finalement, pour permettre de traiter plus d'appariements potentiels tout en obtenant de meilleurs résultats, nous allons aborder dans le chapitre suivant les problèmes et les propositions liées à la mise en correspondance sous contraintes géométriques en proposant une nouvelle approche de l'algorithme RANSAC.

CHAPITRE 3

MISE EN CORRESPONDANCE DE POINTS D'INTERET SOUS CONTRAINTES GEOMETRIQUES

3.1. Filtrage robuste à l'aide de la contrainte géométrique

Une fois qu'un ensemble d'appariements a été extrait des images, nous cherchons à sélectionner un sous-ensemble constitué d'appariements qui sont compatibles avec un modèle géométrique sous-jacent. Supposons que le mouvement de la caméra ne se limite pas à une rotation autour de son centre optique, et que les points 3D observés ne sont pas tous coplanaires. Lorsque l'on suit le modèle de sténopé, la *géométrie épipolaire* est caractérisée par la *matrice fondamentale* (ou la matrice essentielle si les paramètres intrinsèques de la caméra sont connus) [FAU01]. Comme les appariements sont pollués par des couples de points d'intérêt mal mis en correspondance – des appariements qui se ressemblent, mais ne correspondent pas au même point 3D – appelés appariements aberrants par la suite (*outliers* en anglais), des méthodes de statistiques robustes, telles que les LMedS or M-estimateurs sont employées. Une méthode habituellement retenue pour réaliser le filtrage robuste est RANSAC, ainsi que de nombreuses variantes qui en découlent (MSAC, MLESAC [TOR00], MAPSAC [TOR02] et d'autres encore [CHU05] sans vouloir être exhaustif).

3.2. Modélisation de la caméra

Dans cette modélisation, nous définissons 3 systèmes de coordonnées distincts (Figure 3.1):

- Un repère 3D ($O_wX_wY_wZ_w$) orthonormé lié à la scène (au monde): c'est un système de référence absolu défini par l'utilisateur et tous les objets du monde y compris la caméra, sont définis par rapport à ce système de coordonnées. L'indice w se rapporte au monde.

- Un repère 3D ($O_cX_cY_cZ_c$) orthonormé dont l'origine est le centre de la caméra, l'indice c se rapporte à la caméra.

- Un repère 2D ($O_sX_sY_s$) orthonormé dont l'origine est en haut à gauche du plan image et dont les axes X_s et Y_s sont parallèles respectivement aux lignes et colonnes de l'image. L'indice s se rapporte à l'image.

La caméra est modélisée en position canonique (Figure 3.1) où l'axe Z_c est vers l'avant de la caméra, l'axe Y_c est vers le haut et l'axe X_c est vers la gauche.

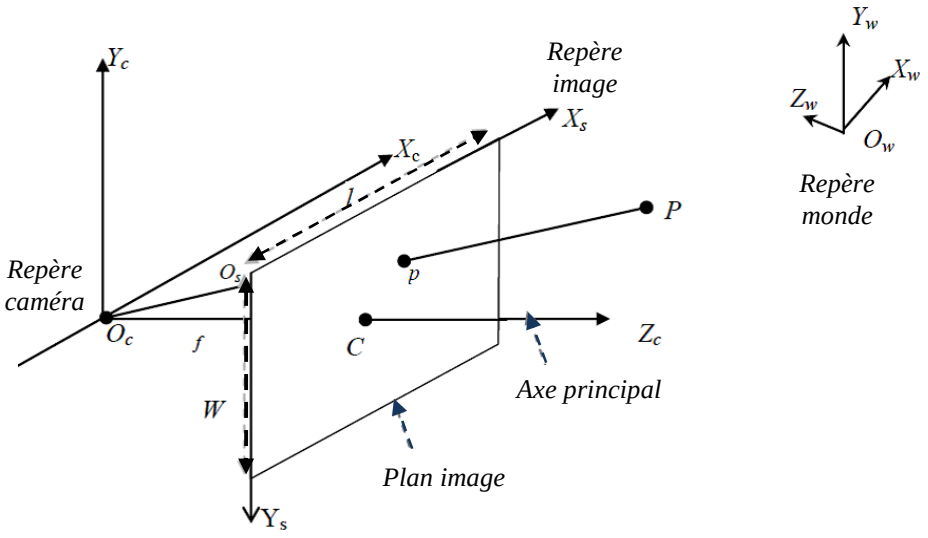


Figure 3.1. *Modélisation de la caméra et projection d'un point 3D.*

où O_c : Centre de la caméra (position de la caméra dans le repère $(O_wX_wY_wZ_w)$.

O_cZ_c : axe optique de la caméra; c'est l'axe principal.

C : Point principal (intersection entre l'axe optique et le plan image).

P : Un point de la scène 3D.

P : Projection du point P dans l'image.

f : Distance focale entre le plan image et le centre Oc de la caméra ($f = OcC$).

W : Largeur du plan image.

l : Longueur du plan image.

La largeur du plan image W et la focale f de la caméra, permettent de déterminer l'angle de vue β de la caméra (Figure 3.2).

$$\tan\left(\frac{\beta}{2}\right) = \left(\frac{W}{2f}\right) \tag{3.1}$$

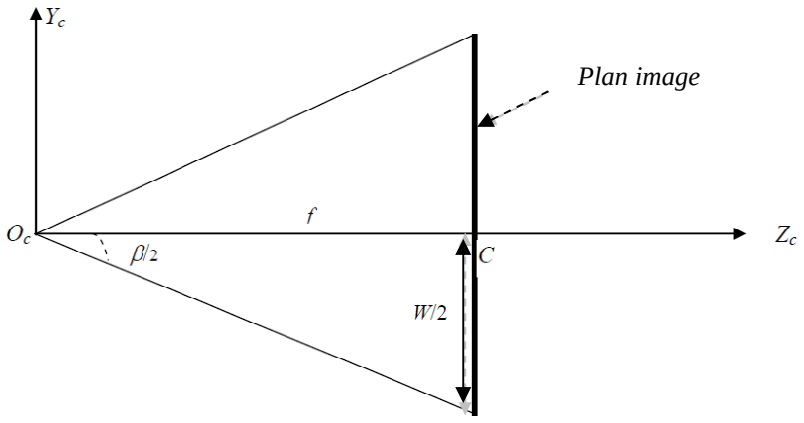


Figure 3.2. *Angle de vue de la caméra.*

3.2.1. Projection d'un point 3D sur l'image

La projection perspective d'un point 3D $P = (x_w, y_w, z_w)$ sur l'image, se fait en lui associant un point 2D $p = (x , y)$ de l'image. Elle se fait selon deux transferts en tenant compte d'abord des paramètres externes et puis des paramètres internes de la caméra. Ainsi une matrice de transformation **M** appelée matrice de pose de la caméra et déterminée par les paramètres externes de la caméra (position de la caméra et son orientation par rapport au repère du monde), fait le transfert du repère du monde vers le repère de la caméra. La position de la caméra est modélisée par le vecteur translation **t** du centre optique de la caméra par rapport au repère du monde et l'orientation de la caméra est modélisée par sa matrice de rotation **R** par rapport au repère du monde. Dans ce premier transfert, le point *P* du monde est équivalent au point P_c dans le repère de la caméra. Le deuxième transfert se fait du repère caméra vers celui de l'image, en tenant compte des paramètres internes de la caméra qui sont la distance focale *f* et les coordonnées du point principal *C*. Ce transfert est effectué par une matrice **K** appelée matrice de calibrage de la caméra. La figure 3.3 illustre ces deux transferts.

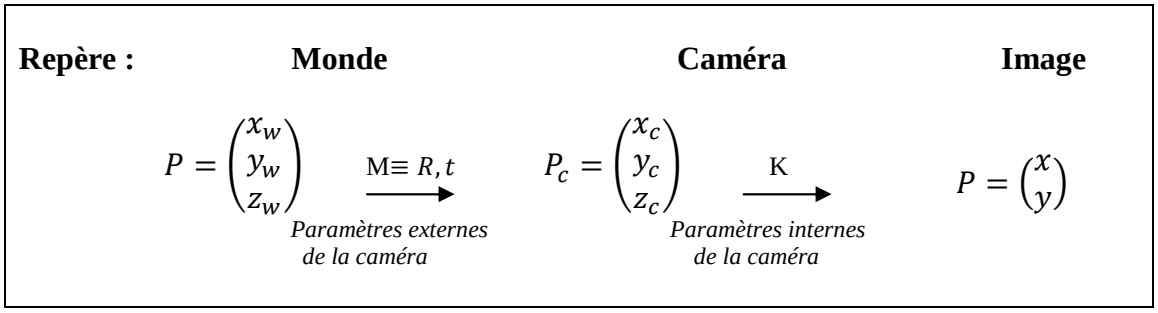


Figure 3.3. *Projection perspective d'un point du monde par changement de repère.*

3.2.1.1. Calcul de la matrice M

La caméra subit par rapport au repère du monde, une translation **t** = (t_x , t_y , t_z) et une rotation $R = \begin{pmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{pmatrix}$ résultant de la combinaison des 3 rotations selon les 3 axes du repère du monde, définies par les angles d'Euler:

$$R_x(\gamma_1) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\gamma_1) & -\sin(\gamma_1) \\ 0 & \sin(\gamma_1) & \cos(\gamma_1) \end{pmatrix} \qquad R_y(\gamma_2) = \begin{pmatrix} \cos(\gamma_2) & 0 & \sin(\gamma_2) \\ 0 & 1 & 0 \\ -\sin(\gamma_2) & 0 & \cos(\gamma_2) \end{pmatrix}$$

$$R_z(\gamma_3) = \begin{pmatrix} \cos(\gamma_3) & -\sin(\gamma_3) & 0 \\ \sin(\gamma_3) & \cos(\gamma_3) & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad (3.2)$$

Ainsi tout point *P* du monde se transfert dans le repère de la caméra en fonction de ces deux transformations **R** et **t** selon l'équation (3.2).

$$P = \begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix} \rightarrow \qquad P_c = \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = \begin{pmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{pmatrix} \begin{pmatrix} x_w - t_x \\ y_w - t_y \\ z_w - t_z \end{pmatrix} \tag{3.3}$$

Le développement du produit donne :

$$P = \begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix} \rightarrow \quad P_c = \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = \begin{pmatrix} r_1 x_w + r_2 y_w + r_3 z_w - r_1 t_x - r_2 t_y - r_3 t_z \\ r_4 x_w + r_5 y_w + r_6 z_w - r_4 t_x - r_5 t_y - r_6 t_z \\ r_7 x_w + r_8 y_w + r_9 z_w - r_7 t_x - r_8 t_y - r_9 t_z \end{pmatrix} \tag{3.4}$$

Dans l'espace projectif, s'ajoute à chaque point une coordonnée homogène égale à 1, d'où le changement de la notation des points:

$$\tilde{P} = \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \rightarrow \quad \widetilde{P_c} = \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} = \begin{pmatrix} r_1 x_w + r_2 y_w + r_3 z_w - r_1 t_x - r_2 t_y - r_3 t_z \\ r_4 x_w + r_5 y_w + r_6 z_w - r_4 t_x - r_5 t_y - r_6 t_z \\ r_7 x_w + r_8 y_w + r_9 z_w - r_7 t_x - r_8 t_y - r_9 t_z \\ 1 \end{pmatrix} \tag{3.5}$$

Le système (3.5) s'écrit :

$$\tilde{P} = \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \rightarrow \quad \widetilde{P_c} = \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} = \begin{pmatrix} r_1 & r_2 & r_3 & -r_1 t_x - r_2 t_y - r_3 t_z \\ r_4 & r_5 & r_6 & -r_4 t_x - r_5 t_y - r_6 t_z \\ r_7 & r_8 & r_9 & -r_7 t_x - r_8 t_y - r_9 t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \tag{3.6}$$

d'où

$$\tilde{P} = \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \rightarrow \quad \widetilde{P_c} = \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} = \begin{pmatrix} r_1 & r_2 & r_3 & 0 \\ r_4 & r_5 & r_6 & 0 \\ r_7 & r_8 & r_9 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & -t_x \\ 0 & 1 & 0 & -t_y \\ 0 & 0 & 1 & -t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \tag{3.7}$$

Le système (3.7) peut s'écrire :

$$\tilde{P} = \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \rightarrow \quad \widetilde{P_c} = \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} = \begin{bmatrix} \boldsymbol{R} & 0 \\ \boldsymbol{0}^T & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & -\boldsymbol{t} \\ \boldsymbol{0}^T & 1 \end{bmatrix} \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \tag{3.8}$$

avec ***I*** est la matrice identité (3x3) et 0 le vecteur nul,

d'où,

$$\tilde{P} = \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} \rightarrow \quad \widetilde{P_c} = \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} = \mathbf{M}\,\widetilde{P} \tag{3.9}$$

avec

$$\mathbf{M} = \begin{bmatrix} \boldsymbol{R} & 0 \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & -\boldsymbol{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$$

3.2.1.2. Calcul de la matrice K

La figure (3.4) montre la projection du point P_c sur le plan image, le calcul de la matrice **K** peut être effectué géométriquement :

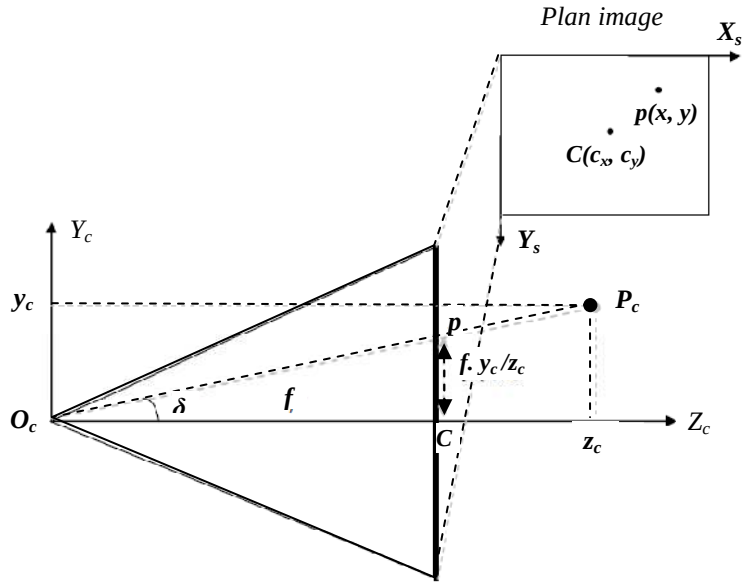


Figure 3.4. *Projection d'un point P sur le plan image.*

Le calcul de la tangente de l'angle δ (figure 3.4) permet d'écrire :

$$\frac{y - c_y}{f} = \frac{y_c}{z_c} \tag{3.10}$$

d'où

$$y = f.\frac{y_c}{z_c} + c_y \tag{3.11}$$

Le même raisonnement avec l'axe X_c et l'axe X_s permet de déduire que :

$$x = f.\frac{x_c}{z_c} + c_x \tag{3.12}$$

De même que pour le calcul de **M**, on écrit les points P_c et p dans l'espace projectif, ainsi le transfert dans l'espace euclidien, de la caméra vers l'image :

$$P_c = (x_c, y_c, z_c) \rightarrow \ \ p = (x, y) = (\ f.\frac{x_c}{z_c} + c_x, \ \ f.\frac{y_c}{z_c} + c_y) \tag{3.13}$$

devient dans l'espace projectif:

$$\tilde{P}_c = (x_c, y_c, z_c, 1) \rightarrow \ \tilde{p} = (x, y, 1) = \left(f.\frac{x_c}{z_c} + c_x, \ \ f.\frac{y_c}{z_c} + c_y, 1\right) \tag{3.14}$$

d'où,

$$\tilde{P}_c = (x_c, y_c, z_c, 1) \rightarrow \ \tilde{p} = (x, y, 1) = (f x_c + z_c c_x, f y_c + z_c c_y, z_c) \tag{3.15}$$

On peut écrire (3.14) sous forme matricielle :

$$\tilde{P} = \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} \rightarrow \tilde{P_c} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & c_x & 0 \\ 0 & f & c_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix} \tag{3.16}$$

d'où

$$\tilde{p} = \mathbf{K}[\mathbf{I} \ | \mathbf{0}] \ \tilde{P_c} \tag{3.17}$$

avec

$\mathbf{K} = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix}$, **I** la matrice identité (3x3) et **0** vecteur nul.

K est la matrice de calibrage de la caméra. Elle est formée des paramètres internes de la caméra: la focale *f* et les coordonnées du point principal *C*.

L'équation (3.9) et (3.17) permettent d'écrire :

$$\tilde{p} = \mathbf{K}[\mathbf{I} \ | 0] \begin{bmatrix} \boldsymbol{R} & 0 \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & -\boldsymbol{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \tilde{P} \tag{3.18}$$

d'où

$$\tilde{p} = \mathbf{K}\,\mathbf{R}\,[\mathbf{I} \quad -\mathbf{t}]\ \tilde{P} \tag{3.19}$$

L'équation (3.19) exprime la projection d'un point du monde sur le plan image selon le modèle perspective et en utilisant les paramètres intrinsèques et extrinsèques de la caméra. Ces calculs nous seront utiles dans la génération des images de simulation.

3.3. Mise en correspondance de deux images

La mise en correspondance des points d'intérêt appartenant à deux images successives détermine pour chaque point d'intérêt p_1 sur $Image_1$ son homologue le point p_2 s'il existe sur $Image_2$, tels que p_1 et p_2 soient la projection d'un même point P de la scène 3D, sur $Image_1$ et $Image_2$ respectivement (Figure 3.5). Cette mise en correspondance se fait en se basant sur le calcul d'une distance ϕ telle que distance euclidienne, distance de Mahalannobis, distance algébrique,… entre les deux descripteurs d_1 et d_2 représentant respectivement les deux points d'intérêt p_1 et p_2.

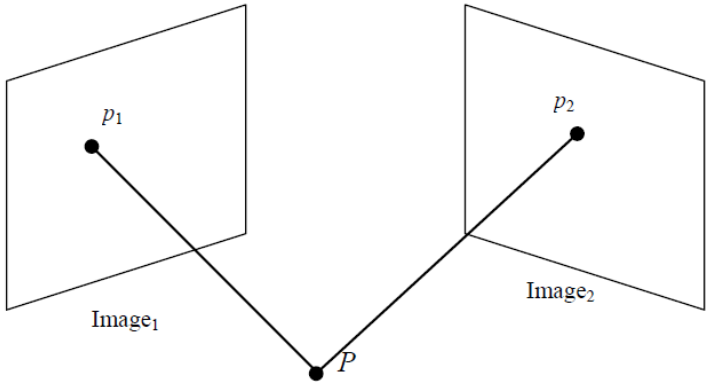


Figure 3.5. *Mise en correspondance de deux points images.*

La mise en correspondance de deux points images peut présenter des ambigüités. Afin de pallier à ce problème et éliminer les fausses cibles, certaines contraintes ont été formulées par Marr et Poggio [MAR79]:

- **Contrainte d'unicité**

Un point de Image$_1$ a au plus un homologue dans Image$_2$. Cette contrainte exclut la possibilité d'alignement des points 3D avec le centre optique de la caméra.

- **Contrainte de continuité**

Le monde physique étant généralement constitué de surfaces continues, cette contrainte permet à partir d'une mise en correspondance initiale de prédire d'autres mises en correspondance qui viennent confirmer la première. Skordas et Horaud [SKO89] ont énoncé les contraintes suivantes :

- **Contrainte de continuité de disparité :** La disparité métrique est définie par:

$$\lambda(P) = p_1C_1 - p_2C_2 \qquad (3.20)$$

C'est la différence des positions des points p_1 et p_2 par rapport aux points principaux respectivement sur Image$_1$ et Image$_2$. La contrainte de continuité de disparité s'appuie sur le fait que les surfaces physiques sont localement continues et que la variation de leur profondeur par rapport à l'observateur est généralement faible.

- **Contrainte de continuité des formes:** Une suite de points de contours connectés dans Image$_1$ devrait apparaître comme une séquence connectée de points de contours dans Image$_2$.

- **Contrainte de ressemblance**

Deux primitives sont mises en correspondance si leurs caractéristiques vérifient les conditions de ressemblance entre eux. Ces conditions peuvent être d'ordre géométrique ou photométrique.

- **Contrainte d'ordre**

Cette contrainte suppose que l'ordre dans lequel sont disposées des primitives appartenant à $Image_1$ est celui dans lequel leurs homologues sont disposés.

L'opération de mise en correspondance des points d'intérêt sur les deux images est conditionnée par les contraintes citées ci-dessus. Elle a pour résultat de fournir une liste de paires de points homologues appartenant respectivement aux deux images.

3.3.1. Stratégies de mise en correspondance

Plusieurs stratégies peuvent être utilisées dans la mise en correspondance de deux points d'intérêt :
Deux points d'intérêt sont classés correspondants si la distance ϕ (que l'on choisit) entre leurs descripteurs est inférieure à un seuil ξ.

$$\phi(d_1, d_2) < \xi \qquad (3.21)$$

Le choix du seuil influe sur la performance de la mise en correspondance. En plus, cette stratégie de mise en correspondance peut être ambiguë si le point p_1 a un plus proche voisin p_2 et un second plus proche voisin p_2'. Dans ce cas, une autre stratégie peut être utilisée. Elle se base sur le calcul du ratio des distances $\phi(d_1,d_2)$ et $\phi(d_1,d_2')$ associées respectivement aux points p_2 et p_2' . Si ce ratio est suffisamment grand, le plus proche voisin est accepté comme correspondant à p_1.

De telles stratégies donnent un ensemble préliminaire de correspondances qui peut contenir des correspondances aberrantes. Elles peuvent être améliorées par le calcul homographique si ces points appartiennent à un même plan. Le calcul homographique peut à son tour être affiné en utilisant la méthode RANSAC.

3.4. Calcul homographique

L'opération de mise en correspondance des points d'intérêt sur les deux images fournit une liste de paires de points homologues. Étant donné que la caméra est en rotation pure, on considère qu'il existe une transformation projective représentée par une matrice **H,** reliant les points mis en correspondance sur les deux images successives.

Rappelons que l'équation (3.19) $\tilde{p} = \mathbf{K}\,\mathbf{R}\,[\mathbf{I} \quad -\mathbf{t}]\,\tilde{P}$ permet d'exprimer la projection d'un point du monde $\tilde{P}$ sur un plan image d'une caméra de matrice de calibrage **K** et dont le repère est positionné, par rapport au centre du monde, avec une rotation **R** et une translation **t**.

Lorsque la caméra est en position initiale, la projection du point $\tilde{P}$ sur Image_1 s'écrit selon l'équation (3.19):

$$\tilde{p}_1 = \mathbf{K}\,\mathbf{R}_1\,[\mathbf{I} \qquad -\mathbf{t}_1]\,\tilde{P}$$

$\mathbf{R}_1$ est une matrice de rotation, elle est donc inversible.
K est une matrice inversible, car elle est triangulaire et tous les éléments diagonaux sont non nuls, d'où

$$\mathbf{R}_1^{-1}\,\mathbf{K}^{-1}\;\tilde{p}_1 = [\mathbf{I} \quad -\mathbf{t}_1]\tilde{P} \tag{3.22}$$

Lorsque cette même caméra fait une rotation pure par rapport à sa position initiale, c.à.d. sans translation par rapport à sa position initiale, la projection du point $\tilde{P}$ sur Image_2 s'écrit selon l'équation (3.19):

$$\tilde{p}_2 = \mathbf{K}\,\mathbf{R}_2\,[\mathbf{I} \quad -\mathbf{t}_1]\,\tilde{P}$$

d'après (3.22), on peut écrire :

$$\tilde{p}_2 = \mathbf{K}\,\mathbf{R}_2\,\mathbf{R}_1^{-1}\mathbf{K}^{-1}\tilde{p}_1 \tag{3.23}$$

d'où

$$\tilde{p}_2 = \mathbf{H}\,\tilde{p}_1 \qquad \text{avec} \quad \mathbf{H} = \mathbf{K}\,\mathbf{R}_2\,\mathbf{R}_1^{-1}\mathbf{K}^{-1} \tag{3.24}$$

Cette transformation projective (3.24) appelée homographie est une transformation linéaire non singulière pour des coordonnées homogènes. Elle est donc, définie dans un espace projectif. Celui-ci est l'espace euclidien augmenté des points à l'infini. Les points à l'infini appelés aussi points idéaux permettent d'envisager entre autres, la projection des lignes parallèles qui se coupent à l'infini dans une scène 3D.

Afin d'exploiter la linéarité de cette transformation projective, il est nécessaire de transformer les points images initialement définis dans l'espace euclidien en des points projectifs. Ainsi les différents points de coordonnées (*x*, *y*) de l'espace euclidien seront transformés dans l'espace projectif par l'ajout d'une troisième coordonnée homogène

La transformation d'un point de l'espace euclidien vers l'espace projectif se fait par :

$$p = \begin{pmatrix} x \\ y \end{pmatrix} \;\rightarrow\; \tilde{p} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \qquad \text{ou bien} \qquad \tilde{p} = \begin{pmatrix} kx \\ ky \\ k \end{pmatrix} \;\; avec \;\; k \neq 0 \tag{3.25}$$

La transformation d'un point de l'espace projectif vers l'espace euclidien se fait par :

$$\tilde{p} = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \rightarrow p = \begin{pmatrix} x/z \\ y/z \end{pmatrix} \qquad avec \quad z \neq 0 \tag{3.26}$$

Les ensembles des points p_1 et p_2 respectivement d'Image$_1$ et d'Image$_2$ forment dans l'espace projectif deux plans π_1 et π_2 contenant les points projectifs $\tilde{p}_1$et $\tilde{p}_2$ correspondants à p_1 et p_2.

Dans l'espace projectif, l'homographie reliant les deux plans π_1 et π_2 est représentée par une matrice $\mathbf{H}_{3x3}$ reliant deux points $\tilde{p}_1(x_1, y_1, 1) et\, \tilde{p}_2(x_2, y_2, 1)$ appariés appartenant respectivement à π_1 et π_2 :

$$\tilde{p}_2 = \mathbf{H}\,\tilde{p}_1 \tag{3.27}$$

d'où

$$\begin{pmatrix} x_2 \\ y_2 \\ 1 \end{pmatrix} = \begin{pmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ 1 \end{pmatrix} \tag{3.28}$$

Il est à noter que **H** peut varier en la multipliant par une constante arbitraire non nulle, sans que cela n'altère la transformation projective. **H** peut donc être considérée comme une matrice avec 8 degrés de liberté même si elle contient 9 éléments. Cela signifie que pour retrouver la matrice **H**, il suffit de déterminer les 8 inconnues au lieu de 9, si on prend $h_9 = 1$. Hartley [HAR04] proposent une méthode d'estimation de l'homographie **H.**

3.4.1. Estimation de l'homographie [DUB09]

3.4.1.1. L'algorithme DLT

L'algorithme DLT (Direct Linear Transform) permet d'estimer l'homographie **H**.

Le développement de (3.28) donne :

$$\begin{cases} h_1x_1 + h_2y_1 + h_3 = x_2 \\ h_4x_1 + h_5y_1 + h_6 = y_2 \\ h_7x_1 + h_8y_1 + h_9 = 1 \end{cases} \tag{3.29}$$

En divisant la ligne 1 par la ligne 3 et la ligne 2 par la ligne 3, dans (3.29), on obtient :

$$\begin{cases} h_1x_1 + h_2y_1 + h_3 - x_2(h_7x_1 + h_8y_1 + h_9) = 0 \\ h_4x_1 + h_5y_1 + h_6 - y_2(h_7x_1 + h_8y_1 + h_9) = 0 \end{cases} \tag{3.30}$$

Les équations (3.30) peuvent être écrites sous la forme :

$$\mathbf{A_j h} = 0 \tag{3.31}$$

avec j : j^{ème} paire de points mis en correspondance.

où

$$\mathbf{A_j} = \begin{pmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x_2 & -y_1x_2 & -x_2 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y_2 & -y_1y_2 & -y_2 \end{pmatrix} \quad (3.32)$$

et

$$\mathbf{h} = (h_1 \quad h_2 \quad h_3 \quad h_4 \quad h_5 \quad h_6 \quad h_7 \quad h_8 \quad h_9)^{\mathrm{T}} \quad (3.33)$$

Chaque paire de points mis en correspondance fournit 2 équations dans la matrice $\mathbf{A_j}$, donc, 4 paires de points sont suffisants pour résoudre le système à 8 degrés de liberté de **H**. En regroupant les 4 paires de lignes $\mathbf{A_j}$ (j=1,..,4), on obtient le système :

$$\mathbf{A\,h} = 0 \quad (3.34)$$

où **A** est une matrice 8x9.

Un nombre de points supérieur à 4 permet de donner plus de robustesse à la solution. Dans ce cas, la solution du système (3.34) est surdéterminée, sa résolution revient à trouver **h** qui minimise une fonction de coût à définir. On peut calculer la solution des moindres carrés $\|\mathbf{Ah}\|$, qui consiste à retrouver le vecteur singulier ayant la plus petite valeur singulière. Cela peut se faire par la méthode SVD (Singular Value Decomposition) [DUB09] qui minimise la distance algébrique d_{alg} (3.35). Celle-ci est la norme du produit vectoriel entre un point et sa reprojection par **H**:

$$\mathrm{d_{alg}}(\tilde{p}_1, \tilde{p}_2, \mathbf{H}) = \|\tilde{p}_2 \times \mathbf{H}\tilde{p}_1\| \quad (3.35)$$

Afin d'obtenir un vecteur **h** non nul, on minimise la distance algébrique avec la contrainte $\|\mathbf{h}\|$=1.

La factorisation SVD permet d'écrire la matrice rectangulaire **A** sous forme de produit de trois matrices: une matrice orthogonale **Q** dont les colonnes sont formées des vecteurs propres de $\mathbf{AA^T}$, une matrice **L** contenant des racines carrées des valeurs propres de $\mathbf{AA^T}$ disposées par ordre décroissant de leurs valeurs et une matrice transposée d'une matrice orthogonale **U**. Les colonnes de **U** sont formées des vecteurs propres de $\mathbf{A^TA}$. La matrice **A** est décomposée en :

$$\mathbf{A}_{mn} = \mathbf{Q}_{mm}\mathbf{L}_{mn}\mathbf{U}^T_{nn} \quad (3.36)$$

Le vecteur **h** qu'on cherche à déterminer, correspond au dernier vecteur propre de la matrice **U** qui lui-même correspond à la plus petite valeur propre de $\mathbf{AA^T}$.

3.4.1.2. Normalisation des pixels

La solution du système à huit équations (3.34) est lié à l'origine et à l'échelle des coordonnées des points, car les pixels dans l'espace projectif ont des coordonnées *x* et *y* beaucoup plus grandes que 1 qui est la troisième coordonnée homogène. Cela engendre de

l'instabilité numérique. Afin de pallier à cet inconvénient, Hartley [HAR04] présente l'algorithme DLT avec normalisation des pixels (Algorithme 3.1).

Entrée: liste des paires de points mis en correspondance.

Sortie: H homographie entre les points mis en correspondance.

Début

1. Déterminer pour chaque image *i* (*i*=1, 2), une transformation similarité (translation et changement d'échelle) $\mathbf{T}_{i(3x3)}$ qui permet de centrer les points d'intérêt à l'origine (leur centroïde devient l'origine) à une distance moyenne de l'origine égale à $\sqrt{2}$.

2. Calculer par SVD une transformation **H'** à partir des paires de points normalisés obtenus en 1.

3. L'homographie **H** est calculée par : $\mathbf{H} = \mathbf{T}_2^{-1}\mathbf{H}'\mathbf{T}_1$

Fin.

Algorithme 3.1. *Calcul de H par DLT avec normalisation des pixels.*

L'estimation de l'homographie par la méthode SVD minimise l'erreur algébrique et la normalisation des pixels permet d'améliorer la solution obtenue par SVD (Algorithme 3.1). Cependant cette solution linéaire n'est pas optimale.

3.4.1.3. Limites de l'estimation aux moindres carrés

L'évaluation de quantités à partir de mesures polluées par des données aberrantes, ou outliers, n'est pas aisée au moyen d'algorithmes tels que les moindres carrés: ce type de méthode est y en effet très sensible. Ceux-ci perturbent fortement la solution trouvée.

Un exemple frappant de l'absence de robustesse de l'algorithme des moindres carrés est illustré par la figure 3.6: la minimisation étant effectuée suivant l'erreur quadratique, les points aberrants prennent le pas sur les points corrects du fait de leur poids important. Six des sept points constituent des données valides, cependant l'algorithme des moindres carrés prend en compte le point aberrant et renvoie une solution attirée par celui-ci.

Ce problème est fréquent en vision par ordinateur, où les données obtenues en analysant les images ne sont pas fiables au sens qu'elles contiennent souvent une proportion importante d'outliers. Afin de résoudre ces difficultés, les estimateurs robustes ont donc été introduits par les statisticiens.

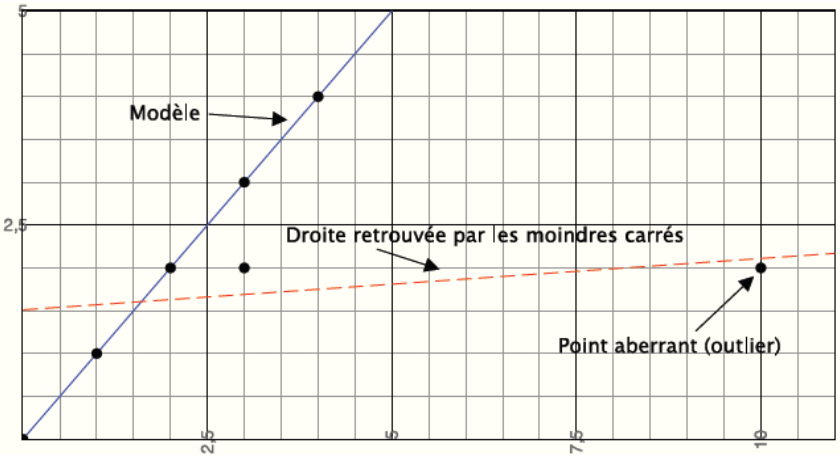


Figure 3.6. *Résolution aux moindres carrés avec des données aberrantes: recherche d'un modèle affine f minimisant* $\sum_{i=1}^{n}|y_i - f(x_i)|^2$ *où les* (x_i, y_i) *sont les données; la solution trouvée est beaucoup plus attirée par le point éloigné que par les quatre points suivant le modèle.*

Les approches populaires en vision sont les M-estimateurs [HUB81], ainsi que les moindres carrés médians (*Least Median of Squares* – LMS). Les M-estimateurs sont des fonctions de résidus, continues et symétriques avec un minimum en 0, utilisées afin d'écrêter les résidus dont l'influence croît linéairement pour une estimation aux moindres carrés classique. Ils sont utilisés pour minimiser de façon robuste la somme d'une fonction de résidus. Les moindres carrés médians, ou LMS, consistent eux à minimiser la médiane des résidus au carré, ce qui permet d'éliminer la deuxième moitié des résidus triés par ordre croissant.

Les performances de ces estimateurs sont mesurées via deux quantités: *le point de rupture*, le plus petit pourcentage de données aberrantes pouvant faire échouer l'estimation, et *l'efficacité relative* qui traduit la précision de l'estimateur. En pratique, le point de rupture est de l'ordre de 20% pour les M-estimateurs et de 50% pour les LMS. La fraction de données aberrantes doit rester inférieure au point de rupture, et 50% n'est souvent pas suffisant en vision par ordinateur.

Une solution existe pour obtenir un point de rupture plus élevé: les approches dérivées de RANSAC, cherchent à extraire les données correctes, dites *inliers*, de celles aberrantes, *outliers*, à partir d'un ensemble de données fournies. Contrairement aux approches précédentes, une fois classifiées comme outliers, les données aberrantes n'interviennent plus dans le résidu de l'estimation. On obtient alors un point de rupture beaucoup plus élevé (entre 70 et 90% suivant les variantes).

La transformée de Hough est une autre solution. Elle a l'avantage d'être robuste, et de permettre la détection de multiples occurrences de structures suivant le même modèle. Cependant, la discrétisation de l'espace des paramètres qu'elle implique rend son application difficile à de nombreux problèmes: paramétrer les cellules de l'espace de vote pour qu'elles soient équiprobables est complexe. La procédure de vote employée devient de plus très coûteuse quand le nombre de paramètres augmente. Enfin, reconnaître les points

d'accumulation afin d'établir les valeurs des paramètres du modèle est aussi un problème en soi.

3.4.2. Optimisation de l'estimation de l'homographie par RANSAC

RANSAC est une méthode itérative robuste qu'on peut utiliser dans l'estimation de l'homographie. Parmi l'ensemble S des paires de points mis en correspondance, on sélectionne aléatoirement un ensemble $\hat{S}$ de 4 correspondances (4 paires de points) et on calcule l'homographie **H** les reliant selon l'algorithme 3.1 qui se base sur la distance algébrique. Les autres correspondances seront soumises à cette homographie et on calcule pour chaque correspondance la distance euclidienne entre un point $\tilde{p}_2$ et sa reprojection par **H:**

$$\mathrm{d}_{\mathrm{alg}}(\tilde{p}_1, \tilde{p}_2, \mathbf{H}) = \|\tilde{p}_2 - \mathbf{H}\tilde{p}_1\| \tag{3.37}$$

Une correspondance est acceptée, donc classée comme vraie, si le carré de la distance euclidienne entre les deux points appariés est inférieure à un seuil τ:

$$\mathrm{d}_{\mathrm{eucl}}(\tilde{p}_1, \tilde{p}_2, \mathbf{H}) < \tau \tag{3.38}$$

Chaque correspondance acceptée est stockée dans l'ensemble $\widehat{S}_n$. Le calcul est réitéré N fois et à chaque itération on retient l'ensemble $\hat{S}_n$ ayant donné le plus grand nombre de vraies correspondances. Il devient l'ensemble référence S^*, dans le sens où il contient le plus grand nombre de correspondances acceptées, pour la prochaine itération. À la fin, on recalcule **H** pour une dernière fois selon l'algorithme 3.1 sur l'ensemble des correspondances retenues S^*.

3.4.2.1. Calcul du nombre d'itérations N

Le nombre d'itérations N est déterminé en se basant sur un calcul de probabilité:

Soit ε la proportion des fausses correspondances (les paires de points faussement appariés).

- La probabilité qu'une correspondance soit vraie est ω=1- ε.
- La probabilité de sélectionner s vraies correspondances est ω^s.
- La probabilité d'avoir au moins une fausse correspondance parmi s correspondances est 1- ω^s.
- La probabilité d'avoir au moins une fausse correspondance dans N choix de sous ensembles est $(1- \omega^s)^N$.
- La probabilité d'avoir au moins un choix de s correspondances sans fausses correspondances après N essais est: $\nu = 1 - (1- \omega^s)^N$.

Le nombre d'essais nécessaires pour obtenir une probabilité ν d'avoir fait au moins un choix de *s* éléments sans fausse correspondance est:

$$N = \frac{\ln(1-\nu)}{\ln(1-(1-\varepsilon)^s)} \tag{3.39}$$

La valeur de la probabilité est choisie ν = 0.99, afin de maximiser les chances des vraies correspondances. La valeur de *s* est 4, c'est le minimum de correspondances nécessaires pour le calcul de l'homographie. La valeur du seuil τ est choisie par expérimentation.

L'estimation de l'homographie par RANSAC est réalisée par l'algorithme 3.2 :

Entrée: $S = \{(\tilde{p}_{1j},\ \tilde{p}_{2j}) |\ j = 1, \dots, |S|\}$, ensemble des points appariés par l'algorithme SIFT.

Sortie : *S** ensemble des paires de points formant des correspondances acceptées.

H : l'homographie estimée.

Paramètres intermédiaires : *N* (nombre maximal d'itérations), $\hat{S}$ (ensemble de *s* correspondances choisies aléatoires), $\hat{S}_n$ (ensemble des correspondances acceptées à l'itération *n*), ε (proportion des fausses correspondances)

Début :

N←∞ , *n* = 1 , *S** ← Ø *// Initialisation*

Tant que (*n* < *N*) **faire**

Choisir aléatoirement $\hat{S} \subset S$ tel que $|\hat{S}| = s$

Estimer **H** sur $\hat{S}$ (selon l'algorithme 5.1)

$\hat{S}_n \leftarrow \{(\tilde{p}_{1j},\ \tilde{p}_{2j}) |\ (\tilde{p}_{1j},\ \tilde{p}_{2j}) \in S$ et $d_{eucl}(\tilde{p}_{1j},\ \tilde{p}_{2j}, \mathbf{H}) < \tau\}$

Si $|\hat{S}_n| > |S^*|$ **alors**

$S^* \leftarrow \hat{S}_n$

$\varepsilon \leftarrow 1 - |\hat{S}_n|/|S|$

$N = \frac{\ln(1-\nu)}{\ln(1-(1-\varepsilon)^s)}$

Fin Si

n←*n*+1

fin Tant que

Estimer **H** sur *S**

Fin

Algorithme 3.2. *Estimation de l'homographie par RANSAC adaptatif.*

Le résultat de l'algorithme 3.2 retient les points les plus stables au sens de l'homographie et la convergence semble être assurée empiriquement.

3.4.3 Évolutions de RANSAC

Afin d'améliorer RANSAC, et de résoudre les difficultés de choix des deux seuils précédemment décrits, les perfectionnements de RANSAC s'articulent autours de plusieurs axes :

- utiliser de meilleures fonctions d'évaluation du consensus: comment déterminer le seuil sur la distance au modèle ou la taille minimale d'un ensemble pour considérer les paramètres du modèle comme corrects vis à vis des données ? Cela peut être réalisé en modélisant la distribution des outliers (MINPRAN [STE95]), en estimant le nombre de fausses alarmes avec les méthodes *A Contrario* : AC-RANSAC [MOI04], MAC-RANSAC [RAB10]), ou encore en distinguant les inliers selon leur qualité (en privilégiant les inliers les plus précis par rapport au résidu géométrique MLESAC [TOR00]).

- obtenir une solution plus rapidement avec des techniques d'échantillonnage différentes:
    - uniforme, ou biaisé suivant une loi aléatoire fixée en utilisant une connaissance a priori [CHU05] (issue de la phase d'extraction) sur la qualité des données pour guider la recherche. Il est aussi possible de biaiser l'échantillonnage en ajoutant des informations de contexte: par exemple, en groupant les points d'intérêt à l'aide d'une segmentation, ou d'une mesure du flot optique [NI09].

On note que dans toutes ces méthodes, les ensembles d'appariements candidats sont toujours fournis par l'étape de mise en correspondance, puis filtrés. L'ajout de la contrainte géométrique pour la sélection des points ne permet pas de récupérer ceux qui sont rejetés par erreur lors de la mise en correspondance à l'aide de la photométrie, ce qui est fatal pour une scène contenant des motifs répétés.

Nous allons maintenant présenter les extensions de RANSAC utilisées dans notre thèse.

3.4.4. Modifier la technique d'échantillonnage par l'algorithme de PROSAC

Chum et Matas proposent dans [CHU05] une modification du paradigme RANSAC. Plutôt que de traiter tous les appariements de la même manière, ils proposent dans leur approche *Progressive Sample Consensus* (PROSAC) d'exploiter les informations sur la ressemblance entre points mis en correspondance, suivant ainsi l'idée proposée par Tordoff et Murray dans [TOR05]. Ils sélectionnent ainsi les échantillons sur un ensemble de correspondances les mieux appariées, ensemble qu'ils élargissent petit à petit. Leur approche est intéressante au sens qu'elle combine une recherche qu'on espère plus rapide du consensus,

avec une fonction d'évaluation du consensus prenant en compte la probabilité de fausses alarmes.

Le choix de cet ensemble est un compromis entre l'hypothèse trop optimiste d'une bonne qualité du tri effectué par la phase d'appariements, et la trop pessimiste approche RANSAC qui traite toutes les correspondances de la même façon. Au lieu d'utiliser la probabilité qu'une correspondance est correcte, ils utilisent l'hypothèse a minima d'un lien entre probabilité de ressemblance et la distance entre descripteurs. De telles séquences de correspondances sont appelées *not-worse-than-random* (pas pires qu'aléatoires).

3.4.4.1. Stratégie d'échantillonnage:

Supposons que T_n représente le nombre moyen d'échantillons provenant de $\{M_i\}_{i=1}^{T_N}$ qui contient des points candidats de U_N seulement

$$T_n = T_N \frac{\binom{n}{m}}{\binom{N}{m}} = T_N \prod_{i=0}^{m-1} \frac{n-i}{N-i} \qquad (3.40)$$

puis,

$$\frac{T_{n+1}}{T_n} = \frac{T_N}{T_N} \prod_{i=0}^{m-1} \frac{n+1-i}{N-i} \prod_{i=0}^{m-1} \frac{N-i}{n-i} = \frac{n+1}{n+1-m} \qquad (3.41)$$

Enfin, la relation récurrente pour T_{n+1} est

$$T_{n+1} = \frac{n+1}{n+1-m} T_n \qquad (3.42)$$

Il y a T_n échantillons contenant des points candidats seulement de U_n. et T_{n+1} échantillons contenant des points candidats seulement de U_{n+1}. Comme $U_{n+1} = U_n \cup \{u_{n+1}\}$, il y a $T_{n+1} - T_n$ échantillons qui contiennent un point candidat u_{n+1} et *m-1* points candidats tirés à partir de U_n. Par conséquent, la procédure pour laquelle $n = m \ldots N$ forme $T_{n+1} - T_n$ échantillons composés de u_{n+1} et *m-1* points candidats tirés de U_n.
Comme les valeurs de T_n ne sont pas généralement des entiers, nous définissons $T'_m = 1$ et

$$T'_{n+1} = T'_n + \lceil T_{n+1} - T_n \rceil \qquad (3.43)$$

La fonction d'évolution est alors définie comme suit:

$$g(t) = \min\{n : T'_n \geq t\} \qquad (3.44)$$

Critères d'arrêt:

Deux conditions à satisfaire sont proposées pour l'arrêt de la recherche :

- *pas pire qu'aléatoire (non-randomness)*: éviter de choisir une solution qui est cohérente "par hasard". On cherche à minimiser la probabilité $P_n(i)$ qu'un groupe de correspondances soit considéré comme correct alors qu'il ne l'est pas :

$$p_n(i) = \beta^{i-m}(1-\beta)^{n-i+m}\binom{n-m}{i-m} \tag{3.45}$$

où *m* est la taille de l'échantillon minimal, *n* le nombre de correspondances à trier, *i* la taille de l'ensemble examiné et β est la probabilité qu'un modèle calculé à partir d'un échantillon contenant un outlier soit considéré comme correct alors qu'il ne l'est pas. Cette probabilité est supposée connue a priori.

Pour chaque n, le nombre minimal d'inliers I_n^{min} est calculé de tel sorte que la probabilité de taille de chaque support (aléatoire) soit plus petite que Ψ.

$$I_n^{min} = \min\left\{ j: \sum_{i=j}^{n} P_n^R(i) < \Psi \right\}. \tag{3.46}$$

On en déduit un critère sur le nombre minimal d'inliers I_{n^*} que doit comporter la solution :

$$I_{n^*} \geq I_{n^*}^{min} \tag{3.47}$$

où Ψ est un seuil (habituellement 5%). Ce critère est à rapprocher de l'évaluation du nombre de fausses alarmes par Moisan et Stival [MOI04]. Cependant, plutôt que d'évaluer ce nombre de façon purement combinatoire, il est ici nécessaire de fixer β a priori : cela revient à ajouter un paramètre à la méthode.

- *maximalité*: imposer que la probabilité qu'il existe un ensemble contenant plus d'inliers que celui trouvé après *k* tirages est inférieure à η_0 (autre seuil habituellement fixé à 5%). Il s'agit ici de l'estimation habituelle du nombre de tirage à effectuer lorsque l'on suppose connaître le taux avec RANSAC.

Pour un ensemble U_n de génération d'hypothèse, la probabilité P_{I_n} qu'un échantillon non trié de taille *m* soit aléatoirement sélectionné à partir d'un ensemble U_n de n correspondances est donné par:

$$P_{I_n} = \frac{\binom{I_n}{m}}{\binom{n}{m}} = \prod_{j=0}^{m-1} \frac{I_n - j}{n-j} \approx \epsilon_n^m \tag{3.48}$$

Où I_n est le nombre d'inliers dans U_n et $\epsilon_n = I_n/n$ est la fraction d'inliers. La probabilité η de manquer un ensemble d'inliers de taille I_n dans l'ensemble U_n après *k* échantillons de PROSAC, où $g(k) \leq n$ est:

$$\eta = (1 - P_{I_n})^k \tag{3.49}$$

Le nombre d'échantillons qui doivent être établis pour assurer la probabilité η soit inférieure au seuil prédéfini η_0 est:

$$k_{n^*}(\eta_0) \geq \log(\eta_0) / \log(1 - P_{I_{n^*}}) \tag{3.50}$$

La longueur d'arrêt n^* est choisi pour minimiser $k_{n^*}(\eta_0)$ soumis à $I_{n^*} \geq I_{n^*}^{min}$.

En ce qui concerne l'algorithme, les échantillons sont choisis dans un sous-ensemble correspondant aux données de meilleures qualités. La taille de cet ensemble est ensuite augmentée peu à peu. Si *m* est la taille de l'échantillon, *m*-1 appariements sont situés dans le sous ensemble et le dernier appariement nécessaire à la construction de l'appariement est ajouté au sous-ensemble à l'étape suivante.

L'estimation de l'homographie par PROSAC est réalisée par l'algorithme 3.3 :

Entrée: $S = \{(\tilde{p}_{1j},\ \tilde{p}_{2j}) | \ j = 1, \dots, |S|\}$, ensemble des points appariés par l'algorithme SIFT.
N: nombre total d'appariements initial.
m: taille initiale des échantillons.

Sortie: Ensemble des paires de points formant des correspondances acceptées.
H : l'homographie estimée.
NULL : échec de la mise en correspondance.

Début :
Si $N<m$ **alors** return *NULL*
Sinon
$t \leftarrow 0$, $n \leftarrow$ m, n* ← N *// Initialisation*
Répéter
// 1. Choix de l'ensemble de génération d'hypothèse:
t ← t+1
Si (t > T'_n) **et** (n < n*) **alors** n ← n+1 (voir eq. 3.43)
// 2. Echantillon semi-aléatoire M_t de taille m:
Si ($T'_n < t$) **alors**
L'échantillon contient m−1 points sélectionnés aléatoirement à partir de U_{n-1} et u_n.
Sinon
Sélectionner aléatoirement m points à partir de U_{n-1}.
// 3. Estimation des paramètres du modèle:
Calculer les paramètres du modèle p_t à partir de l'échantillon M_t.
// 4. Vérification du modèle:
Trouver le support (les meilleurs points candidats) du modèle avec les paramètres p_t
jusqu'à (la solution trouvée satisfait l'équation 3.47 ou 3.50**)**
Fin

Algorithme 3.3. *Estimation de l'homographie par l'algorithme de PROSAC.*

3.5. Conclusion

Dans ce chapitre, nous avons présenté un ensemble de méthodes de mise en correspondance sous contraintes géométriques afin de trouver l'ensemble de consensus minimisant le nombre de fausses alarmes et de valider les meilleurs appariements déjà détectés par l'étape précédente car la mise en correspondance utilisant uniquement l'information photométrique ne suffit pour produire des appariements corrects. Ainsi, aucun algorithme de sélection robuste ne réussira tout seul à extraire tous les appariements corrects.

Nous avons pu constater que lorsque on utilise l'algorithme RANSAC, il est nécessaire de régler plusieurs paramètres à la main (la distance maximale pour déclarer une mesure compatible avec le modèle évalué, le taux d'outliers permettant de fixer le nombre d'itérations, et la taille minimale du meilleur ensemble de consensus pour considérer la recherche réussie), ce qui peut se révéler être une entreprise délicate.

La contribution apportée par la méthode PROSAC est un gain significatif pour le nombre d'appariements obtenus, en particulier en présence de motifs répétés (l'ambiguïté perceptuelle) et pour une ligne de base grandissante. Ces points supplémentaires sont obtenus en allant chercher des appariements plus difficile à obtenir, ou parmi ceux plus proche voisin qui sont rejetés par la méthode RANSAC ou celle du seuil sur le rapport de distances.
De plus, cette méthode ne fait pas intervenir de paramètres difficiles à régler. L'équilibrage des contraintes photométriques et l'influence des différentes distances photométriques ont ainsi été étudiés dans le chapitre précédent.

Dans le chapitre suivant, nous présentons les détails de l'implémentation, les résultats de l'expérimentation ainsi que leur interprétation.

CHAPITRE 4

IMPLEMENTATIONS ET RESULTATS EXPERIMENTAUX

4.1. Motivations et objectifs

L'objectif de notre travail s'inscrit dans le cadre de la construction panoramique des images de télédétection (Images Optiques, Radars, Aériennes et Cartographiques). Ce travail permet notamment de mettre à jour les systèmes d'informations géographiques(SIG) en passant par l'étape de recalage (géo-référencement des cartes géographiques, recalage des images satellitaires/cartes). Le problème majeur de la création des panoramas d'images est de déterminer des points de ressemblances entre différentes images de télédétection prises par des capteurs différents (Multimodal), à partir des points de vue différents (multiview) et à des instants différents (Multitemporal) comme on peut avoir des cas où on doit recaler des images réelles avec des modèles synthétiques (proche du multimodal). La méthodologie choisie consiste à séparer le problème en deux sous problèmes (1. Recalage, 2. Mosaïquage « Stitching-Blending »).

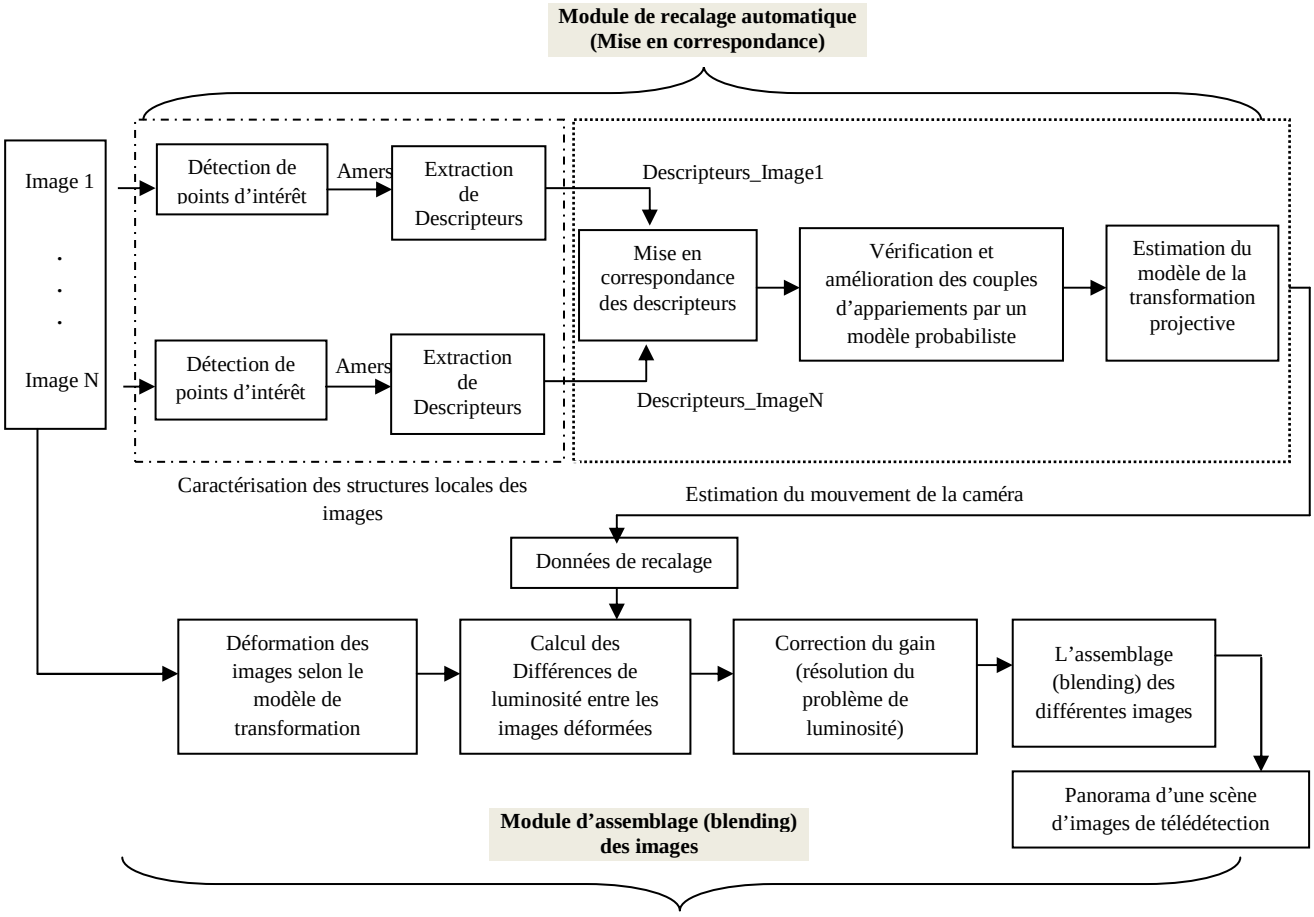


Figure 4.1. *Schéma fonctionnel du système proposé pour la construction panoramique des images de télédétection*

Pour le 1er sous-problème, plusieurs méthodes de recalage local sont proposées dans cette thèse et qui reposent sur les mêmes étapes suivantes:

1. Détection des points intérêts : l'idée est de déterminer dans chaque image des points dits invariants et ce de manière à ce que les points créés ne dépendent que de l'image à laquelle ils appartiennent. Ceci permet d'affranchir de tous les changements de prise de vue d'une image à l'autre (échelle, luminosité, angle,…).
2. Mise en correspondance des points d'intérêts.
3. Création d'un modèle géométrique de transformation.
4. Transformation de l'image.

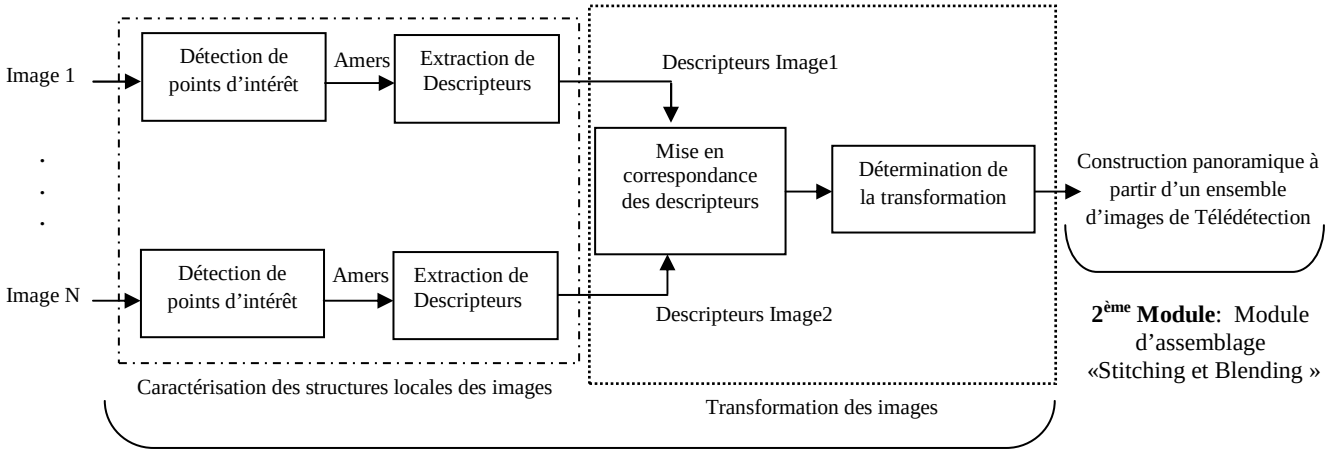


Figure 4.2. *Schéma Synoptique du 1er module de travail « Module de Recalage »*

Comme 1ère étape de construction de panorama, il faut choisir un algorithme de sélection des points d'intérêts qui soit robuste et qui assure un taux d'erreurs d'appariement le plus faible possible. Notre choix s'est donc rapidement porté aux SIFT et ces variantes car leurs caractéristiques font de ces derniers les algorithmes les plus robustes à de multiples déformations d'images. Par exemple, à titre de comparaison, nous avons pu constater que la majorité des détecteurs utilisés étaient incapable de pallier les problèmes rencontrés sur les images de télédétection (Méthodes faibles aux changements d'échelle, aux changements affines et aux changements de luminance ou de contraste). Mais, par contre, en étudiants les outils SIFT qu'ils été utilisés pour d'autres objectifs (Reconnaissance d'objets), nous avons extrait plusieurs avantages dont ils peuvent se résumer en :

- Robuste aux transformations du plan (Rotation & Translation), aux changements d'échelle, aux changements affines, aux changements de luminance ou de contraste.
- Les primitives sont locales, si robustes au chevauchement et à l'encombrement (aucune segmentation antérieure).
- Différents primitives peuvent être appariées à une grande base de données d'objets.
- Beaucoup de primitives peuvent être générées même pour de petits objets.
- Efficacité près des performances de l'exécution en temps réel.

- Extensibilité **:** peuvent être facilement étendus à de large portée de types différents de primitives, avec chaque robustesse ajoutée.

Une fois les primitives extraites à partir des *N* images, nous devons trouver les k plus proches voisins pour chaque primitive c.à.d. rechercher les descripteurs photométriques qui décrivent la même scène sur deux images différentes. Notre première contribution a été de modifier et d'optimiser l'algorithme Best Bin First (BBF) de telle façon qu'il nous permet d'identifier les voisins les plus proches avec une forte probabilité et un temps de calcul réduit en introduisant les ondelettes de Haar. Et pour trouver les primitives conformes appariées géométriquement, nous avons remplacé l'algorithme RANSAC par celui de PROSAC pour résoudre l'homographie entre les paires d'images et ceci constitue notre deuxième contribution dans cette thèse.

4.2. Autres méthodes d'extraction de points d'intérêts

La première étape du travail était de comprendre et d'analyser les différentes méthodes de détection automatique de points d'intérêts de plusieurs détecteurs (Harris, Harris-Stephen, Harris-Laplace, Sift…) afin de choisir et de valider le meilleur d'entre eux. Pour cela, nous avons réalisé un module contenant plusieurs descripteurs permettant de bénéficier des performances de chacun (Robustesse en : précision, localisation, temps de calcul,…).

4.2.1. Algorithme basé sur le détecteur « Harris-Laplace »

Le premier algorithme qui a été choisit est une version améliorée du détecteur de Harris-Stephen [MIK04]. Il a été montré que celui-ci produisait de bons résultats en termes de robustesse et de discrimination [SCH00]. Une version adaptée aux changements d'échelles permet d'obtenir de meilleurs résultats qu'avec l'algorithme SIFT en termes de stabilité géométrique pour des changements d'échelle.

4.2.1.1. Localisation de points caractéristiques

La détermination des points caractéristiques d'une image I est obtenue en utilisant la formule de Harris [HAR88]:

$$E(x,y) = [x,y].M.[x,y]^T \quad \text{Avec } M = \begin{bmatrix} \left(\frac{\partial L}{\partial x}\right)^2 & \left(\frac{\partial L \partial L}{\partial x \partial y}\right) \\ \left(\frac{\partial L \partial L}{\partial x \partial y}\right) & \left(\frac{\partial L}{\partial y}\right)^2 \end{bmatrix} \tag{4.1}$$

Où $L=I*g_{\sigma_d}$ est l'image lissée avec une gaussienne d'écart type σ_D.

g est une gaussienne telle que:

$$g_\sigma(x,y) = \frac{1}{2\pi\sigma^2} \exp\left(\frac{-(x^2+y^2)}{2\sigma^2}\right) \tag{4.2}$$

g_{σ_d} permet un lissage de l'image en faisant disparaître toutes les structures de taille inférieures à la largeur de la gaussienne. Cela permet également de limiter l'influence du bruit. La symétrie de la gaussienne permet de ne privilégier aucune direction lors du lissage. ***E*** correspond à la fonction d'auto-corrélation locale de l'image lissée ***L***. Cette fonction capture la structure locale de l'image. La matrice **M** caractérise le gradient au point (*x, y*) de l'image lissée. Une carte des caractéristiques est obtenue en utilisant une deuxième gaussienne pour pondérer les champs de gradient locaux obtenus :

$$E_2(x,y) = \left(\sigma_d^2 . g_{\sigma_I} * E\right)(x,y) \tag{4.3}$$

Ce second lissage permet d'obtenir les orientations principales du gradient au point (*x, y*) dans le voisinage dont la taille est liée à σ_I.

Les valeurs propres de $\boldsymbol{E_2(x, y)}$ caractérisent les deux principales orientations du gradient au point (x, y). Deux valeurs propres importantes et proches caractérisent un coin tandis que deux valeurs propres proches et faibles caractérisent une surface plane présentant peu de variations, le dernier cas caractérisant des contours.

Partant de ce principe, différentes réponses ont été proposées pour déterminer des points d'intérêts (Coins) :

$$H1: R = \frac{\det(E_2)}{(trace(E_2))^2} \tag{4.4}$$

$$H2: R = \frac{\det(E_2)}{(trace(E_2))} \tag{4.5}$$

$$H3: R = \det(E_2) - k.(trace(E_2))^2 \quad (k\text{: Constante empirique, } k \in [0.04, 0.06]) \tag{4.6}$$

Les maxima locaux de ***R*** correspondent à des points caractéristiques.

Il a été montré [SCH00] que cette réponse était robuste aux changements de point de vue, aux rotations et aux variations d'illumination. Cependant elle est très sensible aux changements d'échelle. Dans la suite, nous appelons carte caractéristique de Harris, l'image obtenue par l'une des trois formules précédentes pour une résolution donnée.

4.2.1.2. Sélection d'une échelle adaptée

Une image possédant des structures de dimensions différentes et ne connaissant pas à priori ces différentes dimensions, nous devons utiliser la formule de Harris pour différentes échelles si nous voulons caractériser l'ensemble des structures locales de l'image. Nous sommes donc amener à travailler dans un ***espace multi-échelle***, une image étant traité par la méthode de Harris à plusieurs niveaux de détail.

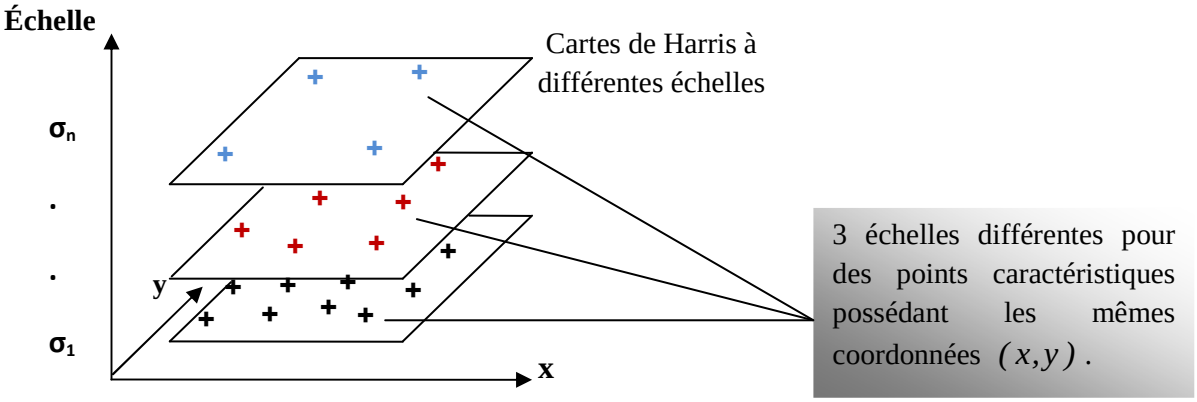


Figure 4.3. *Détecteur Harris sous un espace Multi-Echelles.*

Il apparaît alors qu'à certains points de l'image initiale correspondent plusieurs points caractéristiques à différentes échelles. Il s'agit alors de déterminer parmi ces points, ceux qui caractérisent réellement des structures locales de l'image, c'est-à-dire ceux qui sont trouvés à une échelle caractéristique. Pour cela nous calculons pour chaque niveau de détail $\boldsymbol{\sigma_n}$ pour lequel on a trouvé un point de Harris en (x,y), le Laplacien de Gaussienne de l'image correspondante :

$$|LoG(x,y,\sigma_n)| = \sigma_n^2 \left| \frac{\partial^2}{\partial X^2}(g(\sigma_n) * I)(x,y) + \frac{\partial^2}{\partial y^2}(g(\sigma_n) * I)(x,y) \right| \qquad (4.7)$$

Le Laplacien de Gaussienne lisse une image avec une gaussienne, puis les contours de l'image résultante sont détectés grâce à l'opérateur Laplacien. Ceci permet de faire ressortir des zones caractéristiques de l'image initiale.

L'échelle correspondante associée au point de Harris en (x, y) est une échelle qui caractérise si pour cette échelle le *LoG* en ce point est un maximum local. En effet la réponse du *LoG* est maximale pour une structure de même taille que la gaussienne utilisée dans le *LoG*.

4.2.1.3. Descripteurs

La caractérisation des points d'intérêts à l'aide de descripteurs est nécessaire pour apparier des points pris de différentes vue. Pour rendre cela possible, il est nécessaire que les descripteurs vérifient un certain nombre d'hypothèses :

- *Invariance au bruit.*
- *Invariance au mouvement*: le descripteur ne varie pas avec le changement de point de vue.
- *Discrimination*: les descripteurs associés à deux points distincts n'ont pas les mêmes valeurs.

Les deux premières hypothèses interdisent d'utiliser des informations simples comme la valeur des pixels dans le voisinage du point d'intérêt. La dernière hypothèse est indispensable pour limiter les faux appariements. Elle dépend non seulement de la façon de caractériser un point

d'intérêt (descripteur) mais également de la qualité du détecteur à extraire des points d'intérêt vraiment caractéristiques.

Pour s'assurer de la forte discrimination des descripteurs, ces derniers sont construits à partir des informations de la forme locale de l'image au niveau d'un point d'intérêt:

Soit

$$L_{i_{1,\dots,n}}(\sigma) = \mathrm{I} * \left(\frac{\partial}{\partial i_1} \dots \frac{\partial}{\partial i_n}\right) g_\sigma \quad \text{avec} \quad \mathrm{i}_1,\dots, \mathrm{i}_n \in \{\mathrm{x}, \mathrm{y}\}. \tag{4.8}$$

On définit un jet local de degré N d'une image au point (x, y) par:

$$J^N[i](x, y, \sigma) = \{L_{i_{1,\dots,n}}(x, y, \sigma), n = 0, \dots, N\} \tag{4.9}$$

Plusieurs descripteurs peuvent être construits à partir du jet local. Celui présenté ici est invariant à la rotation dans le plan:

$$\begin{bmatrix} L_x L_x + L_y L_y \\ L_{xx} L_x L_x + 2L_{xy} L_x L_y + L_{yy} L_y L_y \\ L_{xx} + L_{yy} \\ L_{xx} L_{xx} + 2L_{xy} L_{xy} + L_{yy} L_{yy} \\ \dots \end{bmatrix} \tag{4.10}$$

4.2.1.4. Propriétés du détecteur de points d'intérêt

L'objectif étant de réaliser une correspondance entre des points d'images prises sous différents points de vue et dans des conditions distinctes, le détecteur doit respecter certaines contraintes.

4.2.1.4.1. Stabilité géométrique (Robustesse)

a. Critère de répétabilité: La stabilité géométrique ou robustesse du détecteur caractérise sa capacité à détecter des points d'intérêt au même endroit dans les parties portant une information commune de deux images prises dans des conditions différentes (éclairage, transformation géométrique). Schmid, Mohr et Bauckhage ont introduit le critère de répétabilité pour déterminer cette robustesse [SCH00] :

$$R = \frac{C}{\min(n_1, n_2)} \quad \text{où} \quad \begin{cases} \mathbf{C} : \text{Nombre de points mis en correspondances entre les deux images} \\ \mathbf{n_i} : \text{Nombre de points trouvé dans l'image i = 1, 2 dans la zone qu'elle partage avec la seconde image} \\ \mathbf{min(n_1, n_2)} : \text{Caractérise le nombre total de correspondance pour un détecteur qui serait parfaitement stable.} \end{cases} \tag{4.11}$$

La correspondance est effectuée entre les points au sein d'un voisinage de rayon ε .

D'après les expérimentations réalisées, ce taux atteint une valeur moyenne proche de 0.9 dans le cadre d'images liées par des rotations pour ε = 1.5 pixels. Les résultats chutent pour des changements de point de vue généraux mais reste malgré tout supérieur à ceux des autres détecteurs étudiés. Remarquons notamment une meilleure stabilité géométrique pour le détecteur de Harris-Laplace que pour le détecteur SIFT dans le cadre de changements d'échelle pour ε = 1.5 pixels:

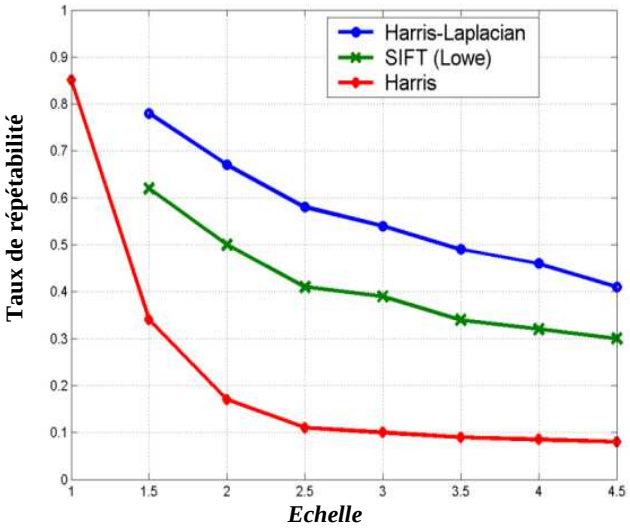


Figure.4.4. *Robustesse du détecteur de Harris-Laplace et du détecteur SIFT pour des changements d'échelle.*

b. Critère d'entropie des descripteurs: Pour éviter le faux appariement, les points d'intérêt obtenus à l'aide du détecteur doivent être caractéristiques de zones particulières afin d'obtenir des descripteurs bien différents. En effet les points obtenus dans des zones vides peuvent produire des descripteurs quasiment identiques car les variations locales de l'image au niveau de ces points selon une direction sont proches.

Pour mesurer la qualité d'un détecteur à trouver des points bien distincts, le critère de quantité d'information est introduit. Etant donné plusieurs images pour lesquelles un descripteur est calculé pour chaque point d'intérêt. Plus les descripteurs sont éloigné les uns des autres, plus ils portent de l'information. La quantité d'information permet de mesurer la distribution de ces descripteurs. Le critère choisi pour comparer les détecteurs est de comparer l'entropie des descripteurs produits par chaque détecteur. Pour cela, l'espace des descripteurs est partitionné en cellules auxquelles on associe des probabilités. Chaque cellule représente un message:

$$H(D) = -\sum_j p_j \log(p_j) \text{ avec } \begin{cases} \boldsymbol{D} : \text{Ensemble de descripteurs de même type.} \\ \boldsymbol{P_j} : \text{Probabilité d'apparition du descripteur } \boldsymbol{d_j} \text{ dans la cellule j.} \end{cases} \quad (4.12)$$

L'entropie mesure la quantité d'information moyenne portée par les descripteurs associés aux points d'intérêt d'un détecteur.

Ce calcul dépend fortement a priori du choix du descripteur étudié, un descripteur peu discriminant risque d'entrainer une chute de l'entropie.

Nous pouvons constater que le descripteur introduit par le détecteur SIFT permet d'obtenir de meilleurs résultats en termes d'entropie que le détecteur « Harris-Laplace ».

Détecteur	**Harris-Laplace**	**SIFT**
Entropie	7.4	8

Tableau 4.1. *Entropie des descripteurs calculée pour des points d'intérêt issus de deux détecteurs : le détecteur de Harris-Laplace et le détecteur «SIFT ».*

4.2.1.5. Algorithme utilisé

L'algorithme utilisé comporte deux étapes:

- Détermination des points d'intérêt de l'image et de l'échelle caractéristique associée.
- Caractérisation des points d'intérêt à l'aide de descripteurs.

La première étape se décompose en plusieurs sous étapes. L'image d'origine est parcourue point par point, et pour chaque point et échelle, nous devons vérifier:

1. Si le point considéré est un point de Harris et s'il est un maximum local à l'échelle considérée (comparaison de la valeur de la réponse de Harris au point considéré avec celle de ses huit voisins).

2. Dans le cas où le point considéré est un point de Harris pour un ensemble d'échelles, il sera sélectionné dans cet ensemble à l'aide du Laplacien de Gaussienne.

3. Des seuils sont utilisés pour extraire des points réellement caractéristiques.

Voici les différentes étapes de sélection des points d'intérêt dans une carte de Harris :

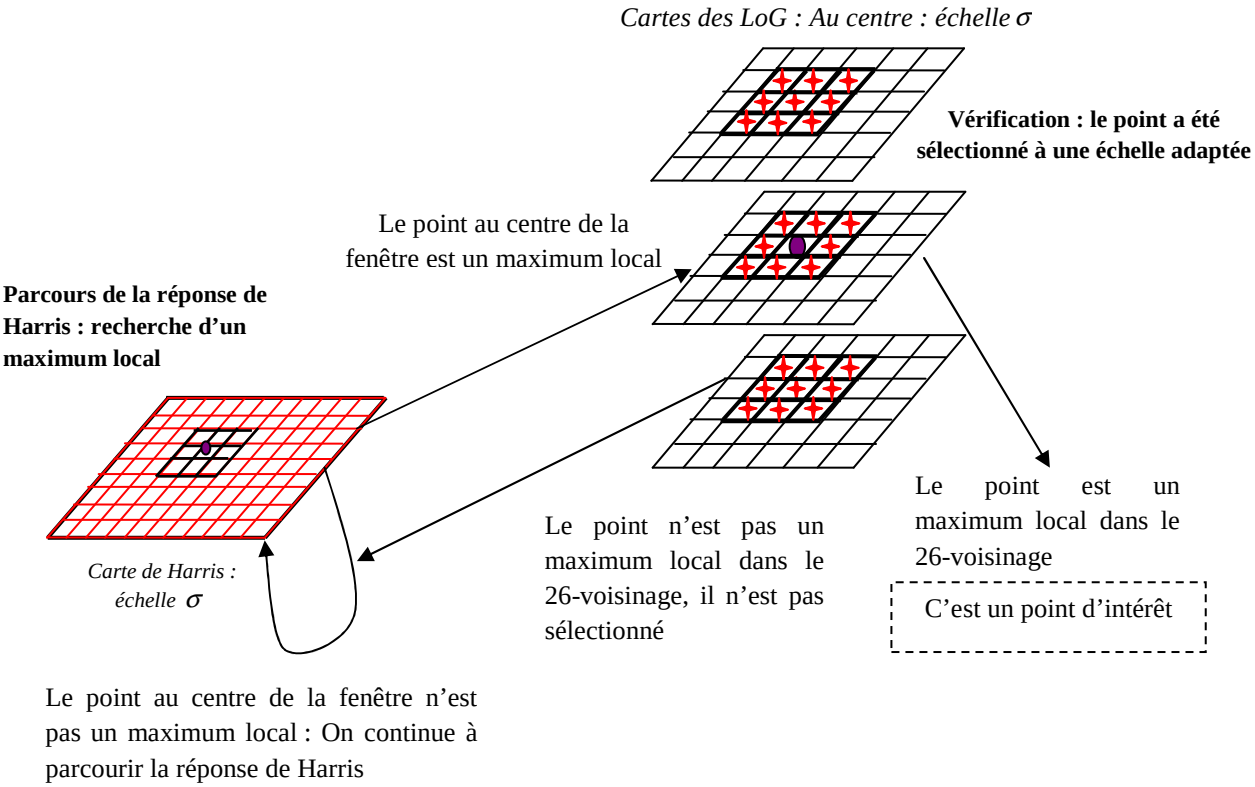


Figure 4.5. *Localisation de points d'intérêt pour une échelle à partir d'une carte de Harris.*

4.2.1.6. Paramétrisation

4.2.1.6.1. Espace multi-échelle, facteur d'échelle

L'objectif final est d'établir des correspondances entre des images à des résolutions différentes c'est pourquoi est utilisée la représentation multi-échelles présentée figure 4.5. Par conséquent il est nécessaire de déterminer combien d'échelles il faut utiliser pour une bonne représentation des structures locales des images. Le problème à résoudre est d'utiliser un panel d'échelles suffisamment important pour une bonne représentation des informations de chaque image tout en s'assurant d'un temps de calcul limité. Dans la littérature est proposé d'utiliser 17 échelles avec un facteur d'échelle $\xi = \sqrt{2}$ entre elles [MIK04].

Pour valider ce facteur nous avons calculé pour plusieurs images une représentation multi-échelle utilisant ce facteur, puis pour des points caractéristiques de ces images nous avons vérifié que le point d'intérêt sélectionné par la méthode de Harris-Laplace possédait une échelle caractéristique proche d'un maximum local réel du LoG en ce point, et que par conséquent ce point caractérisait correctement une structure locale de l'image.

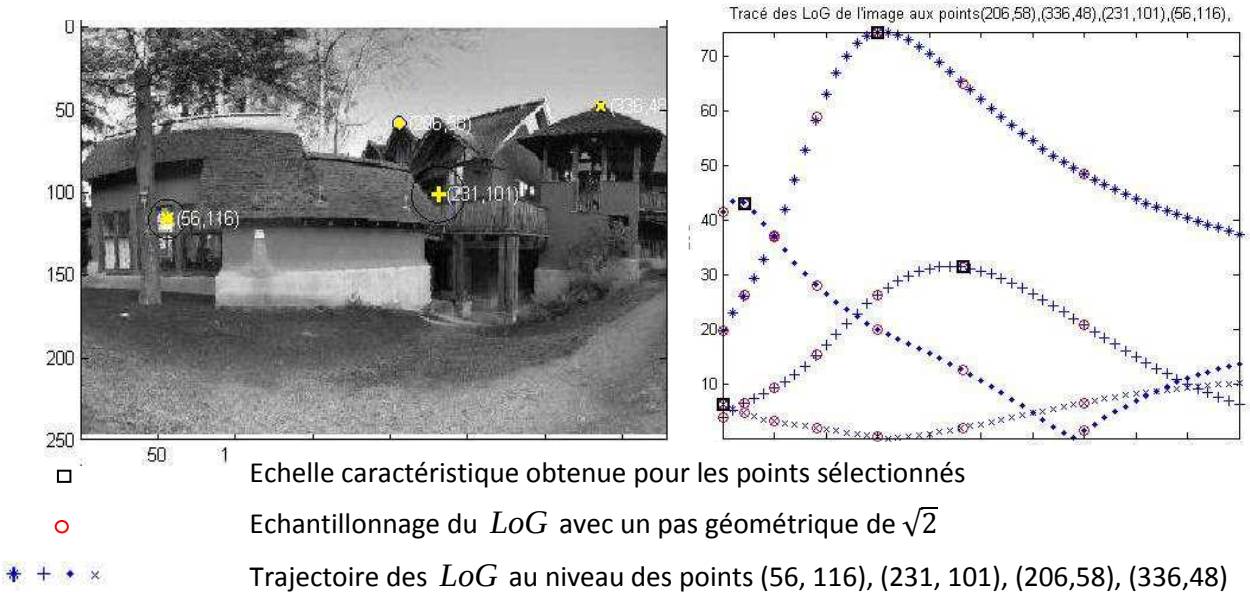


Figure 4.6. *Echelles caractéristiques obtenues pour quatre points d'intérêt dans un espace multi-échelle utilisant un facteur d'échelle de* $\sqrt{2}$.

Nous constatons qu'avec ce facteur d'échelle, l'échantillonnage des LoG est suffisamment important pour obtenir pour les points d'intérêts des échelles caractéristiques proches des échelles caractéristiques théoriques (maxima locaux).

4.2.1.6.2. Détermination des seuils

L'utilisation de la méthode de Harris multi-échelle pose différents problèmes.

Tout d'abord le calcul à différentes échelles implique la détermination d'un grand nombre de points caractéristiques. Si ce nombre est diminué en utilisant le Laplacien de Gaussienne (un point caractéristique étant caractérisé par sa position dans l'image et l'échelle associée, nous ne conservons pas forcément toutes les échelles…), il reste cependant très important. Pour limiter les temps de calcul dus à la détermination des descripteurs et par la suite à l'appariement de points, il ne faut conserver que les points associés à des structures saillantes de l'image.

Ensuite, le calcul à différentes échelles entraîne la délocalisation de points caractéristiques, la structure locale étant « élargie » sur l'image lissée à partir de laquelle nous avons calculé la réponse de Harris comme l'illustre la figure suivante :

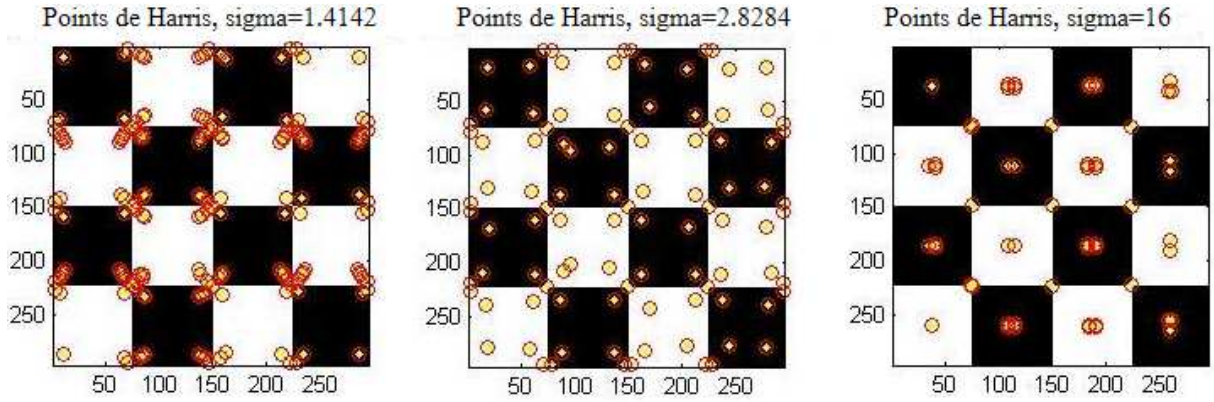


Figure 4.7. *Délocalisation des points caractéristiques avec l'échelle.*

Cette figure a été obtenue en utilisant la formule (H1) de la réponse du " *cornerness* ".

Nous constatons que les points correspondant aux coins du damier sont délocalisés vers les centres des carrés de celui-ci. Il existe en général deux types de points délocalisés :

- Ceux qui caractérisent réellement la structure locale. C'est le cas des points au centre des carrés du damier qui caractérisent le centre de ces structures. Pour ceux-ci, la réponse du LoG est proche du maximum local théorique.

- Ceux qui sont obtenus après un fort lissage dans des zones vide et qui ne caractérise rien. La réponse du LoG pour ceux-ci est faible par rapport au maximum théorique.

Dans le cas du damier, l'utilisation simple du LoG pour la sélection de l'échelle suffit à éliminer l'essentiel des points délocalisés, comme le montre la figure suivante.

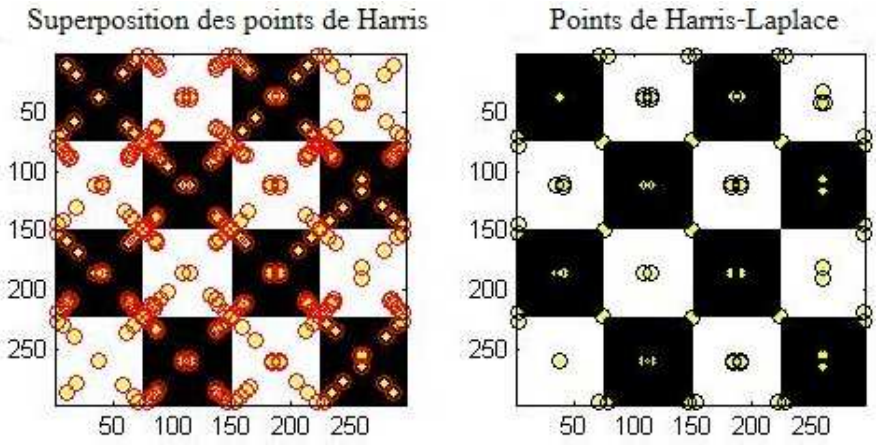


Figure 4.8. *Elimination de points caractéristiques par sélection des échelles adaptées.*

Cependant il arrive que des points soient trouvé dans des zones complètement vide (délocalisation, bruit…), et pour lesquels le LoG à l'échelle du point soit un maximum local.

Pour résoudre les deux problèmes nous avons utilisé un couple de seuils:

- Le seuil utilisé sur le Laplacien de Gaussienne vise principalement à conserver un maximum d'informations sur les structures dans les échelles appropriées et à éliminer les points obtenus pour de mauvaises échelles qui sont délocalisés dans des zones vides.

- Le seuil de Harris vise à ne conserver parmi les points d'intérêts trouvés à partir de la réponse de Harris que les maximums suffisamment saillants. Pour cela on compare pour la réponse de Harris la différence d'amplitude entre un maximum local et les points de son entourage proche avec un seuil : le seuil de Harris. Ce seuil est particulièrement efficace pour éliminer les maxima locaux de la réponse de Harris qui sont dus au bruit, mais également les points obtenus après fort lissage et qui ne sont pas toujours pertinents. La figure 4.9 a été obtenue en utilisant un seuil de Harris de 0.3 sur la réponse du "*cornerness*" de l'équation (H2).

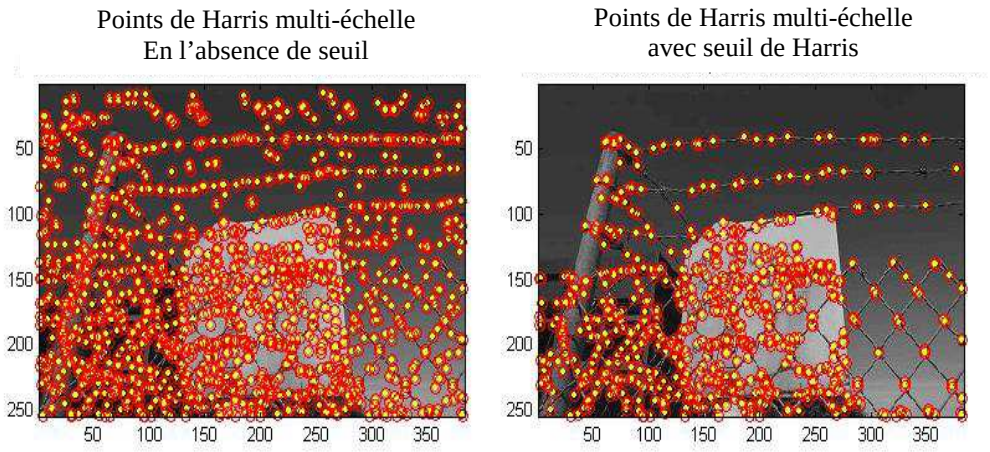


Figure 4.9. *Extraction des points caractéristiques saillants.*

La détermination de seuils pertinents est difficile puisque ceux-ci doivent fonctionner avec tout type d'image. L'objectif est de conserver au final que les points avec la bonne échelle caractéristique. Le tableau suivant résume nos résultats :

cornerness	Seuil de Harris	Seuil sur le *LoG*
(H1)	[0.001-0.01]	[1-20]
(H2)	[0.01-0.5]	[1-20]
(H3)	[0-200]	[1-20]

Tableau 4.2. *Seuils de Harris et de Laplacien de Gaussienne*

Remarquons que la réponse (H1) semble la plus sujette à la délocalisation des points d'intérêts puisqu'en l'absence de seuil on obtient avec les réponses (H2) et (H3) (figure 4.10):

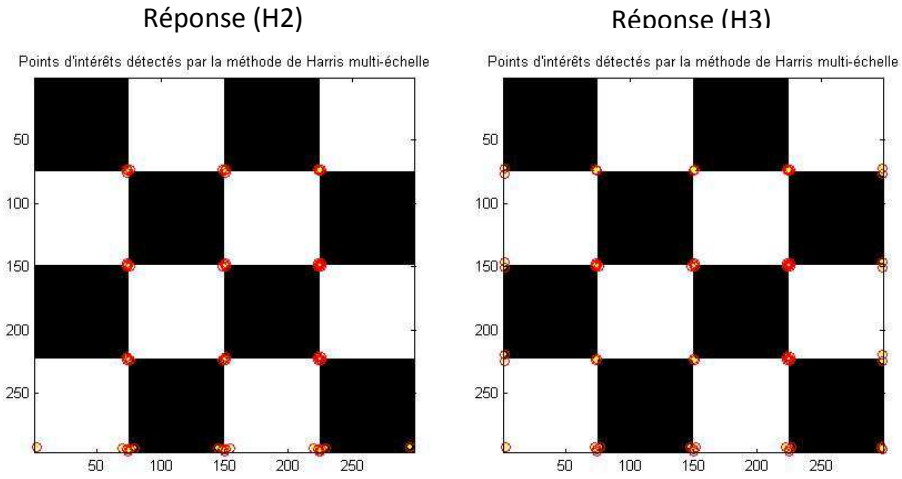


Figure 4.10. *Points caractéristiques obtenus respectivement avec les réponses (H2) et (H3).*

Pour bien visualiser l'utilité des seuils, voici les résultats obtenus avec la réponse (H2) (figure 4.11) en l'absence de seuils pour 4 échelles: $\sigma = 1.4, \sigma = 2.744, \sigma = 5.378, \sigma = 14.758$.

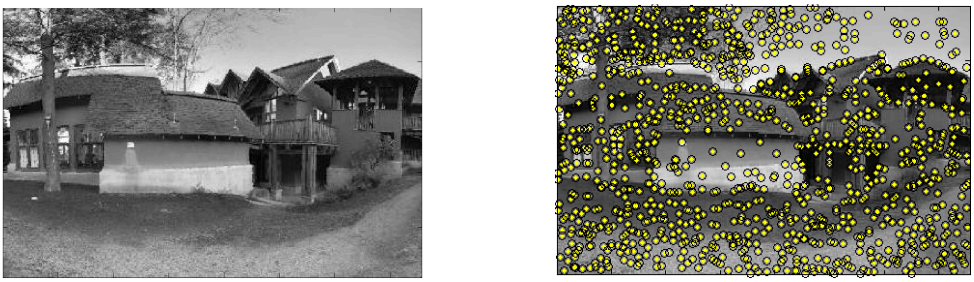


Figure 4.11. *Points de Harris-Laplace en l'absence de seuil.*

Voici maintenant les points obtenus avec la réponse (H2) (figure 4.12) pour les mêmes échelles avec un seuil de Harris de 0.3 à gauche et à droite et un seuil sur le Laplacien de Gaussienne de 15:

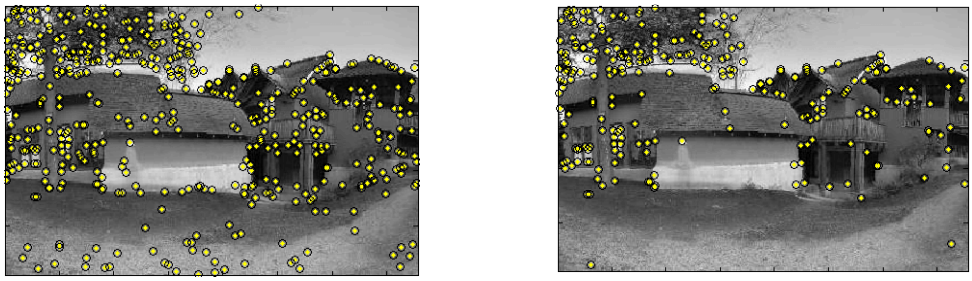


Figure 4.12. *Points d'intérêt de Harris-Laplacien de Gaussienne.*

Le seuil de Harris a permis l'élimination des points peu pertinents dans le ciel. Le seuil sur le Laplacien de Gaussienne [MED10] permet de ne conserver que les points réellement caractéristiques des structures locales.

4.2.2. Algorithme basé sur l'aspect géométrique du point d'intérêt

4.2.2.1. Problématique

Nous présentons dans cette partie une nouvelle approche [MED13] de détection de coins et qui a montré une grande stabilité par rapport à d'autres détecteurs. Dans la suite de cette partie, nous dénommerons par $\boldsymbol{Q}$ le point d'intérêt recherché (un coin) et par $\boldsymbol{\Omega}(\boldsymbol{Q})$ son voisinage.

La plupart des méthodes actuelles travaillent sur tout le voisinage $\boldsymbol{\Omega}(\boldsymbol{Q})$ du point $\boldsymbol{Q}$ afin de déterminer à l'aide d'une fonction de réponse de coin prédéfinie si $\boldsymbol{Q}$ est un coin ou non. Remarquons que l'analyse est pour toute une classification de détecteurs, directement basée sur les valeurs radiométriques de l'image et plus particulièrement sur celles prises par le gradient. Cependant, en considérant tout le voisinage $\boldsymbol{\Omega}(\boldsymbol{Q})$ pour déterminer si le point $\boldsymbol{Q}$ est un coin, les différents algorithmes existants aboutissent parfois à des imprécisions non souhaitables dans les applications. Par exemple, dans la figure 4.13 on souhaite analyser le point $\boldsymbol{Q_1}$, à gauche, lorsque tout le voisinage de $\boldsymbol{Q_1}$ est considéré dans l'étude, il y a de fortes chances que le point $\boldsymbol{Q_2}$ interfère dans la décision finale car les valeurs du gradient y sont importantes. En revanche, à droite, en sélectionnant uniquement une aire intéressante de $\boldsymbol{\Omega}(\boldsymbol{Q})$ on peut prendre une décision raisonnable sur le point $\boldsymbol{Q_1}$.

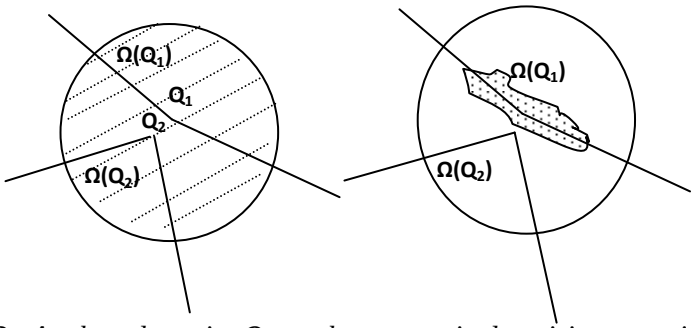


Figure 4.13. *Analyse du point $\boldsymbol{Q_1}$, seule une partie du voisinage est intéressante.*

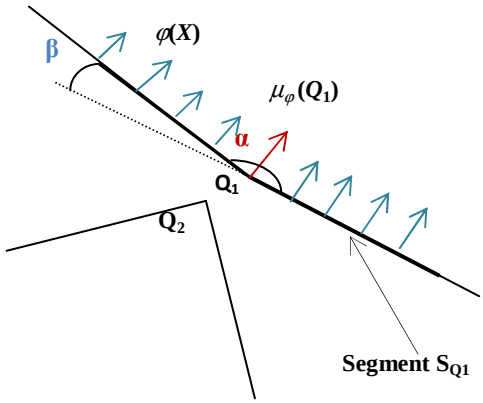
4.2.2.2. Objectif de l'algorithme

Un coin est un point particulier de l'image situé à l'intersection d'au moins deux bords. Nous considérerons que les bords et le coin lui même sont localement situés sur une même ligne de niveau (donc, il est toujours possible d'interpoler l'image).

- Soit $\boldsymbol{S_{Q1}}$ l'ensemble des points situés sur les bords du point $\boldsymbol{Q_1}$ dans la figure 4.14. Cette définition semble en apparence très triviale. Cependant, nous verrons que la principale difficulté se trouve dans la formulation rigoureuse et mathématique de $\boldsymbol{S_{Q1}}$ et que son calcul s'exprime par une approximation explicitée plus loin dans ce document.
- Soit $\boldsymbol{\varphi}(\boldsymbol{X})$ la direction de gradient au point $\boldsymbol{X}$. On définit par ailleurs $\boldsymbol{g}(\boldsymbol{X})$ comme l'intensité de gradient au point $\boldsymbol{X}$

- Supposons que $\boldsymbol{\varphi(X)}$ soit connu pour tout $\boldsymbol{X}$ appartenant à $\boldsymbol{S_{Q1}}$ et notons $\boldsymbol{H_{Q1}}$ l'ensemble des vecteurs $\boldsymbol{\varphi(X)}$ tels que $\boldsymbol{X \in S_{Q1}}$. Quelques éléments de l'ensemble $\boldsymbol{H_{Q1}}$ sont représentés en bleu dans la figure 4.14.

L'idée alors est de calculer les moments d'ordre un et deux des éléments dans l'ensemble $\boldsymbol{H_{Q1}}$. Le moment d'ordre un, noté $\boldsymbol{\mu_{\varphi}(Q_1)}$ est représenté en rouge dans la figure 4.14. Il s'agit en fait tout simplement de la moyenne des vecteurs $(\boldsymbol{\varphi(X)})_{\boldsymbol{X \in S_{Q1}}}$. $\boldsymbol{\mu_{\varphi}(Q_1)}$ définit l'axe du coin $\boldsymbol{Q_1}$("Axis corner"), c'est à dire la droite qui coupe l'angle en deux. Le moment d'ordre deux est noté $\boldsymbol{\sigma^2_{\varphi}(Q_1)}$. Nous pouvons alors montrer que l'angle $\boldsymbol{\beta = \pi - \alpha}$ s'exprime par $\boldsymbol{2\sigma_{\varphi}(Q_1)}$. Il est aussi appelé "angle de cassure" (angle of break)



"Axis Corner": moyenne locale des directions de gradient

β = π-α α : l'angle de coin

Figure 4.14. *Caractérisation d'un coin.*

La décision finale quant au point $\boldsymbol{Q}$ est prise à l'aide de deux fonctions qui dépendent de ces deux valeurs $\boldsymbol{\mu_{\varphi}(Q)}$ et $\boldsymbol{\sigma_{\varphi}(Q)}$:

- ***CRF(Q)*** (Corner Response Function) : C'est la fonction de réponse de coin
- ***Appar(Q)*** Apparence du coin (définit plus loin)

4.2.2.3. Formalisation du problème

Comme nous l'avons déjà signalé ci-dessus, toute la difficulté de l'algorithme consiste à bien estimer l'ensemble $\boldsymbol{S_Q}$. Dans la pratique, on ne calcule pas explicitement $\boldsymbol{S_Q}$ mais on cherche à trouver la probabilité $\boldsymbol{P_{SG}(Q)}$ que chaque point $\boldsymbol{X \in \Omega(Q)}$ soit "proche" de $\boldsymbol{S_Q}$. Intuitivement, nous cherchons à ce que la valeur de $\boldsymbol{P_{SG}(X)}$ soit forte lorsque $\boldsymbol{X \in \Omega(Q)}$ est "proche" de $\boldsymbol{S_Q}$ et faible si $\boldsymbol{X \in \Omega(Q)}$ est "loin" de $\boldsymbol{S_Q}$. Pour formaliser un peu plus rigoureusement cette idée, nous considérons que $\boldsymbol{P_{SG}(X)}$ exprime la probabilité que $\boldsymbol{X \in \Omega(Q)}$ appartienne à un segment d'approximation de la ligne de niveau contenant et pointant

vers le coin Q. La figure 4.15 illustre cette probabilité dans un cas très simple. Pour les points en rouge (X_1 à X_6) la probabilité $(P_{SG}(X_i))_{X\in[1-6]}$ est très forte.

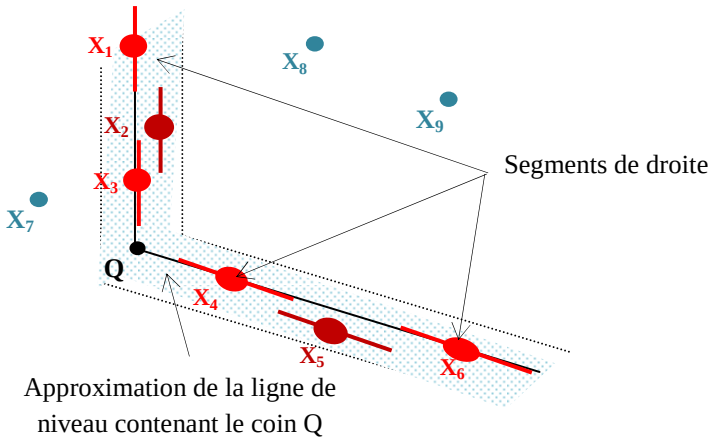


Figure 4.15. *Représentation simplifiée de la probabilité* $P_{SG}(X)$.

Ainsi, la connaissance de $P_{SG}(X), \forall X \in \Omega(Q)$ permet de pondérer les vecteurs de $(\varphi(X))_{X\in\Omega(Q)}$ afin d'estimer les moments $\mu_{\varphi}(Q)$et $\sigma_{\varphi}(Q)$ (Nous verrons plus loin que cette pondération n'est pas suffisante et qu'il faut prendre aussi en compte la distance du point X au coin potentiel Q).

La probabilité $P_{SG}(X)$ se calcule par l'intermédiaire de 3 critères dénommés par ***BrQ***, ***DirQ*** et ***AngQ*** que nous allons expliquer en détail dans la partie suivante. Enfin, il est important de remarquer que l'algorithme est capable de calculer explicitement l'angle formé par la ligne de niveau.

4.2.2.4. Restriction du voisinage étudié : 3 critères

Dans cette partie nous montrons comment calculer une approximation de l'ensemble S_Q (voir figure 4.14) en utilisant la probabilité $P_{SG}(Q)$.
On suppose qu'il existe au moins un objet ayant une courbure dans $\Omega(Q)$. Avant de détailler les critères, il faut considérer les éléments suivants :

- **L'approximation d'une courbe :** un point X appartient à l'approximation d'une courbe si cette dernière passe par le pixel contenant Q. On note alors $d(X)$ la déviation (voir figure 4.16).

- **Lignes de niveaux** Les lignes de niveaux dans $\Omega(Q)$, sont représentées par des segments de droites.

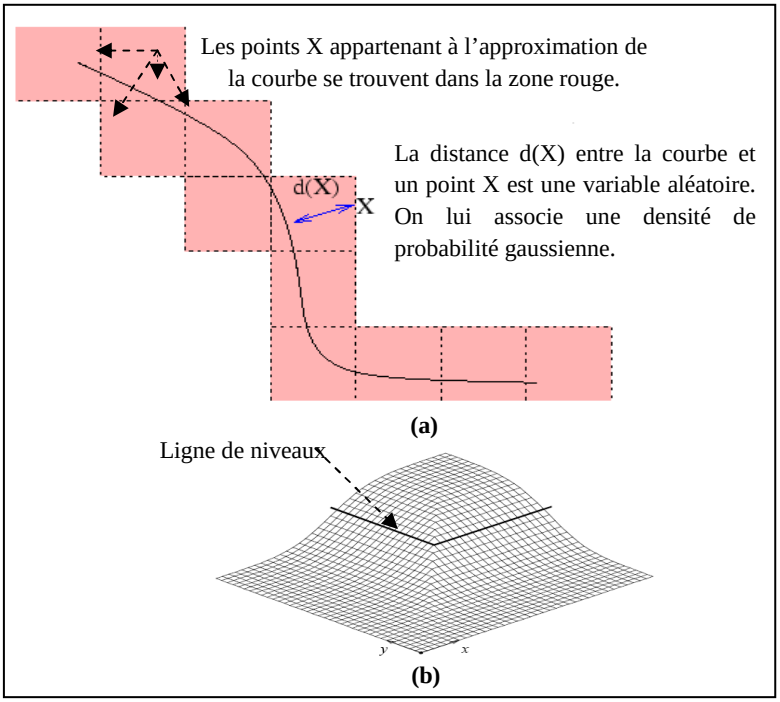


Figure.4.16. *Restriction de voisinage*
(a) Distance d(X) entre une courbe et un point X, (b) Fonction de lignes des niveaux d'un coin.

Afin de calculer la probabilité $\boldsymbol{P_{SG}}$, nous allons introduire trois critères permettant de restreindre de manière pertinente la zone à étudier.

4.2.2.4.1. Premier critère : BrQ

De manière générale, pour un point test $\boldsymbol{Q}$, les points constituant une bonne approximation de $\boldsymbol{S_Q}$ doivent avoir une radiométrie proche de celle de $\boldsymbol{Q}$. La quantité $\Delta \boldsymbol{b}(\boldsymbol{X}) = \boldsymbol{b}(\boldsymbol{X}) - \boldsymbol{b}(\boldsymbol{Q})$ permet de mesurer pour cela la radiométrie d'un point $\boldsymbol{X}$ du voisinage $\boldsymbol{\Omega(Q)}$ par rapport à $\boldsymbol{Q}$ ($\boldsymbol{b}$ est le niveau de l'image au point $\boldsymbol{X}$). Le critère $\boldsymbol{BrQ}$ est l'événement (au sens des probabilités) tel que $\boldsymbol{\Delta b(X)=0}$. En d'autres termes, l'événement $\boldsymbol{BrQ}$ est vérifié pour un certain point $\boldsymbol{X}$, si ce dernier a le même niveau radiométrique que $\boldsymbol{Q}$.

On peut alors définir la probabilité conditionnelle en un point $\boldsymbol{X}$ du voisinage $\boldsymbol{\Omega(Q)}$ telle que l'événement $\boldsymbol{BrQ}$ soit vérifié: $\boldsymbol{P}(\boldsymbol{BrQ}|\Delta(\boldsymbol{X}))$. Les plus fortes valeurs d'une telle probabilité correspondent aux points $\boldsymbol{X} \in \boldsymbol{\Omega}(\boldsymbol{Q})$ qui ont une radiométrie proche de celle de $\boldsymbol{Q}$, ce qui revient à mettre en valeur une approximation des lignes de niveau de $\boldsymbol{Q}$. Dans la figure 4.17, les points $\boldsymbol{X}$ tels que la valeur de $\boldsymbol{P}(\boldsymbol{BrQ}|\Delta(\boldsymbol{X}))$ soit parmi les plus fortes sont situés dans la zone pointillée, et c'est justement en ces points que la radiométrie est la plus proche de celle de Q.

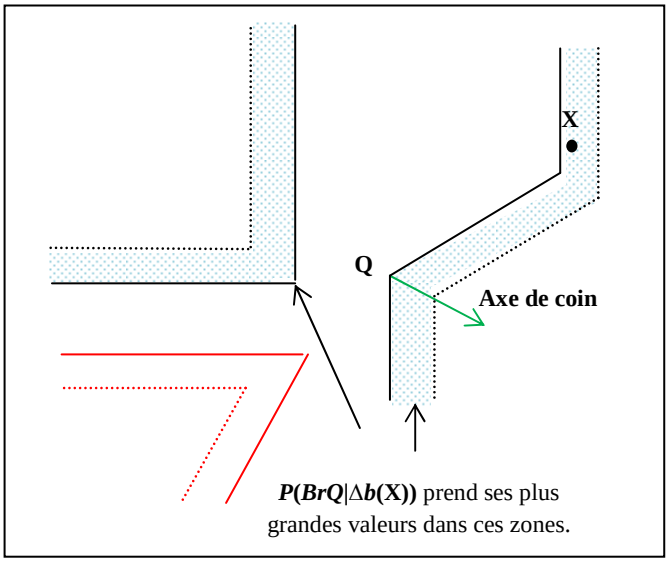


Figure.4.17. *Restriction par le critère* ***BrQ*** *à l'aide la probabilité conditionnelle* $\boldsymbol{P(BrQ|\Delta b(X))}$**.**

4.2.2.4.2. Deuxième critère: DirQ

En observant la figure 4.17, on s'aperçoit immédiatement que le critère ***BrQ*** est loin d'être suffisamment restrictif. En effet, on souhaite que les points appartenant à un segment [de ligne de niveau] ne pointant pas vers le point ***Q*** ne soient pas pris en compte. Il faut alors introduire la déviation ***h(X)*** d'un point $\boldsymbol{X \in \Omega(Q)}$ par rapport au point ***Q***. Pour construire cette déviation, on considère la droite $\boldsymbol{p_x}$ perpendiculaire au vecteur $\boldsymbol{\varphi(X)}$ passant par le point ***Q***. La déviation ***h(x)*** est la distance du point ***X*** à la droite $\boldsymbol{p_x}$. Le critère ***DirQ*** est l'événement (au sens des probabilités) tel que ***h(X)=0***. En d'autres termes, l'événement ***DirQ*** est vérifié pour un certain point ***X***, si ce dernier appartient à un segment [de la ligne de niveau] pointant vers ***Q***.

On définit alors la probabilité conditionnelle en un point $\boldsymbol{X \in \Omega(Q)}$ telle que l'événement ***DirQ*** soit vérifié: $\boldsymbol{P\left(DirQ \middle| h(X)\right)}$. En considérant le produit $\boldsymbol{P\left(BrQ \middle| \Delta(X)\right).P(DirQ|h(X))}$, on construit une nouvelle probabilité qui ses prend ses plus fortes valeurs lorsque le point $\boldsymbol{X \in \Omega(Q)}$ possède une radiométrie proche de celle de ***Q*** et appartient à un segment [de ligne de niveau] pointant vers ***Q***. Il faut bien remarquer que les deux probabilités $\boldsymbol{P\left(BrQ \middle| \Delta(X)\right)}$ et $\boldsymbol{P(DirQ|h(X))}$ sont indépendantes.

Dans la figure 4.18, les points ***X*** tels que la valeur de $\boldsymbol{P\left(BrQ \middle| \Delta(X)\right).P(DirQ|h(X))}$ soit parmi les plus fortes sont situés dans la zone pointillée.

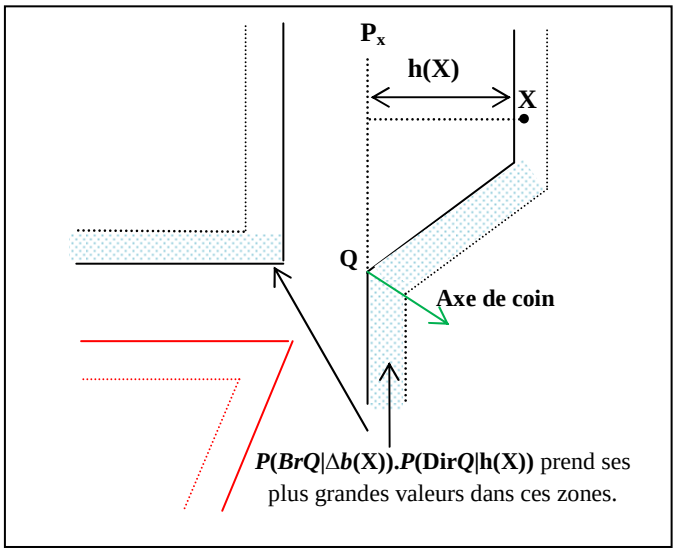


Figure 4.18. *Restriction par les critères* ***BrQ*** *et* ***DirQ*** *à l'aide la probabilité* $\boldsymbol{P(BrQ|\Delta(X))}.\boldsymbol{P(DirQ|h(X))}$ *les points* ***X*** *qui n'appartiennent pas à un segment [de ligne de niveau] pointant vers* ***Q*** *sont supprimés (par rapport à la figure 4.17).*

4.2.2.4.3. Troisième critère : AngQ

Afin d'obtenir une bonne approximation de $\boldsymbol{S_Q}$, un troisième et dernier critère est nécessaire. On note la différence d'angle $\Delta(\boldsymbol{\varphi}(\mathbf{X}))$ entre le vecteur $\boldsymbol{\varphi}(\mathbf{X})$ et l'axe du coin. Les points de l'approximation de $\boldsymbol{S_Q}$ doivent avoir un vecteur $\boldsymbol{\varphi}(\mathbf{X})$, telle que $\Delta(\boldsymbol{\varphi}(\mathbf{X}))$ soit inférieure à un certain seuil. Le critère ***AngQ*** est l'événement (au sens des probabilités) tel que $\Delta(\boldsymbol{\varphi}(\mathbf{X})) \leq \frac{\pi}{2}$. L'angle $\Delta(\boldsymbol{\varphi}(\mathbf{X}))$ sera toujours pris positif ou nul, inférieur ou égal à π.

On peut alors définir une dernière probabilité conditionnelle en un point $\boldsymbol{X} \in \boldsymbol{\Omega(Q)}$ telle que l'événement ***AngQ*** soit vérifié : $\boldsymbol{P(AngQ|\Delta(\varphi(X)))}$ (cette probabilité est encore une fois indépendante des deux précédentes). En considérant finalement le produit $\boldsymbol{P(BrQ|\Delta(X))}.\boldsymbol{P(DirQ|h(X))}.\boldsymbol{P(AngQ|\Delta(\varphi(X)))}$, on construit une nouvelle probabilité qui prend ses plus fortes valeurs sur l'ensemble des points $\boldsymbol{X} \in \boldsymbol{S_Q}$. Dans la figure 4.19, les points $\boldsymbol{X}$ tels que la valeur de $\boldsymbol{P(BrQ|\Delta(X))}.\boldsymbol{P(DirQ|h(X))}.\boldsymbol{P(AngQ|\Delta(\varphi(X)))}$ soit parmi les plus fortes sont situés dans la zone pointillée.

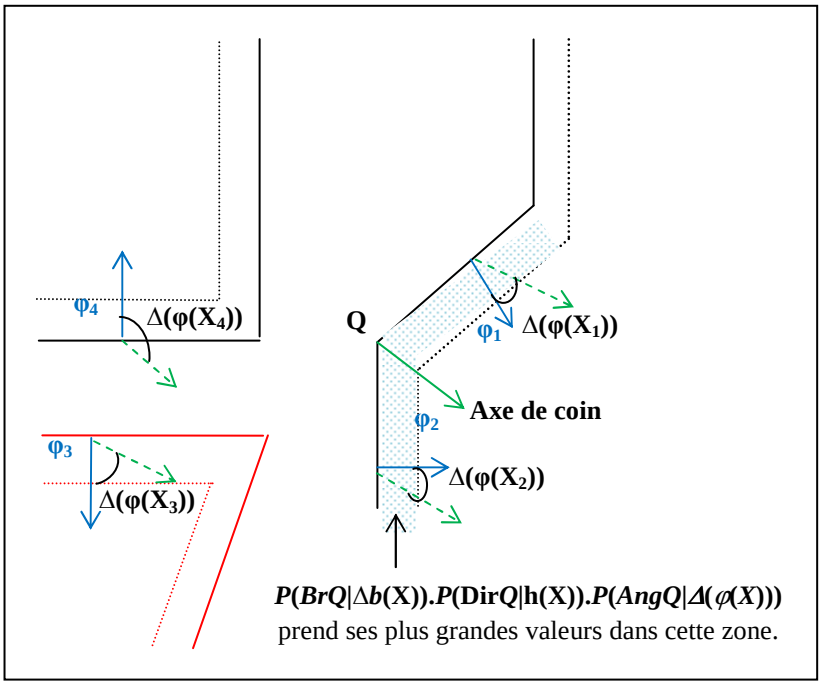


Figure 4.19. *Restriction par les critères* ***BrQ*** *et* ***DirQ*** *à l'aide la probabilité* $\boldsymbol{P(BrQ|\Delta(X)).P(DirQ|h(X)).P(AngQ|\Delta(\varphi(X)))}$.

4.2.2.4.4. Réunion des 3 critères

A partir des différentes probabilités calculées précédemment, on obtient facilement la probabilité $\boldsymbol{P_{SG}(X)}$ qu'un point $\boldsymbol{X}$ appartient à un segment de l'approximation de la ligne de niveau contenant le coin $\boldsymbol{Q}$:

$$P_{SG}(X) = min_{Y \in \overline{QX}}\{P(BrQ|\Delta b(Y)).P(DirQ|h(Y)).P(AngQ|\Delta(\varphi(Y)))\} \qquad (4.13)$$

Dans la pratique, on ne calcule pas directement $\boldsymbol{P_{SG}(X)}$ sous la forme de l'équation ci-dessus, mais à l'aide d'un calcul bayésien, on réécrit $\boldsymbol{P_{SG}(X)}$ de la manière suivante:

$$P_{SG}(X) \approx min_{Y \in \overline{QX}}\left\{P_d\left(\frac{2\Delta b(Y)}{g(Y) + g(Q)}\right)P_d(h(Y))\right\} \qquad (4.14)$$

4.2.2.5. Décision finale: quels sont les coins ?

La section précédente expliquait le calcul de la probabilité $\boldsymbol{P_{SG}(X)}$ à partir des 3 critères (ou événements), ***BrQ***, ***DirQ*** et ***AngQ*** que nous avons proposé. Nous rappelons que $\boldsymbol{P_{SG}}$ exprime la probabilité que $\boldsymbol{X \in \Omega(Q)}$ appartient à un segment de l'approximation de la ligne de niveau contenant le coin $\boldsymbol{Q}$.

4.2.2.5.1. Estimation des moments μ_φ et $\sigma_\varphi(X)$

Le calcul de la moyenne pondérée par les valeurs de $\boldsymbol{P_{SG}(X)}$ dans le voisinage $\boldsymbol{\Omega(Q)}$ de $\boldsymbol{Q}$ ***p***ermettent d'estimer les valeurs de μ_φ et $\sigma_\varphi(X)$ par restriction à une zone intéressante du voisinage $\boldsymbol{\Omega(Q)}$. Cependant, la pondération doit aussi prendre en compte la distance du point $\boldsymbol{X}$ au coin potentiel $\boldsymbol{Q}$ et la pondération utilisée est donnée par :

$$w(X) = P_{SG}(X) w_r\big(r(X)\big) \tag{4.15}$$

où :

- $\boldsymbol{r(X)}$ est la distance du point $\boldsymbol{X}$ au coin potentiel $\boldsymbol{Q}$
- $\boldsymbol{w_r(r(X))}$ est une fonction positive de la distance $\boldsymbol{r(X)}$ prédéfinie

La figure 4.20 illustre la pondération des directions des gradients $\boldsymbol{\varphi(X)}$ dans le voisinage $\boldsymbol{\Omega(Q)}$

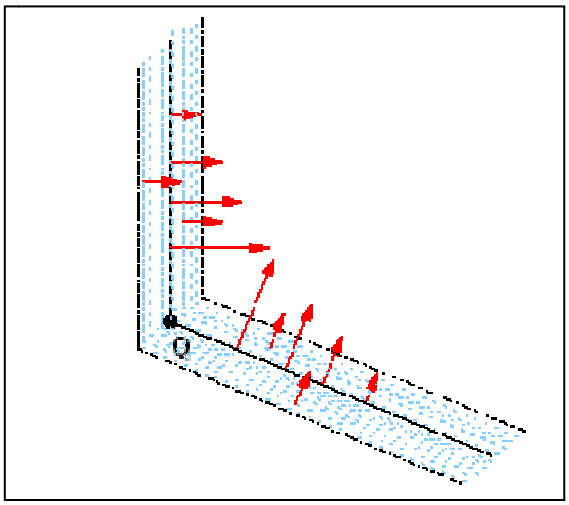


Figure 4.20. *Vecteurs $\boldsymbol{\varphi(X)}$ (directions du gradient) pondérés par $\boldsymbol{w(X)}$ pour le calcul de $\boldsymbol{\mu_\varphi}$ et $\boldsymbol{\sigma_\varphi(X)}$: dépendance de la distance et de la valeur de $\boldsymbol{P_{SG}(X)}$.*

Finalement, $\boldsymbol{\mu_\varphi}$ et $\boldsymbol{\sigma_\varphi(X)}$ s'écrivent :

$$\mu_\varphi(Q) = \frac{\sum_{X_i \in \Omega(Q)} w(X_i)\varphi(X_i)}{\sum_{X_i \in \Omega(Q)} w(X_i)}, \quad C'est\ une\ moyenne\ des\ \varphi(X_i) pondérés\ par\ w(X_i) \tag{4.16}$$

$$\sigma_\varphi^2(Q) = \frac{\sum_{X_i \in \Omega(Q)} w(X_i)[\varphi(X_i) - \mu_\varphi]^2}{\sum_{X_i \in \Omega(Q)} w(X_i)}, C'est\ une\ variance\ des\ \varphi(X_i) pondérés\ par\ w(X_i) \tag{4.17}$$

Nous rappelons que $\boldsymbol{\mu_\varphi(Q)}$ représente l'"axe de coin" (axis corner) et 2* $\boldsymbol{\sigma_\varphi(Q)}$ est "l'angle de cassure" (angle of break)

4.2.2.5.2. Calcul de l'Apparence et de la fonction de réponse de coin

Nous avons précédemment évoqué les fonctions ***CRF(Q)*** et ***Appar(Q)*** que nous allons définir à présent.

- ***CRF(Q)*** est la fonction de réponse du coin "Corner Response Function" qui caractérise les coins, et on peut montrer qu'elle prend sa valeur maximale sur le point du coin théorique.

$$CRF(Q) = g(Q)\sigma_{\varphi}^{2}(Q) \tag{4.18}$$

- ***Appar(Q)*** est l'apparence "Apparency" du coin étudié. Cette fonction caractérise la taille de la zone intéressante (c'est à dire l'approximation de ***S(Q)***) et détermine lorsque sa valeur dépasse un certain seuil fixé si le coin est "apparemment" un coin.

$$Appar(Q) = \sum_{X_i \in \Omega(Q)} P_{SG}(X_i) g(X_i) \left| \varphi(X_i) - \mu_{\varphi} \right| \tag{4.19}$$

4.2.2.6. Algorithme proposé

Entrée:

- Charger l'image
- Pd : Densité de probabilité de la déviation du point approximant une courbe.
- Wr : Fonction de distance (positive).
- Sg : Seuil pour la valeur de g(X) (intensité du gradient).
- S_{σ} : Seuil pour la valeur de σ_{φ}
- S_{Appar} : Seuil pour la valeur de *Appar*.

Coins_potentiels ← {}
Coins ← {}
Pour tout $X \in Image$ **faire**
- Calculer $\varphi(X)$: direction de gradient en X.
- Calculer $g(X)$: intensité de gradient en X.

Si $g(X) > S_g$ **alors :**
- Seuillage du gradient.
- Coins_potentiels ← Ajouter ({X}).

Fin Si.
Fin Pour.
Pour tout $Q \in$ Coins_potentiels **faire**
Si $CRF(Q) == CRF_{MAX}$ **alors :**
Si $\sigma_{\varphi} > S_{\sigma}$ **Alors :**
Si $Appar(Q) > S_{Appar}$ **alors :**
Coins ← Ajouter ({X}).
Fin Si
Fin Si
Fin Si

- Calculer PSG.

Fin Pour.

Nous remarquons que le détecteur demande beaucoup de paramètres en entrée, et d'autant plus dans l'implémentation.

4.2.2.7. *Paramétrisation et tests comparatifs de l'algorithme proposé*

Afin de vérifier la validité du modèle de détection de coins présenté, nous avons testé l'algorithme sur différents types d'images. Dans un souci d'objectivité, nous avons par ailleurs comparé les résultats à ceux obtenus par d'autres détecteurs.

Pour juger la performance des détecteurs de coins les uns par rapport aux autres, nous avons fixé quelques critères. Nous avons notamment observé la précision de détection, la vitesse de calcul, le nombre de points détectés, les paramètres à ajuster, la stabilité par rapport à une rotation ou un changement de point de vue... Ce dernier critère est très important dans certaines applications, et plus particulièrement lorsqu'on cherche à mettre en correspondance des points caractéristique dans une série d'images d'une même scène.

Les tests de comparaison des 3 algorithmes ont été réalisés sur différents types d'images :

1. Des images synthétiques qui représentent des scènes simples et ont une complexité moins importante que les images du 2ème type

2. Les images en conditions réelles sont relativement complexes d'un point de vue détection de coin.

4.2.2.7.1. Images de synthèses

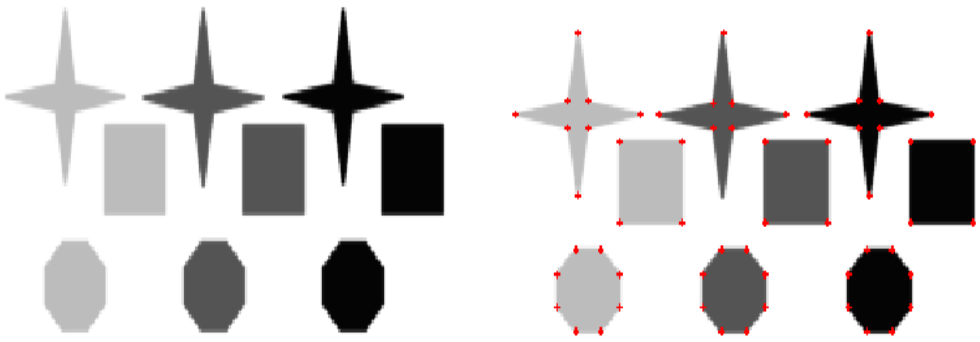


Figure 4.21. *Image synthétique de base et la solution de référence (60 coins).*

Pour réaliser une comparaison objective, l'image est soumise à 3 détecteurs (Harris, Laplace, approche proposé, SIFT). Parmi tous les critères possibles, 2 ont été retenus dans le cadre de notre travail: le nombre de coins correctement détectés (des détections multiples peuvent éventuellement se produire) et l'erreur de localisation. L'expérience a été réalisée sous 4 conditions différentes: sur l'image brute, l'image bruité (bruit uniforme), l'image avec flou et enfin une image pivotée. Les paramètres ont été ajustés pour obtenir les meilleurs résultats possibles.

a. Tests sur l'image brute

Les paramètres ci-dessous réfèrent à la figure 4.22.

Paramètres - Image brute		
Détecteur proposé	**Harris**	**SIFT** (Octave = 4)
halfPsgMaskSize = 5	windowsize = 5	contrastThresh = 0.03
corrAngleThresh =0.5	numPoints = 60	curvatureThresh = 10
noiseGradSizeThresh = 20.0	Sigma = 1.5	Sigma = 1.5
corrApparenceThresh = 0		

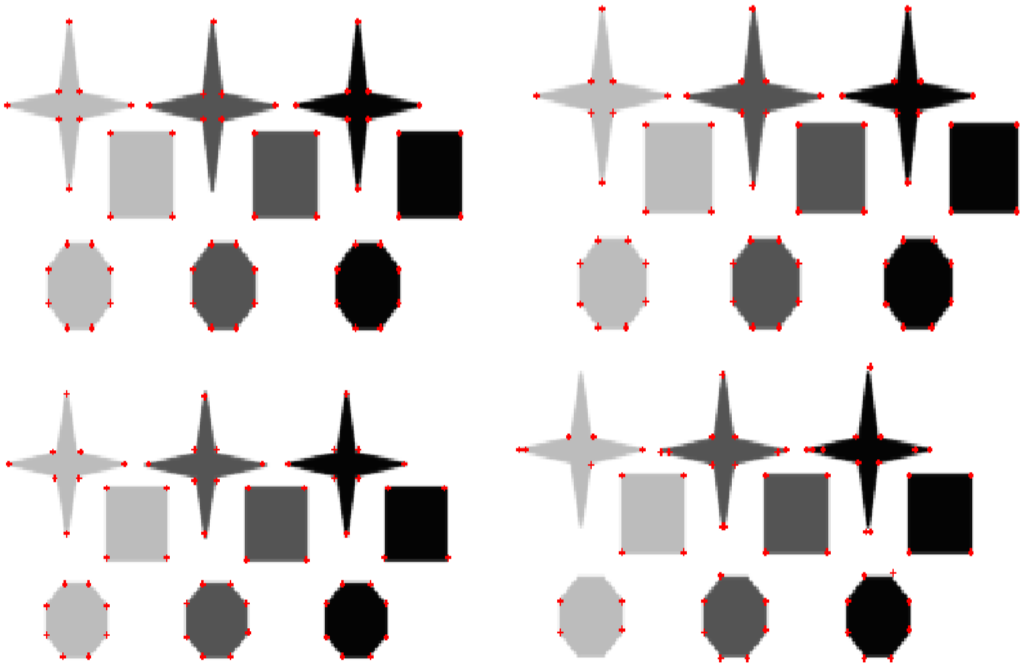


Figure 4.22. *Tests sur images brutes - Dans l'ordre : Solution de référence (60 coins réels), Détecteur proposé, SIFT et Harris.*

Erreurs mesurées - Image brute			
	Détecteur proposé	**Harris**	**SIFT**
Nombre de coins corrects	59	51	60
Erreurs de localisation	0.138	0.597	0.061

La détection se passe globalement bien pour les trois détecteurs avec une précision relativement bonne. Le détecteur proposé semble toutefois donner de meilleurs résultats en précisions et nombre de points correctement détectés.

b. Tests sur l'image bruitée

Les paramètres ci-dessous réfèrent à la figure 4.23.

Paramètres - Image bruitée avec un bruit additif Gaussien (σ = 5%)		
Détecteur proposé	**Harris**	**SIFT** (Octave = 4)
halfPsgMaskSize = 10	windowsize = 5	contrastThresh = 0.03
corrAngleThresh = 1	numPoints = 60	curvatureThresh = 10
noiseGradSizeThresh = 20.0	Sigma = 1.5	Sigma = 1.5
corrApparenceThresh = 0		

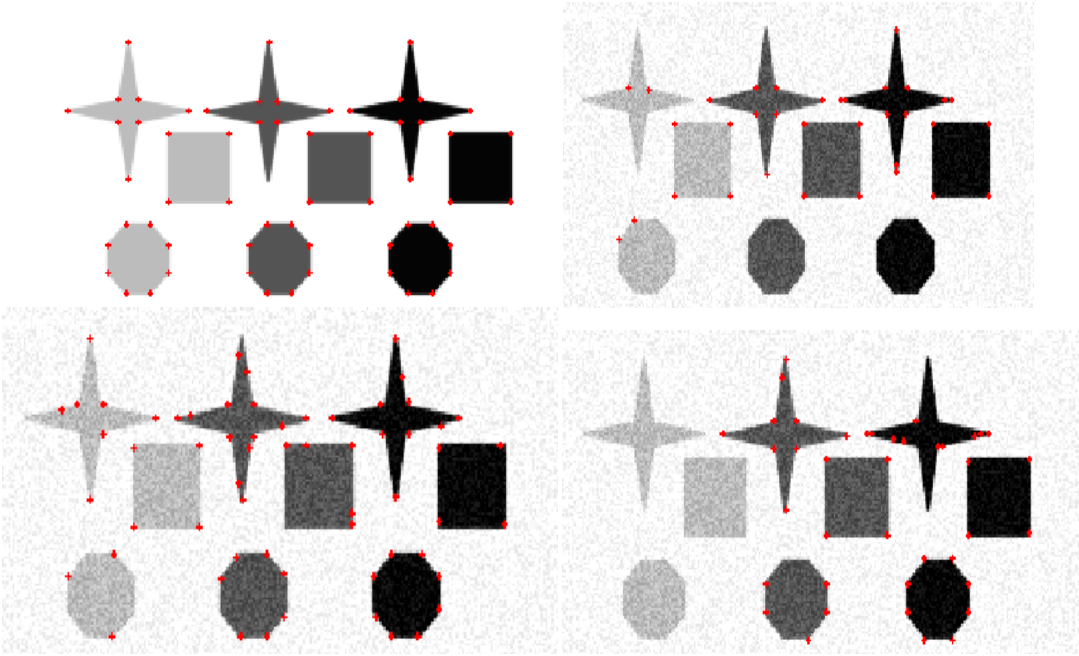


Figure 4.23. *Tests sur image bruitée - Dans l'ordre : Solution de référence, Détecteur proposé, SIFT et Harris.*

Erreurs mesurées - Image bruitée			
	Détecteur proposé	**Harris**	**SIFT**
Nombre de coins corrects	49	32	59
Erreurs de localisation	0.169	0.666	0.35

Dans ces conditions, la détection est plus difficile. On a tendance à penser qu'il faut augmenter le seuil du gradient afin de ne pas détecter tous les pixels de bruit comme des coins mais au prix de la perte de coins. Ceci est vérifié pour SIFT; cependant il donne les meilleurs résultats en termes de nombre de coins corrects.

Le seuil du gradient pour le détecteur proposé n'a pas été modifié mais il a fallu augmenter la taille masque (voisinage $\Omega(X)$). La précision semble être relativement robuste pour ce type de bruit mais il y a une perte effective de coins. Bien que Harris perde la moitié des coins, la précision semble assez bonne.

c. Tests sur l'image avec flou (flou gaussien)

Les paramètres ci-dessous réfèrent à la figure 4.24.

Paramètres - Image avec flou		
Détecteur proposé	**Harris**	**SIFT** (Octave = 4)
halfPsgMaskSize = 10	windowsize = 5	contrastThresh = 0.03
corrAngleThresh = 0.5	numPoints = 60	curvatureThresh = 10
noiseGradSizeThresh = 10.0	Sigma=1.5	Sigma = 1.5
corrApparenceThresh = 0		

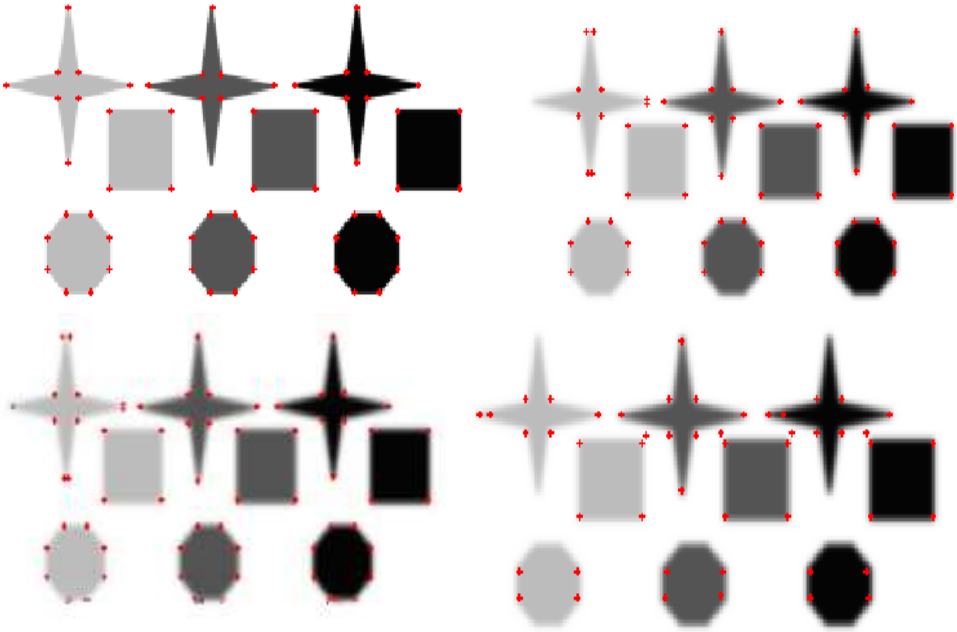


Figure 4.24. *Tests sur image avec flou - Dans l'ordre : Solution de référence, Détecteur proposé, SIFT et Harris.*

Erreurs mesurées - Image avec flou			
	Détecteur proposé	**Harris**	**SIFT**
Nombre de coins corrects	58	24	58
Erreurs de localisation	0.635	2.216	0.491

Le détecteur de Harris donne de très mauvais résultats tant en précision qu'en nombre de coin correct. Encore une fois le détecteur proposé et celui de SIFT donnent de bons résultats en termes de précision et nombre de coins.

d. Tests sur une image pivotée

Nous avons pris les mêmes paramètres que dans le cas de l'image brute afin d'observer la stabilité et l'invariance des détecteurs pour une simple rotation de 45°. L'angle de 45° n'est pas la pire condition mais il permet déjà d'observer des erreurs. Il pourrait aussi être intéressant d'étudier la stabilité vis-à-vis de la rotation pour différentes méthodes d'interpolation. Les paramètres ci-dessous réfèrent à la figure 4.25.

Paramètres - Rotation de l'image		
Détecteur proposé	**Harris**	**SIFT** (Octave = 4)
halfPsgMaskSize = 5	windowsize = 5	contrastThresh = 0.03
corrAngleThresh = 0.5	numPoints = 60	curvatureThresh = 10
noiseGradSizeThresh = 20.0	Sigma=1.5	Sigma = 1.5
corrApparenceThresh = 0		

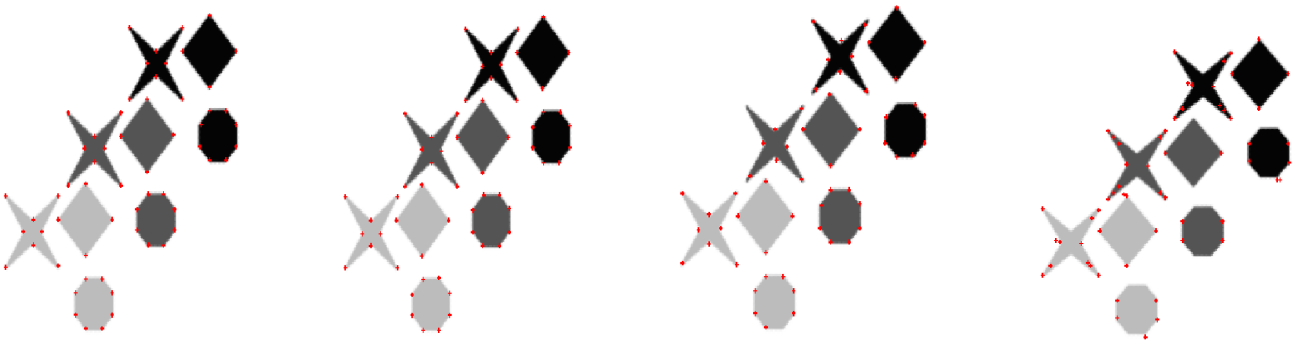


Figure 4.25. *Tests sur image avec rotation - Dans l'ordre : Solution de référence, Détecteur proposé, SIFT et Harris.*

Erreurs mesurées - Rotation			
	Détecteur proposé	**Harris**	**SIFT**
Nombre de coins corrects	60	42	55
Erreurs de localisation	0.418	0.81	0.62

Pour finir avec ce dernier test, le détecteur proposé l'emporte sur tous les points avec une erreur de localisation minimale. SIFT trouve bien plus de coins corrects et une erreur de localisation plus au moins importante par rapport à Harris.
Les résultats des tests sont présentés sous forme de graphiques ci-dessous.

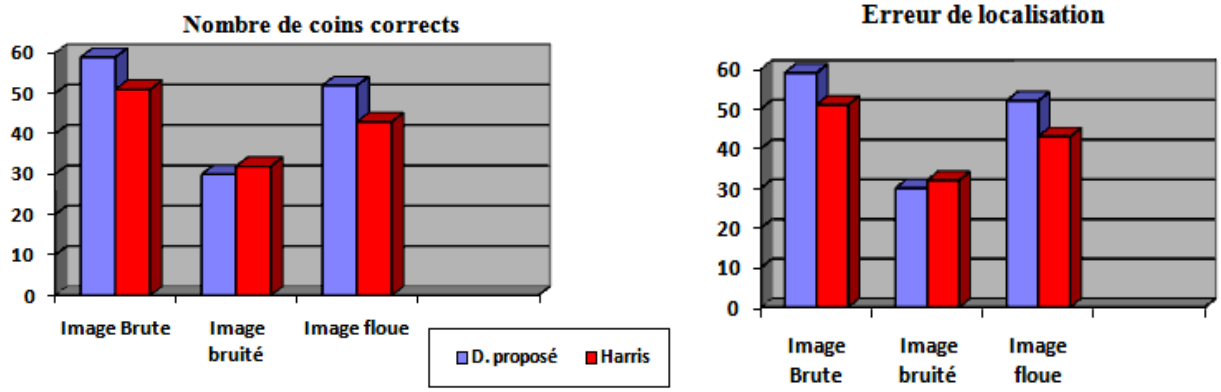


Figure 4.26. *Résultats des tests présentés sous forme d'histogrammes.*

D'après les quelques expériences réalisées précédemment, il est possible de tirer quelques conclusions. La nouvelle approche proposée donne globalement de meilleurs résultats que les deux autres. Le coût en temps calcul est en contrepartie plus important. Le détecteur de Harris est plus simple à utiliser et à implémenter car le nombre de paramètres est faible. Celui proposé demande en effet le réglage au minimum de 4 paramètres. On peut même affiner le résultat à l'aide de paramètres supplémentaires, portant leur nombre total à 10. Comme le détecteur proposé semble démontrer une grande stabilité, il est possible de prérégler les paramètres pour un certain type d'image à l'aide d'une image de référence. Ceci est un grand avantage lorsqu'on souhaite apparier des points caractéristiques dans une séquence d'images. Les résultats des tests sont présentés sous forme de graphiques ci-dessous.

4.2.2.7.2. Images réelles

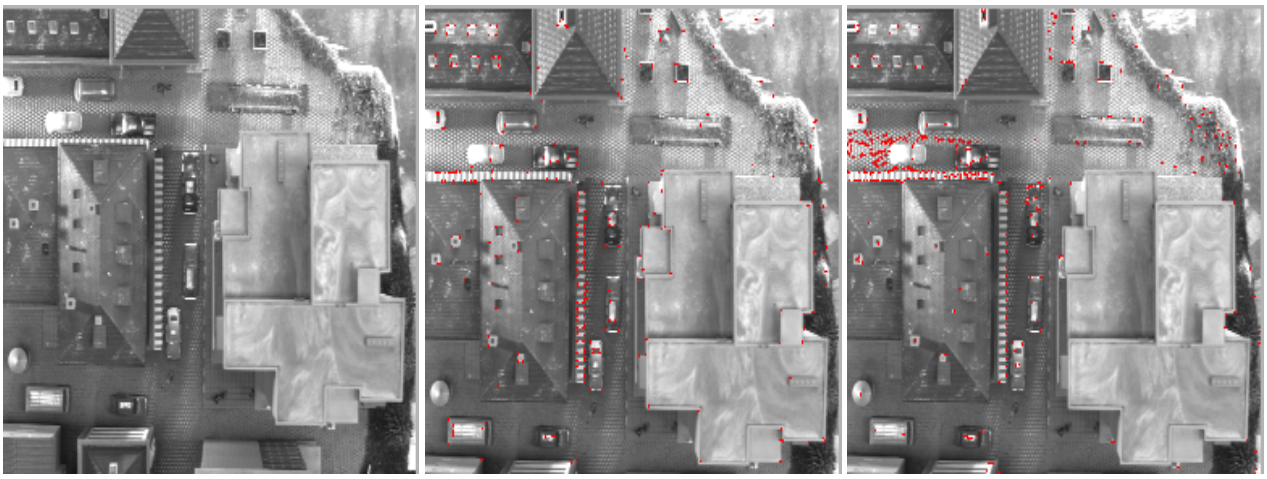
Les différents détecteurs ont été étudiés dans la section précédente avec une méthode rigoureuse à l'aide d'une image synthétique. Cependant, l'intérêt de telles images est très limité et une comparaison sur des images réelles est indispensable.

a. Test sur une image aérienne

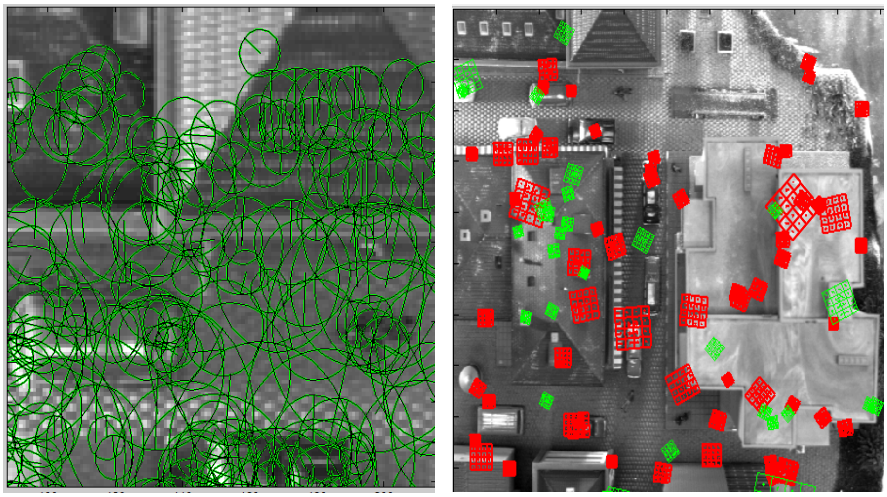
Commençons tout d'abord avec une image aérienne. La figures 4.27 montrent les résultats obtenus avec les 3 détecteurs (Détecteur proposé, Harris, SIFT). Bien que l'image ne soit pas trop complexe, le détecteur Harris a déjà des difficultés à détecter correctement les coins. Les coins de bâtiments sont particulièrement bien détectés par le détecteur proposé. De plus, il détecte un nombre très limité de faux coins.

Les paramètres utilisés sont présentés ci-dessous :

Paramètres – Image aérienne		
Détecteur proposé	**Harris**	**SIFT** (Octave = 4)
halfPsgMaskSize = 10	windowsize = 11	contrastThresh = 0.03
corrAngleThresh = 1	numPoints = 500	curvatureThresh = 10
noiseGradSizeThresh = 25.0	Sigma=1.5	Sigma = 1.5
corrApparenceThresh = 7		



(a) **(b)** **(c)**



(d)

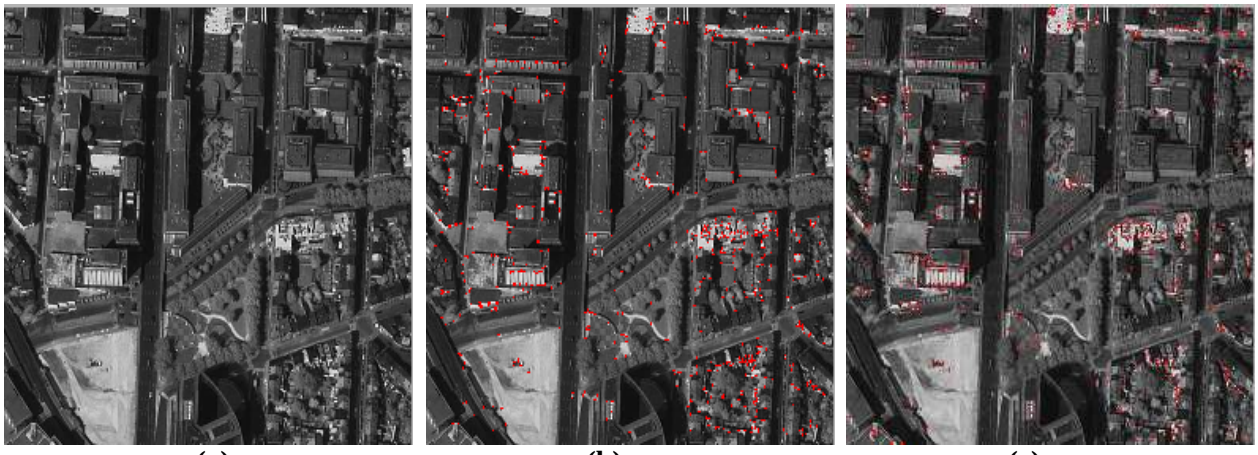
Figure 4.27. *Tests sur image aérienne en conditions idéales-(a) Image brute, (b) Détecteur proposé,(c) Harris, (d) SIFT avec ses descripteurs.*

b. Test sur une image satellitaire

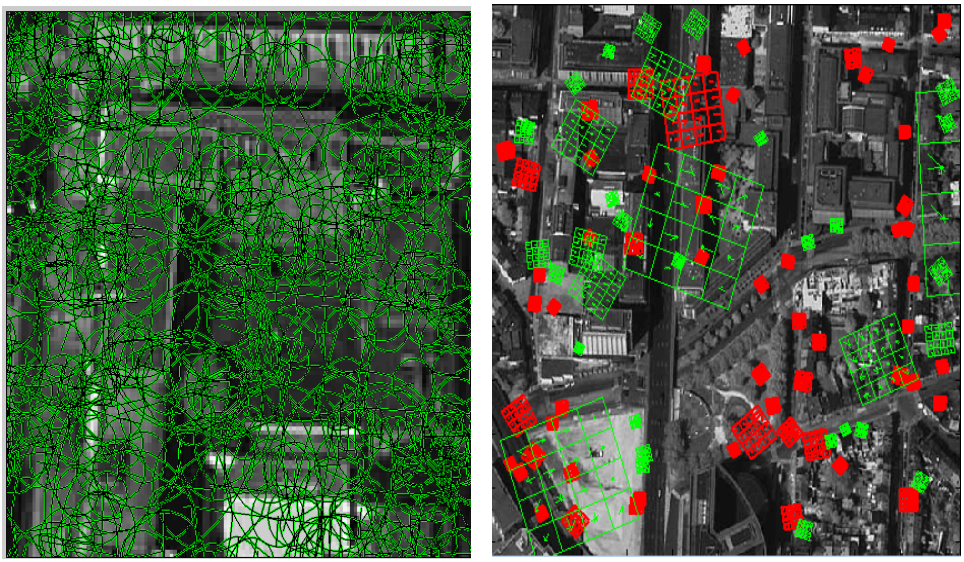
Les tests ont été portés de la même manière sur une image satellite de Bruxelles. On constate globalement de nouveau la même chose : bonne précision meilleure détection par le détecteur proposé et celui de SIFT.

Les paramètres utilisés sont présentés ci-dessous :

Paramètres – Image satellite de Bruxelles		
Détecteur proposé	**Harris**	**SIFT** (Octave = 4)
halfPsgMaskSize = 10	windowsize = 11	contrastThresh = 0.03
corrAngleThresh = 1	numPoints = 500	curvatureThresh = 10
noiseGradSizeThresh = 18.0	Sigma=1.5	Sigma = 1.5
corrApparenceThresh = 6		



(a) **(b)** **(c)**



(d)

Figure 4.28. *Tests sur une image satellite de Bruxelles en conditions idéales-(a) Image brute, (b) Détecteur proposé,(c) Harris, (d) SIFT avec ses descripteurs.*

4.2.2.8. Discussion

Le détecteur proposé possède de bonnes propriétés de stabilités par rapport à d'autres détecteurs de coins. Une bonne précision est obtenue grâce au calcul de la probabilité $P_{SG}(X)$. Le prix à payer pour une précision de la détection est un temps de calcul plus important. Nous avons vu que ce détecteur est relativement bien adapté aux images en milieu urbain pour la réalisation de modèle numérique grâce à sa grande précision et stabilité obtenus dans le cas des changements de point de vue. Enfin un inconvénient majeur du détecteur est le nombre de paramètres à régler notamment si nous ne sommes pas en mesure de prédire la nature des images.

4.3. Fusion d'images "Blending"

Dans plusieurs applications, les images originales sont très dissimilaires qu'une coupure optimale ne suffit pas d'éliminer les différences entre les images. Donc, une méthode de fusionnement est nécessaire surtout pour éliminer la différence de luminosité entre les images recalées, et compenser d'autres erreurs d'alignement.

4.3.1. Fusion "Blending" par pondération

Quand on parle de fusionnement, ou blending, la première idée qui vient à l'esprit, est de prendre simplement la moyenne entre les intensités des deux images dans la zone de chevauchement. Ceci peut donner un bon résultat, mais si les images contiennent des objets mobiles, ou bien si les paramètres de la fonction de transfert ne sont pas très précis, on risque d'avoir un effet de flou indésirable et très visible. Donc, pour éliminer cet effet, on prend en considération la distance entre les pixels et les bordures des deux images.

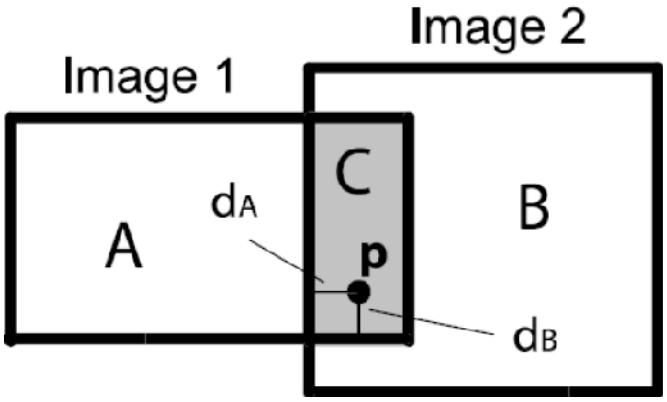


Figure 4.29. *Fusion d'images (Blending) basée sur les distances.*

Comme illustré dans la Figure 4.29, si on prend les poids des pixels dans la zone C pour les images 1 et 2 inversement proportionnels aux distances des pixels de la plus proche bordure des deux images, nous obtiendrons un fusionnement continu des intensités. Par exemple, si la distance du pixel **p** au plus proche pixel de A est d_A, est la plus petite distance à B est d_B, l'intensité au pixel **p** sera égale à :

$$I_p = \frac{I_p^A d_B + I_p^B d_A}{d_A + d_B} \tag{4.20}$$

Où I_p^A et I_p^B sont les intensités de l'image 1 et 2 au point p, respectivement.

Les distances d_A et d_B sont calculées par la transformée de distance des images, représentées par leurs bordures.

4.3.2. Fusion "Blending" par pyramides Laplaciennes (multi-résolution)

Cette approche a été développée par Burt et Adelson, et consiste en trois étapes principales:

1. Chaque image est convertie en une pyramide passe-haut, ou laplacienne, qui implique lisser chaque niveau de l'image par un filtre gaussien et redimensionner l'image lissée par un facteur de ½ (sous-échantillonner), puis, soustraire l'image construite (passe-bas) de l'image originale. Les pixels invalides et ceux des bordures sont remplis par les valeurs voisines pour que ce processus soit bien défini (voir figure 4.30).

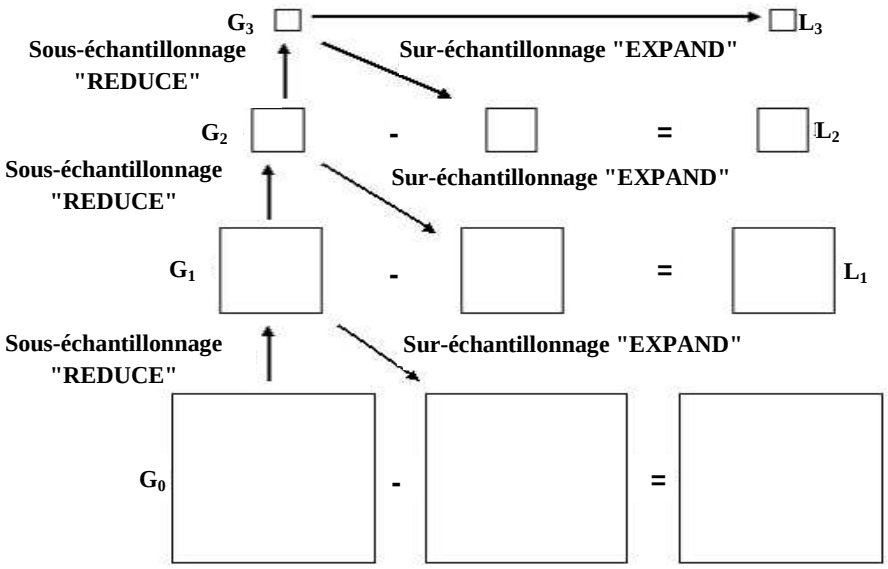


Figure 4.30. *Différentes étapes de génération de la pyramide Laplacienne.*

Construction de la pyramide gaussienne:

Soit *I*(*x,y*) l'image originale, la pyramide gaussienne sur *I* est définie comme suit:

$$\begin{cases} G_0(x,y) = I \\ G_{i+1}(x,y) = REDUCE\big(G_i(x,y)\big) \end{cases}$$

La fonction *REDUCE* est effectuée en convoluant l'image avec un filtre passe-bas gaussien. Le masque du filtre est conçu tels que le pixel central obtient plus de poids que ses voisins et les limites restantes sont choisis de sorte que leur somme est égale à 1. Le noyau gaussien est donné par l'équation 4.21:

$$w(r,c) = w(r)w(c) \qquad (4.21)$$

Où, $w(r) = \left[\frac{1}{4} - \frac{a}{2} \quad , \quad \frac{1}{4} \quad , \quad a \quad , \quad \frac{1}{4} \quad , \quad \frac{1}{4} - \frac{a}{2}\right],\ a \in [0.3 - 0.6]$

Construction de la pyramide laplacienne:

La pyramide laplacienne est construite comme suit:

$$L_i(x,y) = G_i(x,y) - EXPAND\big(G_{i+1}(x,y)\big)$$

La fonction *EXPAND* est définie par l'équation 4.22:

$$G_{i+1}(x,y) = 4\sum_{m=-2}^{2}\sum_{n=-2}^{2} w(m,n)G_i\left(\frac{x-m}{2}, \frac{y-n}{2}\right) \qquad (4.22)$$

2. L'image *masque*, qui spécifie la partie correspondante de chaque image qui intervient dans le résultat, est convertie en une pyramide Gaussien (les mêmes étapes que la pyramide laplacienne, sauf la dernière étape de soustraction). Ces masques lissés et sous-échantillonnés vont servir de poids pour fusionner les différents niveaux dans les pyramides des images.

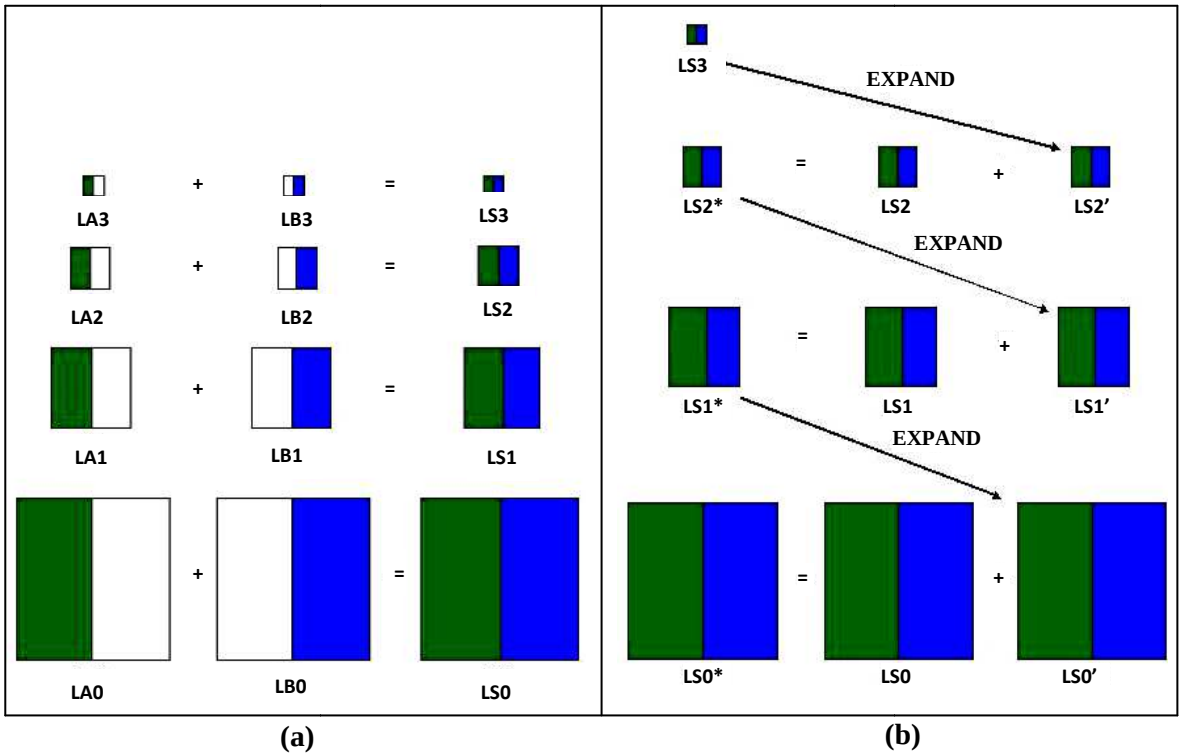


Figure 4.31. *Pyramide finale de reconstruction. (a) Masques de reconstruction des images A et B (LA_n: pyramide gauche, LB_n: pyramide droite), (b) pyramide de reconstruction de l'image mosaïquée finale.*

3. Finalement, l'image composée est construite par l'interpolation et la sommation de toutes les pyramides (images passe-bas), ce qui donne une transition douce entre les deux images.

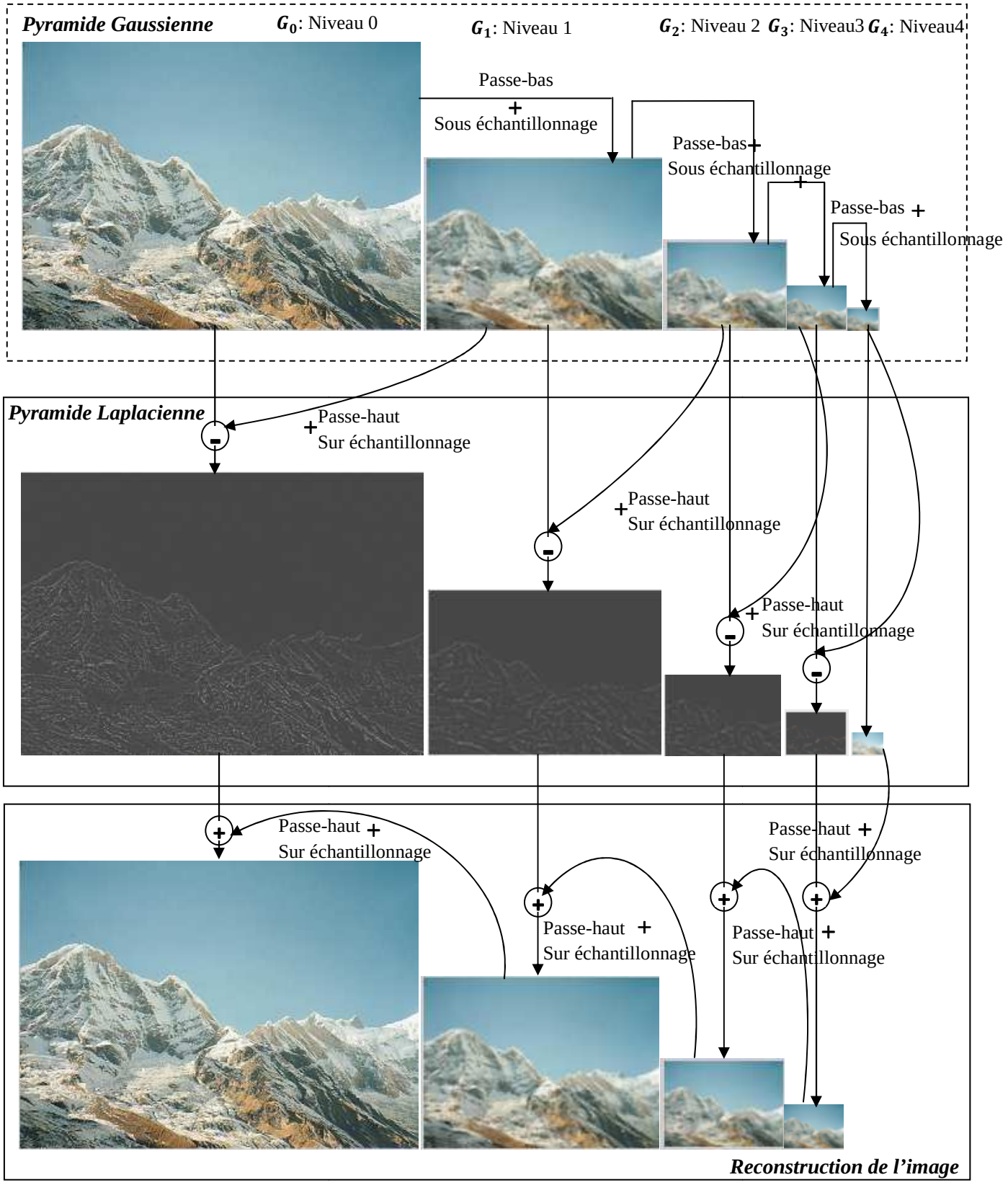


Figure 4.32. *Génération et reconstruction de la pyramide laplacienne.*

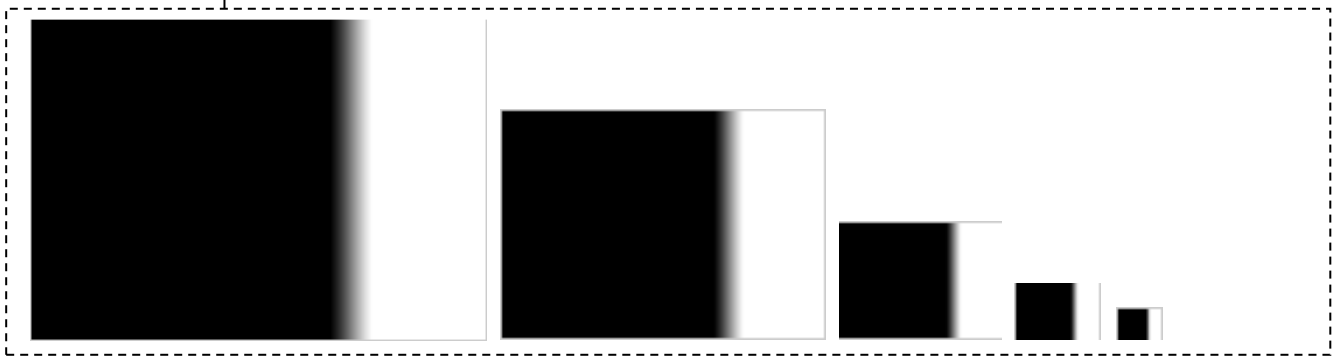


Figure 4.33. *Pyramide des différents masques de reconstruction de la figure 4.32.*

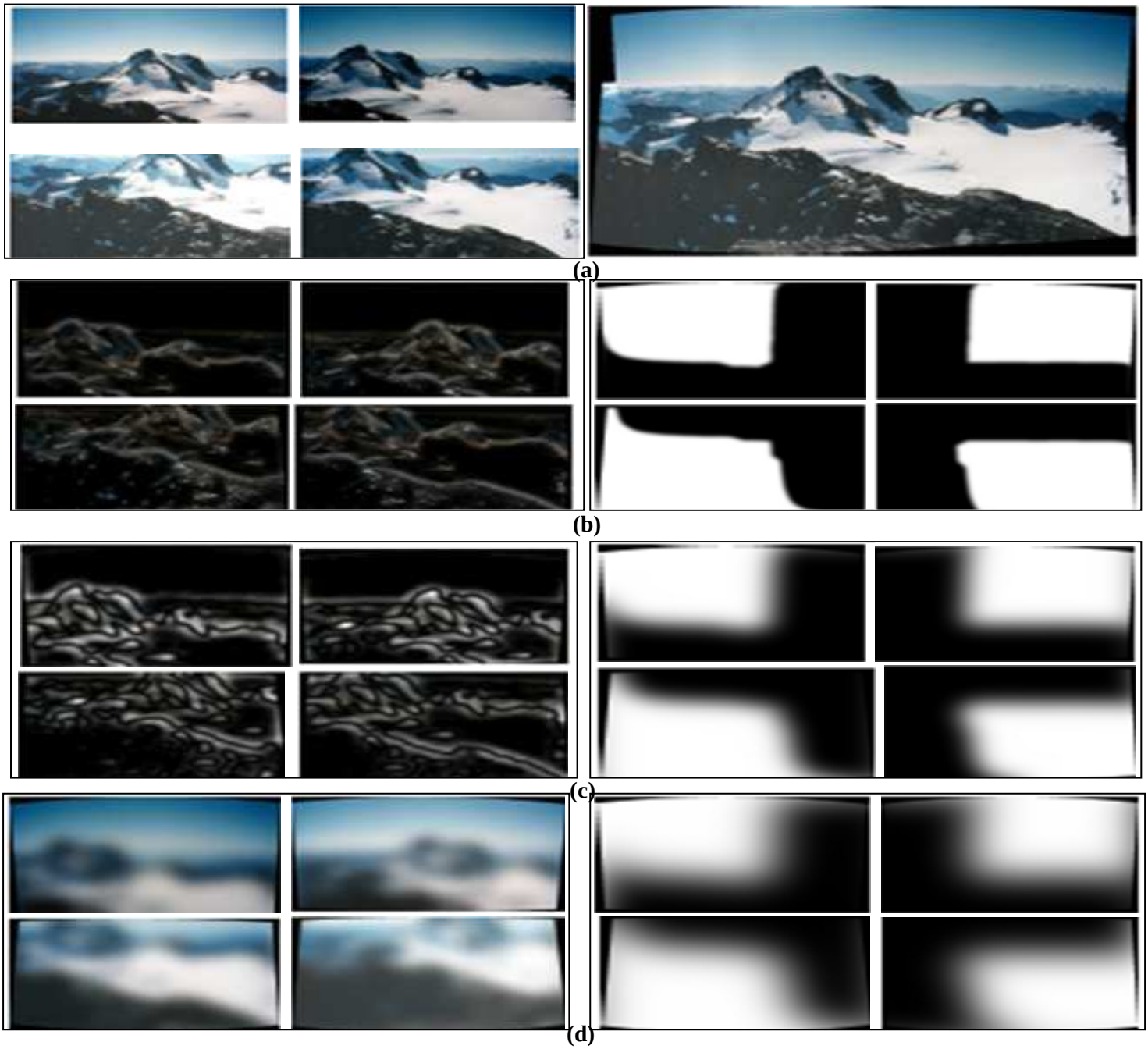


Figure 4.34. *Fusion d'images (blending) à plusieurs bandes. (a). Images originales et l'image finale mosaïquée. (b). images laplaciennes + les masques sur la bande 1 (échelle 0 à σ). (c). images laplaciennes + les masques sur la bande 2 (échelle σ à 2σ). (d). images laplaciennes + les masques sur la bande 3 (échelle $<2\sigma$).*

4.4. Résultats des différentes étapes de la méthode de mosaïquage proposée

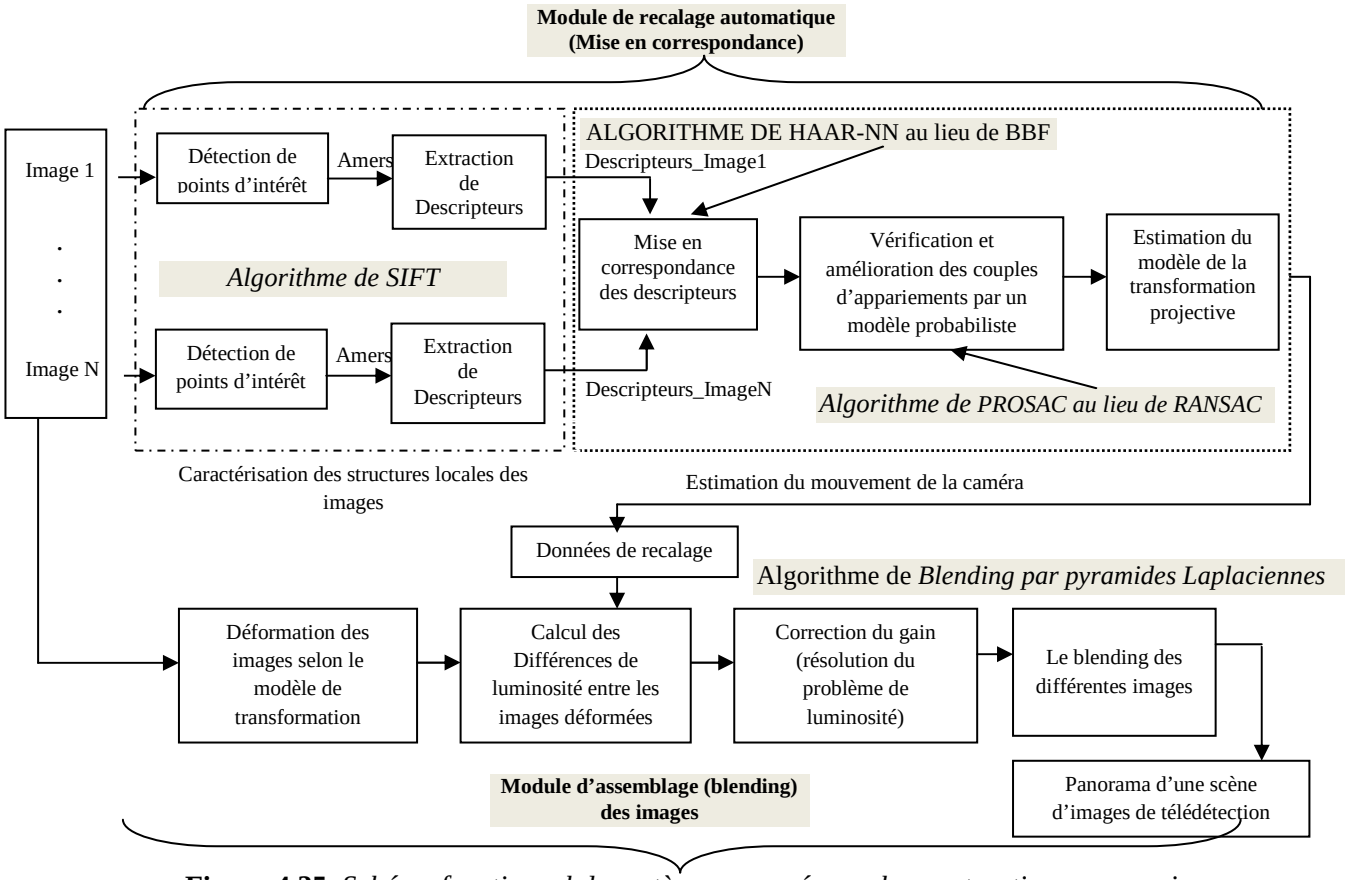


Figure 4.35. *Schéma fonctionnel du système proposé pour la construction panoramique des images de télédétection.*

Pour visualiser les résultats de chacune des étapes de notre méthode, nous allons travailler sur un échantillon typique, qui représente une série d'images aérienne, satellitaires et radars, acquises à des instants différents, chacune couvre une partie de la zone visionnée avec une région commune entre elles. Ils se diffèrent également dans le degré de luminosité, qui sera un bon facteur pour bien illustrer nos résultats.

Figure 4.36. *Images de tests, « Images satellitaires d'IKONOS à haute résolution de Ponte Golden Gate, San Francisco acquises à deux instants séparés »*

4.4.1. Détection et mise en correspondance de primitives

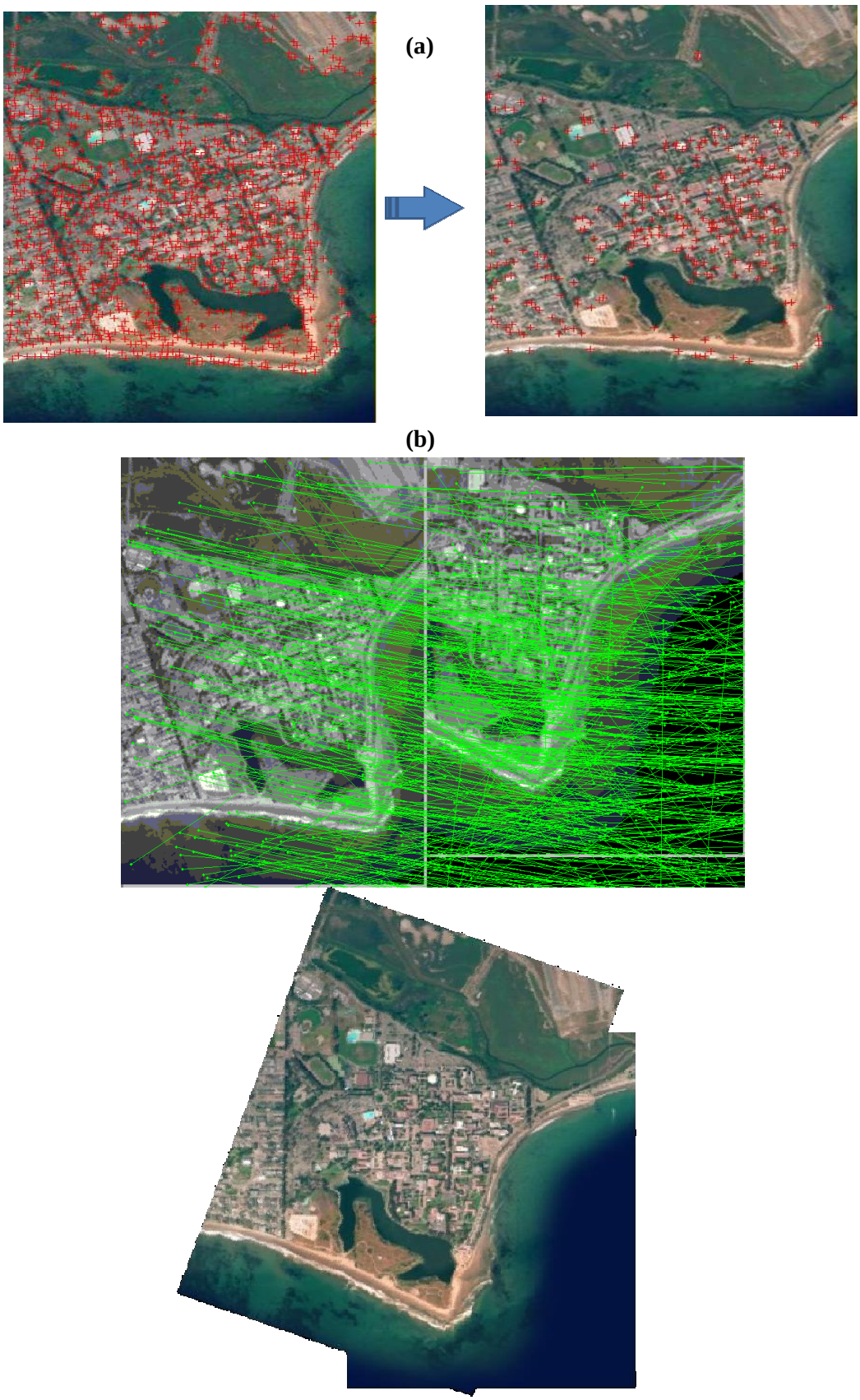
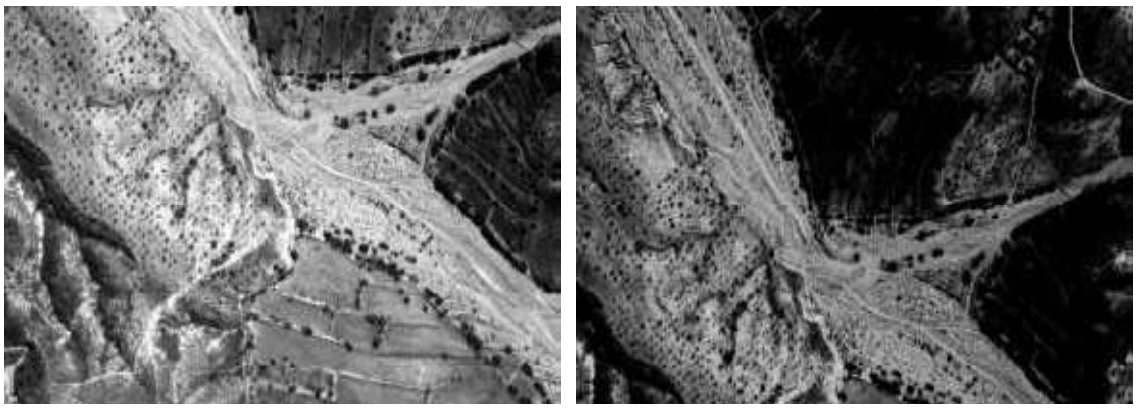


Figure 4.37. *Différentes étapes de l'algorithme de recalage. (a) détection des primitives, (b) appariement avec un seuil de 0.5. Nombre de points détectés : Image1 : 800, Image2 : 756 ; Nombre de points en correspondance : 200.*

4.4.2. Estimation de la transformation

En utilisant les images suivantes, et en assumant que la transformation géométrique qui les relie est affine (scène plate, appareil très loin de la scène qui est le satellite), la matrice des paramètres de la transformation trouvée est la suivante :

$$T = \begin{pmatrix} 1.011 & 0.004 & 27.012 \\ 0.002 & 1.020 & -189.391 \end{pmatrix}$$



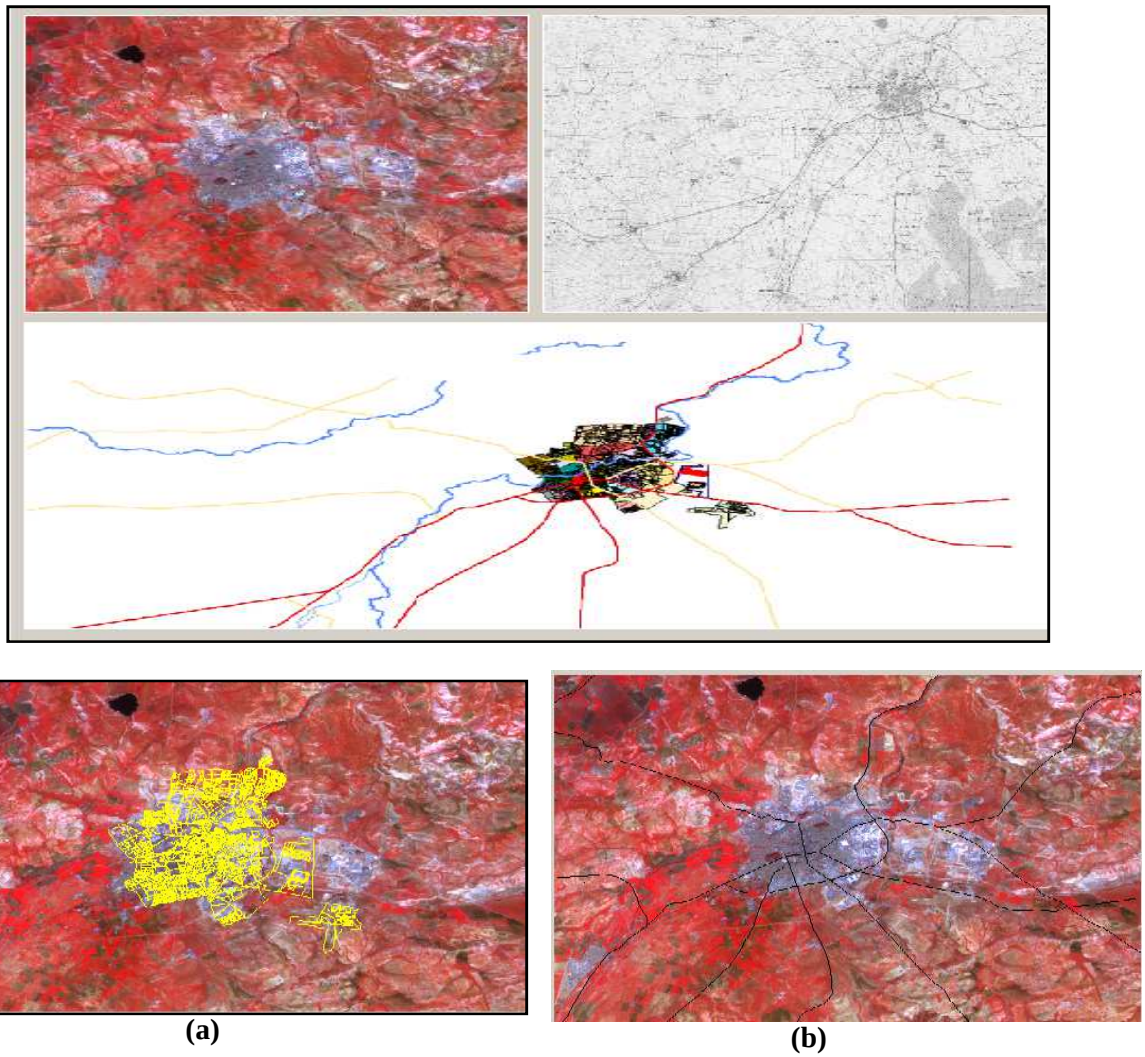
(a) **(b)**

Figure 4.38. *Résultat de fusionnement par interpolation bilinéaire après application de la fonction de transformation affine sur les deux images (a) et (b).*

a. Test de Fusionnement par distances pondérées sur des images satellitaires

Figure 4.39. *Fusionnement des deux images (a) et (b) par distances pondérées.*

b. Test de Fusionnement par distances pondérées sur une image satellitaire et cartographique



(a) **(b)**

Figure 4.40. *Fusionnement d'une image satellite et cartographiques de la ville de sidi Bel-Abbes. (a)image satellite / bâtis, (b) image satellite/ réseau routier.*

4.5. Evaluation des résultats:

Il existe plusieurs méthodes d'estimation de la performance des mosaïques, nous nous sommes intéressés plus particulièrement à celles développées par Pietro Azzari [AZZ08], pour évaluer notre algorithme de mosaïquage.

Nous avons opté pour 3 mesures de performance:

4.5.1. *Moyenne des distances entre les intensités:* elle est définie par l'équation suivante:

$$MSE = \frac{1}{M}\sum_{(x,y)} D_{xy} = \frac{1}{M}\sum_{(x,y)} (m_c(x,y) - m_T(x,y))^2 \qquad (4.23)$$

Avec $(m_c(x,y), m_T(x,y))$ représente les pixels en correspondance entre l'image I_C et l'image I_T. *M* est le nombre de pixels dans la région commune entre les deux images.

L'image I_T représente le résultat exact du recalage. Généralement, ce type d'images est généré soit par une méthode optimale, soit on prend des images de référence, on effectue une division de l'image tout en laissant une zone commune entre les deux parties, et l'algorithme du recalage est exécuté sur les images obtenus.

4.5.2. *Moyenne des distances géométriques:* elle est définie par:

$$\epsilon_{est} = \frac{1}{L}\sum_{i} \|(x_i', y_i') - (x_i, y_i)\| \qquad (4.24)$$

Avec $(x_i', y_i') = f(x_i,\ y_i)$, tel que f : est la fonction de transformation. *L* est le nombre de points en correspondances et ||.|| est la distance entre deux points.

4.5.3. *Moyenne carrée des erreurs:* généralement calculée en racine carrée, elle est définie par:

$$RMSE = \sqrt{\frac{1}{L}\sum_{i} D_i^2} = \sqrt{\frac{1}{L}\sum_{i} \left\|(x_i', y_i') - (x_i, y_i)\right\|^2} \qquad (4.25)$$

Parmi plusieurs ensembles d'images utilisés pour effectuer les tests de notre méthode nous allons présenter ci-dessous des exemples représentatifs de chaque catégorie d'images: des images satellitaires, aériennes et des paysages (images réelles). Le tableau suivant représente les valeurs RMSE et $\boldsymbol{\varepsilon}_{\text{est}}$ calculées pour chaque paire d'images.

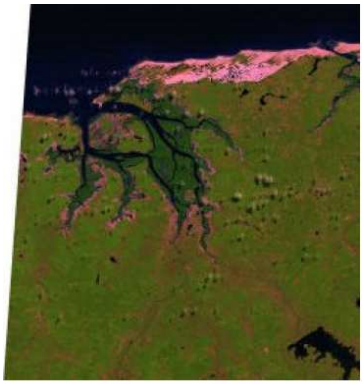
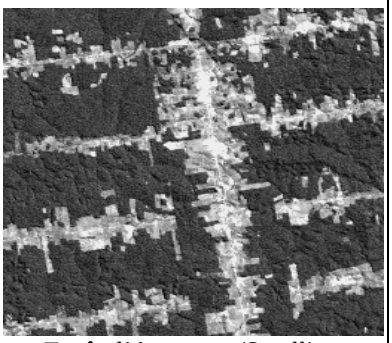
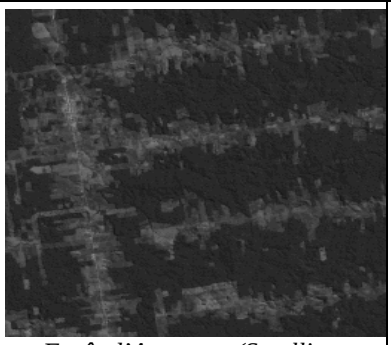
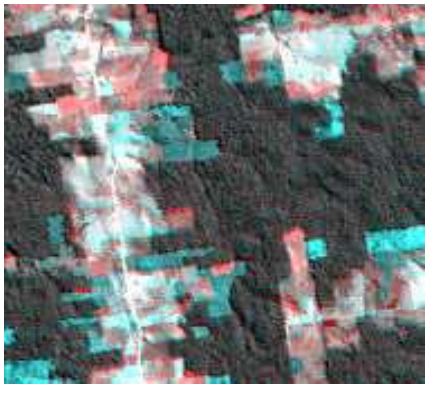
Image 1	Image 2	Résultat du recalage et de mosaïquage
Image aérienne relevée de Yao Jianchao, 2001.	Image aérienne relevée de Yao Jianchao, 2001, acquises à un instant différent de celui d'image1.	***Exemple 1***
 *Région du delta de Rio Parnaiba (Satellite Landsat TM/5 (15/07/1994))*	*Région du delta de Rio Parnaiba (Satellite Landsat TM/5 (20/07/1996))*	***Exemple 2***
 *Forêt d'Amazone (Satellite Landsat TM band 5, 07/06/1992)*	 *Forêt d'Amazone (Satellite Landsat TM band 5, 15/07/1994)*	 ***Exemple 3***

Image de Santa Barbara, California, (Satellite IKONOS)

Image de Santa Barbara, California, (Satellite IKONOS)

Exemple 4

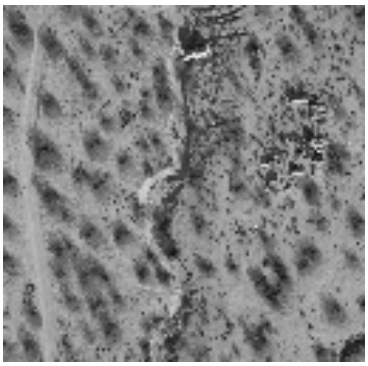


Image aérienne du désert de Mojave

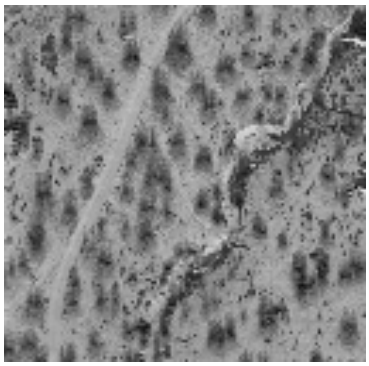
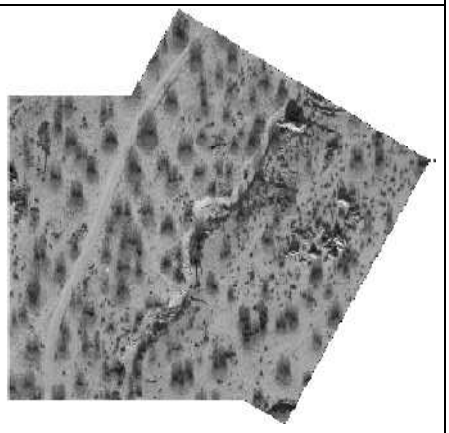


Image aérienne du désert de Mojave



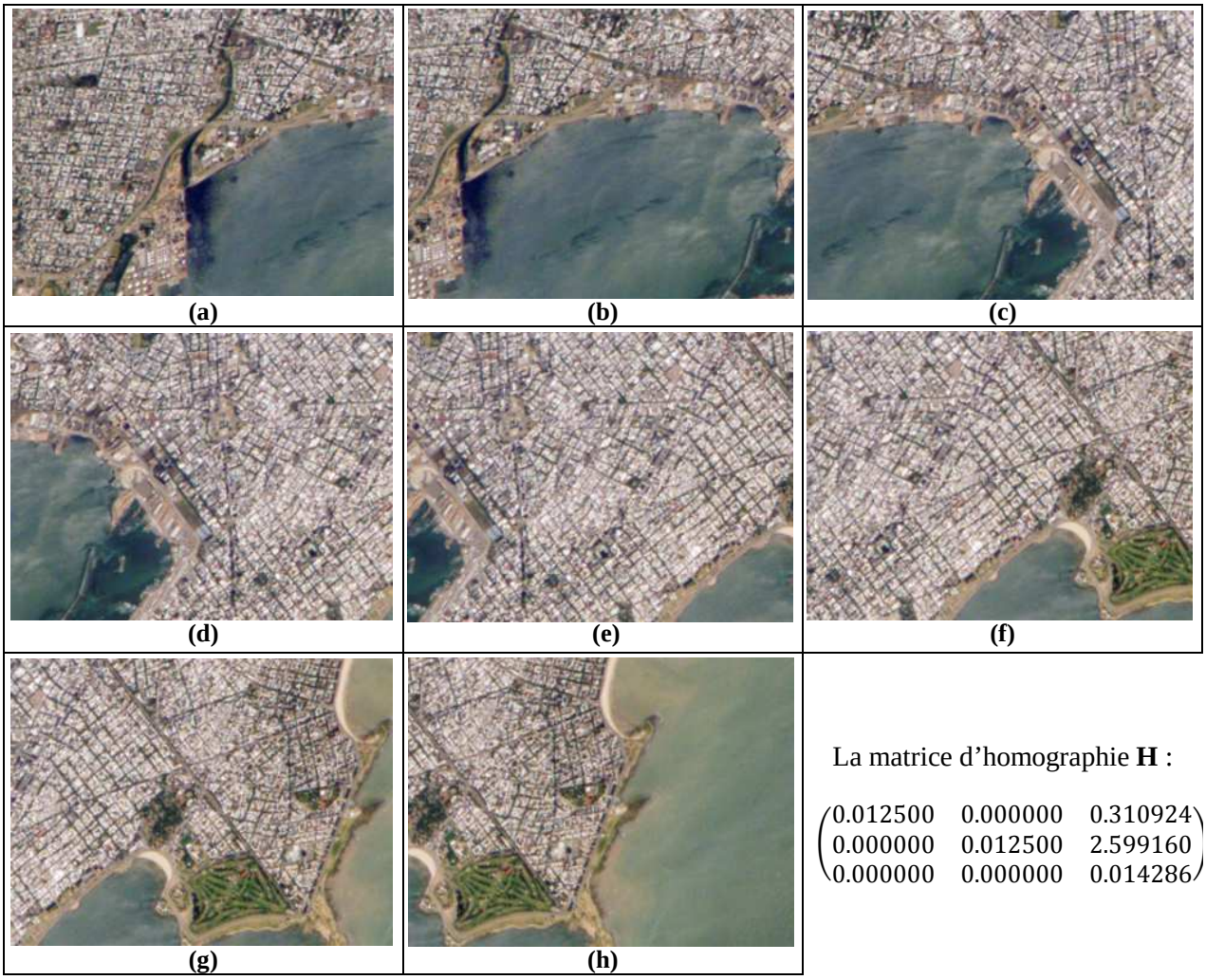
Exemple 5

	RMSE	ε_{est}
Image 1	1.00	0.86
Image 2	0.81	0.72
Image 3	1.76	1.53
Image 4	1.54	1.41
Image 5	0.28	0.18

Tableau 4.3. *Valeurs de RMSE et ϵ_{est} pour des images de tests*

4.6. Recalage et construction panoramique de N images :

4.6.1. Exemple 1: test sur des images satellitaires

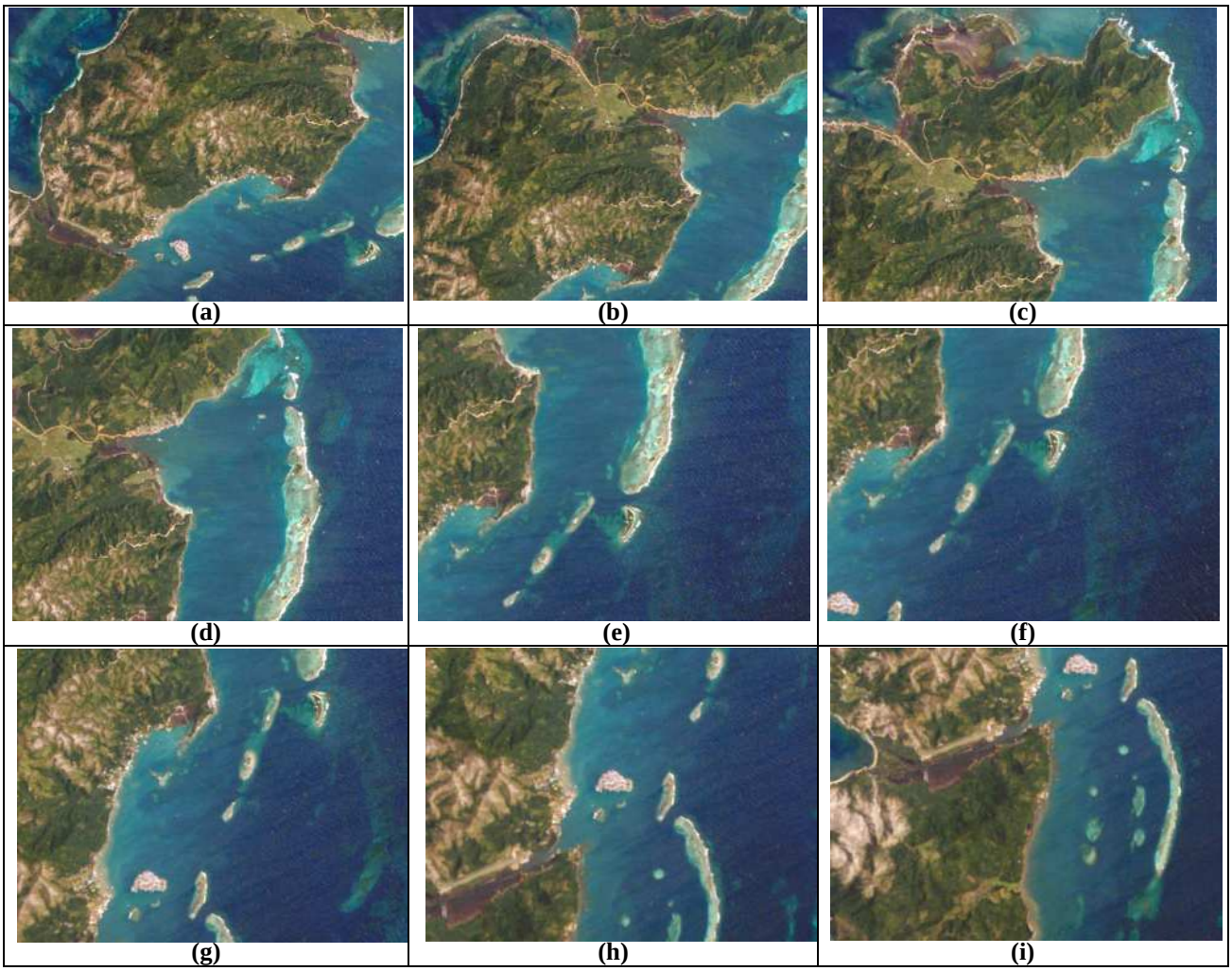


La matrice d'homographie **H** :

$$\begin{pmatrix} 0.012500 & 0.000000 & 0.310924 \\ 0.000000 & 0.012500 & 2.599160 \\ 0.000000 & 0.000000 & 0.014286 \end{pmatrix}$$

Figure 4.41. *Mosaïquage de 8 images satellitaires avec une projection plane.*

4.6.2. Exemple 2: test sur des images satellitaires

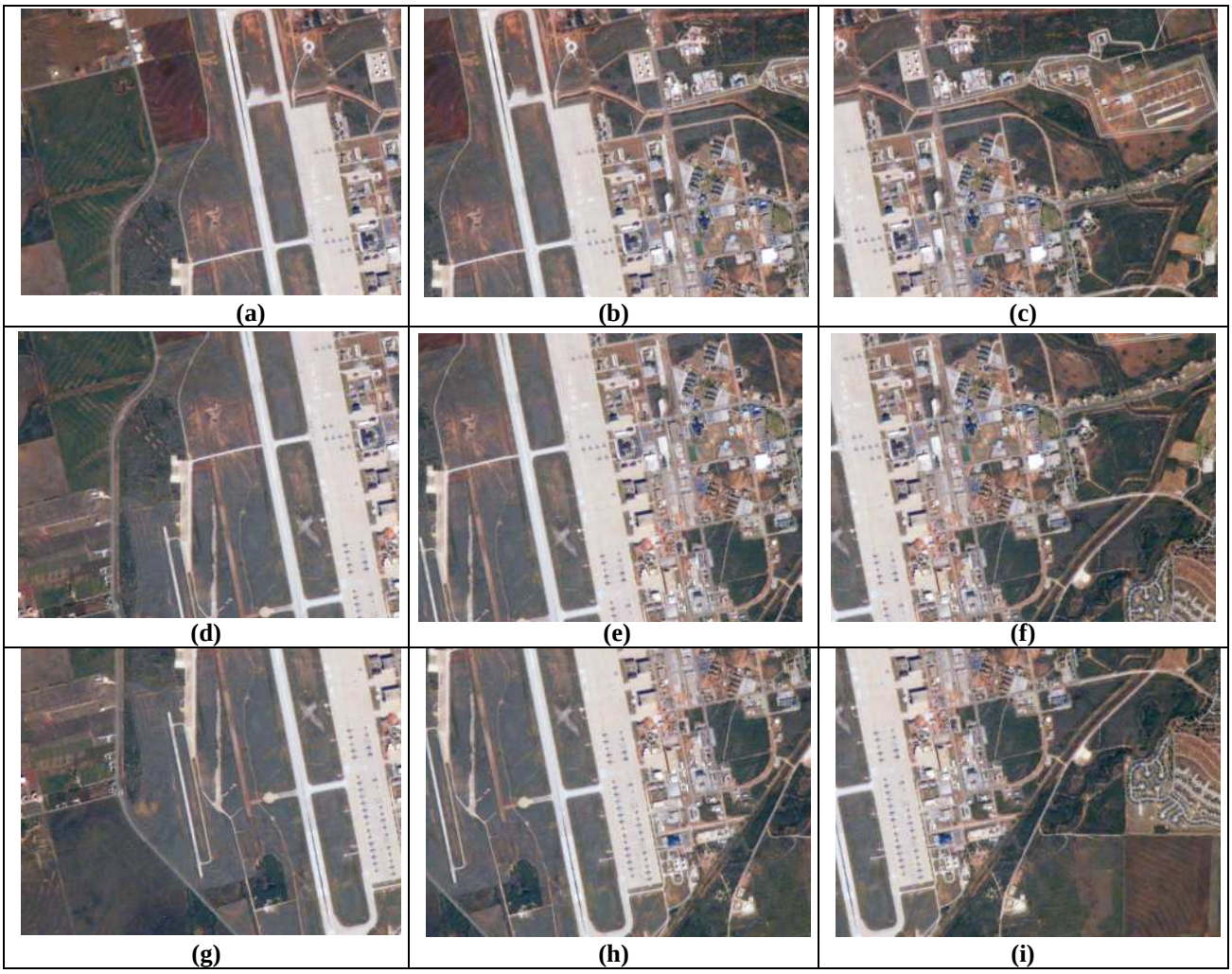


La matrice d'homographie **H** :

$$\begin{pmatrix} 0.003333 & 0.000000 & 0.771814 \\ 0.000000 & 0.003333 & 0.488440 \\ 0.000000 & 0.000000 & 0.002500 \end{pmatrix}$$

Figure 4.42. *Mosaïquage de 9 images satellitaires avec une transformation affine.*

4.6.3. Exemple 3: test sur des images aériennes

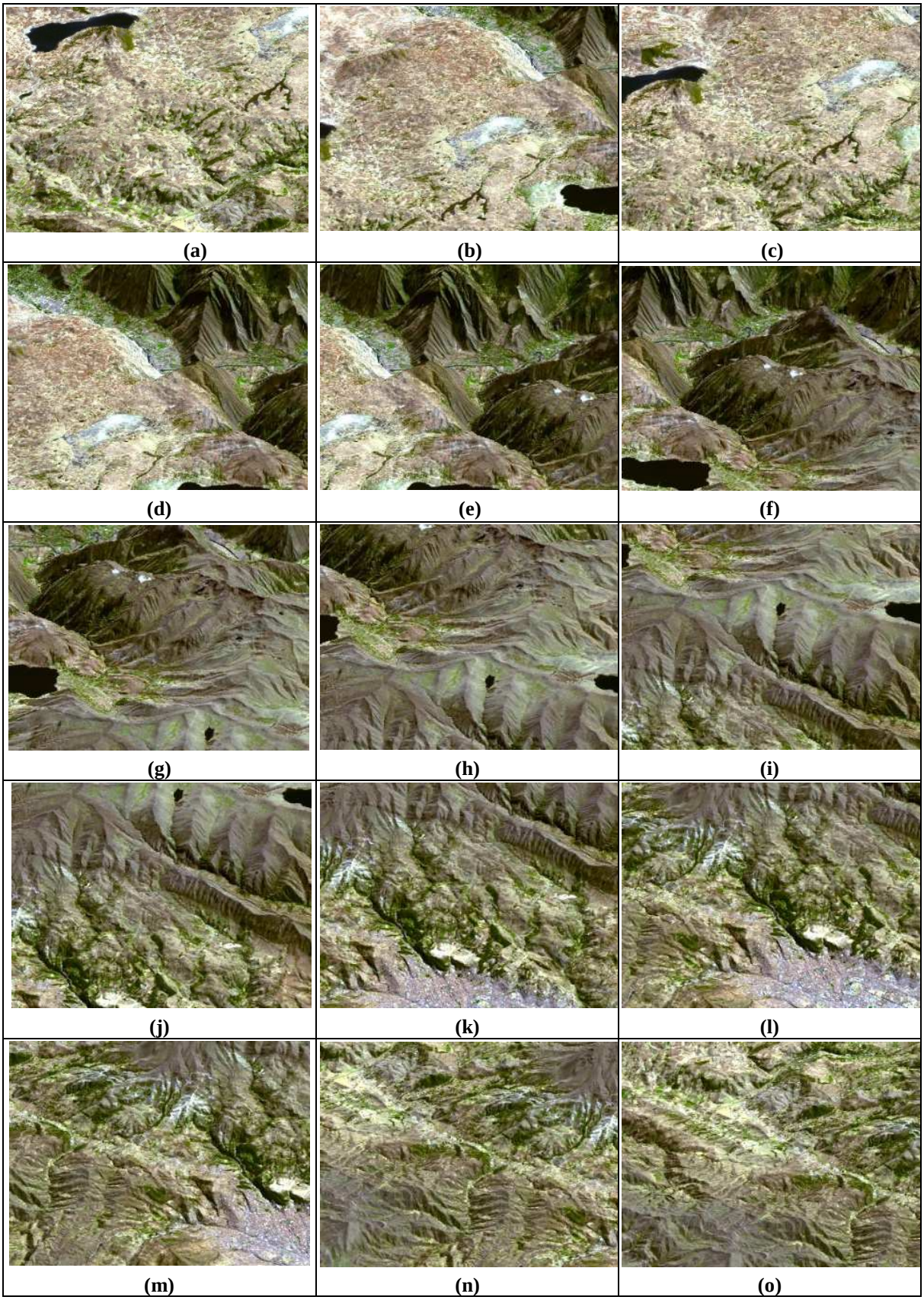


La matrice d'homographie **H** :

$$\begin{pmatrix} 0.003333 & 0.000000 & 0.051773 \\ 0.000000 & 0.003333 & 0.227660 \\ 0.000000 & 0.000000 & 0.002000 \end{pmatrix}$$

Figure 4.43. *Mosaïquage de 9 images aériennes avec une transformation affine*

4.6.4. Exemple 4: test sur des images aériennes



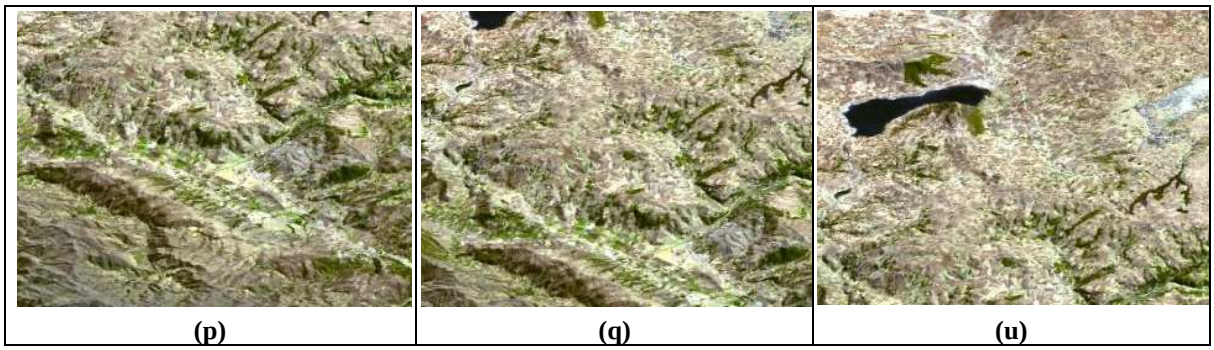
(a) (b) (c)

(d) (e) (f)

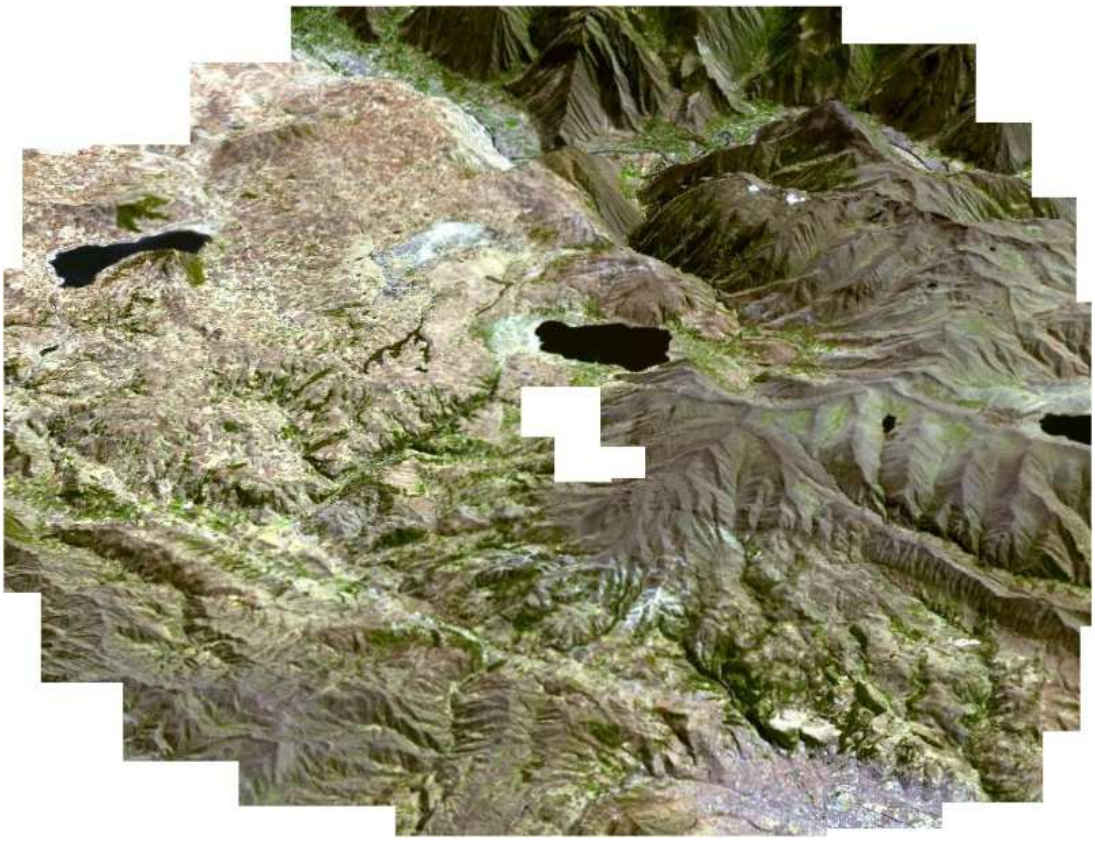
(g) (h) (i)

(j) (k) (l)

(m) (n) (o)



(p) **(q)** **(u)**



La matrice d'homographie **H** :

$$\begin{pmatrix} 0.003333 & 0.000000 & 0.263790 \\ 0.000000 & 0.003333 & 1.257692 \\ 0.000000 & 0.000000 & 0.002941 \end{pmatrix}$$

Figure 4.44. *Mosaïquage de 18 images aériennes avec une transformation projective*

.

Image	Méthode Manuelle	Méthode basée sur les contours [MED09][MED10]	Méthode basée sur SIFT classique	Méthode proposée
Exemple 1	4.72	3.105	2.307	0.012
Exemple 2	Echouée	2.98	2.05	0.14
Exemple 3	2.83	1.9	1.68	0.008
Exemple 4	Echouée	Echouée	2.82	0.32

Tableau 4.4. *Différentes valeurs d' RMSE pour les images des figures 4.41, 4.42, 4.43, 4.44*

4.7. Conclusion:

Au cours de ce chapitre, nous avons montré l'utilisation de notre méthode à travers des situations très variées et complexes.

En tout premier lieu, nous avons utilisé des données synthétiques, dont l'intérêt était de pouvoir paramétrer diverses perturbations et d'analyser le comportement de plusieurs détecteurs de points d'intérêts. Nous avons ainsi pu constater les bonnes performances de l'approche de mosaïquage proposée, même dans des conditions difficiles.

Ensuite, nous l'avons testé sur des données réelles issues de la télédétection, de modalités différentes. La performance de l'algorithme a été évaluée sur plusieurs exemples d'images avec différentes configurations et conditions. Les résultats expérimentaux montrent que globalement la performance et la précision du recalage et du mosaïquage sont relativement élevées en comparaison avec d'autres méthodes de recalage. L'algorithme proposé améliore fortement la qualité des images mosaïquée avec une précision de moins de 0.1 pixels qui a été obtenue.

La contribution apportée par cette méthode est un gain significatif pour le nombre d'appariements obtenus, en particulier en présence de motifs répétés (l'ambiguïté perceptuelle) et pour un nombre important d'images non classifiées. Notre objectif global était d'étudier les performances des algorithmes de recherche des plus proches voisins, pour les descripteurs locaux d'images, et d'en proposer de nouveaux, plus performants. Cette performance se mesure en terme de rapidité d'exécution, mais aussi, comme nous l'avons vu en terme de précision. Dans ce cadre, nous nous sommes restreint aux nuages de points SIFT car ces descripteurs locaux d'images sont très efficaces.

Enfin, cette méthode ne fait pas intervenir de paramètres difficiles à régler. L'équilibrage des contraintes géométriques et photométriques, l'augmentation de la distance entre les caméras, et l'influence des différentes distances photométriques ont ainsi été étudiés.

CONCLUSION GENERALE ET PERSPECTIVES

La construction d'une mosaïque est un sujet important dans le domaine de la vision par ordinateur. Cette importance est vue clairement dans plusieurs situations où il est nécessaire d'avoir une image panoramique composée de plusieurs images distinctes. Une telle situation peut apparaître dans le cas où la taille d'un paysage dépasse le champ visuel de la caméra. Une mosaïque est construite en de quatre étapes que nous avons abordées dans notre thèse. Ces étapes sont: la détection des points d'intérêt, la mise en correspondance, la projection des images sur un plan cylindrique ou plane et enfin la correction photométrique.

Le premier volet de notre travail de recherche a été consacré au problème de détection automatique de points d'intérêt à partir des images de télédétection ou autres. La détection de points d'intérêt (ou coins) est, au même titre que la détection de contours, un composant-clé pour de nombreux processus de traitement d'images et de vision par ordinateur. C'est en effet une étape particulièrement indispensable qui intervient de manière déterminante dans une multitude d'applications telles que le recalage d'images, la mise en correspondance, ou l'estimation de mouvement et le suivi de cibles.

Notre première contribution apportée par ce travail de thèse est d'avoir développé une nouvelle méthode rapide et fiable d'extraction de points d'intérêt [MED13]. L'enjeu du travail était d'établir un compromis des critères que doit satisfaire un détecteur de points d'intérêt. L'accent a été mis sur le problème du coût calculatoire et de fiabilité. L'évaluation comparative du détecteur développé par rapport à d'autres détecteurs bien connus a montré que le nouveau détecteur se révèle plus performant en termes des critères suivants: détection, localisation et temps de calcul.

L'objectif du deuxième volet de ce travail de recherche consistait à enrichir l'étape de mise en correspondance des points d'intérêt en proposant un nouvel algorithme automatique plus fiable et robuste car cette partie est cruciale dans le domaine de la vision par ordinateur. Une détection erronée d'une paire de points parmi plusieurs paires valides conduit à un échec complet dans la construction de la mosaïque. Cette caractéristique nous force à concevoir une méthode plus précise pour assurer la fiabilité des paires mises en correspondance. Pour cela, notre 2ème apport porte sur l'élaboration d'une méthode d'appariement invariante aux fortes déformations des images de télédétection et qui est basée sur les descripteurs SIFT. Nous avons proposé de remplacer le schéma de mise en correspondance KD-tree et BBF par un autre algorithme plus robuste et rapide basé sur la transformée en ondelettes de Haar. Ce dernier, permet d'améliorer la sélection des paires d'images et d'apparier des scènes contenant de nombreux motifs répétés, par exemple des scènes d'intérieur ou d'extérieur en milieu urbain, dans lesquelles de telles zones peuvent être observées sous des angles très différents.

Notre dernier apport concerne l'étape de filtrage des correspondances et plus précisément de remplacer la technique d'échantillonnage de l'algorithme RANSAC par celui de PROSAC *(Progressive Sample Consensus).* Ce dernier consiste à une modification du paradigme RANSAC. Plutôt que de traiter tous les appariements de la même manière, on sélectionne ainsi les échantillons sur un ensemble de correspondances les mieux appariées, ensemble qu'il s'élargit petit à petit. Cette approche est intéressante au sens qu'elle combine une recherche qu'on espère plus rapide du consensus, avec une fonction d'évaluation du consensus prenant en compte la probabilité de fausses alarmes.

Nous constatons que les étapes de la construction d'une mosaïque sont interdépendantes, dans le sens que la réussite ou l'échec d'une étape induit respectivement une amélioration ou une dégradation dans l'étape suivante. C'est pour cette raison que notre travail vise à accomplir une construction correcte d'une mosaïque en réduisant au maximum les erreurs de chaque étape.

Rappelons que l'objectif initial de notre travail de thèse était de réaliser une construction automatique et correcte d'une mosaïque à partir d'un ensemble d'images données. Nous avons atteint ce but en implémentant différentes techniques pour le choix des points d'intérêt ainsi que pour la mise en correspondance. Les résultats que nous avons présentés dans le chapitre 5 montrent la qualité des mosaïques obtenues à partir de plusieurs tests empiriques de notre système. Il serait souhaitable d'ajouter assez tests de mesures quantitatives basées sur l'utilisation d'images synthétiques réalistes. De même il serait possible d'évaluer la robustesse de notre méthode en modifiant les caractéristiques de textures.

Perspectives

Nos expérimentations nous ont permis de mettre en évidence des difficultés qui nous semblent être des problèmes ouverts pour les approches d'analyse de la structure et du mouvement. Les plus importants sont pour nous :

- Progresser dans les mesures de ressemblance photométrique, pour que deux objets semblables puissent être identifiés comme tels même dans des situations difficiles,

- Il est possible d'utiliser I-ASIFT dans une application d'estimation de mouvements à partir de vues éloignées, par exemple pour la construction de cartes 3D non structurées. Nous bénéficierons ainsi d'une meilleure précision de reconstruction grâce à la plus faible incertitude de triangulation donnée par des vues plus écartées. Pour valider ceci, il est nécessaire de le confronter à des mesures de type "vérité-terrain", et de vérifier que les points détectés par SIFT dans des vues simulées n'ont pas leurs positions biaisées.

- La prise en compte des motifs répétés pour améliorer la recherche d'appariements permet d'obtenir une couverture plus dense de certaines scènes par les points d'intérêt mis en correspondance. Lors d'une phase ultérieure de reconstruction, il est nécessaire

de lever les ambiguïtés associées à la présence d'objets identiques à différents endroits de la scène pour les reconstruire séparément les uns des autres.

- Intégrer l'imagerie médicale et la visualisation 3D des images panoramiques afin d'obtenir des informations de la forme 3D à partir du recalage effectué entre les images 2D. Cette visualisation permettrait de réajuster les images conformément à un modèle « réaliste » et de le réorienter. car au cours d'un examen, le clinicien se fait une représentation mentale de la forme 3D de l'organe utilisé. Nous pouvons imaginer présenter les images panoramiques en les projetant sur un volume représentant l'organe du patient.

BIBLIOGRAPHIE

[ALI07] M. Ali and D. Clausi, "Automatic registration of SAR and visible band remote sensing images," in Proceedings of IEEE International Geosciences and Remote Sensing, pp. 1331-1333, 2002.

[AZZ08] P. Azzari, L. Di Stefano, and S. Mattoccia, “An Evaluation Methodology for Image Mosaicing Algorithms”, ACIVS, Springer, vol. 5259, pp. 89-100, 2008.

[BÖH01] C. Böhm, S. Berchtold, and D. A. Keim, "Searching in high dimensional spaces: Index structures for improving the performance of multimedia databases," ACM Computer Survey, vol.33, issue.3, pp. 322-373, 2001.

[BRO03] M. Brown and D. Lowe, "Recognising Panoramas," in Proceedings of the 9th International Conference on Computer Vision, pp. 1218-1225, 2003.

[BRO05] M. Brown and R. Szeliski, "Multi-Image Matching using Multi-Scale Oriented Patches," in Proceedings of the International Conference on Computer Vision and Pattern Recognition, pp. 510-517, 2005.

[BRO06] P. Brodatz, "Textures: A photographic album for artists and designers," New York: Dover, 2006.

[CAN86] J. Canny, "A computational approach to edge detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, pp. 679-698, 1986.

[CAP01] D. P. Capel, "Image Mosaicing and super-resolution," in Departement of Engineering Science. PhD thesis: University of Oxford, 2001.

[CAS87] E. De Castro and C. Morandi, "Registration of translated and rotated images using finite Fourier transforms," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 9, pp. 700-703, 1987.

[CEL06] F. Cellier and E. Colin, "Building height estimation using fine analysis of altimetric mixtures in layover areas on polarimetric interferometric X-band SAR images," in IEEE International Geoscience and Remote Sensing Symposium, 2006.

[CHA03] C. M. Champagne, K. Staenz, A. Bannari, H. Mc Nairn, and J. Deguise, "Validation of a hyperspectral curve fitting model for the estimation of plant water content of agricultural canopies," Remote sensing of environment, vol. 83, pp. 148-160, 2003.

[CHA09] J. L. Chavez, P. H. Gowda, T. A. Howell, and K. S. Copeland, "Radiometric surface temperature calibration effects on satellite based evapotranspiration estimation," International Journal of Remote Sensing, vol. 30, pp. 2337-2354, 2009.

[CHA91] A. K. Chan, C. K. Chui, J. Zha, and Q. Liu, "Corner detection using spline-wavelets," in Proceeding of SPIE Curves Surfaces Comp. Vis. Graph. II, Boston, pp. 211-322, 1991.

[CHE03] H. Chen, P. Varshney, and M. Arora, "Mutual information based image registration for remote sensing data," International Journal of Remote Sensing, vol. 24, pp. 3701-3076, 2003.

[CHE95] C. H. Chen, J. S. Lee, and Y. N. Sun, "Wavelet transformation for gray level corner detection," Pattern Recognition vol. 28, pp. 853-861, 1995.

[CHM03] L. Chmielewski and D. Kozinska, "Image Registration," in Proceedings of the 3rd Polish Conference on Computer Pattern Recognition Systems, pp. 52-57, 2003.

[CHO05] M. Choi, R. Y. Kim, M. R. Nam, and H. O. Kim, "Fusion of multispectral and panchromatic satellite images using the curvelet transform," IEEE Transactions on Geoscience and Remote Sensing, vol. 2, pp. 136-140, 2005.

[CHU04] O. Chum, T. Werner, and J. Matas, "Epipolar geometry estimation via RANSAC benefits from the oriented epipolar constraint," in Proceedings of the International Conference on Pattern Recognition, pp. 112-115, 2004.

[CHU05] O. Chum and J. Matas, "Matching with PROSAC - Progressive Sample Consensus," in IEEE International Conference on Computer Vision and Pattern Recognition, pp. 220-226, 2005.

[CHR06] E. Christophe, "Compression des images hyperspectrales et son impact sur la qualité des Données,". Thèse de doctorat, École Nationale de l'aéronautique et de l'espace, 2006.

[CLO97] S. R. Cloude and E. Pottier, "An entropy based classification scheme for land applications of polarimetric SAR," IEEE Transactions on Geosciences and Remote Sensing, vol. 31, pp. 68-78, 1997.

[COL03] C. Colesanti, A. Ferretti, F. Novali, C. Prati, and F. Rocca, "SAR monitoring of progressive and seasonal ground deformation using the Permanent Scatterers Technique," IEEE Transactions on Geosciences and Remote Sensing, vol. 41, pp. 1685-1701, 2003.

[CUD10] T. Cudahy, M. Jones, M. Thomas, P. Cocks, F. Agustin, M. Caccetta, R. Hewson, M. Verrall, and A. Rodger, "Mapping Soil Surface Mineralogy at Tick Hill, North-Western Queensland, Australia, Using Airborne Hyperspectral Imagery," Progress in Soil Science vol. 1, pp. 211-229, 2010.

[CUI07] M. Cui, P. Wonka, A. Razdan, and J. Hu, "A new image registration scheme based on curvature scale space curve matching," The Visual Computer: International Journal of Computer Graphics, vol. 28, pp. 607-618, 2007.

[CUI09] M. Cui, J. Femiani, J. Hu, M. Wonka, and A. Razdan, "Curve matching for open 2D curves," Pattern Recognition Letters, vol. 30, pp. 1-10, 2009.

[DAT99] B. Datt, "A new reflectance index for remote sensing of chlorophyll content in higher plants: tests using Eucalyptus leaves," Journal of plant physiology, vol. 154, pp. 30-36, 1999.

[DAV93] P. A. Davis, C. S. Breed, J. F. McCauley, and G. G. Schaber, "Surficial geology of the Safsaf region, south-central Egypt, derived from remote-sensing and field data," Remote Sensing of Environment, vol. 46, pp. 183-203, 1993.

[DAV96] C. Davatzikos., "Nonlinear registration of brain images using deformable models," in Proceedings of the IEEE Workshop on Mathematical Methods in Biomedical Image Analysis, pp. 94-103, June 1996.

[DIN01] A. G. L. Ding, "Volume image registration by template matching," Image and Vision Computing, vol. 19, pp. 821-832, 2001.

[DOU04] M. Douze, "Estimation d'homographies inter-images. Cas des mosaïques et du suivi en temps réel. Applications en réalité augmentée,". Thèse de doctorat, Institut National Polytechnique, Toulouse, 2004.

[DUB09] E. Dubrofsky, "Homography Estimation," a Master's Essay, Faculty of Graduate Studies, the University of British Columbia, Mar. 2009.

[DYD03] I. Dydenko, D. Friboulet, and I. E. Magnin, "A variational framework for affine registration and segmentation with shape prior: application in echocardiographic imaging," Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV), pp. 926-938, 2003.

[EUG02] F. Eugenio, F. Marques, and J. Marcello, "A contour-based approach to automatic and accurate registration of multitemporal and multisensory satellite imagery," in IEEE International Geosciences and Remote Sensing, pp. 3390-3392, 2002.

[EUG03] F. Eugenio, "Automatic Satellite Image Georeferencing Using a Contour-Matching Approach," IEEE Transactions on Geosciences and Remote Sensing, vol. 41, 2003.

[FAU01] O. Faugeras, Q.-T. Luong, and T. Papadopoulou, “The Geometry of Multiple Images,” MIT Press, 2001.

[FED02] D. Fedorov, L. M. Fonseca, G. C. Kenney, and B. S. Manjunath, "Automatic registration and mosaicking system for remotely sensed imagery," in Proceedings of the 9th International Symposium on Remote Sensing (SPIE), September 2002.

[FIT01] A. W. Fitzgibbon., "Robust registration of 2D and 3D point sets," in Proceedings of the British Machine Vision Conference, BMVC01, September 2001.

[FLU00] J. Flusser, "On the independance of rotation moment invariants," Pattern Recognition, vol. 33, pp. 1405-1410, 2000.

[FLU06] J. Flusser and T. Suk, "Rotation moment invariants for recognition of symmetric objetcs," IEEE Transactions on Image Processing, vol. 15, pp. 3784-3790, 2006.

[FRA92] P. Fransson, "Using wavelet to detect corner points in images," in Dept. Mathematics: Lund Inst. Technol., Lund, Sweden, 1992.

[GAO07] X. Gao, F Sattar, and R. Venkateswarlu, "Multiscale corner detection of gray-level images based on Log-Gabor wavelet transform," IEEE Transaction on Circuits and Systems for Video Technology, vol. 17, pp. 868 - 875, July 2007.

[GAO12] X. Gao, "A Level Set Image Segmentation Method Based on Prior Shape Statistical Knowledge," International Review on Computers and Software, vol. 7, pp. 3221-3226, 2012.

[GAY06] V. Gay-Bellile, M. Perriollat, A. Bartoli, and P. Sayd., "Image registration by combining thin-plate splines with a 3D morphable model," in Proceedings of the International Conference on Image Processing (ICIP 06), 2006.

[GOM10] M. T. Gomez-Casero, I. L. Castillejo-Gonzalez, A. Garcia-Ferrer, J. M. Peña-Barragan, M. Jurado-Exposito, L. Garcia-Torres, and F. Lopez-Granados, "Spectral discrimination of wild oat and canary grass in wheat fields for less herbicide application," Agronomy for Sustainaible Development, vol. 30, pp. 689-699, 2010.

[GOS84] A. Goshtasby, S. H. Gage, and J. F. Bartholic, "A two-stage cross correlation approach to template matching," IEEE Transactions on Patern Analysis and Machine Intelligence, vol. 6, pp. 374-378, 1984.

[GOV09] M. Govender, P. J. Dye, I. M. Weiersbye, E. T. Witkowski, and F. Ahmed, "Review of commonly used remote sensing and ground-based technologies to measure plant water stress," Water SA, vol. 35, pp. 741-752, 2009.

[GOV99] V. Govindu and C. Shekhar, "Alignment using distributions of local geometric properties," IEEE Transactions of Pattern Analysis on Machine Intelligence, vol. 21, pp. 1031-1043, 1999.

[GRA05] N. Gracias and S. Negahdaripour, "Underwater mosaic creation using video sequences from different altitudes," in Proceedings of the MTS/IEEE OCEANS 05, pp. 1234-1240, 2005.

[GU12] Y. Gu, P. Chen, and C. Liu, "The Algorithm of Preserving the Spatial Location of 3D Image on Android Platform," International Review on Computers and Software, vol. 7, pp. 3227-3231, 2012.

[HA08] S. J. Ha, S. H. Lee, N. I. Cho, S. K. Kim, and B. Son, "Embedded Panoramic Mosaic System Using Auto-Shot Interface," IEEE Transactions on Consumer Electronics, vol. 54, pp. 16-24, 2008.

[HAR00] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision: Cambridge University Press, 2000.

[HAR04] R. I. Hartley and A. Zisserman, "Multiple View Geometry," in Computer Vision, C. U. Press, Ed., 2004.

[HAR88] C. Harris and M. J. Stephens., "A combined corner and edge detector," in Proceedings of the Alvey Vision Conference, pp. 147-152, 1988.

[HEI05] M. Heikkilä and M. Pietikäinen, "An image mosaicing module for wide-area surveillance," in Proceedings of the third ACM International Workshop on Video surveillance & Sensor Networks (VSSN 05), pp. 11-18, 2005.

[HON91] J. Hong, X. Tan, B. Pinette, R. Weiss, and E. M. Riseman, "Image-based homing," in IEEE International Conference on Robotics and Automation, pp. 620 -625, 1991.

[HSI97] J. W. Hsieh, H. Y. M. Liao, K. C. Fan, M. T. Ko, and Y. P. Hung, "Image Registration Using a New Edge-Based Approach," Computer Vision and Image Understanding, vol. 67, pp. 112-130, 1997.

[HSU02] C. Hsu and R. Beuker, "Multiresolution feature-based image registration," in Proceedings of SPIE Visual Communications and Image Processing, pp. 1490-1498, 2002.

[HUA07] D. S. Huang, L. Heutte, and M. Loog, "Feature Based Automatic Stitching of Microscopic Images," Berlin: Springer-Verlag, 2007.

[HWA98] W. J. Hwang and K. W. Wen, "Fast KNN classification algorithm based on partial distance search," in Electronic Letter, vol.34, issue.21, pp. 2062-2063, 1998.

[IOD09] A. Iodice, "A Survey of Differential SAR Interferometry for Surface Displacement Monitoring," in Proceedings of the 6th European Radar Conference, pp. 212-214, 2009.

[IRA96] M. Irani, P. Anandan, J. Bergen, R. Kumar, and S. Hsu, "Efficient representations of video sequences and their applications," Signal processing. Image communication, vol. 8, pp. 327-351, May 1996.

[JON03] S. Jonsson, P. Segall, R. Pedersen, and G. Bjarnsson, "Postearthquake ground movements correlated to pore-pressure transients," Nature vol. 424, pp. 179-183, 2003.

[JOY09] K. Joyce, S. Samsonov, and G. Jolly, "Temperature, color and deformation monitoring of volcanic regions in New Zealand," in IEEE International Geosciences and Remote Sensing Symposium, pp. 17-20, 2009.

[KOP00] M. Koppen, "The curse of dimensionality," In 5th Online World Conference on Soft Computing in Industrial Applications (WSC5), September 2000.

[KOV03] P. Kovesi, "Phase congruency detects corners and edges," in Proceeding of Australian Pattern Recognition. Soc. Conf. DICTA, pp. 309-318, 2003.

[KYB03] J. Kybic and M. Unser, "Fast parametric elastic image registration," IEEE Transactions on Image Processing, vol. 12, pp. 1427-1442, 2003.

[LAC98] R. Lacassin, B. Meyer, L. Benedetti, R. Armijo, and P. Tapponnier, "Signature morphologique de l'activité de la faille des Cévennes," Comptes Rendus de l'Académie des Sciences, vol. 326, pp. 807-815, 1998.

[LI04] W. Li and H. Leung, "A maximum likelihood approach for image registration using control point and intensity," IEEE Transactions on Image Processing, vol. 13, pp. 1115-1127, 2004.

[LI07] Z. Li and H. Leung, "Contour-based multisensor image registration with rigid transformation," in 10th International Conference on Information Fusion, pp. 1-7, 2007.

[LI95] H. Li, B. Manjunath, and S. Mitra, "A contour-based approach to multisensor image registration," IEEE Transactions on Image Processing, vol. 4, pp. 320-334, 1995.

[LIL04] T. M. Lillesand, W. K. Ralph, and J. W. Chipman, Remote sensing and image interpretation John Wiley & Sons, 2004.

[LIN98] T. Lindeberg. "Feature Detection with Automatic Scale Selection," International Journal of Computer Vision, vol. 30, issue. 2, pp. 79-116, 1998.

[LIN01] C. Y. Lin, M. Wu, J. A. Bloom, I. J. Cox, M. L. Miller, and Y. M. Lui, "Rotation, scale, and translation resilient watermarking for images," IEEE Transactions on Image Processing, vol. 10, pp. 767-782, May 2001.

[LIN06] G. Lingjia, G. Shuxu, and R. Ruizhi, "A novel method of dynamic target detection," in Proceedings of SPIE, Sixth International Symposium on Instrumentation and Control Technology: Signal Analysis, Measurement Theory, pp. 1-7, 2006.

[LIU11] X. Liu, Z. Tian, C. Chai, and H. Fu, "Multiscale registration of remote sensing image using robust SIFT features in Steerable-Domain," The Egyptian Journal of Remote Sensing and Space Sciences, vol. 14, pp. 63-72, 2011.

[LOU08] L. Lou, F. M. Zhang, C. Xu, M. F. Li, and G. Xue, "Automatic registration of aerial image series using geometric invariance," in Proceedings of the IEEE International Conference on automation and logistics, pp. 1198-1203, 2008.

[LOW04] D. Lowe, "Distinctive Image Features From Scale-Invariant Keypoints," International Journal of Computer Vision, vol. 60, pp. 91-110, 2004.

[LU10] D. Lu, S. Hetrick, E. Moran, and G. Li, "Detection of Urban Expansion in an Urban-rural Landscape with Multitemporal QuickBird Images," Journal of Applied Remote Sensing, vol. 4, 2010.

[MAK01] T. Makela, P. Clarysse, J. Lötjönen, O.Sipilä, K. Lauerma, H. Hänninen, E.P Pyökkimies, J. Nenonen, J. Knuuti, T. Katila, and I. E. Magnin., "A new method for the registration of cardiac PET and MR images using deformable model based segmentation of the main thorax structures," in Procceedings of the 4th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 01), pp. 557-564, 2001.

[MAN96] B. S. Manjunath, C. Shekharm, and R. Chellappa, "A new approach to image feature detection with applications," Pattern Recognition, vol. 29, pp. 627-640, 1996.

[MAR79] D. Marr and T. Poggio, "A Computational Theory of Human Stereo Vision," *Royal Society*, London B 204, 1979.

[MCN02] H. McNairn, C. Duguay, B. Brisco, and T. J. Pultz, "The effect of soil and crop residue characteristics on polarimetric radar response," Remote sensing of environment vol. 80, pp. 308-320, 2002.

[MCN04] H. McNairn, K. Hochheim, and N. Rabe, "Applying polarimetric radar imagery for mapping the productivity of wheat crops," Canadian journal of remote sensing, vol. 30, 2004.

[MED09] L. Meddeber, N. E. Berrached and A. Taleb Ahmed “An automatic registration and mosaicking system based on contour features and wavelet transform for remote sensing images”, IEEE Conferences, Signals, Circuits and Systems (SCS), ICSCS09, pp. 1-7, 2009.

[MED10] L. Meddeber, N. E. Berrached and A. Taleb Ahmed "The Practice of Automatic Registration System for Remote Sensing Images", International Journal on Graphics, Vision and Image Processing, , GVIP, vol. 09, issue.VI, pp. 67-75, Dec 2010.

[MED13] L. Meddeber, T. Zouagui, A. Taleb Ahmed and N. E. Berrached, "A New Automatic Image Registration Algorithm Based On Corner Detector For Remote Sensing Applications", International Review on computers and Software (IRECOS), Vol. 8, N.1, ISSN: 1828-6003, pp.347-362, January 2013.

[MIK04] K. Mikolajczyk and C. Schmid., " Scale & affine invariant interest point detectors," International Journal of Computer Vision, vol. 60, pp. 63-86, 2004.

[MOI04] L. Moisan and B. Stival, "A probabilistic criterion to detect rigid point matches between two images and estimate the fundamental matrix," International Journal of Computer Vision, vol. 57, issue.3, pp. 201-218, 2004.

[MOK98] F. Mokhtarian and R. Suomela, "Robust image corner detection through curvature scale space," IEEE Transactions on Patern Analysis and Machine Intelligence, vol. 20, pp. 1376-1381, 1998.

[MOR79] H. P. Moravec, "Visual mapping by a robot rover," in Proceedings of the International Joint Conference on Artificial Intelligence, pp. 598-600, 1979.

[MOR07] P. Moreels and P. Perona, "Evaluation of features detectors and descriptors based on 3D objects," International Journal of Computer Vision, vol. 73, issue. 3, pp. 263-284, 2007.

[MOS05] M. Moshir, "Spitzer space telescope data processing and algorithmic complexity," in Astronomical Data Analysis Software and Systems XIV. ASP Conference Series, pp. 434-443, 2005.

[MUN92] A. Z. J. L. Mundy, Geometric Invariance in Computer Vision. Cambridge: The MIT Press, 1992.

[NI 09] K. Ni, H. Jin, and F. Dellaert, "Groupsac: Efficient consensus in the presence of groupings," in Proceedings of the International Conference of Computer Vision, 2009.

[OLI11] C. G. Oliveira, W. R. Paradella, and A. D. da Silva, "Assessment of radargrammetric DSMs from TerraSAR-X Stripmap images in a mountainous relief area of the Amazon region," Journal of Photogrammetry & Remote Sensing, vol. 66, pp. 67-72, 2011.

[ORC05] J. Orchard, "Efficient global weighted least-squares translation registration in frequency domain," in Proceedings of International Conference on Image Analysis and Recognition, pp. 116-124, 2005.

[OSM11] B. Osmanoglu, T. H. Dixon, S. Wdowinski, E. Cabral-Cano, and Y. Jiang, "Mexico City subsidence observed with persistent scatterer InSAR," International Journal of Applied Earth Observation and Geoinformation, vol. 13, pp. 1-12, 2011.

[PAN99] J. S. Pan, Y. L. Qiao and S. H. Sun, "Texture segmentation using separable and non-separable wavelet frames," IEICE Transactions Fundamentals, vol. E82-A, issue 8, pp. 1463-1474, 1999.

[PAN04] J. S. Pan, Y. L. Qiao and S. H. Sun, "A fast *K* nearest neighbors classification algorithm," IEICE Transactions Fundamentals, vol. E87-A, issue. 4, pp. 961-963, 2004.

[PIZ03] O. Pizarro and H. Singh, "Towards Large Area Mosaicing for Underwater Scientific Applications," IEEE Journal of Oceanic Engineering: Special Issue on Underwater Image and Video Processing, vol. 28, pp. 651-672, 2003.

[PLA07] A. Plaza, J. Le Moigne, and N. S. Netanyahu, "Morphological Feature Extraction for Automatic Registration of Multispectral Images," IEEE Transactions Geosciences and Remote Sensing, vol. 43, pp. 421-426, 2007.

[PLU03] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever, "Mutual-informationbased registration of medical images: A survey," IEEE Transactions on Medical Imaging, vol. 22, pp. 986-1004, 2003.

[QUD98] A. Quddus and M. M. Fahmy, "Corner detection using Gabor-type filtering," in Proceeding of IEEE International Symposium of Circuits and Systems (ISCAS 98), pp. 150-153, 1998.

[RAB10] J. Rabin, J. Delon, Y. Gousseau, and L. Moisan, “Mac-ransac: a robust algorithm for the recognition of multiple objects,” In Symposium on 3D Data Processing, Visualization and Transmission (3D’PVT), 2010.

[RAL04] E. Ralph, R. G. Weeks, P. E. Bibhuti Panda, and G. I. T. Kenneth Fergason, "Application of synthetic aperture radar interferometry (InSAR) in defining mine-related ground deformation and subsidence hazards," in Interstate technical group on abandoned underground mines 5th biennial workshop, 2004.

[RAN05] V. Rankovng, R. Locke, R. Edens, P. Barber, and B. Vojnovic, "An algorithm for image stitching and blending," in Proceedings of SPIE on Three-Dimentional Microscopy: Image Acquisition and Processing, pp. 190-199, 2005.

[RED96] B. S. Reddy and B. N. Chatterji, "An FFT-based technique for translation, rotation, and scale-invariant image registration," IEEE Transactions on Image Processing, vol. 5, pp. 1266-1271, 1996.

[ROS05] E. Rosten and T. Drummond., "Fusing points and lines for high performance tracking," in Proceedings of the IEEE International Conference on Computer Vision (ICCV2005), pp. 1508-1511, October 2005.

[ROS98] P. Rosen, C. Werner, E. Fielding, S. Hensley, S. Buckley, and P. Vincent, "Aseismic creep along the San Andreas fault northwest of Parkfield, CA measured by radar interferometry," Geophysical Research Letters, vol. 25, pp. 825-828, 1998.

[RUZ01] M. A. Ruzon and C. Tomasi, "Edge, junction, and corner detection using color distributions," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, pp. 1281-1295, 2001.

[RZH02] Y. Rzhanov, L. Huff, and R. G. Cutter Jr, "Seafloor video mapping: modeling, algorithms, apparatus," in Proceedings of the International Conference on Image Processing (ICIP2002), 2002.

[SAT04] T. Sato, S. Ikeda, M. Kanbara, A. Iketani, N. Nakajima, N. Yokoya, and K. Yamada, "High-resolution video mosaicing for documents and photos by estimating camera motion," in Proceedings of the SPIE 04, pp. 246-253, 2004.

[SCH00] C. Schmid, R. Mohr, and C. Bauckhage., "Evaluation of interest point detectors," International Journal of Computer Vision, vol. 37, pp. 151-172, 2000.

[SHA07] R. Shams, N. Barnes, and R. Hartley, "Image registration in Hough space using gradient of images," in Proceeding of Digital Image Computing: Techniques and Applications, pp. 226-232, 2007.

[SHI94] J. Shi and C. Tomasi, "Good features to track," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 94), pp. 593-600, June 1994.

[SKO89] T. Skordas et R. Horaud, "Mise en Correspondance de Structures 2D dans une Paire d'Images Stéréoscopiques," rapport de recherche INRIA, 2ème congrès TIPI, (Savoie, France), 1989.

[SMI97] S. M. Smith and J. M. Brady, "SUSAN - A new approach to low level," image processing. International Journal of Computer Vision, vol. 23, pp. 45-78, 1997.

[SOJ02] E. Sojka, "A new algorithm for detecting corners in digital images," in Proceeding of Spring Conference on Computer Graphics, Bundmercie, Slovakia, pp. 47-54, 2002

[STE92] C.V. Stewart, "Minpran: A new robust estimator for computer vision," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 17, issue.10, pp.925-938, 1995.

[STE05] D. Steedly, C. Pal, and R. Szeliski, "Efficiently Registering Video into Panoramic Mosaics," in Proceedings of the 9th International Conference on Computer Vision, pp. 1300-1307, 2005.

[SZE96] R. Szeliski, "Video mosaics for virtual environments," IEEE Computer Graphics and Applications, vol. 16, pp. 22-30, 1996.

[SZE97] R. Szeliski and H. Y. Shum, "Creating full view panoramic image mosaics and environment map," in SIGGRAPH 97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques, pp. 251-258, 1997.

[THE03] S. Theodoridis and K. Koutroumbas, "Pattern Recognition," China Machine Press, pp. 44-45, 2003.

[THO96] P. M. Thompson and A. W. Toga, "A surface-based technique for warping three-dimensional images of the brain," IEEE Transactions on Medical Imaging, vol. 15, pp. 402-417, 1996.

[TOU01] T. Toutin, "Elevation Modelling from Satellite VIR Data: A Review," International Journal of Remote Sensing, vol. 22, pp. 1097-1125, 2001.

[TOR05] B.J. Tordoff and D.W. Murray, "Guided-MLESAC: Faster image transform estimation by using matching priors," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, issue.10, pp. 1523-1535, 2005.

[TOZ05] G. Tozer, S. Ameer-Beg, J. Baker, P. Barber, S. Hill, R. Hodgkiss, R. Locke, V. Prise, I. Wilson, and B. Vojnovic, "Intravital imaging of tumour vascular networks using multi-photon fluorescence microscopy," Advanced Drug Delivery Reviews, vol. 57, pp. 135-152, 2005.

[TRA02] M. Trajkovie and M. Hedley, "Fast corner detection," Image and Vision Computing, vol. 16, pp. 75-87, 2002.

[TRI99] B. Triggs, "Bundle Adjustment- a Modern Synthesis," in Proceedings of the International Workshop on Vision Algorithms, pp. 298-372, 1999.

[TUP03] F. Tupin and M. Roux, "Detection of building outlines based on the fusion of SAR and optical features," Journal of Photogrammetry & Remote Sensing, vol. 58, pp. 71-82, 2003.

[TOR00] P. Torr and A. Zisserman, "MLESAC: A new robust estimator with application to estimating image geometry," Computer Vision and Image Understanding, vol. 78, pp.138–156, 2000.

[TOR02] P. Torr, "Bayesian model estimation and selection for epipolar geometry and generic manifold fitting," International Journal of Computer Vision, vol. 50, pp.35-61, 2002.

[UYT01] M. Uyttendaele, A. Eden, and R. Szeliski, "Eliminating Ghosting and Exposure Artifacts in Image Mosaics," in Proceedings of the International Conference on Computer Vision and Pattern Recognition, pp. 509-516, 2001.

[VIO97] P. Viola and W. M. Wells, "Alignment by maximization of mutual information," International Journal of Computer Vision, vol. 24, pp. 137-154, 1997.

[WAN95] H. Wang and M. Brady., "Real-time corner detection algorithm for motion estimation," Image and Vision Computing, vol. 13, pp. 695-703, 1995.

[WAN97] W. H. Wang and Y.C. Chen, "Image Registration by Control Points Pairing Using the Invariant Properties of Line Segments," Pattern Recognition Letters, vol. 18, pp. 269-281, 1997.

[WIJ09] A. Wijaya and R. Gloaguen, "Fusion of ALOS Palsar and Landsat ETM Data for Land Cover Classification and Biomass Modeling using Non-linear Methods," in IEEE International Geosciences and Remote Sensing Symposium, pp. 581-584, 2009.

[WOO93] R. P. Woods, J. C. Mazziotta, and S. R. Cherry, "MRI-PET registration with automated algorithm," Journal of Computer Assisted Tomography, vol. 17, pp. 536-546, 1993.

[WRI01] T. Wright, B. Parsons, and E. Fielding, "Measurement of interseismic strain accumulation across the North Anatolian Fault by satellite radar interferometry," Geophysical Research Letters, vol. 28, pp. 2117-2120, 2001.

[XIA05] D. Xiaolong and S. Khorram, "A feature-based image registration algorithm using improved chain-code representation combined with invariant moments," IEEE Transactions on Geosciences and Remote Sensing, vol. 37, pp. 2351-2362, 1999.

[XIE03] H. Xie, N. Hicks, G. R. Keller, H. Huang, and V. Kreinovich, "An IDL/ENVI implementation of the FFT-based algorithm for automatic image registration," Computers & Geosciences, vol. 29, pp. 1045-1055, March 2003.

[ZAV05] I. Zavorin and J. Le Moigne, "Use of multiresolution wavelet feature pyramids for automatic registration of multisensor imagery," IEEE Transactions on Image Processing, vol. 14, pp. 770-782, 2005.

[ZHA04] Y. Zhao and Y. Q. Chen, "Connected Equi-Length Line Segments For Curve And Structure Matching," International Journal of Pattern Recognition and Artificial Intelligence, vol. 18, pp. 1019-1037, 2004.

[ZHA04] J. Zhang and A. Rangarajan, "A unified feature-based registration method for multimodality images," Proceedings of the 2004 IEEE International Symposium on Biomedical Imaging (ISBI), pp. 724-727, 2004.

[ZHA08] M. Zhang, A. Guo, W. Zhong, and L. Guo, "Image stitching of scenes with large misregistration," in Proceedings of the IEEE International Conference on Computer Science and Information Technology, pp. 803-807, 2008.

[ZIT01] B. Zitova, J. Kautsky, G. Peters, and J. Flusser, "Robust detection of significant points in multiframe images," Pattern Recognition Letters, vol. 20, pp. 199-206, 2001.

[ZIT03] B. Zitova and J. Flusser, "Image registration methods: A survey," Image and Vision Computing, vol. 21, pp. 977-1000, 2003.

[ZOM00] A. Zomet and S. Peleg, "Efficient super-resolution and applications to mosaics," in Proceedings of the 15th International Conference on Pattern Recognition (ICPR00), pp. 579-583, 2000.

[ZOR09] M. A. Zoran, R. Savastru, D. Savastru, M. Tautan, and S. Miclos, "Multispectral and multitemporal satellite remote sensing imagery for Bucharest land cover dynamics assessment," in Proceedings of SPIE, vol. 7469, 2009.

COMMUNICATIONS NATIONALES ET INTERNATIONALES

- L. Meddeber, N.E. Berrached, "Automatic Registration and Mosaicing System for Remote Sensing Images", 3ème séminaire sur les systèmes de détection (DAT'06), Architectures et technologies,(Ministère de la défense nationale, Commandement des forces de défense aérienne du territoire.), 2006.
- L. Meddeber, N.E. Berrached, " An Automatic Registration System based on contour features for Remote Sensing Images", International Conference on Applied Informatics Bordj Bou Arréridj, Nov 2009.
- L. Meddeber, N.E. Berrached, " Development of a fuzzy inference and Markovian system for the local registration of satellite and cartographic images", International Conference on Applied Informatics Bordj Bou Arréridj , ALGERIA, Nov 2009.
- L. Meddeber, N.E. Berrached, "An automatic registration and mosaicking system based on contour features and wavelet transform for remote sensing images", 5ème Symposium International en Images, Multimédias, Applications, Graphiques et Environnements, Biskra, ALGERIA , Nov 2009.
- L. Meddeber and N.E. Berrached, "The practice of automatic registration system for remote sensing images", 5th international Conference: Sciences of Electronics, Technologies of Information Telecommunications March 2009 (SETIT09), TUNISIA, 2009.
- L. Meddeber, N.E. Berrached and A. Taleb Ahmed "Automatic Registration and Mosaicking System for Remote Sensing Images", La 1er journée de rencontre entre les chercheurs en image, en vision et en reconnaissance des formes de la Région Nord Pas-de Calais et Picardie, Université de Lille, 2010.

PUBLICATIONS

- L. Meddeber, N. E. Berrached and A. Taleb Ahmed, " The Practice of Automatic Registration System for Remote Sensing Images", International Journal on Graphics, Vision and Image Processing, GVIP, vol. 09, issue. 6, pp. 67-75, Dec 2009.
- L. Meddeber, N.E. Berrached and A. Taleb Ahmed, "The Practice of an Automatic Registration System Based on Contour Features and Wavelet Transform for Remote Sensing Images", IEEE Conferences, Computer and Electrical Engineering (ICCEE '09), vol. 2, pp. 58-63, 2009.
- L. Meddeber, N.E. Berrached and A. Taleb Ahmed, "An automatic registration and mosaicking system based on contour features and wavelet transform for remote sensing images", IEEE Conferences, Signals, Circuits and Systems (SCS09), vol. 11, pp. 1-7, 2009.
- L. Meddeber and N. E. Berrached, " Development of a fuzzy inference and Markovian system for the local registration of satellite and cartographic images ", IEEE Conferences, Signals, Circuits and Systems (SCS09), vol. 10, pp.1-7, 2009
- L. Meddeber, T. Zouagui, A. Taleb Ahmed and N. E. Berrached, "A New Automatic Image Registration Algorithm Based On Corner Detector For Remote Sensing Applications", International Review on computers and Software (IRECOS), vol. 8, N.1, ISSN: 1828-6003, pp.347-362, January 2013.