Correction des Effets de Relief en Spectro-imagerie Aéroportée

Clara Barbanson-Graziussi

▶ **To cite this version:**

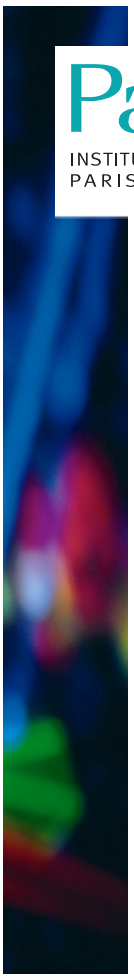
Clara Barbanson-Graziussi. Correction des Effets de Relief en Spectro-imagerie Aéroportée. Traitement du signal et de l'image [eess.SP]. Telecom ParisTech, 2018. Français. NNT : . tel-01915693

HAL Id: tel-01915693
https://theses.hal.science/tel-01915693

Submitted on 7 Nov 2018

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

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



ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

École des Ponts ParisTech

ONERA
THE FRENCH AEROSPACE LAB

TELECOM ParisTech

2018-ENST-0029

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech

Spécialité « Signal et Images »

présentée et soutenue publiquement par

Clara BARBANSON-GRAZIUSSI HERRERO

le 31 mai 2018

Corrections des effets de relief en spectro-imagerie aéroportée

Jury

Andrés ALMANSA	MAP5, Paris Descartes	Co-directeur
Laure BLANC-FÉRAUD	Laboratoire I3S, Université Nice Sophia Antipolis	Examinatrice
Yann FERREC	DOTA, Office National d'Etude et de Recherche Aérospatiales	Encadrant
David FOFI	Le2i, Université de Bourgogne	Rapporteur
Rémi MICHEL	Tridimeo	Examinateur
Pascal MONASSE	Imagine, Ecole Nationale des Ponts et Chaussées	Co-directeur
Pablo MUSÉ	Universidad de la Republica, Uruguay	Examinateur
Marc PIERROT-DESEILLIGNY	LaSTIG, Ecole Nationale des Sciences Géographiques	Rapporteur

T H È S E

TELECOM ParisTech

école de l'Institut Mines-Télécom - membre de ParisTech

46 rue Barrault 75013 Paris - (+33) 1 45 81 77 77 - www.telecom-paristech.fr

Correction des Effets de Relief en Spectro-imagerie Aéroportée

Clara BARBANSON

REMERCIEMENTS

Je voudrais d'abord, et particulièrement, remercier mes directeurs Andrés Almansa et Pascal Monasse ainsi que mon encadrant Yann Ferrec. Ces années de thèse ont été difficiles par beaucoup d'aspects et je suis reconnaissante d'avoir bénéficié de leurs connaissances ainsi que de leur bienveillance tout au long de ces années.

Pour leur relecture attentive et leurs remarques, je remercie les rapporteurs David Fofi et Marc Pierrot-Desseiligny, la présidente du jury Laure Blanc-Féraud et les examinateurs Pablo Musé et Rémi Michel.

Je remercie également spécialement Mila Nikolova et Daniel Chen Soncco pour leur collaboration sur le difficile problème du modèle multiplicatif. Merci aussi à Christophe Coudrain, Laurent Rousset-Rouvière et Rémi Gouyon, qui ont répondu à mes questions et mes sollicitations. Merci à Pauline Tan pour son aide sur le graphcut.

Cette thèse n'aurait pas été possible sans le soutien de l'ONERA, qui a perçu à quel point le sujet était riche et intéressant et m'a accordé une bourse complète. Je remercie aussi le laboration Images de Telecom Paristech de m'avoir accueillie.

Les difficultés s'acceptent mieux quand elles sont partagées, et pour cela je remercie mes collègues doctorants. D'abord mes co-bureaux : Thomas, dont le portrait fait encore rêver le dernier étage du D1, Christophe, mon compagnon matinal, Flo, mon compagnon vespéral et Philippe (qui pourrait bien m'en vouloir ?). Merci aussi à l'ensemble du 3^e étage du D1 : Alex, pour son humeur placide, Armande, pour nos dicussions riches et variées, Edouard, et le duc d'Anjou, François, pour ses histoires du web libre et du monde réel, Fred, Jean, Kevin et ses recommendations hi-tech, Mouna, Tatiana Piña Colada et Thibault. Et je n'oublie pas le sous-sol : Alice, pour toutes ces questions que tu soulèves, Anasthase, pour avoir dû y répondre, et Steve. Je remercie Malik, pour toutes ces discussions sérieuses ou pas du tout, et pour ton aide dont tu n'es jamais avare. Enfin je remercie bien sûr Carlos, Clément, Cristian, David, Flora, Gang Liu, Gilles, Guillaume, Guillaume Tartavel, Hélène, Sonia, Sylvain 2D, Sylvain 3D, Timothée, Thibault, Thuc Trinh et Xiaoyi.

Merci enfin à mes parents, mon frère et mes tantes qui m'ont tant appris et tant soutenue. A Elias, merci pour ta patience et ton soutien, passées et à venir.

RÉSUMÉ

Cette thèse présente une série de traitements appliqués aux images d'un spectro-imageur à transformée de Fourier statique développé par l'Office National d'Etude et de Recherche Aérospatiales (ONERA). Il s'agit d'un instrument hyperspectral aéroporté qui acquiert des séquences denses d'images. Ces séquences forment un cube interférométrique dont la troisième dimension fournit l'information spectrale (le passage d'interférogramme à spectre se fait par une transformée de Fourier). Toutefois, en présence de relief sur la scène, l'erreur de parallaxe entre les différentes images de la séquence génère des erreurs dans le cube si les images ne sont pas corrigées avec un modèle numérique d'élévation (MNE). Ce dernier étant en général inconnu, on peut se servir du cube interférométrique, doté de nombreuses images de la scène, à des fins de stéréoscopie. Ce problème de parallaxe et ces conditions d'acquisitions se retrouvent aussi dans les autres spectro-imageurs à trame défilante, auxquels les traitements stéréoscopiques de ce travail pourraient s'appliquer.

L'objectif de cette thèse est donc de calculer le MNE de la scène survolée à partir des images de l'instrument. Les séquences d'images présentent toutefois diverses difficultés. D'une part, chaque image de la séquence contient des franges d'interférences fixes qui se superposent à la scène. D'autre part, les disparités observées entre deux clichés successifs sont inférieures au centième de pixel.

Les franges sur les images, très contrastées à certains endroits, perturbent les traitements stéréoscopiques. Nous proposons deux décompositions d'image afin de séparer la figure d'interférence de l'image de la scène : une additive, plus classique et accessible car il s'agit d'un problème convexe, et une multiplicative, plus proche du modèle de formation d'image mais plus complexe du fait que le problème devient biconvexe. Une attention particulière est portée à cette étape, qui ne doit pas détériorer la qualité de l'image afin de procéder aux mesures stéréoscopiques.

Afin de tirer profit de la grande quantité d'images et de gérer les disparités sous-pixelliques, nous utilisons les images des plans épipolaires (EPI) pour obtenir une carte de disparité éparse. Celles-ci correspondent à des images spatio-temporelles construites à partir des lignes épipolaires. Elles décrivent le déplacement des points de la scène en fonction de la position de la caméra et donnent ainsi accès à la mesure de disparité. Une des hypothèses pour pouvoir utiliser cette méthode est que tous les sommets de prise de vue soient colinéaires. Or, en imagerie aéroportée, les conditions de vol font que l'avion peut légèrement dévier de sa trajectoire. Nous étudions donc l'impact de ces erreurs sur la mesure de disparité.

Pour obtenir un modèle numérique de surface continu, nous proposons de densifier la carte éparse avec une minimisation d'énergie par graphcut. Nous reprenons un algorithme existant et l'adaptons au cas multi-vues et sous-pixellique.

TABLE DES MATIÈRES

Table des matières iv

Introduction 1

I Préliminaires 5

1 Sieleters, un interféromètre imageur aéroporté 7
1.1 Imagerie hyperspectrale . . . 8
1.2 Généralités sur la spectrométrie par transformée de Fourier . . . 9
1.2.1 Principe de la spectrométrie par transformée de Fourier . . . 9
1.2.2 Avantages et inconvénients . . . 10
1.3 Sieleters . . . 11
1.3.1 Contexte . . . 11
1.3.2 Concept instrumental . . . 11
1.3.3 Reconstruction des spectres et problèmes dus au relief . . . 14
1.4 Instruments existants . . . 15
1.5 Conclusion . . . 17

2 La Stéréoscopie 19
2.1 Généralités . . . 19
2.1.1 Modèle de caméra . . . 20
2.1.2 Triangulation . . . 21
2.2 Etat de l'Art sur la stéréoscopie. . . . 22
2.2.1 Stéréoscopie à deux vues. . . . 23
2.2.2 Stéréoscopie multi-vues. . . . 29
2.3 Conclusion . . . 34

3 Présentation des images utilisées 35
3.1 Images simulées . . . 35
3.2 Séquences Sieleters . . . 36
3.2.1 Généralités sur les images Sieleters . . . 36
3.2.2 Séquence Le Fauga-Mauzac (CFM) . . . 36
3.2.3 Séquence Canjuers . . . 40
3.3 Evaluation quantitative : le PSNR . . . 40

II Extraction des franges d'interférence pour récupérer l'image panchromatique 43

4 Mise en évidence des perturbations causées par les franges sur la stéréoscopie 45
4.1 Stéréoscopie en présence de franges d'interférence . . . 45
4.1.1 Tests sur différents algorithmes . . . 45

4.2 Etat de l'art de l'extraction de franges d'interférences 47
4.3 Caractérisation des franges d'interférences de Sieleters 47
4.4 Conclusion 49

5 Utilisation d'un modèle d'image additif 51
5.1 Modèle de décomposition additif 52
5.1.1 Dérivation physique du modèle 52
5.1.2 De la nécessité d'une approche variationnelle 52
5.1.3 Etat de l'art de la décomposition *Cartoon-Texture* 52
5.2 Outils d'optimisation convexe 53
5.2.1 Notions d'optimisation convexe 53
5.2.2 L'opérateur de proximité 54
5.2.3 L'algorithme Douglas-Rachford 56
5.2.4 Variation totale 56
5.3 Expression du problème 58
5.3.1 La contrainte fréquentielle 58
5.3.2 Le problème de minimisation 58
5.3.3 Algorithme 59
5.4 Choix des paramètres 59
5.5 Résultats 60
5.5.1 Résultats sur image simulée 60
5.5.2 Sur images Sieleters 60
5.6 Conclusion 61

6 Utilisation d'un modèle d'image multiplicatif 67
6.1 Modèle de décomposition multiplicatif 68
6.1.1 Dérivation physique du modèle 68
6.1.2 Difficultés du modèle 68
6.2 Description du problème 69
6.2.1 La contrainte fréquentielle 69
6.2.2 La régularisation 72
6.2.3 Expression du problème 72
6.3 Algorithmes 73
6.3.1 Algorithme par descente proximale 73
6.3.2 Algorithme par filtrage 74
6.3.3 Initialisation et normalisation 75
6.4 Choix des paramètres 75
6.4.1 Paramètres de l'algorithme par descente proximale 76
6.4.2 Paramètres de l'algorithme par filtrage 78
6.5 Résultats 78
6.5.1 Sur images simulées 78
6.5.2 Sur images Sieleters 78
6.6 Limites du modèle 79
6.6.1 L'hypothèse sur la forme des spectres 79
6.6.2 Cas particuliers de textures 81
6.7 Conclusion 83

Conclusion 85

III Construction du modèle numérique d'élévation 89

7 Etude de la contrainte sur la ligne de visée 91
7.1 Les problèmes d'attitude et de trajectoire dans la littérature . . . 91
7.2 Influence des écarts de trajectoire sur les images des plans épipolaires . . . 93
7.3 Recalage des images à l'aide des données de navigation . . . 94
7.4 Quantification de l'erreur sur la disparité lorsque le porteur s'écarte de la ligne de visée . . . 95
7.5 Résultats pour les paramètres de vol Canjuers . . . 99
7.6 Conclusion . . . 100

8 Calcul d'une carte de disparité éparse à partir des images des plans épipolaires 103
8.1 Construction et exploitation des EPI . . . 103
8.1.1 EPI des images Sieleters . . . 103
8.1.2 Extraction des segments . . . 105
8.2 Résultats . . . 107
8.3 Conclusion . . . 109

9 Densification de la carte éparse par coupure de graphe 111
9.1 Généralités sur la minimisation par coupure de graphe . . . 111
9.2 Adaptation d'un algorithme de coupure de graphe . . . 113
9.2.1 Algorithme utilisé . . . 113
9.2.2 Adaptation aux multi-vues . . . 115
9.3 Résultats . . . 115
9.4 Conclusion . . . 117

Conclusion Générale 119

A Calcul du terme de régularisation 123

Bibliography 127

Table des figures 137

INTRODUCTION

Le traitement d'image au service des systèmes optiques

L'acquisition d'images est désormais souvent indissociable du traitement d'images, qui permet d'améliorer la qualité de la mesure ou l'exploitation des données. Cela amène à aborder le développement des instruments imageurs comme une co-conception hybride entre les sytèmes optiques et les algorithmes de traitement. Ces derniers prennent des rôles divers. Ils peuvent avoir pour but d'améliorer la qualité de l'image, comme le HDR (High-Dynamic Range imaging) [Man+95], qui combine des images avec des temps d'exposition différents afin d'améliorer le contraste à la fois sur les zones très claires et très sombres. On augmente ainsi la capacité dynamique du détecteur. De cette façon, des défauts liés au capteur peuvent être compensés par des traitements qui permettent en retour l'utilisation de capteurs miniaturisés et/ou peu onéreux, comme dans le cas des téléphones portables dotés d'appareil photos.

Les algorithmes permettent également d'obtenir de nouvelles fonctionnalités, souvent à partir de systèmes simples, tels que l'obtention d'imagerie polarimétrique sans polariseur grâce à une matrice de miroirs et deux détecteurs mono-pixel, [FPB18].

Certains traitements sont quant à eux indissociables de l'acquisition de l'image, comme c'est le cas pour l'imagerie plénoptique [Ng+05]. Elle utilise des matrices de lentilles pour obtenir des images de la même scène avec une légère parallaxe. Le traitement de ces séquences d'images permet alors d'obtenir des cartes du relief ou de modifier la mise au point en post-traitement.

Le travail effectué durant cette thèse se situe dans ce contexte, à savoir le traitement d'images dans un but de compensation des limites d'utilisation d'un instrument et par voie de conséquence, l'ajout d'une fonctionnalité à l'instrument.

Problématique de la thèse

L'Office National d'Etudes et de Recherches Aérospatiales (ONERA) a developpé un instrument imageur hyperspectral baptisé Sieleters. Celui-ci est embarqué sur un avion et restitue des images du rayonnement émis par la scène survolée pour une certaine longueur d'onde.

Le principe de fonctionnement de Sieleters engendre des erreurs lorsqu'il s'agit de restituer des scènes texturées avec des variations de relief. Un logiciel conçu pour l'ONERA permet de corriger ces erreurs dès lors que l'élévation de la scène est connue.

Cette thèse a donc pour but de fournir une solution afin de connaître le relief d'une scène survolée, en utilisant les images d'un spectro-imageur comme Sieleters.

Organisation de la thèse

Le manuscrit est divisé en trois parties indépendantes : une partie d'introduction avec les pré-requis à la compréhension du problème, une partie dédiée à un pré-traitement des images de Sieleters, et enfin une partie dédiée à la stéréoscopie. Chaque partie est subdivisée en trois chapitres.

Cette thèse a été pensée et testée pour les images d'un instrument spécifique : un spectro-imageur à transformée de Fourier statique (ISFTS). Elle propose donc une solution complète pour résoudre le problème spécifique de la correction des effets de relief sur les spectres de ce type d'instrument. Il faut toutefois préciser que bien qu'étant axée sur un type d'instrumentation spécifique, la thèse, et en particulier la partie III sur la stéréoscopie, peut être destinée à d'autres instruments hyperspectraux pour lesquels ce type de problème se pose aussi. Il s'agit par exemple d'autres appareils utilisant la spectro-imagerie par transformée de Fourier, ou même les filtres en coin.

Le **chapitre 1** détaille le concept instrumental de Sieleters. La nature de l'instrument est particulièrement importante car elle permet de comprendre les choix faits par la suite. La spécificité d'un instrument comme Sieleters est de fournir des images présentant des franges d'interférences, sous la forme de lignes horizontales, en superposition à la scène. De telles images sont en général traitées d'un point de vue instrumental et n'ont que très rarement eu l'occasion de faire l'objet de travaux en traitement d'image. Le **chapitre 3** prend donc le temps de présenter les images utilisées et les scènes survolées.

L'autre particularité des images est qu'elles nécessitent de traiter de la stéréoscopie sous-pixellique. De plus, le mode d'acquisition à trame défilante offre la possibilité de bénéficier de plusieurs centaines d'images de la scène. Pour ces raisons, le **chapitre 2** est consacré aux bases et à une étude bibliographique sur la stéréoscopie. Il justifie le choix de la méthode des images des plans épipolaires.

Décomposition d'image

La partie II présente un traitement préalable à la stéréoscopie nécessaire pour les ISFTS, puisque les franges d'interférence sur les images perturbent les traitements stéréoscopiques. Ce problème est mis en évidence en **chapitre 4** . On y trouve également une caractérisation classique des franges dans le domaine de Fourier.

Les deux chapitres suivants présentent chacun une solution de décomposition afin de séparer la texture des franges d'interférence et la scène. Le **chapitre 5** expose une méthode de décomposition additive par optimisation convexe. Plus classique, cette méthode reprend la philosophie de la décomposition Cartoon-Texture. Le **chapitre 6** utilise un modèle multiplicatif d'image. Inédit à notre connaissance dans la littérature, il demande de résoudre un problème d'optimisation bi-convexe complexe, mais a l'avantage d'être physiquement plus fidèle au modèle réel de formation d'image. La conclusion de cette partie compare les deux modèles de décomposition.

Stéréoscopie

La dernière partie répond au problème de l'obtention d'une carte de relief à partir des images obtenues au terme de la partie II. La méthode proposée est d'utiliser les images des plans épipolaires

(EPI) pour obtenir une carte éparse puis de la densifier. Seulement, nos images présentent deux différences par rapport à ce qu'a produit la littérature sur le sujet.

D'une part, l'acquisition des images est aéroportée, ce qui implique une distance caméra-objet très grande et des variations très faibles sur l'EPI (non constatables à l'oeil nu). Cela implique aussi que la trajectoire peut enregistrer des écarts sensibles et invalider l'hypothèse de déplacement rectiligne de la caméra. **Le chapitre 7** étudie donc l'impact de déviations en translation de la ligne de visée de l'avion sur les images de plans épipolaires et quantifie l'erreur résultante sur la mesure du relief. Ces résultats peuvent aussi être utilisés pour définir un cahier des charges de tolérances sur une trajectoire.

D'autre part, les effets de parallaxe observés sur les images sont sous-pixelliques. Le traitement des images des plans épipolaires, présenté au **chapitre 8**, utilise un algorithme sous-pixellique de détection de segments existant pour obtenir la carte éparse d'élévation. La précision de la méthode est quantifiée à partir d'une vérité terrain. L'étape finale de densification de la carte est réalisée par minimisation d'une énergie par coupure de graphe. Le traitement adapte un algorithme classique à des acquisition multi-vues et sous-pixelliques. Il fait l'objet du **chapitre 9**.

Contributions

Cette thèse apporte une solution au calcul d'un modèle numérique de terrain pour corriger les spectre (ou interférogrammes) de spectro-imageurs aéroportés à trame défilante en étudiant le cas particulier des ISFTS. Nous donnons ainsi à ces instruments une nouvelle fonctionnalité de stéréoscopie en plus de leur fonction hyperspectrale native.

Décomposition d'image. Nous proposons un modèle de décomposition multiplicatif inédit exploitant le domaine spectral de la composante de l'image à extraire. Dans ce cas comme celui de la décomposition additive proposée, une attention particulière est portée à la qualité des images obtenues en vue des traitements sous-pixelliques consécutifs. Nous utilisons par exemple des versions interpolables ou lisses de la variation totale.

Images des plans épipolaires. Nous étendons le champ d'application des EPI à des images aéroportées avec des disparités inférieures au centième de pixel entre deux images successives pour obtenir une carte de hauteur éparse. Nous apportons en plus une étude de l'impact des écarts à la trajectoire sur la mesure relief par cette méthode.

Densification d'une carte éparse. Nous proposons d'adapter une méthode courante de stéréoscopie par coupure de graphe afin de densifier le résultat des EPI. Pour cela nous accomodons l'algorithme à des disparités sous-pixelliques sur une séquence de prise de vues dense.

Publications

Articles

- D. C. Soncco, C. Barbanson, M. Nikolova, A. Almansa, and Y. Ferrec. “Fast and Accurate Multiplicative Decomposition for Fringe Removal in Interferometric Images”. In: *IEEE Transactions on Computational Imaging* 3.2 (2017), pp. 187–201. DOI: 10.1109/TCI.2017.2678279
- C. Barbanson, A. Almansa, Y. Ferrec, and P. Monasse. “Digital Elevation Model estimation with an airborne hyperspectral imager”. In: (2018) (en cours)

Communications

- C. Barbanson, A. Almansa, Y. Ferrec, and P. Monasse. “Reconstruction du relief par images des plans épipolaires à partir d’acquisitions aériennes denses”. In: *SFPT 2016 Colloque « Photogrammétrie Numérique et Perception 3D : les Nouvelles Conquêtes »*. Marne La Vallée, France, 2016
- C. Barbanson, A. Almansa, Y. Ferrec, and P. Monasse. “Relief Computation from Images of a Fourier Transform Spectrometer for Interferogram Correction”. In: *Light, Energy and the Environment*. Washington, D.C.: OSA, 2016, FM3E.6. DOI: 10.1364/FTS.2016.FM3E.6
- C. Barbanson, A. Almansa, Y. Ferrec, and P. Monasse. “Reconstruction 3D par Images des Plans Epipolaires : Application à un interférométre imageur hyperspectral aéroporté”. In: *Gretsi 2017 - XXVIeme colloque*. 2017
- C. Barbanson, A. Almansa, Y. Ferrec, and P. Monasse. “Correction des effets de relief en spectro-imagerie aéroportée”. In: *GDR-ISIS - Co-conception : capteurs hybrides et algorithmes pour des systèmes innovants*. 2017

Première partie

Préliminaires

CHAPITRE 1

SIELETERS, UN INTERFÉROMÈTRE IMAGEUR AÉROPORTÉ

Sommaire

1.1 Imagerie hyperspectrale **8**

1.2 Généralités sur la spectrométrie par transformée de Fourier **9**

1.2.1 Principe de la spectrométrie par transformée de Fourier 9

1.2.2 Avantages et inconvénients 10

1.3 Sieleters **11**

1.3.1 Contexte 11

1.3.2 Concept instrumental 11

1.3.3 Reconstruction des spectres et problèmes dus au relief 14

1.4 Instruments existants **15**

1.5 Conclusion **17**

Les traitements stéréoscopiques développés au cours de cette thèse ont pour but d'être appliqués à la classe des spectro-imageurs à trame défilante. Cela comprend les instruments à filtre en coin ou à transformée de Fourier, et implique des acquisitions denses d'images en trajectoire rectiligne.

Dans le cadre de ce travail, nous nous intéressons plus spécifiquement au cas de Sieleters, un spectro-imageur à transformée de Fourier statique. Afin de permettre au lecteur de mieux comprendre les hypothèses posées et les choix faits dans ces travaux, ce chapitre est dédié à la compréhension du fonctionnement de cet instrument.

La première section est dédiée à l'imagerie hyperspectrale et ses applications, afin de présenter l'intérêt du développement de tels instruments. La section suivante fournit des généralités sur la spectrométrie par transformée de Fourier ainsi que ses avantages et inconvénients pour un instrument imageur. La suite se concentre sur le cas spécifique de Sieleters, pose le contexte de son développement, évoque son concept instrumental et le problème de reconstruction des spectres, objet de cette thèse. Enfin, la dernière section cite quelques exemples d'instruments et d'autres technologies pour lesquels tout ou partie des éléments de ce travail peuvent être utiles.

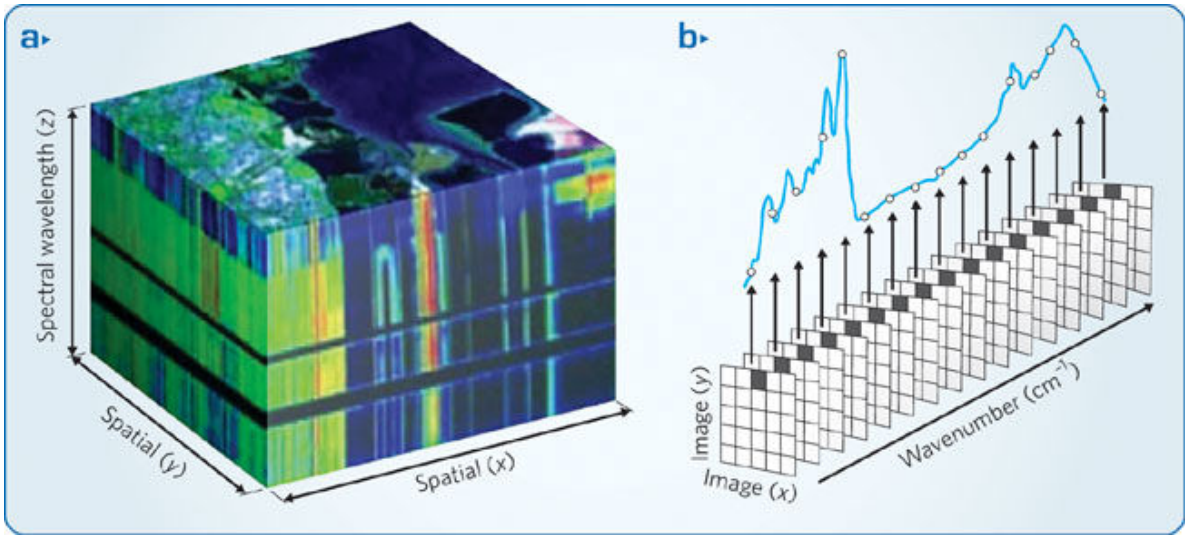


FIGURE 1.1 – **Imagerie hyperspectrale. -** (a) Cube de données hyperspectrales avec les coordonnées spatiales (x,y) et la coordonnée spectrale (z). Dans ce cube, tous les plans perpendiculaires à $\vec{z}$ correspondent à une image de la scène à une longueur d'onde donnée. (b) Spectre associé à un pixel. *Source : doi :10. 1038/ nphoton.2009.205*

1.1 Imagerie hyperspectrale

Le spectre électromagnétique d'un objet correspond au rayonnement émis, transmis, réfléchi ou diffusé par celui-ci en fonction de la longueur d'onde. On qualifie un instrument d'hyperspectral lorsqu'il fournit des mesures sur plusieurs dizaines ou centaines d'intervalles spectraux. En imagerie, on observe donc une scène dont on mesure les spectres des éléments contenus dans chacun des pixels. On parle alors de cube hyperspectral suivant les deux dimensions spatiales de l'image de la scène et la dimension spectrale. Ce principe est illustré en figure 1.1.

Les domaines spectraux généralement concernés par l'imagerie aéroportée sont le domaine visible (VIS : $0,35 - 0,78\mu m$), le proche infrarouge (NIR[1] : $0,7 - 1,4\mu m$), l'infrarouge court (SWIR[2] : $1,4 - 3\mu m$), l'infrarouge moyen (MWIR[3] : $3 - 5\mu m$) et plus rarement l'infrarouge lointain (LWIR[4] : $8\mu m - 14\mu m$). Les rayonnements à ces longueurs d'onde sont peu absorbés par l'atmosphère, ce qui rend possible leur observation par avion, voir figure 1.2. On s'intéresse dans cette thèse au MWIR et LWIR, que l'on appellera respectivement bande 2 et bande 3.

L'infrarouge, puisqu'il informe sur l'émission propre du corps étudié, est avantageux pour la télédétection. En géologie, ce domaine spectral est utilisé pour la prospection minière [Sal+], ou la détermination de la composition des sols [Tav+08]; dans le registre environnemental, pour l'identification des zones en voie de désertification [Eis+15] ou l'évaluation de la pollution de l'eau [Mar+02]; en vulcanologie, pour l'analyse des nuages de fumée [Spi+08]. On utilise aussi les bandes infrarouges dans la classification de l'occupation des sols pour des thématiques d'urbanisme [CK16] ou d'identification des essences végétales [DBG08]. Il est également intéressant de

[1] Near InfraRed
[2] Short Wavelength InfraRed
[3] Mid-Wavelength InfraRed
[4] Long Wavelength InfraRed

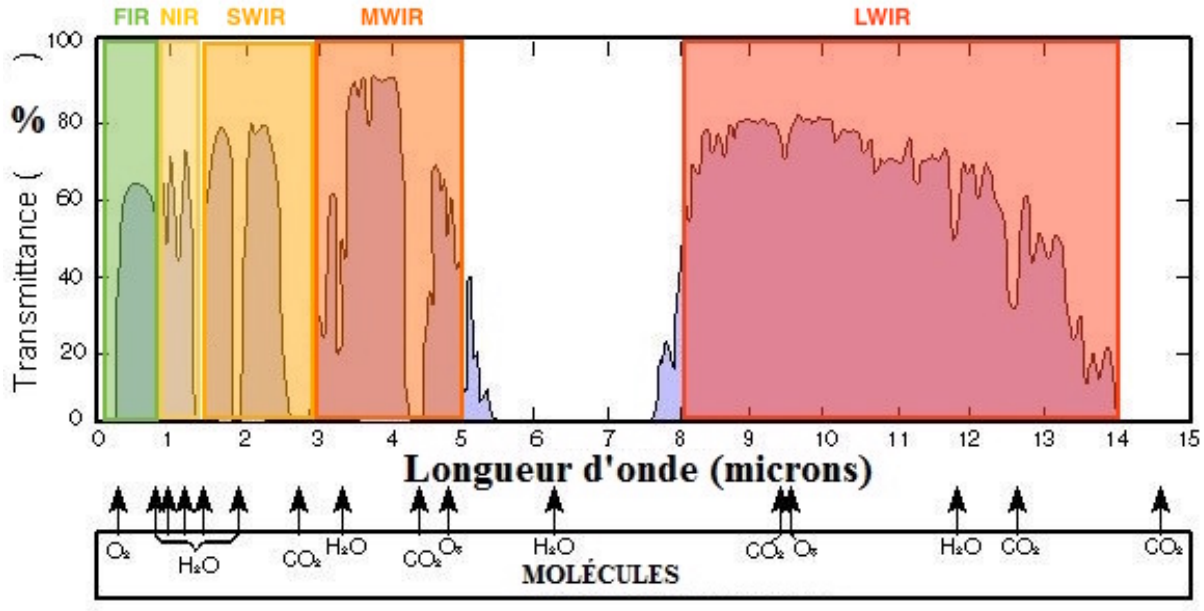


FIGURE 1.2 – **Transmittance de l'atmosphère en fonction de la longueur d'onde. -** L'absorption par la vapeur d'eau et par le dioxide de carbone définit les fenêtres atomsphériques. *Source : https:// fr.wikipedia. org/ wiki/ Fenêtre_atmosphérique*

citer des travaux de photo-interprétation dans le domaine militaire [Bri+06]. De nombreux autres exemples existent, cette liste n'étant qu'un aperçu des applications possibles.

1.2 Généralités sur la spectrométrie par transformée de Fourier

Cette section donne des bases de compréhension sur le principe de la spectrométrie par transformée de Fourier, qui est à l'origine de la conception d'un certain nombre de systèmes optiques hyperspectraux, dont celui de Sieleters. Afin d'avoir plus de détails, le lecteur peut se référer à la partie 1 de [Fer08].

1.2.1 Principe de la spectrométrie par transformée de Fourier

La spectrométrie par transformée de Fourier (FTS) s'appuie sur la théorie des interférences lumineuses. Deux ondes cohérentes monochromatiques de même fréquence, lorsqu'elles sont superposées, forment ce que l'on appelle des interférences lumineuses. Dans ces conditions, l'intensité observée est une fonction sinusoïdale dépendant de la différence de marche, qui est la différence entre les chemins optiques des deux rayons depuis la source jusqu'au lieu d'interférence. On observe alors ce que l'on appelle des franges d'interférences. On peut construire l'interférogramme de l'onde, qui enregistre, en fonction de la différence de marche, les valeurs d'intensité de la somme des deux ondes. Si de plus les deux ondes ont la même amplitude (ce qui est souvent le cas pour les systèmes interférentiels utilisés en imagerie hyperspectrale), l'équation d'interférences

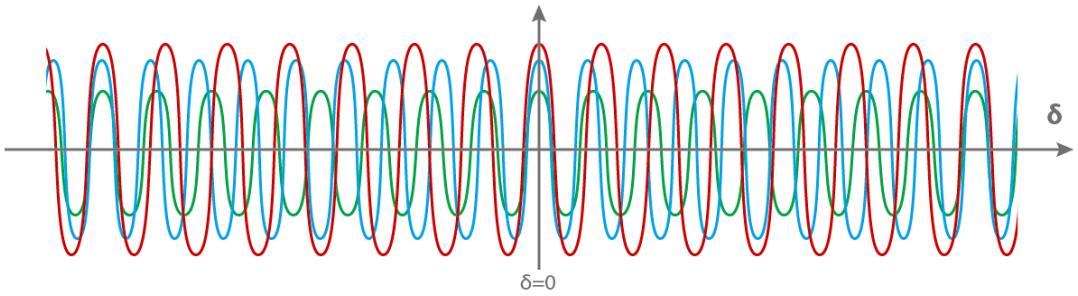


FIGURE 1.3 – **Superposition des interférences. -** Chaque onde génère son propre système d'interférences. A la différence de marche nulle, toutes les ondes sont en phase donc les franges sont contrastées. Lorsque l'on s'éloigne, les figures ne se superposent plus, et l'on ne distingue plus de franges. Elles sont brouillées. Il peut arriver que certaines interférences coïncident à nouveau, ce qui peut fournir des franges légèrement contrastées au-delà de l'origine. C'est ce que l'on voit sur la dernière oscillation du schéma.

entre deux ondes monochromatiques d'intensité I_0, de longueur d'onde $\lambda = \frac{1}{\sigma}$ (σ est le nombre d'onde, en cm^{-1}) est la suivante :

$$I(\delta) = 2I_0(1 + \cos(2\pi\sigma\delta))$$

δ est la différence de marche (en mètres). L'expression de δ indique la forme de la figure d'interférences.

Lorsque les ondes ne sont plus monochromatiques, chaque longueur d'onde génère son système d'interférences, et tous se superposent. Les franges éloignées de la différence de marche zéro sont alors brouillées. C'est ce qui est schématisé en figure 1.3.

L'intensité des interférences en présence d'une source de largeur spectrale $[\sigma_1; \sigma_2]$ s'écrit :

$$I(\delta) = \frac{1}{2}\int_{\sigma_1}^{\sigma_2} B(\sigma)(1 + C\cos(2\pi\delta\sigma + \phi(\sigma)))d\sigma \tag{1.2}$$

$B(\sigma)$ correspond à la luminance spectrale pour chaque longueur d'onde, C au contraste des franges et ϕ à la phase des franges. Le terme $B(\sigma)\cos(2\pi\delta\sigma)$ exprime que l'on effectue physiquement une transformée de Fourier, d'où la dénomination spectrométrie par transformée de Fourier. La mesure de la figure d'interférences est un interférogramme de la source. On obtient donc le spectre en faisant la transformée de Fourier inverse de l'interférogramme.

1.2.2 Avantages et inconvénients

L'un des principaux intérêts de la spectrométrie par transformée de Fourier est le gain radiométrique dont elle bénéficie. En effet, l'ensemble du flux incident est utilisé pour la mesure, contrairement à d'autres techonologies comme celles basées sur des filtres, où l'augmentation de la résolution spectrale diminue fortement la quantité de photons utilisés pour la mesure. On parle de gain multiplex ou d'avantage de Fellgett.

D'autre part, l'échantillonnage du spectre est déterminé par le domaine de l'interférogramme. Il est donc possible d'obtenir une meilleure résolution spectrale uniquement en augmentant les bornes de l'intervalle de différence de marche mesuré.

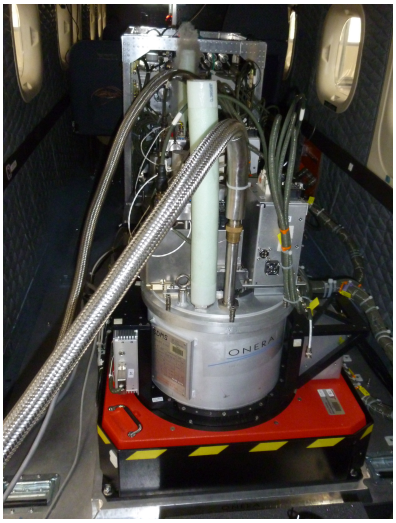


FIGURE 1.4 – **Sieleters BIII monté sur plateforme stabilisée.**

Cependant, plusieurs inconvénients existent. D'une part, les montages et la physique de formation de l'image sont complexes, en particulier si on les compare à un détecteur à filtre. De plus, l'obtention du spectre ne se fait pas par mesure directe. Il faut passer par une étape de traitement du signal, plus ou moins complexe selon l'architecture instrumentale, pour y avoir accès. Enfin, dans le cas des interféromètres dits statiques, bien que l'on puisse augmenter la résolution spectrale, celle-ci est limitée par la taille du détecteur, qui doit être assez grand pour contenir toute la figure d'interférences.

1.3 Sieleters

1.3.1 Contexte

L'Onera (Office National d'Etudes et de Recherches Aérospatiales) a récemment développé un instrument aéroporté d'acquisition d'images hyperspectrales pour observer les bandes 2 (BII : $3 - 5.3\mu m$) et 3 (BIII : $8 - 11.5\mu m$) de l'infrarouge, baptisé Sieleters [Ca15] dont la photo est en figure 1.4. Chaque bande a respectivement une résolution spectrale de $13\mathrm{cm}^{-1}$ et $6\mathrm{cm}^{-1}$, ce qui correspond à 120 et 60 bandes spectrales étroites et contigües. Sieleters enregistre des images avec un pas d'échantillonnage au sol de $50\mathrm{cm}$ sur une fauchée de $500\mathrm{m}$ (1000 pixels) à $2000\mathrm{m}$ de hauteur de vol. Il fait partie de la plateforme SYSIPHE, qui intègre également HySpex Odin-1024, développé par NEO (Norsk Elektro Optikk), qui opère dans les bandes spectrales visible et proche-infrarouge.

1.3.2 Concept instrumental

Sieleters est un spectromètre imageur à transformée de Fourier avec un interféromètre statique.

La fonction spectrale de Sieleters est exécutée par un interféromètre à décalage latéral réalisé avec un interféromètre de Michelson à dièdres. Le schéma de principe d'un tel montage est sur la

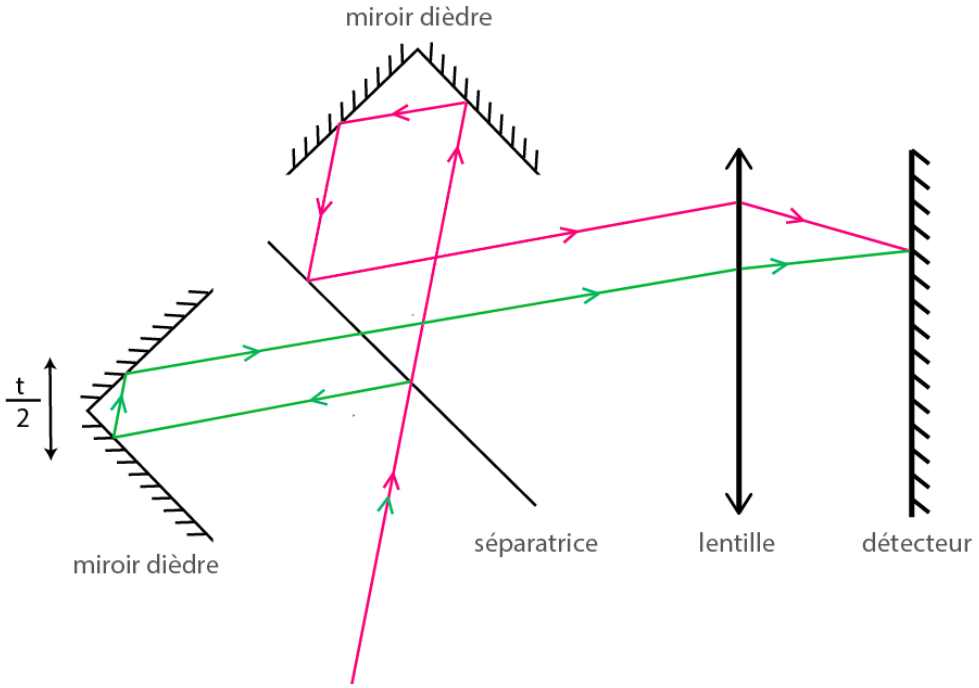


FIGURE 1.5 – **Schéma optique de l'interféromètre de Sieleters. -** Il s'agit d'un interféromètre de Michelson à dièdres de décalage latéral t (par rapport à une configuration symétrique des deux miroirs selon la séparatrice). Le rayon entrant est divisé par la séparatrice, chaque rayon résultant est réfléchi par un dièdre et une lentille en sortie les fait converger en un même point du détecteur.

figure 1.5 . Un rayon incident est divisé en deux par la séparatrice et arrive sur chacun des miroirs dièdres, qui renvoient des ondes translatées en accord avec le décalage posé entre les arêtes des deux miroirs. Une lentille placée en aval permet de faire converger les deux rayons sur le détecteur. On obtient alors l'image de la scène modulée par la figure d'interférences.

En reprenant l'équation 1.2, l'intensité I pour une bande spectrale $[\sigma_1, \sigma_2]$ en chaque point de coordonnées (x, y) de l'image s'écrit :

$$I(x,y) = \int_{\sigma_1}^{\sigma_2} B(x,y,\sigma)(1 + C\cos(2\pi\sigma\delta(x,y) - \phi(\sigma)))d\sigma \tag{1.3}$$

où $B(x, y, \sigma)$ est la luminance d'un objet résolu spatialement et spectralement, C est le contraste des franges.Les perturbations liées à la phase des franges d'interférences $\phi(\sigma)$ fluctuent peu et peuvent donc être négligées dans le cadre de notre travail. On adopte donc l'équation de formation d'image suivante :

$$I(x,y) = \int_{\sigma_1}^{\sigma_2} B(x,y,\sigma)(1 + C\cos(2\pi\sigma\delta(x,y)))d\sigma \tag{1.4}$$

Avec les notations de la fig 1.6, la différence de marche δ s'exprime :

$$\delta = t\sin(\theta)$$

Les franges obtenues sont des franges d'égale inclinaison localisées à l'infini. Du fait que le décalage de l'interféromètre ne se fait que dans une direction précise (indiquée en figure 1.6), elles

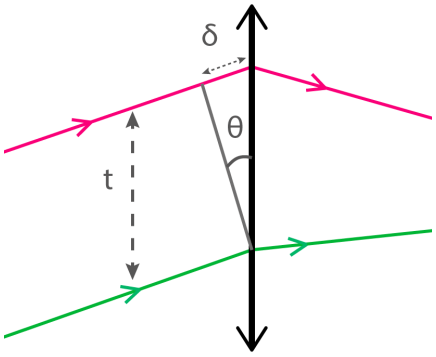


FIGURE 1.6 – **Figure de calcul de la différence de marche. -** Le décalage entre les deux dièdres est $\frac{t}{2}$, ce qui après réflexion sur le dièdre donne un décalage t entre les deux branches de l'interféromètre. δ est la différence de marche.

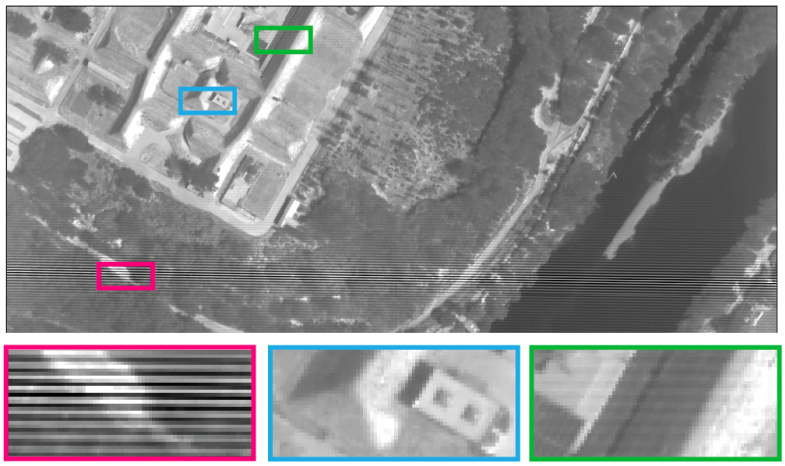


FIGURE 1.7 – **Image issue de Sieleters. -** Les franges du carré rose, autour de $\delta = 0$ sont très contrastées. Elles sont brouillées sur le carré bleu en raison de la grande ouverture spectrale de l'instrument. Puis, vers le haut de l'image, dans le carré vert, les franges sont à nouveau contrastées mais plus légèrement.

sont quasiment rectilignes. On peut observer une image obtenue avec Sieleters en figure 1.7. On observe dans le carré rose des franges très contrastées. Il s'agit du voisinage de la différence de marche nulle. En revanche, les franges sont brouillées sur une partie de l'image. Cela s'explique par le fait que le domaine spectral observé est large. Chaque longueur d'onde génère son système d'interférences, et tous se superposent. On observe sur le haut de l'image un retour à des franges légèrement contrastées.

Chaque ligne de l'image correspond à une différence de marche (δ) différente, donc à un état d'interférences différent. L'interféromètre n'a pas de partie mobile, les différences de marche sont donc fixes. On parle alors d'interféromètre statique. Pour obtenir l'interférogramme d'un point au sol, on utilise le déplacement de l'avion, sur lequel est embarqué l'instrument. Une cadence d'acquisition élevée permet d'assurer que tout point de la scène est vu sous tous les états d'interférence. Il faut alors faire correspondre les différentes images pour reconstruire l'interférogramme de chacun des objets observés. Ce principe est illustré en figure 1.8.

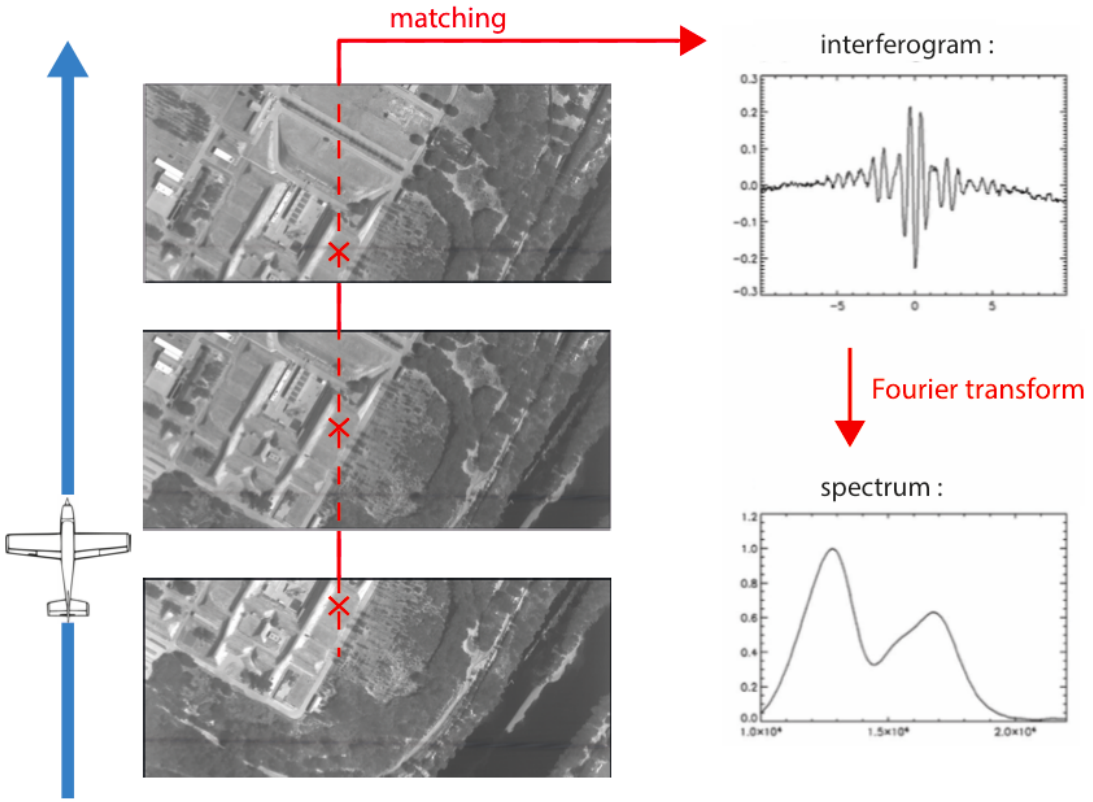


FIGURE 1.8 – **Principe d'acquisition hyperspectrale de Sieleters. -** Le déplacement de l'avion (indiqué par la flèche bleue) permet aux éléments de la scène d'être vus sous différents états d'interférence. En associant les différentes valeurs prises par un objet (désigné par une croix rouge) sur chaque image, on accède à son interférogramme. Le spectre est obtenu ensuite en appliquant la transformée de Fourier.

Dans le cas de Sieleters, les interférogrammes extraits ont environ 424 et 440 échantillons respectivement pour la bande 2 et 3.

1.3.3 Reconstruction des spectres et problèmes dus au relief

La reconstruction du spectre telle qu'elle vient d'être expliquée présente une difficulté supplémentaire lorque l'on survole des zones au relief marqué (bâtiments, collines, végétation haute, par exemple). En effet, dans le modèle de caméra à projection centrale, à mesure que l'on s'éloigne du centre de l'image, et que l'angle de visée augmente, on observe un défaut de parallaxe entre les différentes images. De ce fait, la position des éléments surélevés n'est pas conforme à leur position planimétrique réelle, comme le schématise la figure 1.9.

Dans le cas de Sieleters, ce phénomène a pour conséquence de faire correspondre des éléments différents de l'image. Alors, l'interférogramme extrait est incorrect, comme l'interférogramme rouge de la figure 1.10. Sur des zones marquées d'un relief, en général au-delà de 4 mètres (soit un pixel de décalage), les spectres fournis par l'instrument peuvent donc être faux. Il est donc nécessaire de connaître le modèle numérique d'élévation de la zone survolée pour le traitement des images Sieleters.

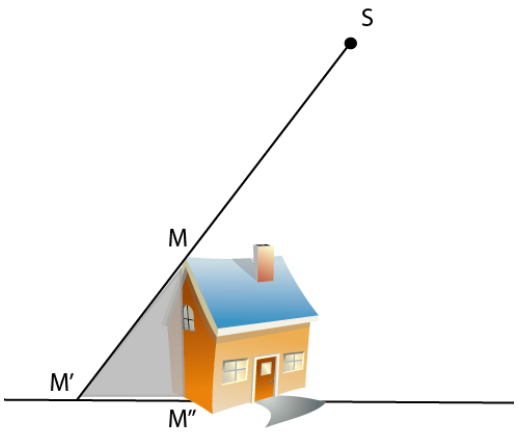


FIGURE 1.9 – **Effet du relief. -** Le sommet de prise de vue (S) observe le point M et lui attribue la position M' au lieu de sa position réelle M". La zone grisée est invisible pour le capteur, on parle d'occlusion.

1.4 Instruments existants

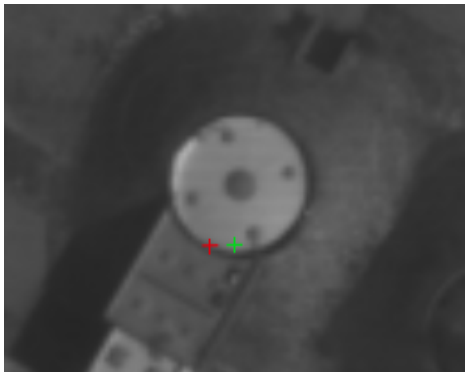
Parmi les instruments aéroportés existants, nous en évoquons ici plusieurs pour lesquels les travaux que nous avons mené sont applicables.

Le THI de l'Université de Hawai'i [Luc+12] utilise un interféromètre de Sagnac et observe l'infrarouge lointain. L'ONERA a également développé SiBi [Fos+16], un spectromètre imageur miniaturisé qui emploie un interféromètre biréfringent à décalage latéral. Il a pour vocation d'être utilisé au sol ou sur drone. Il y a également LASIS, qui appartient au Xi'an Institute of Optics and Precision Mechanics et qui est un instrument dont le concept est semblable à celui de Sieleters [Wan+15], mais qui opère dans le domaine visible. Ces trois instruments fournissent des images avec des franges, comme Sieleters.

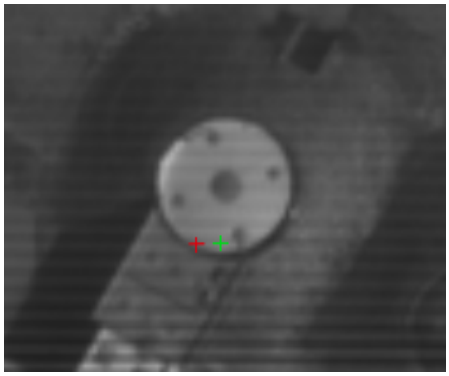
On aussi peut citer l'Hypercam de Telops [Cha+09], qui utilise un interféromètre mobile balayant la scène à des différences de marches variables : celle-ci est donc différente pour chaque image. Dans ce cas, c'est un miroir pivotant qui assure le pointage sur la même scène pendant l'intervalle de temps nécessaire pour balayer toute la plage de différences de marche. Les images n'ont dans ce cas pas de franges d'interférences, puisqu'on n'y observe qu'un seul état d'interférence. Le problème de la reconstruction des spectres en zone de relief se pose néanmoins là encore puisque la géométrie d'acquisition n'est pas tout à fait la même sur chaque image du fait que l'avion se déplace au cours de l'acquisition.

Les instruments cités jusqu'ici sont des spectromètres à transformée de Fourier et le problème évoqué se situe sur la reconstruction des interférogrammes. Cependant, on peut aussi élargir l'application de la partie III aux instruments à filtre en coin. Le principe d'acquisition est cette fois de placer des filtres passe-bande différents devant chaque ligne du détecteur. Le spectre est reconstruit directement à partir des valeurs mesurées, en faisant correspondre les pixels sur les différentes images.

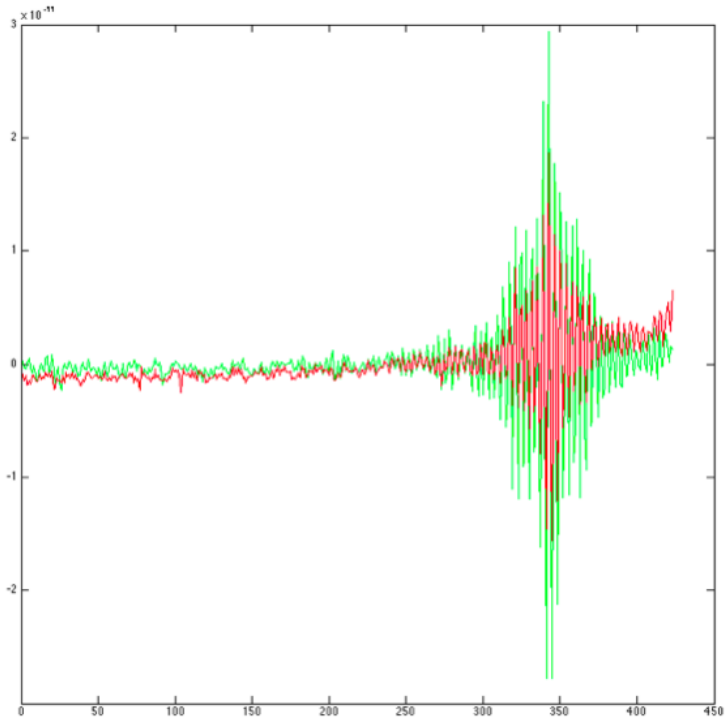
Cette liste n'est pas exhaustive, et d'autres instruments pourraient être ajoutés. L'idée ici est seulement d'évoquer le fait que bien que cette thèse ait été pensée pour Sieleters, d'autres instruments pourraient directement bénéficier des traitements proposés.



(a) Image en début de séquence



(b) Image en fin de séquence



(c) Interférogrammes

FIGURE 1.10 – **Interférogramme mal reconstruit sur un bâtiment. -** (a) et (b) montrent des détails d'images du début et de la fin de la séquence acquise par Sieleters. Ces images ont été recalées en faisant l'hypothèse d'un sol plat. Les interférogrammes en (c) correspondent aux croix rouge et verte sur les imagettes. Ces croix sont situées sur le haut d'un bâtiment et chacune pointe le même pixel dans les deux images. La croix rouge, en bord de toit, pointe sur le bord noir dans (a) et sur l'intérieur blanc du toit en (b) en raison de la parallaxe. L'interférogramme rouge est donc incorrect, et c'est effectivement ce que l'on observe en le traçant. L'interférogramme vert est correct car il se trouve sur une zone homogène.

1.5 Conclusion

L'explication du principe de fonctionnement de Sieleters dans ce chapitre permet de mieux comprendre la problématique de la thèse. D'une part, la raison de la présence de franges sur les images, dans un intérêt de mesure spectrale, est expliquée. Il est clair qu'elles ne peuvent être traitées comme des images "classiques". D'autre part, l'obtention d'un modèle numérique d'élévation (MNE) des zones survolées par Sieleters est nécessaire pour pouvoir obtenir tout le potentiel de l'instrument en zone de relief. Ce deuxième constat s'applique par ailleurs à l'ensemble des spectro-imageurs aéroportés à trame défilante. Trois solutions apparaissent :

- Acheter les MNE des zones d'intérêt. Cette solution est simple mais coûteuse, d'autant plus que bien que des modèles numériques de terrain à la précision requise soient disponibles facilement à l'achat via l'Institut Géographique National, ceux-ci ne concernent que la France, et ne contiennent pas le sursol, qui est capital dans le cas présent.
- Ajouter un LiDAR (Light Detection and Ranging) à l'avion. Le LiDAR est une technologie très efficace pour obtenir des MNE précis. Cependant, un tel instrument a un coût (horaire et pécuniaire), et SYSIPHE intègre déjà trois instrument : ODIN, Sieleters II et Sieleters III, ces deux derniers étant d'autant plus encombrants du fait qu'ils sont refroidis. Il n'y qu'une poignée d'avions avec une trappe suffisamment grande pour contenir ces trois instruments, donc en ajouter un autre n'est pas envisageable.
- Utiliser des algorithmes de stéréoscopie sur les images de Sieleters pour obtenir le MNE de la scène survolée. Cette méthode présente évidemment un problème lié à la présence de franges sur les images. De surcroît, les disparités observées sont sous-pixelliques, de l'ordre du centième de pixel, entre deux images successives.

La solution la plus adéquate est toutefois la troisième. Elle nécessite de développer des algorithmes de stéréoscopie sous-pixellique adaptés à des images avec franges d'interférences. Ceci constitue donc le travail de thèse décrit dans ce manuscrit.

Afin de mieux appréhender cette question de la stéréoscopie, le chapitre suivant fournit des généralités à ce sujet.

CHAPITRE 2

LA STÉRÉOSCOPIE

Sommaire

2.1 Généralités **19**

2.1.1 Modèle de caméra 20

2.1.2 Triangulation 21

2.2 Etat de l'Art sur la stéréoscopie. **22**

2.2.1 Stéréoscopie à deux vues. 23

2.2.2 Stéréoscopie multi-vues. 29

2.3 Conclusion **34**

La stéréoscopie est le principal phénomène sur lequel repose la perception de la profondeur chez l'homme. En effet, lorsque l'on observe une scène, chacun de nos yeux la voit depuis un point de vue légèrement différent. Ce décalage entre nos yeux suffit à introduire une parallaxe. En imagerie, elle est obtenue à partir de deux clichés pris à des emplacements différents. La stéréovision a fait l'objet de nombreuses études et la littérature sur le sujet est donc très vaste. Ce chapitre a pour but de faire un tour d'horizon des méthodes existantes afin de justifier le choix des algorithmes utilisés par la suite. Il commence par fournir quelques généralités sur la mesure stéréoscopique. La deuxième section expose les principaux algorithmes à deux vues, dont les principes sont repris sur certains algorithmes multi-vues. Cette dernière catégorie est celle qui nous intéresse plus particulièrement, et un état de l'art en est présenté pour finir.

2.1 Généralités

De même que le fait d'avoir deux yeux nous permet de voir en trois dimensions, il est possible de mesurer la hauteur d'un point au sol dès lors qu'il est observé sur deux clichés aux sommets d'image non confondus. Les notions qui suivent expliquent ce principe et peuvent être retrouvées avec plus de détails dans [HZ04].

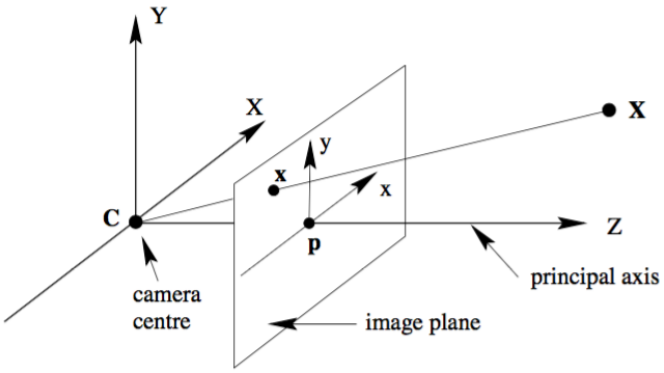


FIGURE 2.1 – **Schéma du modèle de caméra sténopé. -** *Source : [HZ04]*

2.1.1 Modèle de caméra

Le modèle d'imagerie utilisé est le modèle sténopé, illustré figure 2.1. Ce modèle projette la scène sur un plan image par projection centrale. Le centre de la projection est noté C.[1] L'axe optique, ou axe principal, intersecte le plan image au point P appelé point principal. La distance CP est notée p et s'appelle distance principale. En général, elle correspond à la distance focale f.

On peut écrire cette projection de la forme suivante, où les vecteurs $\mathbf{x} = (x\ y\ 1)^T$ de l'image, et $\mathbf{X} = (X\ Y\ Z\ 1)^T$ de la scène sont écrits en coordonnées homogènes :

$$\mathbf{x} = P\mathbf{X} \tag{2.1}$$

P, la matrice de projection a la forme suivante :

$$P = K[R|t] \tag{2.2}$$

K correspond à la matrice de calibration, ou orientation interne. Elle contient les paramètres intrinsèques de la caméra :

- f_x, f_y : Distance focale. Il peut arriver que $f_x \neq f_y$ pour des capteurs dont les pixels ne sont pas carrés,
- P_x, P_y : Coordonnées du point principal,
- s : Inclinaison (skew), pour des axes d'image non orthogonaux. En général $s = 0$.

$$K = \begin{pmatrix} f_x & s & P_x \\ 0 & f_y & P_y \\ 0 & 0 & 1 \end{pmatrix} \tag{2.3}$$

R et t sont les paramètres (rotation et translation) de l'orientation externe de la caméra, c'est-à-dire sa position et son orientation dans le référentiel de la scène.

Ce modèle de caméra est appliqué dans l'ensemble de cette thèse.

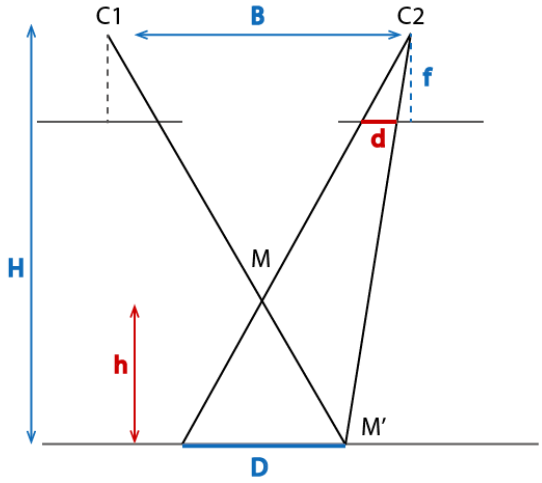


FIGURE 2.2 – **Schéma de principe pour la stéréovision après prise en compte du déplacement de la caméra de C1 à C2. -** C1 et C2 sont les sommets des images , B la base, H la hauteur de vol, f la focale, d la disparité et h la hauteur du point M que l'on cherche à déterminer. M' correspond à la reprojection au sol de M via C1.

2.1.2 Triangulation

Le principe de la stéréovision s'explique aisément dans la configuration de la figure 2.2. On obtient par triangles semblables la relation suivante :

$$\frac{B}{D} = \frac{H-h}{h} \tag{2.4}$$

Sous l'hypothèse $H \gg h$, acceptable pour l'imagerie aéroportée par avion, on a :

$$h \simeq \frac{D}{B/H} \tag{2.5}$$

D'autre part, la disparité, notée d, mesure sur l'image le déplacement du point M entre les deux images (après prise en compte du déplacement de la caméra de C1 à C2, donc $d = 0$ pour un point du sol). On trouve :

$$d = \frac{fD}{H} \tag{2.6}$$

On peut alors relier la disparité à la hauteur du point correspondant :

$$\frac{h}{H} \simeq \frac{d}{f}\frac{1}{B/H} \tag{2.7}$$

L'équation 2.5 met en évidence le rapport de la base sur la hauteur de vol. Ce quotient décrit la configuration de l'acquisition. En effet, un grand $\frac{B}{H}$ assure traditionnellement une plus grande précision sur l'estimation de la hauteur mais modifie davantage la géométrie des images successives par l'augmentation du dévers et des occultations. En revanche, un faible $\frac{B}{H}$ peut fournir des images plus rapprochées, qui faciliteront la mise en correspondance des éléments de la scène. L'estimation est en revanche moins précise si la résolution spatiale de l'image n'est pas assez fine. Cela peut être compensé par une méthode sous-pixellique de calcul de disparité comme dans [DR07] ou [SAM12].

[1] Il est aussi noté O ou S et peut être également appelé centre optique ou sommet de prise de vue.

Géométrie épipolaire

Afin de pouvoir calculer la disparité, il faut être capable d'apparier des points entre les images. La géométrie épipolaire permet de transformer une paire d'images de telle sorte que sur les deux clichés, une même ligne (ou colonne, selon l'orientation) contienne les mêmes éléments de la scène. Cela simplifie ensuite grandement la recherche de correspondants, puisqu'alors la recherche s'effectue sur une seule dimension.

Plus concrètement, lorsque l'on prend deux clichés avec des centres d'image différents qui observent le même point $\mathbf{X}$, on peut construire le plan épipolaire défini par $\mathbf{X}$ et les deux sommets de prise de vue C et C' . Comme le montre la figure 2.3(a), le plan intersecte les deux clichés selon une ligne appelée ligne épipolaire, où sont projetés tous les points contenus dans le plan épipolaire. Ainsi, tous les éléments contenus dans une ligne épipolaire sur une des deux images se retrouveront sur la ligne épipolaire correspondante de l'autre image. Cela est mis en équation sous la forme de la contrainte épipolaire, qui lie $\mathbf{x}$, un point sur l'image I, et $\mathbf{x}'$, son correspondant sur l'autre image I'. On montre qu'il existe une matrice F telle que :

$$\mathbf{x}^T F \mathbf{x}' = 0 \tag{2.8}$$

F est appelée matrice fondamentale. Elle est unique, à un facteur multiplicatif près, pour une paire d'images. Elle définit aussi les lignes épipolaires associées à chaque point :

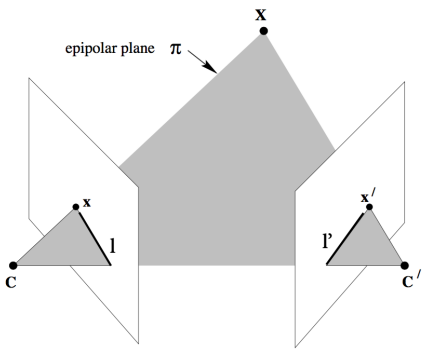
- $l = F\mathbf{x}'$ est la ligne épipolaire sur I associée au point $\mathbf{x}'$,
- $l' = F^T\mathbf{x}$ est la ligne épipolaire sur I' associée au point $\mathbf{x}$.

Toutes les lignes épipolaires d'une image s'intersectent à l'épipole, qui correspond aussi pour chaque image à l'intersection du plan qu'elle définit et de la droite reliant les deux sommets de prise de vue. L'ensemble des plans épipolaires d'un couple d'images est l'ensemble des plans contenant l'axe des épipoles, schématisés en figure 2.3.

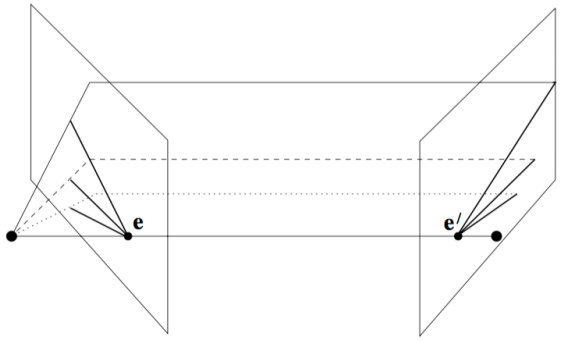
Le concept de géométrie épipolaire est fondamental dans la construction de nombreux algorithmes de stéréovision, que ce soit de manière plus traditionnelle, pour simplifier la recherche de correspondants, ou de manière plus avancée, pour déduire de nouveaux modèles de traitement. Ces différentes applications sont expliquées dans la suite de ce chapitre.

2.2 Etat de l'Art sur la stéréoscopie.

La stéréoscopie étant un thème de recherche ancien et dynamique, la littérature est très vaste. Plusieurs états de l'art ont par ailleurs été publiés. On peut citer [SS02] consacré aux algorithmes à deux vues, et qui a donné naissance également au *benchmark* Middlebury www.middlebury.edu/stereo permettant à chacun de tester son algorithme sur des jeux de données et d'en publier les résultats. Par la suite Middlebury ajoute un jeu de tests multi-vues http://vision.middlebury.edu/mview et publie [Sei+06], un état de l'art sur les problèmes de reconstruction multi-vues. On peut aussi citer [Sze] qui en 2011 fait un tour d'horizon exhaustif des algorithmes binoculaires et multi-vues.



(a)



(b)

FIGURE 2.3 – **(a) Géométrie epipolaire -** Les centres des deux clichés, C et C', et un point X de la scène constituent un plan épipolaire π. L'intersection entre ce plan et chacune des images est une ligne : les lignes épipolaires l et l'. **(b) Axe des épipoles -** Les épipoles e et e' définissent un axe. L'ensemble des plans contenant cet axe forment l'ensemble des plans épipolaires de la paire d'images. *Source : [HZ04]*

Cet état de l'art commence par présenter les principaux algorithmes à deux vues. Ceux-ci ne nous intéressent pas directement dans le cadre de ce travail, mais leurs principes sont repris dans nombre d'algorithmes multi-vues. La seconde partie est dédiée au traitement des vues multiples.

2.2.1 Stéréoscopie à deux vues.

L'approche classique, proposée dans l'état de l'art de [SS02] est de distinguer les méthodes selon que le résultat dépende d'un calcul local ou global. Les méthodes locales utilisent exclusivement le voisinage du pixel dont on veut déterminer la disparité tandis que les méthodes globales optimisent les valeurs sur l'ensemble de l'image en les liant entre elles.

Méthodes locales

Les premières méthodes appliquées étaient des méthodes locales. L'idée est en général de trouver un maximum de points correspondants entre les images et d'en mesurer ensuite la disparité. Ces méthodes fournissent souvent des cartes éparses qu'il faut ensuite densifier. Des mesures sous-pixelliques peuvent être obtenues par interpolation [BT98], [TH86].

- **Block matching**
  Une des méthodes les plus anciennes et les plus courantes dans la littérature est sans doute le *block-matching*. Elle consiste à chercher, pour chaque pixel, celui de radiométrie la plus proche sur la ligne épipolaire correspondante de la deuxième image. On mesure alors la disparité entre ces deux pixels. Afin d'obtenir une mesure plus robuste, ainsi que de fournir une certaine régularisation, on choisit plutôt une métrique sur le voisinage du pixel. On cherche alors à trouver la disparité qui maximise la similarité entre les mesures associées à une fenêtre autour des deux pixels à faire correspondre. Il existe de nombreux calculs possibles, l'un des plus classiques étant le SSD [OK90] (*sum of squared differences*), qui somme sur un voisinage $\nu(\mathbf{x_0})$ le carré des différences d'intensité entre un pixel $\mathbf{x_0} = (x_0, y_0)$ de l'image de gauche I et son correspondant à une certaine disparité $\mathbf{d} = (d, 0)$ sur l'image de droite I'.

$$SSD(\mathbf{x_0}, \mathbf{d}) = \sum_{\mathbf{x} \in \nu(\mathbf{x_0})} (I(\mathbf{x}) - I'(\mathbf{x} + \mathbf{d}))^2 \tag{2.9}$$

  D'autres coûts sont aussi fréquemment utilisés comme le SAD (*sum of absolute difference*) qui diffère du SSD en utilisant une norme $l1$ [Kan+95]. Le NCC (*normalised cross-correlation*) [KMS16] est normalisé, ce qui permet de s'affranchir des problèmes liés à des images avec des gains ou expositions différentes. [BT98] réalise une interpolation linéaire sur le pixel central avant le calcul du coût afin de compenser les erreurs d'échantillonnage.

  Le choix de la métrique n'est pas anodin et peut être différent en fonction du type d'image. [HS09] évalue la robustesse de différents coûts sur des images avec de fortes variations radiométriques. De plus, la taille du voisinage choisi est important : trop petit, il risque de manquer de texture et de ne pas être assez discriminant, trop grand il risque de n'être pas assez précis et de fournir des valeurs trop moyennées. Les approches multi-échelle (*coarse-to-fine*) permettent d'apporter un compromis [XTA87].

  Ce type d'algorithme nécessite en général une étape indépendante pour le traitement des occultations. Selon une idée largement reprise, [CCK91] propose de calculer deux cartes de disparités : l'une en cherchant les appariements depuis l'image de gauche sur l'image de droite et l'autre dans le sens opposé. Les deux résultats obtenus sont alors comparés sur les zones de fort dénivelé, puisque c'est là qu'ont lieu les occultations, qui donneront des valeurs différentes sur les deux images.

  S'intéressant particulièrement à l'exploitation des images avec un faible $\frac{B}{H}$, [DR07] calcule des NCC sous-pixelliques sur des images interpolées par *zero-padding* suivant les conditions de Shannon.

  Un défaut de ces méthodes est le problème d'adhérence. Il s'agit d'une dilatation autour de certains objets en relief avec une frontière très contrastée. Il arrive alors que lorsque la fenêtre de calcul contient une partie de l'objet et du sol, son meilleur correspondant dans la deuxième image sera une fenêtre similaire. L'algorithme néglige alors l'occultation et dilate l'objet. Ce phénomène est illustré en figure 2.4.

  Un autre défaut est l'erreur causée sur les appariements sur les zones homogènes ou lors de l'apparition d'éléments répétitifs. L'algorithme a alors du mal à choisir le bon appariement.

  La figure 6.1 montre le résultat obtenu sur deux images Sieleters par un algorithme de *block-matching* [Mon11] en calculant les similarités par SSD. Bien que l'algorithme propose un

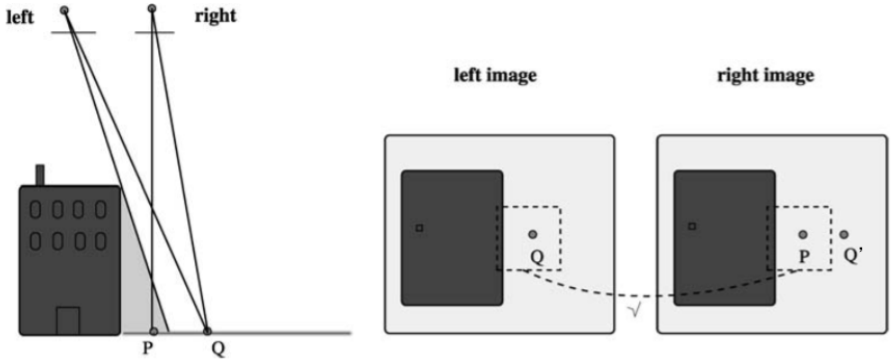


FIGURE 2.4 – **Illustration du phénomène d'adhérence. -** Les points P et Q étant radiométriquement similaires sur les images, ils sont appariés. Le point Q prend alors la même disparité que P. L'emprise du bâtiment est alors étendue. *Source : [DR07]*

raffinement sous-pixellique du quart de pixel, il n'arrive pas à gérer les zones homogènes et y génère beaucoup de bruit.

- **Feature matching**
  On peut obtenir des cartes éparses à partir d'éléments remarquables de l'image. C'est le principe du *feature matching*. Il peut s'agir du détecteur de coins Harris [Zha+95], des points SIFT (*Scale-Invariant Feature Transform*), robustes aux changements de points de vue par transformation affine et d'illumination [Low04]. Des éléments à une dimension sont également utilisés, comme des segments [AF87], [Jel+17] ou des contours d'objets [Mus+06]. [FM05] apparie des régions selon leur radiométrie et leur forme.

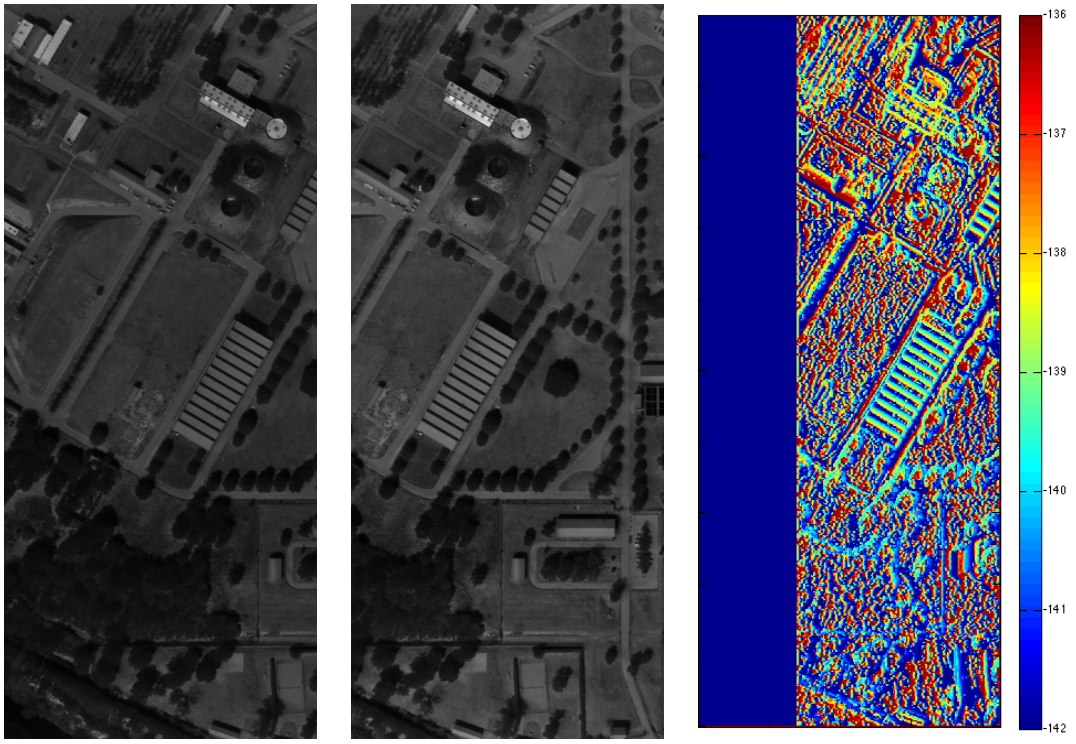
  Ces algorithmes fournissent des cartes éparses qui peuvent être par la suite complétées par des méthodes d'interpolation.

- **Phase de la transformée de Fourier**
  Le spectre croisé entre deux images fournit aisément la translation et la rotation par passage en coordonnées polaires [MBM97]. Les mesures doivent cependant être répétées sur des voisinages suffisamment grands pour contenir le même motif dont on calcule la transformée de Fourier, ce qui implique une perte de résolution spatiale et une régularisation implicite similaire au block-matching. La méthode reste cependant intéressante pour ses précisions sous-pixelliques [YL08], [FZB02].

- **Flot optique**
  Une carte de disparité peut être obtenue à partir du flot optique. Sur deux images d'une séquence d'une scène dynamique, le flot optique correspondant est le champ de vecteurs à deux dimensions du déplacement de chaque pixel de l'image. Sous l'hypothèse d'une illumination constante, on cherche alors pour tout pixel $\mathbf{x} = (x, y)$ son déplacement $(\delta x, \delta y)$ tel que :

  $$I(x, y, t_0) = I(x + \delta x, y + \delta y, t_1)$$

  On voit bien que si les images sont en géométrie épipolaire, alors $\delta y = 0$ et l'on se rapporte bien à un problème classique de stéréoscopie binoculaire. Ce problème est souvent résolu



(a) Image de gauche (b) Image de droite (c) Carte de disparité

FIGURE 2.5 – **Algorithme de block matching sur un couple d'images Sieleters. -** Les images (a) et (b) sont deux images d'une séquence Sieleters dont on a enlevé les franges (cf. Partie 2). (c) est la carte de disparité correspondante obtenue par l'algorithme de [Mon11]. L'intervalle de disparités est [-142 -137] pixels au quart de pixel près.

par des méthodes locales, qui ont l'inconvénient de manquer d'information dans les zones homogènes [BBM90; UWI11].

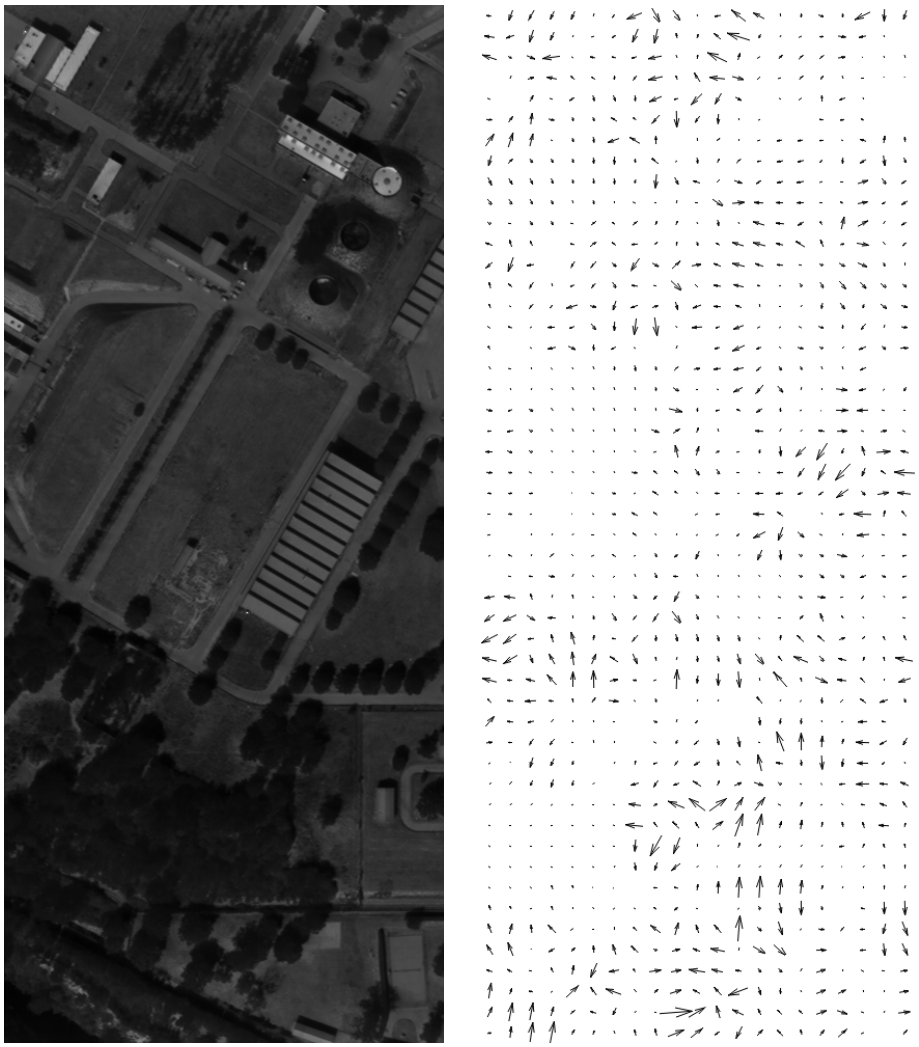
La figure 2.6 rapporte le résultat sur des images Sieleters de l'algorithme [MSS16] de flot optique. Malgré le bon comportement sous-pixellique des algorithmes de flot optique, les images de Sieleters utilisées présentent des disparités maximales de 1 pixel. Ces valeurs trop faibles entraînent probablement de la confusion dans l'algorithme qui donne un résultat insatisfaisant, avec des flèches qui pointent dans des directions opposées.

Méthodes Globales

Les méthodes globales permettent de faire des hypothèses globales de régularité. Elles cherchent la solution f de la minimisation d'une énergie du type :

$$E(f) = E_{donnée} + \lambda E_{régularisation}$$

Le terme de fidélité aux données mesure la cohérence radiométrique des appariements. Le terme de régularisation peut varier en fonction du type d'image. Il est possible d'ajouter des termes afin de formuler des hypothèses explicites sur les occultations.



(a) Image de référence (b) Flot optique

FIGURE 2.6 – **Algorithme de flot optique sur un couple d'images Sieleters. -** Les images Sieleters utilisées sont les mêmes que sur la figure 6.1. L'image (a) est l'image de référence et (b) montre le flot optique correpondant, calculé avec l'algorithme [MSS16].

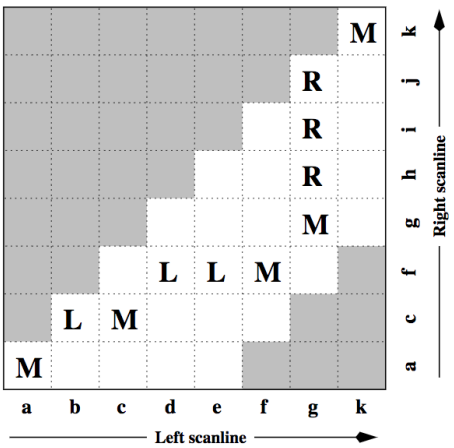


FIGURE 2.7 – **Correspondances en programmation dynamique -** Les abscisses représentent une ligne épipolaire de l'image de gauche, les ordonnées sa correspondante sur l'image de droite. Les lettres en minuscules associées sont les valeurs d'intensité de chaque pixel de la ligne. L'intervalle de disparités autorisées est signalé par les cases en blanc. L'appariement de deux pixels est signifié par un **M**. **L** et **R** indiquent que le pixel est occulté dans l'image de gauche et de droite respectivement. Ces lettres constituent donc un chemin. *Source : [SS02]*

- **Programmation dynamique**

  L'algorithme [OK85] voit le problème d'appariement globalement sur une ligne épipolaire. Il recherche un chemin optimal sur un plan tel que pour une ligne épipolaire, les abscisses décrivent les pixels de l'image de gauche et les ordonnées correspondent à l'image de droite. Un terme d'homogénéité verticale sur les bords est ajouté afin de donner une cohérence entre lignes épipolaires successives. Les occultations peuvent être gérées explicitement dans la construction du chemin, voir figure 2.7. Pour réduire le coût du traitement inter-lignes [BT99] propose de réaliser une régularisation verticale dans un traitement indépendant, selon une heuristique autorisant des changements de disparité suivant la variation radiométrique entre pixels voisins et un indice de confiance.

- ***Semi-global matching***

  Le *semi-global matching* associe des mesures radiométriques locales à des contraintes de régularité globales. Il a été développé par [Hir05]. Afin de ne pas diminuer la netteté du résultat, le coût d'appariemment n'est pas calculé sur un voisinage mais sur la seule valeur du pixel. A cela est associé une régularisation globale utilisant des contraintes unidimensionnelles, reprenant ainsi des idées de programmation dynamique. Toutefois, pour obtenir une régularisation plus robuste, celle-ci n'est pas seulement verticale, mais dans une multitude de directions différentes. Les occlusions peuvent être traitées de manière classique par comparaison entre les deux cartes épipolaires résultantes. Ces méthodes fournissent de très bons résultats. [Hir07] propose également un raffinement sous-pixellique.

- **Coupure de graphe**

L'utilisation des algorithmes de coupure de graphe (*graph cut*) pour la stéréovision est récente mais fournit de très bons résultats, de telle sorte que les meilleurs scores du *benchmark* Middleburry sont souvent obtenus avec ces algorithmes [Tan+16].

Pour résoudre le problème, [BVZ98] propose une approche multi-étiquette permettant de traiter l'ensemble de l'intervalle des disparités sur le même graphe. [Ish03] montre que cette approche donne une solution exacte à condition que les étiquettes soient ordonnées linéairement et que le terme de régularité soit convexe. Cette dernière condition tend à trop pénaliser les fortes discontinuités et donc à lisser le résultat.

[BVZ01; BV06] proposent d'utiliser un terme de régularisation qui ne pénalise que les pixels adjacents ayant des valeurs de disparités différentes. Toutes les discontinuités ont le même coût, donnant un résultat continu par morceaux. Ce problème est résolu de manière binaire avec les *expansion-moves* ou *swap-moves* et ne fournit pas une solution exacte mais suffisamment proche du minimum global. Afin de tenir compte des occlusions, [KZ01] propose d'ajouter un terme qui pénalise le nombre d'occlusions avec un coût constant. Cet algorithme est abordé en détails dans le chapitre 8.

2.2.2 Stéréoscopie multi-vues.

De nombreux algorithmes de stéréoscopie à vues multiples ont été conçus en parallèle du développement d'instruments d'acquisition capables de générer de grandes quantités d'images, comme les caméras plénoptiques [Wet+12]. Ces algorithmes allient les avantages traditionnels d'un grand $\frac{B}{H}$ avec une bonne précision d'estimation et ceux d'un faible $\frac{B}{H}$ avec peu de différences dans la géométrie des images successives.

Méthodes dérivées de la stéréoscopie binoculaire

Une façon intuitive d'exploiter un grand nombre de vues à partir d'un algorithme à deux vues est de les utiliser dans les scores de similarité. [OK93] utilise un algorithme de *block-matching* avec une approche multi-vues, qui lui permet de diminuer l'intervalle de disparités possibles. Il étend le score du SSD de l'équation 2.9 en le sommant pour toutes les paires d'images de la séquence et en l'exprimant en fonction de l'inverse de la profondeur du point de la scène considéré. [Cox94] reprend la même idée dans le cadre de la programmation dynamique en calculant la vraisemblance du terme de fidélité aux données sur l'ensemble des images. [PKF07] propose une méthode variationnelle où le terme d'attache aux données à minimiser cumule les erreurs de reprojection entre les différentes images.

Dans les méthodes dérivées des *graph cuts*, [KZ02a] propose d'ajouter un terme de visibilité qui interdit certaines valeurs de disparités entre pixels voisins afin d'assurer qu'un point non occulté ne soit pas masqué. [Vog+07] adopte une approche par voxels, où l'expression de l'énergie se compose d'un terme surfacique (radiométrique) et d'un terme volumétrique (maximisation d'un volume, il s'agit d'une constante qui tend à "gonfler" le résultat). Le terme de consistance radiométrique gère les occlusions, où chaque image incrémente une valeur (un vote). Le vote est un calcul des scores de corrélation sur les projections du point courant de l'image de référence sur les autres.

Méthodes dédiées aux vues multiples

- **Ajustement de faisceaux**
  Lorsque l'on dispose de nombreuses vues redondantes, il est possible d'estimer les coordonnées d'un ensemble de points de la scène $\mathbf{X}_j$ (ainsi que les matrices de projection des caméras P^i) par ajustement de faisceaux (*bundle adjustment*). On minimise alors pour l'ensemble des observations l'erreur entre la position estimée du point $\tilde{\mathbf{X}}_j$ sur l'image et sa position mesurée $\mathbf{x}_j^i$:

  $$\min_{\tilde{\mathbf{X}}_j, \tilde{P}^i} ||\tilde{P}^i \tilde{\mathbf{X}}_\mathbf{j} - \mathbf{x}_j^i||_2^2$$

  Une synthèse détaillée sur cet ajustement est proposée par [Tri+00]. Cette méthode a l'avantage d'être robuste au bruit mais nécessite une bonne initialisation. C'est pourquoi il est suggéré de l'utiliser comme la dernière étape en complément d'un autre algorithme de reconstruction 3D [DP14]. Le *bundle adjustment* est néanmoins très répandu et utilisé dans des cadres industriels avec par exemple la suite logicielle APERO MICMAC de l'Institut Géographique National [DC11] ou Bundler [SSS06] développé par l'Université de Washington en partenariat avec Microsoft.

- **Shape Carving**

  Le *shape carving*, proposé par [KS00] découpe un volume constitué de voxels en éliminant ceux dont la radiométrie n'est pas constante sur les vues multiples de la scène. Cette méthode est donc sensible aux variations d'illumination de la scène comme les ombres ou des surfaces non-lambertiennes. De plus, elle nécessite des images avec points de vues très différents, ce qui ne favorise pas son utilisation dans le cas d'acquisitions à faibles $\frac{B}{H}$.

- **Images des plans epipolaires**

  C'est la méthode qui est le plus intéressante pour des instruments réalisant des prise de vues en mode à trame défilante, comme Sieleters. D'une part, car elle s'adapte parfaitement à la géométrie d'acquisition de ces images. D'autre part, elle donne accès à des précisions sous-pixelliques au prix de conditions d'acquisitions semblables à celles de Sieleters pour l'acquisition spectrale.

  Les images des plans épipolaires (EPI) se basent sur le fait que lorsque la caméra a une trajectoire rectiligne, les centres d'images de l'ensemble des clichés sont colinéaires. Dans ce cas, les plans épipolaires sont confondus entre toutes les paires d'images. On exploite alors le fait que toutes les images d'une telle séquence ont une géométrie épipolaire commune. Si l'acquisition est suffisamment dense, chaque plan intersecte de nombreux clichés, et un grand nombre de lignes épipolaires correspondantes sont ainsi obtenues.

  On peut construire une nouvelle image à partir de ces lignes : l'image des plans épipolaires (EPI), illustrée en figure 2.8. Il s'agit d'une image spatiotemporelle : l'un des axes (s) décrit le temps, ou plus exactement les numéros de photos prises successivement. Le second axe (u) correspond à la direction du déplacement de l'instrument. L'emplacement d'un point de la scène le long de la ligne épipolaire varie suivant la position de prise de vue. Ce que l'on observe sur l'EPI sont donc les trajectoires des points de la scène en fonction de la

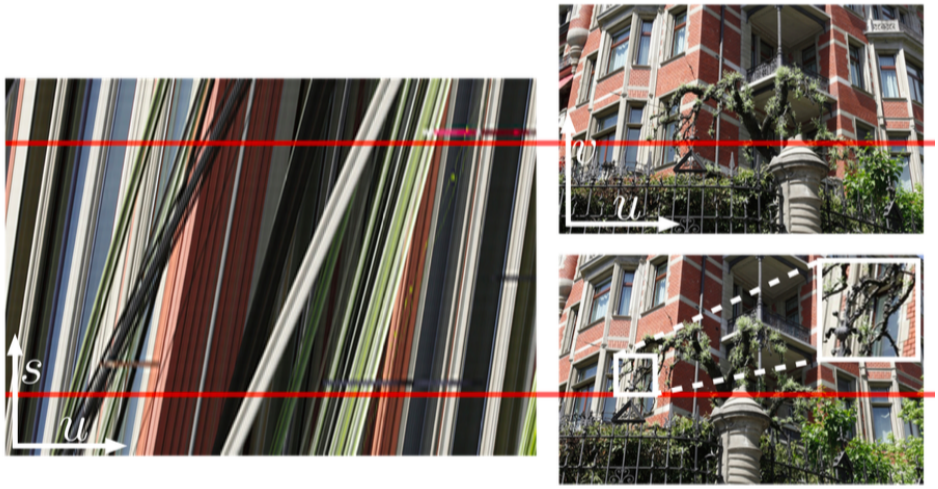


FIGURE 2.8 – **Construction d'une image des plans épipolaires. -** A droite, deux images issues d'une séquence acquise par un déplacement rectiligne vers la droite. Les lignes épipolaires correspondantes sont surlignées en rouge, et la ligne est prolongée pour montrer leur position sur l'image des plans épipolaires à gauche. s est l'axe des numéros d'images, u est l'axe spatial de la direction du mouvement et v est l'axe spatial transversal au mouvement. Source : [KZP13]

position de la caméra selon l'axe de déplacement. La pente de ces trajectoires renseigne sur la position de l'objet dans la scène, comme nous le verrons après.

L'EPI peut aussi être interprétée comme une coupe du cube spatio-temporel formé par une séquence d'images, tel qu'on le voit sur la figure 2.9.

Le cas le plus commun est celui d'un déplacement linéaire de la caméra en visée orthogonale à la trajectoire où l'orientation de la caméra n'est pas altérée au cours de la prise de vue. Dans ce cas, et d'après les notations de la figure 2.10, la disparité d s'exprime de la façon suivante :

$$d = d_2 - d_1 = \frac{f(B+b)}{H} - \frac{fb}{H} \tag{2.10}$$

$$d = B\frac{f}{H} \tag{2.11}$$

L'équation 2.11 montre bien que la disparité varie linéairement par rapport à la base. Si l'orientation de la caméra entre les clichés diffère (la caméra ne vise pas le nadir, perpendiculairement à la trajectoire), l'EPI présente des courbes et non des droites.

Les images des plans épipolaires (EPI) ont d'abord été présentées par [BBM87]. Les EPI peuvent s'assimiler à des coupes de lumigraphe[2] à 3 dimensions, en se limitant à une observation selon une seule dimension. Ce rapprochement justifie qu'au delà d'une reconstruction du relief, certains articles comme [Cri05] proposent aussi des analyses des réflexions spéculaires de la scène à partir des EPI. Pour ce qui est de la construction des cartes de profondeur, les approches sont diverses. [Cri05] segmente l'ensemble du cube d'images en objets cohérents de même disparité, en prenant appui sur l'image de l'espace des disparités (DSI). [BD06] segmente également le cube spatio-temporel des EPI en tubes de même

[2]Terme employé pour la première fois dans [Gor+96]. On emploie aussi le terme *light fields*, ou champs de lumières en français.

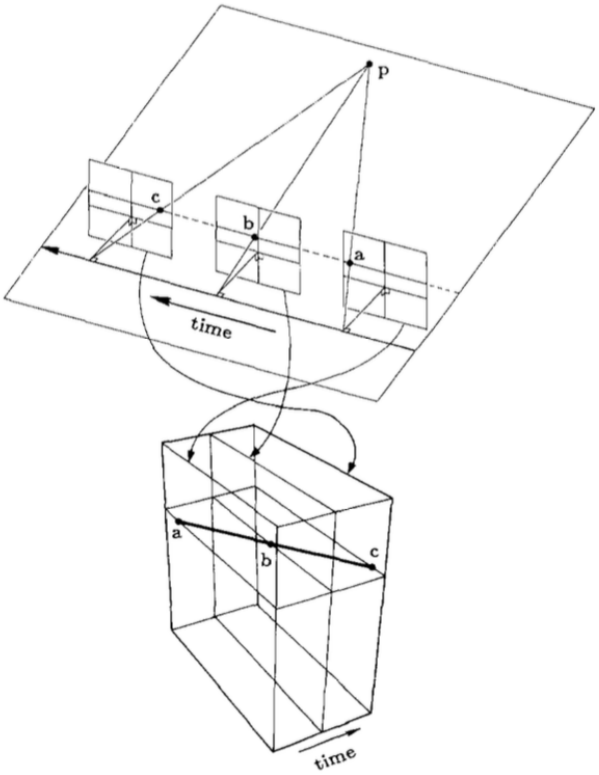


FIGURE 2.9 – **Construction du cube spatio-temporel. -** Les trois clichés, obtenus par un déplacement rectiligne de la caméra vers la gauche, observent un même point p. A partir de ces images on peut construire un cube, dont l'EPI est une coupe transversale qui décrit par une ligne le déplacement du point selon les différents points de vue. Crédits : [BBM87]

pente à l'aide de contours actifs. [KZP13] propose de minimiser un score basé sur l'intensité de l'EPI pour différentes valeurs de disparité, sur les zones de l'image dont les estimations sont susceptibles d'être les plus fiables. Le reste de la carte est complété par propagation à partir de ces estimations en commençant à une échelle fine puis en dégradant la résolution. [WG12; DG13] commencent par estimer la disparité pour chaque pixel à partir du gradient, puis intègrent ces valeurs dans un schéma de minimisation globale en imposant des contraintes sur les occultations. Enfin, [STM06] minimise par *graph cut* une énergie qui prend en compte des occultations et des surfaces non lambertiennes. A partir d'une séquence d'images d'un match de football, [Jor+15] obtient des EPI avec des chemins discontinus. La solution proposée est de commencer par détecter les pixels correspondant à un même objet puis d'ajuster des lignes dessus. Tous ces algorithmes ont été testés sur des images synthétiques ou terrestres, dans des conditions idéales d'acquisition, où la distance entre la caméra et la scène est de l'ordre de quelques mètres et la trajectoire est parfaitement rectiligne. La méthode des plans épipolaires n'a à notre connaissance pas encore été testée sur des images aériennes ou satellitaires.

La restitution par images des plans épipolaires possède plusieurs avantages par rapport aux autres méthodes de stéréoscopie. Les EPI renseignent de manière explicite sur les occulta-

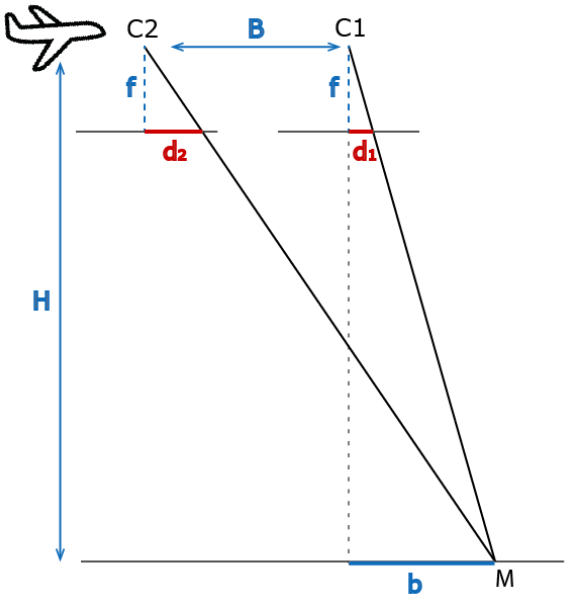


FIGURE 2.10 – **Schéma de deux images en géométrie épipolaire. -** Le point M de la scène (en haut) est observé par les images (en bas) de centres C1 et C2, de focale f et base B. D'après : [BBM87]

tions. En effet, les objets les plus proches de l'image ont des pentes plus faibles, se déplacent donc "plus vite" et leur trajectoire est alors susceptible de masquer celle des objets en arrière-plan. Un croisement sur une image de plan épipolaire correspond ainsi à une occultation.

De plus, la méthode combine les avantages d'une grande base pour la précision de la mesure, et d'une petite base, pour la recherche de correspondants. En effet, si les sommets des clichés ne sont pas trop distants, au vu de la taille des objets, alors les points correspondants suivent une ligne continue.

Aussi, contrairement aux autres méthodes discutées, elle permet des mesures sous-pixelliques sans avoir besoin de passer par une étape d'interpolation des images. La qualité de la mesure de disparité dépendra néanmoins de la méthode de calcul des pentes des chemins des EPI.

Enfin, la détermination de la disparité s'appuie sur autant d'échantillons qu'il y a d'images, ce qui la rend robuste aux erreurs locales de calcul de gradient ainsi qu'aux variations radiométriques des surfaces non lambertiennes.

La méthode des images des plans épipolaires impose cependant de vérifier quelques hypothèses importantes. La plus importante est que la trajectoire du porteur doit être rectiligne. Si cette condition n'est pas respectée, les plans épipolaires entre les paires consécutives d'images ne sont pas confondus, et les images des plans épipolaires ne peuvent être construites. Par ailleurs, si l'orientation de la caméra est modifiée entre les clichés, les trajectoires sur l'EPI sont des courbes et non des lignes. Ce cas est illustré dans [BBM87]. Enfin, les images doivents être acquises avec une fréquence temporelle suffisamment grande, afin d'obtenir des lignes continues sur l'EPI.

D'autres cas apparaissent si l'orientation de l'instrument est modifiée, ou si le déplacement de la caméra est différent [FEK03; MTE02]. Il n'ont été que peu exploités car ils corres-

pondent à des cas rares d'acquisition, pour lesquels d'autres méthodes comme l'ajustement par faisceaux, sont plus efficaces.

2.3 Conclusion

Le choix fait pour le traitement stéréoscopique des images de Sieleters est d'utiliser les EPI. En effet, les séquences d'images acquises s'adaptent particulièrement bien aux hypothèses de trajectoire et fréquence d'acquisition requises. De plus, c'est la méthode la mieux adaptée au calcul de disparités sous-pixelliques, qu'elle intègre par définition. Or, les séquences de Sieleters ont des disparités de 3 pixels maximum sur 424 images, ce qui correspond à du centième ou millième de pixel pour des images successives.

Il faudra néanmoins tenir compte du fait que les images Sieleters contiennent des franges d'interférences fixes, ce qui ne permet pas de les utiliser directement. Le traitement des franges est l'objet de la partie II.

D'autre part, dans le cas d'une acquisition aéroportée, les conditions sur la trajectoire et l'orientation de la caméra ne sont pas parfaitement vérifiées. L'étude de ces erreurs est discutée au chapitre 7.

CHAPITRE 3

PRÉSENTATION DES IMAGES UTILISÉES

Sommaire

3.1 Images simulées **35**

3.2 Séquences Sieleters **36**

3.2.1 Généralités sur les images Sieleters 36

3.2.2 Séquence Le Fauga-Mauzac (CFM) 36

3.2.3 Séquence Canjuers 40

3.3 Evaluation quantitative : le PSNR **40**

Ce chapitre présente les images utilisées dans les chapitres suivants. Elles sont de deux types : soit des images simulées, soit des images mesurées par Sieleters. Parmi les séquences d'images de Sieleters disponibles, deux sont présentées ici. La motivation de ce choix est la composition de la scène ou la disponibilité d'une vérité terrain afin de contrôler les résultats de la stéréoscopie.

3.1 Images simulées

Les images simulées sont construites à partir d'images à très haute résolution. On utilise deux images de la même scène : l'une fournit des valeurs plausibles pour la température, et l'autre des valeurs plausibles pour l'émissivité (supposée constante spectralement). A partir de ces valeurs et d'un modèle atmosphérique, les spectres de luminance pour chaque pixel sont calculés en appliquant la loi de Planck. Il est alors possible d'obtenir trois images : une image des franges, une image panchromatique et une image interférométrique dont les caractéristiques sont proches des images mesurées expérimentalement par Sieleters. Un exemple pour chacune d'elles est donné en figure 3.2. L'image des franges moyennes est obtenue en réalisant la transformée de Fourier du spectre moyen de tous les pixels de l'image. L'image panchromatique contient la luminance moyenne de chaque pixel. Enfin, l'image interférométrique (aussi appelée image physique) se calcule selon l'équation (1.3) en utilisant l'image panchromatique et les caractéristiques de Sieleters : carte de différence de marche, contraste, phase et efficacité spectrale de l'insrument. La construction des différentes images est résumé dans le schéma 3.1.

Plusieurs images montrant des scènes différentes (rurales et urbaines) sont ainsi générées. Un exemplaire de chacune des images, dans leur version interférométrique, est montré en figure 3.3.

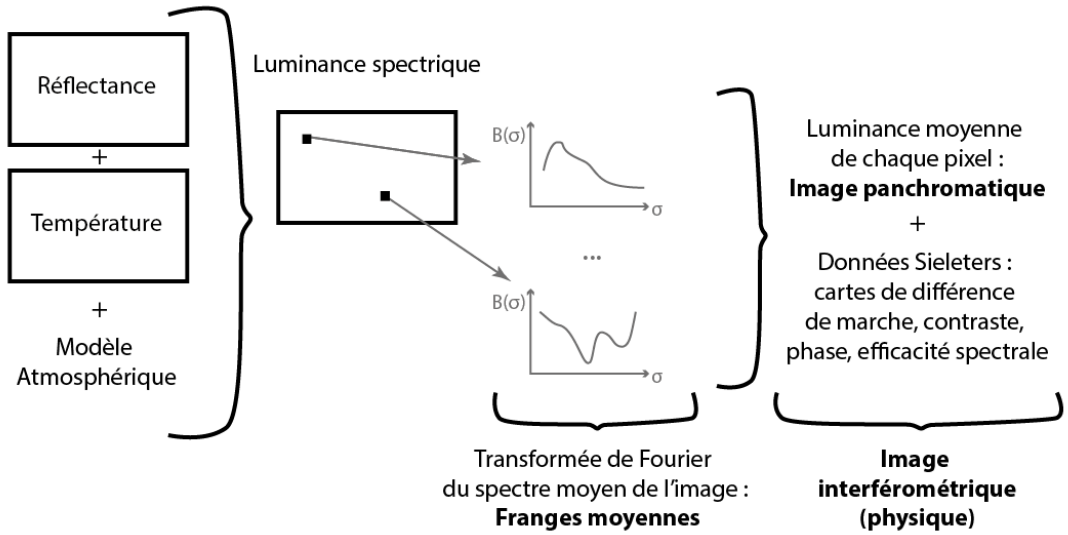


FIGURE 3.1 – **Schéma de calcul des images simulées de franges, panchromatique et interférométrique.**

L'objectif derrière la construction de ces images, est d'une part de pouvoir construire des images selon divers modèles en combinant l'image panchromatique et l'image des franges moyennes. D'autre part, il est de disposer d'un modèle proche de la réalité dont on connaît l'image originale sans franges afin de quantifier les résultats et de choisir les paramètres dans la partie II. Toutefois, il faut noter que les images interférométriques ainsi construites n'en demeurent pas moins qu'une simulation de mesure.

3.2 Séquences Sieleters

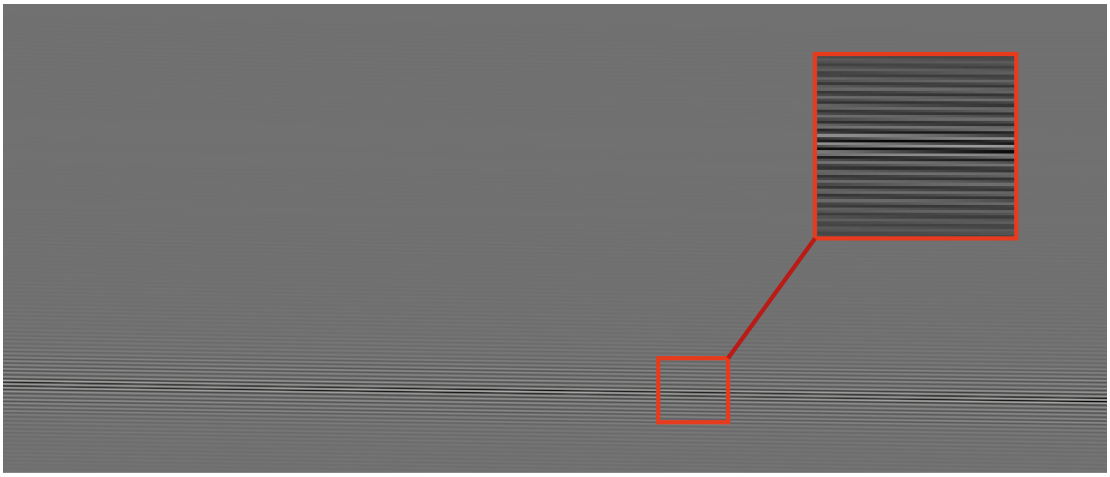
Trois campagnes de vol ont été réalisées avec Sieleters, chacune comprenant plusieurs lignes de vol. Parmi elles, deux séquences sont utilisées comme illustration dans ce manuscrit.

3.2.1 Généralités sur les images Sieleters

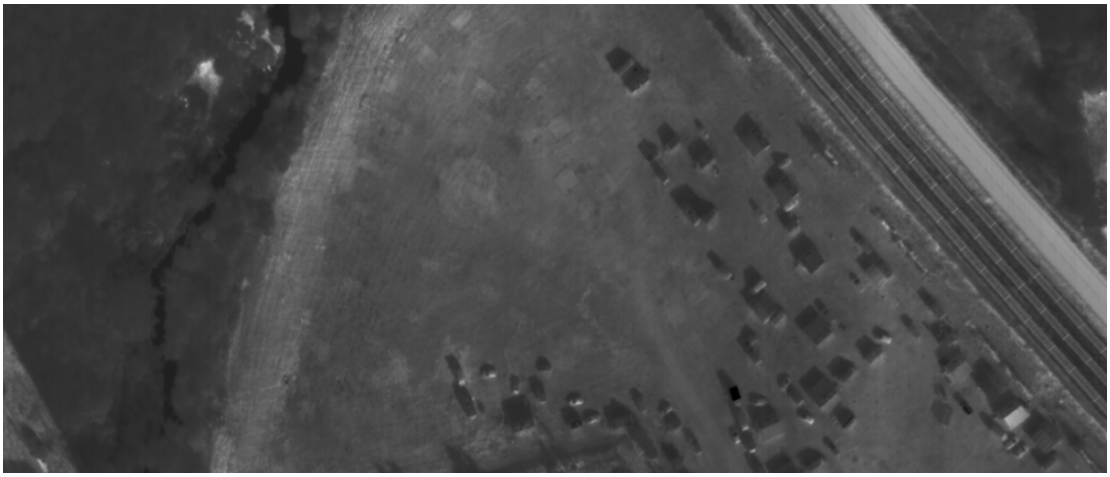
Le tableau 3.1 rappelle les propriétés des images de Sieleters énoncées au chapitre 1. En particulier, le rapport signal-à-bruit pour les images Sieleters est d'environ 500 sur l'image interférométrique, ce qui explique que le bruit sur l'image sera négligé par la suite.

3.2.2 Séquence Le Fauga-Mauzac (CFM)

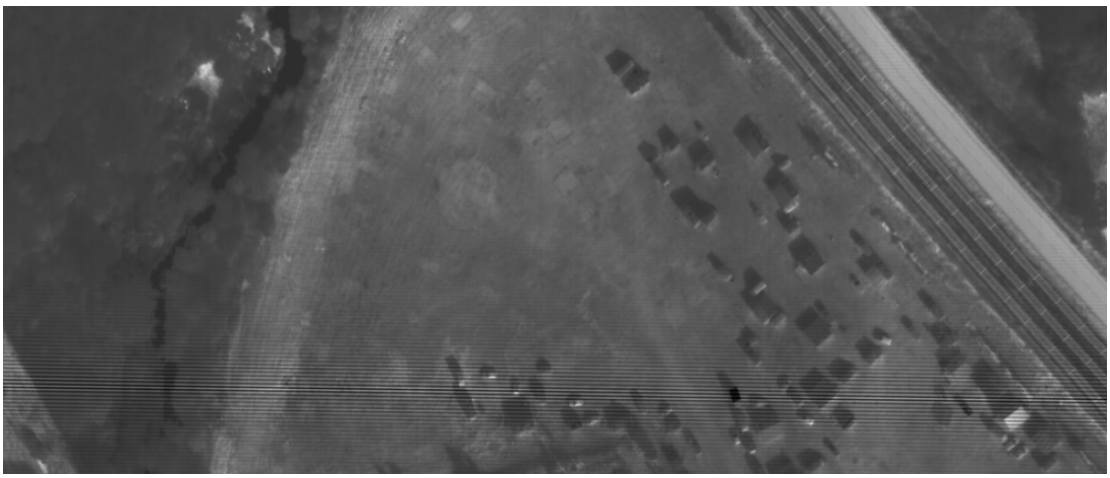
La séquence entière comprend 6330 images sur une ligne de vol d'environ $3\mathrm{km}$. Elle date de septembre 2013 et survole en particulier le centre ONERA des communes de Le Fauga et Mauzac, qui est la scène la plus intéressante de la séquence, en raison de son contenu et des données de terrain disponibles, voir figure 3.4.



(a) Franges moyennes

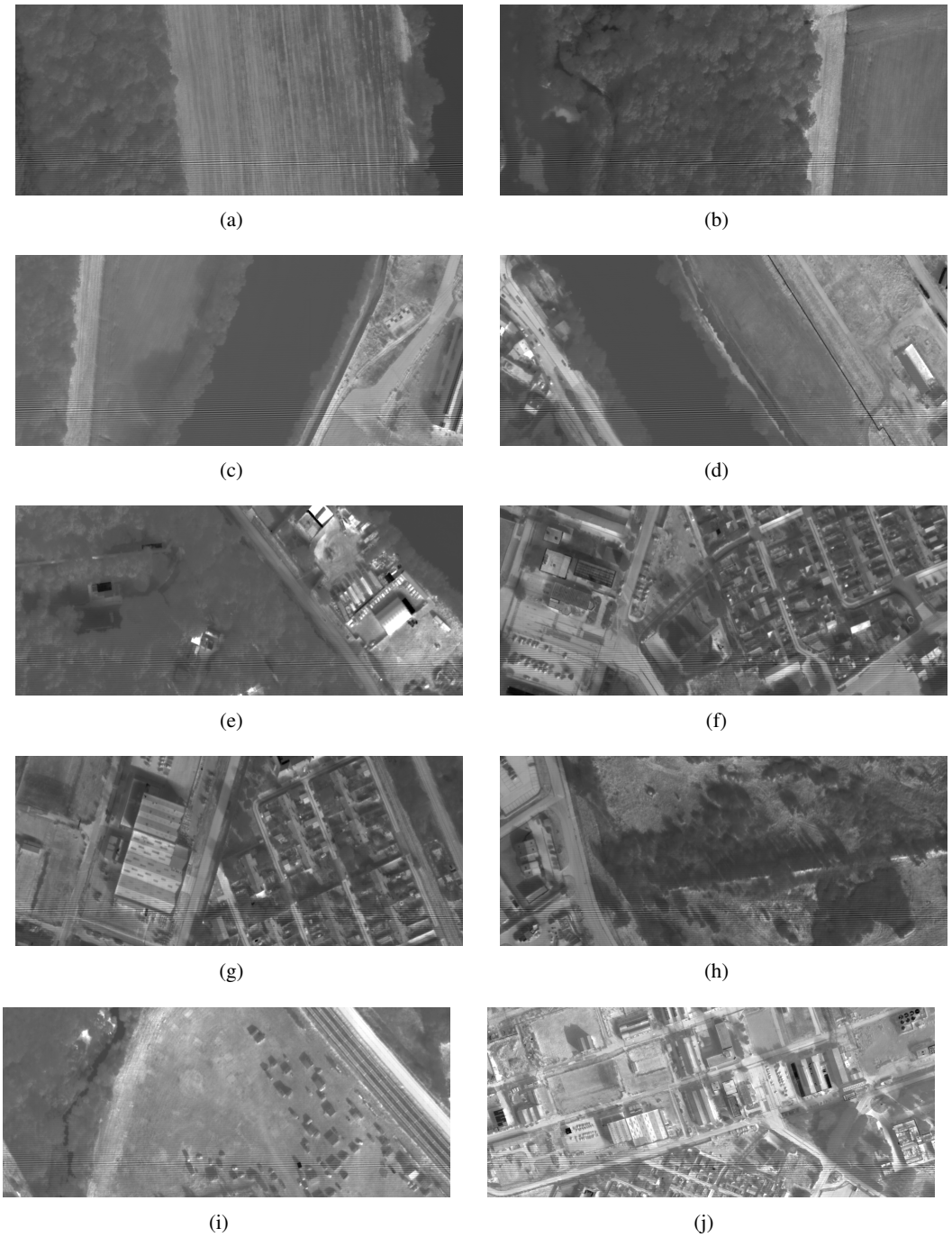


(b) Image Panchromatique



(c) Image interférométrique

FIGURE 3.2 – **Images simulées pour une même scène.**



(a) (b)

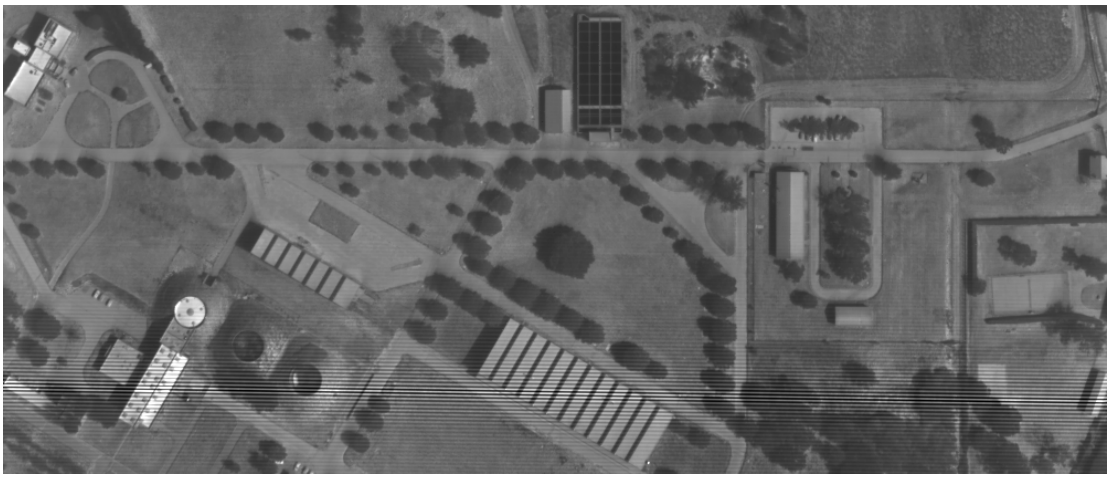
(c) (d)

(e) (f)

(g) (h)

(i) (j)

FIGURE 3.3 – **Images simulées interférométriques. -** L'image (j) a été suréchantillonnée pendant les traitements, ce qui explique qu'elle soit plus nette que les autres.



(a) Image Sieleters BII

(b) Image Géoportail

FIGURE 3.4 – **Images du centre ONERA de Le Fauga-Mauzac. -** Les points rouges sont les points de contrôle utilisés au chapitre 8. *Source : Géoportail*

Largeur BII	$3 - 5.3\mu m$
Largeur BIII	$8 - 11.5\mu m$
Résolution spectrale BII	$13\mathrm{cm}^{-1}$
Résolution spectrale BIII	$6\mathrm{cm}^{-1}$
Résolution spatiale	$50\mathrm{cm}$
SNR	500
Base	$50\mathrm{cm}$
Fauchée	$500\mathrm{m}$
Hauteur de vol	$2000\mathrm{m}$

TABLE 3.1 – **Tableau des propriétés des images Sieleters. -** SNR désigne le rapport signal-à-bruit

Données de vérité terrain. L'intérêt de la zone s'explique par la diversité du sursol. Au-delà de la présence de végétation haute et basse et des bâtiments, on trouve un chateau d'eau ainsi que des garages avec des toitures aux formes particulières, voir figure 3.5. Un autre élément du paysage attrayant est la présence de dunes artificielles.

Des levés de scanner laser 3D existent pour une partie de la zone présentée. Elles se sont avérées être toutefois difficile à exploiter en raison d'erreurs de recalage entre les nuages de points. Elles sont par conséquent utilisées ici seulement pour quelques points de contrôle.

3.2.3 Séquence Canjuers

Cette séquence est principalement composée de relief naturel, pour lequel on dispose d'un modèle numérique de terrain. Elle a été acquise en août 2015. La figure 3.6 montre l'image géoportail de la scène.

3.3 Evaluation quantitative : le PSNR

Le PSNR (Peak Signal to Noise Ratio) est le critère le plus couramment utilisé pour évaluer la qualité d'une image. On le mesure en décibels (dB) pour l'image calculée u^* et sa référence $\tilde{u}$ de la manière suivante :

$$PSNR(u^*) = 10 \log \frac{max_x |u^*(x)|^2}{\|u^* - \tilde{u}\|_2^2} \tag{3.1}$$

Un valeur élevée est donc indicative d'un meilleur résultat. La multiplication par la valeur maximum du résultat sert à faciliter la comparaison d'image différentes, en particulier des images avec des échelles radiométriques différentes.

(a) Cheminées

(b) Recette

(c) Dune

FIGURE 3.5 – **Photographies du centre ONERA de Le Fauga-Mauzac. -** Les photographies ont été prises en septembre 2017.

FIGURE 3.6 – **Image de la scène Canjuers. -** *Source : Géoportail*

Deuxième partie

Extraction des franges d'interférence pour récupérer l'image panchromatique

CHAPITRE 4

MISE EN ÉVIDENCE DES PERTURBATIONS CAUSÉES PAR LES FRANGES SUR LA STÉRÉOSCOPIE

Sommaire

4.1 Stéréoscopie en présence de franges d'interférence **45**
4.1.1 Tests sur différents algorithmes 45
4.2 Etat de l'art de l'extraction de franges d'interférences **47**
4.3 Caractérisation des franges d'interférences de Sieleters **47**
4.4 Conclusion **49**

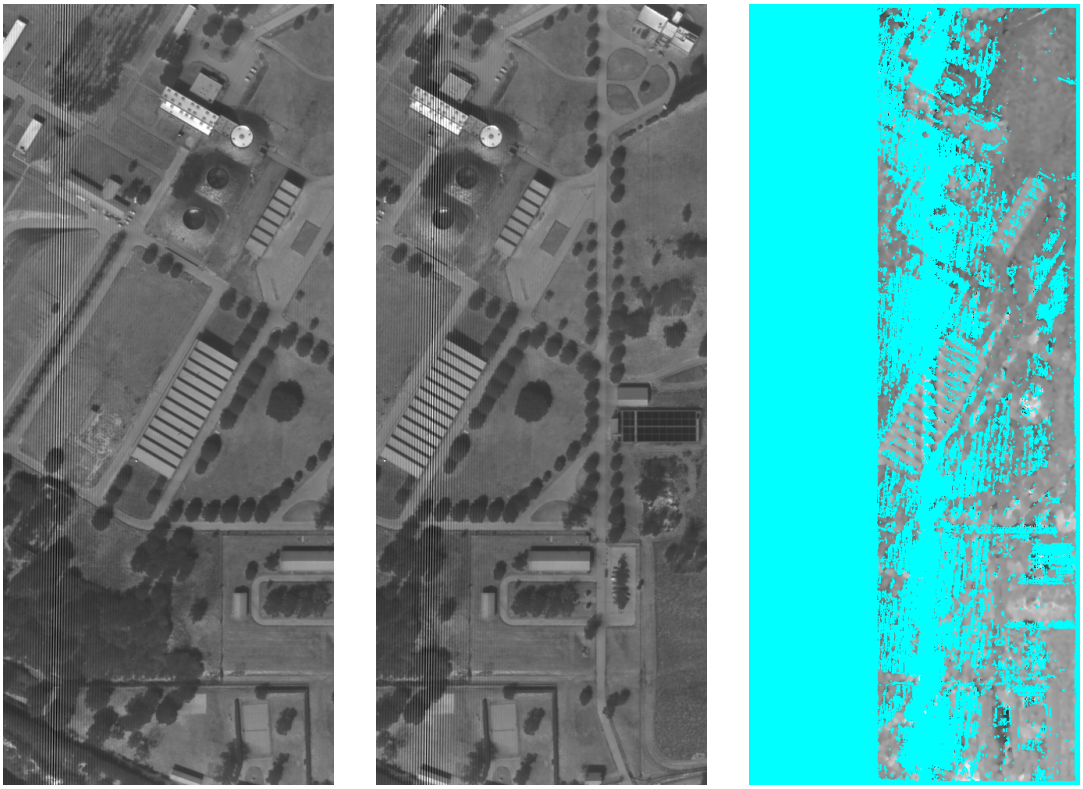
Comme vu au chapite 1, les images ont des franges d'interférences. Celles-ci sont présentes sur toute l'image, où elles sont très contrastées dans la moitié inférieure de l'image autour de la différence de marche nulle et brouillées dans la partie supérieure. Ce chapitre montre les perturbations qu'elles causent sur les algorithmes de stéréoscopie, puis s'intéresse aux solutions proposées dans la littérature pour les extraire de l'image. Enfin, une caractérisation spectrale des franges de Sieleters est donnée, afin d'être utilisée par la suite au service de la séparation des franges d'interférence du reste de l'image.

4.1 Stéréoscopie en présence de franges d'interférence

Les franges d'interférences sont présentes sur toute l'image, où elles sont très contrastées dans la moitié inférieure de l'image autour de la différence de marche nulle et brouillées dans la partie supérieure. Elles n'ont aucune visée stéréoscopique et sont dédiées à la construction de l'information spectrale. Plus que ça, elles sont fixes par rapport au détecteur et donc génèrent des bords très contrastés au même endroit sur toutes les images.

4.1.1 Tests sur différents algorithmes

Nous testons les images de Sieleters sur deux algorithmes sous-pixelliques : l'algorithme de block matching de [SAM11] et les images des plans épipolaires présentées au chapitre 2.



(a) Image de gauche (b) Image de droite (c) Résultat

FIGURE 4.1 – **Résultat de l'algorithme de block-matching. -** (c) montre le résultat de l'algorithme de block-matching [SAM11] pour le couple d'image (a) et (b). Les pixels colorés en cyan correspondent à des points pour lesquels l'algorithme n'a pas trouvé d'appariemment.

Algorithme de *Block-Matching*. L'algorithme utilisé ici est un algorithme de block-matching réalisant des appariemments sous-pixelliques [SAM11]. Il réalise un test statistique pour savoir si un appariement est significatif. S'il est validé, la disparité associée est donnée en niveaux de gris. Lorsqu'aucun score d'appariemment n'est assez bon, le pixel est coloré en cyan, pour montrer qu'aucune valeur de disparité n'a pu être attribuée. L'implémentation utilisée est celle de la chaîne MissStereo [Mon11]. Dans [SAM11], les auteurs montrent qu'un score de moins de 70% de pixels appariés signifie en général que les images sont corrompues par du bruit ou que les images représentent des scènes différentes. Le score pour les images de la figure 4.1 est de 56%. La zone commune utilisée par l'algorithme contient des franges d'interférence sur les bâtiments. Le résultat est très mauvais sur cette région, malgré le fait que le bâti fournit en général de meilleurs résultats que les zones végétales faiblement texturées comme la pelouse. Ce mauvais résultat est clairement attribué aux franges d'interférence.

Images des plans épipolaires. Les franges perturbent les orientations des gradients de l'image. On le montre ici sur une image des plans épipolaires (EPI). Un exemple d'EPI est donné figure 4.2. L'exploitation de cette image se fait par extraction de segments basée sur la mesure de l'orientation du gradient [Gio+12]. Cependant, comme les franges engendrent de forts gradients de direction

oblique aux droites décrites par les éléments de la scène, l'algorithme n'a pas assez d'information pour assurer une bonne détection. Cela est particulièrement vrai sur la partie droite de l'image, où se trouvent les franges très contrastées. L'orientation des gradients ne traduit là que la direction des franges, avec des valeurs à $\pm 180°$.

Au vu des résultats présentés, il pourrait être décidé de ne traiter que la partie où les franges sont suffisamment brouillées. Cela ne représente néanmoins qu'une centaine de lignes, ce qui limiterait la redondance à une centaine d'images et dans le cas des bâtiments les plus hauts (2 étages) représenterait une disparité maximale de 0,7 pixel entre les images de début et fin de séquence. Il est donc nécessaire d'établir une stratégie pour s'affranchir des franges afin d'utiliser l'ensemble des lignes de l'image.

4.2 Etat de l'art de l'extraction de franges d'interférences

Un problème similaire à celui proposé ici, est celui de [WZC15] sur les images de LASIS, un spectromètre imageur à transformée de Fourier statique. Les auteurs cherchent à séparer l'arrière-plan des franges parce qu'elles perturbent les algorithmes de compression. Pour cela ils régularisent l'image avec la variation totale combinée à une analyse morphologique de l'arrière-plan par ondelettes, et des franges par la transformée de Fourier. Le modèle d'image utilisé est additif. Néanmoins, une différence importante avec notre problématique est que les images utilisées ont un arrière-plan très lisse de texture quasi-perpendiculaire aux franges, contrairement aux scènes survolées par Sieleters, qui sont un cas plus général contenant des textures irrégulières et variées.

La plupart des études de séparation des franges de l'arrière-plan visent à réaliser un débruitage pour améliorer la détection de franges. C'est le cas de [Zho+12], qui utilise des descripteurs morphologiques afin d'extraire des franges non nécessairement rectilignes par du débruitage d'un arrière-plan très lisse. Le cas inverse est également présent dans la littérature : [RH06] cherche à récupérer le fond uniforme d'une figure d'interférence, pour des problématiques de calibration, en utilisant une description des franges par la transformée en ondelettes.

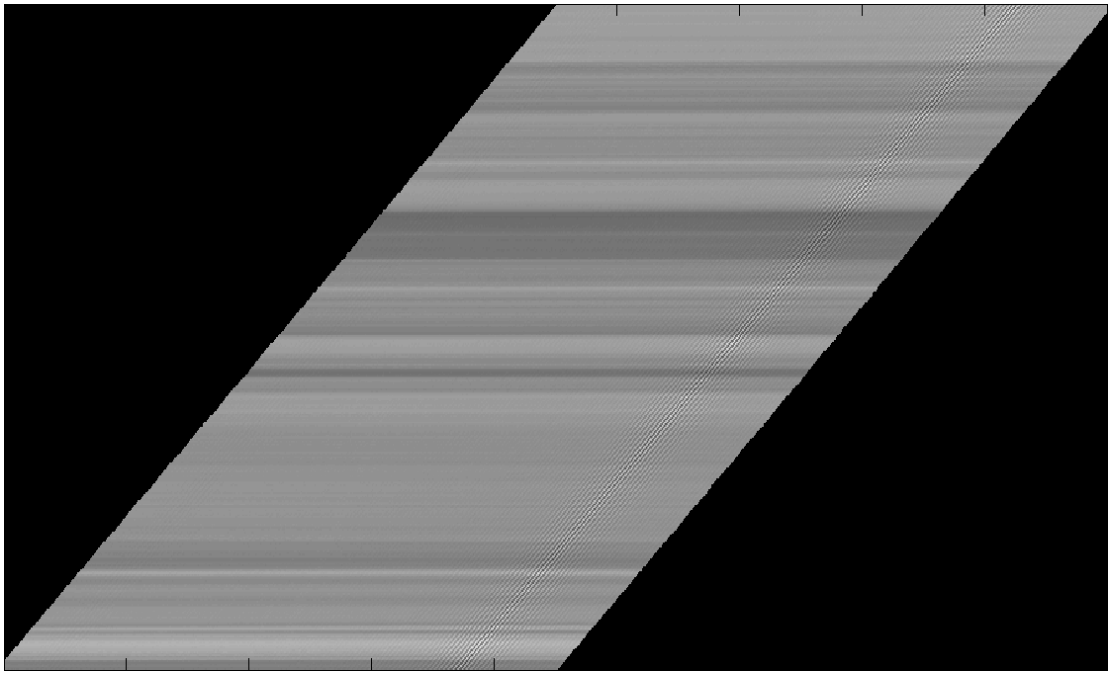
Un contexte plus général de ce type de problème est celui de la séparation de la texture d'une image, tel que présenté dans [MAP11], qui utilise un modèle additif et une norme adaptative pour gérer des textures à orientations et fréquences variables. Il réalise ainsi une décomposition *cartoon-texture*, qui sera l'objet de la section 5.1.3.

4.3 Caractérisation des franges d'interférences de Sieleters

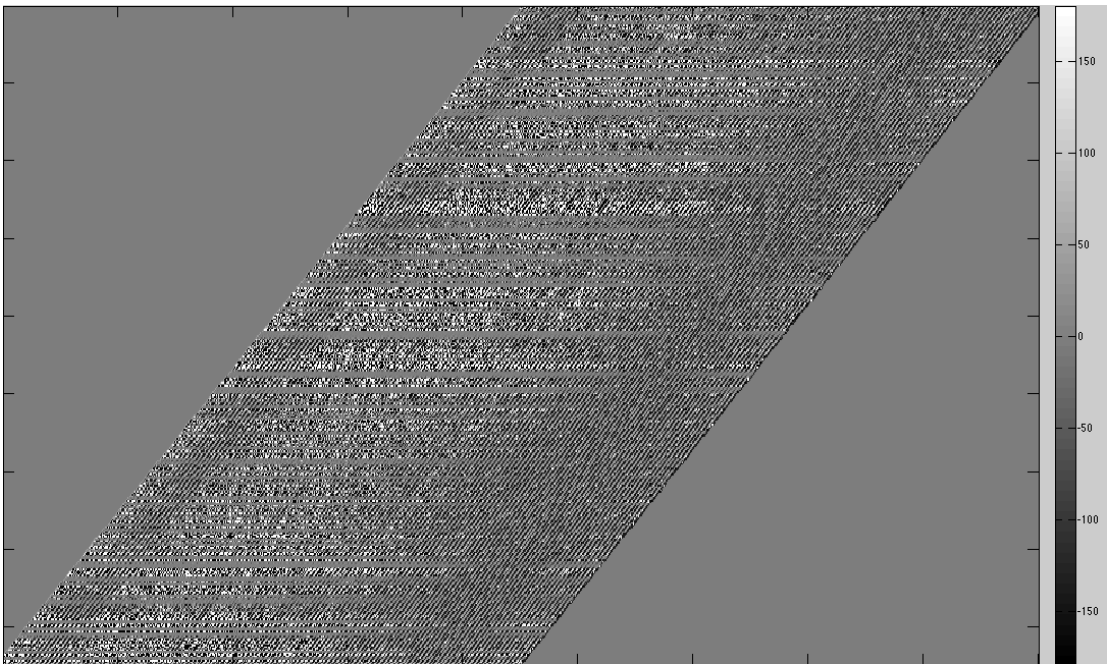
Les franges de Sieleters sont majoritairement localisées sur une zone du spectre en fréquences spatiales, que l'on appellera Ω. Ces fréquences, identifiables sur la figure 4.3, correspondent à la réponse spectrale des franges en observation d'une scène homogène.

Pour déterminer les fréquences limites de la zone Ω, on utilise une carte des différences de marche. Il s'agit d'une image donnant pour chaque pixel la valeur de différence de marche $\delta(x, y)$, obtenue par éclairement par une source homogène et monochromatique. Les franges étant globalement rectilignes, la différence de marche s'écrit :

$$\delta(x, y) = ax + by + \epsilon(x, y) \tag{4.1}$$



(a) Image des plans épipolaires



(b) Orientations des gradients, en degrés

FIGURE 4.2 – **Image des plans épipolaires obtenue à partir d'images Sieleters. -** On observe des franges d'interférence obliques sur l'ensemble de l'EPI, et en particulier sur la droite, où elles sont très contrastées. Cela se répercute sur les orientations des gradients de l'image (en degrés), avec des nombreuses valeurs proches de la verticale ($\pm 180°$).

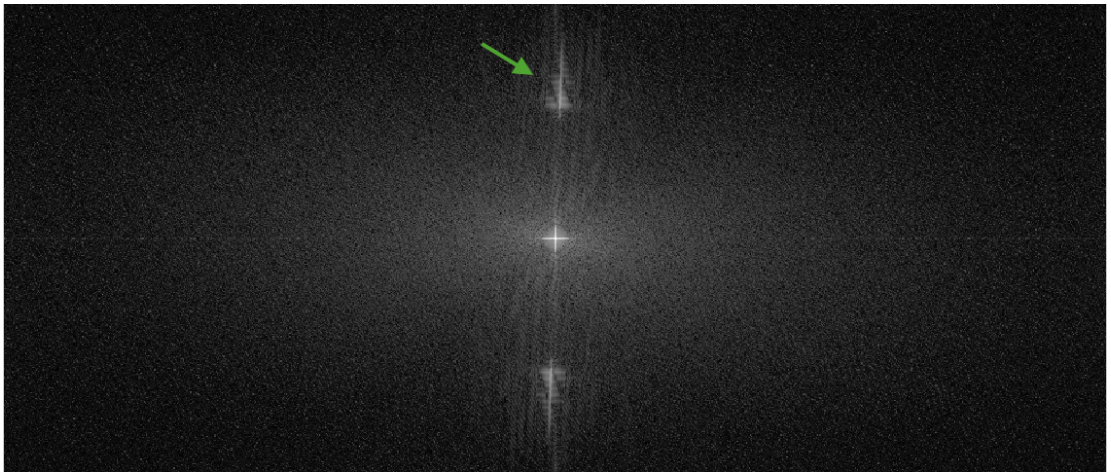


FIGURE 4.3 – **Transformée de Fourier d'une image Sieleters -** Dans le voisinage des basses fréquences verticales, deux lignes quasi verticales correspondent aux fréquences propres des franges. L'une d'elles est désignée par une flèche verte.

$\epsilon(x, y)$ varie lentement et traduit le fait que le plan d'interférence est légèrement courbe (les franges ont en réalité une forme hyperbolique). Il est négligeable. Le terme d'interférences s'exprime alors de la façon suivante, où σ désigne le nombre d'onde :

$$cos(2\pi\sigma\delta(x, y)) = \cos(2\pi\sigma ax + 2\pi\sigma by) \tag{4.2}$$

On déduit que les fréquences d'interférences sont l'ensemble :

$$\{(a\sigma, b\sigma)/\sigma \in [\sigma_{min}; \sigma_{max}]\} \tag{4.3}$$

On trouve alors les bornes de la zone Ω :

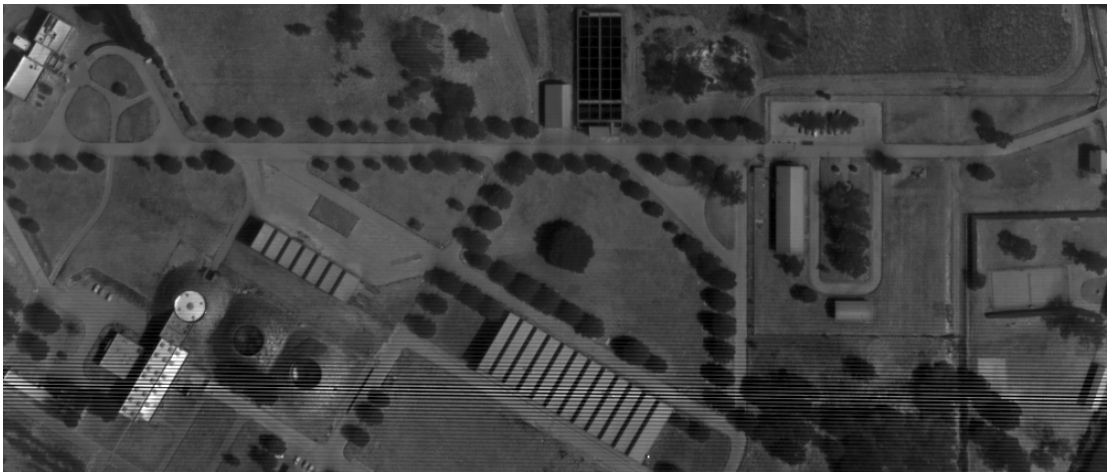
$$\{(N_x a\sigma, N_y b\sigma)/\sigma \in [\sigma_{min}; \sigma_{max}]\} \tag{4.4}$$

où N_x et N_y correspondent au nombre de pixels de l'image dans chacune des directions.

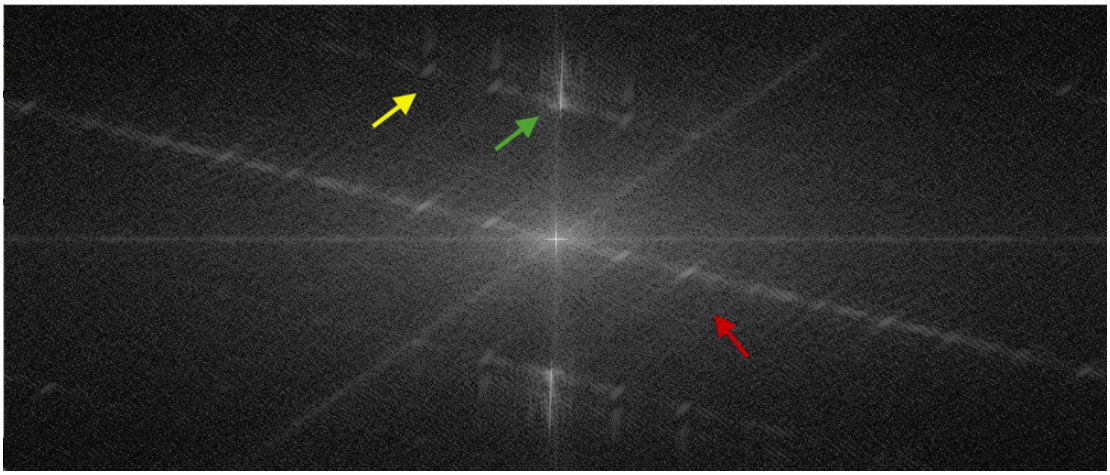
Néanmoins, ces valeurs sont théoriques et plusieurs effets font que des fréquences des franges peuvent être présentes en-dehors de la zone calculée. Les franges n'étant pas tout à fait alignées sur les lignes du détecteur, le signal est de fait non-périodique, ce qui cause l'apparition d'artefacts autour des fréquences des franges, voir figure 4.3. En pratique, la "source" observée n'est pas homogène. Les fréquences de l'image de la scène observée étant modulée par celles des franges, cela élargit la zone Ω. Sur certaines images, cet élargissement peut être très important, comme c'est le cas sur l'exemple de la figure 4.4.

4.4 Conclusion

Les résultats de la stéréoscopie sur des images avec franges d'interférences mettent en évidence la nécessité d'une étape de pré-traitement afin de séparer les franges de l'image de la scène. Ce problème a été traité dans le cas d'arrière-plans très lisses, très distincts du point de vue fréquentiel et morphologique de la figure d'interférence. Des problématiques connexes d'extraction de textures



(a) Images Sieleters



(b) Transformée de Fourier de l'image (a)

FIGURE 4.4 – **Image Sieleters et sa transformée de Fourier. -** La flèche verte montre les fréquences des franges pour une scène homogène. La flèche rouge indique les fréquences propres à la scène, particulièrement au bâtiment de la recette en bas au milieu de l'image. Enfin, en jaune sont désignées les répliques des franges dues à la texture de la scène.

quelconques sont également présentes dans la littérature. Néanmoins, la spécificité du problème présent, avec des scènes naturelles variées et fréquentiellement riches, pose un problème inédit. En outre, la nature particulière des images d'interféromètres imageurs à transformée de Fourier statique fournit un a priori précis sur les franges. Deux solutions expoitant cette connaissance sont proposées dans les chapitres suivants.

CHAPITRE 5

UTILISATION D'UN MODÈLE D'IMAGE ADDITIF

Sommaire

5.1 Modèle de décomposition additif **52**
5.1.1 Dérivation physique du modèle 52
5.1.2 De la nécessité d'une approche variationnelle 52
5.1.3 Etat de l'art de la décomposition *Cartoon-Texture* 52
5.2 Outils d'optimisation convexe **53**
5.2.1 Notions d'optimisation convexe 53
5.2.2 L'opérateur de proximité 54
5.2.3 L'algorithme Douglas-Rachford 56
5.2.4 Variation totale 56
5.3 Expression du problème **58**
5.3.1 La contrainte fréquentielle 58
5.3.2 Le problème de minimisation 58
5.3.3 Algorithme 59
5.4 Choix des paramètres **59**
5.5 Résultats **60**
5.5.1 Résultats sur image simulée 60
5.5.2 Sur images Sieleters 60
5.6 Conclusion **61**

Les franges d'interférences codent l'information spectrale et sont donc indispensables au recueil des données hyperspectrales. En revanche, pour la problématique qui nous intéresse, qui est celle de la stéréoscopie, elles sont une complication. Ce chapitre propose une méthode de séparation des franges et de la scène basée sur un modèle additif.

5.1 Modèle de décomposition additif

5.1.1 Dérivation physique du modèle

En partant du modèle physique de formation d'image de l'équation (1.4), cette section regarde la signification physique d'un modèle additif. En négligeant les variations de la phase des franges, on obtient l'équation suivante :

$$I(x,y) = \int_\sigma B(x,y,\sigma) + \int_\sigma B(x,y,\sigma) C \cos(2\pi\sigma\delta(x,y)) d\sigma \tag{5.1}$$

Le premier terme de cette équation correspond à l'image panchromatique, c'est-à-dire l'image de la scène sur l'ensemble de la bande spectrale d'intérêt. Le deuxième terme correspond aux franges d'interférences. En supposant que $B(x,y,\sigma)$ est à variations spatiales lentes, alors son support fréquentiel est compris uniquement dans la zone fréquentielle $\tilde{\Omega}$ définie en section 4.3.

On note w l'image mesurée par l'instrument, u la composante panchromatique de la scène, v la composante des franges. La décomposition additive s'écrit de la manière suivante :

$$w(x,y) = u(x,y) + v(x,y) \tag{5.2}$$

5.1.2 De la nécessité d'une approche variationnelle

A la lumière de la connaissance du domaine fréquentiel propre des franges (voir section 4.3) et du modèle additif d'image, on peut se demander si un simple filtrage des fréquences des franges ne suffirait pas à régler le problème. Néanmoins, comme expliqué précédemment, les fréquences des franges ne sont pas contenues strictement dans Ω, mais dans une zone diffuse autour, que l'on appelle $\tilde{\Omega}$. Il faudrait donc faire un filtre passe-bande sur $\tilde{\Omega}$ et cette zone ne contient évidemment pas que des fréquences des franges. Appliquer brutalement un filtre passe-bande filtrerait également les hautes fréquences de l'image, vitales pour la stéréoscopie. La stratégie choisie emploie donc une méthode variationnelle, formulée plus loin dans la section.

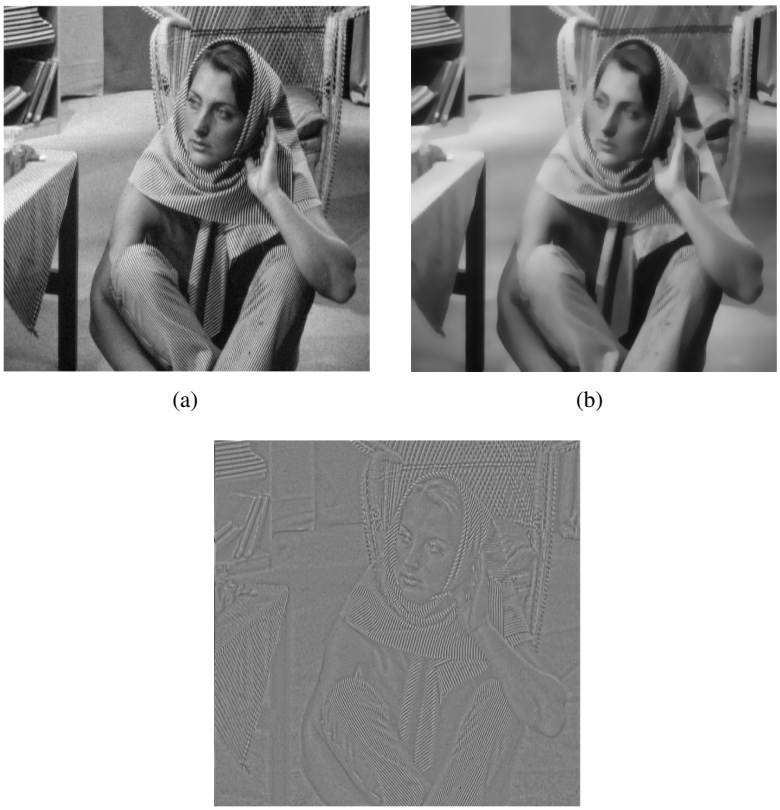
5.1.3 Etat de l'art de la décomposition *Cartoon-Texture*

La méthode de séparation des franges de l'image s'inspire de la décomposition d'image *cartoon-texture*. Celle-ci, présentée dans [OSV03], dissocie l'image w en deux composantes u et v. u désigne une image lisse par morceaux (*cartoon* en anglais). La seconde composante, v, contient la *texture*, la partie oscillatoire de l'image. Le problème de minimisation, dans sa configuration la plus basique, est le suivant :

$$\min_u \int |\nabla u| + \lambda \int \|w-u\|^2 \tag{5.3}$$

La régularisation par la variation totale pénalise les oscillations pour renforcer les structures continues par morceaux. Le terme de fidélité s'assure que l'erreur quadratique à l'image initiale reste suffisamment faible, et que les oscillations aient une moyenne nulle. Un exemple de résultat est donné en figure 5.1. Cette décomposition a été reprise dans [Auj+06] ou [Bua+10], en utilisant des modèles mieux adaptés pour décrire la composante texturale.

De manière similaire, les images Sieleters peuvent être abordées comme des images d'une scène texturée à laquelle s'ajoute la composante oscillatoire des franges d'interférences. Le problème



(a) (b)

(c)

FIGURE 5.1 – **Décomposition cartoon-texture -** (a) Image originale. (b) Composante lisse (*cartoon*). (c) Composante contenant la texture. Crédits : [OSV03]

sera donc envisagé comme un problème variationnel visant à extraire de l'image la texture des franges, qu'il nous est possible de caractériser fréquentiellement.

5.2 Outils d'optimisation convexe

5.2.1 Notions d'optimisation convexe

Cette première section fournit des notions succintes d'optimisation convexe. Elle s'appuie sur [BV04] et [Ber09], qui permettront au lecteur d'approfondir le sujet.

Un problème d'optimisation recherche une variable $x = (x_1, ..., x_n)$ qui minimise une fonction objectif $f : \mathbb{R}^n \to \mathbb{R}$. En plus, plusieurs fonctions de contrainte $c_i : \mathbb{R}^n \to \mathbb{R}, i = 1, ..., m$ peuvent être appliquées. Ce problème s'écrit :

$$\begin{array}{ll} \arg\min_x & f(x) \\ \text{sous contraintes} & c_i \leq b_i \end{array} \tag{5.4}$$

Le problème peut avoir plusieurs minima locaux. Le minimum global du problème, s'il existe, est la solution et se note x^*. Une condition nécessaire pour qu'un point soit un minimum est qu'il soit un point stationnaire :

$$\nabla f(x^*) = 0 \tag{5.5}$$

Cependant, cette condition n'est pas suffisante, et un point stationnaire n'est pas forcément un minimum. Il peut s'agir d'un point d'inflexion.

Si les fonctions f et c_i sont convexes et continues, alors la condition de recherche de minimum (5.5) est suffisante. De plus, dans ce cas, un minimum local est un minimum global. Il est en outre unique pour une fonction strictement convexe.

Ces problèmes, s'ils n'ont en général pas de solution analytique, ont néanmoins à leur service des méthodes de résolution itératives très efficaces. C'est le cas de l'algorithme Douglas-Rachford et des algorithmes basés sur l'opérateur de proximité, qui occupent le reste de cette section.

5.2.2 L'opérateur de proximité

Cette section présente brièvement quelques rappels sur l'opérateur de proximité. Pour des compléments sur ce sujet, on peut se référer à [PB13], [CP10] et [CW05].

L'opérateur de proximité a d'abord été évoqué dans [Mor62]. Sa genèse est en lien étroit avec celle de l'enveloppe de Moreau d'une fonction $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ propre, convexe et semi-continue inférieurement sur $\mathbb{R}$ et d'indice $\gamma \in]0; +\infty[$ qui se définit comme :

$$M_{\gamma f}(y) = \min_x f(x) + \frac{1}{2\gamma}||x - y||_2^2 \tag{5.6}$$

L'enveloppe de Moreau fournit donc le minimum d'une version lisse de la fonction f. Ce problème atteint sa borne inférieure en un unique point x noté $\mathrm{prox}_{\gamma f}(y)$, appelé le point proximal :

$$\mathrm{prox}_{\gamma f}(y) = \arg\min_x f(x) + \frac{1}{2\gamma}||x - y||_2^2 \tag{5.7}$$

Une propritété fondamentale de l'opérateur de proximité est que ses points fixes sont les minimiseurs de la fonction f. De ce fait, dans certains problèmes d'optimisation, il est parfois plus aisé d'utiliser $\mathrm{prox}_{\gamma f}$ plutôt que f, puisqu'il permet soit de minimiser une fonction lisse soit d'obtenir une expression simplifiée du problème. Dans ce deuxième cas l'opérateur de proximité est de surcroît une solution rapide.

Un exemple simple est celui de la fonction indicatrice d'un ensemble :

$$\iota_C(x) = \begin{cases} 0 & \text{si } x \in C \\ \infty & \text{sinon} \end{cases} \tag{5.8}$$

Son opérateur de proximité a une solution analytique : il s'agit de la projection euclidienne P sur l'ensemble. A ce titre, l'opérateur de proximité est une généralisation de la projection.

$$\mathrm{prox}_{\iota_C}(y) = \arg\min_{x \in C} ||x - y||_2^2 = P_C(y) \tag{5.9}$$

Une autre interprétation est de voir le point proximal comme un compromis entre la minimisation de f et la proximité à une valeur y. Cet équilibre est modulé par le paramètre γ. Plus celui-ci sera

grand, plus le résultat s'éloignera de la valeur d'origine. La figure 5.2 illustre cette interprétation. On y observe l'estimation de l'opérateur de proximité en divers points bleus. Les points appartenant au domaine de définition de la fonction évoluent vers un point rouge en direction du minimum de la fonction, mais tous ne l'atteignent pas. Les points en dehors de cet ensemble atteignent tous la frontière en évoluant également vers le minimum.

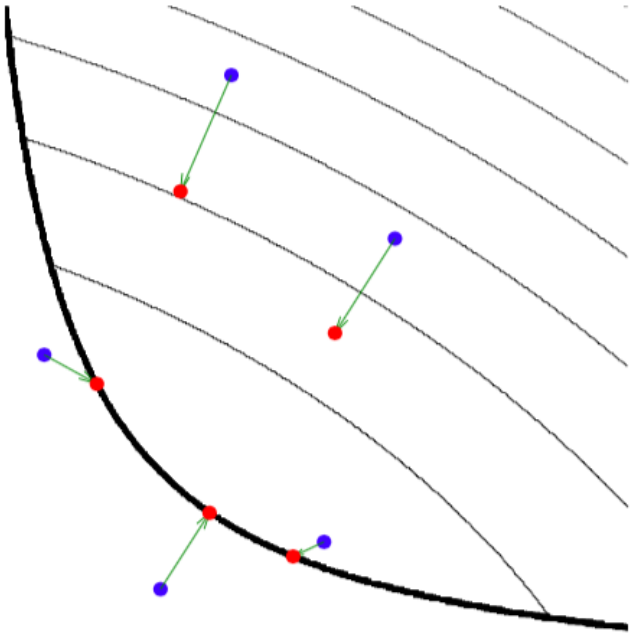


FIGURE 5.2 – **Illustration de l'estimation de l'opérateur de proximité. -** Les lignes noires désignent les lignes de niveau de la fonction, la ligne en gras est la frontière de son domaine. Les estimations sur les points bleus donnent un résultat en rouge. Tous tendent vers le minimum de la fonction. *Source : [PB13]*

Plusieurs algorithmes ont été conçus pour exploiter l'opérateur de proximité. Ils s'apparentent parfois à des algorithmes classiques adaptés à la recherche du point proximal. C'est le cas de la méthode de descente proximale. Elle s'applique à des problèmes du type :

$$\min_x f_1(x) + f_2(x) \tag{5.10}$$

où f_1 et f_2 sont propres, convexes et semi-continues inférieurement et f_1 est dérivable. Alors l'algorithme s'applique en deux étapes :

$$\begin{cases} x_{n+\frac{1}{2}} = x_n - \tau \nabla f_1(x_n) \\ x_{n+1} = \text{prox}_{\gamma f_2}(x_{n+\frac{1}{2}}) \end{cases} \tag{5.11}$$

La première étape est une descente de gradient selon f_1, τ étant le pas de descente. La seconde étape, qui évalue l'opérateur de proximité en f_2, renvoie à une recherche de point fixe. Cet algorithme sera utilisé dans le chapitre 6.

5.2.3 L'algorithme Douglas-Rachford

L'algorithme de Douglas-Rachford (abrégé DR par la suite) est un algorithme d'optimisation qui utilise la séparation de la fonction objectif en deux termes, tel que $f(x) = f_1(x) + f_2(x)$. Il permet d'obtenir un résultat de convergence d'après des hypothèses de départ assez faibles : f_1 et f_2 sont convexes, coercives ($f_1 + f_2 \xrightarrow{||x|| \to +\infty} \infty$), semi-continues inférieurement, et l'intersection de leurs domaines est non nulle. Alors le problème de la minimisation de f admet une unique solution, caractérisée par :

$$\begin{cases} x & = \text{prox}_{\gamma f_2}\, y \\ \text{prox}_{\gamma f_2}\, y & = \text{prox}_{\gamma f_1}(2(\text{prox}_{\gamma f_2}\, y) - y) \end{cases} \tag{5.12}$$

où prox désigne l'opérateur de proximité. L'algorithme DR peut donc être interprété comme la projection successive dans l'un puis l'autre des ensembles de f_1 et f_2. Pour plus de détails, [CP10] fournit des explications approfondies de cet algorithme.

5.2.4 Variation totale

La variation totale (TV) a été employée pour la première fois dans le traitement d'image par [ROF92] dans le cadre du débruitage. Pour une image f définie sur un sous-ensemble $\mathcal{O}$ de $\mathbb{R}^2$ à valeurs dans $\mathbb{R}$, on la définit par :

$$TV(f) = \int_{\mathcal{O}} ||\nabla f||_2 \, dx \tag{5.13}$$

où ∇f désigne le gradient de l'image. Une image discrète prend ses valeurs dans $\mathbb{Z}^2$ telle que $f : \mathcal{O}^d \to \mathbb{R}$ et sa variation totale se calcule par :

$$TV^d(f) = \sum_{(x,y) \in \mathcal{O}^d} ||\nabla^d f||_2 \tag{5.14}$$

Le gradient pour une image discrète $\nabla^d f = (D_h, D_v)$ s'exprime de la manière suivante :

$$\nabla^d f(x,y) = \begin{cases} f(x+1,y) - f(x,y) \\ f(x,y+1) - f(x,y) \end{cases} \tag{5.15}$$

Utilisée à la place d'une norme quadratique comme terme de régularisation, la TV autorise les variations brusques et permet ainsi de lisser les zones homogènes tout en conservant les discontinuités de l'image. C'est pour cela qu'elle est utilisée de manière extensive dans les problèmes variationnels, pour lesquels les termes quadratiques ont tendance à trop lisser le contenu de l'image. Des exemples d'applications se trouvent en débruitage [CP04], mais aussi par exemple pour la déconvolution [CW98], l'interpolation [CS01] ou l'extrapolation de spectre [ADR04].

Malgré le fait d'être un outil usuel d'optimisation, la variation totale définie dans l'équation (5.14) présente un certain nombre de défauts à l'origine de l'apparition d'artefacts sur l'image.

D'une part, elle n'est pas différentiable en zéro. [Nik00] montre que les fonctions de régularisation non lisses favorisent les images présentant des zones très homogènes. Cet artefact, appelé *staircasing*, fait apparaître des aplats sur l'image. L'utilisation d'une fonction lisse ϕ_α, à la place

de la norme l_1 permet d'éviter ce défaut. [NWC13] propose différents choix de fonctions ϕ. En particulier :

$$\phi_\alpha = |t| - \alpha \log(1 + \frac{|t|}{\alpha}) \tag{5.16}$$

Avec un α suffisamment petit, cette fonction constitue une bonne approximation lisse de la norme l_1, voir figure 5.3. L'autre intérêt de cette fonction est la simplicité du calcul de sa dérivée :

$$\phi'_\alpha(t) = \frac{t}{\alpha + |t|} \tag{5.17}$$

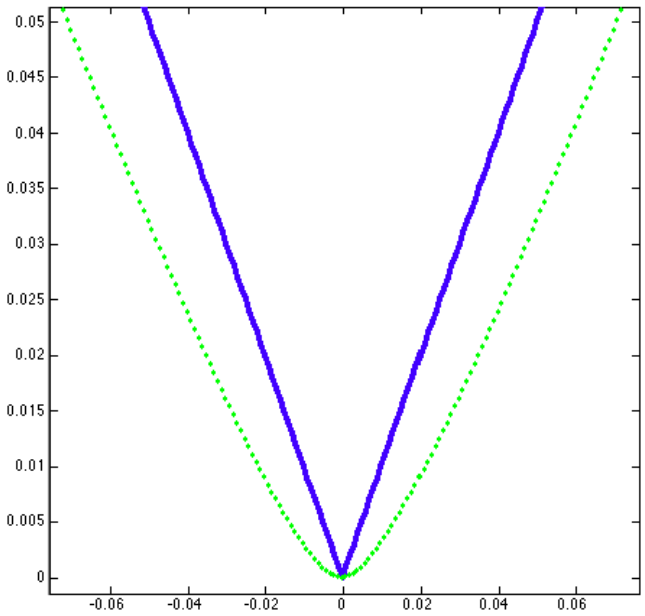


FIGURE 5.3 – **Approximation lisse de la norme l_1. -** En bleu : $\phi(t) = |t|$. En vert pointillé : $\phi_\alpha = |t| - \alpha \log(1 + \frac{|t|}{\alpha})$. On observe que ϕ_α est une bonne approximation dérivable en 0 de ϕ.

Plusieurs autres défauts de la variation totale peuvent encore être corrigés. Ces problèmes apparaissent tout particulièrement pour des traitements sous-pixelliques des images. En effet, l'utilisation de la TV^d suppose produit une image restaurée qui n'est pas interpolable et dépend de la grille d'échantillonnage. Un autre problème est le manque d'isotropie de la TV^d, dont la valeur varie par une modification de l'orientation de l'image, comme le détaille [Con17]. [AM17] fournit d'amples détails sur ces problèmes et propose une nouvelle définition de la variation totale afin d'apporter une solution, au prix de quelques transformées de Fourier. Cette nouvelle expression de la variation totale, dont la genèse se trouve dans [MG01] et [Moi07], est dénommée variation totale de Shannon (STV). Elle cherche une solution continue et bande limitée (donc interpolable) par l'estimation de la variation totale exacte de l'interpolée de l'image discrète. L'utilisation d'une somme de Riemann pour déterminer une approximation de l'intégrale permet aussi que le résultat soit beaucoup moins dépendant de la grille d'échantillonnage. Cette quasi-invariance aux translations est un atout important lorsque l'on a besoin d'interpoler finement ces images pour faire de la stéréoscopie sous-pixellique.

Pour un signal 1D :

$$STV^n(f) = \frac{1}{n} \sum_{k \in I_{nM}} |D_n f(k)| \tag{5.18}$$

avec $n \in \mathbb{N}$. La dérivée est obtenue analytiquement par l'expression suivante :

$$D_n f(k) = \frac{1}{nM} \sum_{-\frac{nM}{2} \leq \nu \leq \frac{nM}{2}} \mathcal{Z}_n \hat{f}(\nu) 2i\pi n \frac{\nu}{M} \exp^{\frac{2i\pi\nu k}{nM}} \tag{5.19}$$

$$\text{où : } \mathcal{Z}_n \hat{f}(\nu) = \begin{cases} 1 & \text{si } |\nu| < \frac{M}{2} \\ \frac{1}{2} & \text{si } \nu = -\frac{M}{2} \text{ et M est pair} \\ O & \text{sinon} \end{cases} \tag{5.20}$$

Elle correspond à la transformée de Fourier inverse de la fonction $(\nu \rightarrow \mathcal{Z}_n \hat{s}(\nu) 2i\pi n \frac{\nu}{M})$. L'implémentation du gradient nécessite donc le calcul de la transformée de Fourier du signal suréchantillonné d'un facteur n. [Abe16] vérifie qu'un facteur de suréchantillonnage de 2 ou 3 est en général suffisant.

Il faut néanmoins noter que cette expression présente toujours une singularité en 0, et qu'il peut donc être utile de la combiner à la version lisse pour s'affranchir de la difficulté d'un problème non lisse.

5.3 Expression du problème

5.3.1 La contrainte fréquentielle

Comme vu précédemment (en section 4.3), les bornes fréquentielles du domaine des franges $f_{\min}$ et $f_{\max}$ sont connues. Cette zone est étendue pour tenir compte de la modulation des franges par les fréquences de la scène non-homogène. On l'appelle $\tilde{\Omega}$. De manière empirique, on choisit une extension de 70 pixels qui couvre bien la zone décrite sans trop l'étendre. Il s'agit ici d'appliquer la régularisation à l'image uniquement dans la zone $\tilde{\Omega}$ et à maintenir son complémentaire intact. Pour cela, un masque à forme rectangulaire est appliqué sur le spectre de Fourier de u. La transformée de Fourier d'une image est notée par un accent circonflexe. L'opérateur transformée de Fourier est noté par $\mathcal{F}$. Afin de se libérer des artefacts liés à la non-périodicité du signal, l'image est étendue par symétrie axiale sur les bords de l'image et une fenêtre de Hamming est appliquée dessus (M) :

$$\hat{u}_{|_{\tilde{\Omega}^c}} = (1 - \tilde{\Omega}) \circ \mathcal{F}(M \circ u) \tag{5.21}$$

5.3.2 Le problème de minimisation

En utilisant le modèle additif présenté en 5.2 et la contrainte fréquentielle développée dans le paragraphe précédent, le problème variationnel s'écrit :

$$\begin{array}{rl} \min_{u,v} & STV(u) \\ \text{sous contraintes} & \hat{u}_{|_{\tilde{\Omega}^c}} = \hat{w}_{|_{\tilde{\Omega}^c}} \\ \text{et} & w = u + v \end{array} \tag{5.22}$$

Ce problème peut être formulé avec le multiplicateur de Lagrange β approprié sous la forme :

$$\min_{u,v} J(u,v) = STV(u) + \iota_{BL(\tilde{\Omega})}(v) + \beta||w-(u+v)||_2^2 \tag{5.23}$$

où $\iota_{BL(\tilde{\Omega})}(v)$ désigne la fonction indicatrice :

$$\begin{array}{ll} \iota_{BL(\Omega)}(v) & = \begin{cases} 0 \text{ si } v \in BL(\tilde{\Omega}) \\ \infty \text{ sinon} \end{cases} \\ \text{avec } BL(\Omega) & = \{v / \operatorname{supp}(\hat{v} \leq \tilde{\Omega})\} \end{array} \tag{5.24}$$

Par la suite la contrainte sur le modèle d'image est imposée tel que $\beta = \infty$. On peut alors écrire $v = w - u$ et minimiser seulement en fonction de u.

5.3.3 Algorithme

Une méthode classique de résolution du type de problème non lisse de l'équation (5.23) est l'utilisation d'algorithmes proximaux. L'opérateur de proximité de $J(u,v)$ est difficile à résoudre, mais si l'on considère cette fonction comme la somme de deux sous-fonctions, chacune d'elles a une expression plus simple. En posant :

$$\begin{array}{ll} J_1(u) & = STV(u) \\ J_2(u) & = \iota_{BL(\tilde{\Omega})}(w-u) \end{array} \tag{5.25}$$

On peut ne minimiser que par u puisque $v = w - u$. Alors le problème 5.22 s'écrit :

$$\min_u J_1(u) + J_2(u) \tag{5.26}$$

Comme vu à la section 5.2.2, l'opérateur de proximité de l'indicatrice est la projection, soit : $\operatorname{prox}_{J_2} = P_{BL(\tilde{\Omega})}$. Il n'y a pas d'expression analytique pour $\operatorname{prox}_{\gamma J_1(u)}$, mais son calcul consiste en la résolution du classique problème de Rudin-Osher-Fatemi [ROF92]. Nous utilisons pour cela l'implémentation de Rémy Abergel de l'algorithme de Chambolle-Pock pour la variation totale de Shannon [AM17]. Enfin, l'algorithme de Douglas-Rachford, expliqué en section 5.2.3, exploite la décomposition de la fonction objectif pour minimiser le problème. L'*Optimization Toolbox* de Gabriel Peyré [Pey10] est utilisée pour son implémentation. Elle reprend la version relaxée de l'algorithme de [CP10] :

$$\left\{ \begin{array}{ll} u_{k+\frac{1}{2}} & = \operatorname{prox}_{J_2}(u^k) \\ u^{k+1} & = u^k + \mu(\operatorname{prox}_{\gamma J_1}(2u^{k+\frac{1}{2}} - u^k) - u^{k+\frac{1}{2}}) \end{array} \right. \tag{5.27}$$

Aux itérations de cet algorithme, il faut ajouter la résolution itérative du sous-problème d'estimation de $\operatorname{prox}_{\gamma J_1}$. Le paramètre de relaxation μ est choisi dans $]0,2[$. Pour l'algorithme complet de la résolution du problème additif, voir l'encadré Algorithm 1.

5.4 Choix des paramètres

Les bornes fréquentielles des franges f_{min} et f_{max} sont calculées en accord avec le paragraphe 4.3. Les autres paramètres sont choisis de manière expérimentale. Le paramètre de relaxation vaut $\mu = 1$. Le critère d'arrêt est choisi à $\epsilon = 10^{-5}$. Le paramètre de l'opérateur proximal de la TV dépend de la valeur moyenne de l'image et vaut $\gamma = 5$.

Algorithm 1 Minimisation du problème additif

INPUT : $f_{\min}$, $f_{\max}$, γ, μ, ϵ
Require: (a) Normalise $w \leftarrow \frac{w}{\max(w)}$
(b) Initialise $u_0 = w$
while $\|u^{k+1} - u^k\| \leq \epsilon$ **do**
$uk + \frac{1}{2} = P_{BL(\Omega)}(u^k)$
$u^{k+1} = u^k + \mu(\text{prox}_{\gamma J_1}(2u^{k+\frac{1}{2}} - u^k) - u^{k+\frac{1}{2}})$ {*Iterate to solve subproblem* $\text{prox}_{\gamma J_1}$}
end while
OUTPUT : Recovered image : $u = u \times \max(w)$

5.5 Résultats

5.5.1 Résultats sur image simulée

L'algorithme est d'abord testé sur les images simulées décrites en section 3.1 afin d'évaluer quantitativement ses performances. D'abord, les images simulées sont exploitées pour construire des images sur un modèle additif :

$$\text{image simulée additive} = \text{image simulée panchromatique} + \text{image simulée des franges} \quad (5.28)$$

Les images simulées selon le modèle physique sont également testées.

Globalement, les deux types d'images fournissent des résultats où les franges ont bien été séparées. Elles sont en outre nettes, ce qui est nécessaire dans la perspective d'une utilisation stéréoscopique. Ces observations peuvent être confirmées sur les figures 5.4 et 5.5. Les transformées de Fourier mettent en évidence que des franges de la scène ont bien été recouvrées dans la zone des franges, tout en excluant celles-ci.

Néanmoins, certains défauts apparaissent sur les images physiques, comme le montre la figure 5.6, sur des éléments avec de fortes variations radiométriques. Cela ne s'observant pas sur les simulations additives, ce défaut est dû à une inadéquation entre le modèle additif utilisé pour résoudre le problème et le modèle "réel" de formation d'image. Ce problème est davantage commenté par la suite.

Ces constatations qualitatives se constatent sur les résultats du PSNR, défini en 3.3. La table 5.1 donne systématiquement des résultats légèrement meilleurs pour l'image simulée sur un modèle additif, ce qui peut se justifier par les limites du modèle sur les images physiques.

5.5.2 Sur images Sieleters

Les résultats sur les images Sieleters confirment les observations sur images simulées. La solution est globalement très satisfaisante. Sur les images, les franges ont été presque totalement effacées, comme le montre la figure 5.7. On remarquera que le choix de la variation totale se justifie bien dans le fait que les images n'ont pas été détériorées en termes de netteté. Les transformées de Fourier montrent bien que dans le domaine $\tilde{\Omega}$, des fréquences de la scène ont été restituées. Les temps de calculs, pour un ordinateur de bureau classique (Processeur Intel Core i7 2GHz bi-coeur) sont d'un peu moins de 5 minutes par image. Ce temps de calcul peut sembler long, et gagnerait à être optimisé. Cependant, cela suffit pour une utilisation en post-traitement.

TABLE 5.1 – **Résultats du PSNR pour les images simulées. -** Les images correspondent à celles montrées en figure 3.2. Le PSNR est mesuré entre l'image panchromatique des images simulées et le résultat de l'algorithme. L'image simulée additive correspond à une image initiale où les franges et l'image panchromatique ont été additionnées. C'est pourquoi les résultats sont légèrement meilleurs. L'image simulée physique a été mesurée de telle sorte à approcher le modèle de formation d'image réel, avec les mêmes paramètres d'acquisition que Sieleters (voir section 3.1).

	Image avec franges	Résultats sur image simulée additive	Résultats sur image simulée physique
Image a	25.04	49.86	49.30
Image b	24.95	48.91	48.72
Image c	24.74	48.71	47.82
Image d	25.12	48.12	47.40
Image e	24.73	47.77	47.08
Image f	24.34	46.45	45.39
Image g	24.73	47.02	46.00
Image h	25.06	45.58	44.98
Image i	25.00	46.43	45.60
Image j	25.02	40.40	39.97

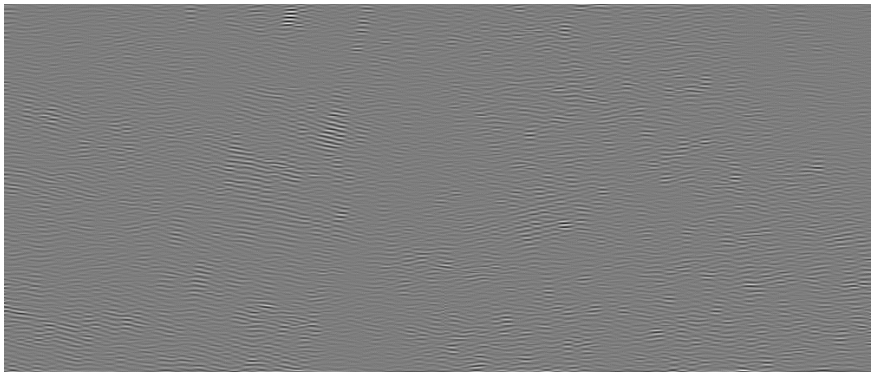
Néanmoins, la figure 5.7 et 5.8 met en évidence à nouveau que l'on perçoit des résidus de franges sur les zones à fort contraste. Ce problème résulte de la non adéquation du modèle additif à la réalité physique. En effet, les transformées de Fourier de la figure 5.7 montrent des répliques des franges d'interférence. Elles résultent de leur modulation par les hautes fréquences de la scène. On observe même que ces répliques influencent le résultat à l'intérieur de $\tilde{\Omega}$ en y réintroduisant des fréquences des franges. Or, le modèle additif n'a pas d'effet en dehors de $\tilde{\Omega}$ et s'avère insuffisant pour ce type de texture. Une mise en évidence plus précise de ce problème est apportée par les images en 5.9, qui isolent une texture haute fréquence, sa transformée de Fourier, et le masque appliqué par l'algorithme.

5.6 Conclusion

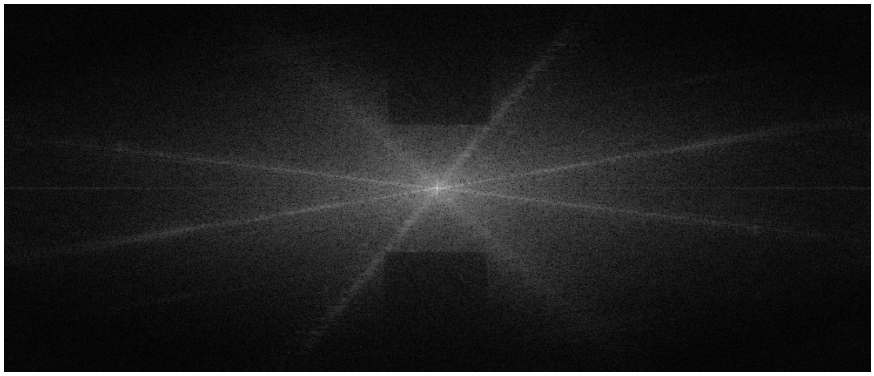
Ce chapitre a proposé un algorithme de séparation des franges de l'image de la scène en utilisant un modèle additif d'image. Celui-ci fournit globalement de très bons résultats, et ce modèle d'image est acceptable dans de nombreux cas puisque les basses fréquences sont dominantes. Toutefois, en présence de textures haute fréquence à fort contraste, il demeure des franges sur ces zones, bien qu'elles soient atténuées. Il semble donc attrayant de s'intéresser à l'utilisation d'un modèle multiplicatif. C'est le sujet du chapitre suivant.

(a) Référence

(b) Résultat de la décomposition : la scène u

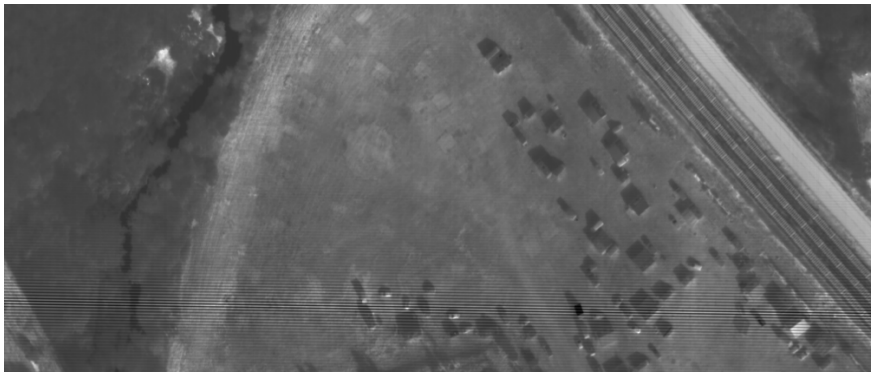


(c) Différence entre la référence et le résultat u

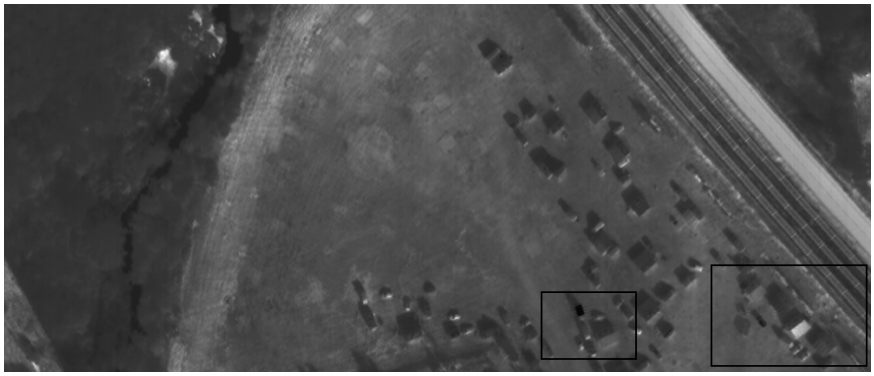


(d) Transformée de Fourier du résultat u

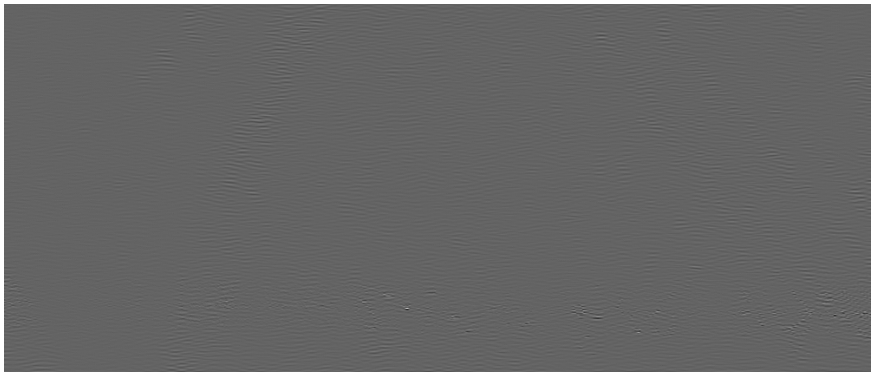
FIGURE 5.4 – **Résultat sur une image simulée additive.**



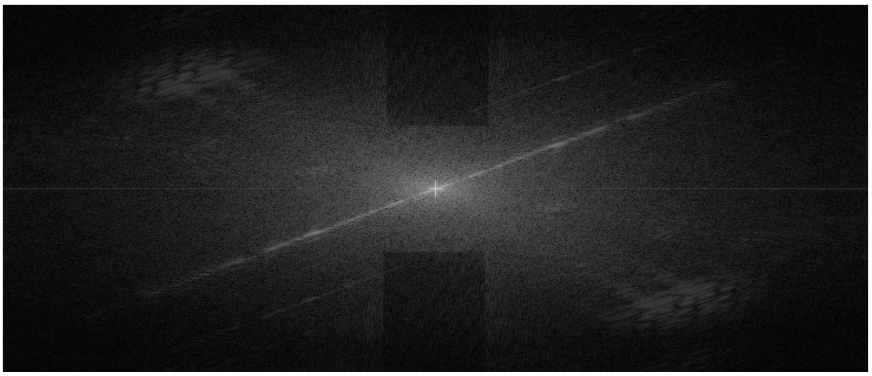
(a) Référence



(b) Résultat de la décomposition : la scène u

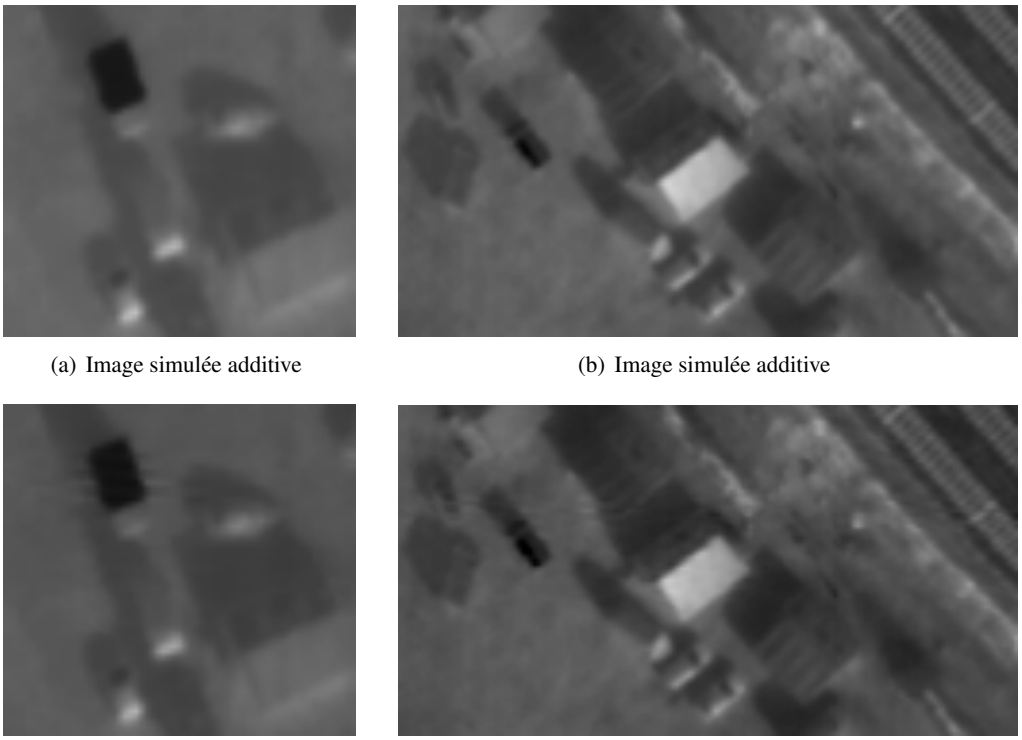


(c) Différence entre la référence et le résultat u



(d) Transformée de Fourier du résultat u

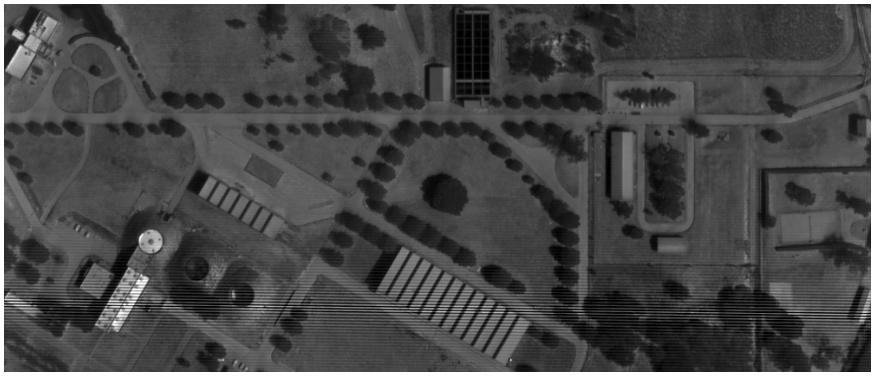
FIGURE 5.5 – **Résultat sur une image simulée physique.**



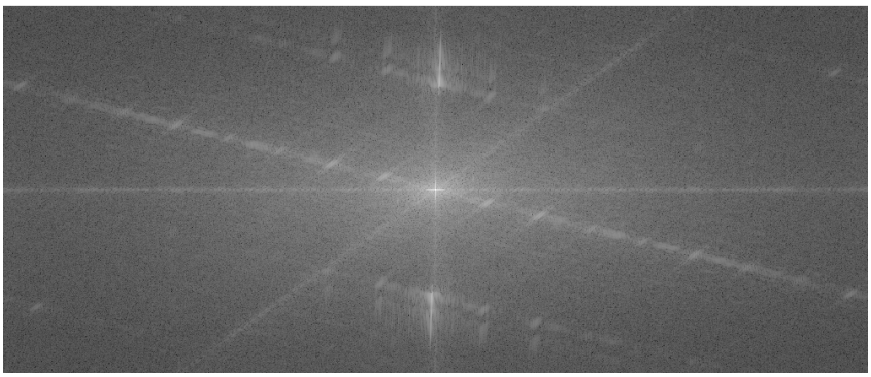
(a) Image simulée additive (b) Image simulée additive

(c) Image simulée physique (d) Image simulée physique

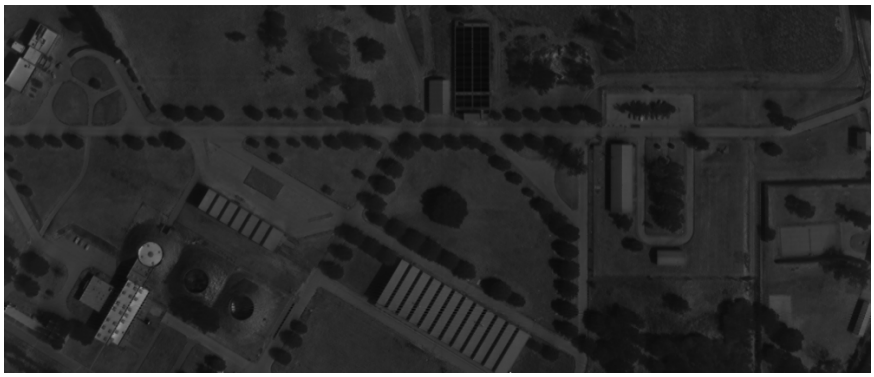
FIGURE 5.6 – **Détails de la scène en figure 5.5 sur les résultats obtenus à partir des images simulées physiques et additives. -** Sur les résultats des images simulées physiques, des franges demeurent autour des zones à fort contraste.



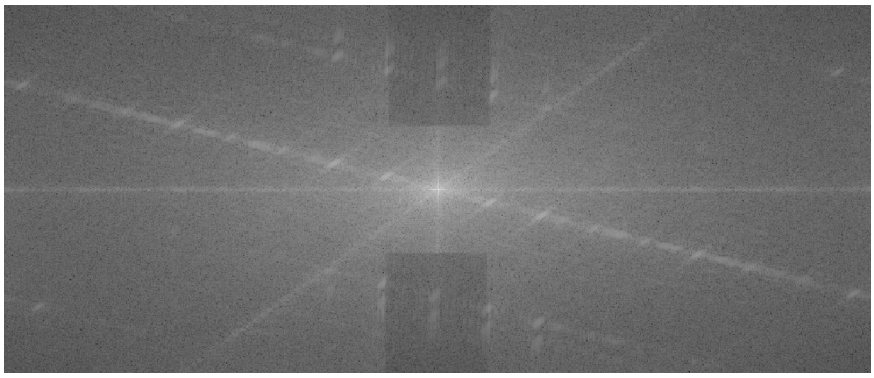
(a) Image originale Sieleters



(b) Transformée de Fourier de l'image originale

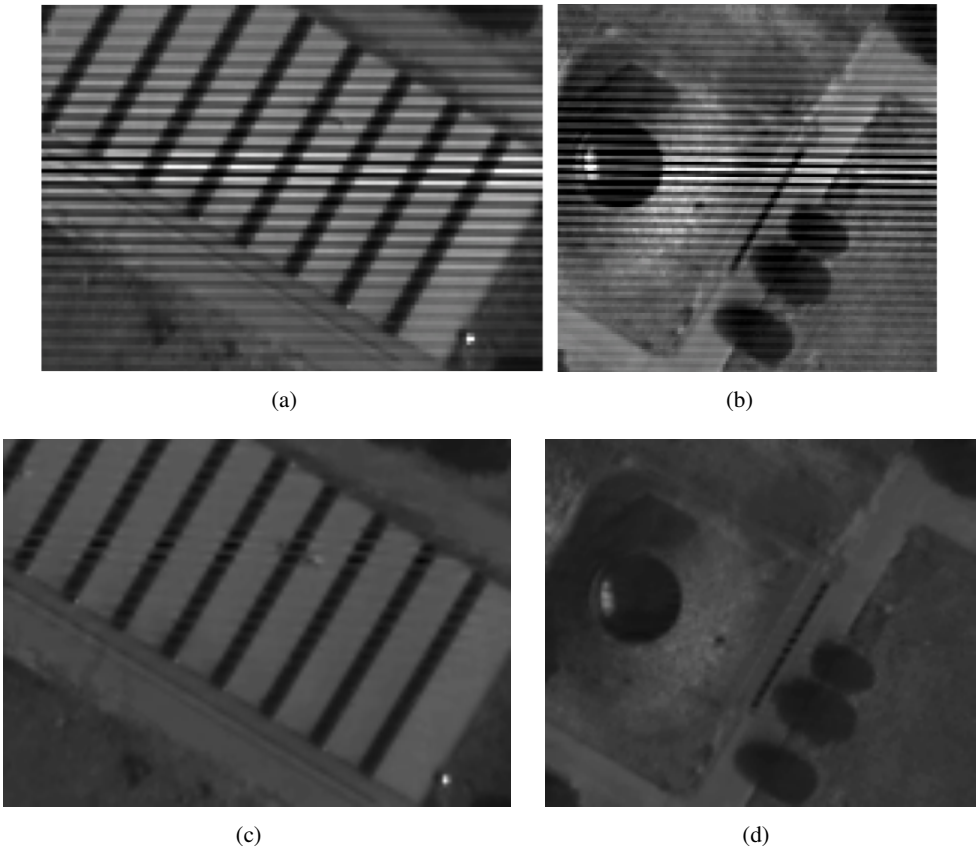


(c) Résultat



(d) Transformée de Fourier du résultat

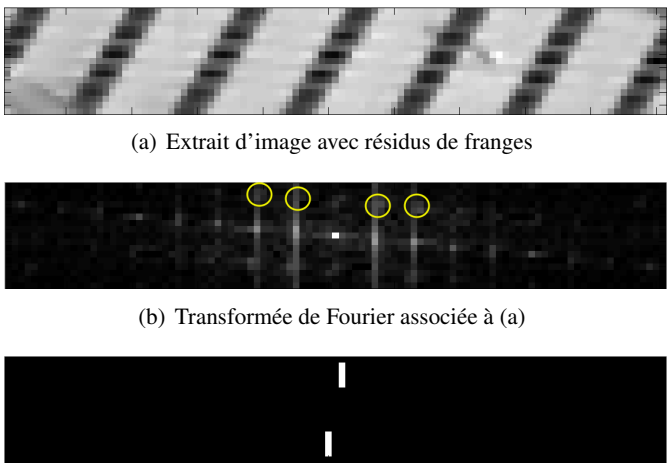
FIGURE 5.7 – **Résultats sur une image Sieleters. -** Les transformées de Fourier sont obtenues après prolongement symétrique de l'image et apodisation par une fenêtre de Hanning. On remarque sur l'image (d) que la zone $\widetilde{\Omega}$, bien que discernable, n'est pas vide, et qu'elle a bien été complétée par des fréquences de l'image.



(a) (b)

(c) (d)

FIGURE 5.8 – **Détails des images. -** (a) et (b) sont des imagettes issues de l'image Sieleters, dans une zone où les franges sont très contrastées, (c) et (d) sont ces mêmes détails après régularisation.



(a) Extrait d'image avec résidus de franges

(b) Transformée de Fourier associée à (a)

(c) Région Ω associée à (a)

FIGURE 5.9 – **Extrait de l'image de la figure 5.7 avec des résidus de franges -** Dans (a), les résidus de franges correspondent aux lignes horizontales. On remarque dans (b) que l'imagette contient des fréquences de franges dans Ω, mais aussi dans Ω^C (entourées en jaune).

CHAPITRE 6

UTILISATION D'UN MODÈLE D'IMAGE MULTIPLICATIF

Sommaire

6.1 Modèle de décomposition multiplicatif **68**
6.1.1 Dérivation physique du modèle 68
6.1.2 Difficultés du modèle 68
6.2 Description du problème **69**
6.2.1 La contrainte fréquentielle 69
6.2.2 La régularisation 72
6.2.3 Expression du problème 72
6.3 Algorithmes **73**
6.3.1 Algorithme par descente proximale 73
6.3.2 Algorithme par filtrage 74
6.3.3 Initialisation et normalisation 75
6.4 Choix des paramètres **75**
6.4.1 Paramètres de l'algorithme par descente proximale 76
6.4.2 Paramètres de l'algorithme par filtrage 78
6.5 Résultats **78**
6.5.1 Sur images simulées 78
6.5.2 Sur images Sieleters 78
6.6 Limites du modèle **79**
6.6.1 L'hypothèse sur la forme des spectres 79
6.6.2 Cas particuliers de textures 81
6.7 Conclusion **83**

Au vu des lacunes du modèle additif de formation d'image, ce chapitre explore l'utilisation d'un modèle plus proche de la physique de formation de l'image. Le modèle multiplicatif proposé ici a été développé en collaboration avec Daniel Soncco et Mila Nikolova. Le modèle de bruit multiplicatif fait l'objet d'études [SO08], mais à notre connaissance, l'utilisation d'un tel modèle pour la décomposition d'image est inédit.

6.1 Modèle de décomposition multiplicatif

6.1.1 Dérivation physique du modèle

Si l'on suppose que tous les objets de la scène ont la même forme spectrale, alors les variables spatiales et spectrales sont indépendantes, et la luminance spectrique S peut s'exprimer de la façon suivante :

$$S(x, y, \sigma) = B(\sigma)M(x, y) \tag{6.1}$$

L'équation (1.4) devient :

$$I(x, y) = \int_{\sigma_1}^{\sigma_2} B(\sigma)M(x, y)d\sigma + \int_{\sigma_1}^{\sigma_2} B(\sigma)M(x, y)C\cos(2\pi\delta(x, y)\sigma)d\sigma \tag{6.2}$$

$$I(x, y) = M(x, y)\left(\int_{\sigma_1}^{\sigma_2} B(\sigma)d\sigma + \int_{\sigma_1}^{\sigma_2} B(\sigma)C\cos(2\pi\delta(x, y)\sigma)d\sigma\right) \tag{6.3}$$

En posant :

$$B_{tot} = \int_{\sigma_1}^{\sigma_2} B(\sigma)d\sigma \tag{6.4}$$

On obtient :

$$I(x, y) = B_{tot}M(x, y)\left(1 + \frac{\int_{\sigma_1}^{\sigma_2} B(\sigma)C\cos(2\pi\delta(x, y)\sigma)d\sigma}{B_{tot}}\right) \tag{6.5}$$

Alors on obtient donc le modèle de décomposition suivant :

$$w(x, y) = u(x, y)(1 + v(x, y))$$

où le bruit est négligé au vu de la bonne qualité des images. w est l'image mesurée par l'instrument et u et v correspondent aux différents termes :

$$u(x, y) = B_{tot}M(x, y) \tag{6.6}$$

$$v(x, y) = \frac{\int_{\sigma_1}^{\sigma_2} B(\sigma)C(x, y, \sigma)\cos(2\pi\delta(x, y)\sigma)d\sigma}{B_{tot}} \tag{6.7}$$

Dans (6.6), en remplaçant B_{tot} par son expression, on retrouve bien l'expression de l'image panchromatique de la scène. v correspond strictement à la composante des franges sur Ω décrite à la section 4.3.

6.1.2 Difficultés du modèle

L'exploitation du problème en équation (5.23) avec un modèle multiplicatif d'image correspond à la recherche de u et v qui minimisent la fonctionnelle $J(u, v)$ suivante :

$$J(u, v) = TV(u) + \iota_{BL(\Omega)}(v) + \beta\|w - u \circ (1 + v)\|_2^2 \tag{6.8}$$

où $\circ$ dénote le produit d'Hadamard.

Or ce problème est biconvexe, c'est-à-dire qu'en fixant respectivement u et v quelconques, on obtient deux fonctions convexes J_u et J_v telles que :

$$\begin{array}{lll} \forall v \in (v), & J_u(v) & = \iota_{BL(\Omega)}(v) + \|w - u \circ (1+v)\|_2^2 \\ \forall u \in (u), & J_v(u) & = TV(u) + \|w - u \circ (1+v)\|_2^2 \end{array} \tag{6.9}$$

[GPK07] fournit une référence sur l'étude de ces problèmes et leur résolution. La minimisation d'une fonction biconvexe est rendue difficile du fait de la possible présence de nombreux minima locaux et points de selle, et le résultat dépendra alors de la qualité de l'initialisation. De ce fait, les algorithmes ne démontrent souvent qu'une convergence vers un point critique. Une illustration simple peut être donnée par la fonction $f(x, y) = |xy - 1|$ qui fournit un ensemble de minima $\{(x, y) : xy = 1\}$. On trouvera également des points critiques correspondants à des points singuliers sur la fonction $f(x, y) = |xy - 1| + |x| + |y|$. Elle possède également un minimum local en $(-1, -1)$. La figure 6.1 illustre ces deux fonctions.

Ces problèmes sont donc en général résolus par des algorithmes d'optimisation globale, mais il est également possible d'exploiter la structure convexe selon chacune des variables afin de dériver des algorithmes adaptés et plus efficaces. Ceux-ci peuvent être trouvés dans [GPK07], et davantage d'indications sont fournies sur l'algorithme utilisé dans la section 6.3.

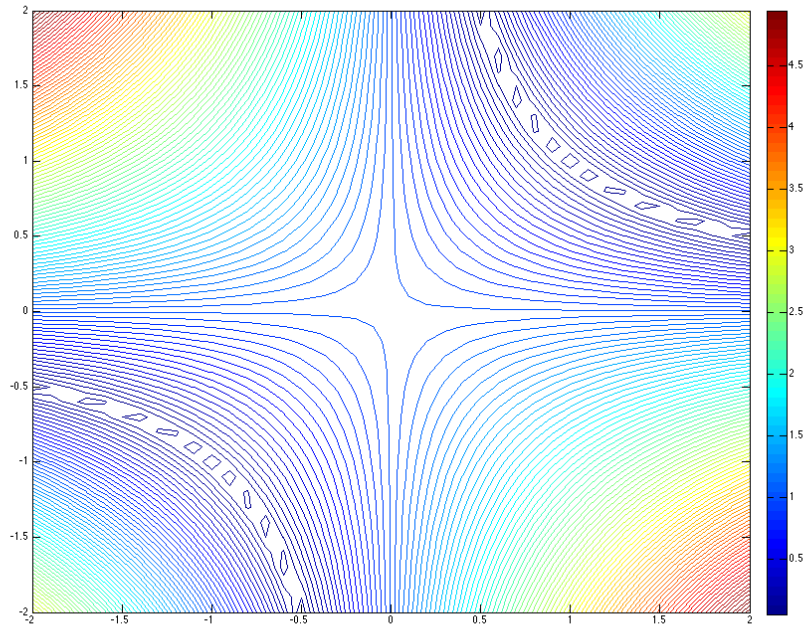
La spécificité et la complexité des problèmes biconvexes explique que la littérature associée soit restreinte. Dans le domaine du traitement de l'image, on peut citer [Agu+17], qui minimise une fonction biconvexe pour une application de débruitage. L'algorithme de minimisation est une descente alternée avec une preuve de convergence vers un point critique.

6.2 Description du problème

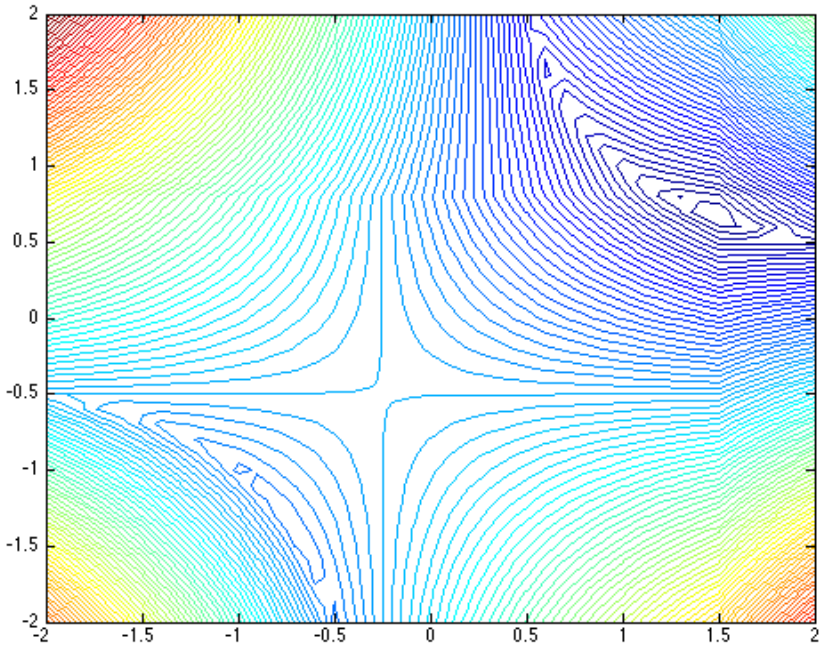
Le problème multiplicatif affine certaines des hypothèses présentes dans le modèle additif. Elles sont présentées dans cette section, avant d'aborder les deux algorithmes développés. Le premier est convergent et pourrait suffire, mais dans la pratique c'est davantage le second qui est utilisé pour sa rapidité, ce qui justifie qu'il soit évoqué ici, malgré le fait qu'il n'ait pas de preuve de convergence.

6.2.1 La contrainte fréquentielle

La contrainte fréquentielle des franges subit un traitement légèrement différent par rapport à l'algorithme additif. L'observation sur les images simulées de la transformée de Fourier (DFT) 1D selon les lignes de la vérité terrain et de w, l'image avec franges, révèle qu'elles sont très proches. Cela peut se voir sur la figure 6.2. En revanche, la DFT selon les colonnes donne des profils très différents, marqués pour w par les fréquences des franges, que l'on retrouve bien entre 40 et 100 puis 320 et 380. Cela s'explique par le fait que les franges sont quasi-horizontales, et qu'elles affectent donc uniquement les colonnes de la transformée de Fourier. La contrainte fréquentielle se réduit donc à une transformée de Fourier 1D verticale. Cette configuration apporte également l'avantage de diminuer le temps de calcul par rapport à une DFT 2D, et de limiter l'apparition possible d'artefacts liés à l'approximation de la transformée de Fourier.

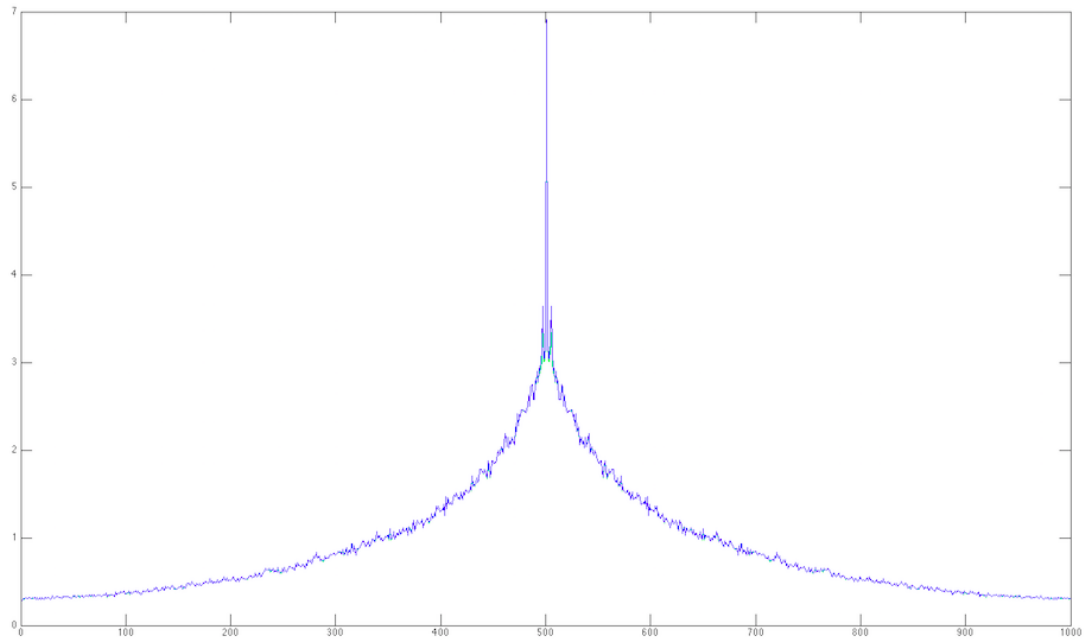


(a) $f(x, y) = |xy - 1|$

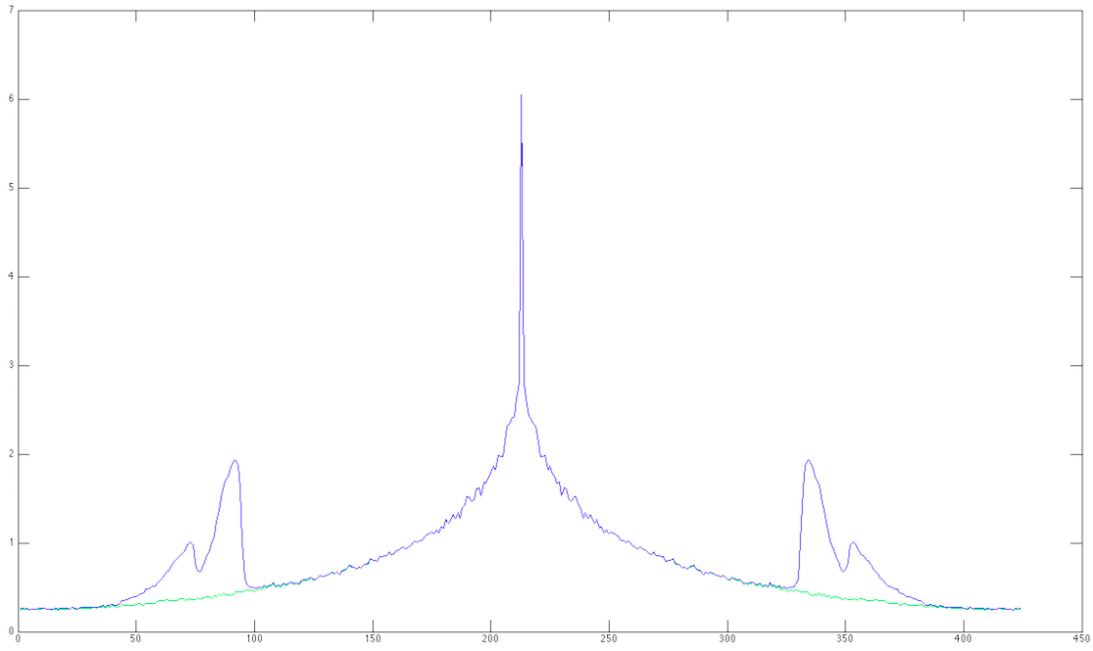


(b) $f(x, y) = |xy - 1| + |x| + |y|$

FIGURE 6.1 – **Isolignes de fonctions biconvexes -** (a) La fonction présente des minima locaux sur les courbes des deux hyperboles $xy = 1$. (b) La fonction présente un minium et plusieurs points stationnaires ne correspondant pas à des minima.



(a) Selon les lignes



(b) Selon les colonnes

FIGURE 6.2 – **Comparaison des transformées de Fourier d'une image simulée panchromatique et sa correspondante avec franges -** En vert, valeurs pour l'image panchromatique et en bleu l'image avec franges. Les courbes sont réalisées à partir de la moyenne des lignes ou colonnes d'une image. (a) Les transformées de Fourier selon les lignes présentent des valeurs très proches pour les deux images. (b) Pour les colonnes, les courbes diffèrent autour des fréquences propres des franges.

L'application de cette contrainte comme un filtre passe-bande fournit une initialisation à l'algorithme.

6.2.2 La régularisation

Comme sur la méthode additive, u, la composante de l'image de la scène, est régularisée. On ajoute également une régularisation sur les franges afin de renforcer la contrainte sur la géométrie des franges.

Régularisation sur u

Les perturbations des franges sont horizontales. La régularisation sur la composante de la scène est donc faite uniquement selon la dimension verticale afin de contraindre la disparition des franges en minimisant la variation totale.

Régularisation sur v

Suivant le même raisonnement que précédemment, sur la composante v, c'est la dimension horizontale seule qui est régularisée. En effet, les oscillations selon les colonnes sont caractéristiques des franges d'interférences, et ne doivent donc pas être pénalisées. En revanche, les lignes se doivent d'être régulières pour renforcer la contrainte fréquentielle. Les franges n'étant pas tout à fait horizontales par rapport au détecteur, cette régularisation doit néanmoins être pondérée.

6.2.3 Expression du problème

En reprenant les points précédents, le problème à résoudre peut s'exprimer comme la minimisation de la fonctionnelle suivante :

$$\inf_{u,v} J(u,v) \tag{6.10}$$

$$\text{avec } J(u,v) = \lambda\Phi(u) + \Psi(v) + \beta\frac{\|T(v)\|_2^2}{2} + \gamma\frac{\|u\circ(1+v)-w\|_2^2}{2} \tag{6.11}$$

Les deux premiers termes correspondent aux régularisations sur u et v. On utilise une variation totale de Shannon lissée. Réaliser les régularisations sur une seule dimension présente l'avantage de réduire les temps de calcul. Cela est d'autant plus nécessaire du fait de l'utilisation de la variation totale de Shannon, qui ajoute le calcul d'une DFT sur une image interpolée. Ainsi, pour une image de taille $M \times N$ sur le domaine $I_M \times I_N$:

$$\Phi(u) = \sum_{k\in I_{nM}} \phi_\alpha(D_n^v u(k)) \tag{6.12}$$

$$\Psi(v) = \sum_{k\in I_{nN}} \phi_\alpha(D_n^h v(k)) \tag{6.13}$$

D_n est l'opérateur différentiel défini en (5.19) et ϕ_α la fonction définie en (5.16). Le troisième terme correspond à la contrainte fréquentielle : $T = \Omega_{1D} \circ \mathcal{F}_v(M \circ v)$ où M est l'opérateur de prolongement de l'image par symétrie sur laquelle est appliquée une fenêtre de Hamming, $\mathcal{F}_v$ est la transformée de Fourier 1D appliquée colonne à colonne, Ω est le masque fréquentiel et $\circ$ dénote le produit terme à terme.

Le dernier terme de $J(u, v)$ est l'attache aux données suivant le modèle d'image multiplicatif. Cette contrainte est ici relachée par rapport au modèle additif.

Le choix des paramètres λ, β et γ est détaillé plus loin de ce chapitre. La fonctionnelle (6.11) est biconvexe. Les deux algorithmes développés pour résoudre le problème sont expliqués ci-après.

6.3 Algorithmes

Cette section présente deux algorithmes pour résoudre le problème de la minimisation de la fonction (6.11). Le premier réalise une descente proximale alternée sur chacune des variables. Le second exploite les expressions explicites des dérivées pour procéder à la façon d'un filtrage non-linéaire. Ce dernier n'a pas de preuve de convergence mais fournit une bonne solution en quelques itérations. Ces deux algorithmes sont initialisés par un oracle dont l'obtention est également détaillée.

6.3.1 Algorithme par descente proximale

Une méthode commune pour minimiser une fonctionnelle biconvexe est d'utiliser un algorithme de descente convexe alternée (*Alternate Convex Search*), voir [GPK07]. Celui-ci exploite la convexité de J_u et J_v telles que définies en section 6.1.2. L'idée est de fixer tour à tour chacune des variables u et v et d'utiliser des méthodes de minimisation convexe. Pour cela, on exprime la fonctionnelle sous la forme suivante :

$$J(u, v) = F(u) + G(v) + H(u, v) \tag{6.14}$$

avec

$$\begin{aligned} F(u) &= \Phi_{\alpha_1}(u) \\ G(v) &= \lambda\Psi_{\alpha_2}(v) + \beta\frac{\|T(v)\|_2^2}{2} \\ H(u, v) &= \gamma\frac{\|u(1+v)-w\|_2^2}{2} \end{aligned} \tag{6.15}$$

L'algorithme de minimisation de $J(u, v)$ procède par descente proximale, telle qu'elle a été expliquée en section 5.2.2, suivant la méthode de Gauss-Seidel en déduisant u et v de leurs valeurs antérieures et en les mettant à jour successivement. Une itération s'écrit donc :

$$\begin{cases} u^{(k+1)} &= \mathrm{prox}_{\tau_1 H(\bullet, v^{(k)})}(u^{(k)} - \tau_1 \nabla F(u^{(k)}) \\ v^{(k+1)} &= \mathrm{prox}_{\tau_2 H(u^{(k+1)}, \bullet)}(v^{(k)} - \tau_2 \nabla G(v^{(k)}) \end{cases} \tag{6.16}$$

L'algorithme entier est détaillé ci-après. [NT17] fournit une preuve de convergence globale vers un point critique pour des algorithmes biconvexes ou multi-convexes de ce type avec des fonctions F et G lisses.

L'initialisation et la normalisation de l'algorithme sont expliquées dans la section 6.3.3. Les gradients des fonctions F, G et H ont des expressions analytiques simples. $D_{n,v}$ et $D_{n,h}$ désignent l'opérateur de la dérivée de l'interpolée de Shannon défini par l'équation (5.19).

Algorithm 2 Minimisation de $J(u, v)$ par descente proximale

INPUT : $f_{\min}$, $f_{\max}$, τ_1, τ_2, n_{iter}, α_1, α_2, λ, β, γ
Require: (a) Compute $c_1 := \text{mean}(w)$ and $c_2 := \frac{1}{8}\ \text{std}(w)$
(b) Compute oracle $u^{(0)}$
(c) Normalize : $w \leftarrow +\frac{w-c_1}{c_2}$ and $u^{(0)} \leftarrow +\frac{u^{(0)}-c_1}{c_2}$
for $k = 0$ to $n_{\text{iter}} - 1$ **do**
$u^{(k+1)} = \text{prox}_{\tau_1 H(\,\cdot\,, v^k)}(u^{(k)} - \tau_1 \nabla F(u^{(k)}))$ {*Proximal descent in* u }
$v^{(k+1)} = \text{prox}_{\tau_2 H(u^{k+1}, \,\cdot\,)}(v^{(k)} - \tau_2 \nabla G(v^{(k)}))$ {*Proximal descent in* v}
end for
OUTPUT : recovered image $u = c_1 + (u^{n_{\text{iter}}} - 1)\, c_2$

$$\begin{aligned} \nabla F(u) = &\quad \lambda D_{n,v}^T \phi'_{\alpha_1}(D_{n,v} u) \\ \nabla G(v) = &\quad \beta T^T T v + D_{n,h}^T \phi'_{\alpha_2}(D_{n,h} v) \\ \nabla_u H(u,v) = &\quad \gamma(v+1) \circ (u \circ (v+1) - w) \\ \nabla_v H(u,v) = &\quad \gamma u \circ (u \circ (1+v) - w) \end{aligned} \tag{6.17}$$

Ceci permet de déduire une expression analytique de l'opérateur de proximité :

$$\text{prox}_{\tau_1 H(.,v)}(z) = \arg\min_u \|u - z\|_2^2 + \|u \circ (v+1) - w\|_2^2 \tag{6.18}$$

$$\frac{\partial}{\partial u}(\|u - z\|_2^2 + \|u \circ (v+1) - w\|_2^2) = u - z + \gamma\tau_1 (v+1) \circ (u \circ (v+1) - w) \tag{6.19}$$

Et donc :

$$\text{prox}_{\tau_1 H(.,v)}(z) = \frac{z + \tau_1 \gamma \circ (v+1) w}{1 + \tau_1 \gamma (v+1) \circ (v+1)} \tag{6.20}$$

De la même façon, on trouve :

$$\text{prox}_{\tau_2 H(u,.)}(z) = \frac{z + \tau_2 \gamma u \circ (u - w)}{1 - \tau_2 \gamma u \circ u} \tag{6.21}$$

Les valeurs choisies pour les différents paramètres sont davantage discutées en 6.4.

6.3.2 Algorithme par filtrage

En s'inspirant des travaux de [NS14], il est possible, puisque les dérivées ont une expression analytique, de procéder de manière semblable à une méthode itérative de filtrage non linéaire. L'utilisation d'un tel algorithme présente l'avantage d'être plus rapide. C'est à ce titre qu'il est en pratique préféré à l'algorithme par descente proximale.

L'algorithme 3 est une version approchée d'un filtrage. En effet, chacune des itérations est composée d'une seule itération de descente de gradient sur la variation totale sur u et v et du filtrage des fréquences des franges. Une vingtaine d'itérations de l'algorithme sont suffisantes à l'obtention d'un résultat proche de celui de l'algorithme par descente proximale.

Algorithm 3 Filtrage non-linéaire itératif

INPUT : $f_{\min}$, $f_{\max}$, α_1, α_2, δ_1, δ_2, n^{iter}

Require: (a) Compute $c_1 := \text{mean}(w)$ and $c_2 := \frac{1}{8}\ \text{std}(w)$

(b) Compute oracle $u^{(0)}$

(c) Normalize $w \leftarrow +\frac{w-c_1}{c_2}$ and $u^{(0)} \leftarrow 1 + \frac{u^{(0)}-c_1}{c_2}$

for $k = 0$ to $n^{\text{iter}} - 1$ **do**

$\bar{u}^{(k)} = u^{(k)} - \delta_1 \nabla \Phi_1(u^{(k)})$ {*Decrease* $\Phi(u)$ }

$\bar{v}^{(k)} = w/\bar{u}^{(k)} - 1$ { *Preserve data-fidelity*}

$\tilde{v}^{(k)} = M^{\dagger} \circ \left(\mathcal{F}^{-1}\Big((1-\Omega) \circ \mathcal{F}(M \circ \bar{v}^{k}) \Big) \right)$ { *Filtering* v *to approach* $T(v) = 0$ }

$v^{(k+1)} = \tilde{v}^{(k)} - \delta_2 \nabla \Psi_2(\tilde{v}^{(k)})$ { *Decrease* $\Psi(v)$ }

$u^{(k+1)} = w/(1 + v^{(k+1)})$ { *Preserve data-fidelity*}

end for

OUTPUT : recovered image $u = c_1 + (u^{n_{\text{iter}}} - 1)\, c_2$

FIGURE 6.3 – **Oracle pour les algorithmes multiplicatifs. -**

Bien qu'il n'ait pas de preuve de convergence, les tests sur différentes images montrent que les résultats sont proches de ceux de l'algorithme par descente proximale (et même meilleurs). Ces résultats peuvent être trouvés en section 6.5.

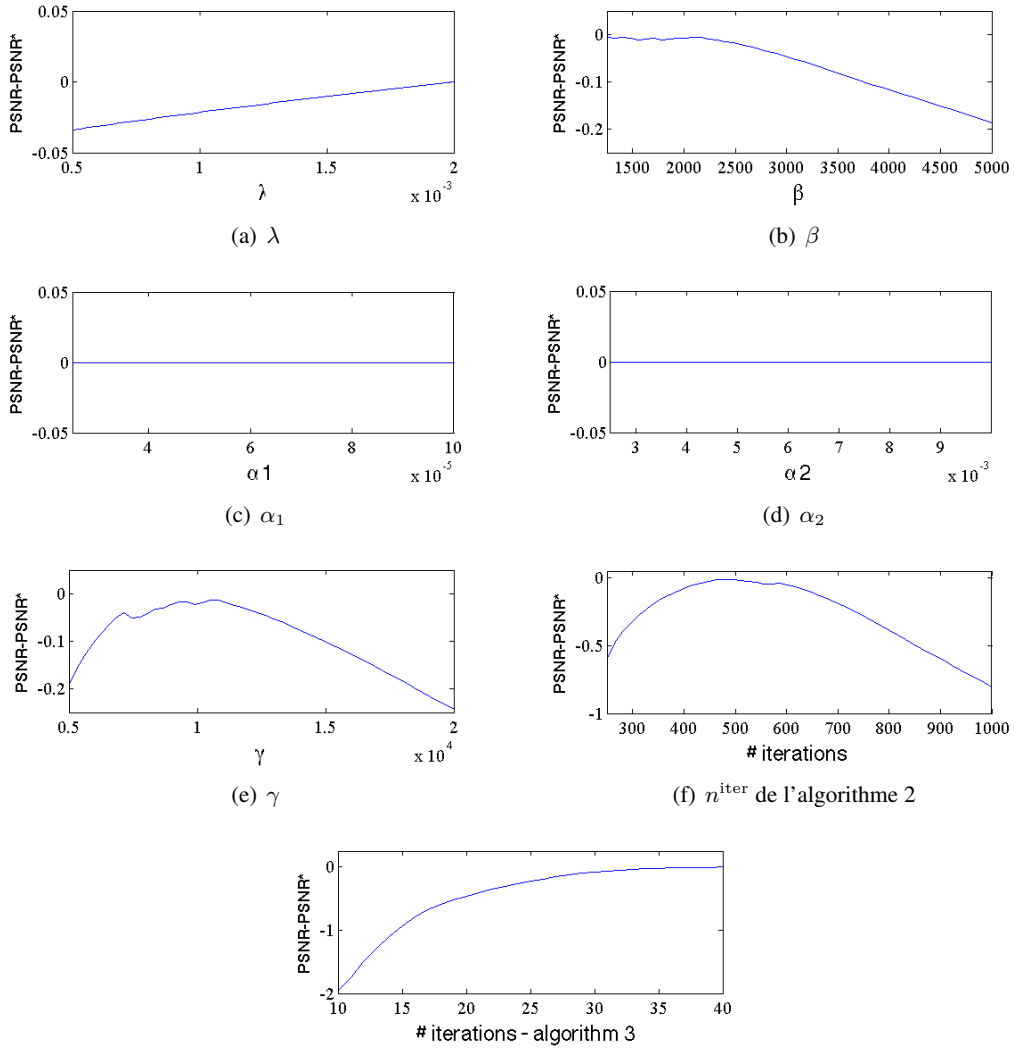
6.3.3 Initialisation et normalisation

Ces deux algorithmes sont initialisés par un oracle u^0 correspondant à l'image d'origine à laquelle est appliqué un filtre passe-bande sur la fréquence des franges. Le filtre est appliqué uniquement sur les colonnes, d'après les conclusions de la section 6.2.1.

De plus, l'image est normalisée avant l'algorithme. Cela permet de conserver les paramètres obtenus à partir des images simulées pour des images d'origines différentes. D'autre part, le résultat de la normalisation donnant des images de moyenne 1 et d'écart-type 0.125 évite en pratique d'avoir des pixels négatifs qui perturberaient les algorithmes.

6.4 Choix des paramètres

Cette section concerne le choix des paramètres des deux algorithmes de ce chapitre. Certains d'entre eux possèdent des bornes théoriques, tandis que d'autres ont été déterminés à partir d'expériences sur les images simulées évoquées en section 3.1. Pour l'ensemble d'entre eux, une analyse expérimentale de sensibilité du résultat aux variations de la valeur mesurée de chaque paramètre (les autres étant fixés) est présentée. Le calcul des bornes fréquentielles des franges f_{min} et f_{max} ne sera pas commenté puisqu'il a déjà été décrit en détail auparavant (section 4.3).



(a) λ (b) β

(c) α_1 (d) α_2

(e) γ (f) n^{iter} de l'algorithme 2

(g) n^{iter} de l'algorithme 3

FIGURE 6.4 – **Influence des variations des paramètres. -** Les valeurs données correspondent à la médiane des résultats sur images simulées. L'influence est mesurée par la différence entre le score de PSNR pour la meilleure valeur du paramètre et le score pour la valeur de paramètre donnée en abscisse. Les valeurs résultantes sont donc nulles (en particulier lorsque la valeur testée est celle qui a été choisie) ou négatives. Les algorithmes ne sont pas sensibles à des variations d'un facteur 2 autour de la valeur choisie.

6.4.1 Paramètres de l'algorithme par descente proximale

Le choix des paramètres α_1 et α_2 de la variation totale lisse suit les recommandations de [NWC13]. Ils doivent prendre des valeurs faibles afin de rester proche de la variation totale non lisse et ainsi de conserver les fortes discontinuités de l'image : $\alpha_1 = 5 \times .10^{-5}$, $\alpha_2 = 5 \times .10^{-3}$. La figure 6.4 montre que des variations autour de la valeur optimale mesurée n'influent guère sur le résultat.

Le paramètre λ gère l'équilibre de la régularisation de u. Il faut une valeur suffisamment petite pour ne pas trop lisser l'image, mais assez grande pour séparer la texture des franges : $\lambda = 0.001$.

Les coefficients β **et** γ doivent être élevés pour imposer les contraintes sur le modèle d'image et le domaine fréquentiel des franges. Les variations de ces paramètres (figure 6.4) montrent qu'ils correspondent à un optimum.

Le nombre d'itérations de l'algorithme n_{iter} est fixé. Il est choisi à 500 itérations d'après des observations expérimentales. En effet, c'est la valeur optimale observée en terme de PSNR.

D'après [NT17], le pas de descente τ est fixé à partir de la constante de Lipschitz Lip tel que :

$$\begin{aligned} \tau_1 &< \tfrac{2}{Lip\nabla F} \\ \tau_2 &< \tfrac{2}{Lip\nabla G} \end{aligned} \tag{6.22}$$

La constante de Lipschitz peut être déterminée à partir de la borne supérieure de sa dérivée :

$$\begin{aligned} Lip(\nabla F) &= \sup_{x,y} \tfrac{\|\nabla Fx - \nabla Fy\|}{x-y} \\ &\leq \|\nabla^2 F\| \end{aligned} \tag{6.23}$$

et

$$Lip(\nabla G) \leq \|\nabla^2 G\| \tag{6.24}$$

D'après 6.17 :

$$\begin{aligned} \nabla^2 F(u) &= \lambda D_{n,v} D_{n,v}^T \phi''_{\alpha_1}(D_{n,v}u) \\ \nabla^2 G(v) &= \beta T^T T + D_{n,h} D_{n,h}^T \phi''_{\alpha_2}(D_{n,h}v) \end{aligned} \tag{6.25}$$

Où :

$$\phi''_\alpha(t) = \frac{\alpha + |t| - t^2}{(\alpha + |t|)^2} \tag{6.26}$$

Puisque $\phi''_\alpha(t) \leq \frac{1}{\alpha}$ et que d'après ([Abe16],Eq.3.39) $\|\nabla_n\| \leq n\pi\sqrt{2}$, il est possible de calculer les bornes de $\nabla^2 G$ et $\nabla^2 F$:

$$\begin{aligned} \nabla^2 F(u) &\leq \tfrac{n\pi\sqrt{2}}{\alpha_1} \\ \nabla^2 G(v) &\leq \beta + \tfrac{n\pi\sqrt{2}}{\alpha_2} \end{aligned} \tag{6.27}$$

On a alors :

$$\begin{aligned} \tau_1 &< \tfrac{2\alpha_1}{n\pi\sqrt{2}} \\ \tau_2 &< \tfrac{2\alpha_2}{\beta\alpha_2 + n\pi\sqrt{2}} \end{aligned} \tag{6.28}$$

Globalement, les valeurs observées ne montrent pas de variations de plus de $1\mathrm{dB}$ pour l'ensemble de ces paramètres pour une variation d'un facteur 2, ce qui garantit une certaine robustesse.

6.4.2 Paramètres de l'algorithme par filtrage

Le choix du nombre d'itérations n_{iter} pour l'algorithme par filtrage non-linéaire est plus délicat. La figure 6.4 montre qu'une variation d'un facteur 2 cause un écart de $2\mathrm{dB}$, mais varie lentement au-delà de 20 itérations. C'est donc la valeur choisie.

Par analogie à l'algorithme par descente proximale, δ_1 et δ_2 sont choisis égaux à τ_1 et τ_2 respectivement.

6.5 Résultats

6.5.1 Sur images simulées

Les images simulées physiques, dont le calcul est présenté en section 3.1 permettent de fournir des résultats quantitatifs par rapport à une image panchromatique de référence. Le tableau 6.1 montre les résultats pour les deux algorithmes présentés dans ce chapitre. Ils sont tous deux très bons. Néanmoins, l'algorithme par filtrage se démarque par sa rapidité, de 16 à 25 fois plus rapide. On peut en déduire que malgré le fait que l'algorithme par filtrage n'ait pas de preuve de convergence, son efficacité le rend plus intéressant que l'algorithme par descente proximale.

	Image avec franges	Algorithme 2 Descente proximale		Algorithme 3 Filtrage non linéaire	
	PSNR	PSNR	CPU	PSNR	CPU
Image a	25.04	50.52	189	54.75	48
Image b	24.95	48.80	190	53.22	47
Image c	24.74	48.71	198	53.68	47
Image d	25.12	48.47	192	53.44	47
Image e	24.73	46.48	177	51.95	47
Image f	24.34	45.28	180	50.56	47
Image g	24.73	46.83	191	51.85	47
Image h	25.06	46.42	185	50.60	47
Image i	25.00	44.20	188	51.07	46
Image j	25.02	36.30	192	41.69	47

TABLE 6.1 – **Score de PSNR et temps de calcul (Processeur Intel Core i7 2GHz bi-coeur) pour les algorithmes utilisant un modèle multiplicatif.** Les images correspondent à celles montrées en figure 3.2. Le PSNR est mesuré entre l'image panchromatique des images simulées et le résultat de l'algorithme. L'image simulée physique a été mesurée de telle sorte à approcher le modèle de formation d'image réel, avec les mêmes paramètres d'acquisition que Sieleters (voir section 3.1).

6.5.2 Sur images Sieleters

En s'appuyant sur les images Sieleters, il est possible de comparer, en utilisant le PSNR, les résultats des algorithmes 2 et 3. Cela permet de vérifier que les résultats de l'algorithme rapide par filtrage ne s'éloignent pas trop de ceux de l'algorithme convergent.

Les histogrammes des valeurs de PSNR obtenues à partir de 30 images non redondantes sont donnés en figure 6.5. On constate que le PSNR est supérieur à $70\mathrm{dB}$. Les bons résultats de l'algorithme par filtrage non-linéaire sur les images simulées sont confirmées sur les données expérimentales.

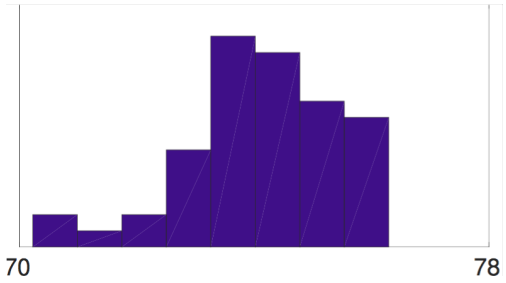


FIGURE 6.5 – **Histogrammes des résultats de la comparaison des deux algorithmes multiplicatifs par calcul du PSNR. -** Ces résultats sont obtenus par comparaison de 30 images non redondantes issues de Sieleters.

Pour une observation qualitative, les résultats de l'algorithme 3 sont montrés en figure 6.6.

6.6 Limites du modèle

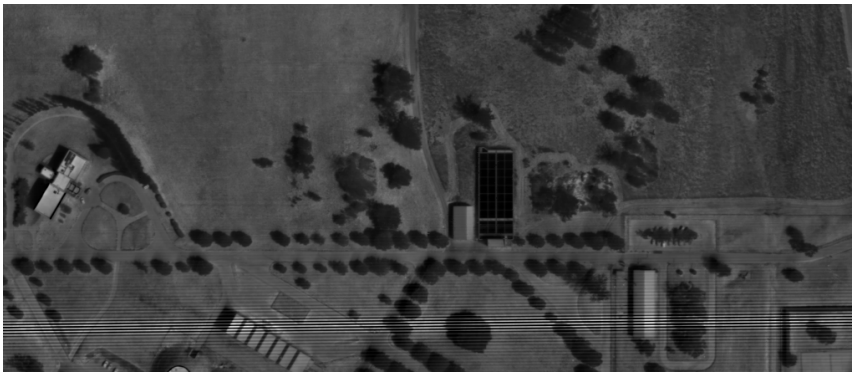
Plusieurs limitations peuvent conditionner la réussite de l'algorithme. La première se pose sur le modèle d'image, pour lequel une hypothèse sur la forme spectrale des objets de la scène a été posée. La seconde porte sur les textures de la scène.

6.6.1 L'hypothèse sur la forme des spectres

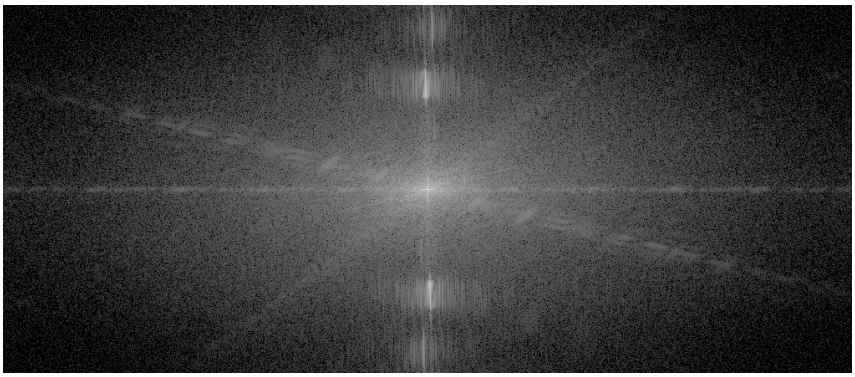
L'hypothèse émise à la section 6.1.1 afin de dériver le modèle multiplicatif suppose que tous les spectres des objets de la scène ont la même forme. Cette hypothèse n'est pas strictement vraie en pratique, mais nous vérifions ici si l'on s'en approche suffisamment pour les bandes spectrales de Sieleters ainsi que pour le spectre visible.

La figure 6.7 montre les spectres obtenus avec Sieleters sur la bande 3. Elle correspond à l'infrarouge thermique et traduit la température et l'émissivité propre des objets. On observe que tous les spectres ont une forme similaire. La bande 2, dans le moyen infrarouge, rend compte là encore de la température et de l'émissivité propres de l'objet. Les principales perturbations viennent de l'absorption du dioxide de carbone et de l'eau respectivement autour de $\sigma = 2300\mathrm{cm}^{-1}$ et $\sigma = 2000\mathrm{cm}^{-1}$. Les formes spectrales sont néanmoins assez proches, à l'exception des cibles chaudes. Pour ces deux intervalles spectraux, l'hypothèse est donc valable.

En revanche, pour le domaine visible, observable sur en figure 6.9, l'hypothèse ne tient plus. Les spectres correspondent alors uniquement à la réflectivité des objets, et les formes spectrales sont cette fois très différentes selon l'objet considéré.

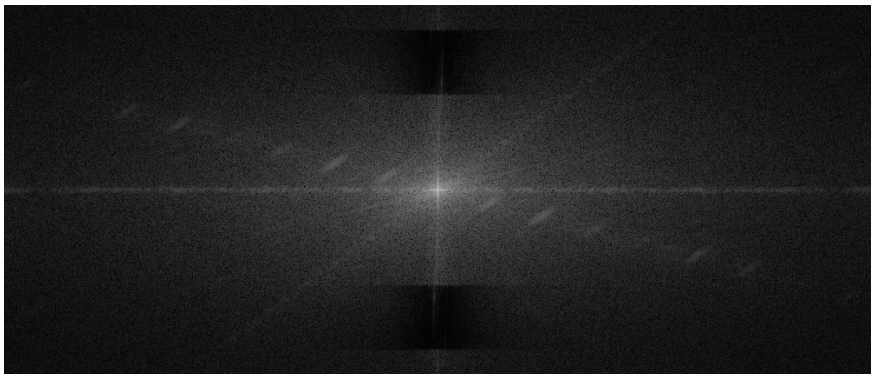


(a) Image originale Sieleters



(b) Transformée de Fourier de l'image originale

(c) Résultat par l'algorithme 3



(d) Transformée de Fourier du résultat

FIGURE 6.6 – **Résultats de l'algorithme 3 sur une image Sieleters. -**

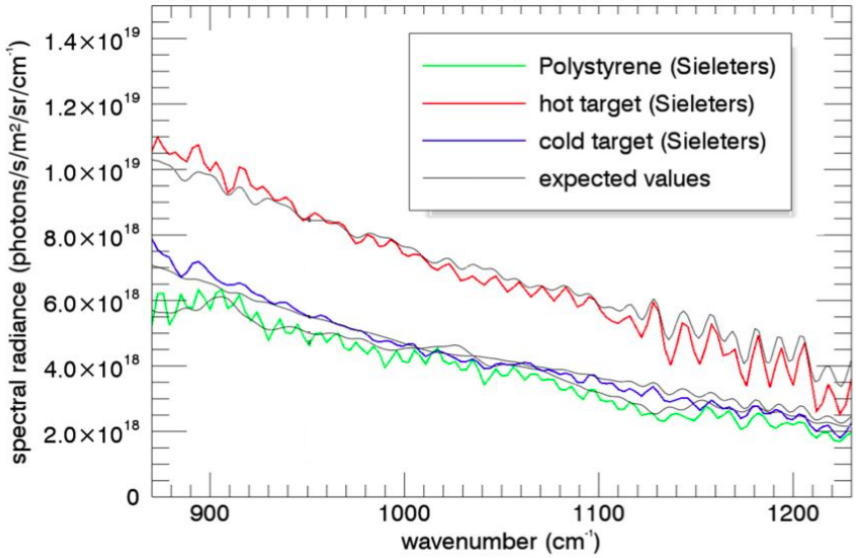


FIGURE 6.7 – **Spectres sur l'infrarouge thermique. -** Obtenu avec Sieleters sur diverses cibles. *Source : ONERA.*

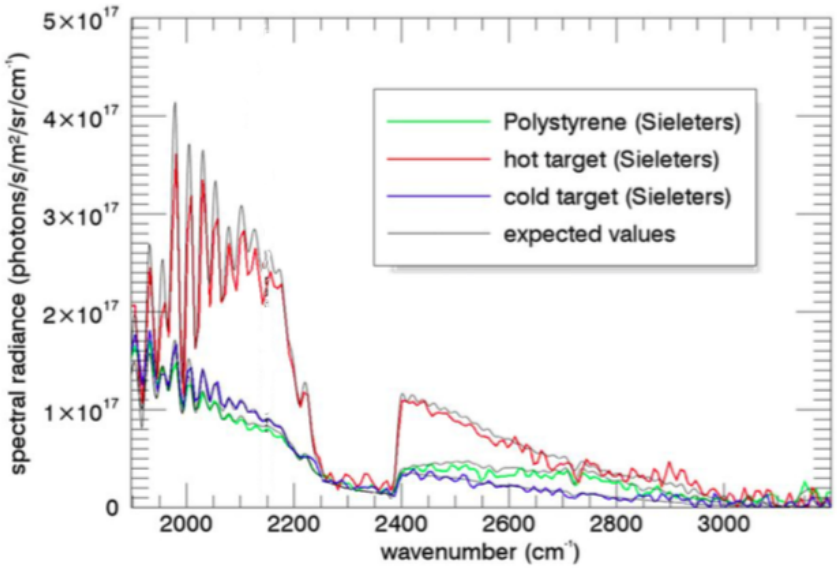


FIGURE 6.8 – **Spectres sur le moyen infrarouge. -** Obtenu avec Sieleters sur diverses cibles. *Source : ONERA.*

6.6.2 Cas particuliers de textures

Une autre limite de l'algorithme réside dans le cas où la scène possède des textures horizontales de même fréquence que les franges. Dans ce cas, l'algorithme a tendance à les traiter comme des franges et à les séparer de la scène. C'est ce que l'on observe sur les champs de la figure 6.10 : les traces horizontales sur les champs sont peu visibles sur le résultat de l'algorithme 3 alors que les

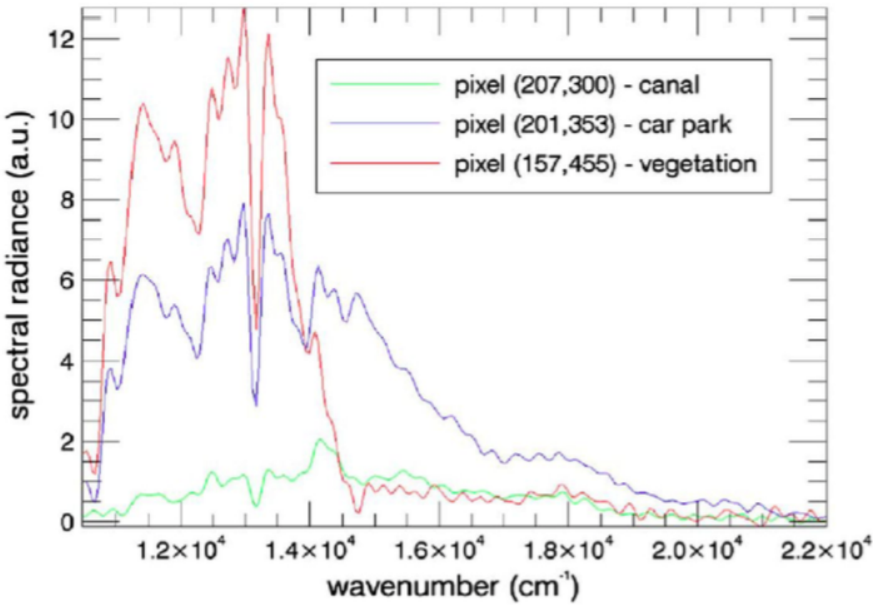
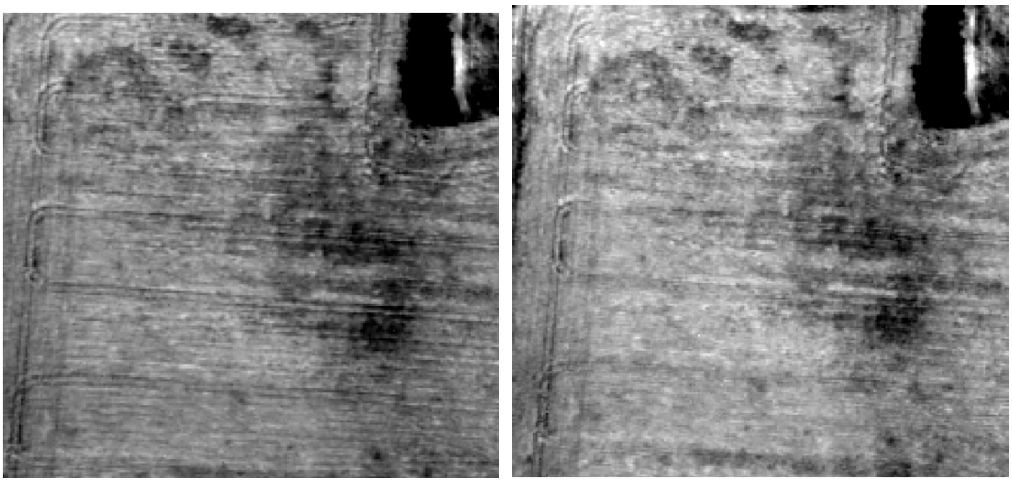


FIGURE 6.9 – **Spectres dans le domaine visibles. -** Obtenus avec l'instrument CaHyD. *Source : ONERA.*

traces verticales sont restées. Il faut toutefois admettre que ce type de texture très spécifique est rare.



(a) Image originale (b) Résultat après traitement

FIGURE 6.10 – **Détail d'une image Sieleters. -** Les traces horizontales des champs ont une fréquence et orientation proche de celle des franges. Pour cela, elles ont tendance à être effacées.

6.7 Conclusion

Ce chapitre a présenté deux algorithmes utilisant un modèle multiplicatif pour séparer des franges de la scène observée. La principale difficulté de ce travail réside dans le fait que le problème est biconvexe. L'un des algorithmes utilise une descente proximale et a une preuve de convergence. L'autre s'apparente à un filtrage non-linéaire. Les très bons résultats présentés pour ces algorithmes s'expliquent également car ils vont au-delà de la simple modification du modèle d'image, et optimisent les différentes contraintes pour obtenir des algorithmes rapides et efficaces.

Le modèle multiplicatif tout comme le modèle additif présenté dans le chapitre précédent supposent de formuler des hypothèses sur le spectre de l'image. La conclusion de cette partie compare ces deux modèles et précise leur cadre d'utilisation.

CONCLUSION

Avant de passer à la suite consacrée à la stéréoscopie, qui utilisera les images panchromatiques obtenues dans cette partie, nous comparons les deux modèles d'image utilisés pour la décomposition.

Comparaison des résultats des deux modèles d'images

Les résultats présentés ici confirment les suppositions avancées dans les chapitres précédents. En effet, l'observation de la première ligne de la figure 6.11 montre comment, sur une texture à variations radiométriques rapides, l'utilisation d'un modèle d'image multiplicatif parvient à enlever la totalité des franges d'interférence, alors que le résultat additif laisse des résidus. Cela se confirme en observant les spectres de Fourier sur l'illustration 6.12. On y observe l'impact sur les répliques fréquentielles des franges, qui disparaissent uniquement avec l'algorithme multiplicatif.

En revanche, dans les zones lisses, où les basses fréquences sont plus significatives, comme la deuxième et troisième ligne de la figure 6.11, les deux modèles de décomposition fournissent des résultats similaires.

Le tableau 6.2, qui récapitule les résultats de PSNR pour chaque modèle, confirme que l'utilisation d'un modèle multiplicatif améliore en général nettement le résultat de la séparation des franges.

	Image avec franges	Algorithme 1 Modèle Additif	Algorithme 3 Modèle multiplicatif
Image a	25.04	49.30	54.75
Image b	24.95	48.72	53.22
Image c	24.74	47.82	53.68
Image d	25.12	47.40	53.44
Image e	24.73	47.08	51.95
Image f	24.34	45.39	50.56
Image g	24.73	46.00	51.85
Image h	25.06	44.98	50.60
Image i	25.00	45.60	51.07
Image j	25.02	39.97	41.69

TABLE 6.2 – **Score de PSNR pour l'image panchromatique u résultante de la méthode additive et de la meilleure méthode multiplicative.** Les images correspondent à celles montrées en figure 3.2. Le PSNR est mesuré entre l'image panchromatique des images simulées et le résultat de l'algorithme. L'image simulée physique a été mesurée de telle sorte à approcher le modèle de formation d'image réel, avec les mêmes paramètres d'acquisition que Sieleters (voir section 3.1). Le modèle multiplicatif fournit des valeurs sensiblement meilleures que le modèle additif.

Limite de chaque modèle d'image

Cette analyse des résultats permet de conclure sur le cadre d'utilisation le plus approprié pour chacun des modèles d'image. De manière générale, le modèle multiplicatif assure une meilleure séparation des franges. Cela est clair pour une image texturée. Pour une image lisse, bien que les résultats quantitatifs ne soient pas forcément meilleurs, l'observation de l'image calculée ne révèle pas de différence qualitative.

Toutefois, l'application du modèle de décomposition multiplicatif fait l'hypothèse que tous les spectres de la scène ont une forme semblable. Dans le cas de Sieleters, qui opère dans l'infrarouge, l'hypothèse tient. Nous n'avons cependant pas fait de tests sur des images du domaine spectral visible, où les spectres ont des formes plus diverses.

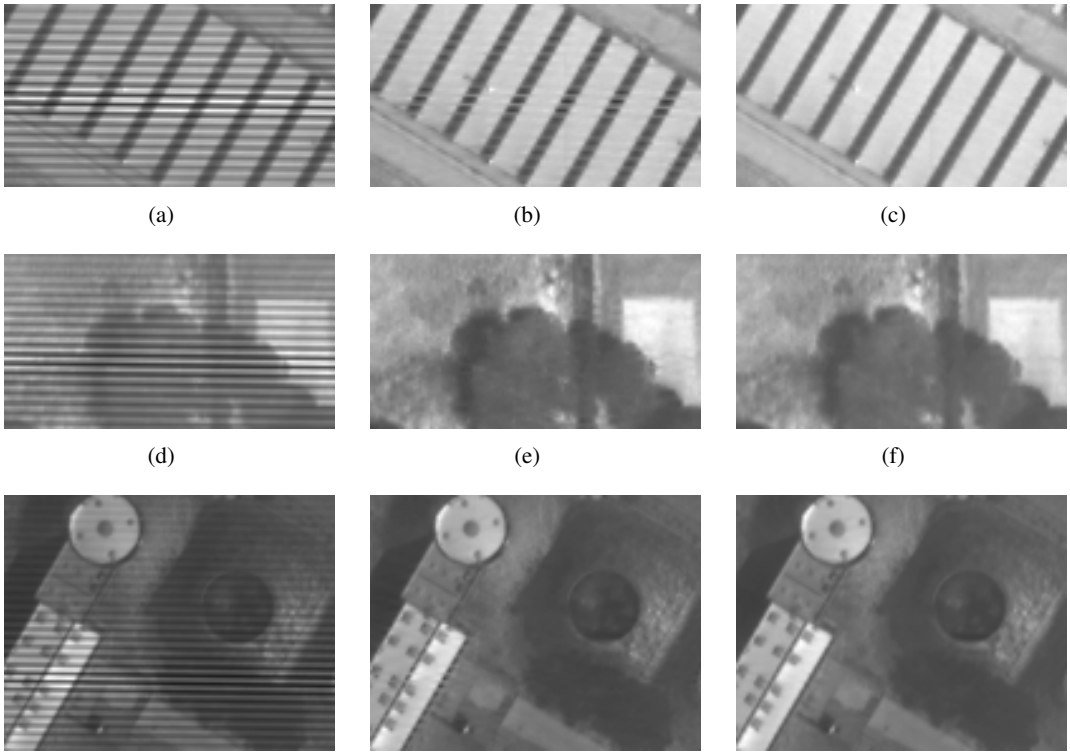
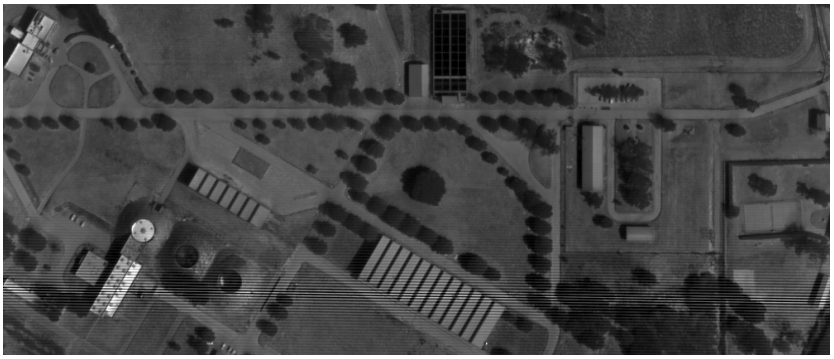
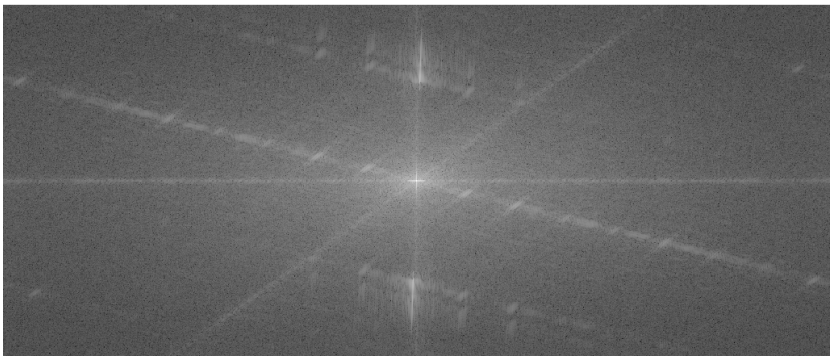


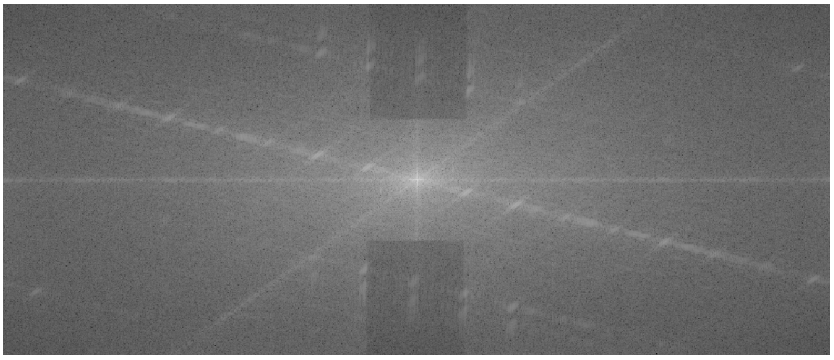
FIGURE 6.11 – **Détails de la scène en figure 6.12 avant et après traitement. -** Les imagettes de la première colonne sont issues de l'image d'origine (avec franges). Celles du milieu du résultat du modèle additif et celles de la troisième colonne du modèle multiplicatif. La première ligne montre les lacunes du modèle additif sur les motifs à haute fréquence. Les deux autres montrent des scènes plus lisses, où peu de différences sont observées entre les deux modèles d'image.



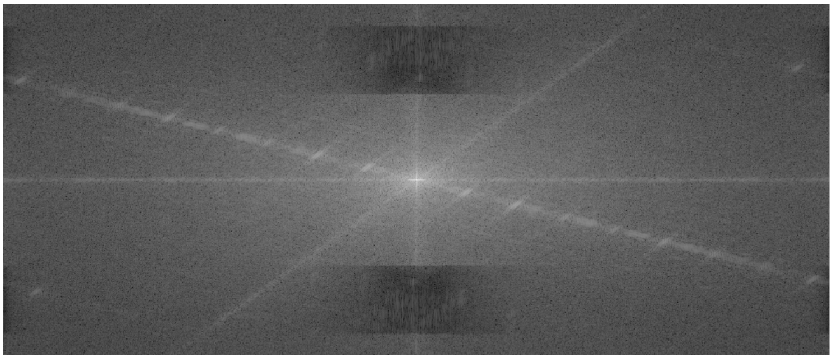
(a) Image avec franges



(b) Transformée de Fourier de l'image avec franges



(c) Transformée de Fourier du résultat additif



(d) Transformée de Fourier du résultat multiplicatif

FIGURE 6.12 – **Comparaison des résultats sur le spectre de Fourier. -** L'algorithme utilisé pour l'image (c) est celui par filtrage non-linéaire. Les fréquences des franges et leurs répliques sont enlevées par le modèle multiplicatif mais conservées par le modèle additif.

Troisième partie

Construction du modèle numérique d'élévation

CHAPITRE 7

ETUDE DE LA CONTRAINTE SUR LA LIGNE DE VISÉE

Sommaire

7.1 Les problèmes d'attitude et de trajectoire dans la littérature **91**
7.2 Influence des écarts de trajectoire sur les images des plans épipolaires **93**
7.3 Recalage des images à l'aide des données de navigation **94**
7.4 Quantification de l'erreur sur la disparité lorsque le porteur s'écarte de la ligne de visée **95**
7.5 Résultats pour les paramètres de vol Canjuers **99**
7.6 Conclusion **100**

La mise en oeuvre de la méthode stéréoscopique des images des plan épipolaires (EPI) nécessite que l'acquisition des images se fasse selon une trajectoire rectiligne et que l'orientation des clichés soit constante afin que les trajectoires des objets sur l'EPI (appelées chemins) soient des droites. Cependant, les conditions de prise de vue réelles entraînent des déviations du porteur par rapport à la trajectoire théorique et des variations de sa vitesse, qui n'est plus uniforme et engendre des bases successives différentes. Nous proposons ici d'étudier l'impact de ces perturbations sur la mesure de disparité.

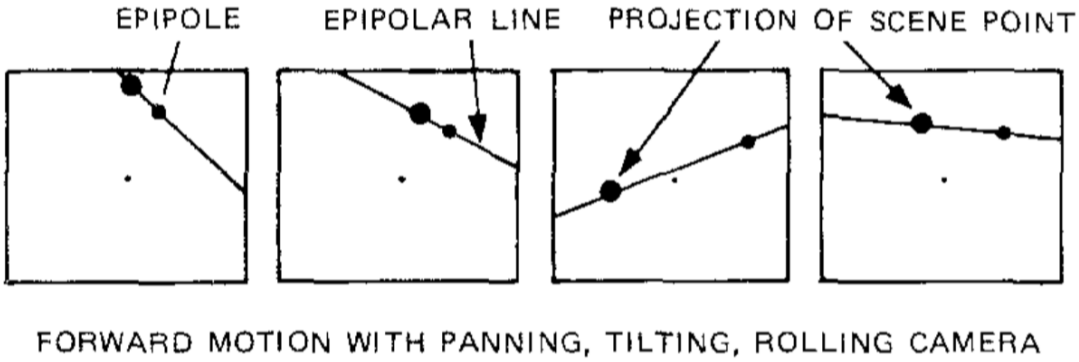
7.1 Les problèmes d'attitude et de trajectoire dans la littérature

Les problèmes d'attitude (c'est-à-dire de variation des angles de roulis, tangage et lacet, illustrés en figure 7.2) et de trajectoire dans la littérature sont multiples. Il s'agit souvent d'évaluer ces paramètres à partir des images afin de pouvoir les recaler. En imagerie satellitaire, on cherche à estimer les microvibrations des instruments de type *pushbroom* afin de mieux construire les images. [Roq+04] estime dans ce cadre les vibrations selon les trois angles d'attitude en analysant les variations de la disparité. [Per+10] répond au même problème en modélisant les variations par une fonction polynomiale par morceaux à l'aide de quelques hypothèses sur la trajectoire ou l'évolution temporelle des vibrations.

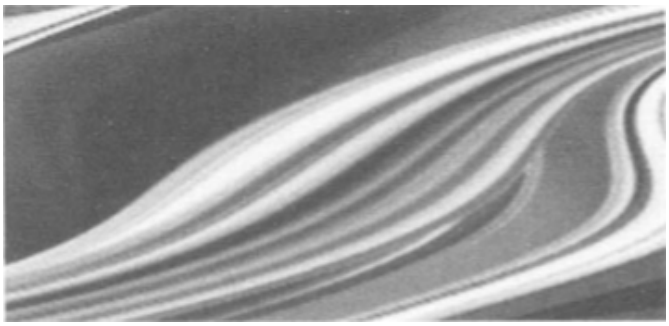
Que les données de navigation soient calculées à partir des clichés ou fournies par un récepteur GNSS (Global Navigation Satellite System) et une centrale inertielle, les images sont en général recalées par transformation homographique. [SAA12] corrige l'attitude et la translation selon la

base d'un assemblage de deux caméras en estimant la matrice de transformation entre la paire d'images mesurée.

[ZN96] propose une évaluation de l'influence des variations d'attitude sur la disparité. Toutefois, en imagerie aéroportée, les problèmes d'attitude sont souvent négligés, puisque les instruments sont en général montés sur une plateforme stabilisatrice. De plus, aucune hypothèse n'est nécessaire sur la ligne de visée dans le cas classique et prédominant de la stéréoscopie à deux vues. Les études sur l'impact des écarts d'attitude et de trajectoire sur la mesure de disparité ne sont donc pas nombreuses (contrairement à la problématique de l'incidence sur la disparité d'une altération de la calibration). La littérature sur les EPI n'évoque par ailleurs pas ce problème, puisque les acquisitions sont faites dans des environnements contrôlés. [BBM87] s'intéresse néanmoins à des configurations d'EPI différentes, en étudiant l'effet d'une rotation de la caméra au cours de l'acquisition. Cela nous donne une idée des effets sur les EPI lors d'une perturbation sur l'attitude. Dans ce cas, les EPI ne sont plus des coupes planes du cube spatio-temporel, mais des surfaces ondulées. En effet, les couples d'images successifs subissent chacun une transformation homographique. On a dans ce cas des chemins épipolaires qui ne sont pas des droites, comme le montre la figure 7.1.



(a)



(b)

FIGURE 7.1 – **(b) Image des plans épipolaires lorsque l'orientation de la caméra varie selon ses trois axes. -** Le schéma (a) montre, pour plusieurs images successives, la position d'une même ligne épipolaire. La variation d'orientation a pour résultat de modifier la position des épipoles, et donc des lignes épipolaires. Extraire les lignes épipolaires de chaque image pour construire l'EPI devient plus difficile. Le traitement de l'EPI est lui aussi compliqué. Source : [BBM87]

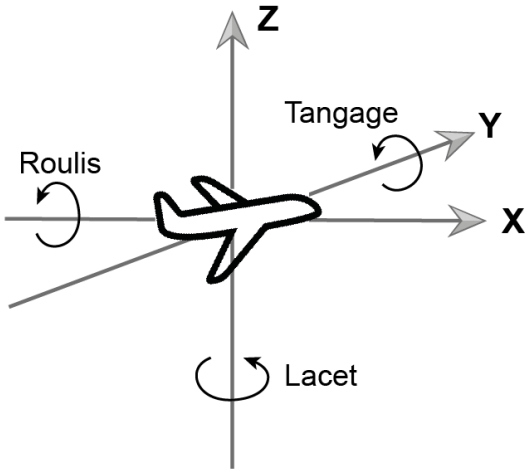


FIGURE 7.2 – **Angles de roulis, tangage et lacet.**

Puisque les modifications d'attitude peuvent être corrigées exactement par homographie avec la seule connaissance des écarts angulaires engagés (en particulier, sans nécessité de connaître la profondeur des objets de la scène), leur impact sur la disparité n'est pas étudié ici. En effet, ce type de correction est classique et très bien maîtrisée dans la littérature. [HZ04] y consacre la partie 0 pour la rectification d'images.

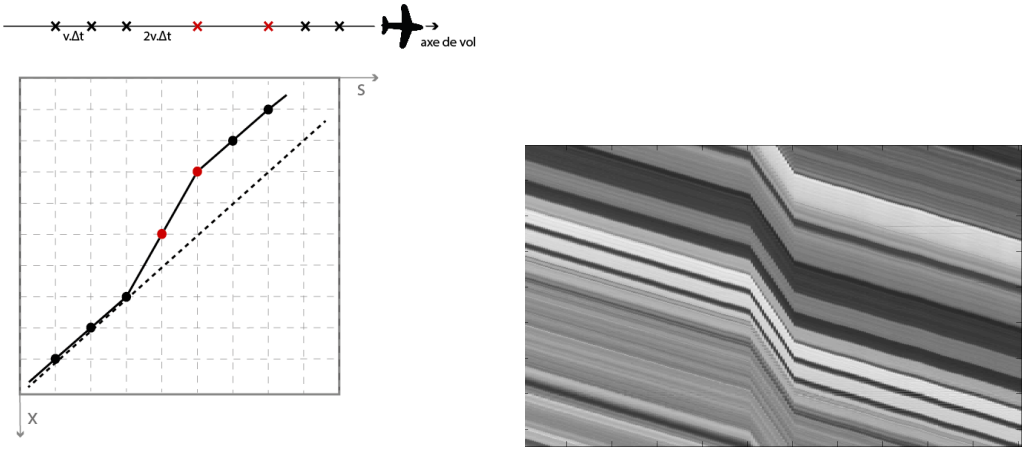
7.2 Influence des écarts de trajectoire sur les images des plans épipolaires

Afin de comprendre les effets de variations de trajectoire (donc les déviations par translation) sur l'image des plans épipolaires, chaque déviation est ici étudiée indépendamment des autres.

Comme la vitesse de l'avion n'est pas tout à fait constante sur la ligne de vol, les bases entre les images successives ne sont pas égales, et l'EPI n'est donc pas échantillonnée régulièrement selon son axe temporel. Il en résulte des variations de pente des chemins des EPI, comme l'illustre la figure 7.3.

Lorsque l'avion dévie de la trajectoire dans la direction perpendiculaire à l'axe de vol, les plans épipolaires des images ne sont plus alignés sur les mêmes colonnes de l'image. Alors, la conséquence sur l'EPI dépendra de la taille et de l'inclinaison de l'objet. Si celui-ci est trop petit, il disparaitra et le chemin épipolaire correspondant sera discontinu, voir figure 7.4. Si l'objet de la scène est assez large, l'EPI ne sera possiblement pas affectée, à condition que la forme et les caractéristiques radiométriques de l'objet soient homogènes. Enfin, il faut également que l'objet soit aussi perpendiculaire à l'axe de vol que possible. Comme le montre la figure 7.5, l'orientation de l'objet n'est pas anodine, puisque plus il est aligné sur l'axe de vol, plus l'erreur sur la disparité augmentera.

Une altération de la hauteur de vol correspond à une variation de l'échelle de l'image. Un tel mouvement entraîne un déplacement selon les deux axes de l'image. On retrouvera donc les effets décrits précédemment d'une translation selon l'axe transversal ainsi que les effets d'une translation dans la direction de l'axe de vol.



(a) (b)

FIGURE 7.3 – **Effet de la non uniformité de la vitesse de vol sur l'EPI -** (a) Chaque point correspond à un échantillon d'un chemin épipolaire. La ligne horizontale en haut de la figure montre les positions successives de l'avion le long de la ligne de vol, correspondant à chacune des colonnes de l'EPI. Sur les points rouges, la vitesse du porteur a doublé. L'échantillonnage de l'EPI ne rend toutefois pas compte de cette variation et reste constant. Dans ce cas, le chemin dévie de sa direction attendue (en pointillés) et sa pente augmente brusquement. Les estimations de disparité sont alors mauvaises. (b) Sur cette EPI, la partie centrale a été obtenue en n'utilisant qu'une image sur cinq, ce qui permet de simuler le cas d'une variation (exagérée) de la base. On voit bien que sur la portion centrale, la pente est différente du reste de l'EPI, ce qui correspond bien au cas schématisé en (a).

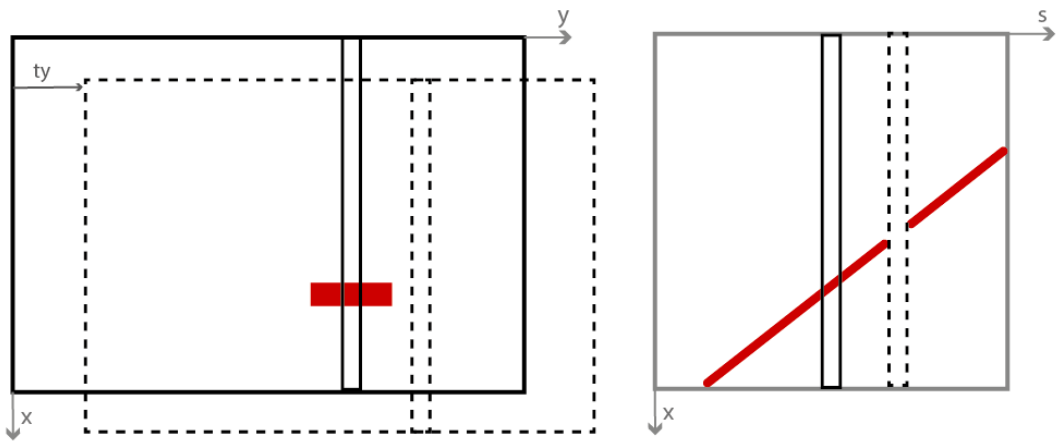


FIGURE 7.4 – **Effet d'une translation transversale. -** Le schéma de gauche montre deux images rectangulaires prises à des instants différents et ayant un décalage latéral t_y. La même colonne du détecteur est tracée sur chacune d'elles. Sur le dessin de droite, on retrouve ces même colonnes qui composent l'EPI résultante d'une séquence d'images dont font partie celles de gauche. L'objet rouge qu'observe la caméra n'apparaît pas sur la bonne colonne de l'image en pointillé en raison du décalage t_y. Sur l'EPI on observe que le chemin correspondant à cet objet est alors discontinu.

7.3 Recalage des images à l'aide des données de navigation

L'avion dispose d'un récepteur GNSS ainsi que d'une centrale inertielle afin d'enregistrer les données de navigation du vol. Elles permettent en post-traitement de connaître l'attitude et la

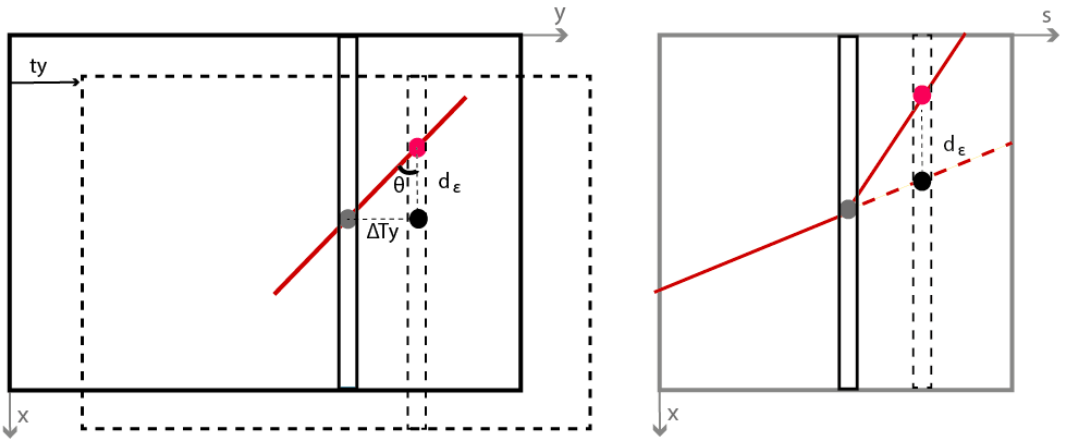


FIGURE 7.5 – **Influence de l'orientation de l'objet sur l'erreur de disparité. -** A nouveau, on observe à droite deux images successives et avec un décalage latéral t_y. A droite est dessinée l'EPI pour la ligne épipolaire correspondant aux colonnes surlignées. Si l'objet observé, en rouge, est incliné par rapport à l'axe de vol, alors l'erreur de disparité est accrue. En effet, le point gris, au lieu d'être associé à son correspondant de l'autre image qui devrait se trouver au point noir, est associé à l'autre point de même couleur, soit le point rouge. Le chemin de l'EPI sera continu, mais sa pente sera modifiée.

position de l'avion au moment de l'acquisition de chaque image. La précision en positionnement est de 10cm en planimétrie et 25cm en altimétrie. Celle en attitude est de $70\mu\text{rad}$.

Ces données sont utilisées avec un logiciel de reprojection d'image afin de corriger les déviations de la ligne de visée au cours du vol et d'aligner les images dans une géométrie épipolaire commune. L'algorithme corrige l'attitude pour se placer en visée nadirale et les écarts en translations par rapport à une trajectoire moyenne rectiligne en utilisant un MNE plat (car le MNE réel est inconnu).

La figure 7.6 montre la trajectoire réelle et corrigée dans le plan vertical et horizontal. Les histogrammes en 7.7 donnent les écarts selon chaque direction. Les valeurs sont particulièrement faibles, mais sont réalistes du fait que les conditions de vol étaient bonnes, que le pilote était expérimenté et habitué à l'avion, et qu'il s'agit de vols sur de faibles distances (un échantillon de 600m de la ligne de visée). Du fait que l'instrument est monté sur une plateforme stabilisatrice qui compense les erreurs d'orientation durant le vol, l'attitude présente de faibles variations. Les histogrammes des écarts pour chacun des angles sont donnés en figure 7.8. Comme vu précédemment, ces écarts se corrigent parfaitement par transformation homographique. Nous nous concentrons donc sur les déviations en translation du porteur. Or, ne connaissant pas l'élévation des objets de l'image au moment de la reprojection, le logiciel, en supposant que la scène est plate, fait une erreur.

7.4 Quantification de l'erreur sur la disparité lorsque le porteur s'écarte de la ligne de visée

Cette section propose de quantifier l'erreur faite sur la disparité lorsque la reprojection des images est effectuée sans connaissance a priori de l'élévation de la scène, ce qui est le cas des images

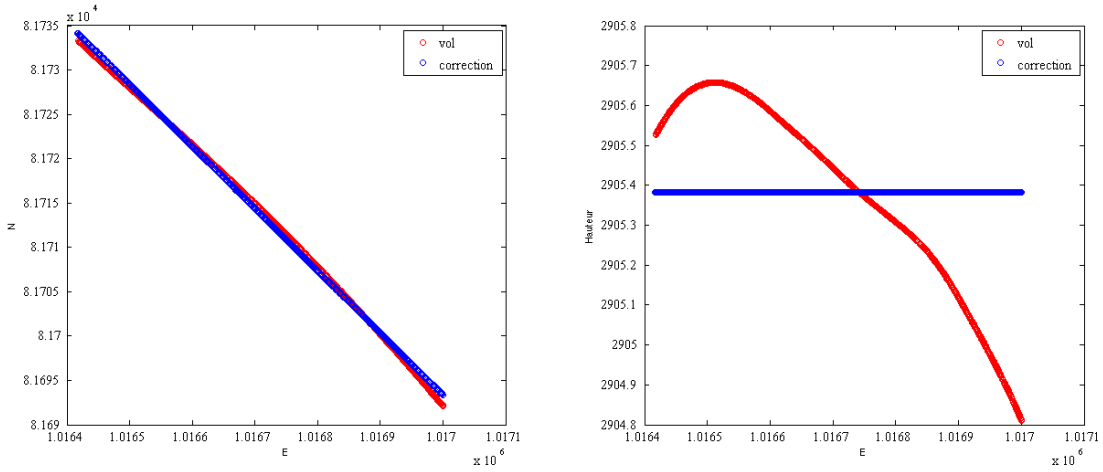


FIGURE 7.6 – **Trajectoire du vol canjuers. -** La ligne rouge correspond à la trajectoire réelle, tandis que la bleue correspond à la trajectoire attendue, qui sert de référence à la correction. Les trajectoires sont données en coordonnées planes, en mètres.

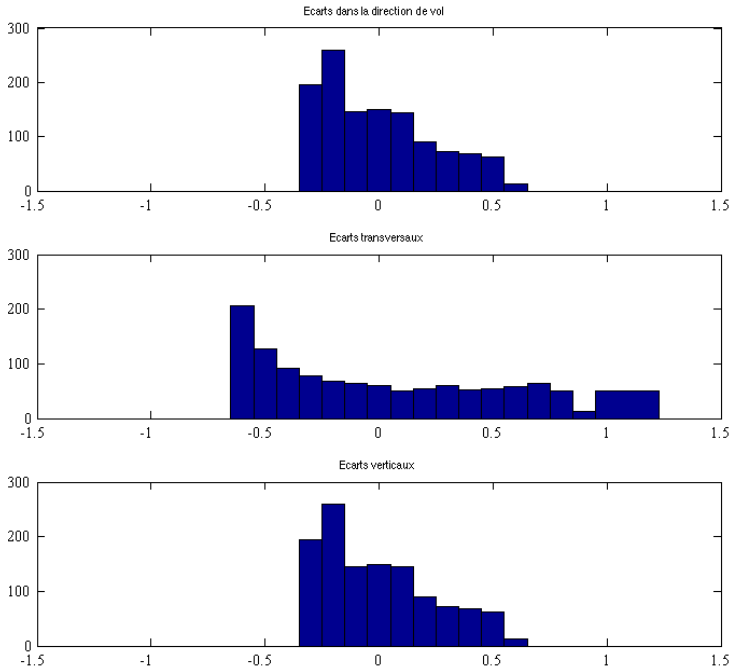


FIGURE 7.7 – **Histogrammes des écarts en translation de la trajectoire réelle par rapport à la trajectoire corrigée (en mètres).**

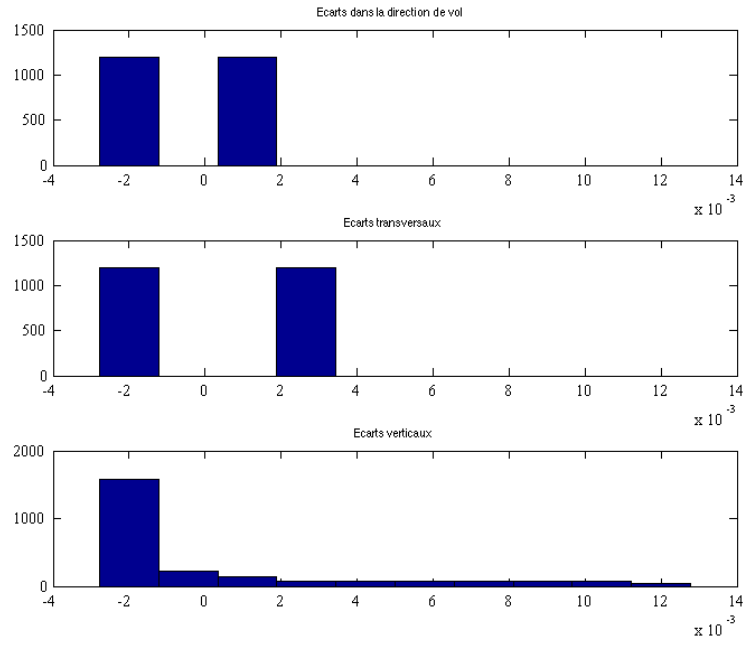


FIGURE 7.8 – **Ecarts en attitude par rapport à la valeur moyenne en vol (en radians).**

Sieleters. On utilise dans ce cas un MNE plat. La déviation selon chacun des axes est étudiée indépendamment des autres. A chaque fois, on étudie une déviation de la base $\delta B_\bullet$ de sa position théorique C. Le sommet de prise de vue résultant C' est la position "réelle" du porteur au moment de l'acquisition. La reprojection a pour but de replacer le centre de projection a sa position C. Comme l'élévation des points de la scène est inconnue, si celle-ci n'est pas plate, alors il y aura une erreur de reprojection δd. Comme le montre la figure 7.9 le point M, projeté en I sur l'image "réelle", se trouve reprojeté en I' au lieu de J, qui serait sa position si la hauteur h de M avait été connue.

On en déduit que pour une déviation de l'avion δB_x selon l'axe de vol, l'erreur de disparité après reprojection est :

$$\delta d_x(\delta B_x) = \frac{fh}{H^2}\delta B_x \tag{7.1}$$

Dans le cas d'un décalage transversal à la ligne de visée δB_y, la reprojection inexacte fait que l'objet (et donc le point J) est décalé le long de l'axe y, voir figure 7.10(a). Toutefois, la disparité se mesure le long de l'axe x. L'impact sur l'axe x dépend en particulier de l'orientation de l'objet, comme c'est illustré en 7.10(b). L'erreur sur la disparité est donc :

$$\delta d_y(\delta B_y) = \frac{fh}{H^2 \tan\theta}\delta B_y \tag{7.2}$$

Plus l'objet est aligné sur la direction de vol, plus l'erreur sur la disparité sera grande. En revanche, suivant des considérations purement géométriques, un objet horizontal n'entraîne pas d'erreur.

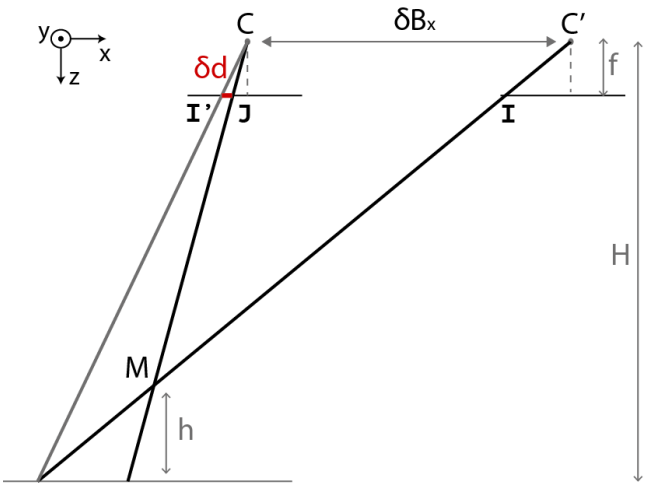
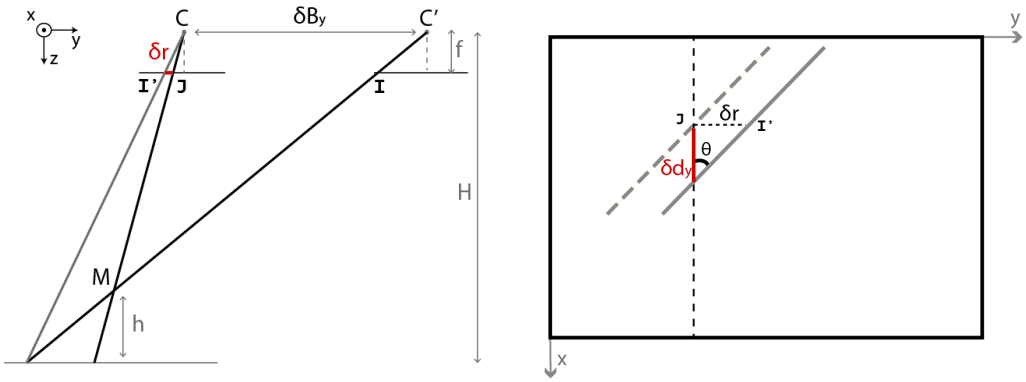
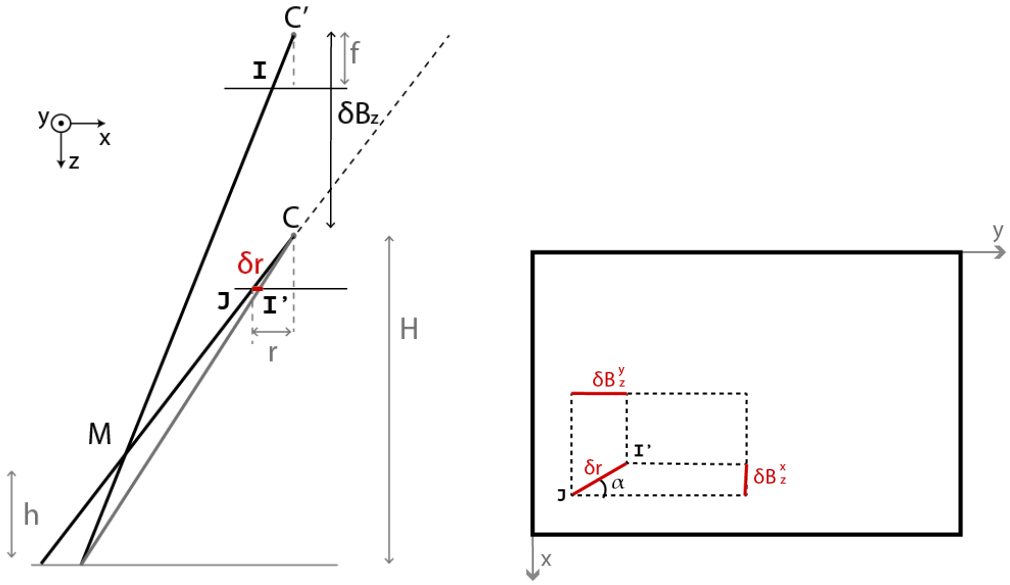


FIGURE 7.9 – **Schéma de calcul de l’erreur sur la disparité δd_x. -** La reprojection du point I pour corriger la déviation δB_x selon l’axe de vol devrait se faire en J. Comme on ne connaît pas l’élévation de la scène, elle se fait toutefois en I’.



(a) (b)

FIGURE 7.10 – **(a) Schéma de calcul de l’erreur sur la position de la reprojection δr d’un point M pour une déviation transversale δB_y. -** La reprojection du point I se fait en I’ au lieu de J. **(b) Schéma de la relation entre le décalage transversal de l’avion par rapport à son axe de visée δr et l’erreur sur la mesure de disparité δd_y. -** Le schéma représente l’image reprojetée. Le trait noir vertical en pointillé dessine une colonne quelconque (une ligne épipolaire). Le trait gris continu désigne la reprojection d’un objet surélevé. Le trait gris en pointillé correspond à la position correcte de l’objet. L’erreur faite sur la disparité est donc le segment rouge.



(a) (b)

FIGURE 7.11 – **(a) Erreur de reprojection pour une variation d'altitude. -** La reprojection du point I pour corriger la déviation δB_z devrait se faire en J. Comme on ne connaît pas l'élévation de la scène, elle se fait toutefois en I'. **(b) Décomposition de l'erreur due à l'altitude. -** Le schéma représente l'image reprojetée. L'erreur de disparité se décompose en une erreur selon la ligne de vol et une erreur transversale.

Pour une déviation à la hauteur de vol, l'erreur peut se décomposer en une erreur purement selon l'axe de vol et une erreur transversale. En utilisant les schémas de calcul de la figure 7.11, on trouve :

$$\vec{\delta r} = \delta B_z^x.\vec{x} + \delta B_z^y.\vec{y} \tag{7.3}$$

Où :

$$\begin{cases} \delta B_z^x &= \delta r \sin\alpha \\ \delta B_z^y &= \delta r \cos\alpha \end{cases} \tag{7.4}$$

Et :

$$\delta r = \frac{hr}{H(H+\delta B_z - h)}\delta B_z \tag{7.5}$$

Alors l'erreur de disparité à calculer est :

$$\delta d_z(\delta B_z) = \delta d_x(\delta B_z^x) + \delta d_y(\delta B_z^y) \tag{7.6}$$

7.5 Résultats pour les paramètres de vol Canjuers

En considérant les valeurs fournies à la section 7.3, la figure 7.12 donne l'évolution de l'erreur de déviation en fonction de plusieurs paramètres. Elles sont réalisées suivant les paramètres de Sieleters, c'est-à-dire une focale $f = 100\text{mm}$ et la hauteur de vol $H = 2072\text{m}$. Lorsque la hauteur du point M est fixé, il vaut $h = 20\text{m}$, ce qui correspond raisonnablement à un bâtiment à

6 étages. Lorque les déviations sont fixées, elles sont données par les maxima des écarts sur le vol de la ligne Canjuers : $dB_x = 0,79\mathrm{m}$, $dB_y = 1,23\mathrm{m}$, $dB_z = 0,57\mathrm{m}$.

Il en ressort d'abord que les mesures de disparité sont assez robustes aux déviations de l'altitude de vol. En effet, tant pour des hauteurs h de $100\mathrm{m}$ que pour des écarts de hauteur de vol de $5\mathrm{m}$, δd_z reste largement négligeable.

L'erreur dans la direction de vol n'est pas non plus trop critique, avec des valeurs de l'ordre de quelques centièmes de pixel. Elle peut toutefois ne plus être négligeable pour un relief très accidenté, ce qui n'est pas le cas de scènes survolées par Sieleters.

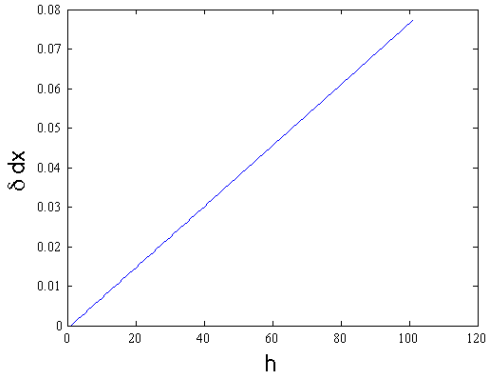
L'erreur transversale peut elle poser problème pour des objets avec un alignement proche de la direction de vol. L'erreur reste toutefois négligeable jusqu'à $40\mathrm{m}$ de dénivelée et pour des objets formant un angle de plus de $45°$ avec l'axe de vol.

7.6 Conclusion

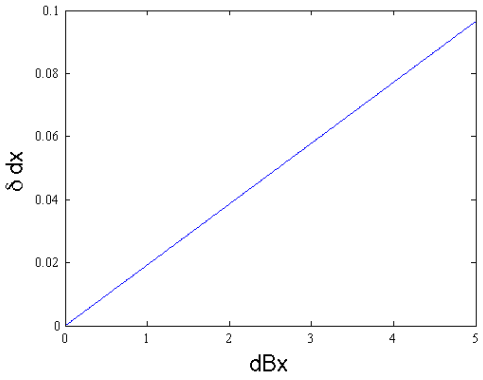
Les résultats de ce chapitre ont permis de vérifier que les conditions de vol de Sieleters permettent d'utiliser les images de plans épipolaires. En effet les déviations de la trajectoire n'engendrent en général que des erreurs inférieures au dixième de pixel, ce qui correspond à une erreur largement acceptable de $40\mathrm{cm}$ sur la hauteur. Une perte de précision plus notable pourra néanmoins être attendue sur les rares éléments dans un alignement proche de l'axe de vol.

Les calculs présentés peuvent également être utilisés dans la planification de prochains vols, et la formulation de contraintes pour le pilote.

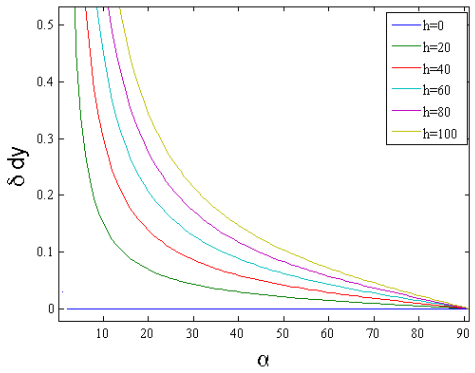
Pour des cas de très fortes dénivelées, des hauteurs de vol basses ou des trajectoires très instables, il sera nécessaire de corriger la géométrie des images. Il semble envisageable d'utiliser une méthode itérative pour calculer des MNE successifs et reprojeter les images avec, jusqu'à obtenir la précision requise. La déviation en altitude se corrige par un changement d'échelle, celle transversale par une translation. Le problème est légèrement plus complexe pour les corrections le long de l'axe de vol si l'on veut tenir compte de l'évolution temporelle. En effet, dans ce cas il y aura d'une part une translation de l'image, mais aussi une interpolation temporelle puisque l'on modifie aussi l'instant de prise de vue.



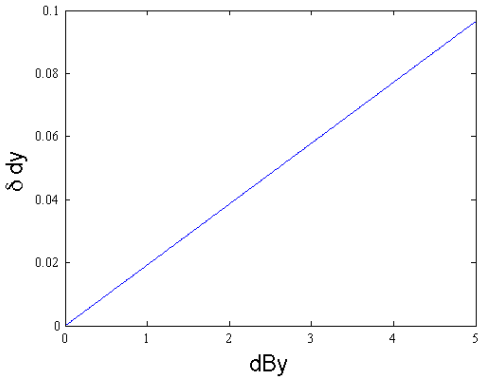
(a) $dB_x = 0,79\text{m}$



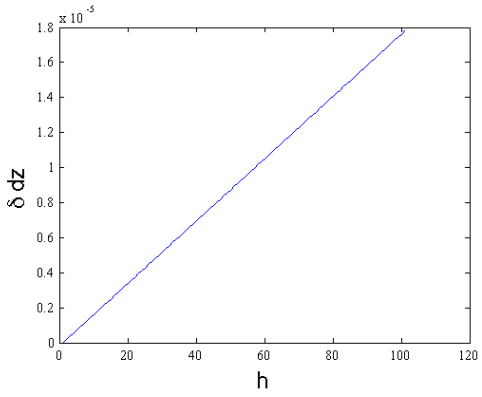
(b) $h = 20\text{m}$



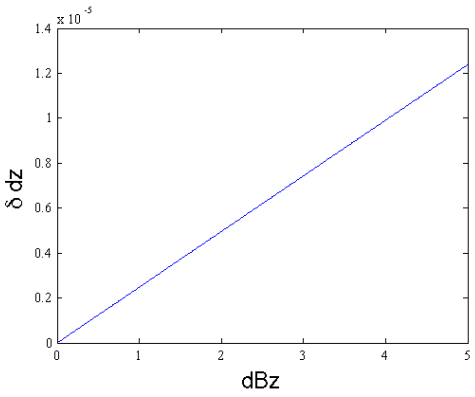
(c) $dB_y = 1,23\text{m}$



(d) $h = 20\text{m}, \theta = 45°$



(e) $dB_z = 0,57\text{m}$



(f) $h = 20\text{m}, \theta = 45°$

FIGURE 7.12 – **Evolution de l'erreur de disparité. -** Chaque ligne correspond à la déviation de l'avion selon une direction en particulier, respectivement l'axe de direction de vol X, l'axe transverse Y et l'axe vertical Z. Pour les valeurs maximales de déviation données en 7.7, les courbes (a) et (e) dépendent des variations de la hauteur du point au sol h et la courbe (c) des variations de l'orientation θ pour plusieurs valeurs de h. Les courbes de la deuxième colonnes sont calculées avec $h = 20\text{m}$, $\theta = 45°$. C'est la valeur de la déviation de la base qui varie.

CHAPITRE 8

CALCUL D'UNE CARTE DE DISPARITÉ ÉPARSE À PARTIR DES IMAGES DES PLANS ÉPIPOLAIRES

Sommaire

8.1 Construction et exploitation des EPI **103**
8.1.1 EPI des images Sieleters 103
8.1.2 Extraction des segments 105
8.2 Résultats **107**
8.3 Conclusion **109**

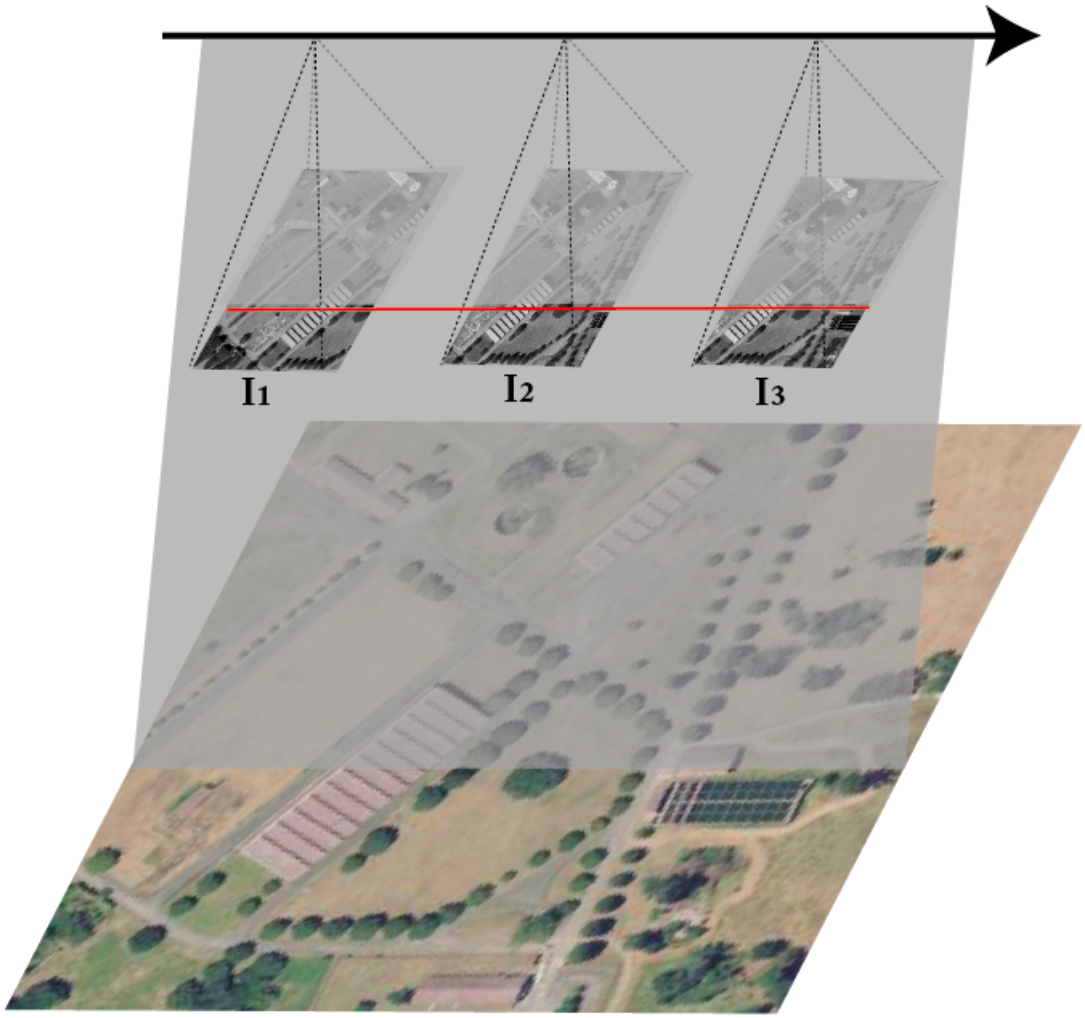
L'obtention du modèle numérique d'élévation se fait en deux étapes. D'abord par l'obtention d'une carte éparse en utilisant les images des plans épipolaires, dont le principe a déjà été détaillé au chapitre 2. Puis par une densification du résultat obtenu, qui fera l'objet du dernier chapitre. Ce chapitre détaille la première étape, appliquée aux images de Sieleters.

8.1 Construction et exploitation des EPI

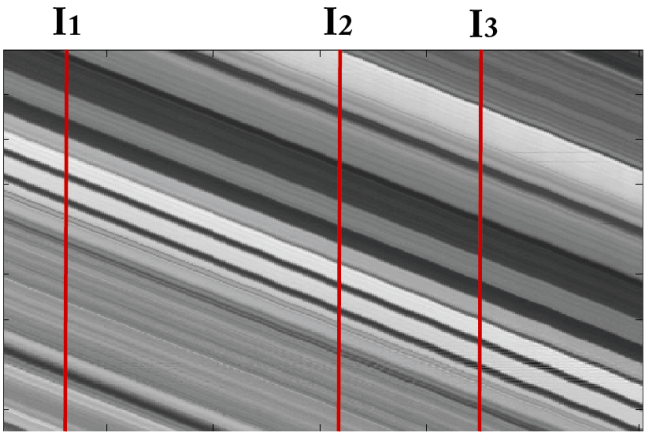
8.1.1 EPI des images Sieleters

Une fois que toutes les images ont été reprojetées pour corriger les déviations de la trajectoire de l'avion (étudiées au chapitre précédent), les images des plans épipolaires sont construites en rassemblant sur une nouvelle image les lignes épipolaires correspondantes. Comme illustré en figure 8.1, pour Sieleters, les lignes épipolaires sont les colonnes des images. L'image épipolaire est donc la juxtaposition de la même colonne d'un ensemble d'images. Pour décrire toute la scène, il faut donc une EPI par colonne de l'image centrale.

Comme expliqué au chapitre 2, chaque droite visible sur l'EPI correspond à la trajectoire d'un objet sur le détecteur. La pente de ces droites, appelées aussi chemins, renseignent sur la profondeur de l'élément dans la scène. La particularité des EPI de Sieleters par rapport à ce qui se trouve dans la littérature, est que tous les chemins paraissent parallèles. Cela s'explique par les hauteurs relatives des objets de la scène qui sont bien inférieures à la hauteur de vol et les faible dispa-



(a)



(b)

FIGURE 8.1 – **Construction des EPI avec les images Sieleters. -** La ligne rouge en (a) correspond à une ligne épipolaire, à partir de laquelle est construite l'EPI en (b). En gris, le plan épipolaire associé. La flèche indique la direction de vol. Sur l'EPI, nous appelons les lignes que l'on observe des chemins.

rités observables sur les images. Il est donc nécessaire d'utiliser une méthode sous-pixellique de détermination des pentes.

8.1.2 Extraction des segments

Le parti pris pour déterminer les pentes des chemins des EPI est d'utiliser un algorithme de détection d'alignements existant. Il s'agit du *Line Segment Detector* (LSD) de [Gio+12], une méthode d'extraction de segments par détermination a contrario. Cette section donne une brève explication des méthodes a contrario d'après l'ouvrage de référence [DMM08], et de la détection de segments en particulier.

Détection a contrario

Les méthodes de détection a contrario se basent sur la théorie de la gestalt, qui vise à formuler de manière scientifique les lois de la perception visuelle. Le terme gestalt désigne un ensemble de points dont la structure a un sens. C'est le cas par exemple des points figure 8.2, que l'on identifie volontiers à un cercle. L'un des principe fondamentaux de cette théorie est le principe de Helmholtz, qui définit un gestalt comme un événement rare dans une image aléatoire. On peut traduire ce principe mathématiquement par un indicateur : le nombres de fausses détections (NFA : number of false alarms), qui correspond à l'espérance du nombre d'événements dus au hasard. Plus le NFA est faible, moins la structure correspond à un agencement aléatoire. On parle alors d'événement significatif.

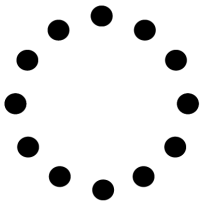


FIGURE 8.2 – **Gestalt. -** Les points sont agencés de telle sorte qu'ils forment un cercle. Un telle structure aurait peu de chances de se produire dans une image de bruit. Il s'agit donc d'un gestalt.

Le choix de structure du bruit n'est pas anodin : il faut qu'il représente des variables aléatoires indépendantes identiquement distribuées pour la grandeur considérée. Par exemple, [Des+02] montre que si la variable est l'orientation des gradients, une image de bruit blanc gaussien est préférable car elle n'introduit pas de biais dans l'orientation des gradients. D'autres modèles de bruit auront tendance à générer un biais dans les valeurs d'orientation du gradient lors de la quantification.

Alignements significatifs

La détection d'alignements recherche des suites de pixels dont les gradients sont orientés dans la même direction. Mathématiquement, pour un segment S de longueur l, cela revient à définir

des variable X_i indépendantes et identiquement distribuées valant 1 si le pixel i est aligné sur la direction du segment à une précision $p = \frac{1}{n}$, $n \in \mathbb{N}$, tel que :

$$S_l = X_1 + X_2 + ... + X_l \tag{8.1}$$

Alors la probabilité que le segment contienne k_0 points alignés suit une loi binomiale :

$$\mathbb{P}[S_l = k_0] = \binom{l}{k_0} p^k (1-p)^{l-k_0} \tag{8.2}$$

Le nombre de fausses détections se définit alors de la manière suivante :

$$NFA(l, k_0, p) = N_{test}.\mathbb{P}[S_l \geq k_0] \tag{8.3}$$

où N_{test} désigne le nombre de segments possibles à une précision p donnée. Pour des segments à une dimension (sans épaisseur) dans une image carrée de taille N, N_{test} englobe tous les alignements possibles ayant pour extrémités deux pixels quelconques et différents de l'image. Alors :

$$N_{test} = N^2(N^2 - 1) \simeq N^4 \tag{8.4}$$

Le nombre de fausses détections pour un segment de longueur l avec k_0 point alignés à une précision p vaut alors :

$$NFA(l, k_0, p) = N^4.B(l, k_0, p) \tag{8.5}$$

Un alignement est donc dit significatif en un seuil ϵ si :

$$NFA(l, k, p) \leq \epsilon \tag{8.6}$$

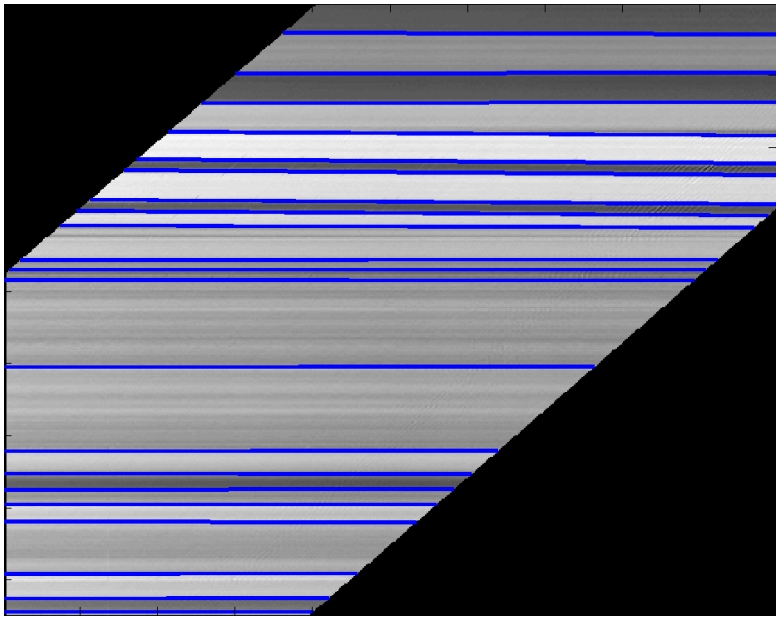


FIGURE 8.3 – **Extraction de segments avec l'algorithme LSD. -** Les segments extraits sont en bleu. La direction des chemins de l'EPI est différente de celle de la figure 8.1, pour laquelle le déplacement de l'avion n'a pas été compensé. L'utilisation de l'un ou l'autre est indifférente.

Le code utilisé est celui publié avec l'article [Gio+12]. Le calcul des segments est sous-pixellique. L'utilisation qui en est faite pour les EPI impose d'ajouter une étape de sélection pour les seuls segments qui traversent intégralement l'EPI. En effet, en raison de la faible largeur des images, les occlusions et dévers sont négligeables. La figure 8.3 montre le résultat de l'extraction de segments, surlignés en bleu. On remarque que la direction des chemins de l'EPI est différente à celle de la figure 8.1. Cela s'explique dans le recalage choisi : celle-ci est issue d'images formant un cube interférométrique : elle sont superposables. Les autres sont seulement corrigées des déviations de trajectoire. L'utilisation de l'une ou l'autre est indifférente.

Une fois l'extraction des segments réussie, on peut connaître la disparité des bords des objets de la scène. On obtient une carte de disparité discontinue.

8.2 Résultats

La carte éparse obtenue sur la ligne de vol CFM est en figure 8.4. Les valeurs de disparité y sont en couleur, et évaluées en mètres. Le résultat est assez dense, mais les bords verticaux n'ont pas de valeurs. Cela est lié uniquement à leur géométrie, car les EPI, qui suivent les colonnes, ne voient que les bords inclinés. L'observation des images de plans épipolaires pour les colonnes de part et d'autre d'un tel bord en figure 8.6 illustre bien ce phénomène. On apprécie également sur les contours des arbres une nette différence d'élévation entre l'arbre et son ombre. L'utilisation d'images infrarouges ne permet pas en revanche de distinguer la frontière entre les deux, ce qui explique les formes ovales observées.

La scène Canjuers, en figure 8.5, contient en plus des bâtiments une forte dénivelée du terrain, que l'on retrouve bien sur la carte d'élévation. Cette carte apporte en plus l'exemple d'un objet mobile pendant l'acquisition de la séquence (en haut à gauche de l'image). On le distingue par les mesures de hauteur très hautes qu'il cause.

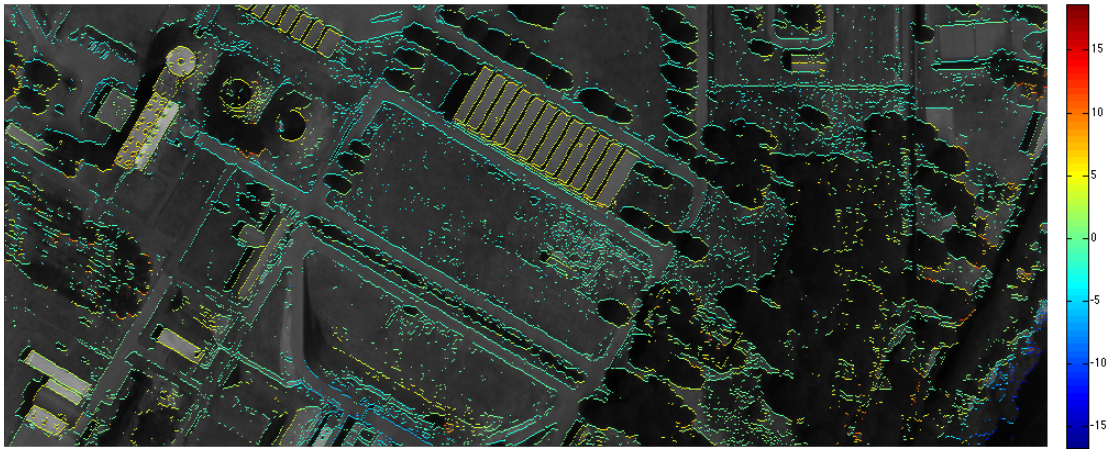


FIGURE 8.4 – **Carte éparse obtenue par EPI pour la scène CFM. -** La carte, en couleur, désigne les hauteurs relatives en mètres. Elle est superposée à l'image Sieleters correspondante (en gris) pour faciliter l'interprétation.

Le tableau 8.1 compare les valeurs obtenues sur une vérité terrain à celles calculées par les EPI. Les 5 premiers points de contrôle sont comparés au nuage de point lidar terrestre (voir chapitre

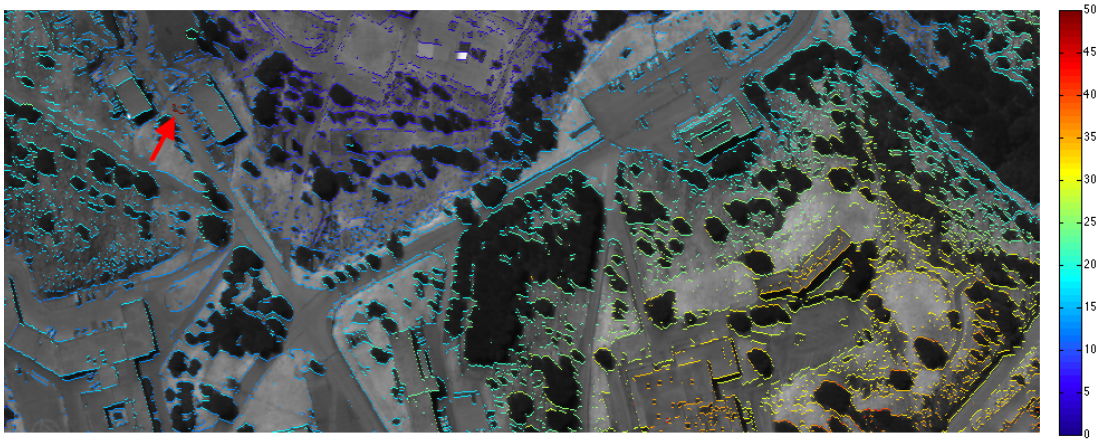


FIGURE 8.5 – **Carte éparse obtenue par EPI pour la scène Canjuers. -** La carte, en couleur, désigne les hauteurs relatives en mètres. Elle est superposée à l'image Sieleters correspondante (en gris) pour faciliter l'interprétation. En haut à gauche en rouge foncé, désigné par une flèche, les valeurs de hauteurs anormalement élevées sont causées par un véhicule qui se déplace au cours de l'acquisition.

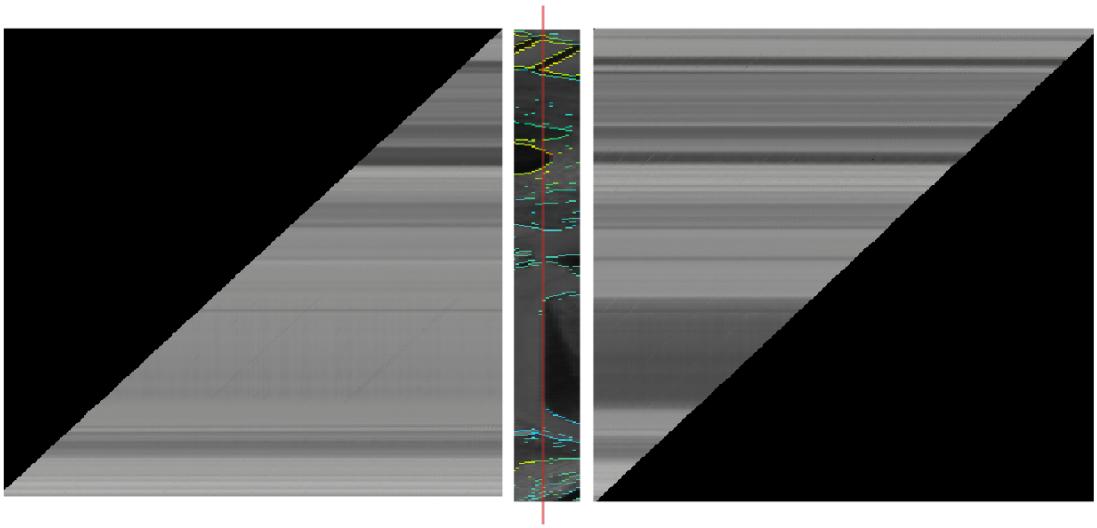


FIGURE 8.6 – **Illustration du manque d'information sur les bords verticaux. -** L'élément central est un extrait de la carte éparse en 8.4, entourée de la moitié de l'EPI des colonnes directement à gauche et à droite du trait rouge. La carte éparse est élargie afin de la rendre identifiable.

3). Les mesures sont faites en pointant des éléments du nuage de point, elles sont limitées par la précision du pointé. Il en ressort que sur les bâtiments (points P1 à P4), la méthode est exacte à $20\mathrm{cm}$ près, soit inférieure au dixième de pixel.

Les mesures sur la végétation, comme le point de contrôle P5, sont elles toujours plus approximatives, du fait que la hauteur n'est pas homogène sur l'objet, que les EPI ne donnent que les contours, et que le pointé sur le nuage de points est moins précis. On enregistre un écart de $2,4\mathrm{m}$, soit un demi pixel.

Le point de contrôle P7 a pu être mesuré sur la scène Canjuers, à partir d'une mesure par télémètre effectuée sur le terrain. La mesure sur ce point est aussi très bonne à $10\mathrm{cm}$ près.

Enfin, les écarts-type ont été mesurés sur des zones planes. Ils sont de l'ordre de $0,2$ pixels, ce qui nous permet de dire que la méthode garantit une précision de $1\mathrm{m}$ sur les images Sieleters.

	Objet	Verité terrain (m)	EPI (m)
P1	Bâtiment	8,7	8,8
P2	Bâtiment	6,5	6,6
P3	Bâtiment	6,0	6,1
P4	Bâtiment	8,1	8,3
P5	Arbre	6,6	4,2
P6	Bâtiment	11,2	11,1

TABLE 8.1 – **Points de contrôle. -** La position des points de contrôle est indiquée dans le chapitre 3.

8.3 Conclusion

A partir des images d'un spectro-imageur aéroporté, nous avons contruit des images des plans épipolaires avec de très faibles variations de pentes. Nous avons pu mesurer celles-ci avec un algorihtme de détection d'alignements existant. Nous avons ainsi obtenu des cartes de disparité sous-pixelliques, à $0,2$ pixel près. La carte obtenue reste toutefois éparse, et il reste encore à compléter les valeurs inconnues.

CHAPITRE 9

DENSIFICATION DE LA CARTE ÉPARSE PAR COUPURE DE GRAPHE

Sommaire

9.1 Généralités sur la minimisation par coupure de graphe **111**

9.2 Adaptation d'un algorithme de coupure de graphe **113**

9.2.1 Algorithme utilisé 113

9.2.2 Adaptation aux multi-vues 115

9.3 Résultats **115**

9.4 Conclusion **117**

Ce chapitre porte sur la densification de la carte éparse obtenue par EPI. Le choix fait ici est d'adapter des algorithmes de coupure de graphe aux images Sieleters pour minimiser une énergie tenant compte des valeurs déjà déterminées.

9.1 Généralités sur la minimisation par coupure de graphe

Un graphe non orienté G = (E,V) est un ensemble de sommets (ou noeuds) (*vertices*) V et d'arêtes (*edges*) E. On distingue deux sommets spéciaux : la source et le puits (*sink*). Aux arêtes sont associés des coûts qui rendent compte des propriétés de chaque sommet et de leurs relations. Une fonctionnelle peut donc être modélisée par un graphe, en définissant des noeuds adaptés à l'objet auquel elle s'applique. Les coûts sont alors les différents termes de la fonctionelle.

Une coupure de graphe est un partitionnement de V en deux parties, l'une contenant la source et l'autre le puits. Chaque noeud prend alors une valeur différente suivant qu'il est dans la partie contenant la source ou celle contenant le puits. On parle aussi d'étiquettage f, puisque l'on attribue à chaque noeud l'étiquette associée à la source ou au puits. Le coût d'une partition est la somme des coûts des arêtes entre un noeud de la source et un noeud du puits. Le résultat représente la solution optimale de la minimisation de la fonctionnelle associée au graphcut. La figure 9.1 illustre un graphe et une coupure.

Pour le traitement d'image, les noeuds sont en général des pixels ou des correspondances entre pixels. Ce sont souvent des problèmes avec plusieurs étiquettes, mais les algorithmes réalisent des

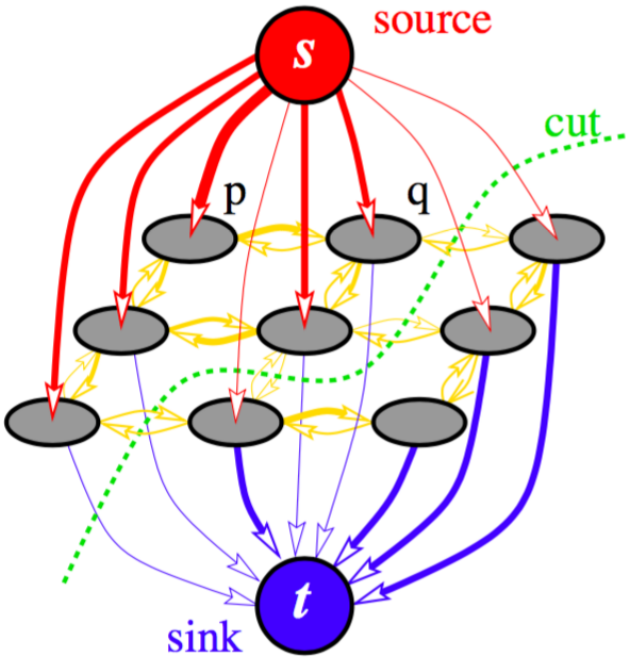


FIGURE 9.1 – **Graphe. -** Les liens entre noeuds sont à la fois arêtes jaunes gérant les relations de voisinage, et les liens rouges et bleus correspondants à des propriétés des noeuds. Le coût de la coupure de graphe en vert est la somme des liens qu'elle brise. p et q sont deux noeuds quelconques différents de la source et du puits. Source : [BV06]

optimisations intermédiaires (les *expansion-moves*) faisant intervenir seulement des étiquettages binaires.

Les liens entre sommets voisins ont des coûts qui assurent la cohérence spatiale de la solution. Le coût de régularisation imposé à des noeuds voisins p et q s'écrit donc V_{f_p,f_q}. Les liens reliés à la source ou au puits correspondent au coût d'attribuer une certaine valeur au noeud. Ils ne dépendent que du sommet considéré, et appliquent une attache aux données $D(f_p)$. La fonctionnelle à minimiser a alors la forme suivante, où $\mathcal{P}$ est l'ensemble des pixels de l'image et N l'ensemble des paires de pixels voisins :

$$E(f) = \sum_{p \in \mathcal{P}} D(f_p) + \sum_{(p,q) \in N} V_{f_p,f_q} \tag{9.1}$$

L'intérêt du graphcut est qu'il permet de minimiser de façon approchée une fonctionnelle même avec un terme de régularisation non-convexe. En effet, il fournit un minimum global exact aux problèmes d'étiquettage binaire avec des fonctions dépendant d'au plus deux variables, à condition que les termes de l'énergie qui dépendent de deux variables soient sous-modulaires :

$$V_{pq}(0,0) + V_{pq}(1,1) \leq V_{pq}(0,1) + V_{pq}(1,0) \tag{9.2}$$

Toutefois, en général, le nombre d'étiquettes à considérer pour résoudre un problème est supérieur à deux. En conséquence, un algorithme communément adopté est de procéder de manière itérative avec à chaque fois un étiquettage binaire issu de l'intervalle d'étiquettes possibles. Chaque

itération réalise un déplacement (*move*), c'est-à-dire une seule modification de chaque noeud. L'α-*expansion move* cherche à minimiser la solution initiale en attribuant l'étiquette de valeur α à certains noeuds, [YBVRZ01]. Les noeuds qui ne prennent pas la valeur α conservent l'étiquette qu'ils avaient avant.

La solution finale est donc approchée : il s'agit d'un optimum global d'un problème abstrait, approché du modèle réel. L'espace solution est néanmoins très proche de la solution réelle, ce qui fait le succès du graphcut. C'est le cas particulièrement pour le modèle de Potts, qui prend pour terme de régularisation :

$$V(f_p, f_q) = \lambda \sum_{p,q \in \mathcal{P}^2, p \sim q} 1(f_p \neq f_q) \tag{9.3}$$

où l'on note $1(\bullet)$ la fonction qui vaut 1 si son argument est vrai, 0 sinon. Pour deux pixels voisins ($p \sim q$) dans l'ensemble des pixels de l'image $\mathcal{P}$, le fait que les étiquettes associées à chacun des pixels soient différentes engendre une pénalité de coût λ fixe. Si les deux pixels ont la même étiquette, alors il n'y a pas de pénalité et $V(p, q) = 0$.

L'ensemble de ces notions sont détaillées dans [KZ02b] ou [BV06]. Il existe également d'autres formulation du problème par graphcut, en particulier avec des étiquettages multiples, comme [Ish03].

9.2 Adaptation d'un algorithme de coupure de graphe

9.2.1 Algorithme utilisé

L'algorithme utilisé est celui décrit dans [KMT14], qui met à disposition le code. Il s'agit d'un algorithme de stéréoscopie à deux vues, pour des images en géométrie épipolaire. Les noeuds correspondent à des assignements $a = (p, q)$, c'est-à-dire des pixels correspondants entre deux images I et I'. Alors la disparité associée à un assignement est : $d(a) = (q - p)$. On crée un noeud pour chaque couple de pixels d'un plan épipolaire. La recherche de pixels correspondants se fait donc seulement selon les abscisses, car les images sont toutes dans une géométrie épipolaire commune. Un même pixel est représenté dans un nombre de sommets égal au nombre de disparités possibles. Si un noeud est actif dans une configuration f, alors les pixels associés se voient attribuer la disparité $d(a)$ (ou $-d(a)$ suivant qu'il s'agit du pixel de l'image de gauche ou de droite). L'activation d'un assignement se note $f(a) = 1$. Si toutes les assignations associées à un pixel sont inactives, celui-ci n'a pas de correspondant. Alors c'est un pixel occulté.

L'énergie associée à ce problème pour un étiquettage f est la suivante :

$$E(f) = E_{données}(f) + E_{occ}(f) + E_{reg}(f) + E_{unicité}(f) \tag{9.4}$$

Le terme de fidélité s'assure que toutes les assignations actives de la configuration f soient le plus proche possible radiométriquement. Il n'y a donc pas de pénalité si les deux pixels ont la même couleur.

$$E_{données}(f) = \sum_a D(a).1(f(a) = 1) \tag{9.5}$$

où

$$D(a) = D(p, q) = (I(p) - I'(q))^2 \tag{9.6}$$

Le terme d'occultation cherche à minimiser le nombre de pixels occultés en comptant le nombre d'assignations inactives. Il multiplie alors ce nombre par un coefficient de pénalité K. Plus celui-ci sera élevé, plus le coût d'une occultation sera grand, ce qui aura un effet dissuasif.

$$E_{occ}(f) = \sum_{a, f(a)=0} K \tag{9.7}$$

Afin de respecter l'unicité de la solution, c'est-à-dire que pour tous les pixels il y ait au plus une assignation active, le terme d'unicité vaut l'infini si plus deux assignments ou plus sont actifs pour un même pixel.

$$E_{unicité}(f) = \sum_{\substack{a_1 = (p, q_1) \\ a_2 = (p, q_2) \\ q_1 \neq q_2}} \iota(f(a_1) = f(a_2) = 1) + \sum_{\substack{a_1 = (p_1, q) \\ a_2 = (p_2, q) \\ p_1 \neq p_2}} \iota(f(a_1) = f(a_2) = 1) \tag{9.8}$$

$\iota()$ désigne la fonction indicatrice, qui vaut 1 si son argument est vrai, 0 sinon.

Le terme de régularisation est celui du modèle de Potts, en équation (9.3), qui favorise les cartes constantes par morceaux. Il privilégie les disparités égales pour des pixels voisins. Deux pixels voisins de disparités différentes ont donc une pénalité nominale (ie. qui ne dépend pas de leur écart). Le fait qu'elle soit plafonnée autorise des discontinuités et évite des cartes lisses, car alors, il est plus rentable pour l'algorithme de payer la pénalité de régularisation que le coût d'une attache aux données mal appariée. On dit que des assignements a_1 et a_2 sont voisins si p_1 et p_2 sont adjacents et $d(a_1) = d(a_2)$.

$$E_{smoothness} = \sum_{a_1 \sim a_2} \lambda.1(f(a_1) \neq f(a_2)) \tag{9.9}$$

λ prend en réalité deux valeurs en fonction de la différence de radiométrie entre les pixels p_1 et q_1, et entre les pixels p_2 et q_2. Ainsi, une différence de disparité pour des pixels de couleur très différente est moins pénalisée, car un fort gradient dans l'image a des chances de délimiter deux objets différents, donc à des profondeurs différentes.

L'algorithme procède alors par α-*expansion move* successifs (tel que α soit une valeur de disparité dans un intervalle choisi) pour passer d'un configuration initiale f à un nouvel étiquettage f' :

- Si un assignement est actif et sa disparité est égale à α, l'assignement est actif dans la nouvelle configuration f'. Sa disparité sera inchangée.
- Si un assignement est inactif et sa disparité ne vaut pas α, l'assignement est inactif dans f'. Au cours d'un mouvement, un pixel ne peut prendre une valeur de disparité autre que α.
- Les autres assignements (ceux de disparité α inactifs) peuvent changer d'état : ceux de disparité α inactifs peuvent devenir actifs, et donc le pixel aura pour disparité α. Alors l'assignement qui était actif et de disparité différente de α deviendra inactif dans f'.

La version de l'algorithme utilisée est en réalité celle décrite par [Tan16], qui implémente une solution afin de pouvoir fixer les valeurs de disparité de pixels provenant d'une carte de disparité éparse. En conséquence, à l'énergie en 9.4 s'ajoute un terme pour la carte de disparité éparse :

$$E_{eparse} = \sum_{a \in \mathcal{F}} \iota(d_{fixées}(a) \neq d(a)) \tag{9.10}$$

Ce terme impose un coût infini à la modification des disparités fixées. Les assignements associés ne sont de fait plus modifiables. On note $\mathcal{F}$ l'ensemble de ces assignements. Le terme de régularisation devient plus compliqué et a été remanié. Le détail de son implémentation est fourni en annexe A.

9.2.2 Adaptation aux multi-vues

En multi-vues, on ne considère plus une paire d'images mais une multitude d'images N. La disparité est alors calculée pour l'image centrale I_0. Les assignements ne contiennent plus des paires de pixels, mais des ensembles de points correspondants définis par un pixel de l'image centrale et sa disparité $a = (x_0, d(x_0))$ tels que :

$$\forall k \in \{1..N\}, I_0(x_0) = I_k(x_0 + kd(x_0)) \tag{9.11}$$

On peut alors modifier le terme d'attache aux données afin de tenir compte de l'ensemble des images :

$$D(a) = D(x, d(x)) = \frac{1}{N} \sum_{k=1}^{N} |I_0(x) - I_k(x + kd(x))|^2 \tag{9.12}$$

On construit donc un cube d'attache aux données d'axes (x, y, d). Chaque voxel de ce cube contient la valeur D(a). Cette fonction est représentable par un graphe car elle est la somme de fonction réprésentables par graphe.

I_0 est bien sûr choisie pour correspondre à la carte éparse : c'est l'image centrale de l'EPI. De plus, afin de mesurer des disparités sous-pixellique, l'intervalle de disparité considéré pour les étiquettes a un pas de 0,1 pixel, et les images du calcul de l'attache aux données sont interpolées d'un facteur 10.

9.3 Résultats

L'image 9.2(b) montre le résultat de la densification. C'est un bon résultat où tous les éléments de sursol figurent.

Afin de comprendre l'effet de la régularisation, le résultat sans celle-ci est montré en 9.2(a). On constate que les princpiales formes du relief sont trouvées, mais que le résultat est très bruité. La régularisation élimine ce bruit. Mais, en conséquence de la fonction utilisée, le MNE résultant a un aspect constant par morceaux. On observe en particulier que la butte (en bas au centre de l'image), qui est une variation de relief relief douce, est décrite par larges paliers de valeurs. Il faut toutefois avoir à l'esprit que cette dénivelée, d'environ $6,5\mathrm{m}$, correspond sur l'image à une disparité de seulement $1,4$ pixels. On observe un peu la même chose sur la recette, dont le toit a la forme d'une suite de triangles (voir le chapitre 3 pour une image du bâtiment). Sur la figure 9.4, on voit que cette forme a été très clairement retranscrite par les EPI, mais que le *graphcut* ne réussit pas à restituer ce découpage. Cependant, les variations de disparité en question sont de l'ordre d'un ou deux dixièmes de pixel.

Sur certaines zones d'ombre au contour totalement délimité (souvent les ombres des bâtiments), le résultat présente parfois des erreurs. Les valeurs radiométriques sont trop basses pour influencer

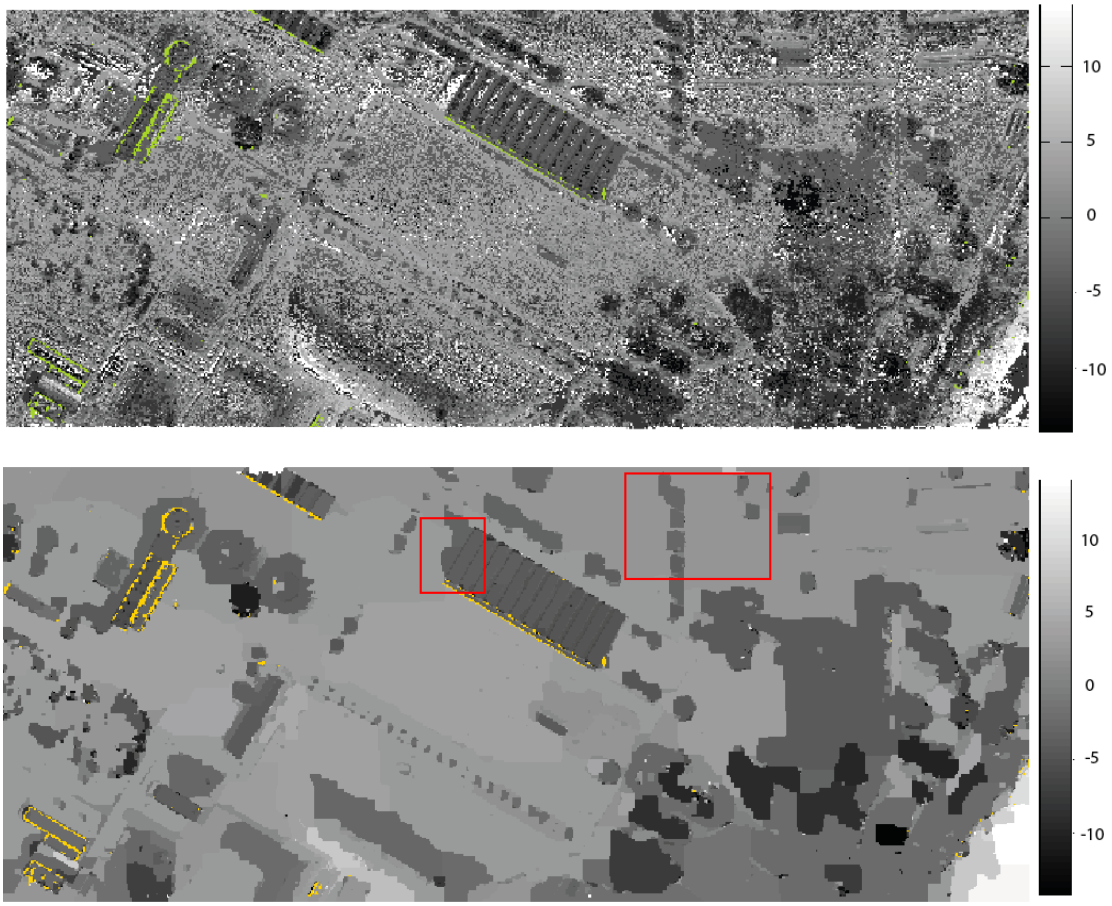


FIGURE 9.2 – **(a) Résultat de la minimisation sans régularisation sur image CFM. -** Le résultat est dicté par le terme d'attache aux données seul. Les pixels verts sont des pixels occultés. **(b) Résultat avec tous les termes d'énergie.** Les pixels occultés sont en jaune. Les échelles sont en mètres.

le résultat et la carte éparse founit des contours précis et avec des disparités très différentes. Alors la régularisation peine à décider à quel bord se rattacher et se trompe, comme sur la zone montrée en figure 9.3(b), qui est l'ombre de la recette sur le sol, mais est vue comme un prolongement du bâtiment. Ces erreurs sont gênantes pour un MNE, mais n'ont cependant pas tant d'impact sur la construction des interférogrammes, puisqu'elles arrivent sur des zones peu texturées.

En revanche, sur la végétation, là où les EPI ne séparaient pas le contour de l'arbre de son ombre, le *graphcut* réussit à délimiter le feuillage de l'ombre. Il le fait en s'appuyant sur les variations de disparité du contour donné par la carte éparse, comme le montre la figure 9.3(a).

Un autre exemple de résultat est donné à partir de la scène Canjuers, figure 9.5. La dénivelée est bien plus importante, et est à la fois liée au terrain accidenté et aux bâtiments. On constate à nouveau des groupements d'éléments avec des valeurs de disparité constantes par morceaux. Le grand nombre de détails avec des variations très faibles de disparité donnent un rendu plus bruité.

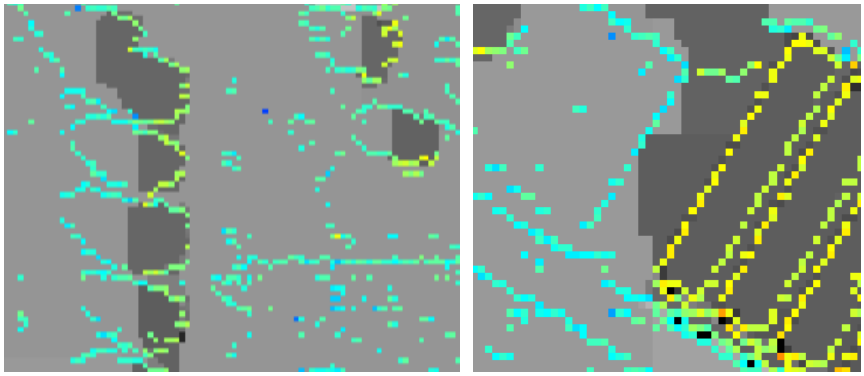


FIGURE 9.3 – **Deux cas de gestion des ombres. -** On montre ici le résultat du graphcut en niveaux de gris sur lequel est superposée en couleur la carte éparse. Les vignettes correspondent aux zones dans les rectangles de la figure 9.2(b). (a) En général, sur les arbres isolés, l'algorithme parvient à séparer le feuillage et son ombre, en suivant les contours donnés par la carte éparse. (b) Entre deux alignements de disparité différente, l'algorithme a des difficultés dans les ombres, où le terme d'attache aux données est trop faible pour être déterminant.

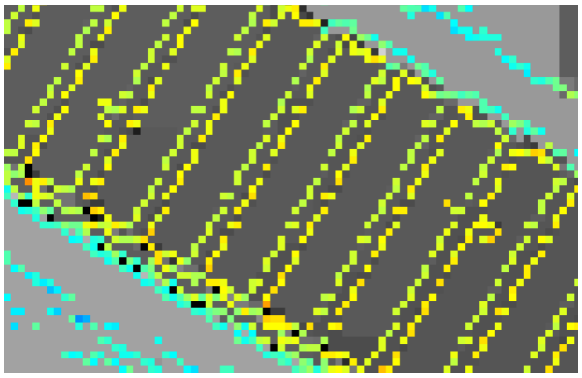


FIGURE 9.4 – **Toit de la recette. -** On montre ici le résultat du graphcut en niveaux de gris sur lequel est superposée en couleur la carte éparse.

9.4 Conclusion

L'utilisation du graphcut a été motivée par ses excellents résultats sur les *benchmark* et par le fait que nous disposions d'un code fonctionnel pour la stéréoscopie à deux vues. Les résultats obtenus sont bons, mais la spécificité des images, avec des disparités très sous-pixelliques, limitent la qualité des résultats qui perdent certains détails de la carte éparse obtenue par EPI au profit d'ensembles de pixels avec des valeurs constantes par morceaux. D'autres énergies globales mériteraient d'être testées.

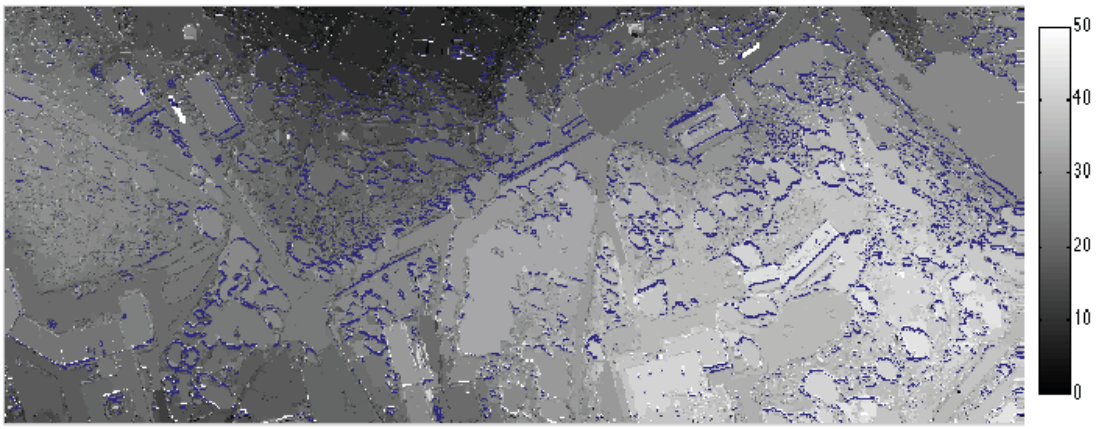


FIGURE 9.5 – **Résultat du *graphcut* pour Canjuers. -** Les pixels occultés sont en bleu foncé. L'échelle est en mètres.

CONCLUSION GÉNÉRALE

Correction des interférogrammes

Les travaux de cette thèse ont pour but d'être utilisés pour construire de bons interférogrammes sur les zones de relief. La figure 9.6 montre un interférogramme sur un bord de bâtiment. L'interférogramme rouge a été réalisé avec un MNE plat. Les valeurs finales s'écartent de la valeur moyenne et sont fausses, puisqu'elles correspondent à un autre objet. La connaissance du modèle numérique d'élévation a permis un meilleur recalage, et l'on constate que l'interférogramme associé, en vert, limite significativement ces erreurs. L'objectif est accompli.

Nous concluons par un résumé des contributions de cette thèse et plusieurs perspectives intéressantes.

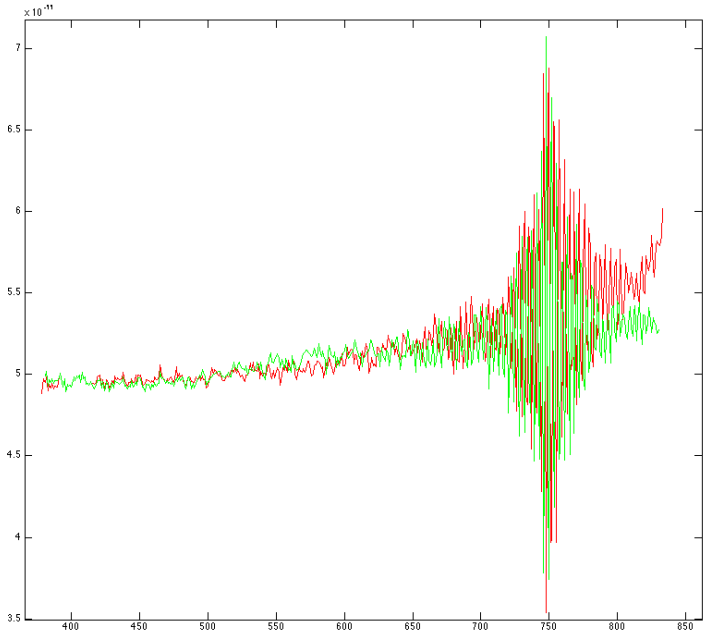


FIGURE 9.6 – **Interférogrammes sur un bâtiment. -** L'interférogramme rouge a été calculé après recalage avec un MNE plat. On observe qu'il est incorret par les déviations de la valeur moyenne. Celui en vert est issu d'images recalées avec un MNE obtenu par les traitements présentés dans cette thèse.

Résumé

Nous avons proposé une méthode de calcul de modèle numérique d'élévation pour les spectro-imageurs aéroportés à trame défilante. Ces instruments se caractérisent par une acquisition dense d'images suivant une trajectoire rectiligne. Ces paramètres ont permis d'exploiter la méthode stéréoscopique des images de plans épipolaires. Les traitements proposés sont adaptés à des mesures de disparités sous-pixelliques. Les résultats montrent des mesures précises au quart de pixel sur la carte éparse résultante. Nous avons aussi proposé une étude sur l'impact sur la disparité des écarts de l'avion à la trajectoire théorique. Celle-ci a permis de confirmer la pertinence des EPI sur des images aéroportées et permettra d'indiquer des tolérances de pilotage sur de prochains vols. Afin de densifier la carte éparse nous avons adapté un algorithme de *graphcut* à une utilisation multi-vues sous-pixellique.

Dans le cas des spectro-imageurs à transformée de Fourier statique, les images acquises de la scène sont superposées à une figure d'interférences. Celles-ci posent des problèmes pour les traitements stéréoscopiques. Nous avons donc développé des algorithmes afin d'extraire les franges d'interférence pour recouvrer l'image panchromatique de la scène. Les algorithmes suivent un modèle de décompostion additif ou multiplicatif. Les deux modèles donnent de bons résultats, mais le modèle multiplicatif, plus proche du modèle de formation d'image, gère mieux les effets de modulation des fréquences de l'image par les franges.

Perspectives

Le succès des calculs est toutefois conditionné à plusieurs critères, qui posent les limites d'utilisation de ce travail, et des perspectives pour la suite.

La hauteur de vol grande devant les variations relatives de relief considérée dans ce travail ne permettrait pas par exemple d'appliquer la méthode à des images acquises par drone. Or la miniaturisation d'instruments hyperspectraux est un domaine qui suscite beaucoup d'intérêt, notamment dans le cadre d'une utilisation sur drone.[1] Dans ce cas, les effets de relief sur les images seront bien plus marqués. Les images des plans épipolaires auront des chemins qui disparaissent sous un autre, lors d'une occultation, ou des chemins qui apparaissent lorsqu'un objet n'est plus occlus. La solution qui peut être envisagée serait d'ajouter après l'extraction de segments une étape d'interprétation des positions relatives des segments en termes d'occultation, désoccultation et dévers sur les images de la scène.

Dans un registre connexe, la trajectoire de l'avion doit être maîtrisée et respecter certains seuils de tolérances en fonction de la précision voulue. Ainsi, lors de vols turbulents ou de basse altitude, les erreurs sur la disparité ne pourront plus être négligées. Il faudra alors corriger les images des écarts en attitude ou translation. Cette correction peut être effectuée à partir des données de navigation. Un écart en translation verticale se corrige par un facteur d'échelle sur l'image et une déviation transversale par une translation. Mais la géométrie de l'image ne sera alors plus cohérente vis-à-vis du centre de l'image. Afin de tenir compte de ce problème, une solution itérative avec pour initialisation un MNE grossier de type SRTM (Shuttle Radar Topography Mission, les données sont téléchargeables gratuitement en [NJ18]), permettra une correction des images

[1] L'ONERA a réalisé une version miniaturisée utilisant les mêmes concepts physiques que Sieleters, [Fos+16].

afin de construire les EPI, pour obtenir un nouveau MNE. On raffinerait ainsi les EPI, et le MNE progressivement.

Une autre solution serait de segmenter la trajectoire en morceaux suivant les écarts de trajectoire observés. On traiterait ensuite l'EPI par régression linéaire pondérée selon la fiabilité de l'estimation (donnée par les écarts à la trajectoire). Afin d'augmenter le nombre d'échantillons fiables, on pourrait utiliser les EPI voisines.

Le cas d'une erreur significative dans la direction de l'axe de vol est plus délicat. On pourrait adapter l'extraction de segments afin qu'elle travaille sur une grille irrégulière selon l'axe de la direction de vol.

Le MNE obtenu est bon, mais présente le défaut d'être constant par morceaux. Il perd certains détails et ne restitue pas bien les variations douces de relief telles que les pentes. Pour éviter cela on pourrait utiliser des énergies du type de [Ish03], qui sont minimisables par *graphcut* mais permettent d'utiliser un terme de régularisation convexe. On aurait alors un relief plus lisse. Cette solution ne gère pas les occultations, mais celles-ci étant minimes sur les images considérées, il pourrait être acceptable de les négliger. On peut également envisager d'utiliser la programmation dynamique. Au lieu de comparer les lignes d'une paire d'images, on pourrait exploiter le fait que toutes les images sont dans une géométrie épipolaire commune. Le point faible de ces méthodes est le manque de cohérence entre les différentes lignes épipolaires, mais il pourrait éventuellement en partie être compensé par la carte éparse.

Enfin, les reprojections d'images induisent des interpolation d'image et donc des interférogrammes échantillonés irrégulièrement. Pour traiter le calcul du spectre à partir d'interférogrammes échantillonnés irrégulièrement, il est possible de s'appuyer sur les travaux de [Car12] où le chapitre 5 s'intéresse au rééchantillonnage des interférogrammes d'un instrument du CNES. Il s'appuie sur deux méthodes de rééchantillonnage sur une grille régulière. La première solution obtient un nouvel interférogramme régulier avec moins d'échantillons avant d'appliquer la transformée de Fourier pour obtenir le spectre. La seconde option dérive une relation matricielle entre l'interférogramme irrégulier et le spectre régulier sans passer par la transformée de Fourier.

ANNEXE A

CALCUL DU TERME DE RÉGULARISATION

Cette annexe détaille le calcul du terme de régularisation pour une expansion α où certains pixels ont une disparité fixée immuable. Le code correspondant dans la fonction `build_smoothness` est donné à titre indicatif (α y est noté `a`). Cette fonction sert à construire le *graphcut* à minimiser en attribuant les coûts correspondants à chaque arête. Pour une description de l'ensemble de l'algorithme et son code on peut se référer à l'article original [KMT14].

Pour rappel, le terme de régularisation pénalise le fait que des pixels adjacents aient des disparités différentes. Dans le cas où la disparité est la même, aucune pénalité n'est attribuée. Ce cas n'est donc pas traité par le code.

On considère 2 pixels adjacents p_1 et p_2 de disparités d_1 et d_2. α est la disparité considérée dans l'*expansion move* courant. Le terme de régularisation impose que des pixels adjacents doivent avoir la même disparité. Il faut donc s'intéresser à trois paires d'assignements possibles :

$$(p_1,\ p_1+\alpha) \text{ et } (p_2,\ p_2+\alpha)$$
$$(p_1,\ p_1+d_1) \text{ et } (p_2,\ p_2+d_1)$$
$$(p_1,\ p_1+d_2) \text{ et } (p_2,\ p_2+d_2)$$

Un assignement est actif si la disparité correspondante est attribuée au pixel à la fin du calcul. Pour chacun de ces cas, les deux assignements doivent être tous deux actifs ou inactifs, sinon il faut ajouter une pénalité.

Dans le code, un noeud peut avoir l'un des statuts suivants :

- De disparité fixe
- Non variable de disparité égale à α
- Variable de disparité égale à α
- Variable de disparité quelconque égale à d
- Occulté

Un statut variable indique que le noeud peut devenir actif, tandis qu'un statut non variable empêche tous les autres de se réaliser car il ne peut devenir inactif.

Les pénalités peuvent être de trois types :

- Constante : s'applique quel que soit l'état final du noeud.
- Dépendante de l'état d'un seul noeud : `add_term1(node,E0,E1).`
- Dépendant de l'état de deux noeuds : `add_term2(node1, node2, E00, E01, E10, E11)`

Première paire d'assignements : $a_1 = (p_1,\ p_1 + \alpha)$ **et** $a_2 = (p_2,\ p_2 + \alpha)$

Ce cas ne s'applique qu'aux noeuds de disparité α variables, ce qui correspond à des pixels dont la disparité est différente de α. Les assignements considérés ne sont donc pas actifs.

```
// 1. Neighbouring assignements (p1 p1+a) and (p2 p2+a)
     if (inRect(p1+a,imSizeR) && inRect(p2+a,imSizeR)){
     // if assignment exists
```

Si les deux assignements sont variables (c'est-à-dire qu'ils peuvent être actifs ou inactifs), il y a pénalité s'ils ne sont pas tous les deux actifs ou inactifs car alors leur disparité sera différente.

```
        if (IS_VAR(a1) && IS_VAR(a2)) // les deux pixels ont une disparite differente de a
            e.add_term2(a1, a2, 0, smoothness_penalty(p1, p2, a),
                smoothness_penalty(p1, p2, a), 0);
```

Si seul a_1 (resp. a_2) est variable, l'autre est forcément occulté ou de disparité différente de α ou de disparité α invariable. Dans ce dernier cas, il faut seulement pénaliser le cas où a_1 (resp. a_2) est inactif. Pour les deux autres cas, d_2 (resp. d_1) est forcément différent de α et il faut donc pénaliser le fait que a_1 (resp. a_2) soit activé, puisque dans ce cas il prendra la valeur α.

```
        else if (IS_VAR(a1)) { // p2 est non variable
            if (d2==a) // on penalise si a1 est desactive
                e.add_term1(a1, smoothness_penalty(p1, p2, a), 0 );
            else // d2!=a, on penalise si a1 est actif
                e.add_term1(a1,0, smoothness_penalty(p1, p2, a) );
        }
        else if (IS_VAR(a2)) { // p1 est non variable
            if (d1==a) // on penalise si a2 est desactive (s'il ne change pas d'etat)
                e.add_term1(a2, smoothness_penalty(p1, p2, a), 0 );
            else // d1!=a, on penalise si a2 est actif (s'il change d'etat)
                e.add_term1(a2,0, smoothness_penalty(p1, p2, a) );
        }
```

Enfin si un seul assignement est actif et de disparité α (donc invariable), l'autre est soit actif de disparité différente à son voisin, soit occulté. Dans ce cas, puisque l'assignement n'est pas variable égal à α (ce qui a été déjà traité dans le cas précédent), il faut forcément pénaliser.

```
        // one is active and d=a, the other is not active or d=a
        else if (d1!=d2 && (d1==a || d2==a))
            e.add_constant(smoothness_penalty(p1, p2, a));
    }
```

Deuxième paire d'assignements : $a_1 = (p_1,\ p_1 + d_1)$ **et** $a_2 = (p_2,\ p_2 + d_1)$

Ce cas ne contient pas les cas où $d_1 = \alpha$ (traité précédemment). Le pixel p_1 est maintenant de disparité différente à α et actif, c'est-à-dire qu'il est fixé en entrée de l'algorithme et immuable. Il n'est donc en particulier pas occulté, puisqu'une valeur fixe ne peut pas prendre la valeur occulté.

```
// 2. Neighbouring assignments (p1 p1+d1) and (p2 p2+d1) - o1 is active
if (inRect(p1+d1,imSizeR) && inRect(p2+d1,imSizeR) && d1!=a) {
    assert(d1!=OCCLUDED);
```

On s'intéresse d'abord aux cas $d_1 = d_2$. Si les deux pixels ont la même disparité, c'est qu'ils sont tous deux actifs (immuables). Ce cas ne nous intéresse donc pas, car il n'est pas pénalisé.

S'ils sont tous deux variables, il y a pénalité si leur états sont différents.

```
if (d1 ==d2 && IS_VAR(o1) && IS_VAR(o2))// on penalise si etats differents
    e.add_term2(o1, o2, 0, smoothness_penalty(p1,p2,d1),
        smoothness_penalty(p1,p2,d1), 0);
```

Si un seul est variable, alors l'autre est forcément fixe (donc invariable). On pénalise si l'assignement variable est désactivé.

```
else if (d1==d2 && IS_VAR(o1)) // penalite si o1 est desactive
    e.add_term1(o1, 0, smoothness_penalty(p1,p2,d1));
else if (d1==d2 && IS_VAR(o2)) // idem
    e.add_term1(o2, 0, smoothness_penalty(p1,p2,d1));
```

Si maintenant $d_1 \neq d_2$:

p_2 est soit occulté soit de disparité d_2 (avec $d_2 = \alpha$ possible) différente de d_1. Il ne prendra donc jamais la valeur d_1. Il faut donc pénaliser si l'assignement a_1 est actif.

```
// si d1!=d2 alors (p2 p2+d1) est forcement inactif
// donc seul l'etat de (p1 p1+d1) compte :
else if (d1!=d2 && IS_VAR(o1) ) // penalite si o1 actif (s'il ne change pas d'etat)
    e.add_term1(o1, smoothness_penalty(p1,p2,d1), 0);
    // penalite obligatoire car un des deux est constant
```

Si d_1 est fixe, alors on pénalise systématiquement.

```
    else if (d1!=d2 && o1==VAR_ALPHA )
        e.add_constant(smoothness_penalty(p1,p2,d1));
}
```

Troisième paire d'assignements : $a_1 = (p_1,\ p_1 + d_2)$ **et** $a_2 = (p_2,\ p_2 + d_2)$

Ce cas ne contient pas les cas où $d_2 = \alpha$ ou bien $d_1 = d_2$ (traités précédemment). On considère ici le pixel p_2 de disparité différente à α actif, il n'est donc en particulier pas occulté. Les cas sont les mêmes que pour le deuxième assignement.

```
// 3. Neighbouring assignments (p1 p1+d2) and (p2 p2+d2) - o1 is active
if (inRect(p1+d2,imSizeR) && inRect(p2+d2,imSizeR) && d2!=a && d2!=d1) {
    assert(d2!=OCCLUDED);
    // si d1!=d2 alors (p1 p1+d2) est forcement inactif donc seul (p2 p2+d2) compte :
    if (d1!=d2 && IS_VAR(o2) ) // penalite si o1 est actif (donc s'il ne change pas d'etat)
        e.add_term1(o2, smoothness_penalty(p1,p2,d2), 0);
        // penalite obligatoire car un des deux est constant
    else if (d1!=d2 && o2==VAR_ALPHA )
        e.add_constant(smoothness_penalty(p1,p2,d2));
}
```

BIBLIOGRAPHY

[Abe16] R. Abergel. “Several mathematical models and fast algorithms for image processing”. PhD thesis. Université Paris Descartes, 2016.

[ADR04] A. Almansa, S. Durand, and B. Rougé. “Measuring and Improving Image Resolution by Adaptation of the Reciprocal Cell”. In: *Journal of Mathematical Imaging and Vision* 21.3 (2004), pp. 235–279. DOI: 10.1023/B:JMIV.0000043739.51886.01.

[AF87] N. Ayache and B. Faverjon. “Efficient registration of stereo images by matching graph descriptions of edge segments”. In: *International Journal of Computer Vision* 1.2 (1987), pp. 107–131. DOI: 10.1007/BF00123161.

[Agu+17] C. Aguerrebere, A. Almansa, J. Delon, Y. Gousseau, and P. Musé. “A Bayesian Hyperprior Approach for Joint Image Denoising and Interpolation, With an Application to HDR Imaging”. In: *IEEE Transactions on Computational Imaging* 3.4 (2017), pp. 633–646. DOI: 10.1109/TCI.2017.2704439.

[AM17] R. Abergel and L. Moisan. “The Shannon Total Variation”. In: *Journal of Mathematical Imaging and Vision* 59.2 (2017), pp. 341–370. DOI: 10.1007/s10851-017-0733-5.

[Auj+06] J.-F. Aujol, G. Gilboa, T. Chan, and S. Osher. “Structure-texture images decomposition - Modeling, algorithms and parameter selection”. In: *International Journal of Computer Vision* (2006).

[Bar+16a] C. Barbanson, A. Almansa, Y. Ferrec, and P. Monasse. “Reconstruction du relief par images des plans épipolaires à partir d’acquisitions aériennes denses”. In: *SFPT 2016 Colloque « Photogrammétrie Numérique et Perception 3D : les Nouvelles Conquêtes »*. Marne La Vallée, France, 2016.

[Bar+16b] C. Barbanson, A. Almansa, Y. Ferrec, and P. Monasse. “Relief Computation from Images of a Fourier Transform Spectrometer for Interferogram Correction”. In: *Light, Energy and the Environment*. Washington, D.C.: OSA, 2016, FM3E.6. DOI: 10.1364/FTS.2016.FM3E.6.

[Bar+17a] C. Barbanson, A. Almansa, Y. Ferrec, and P. Monasse. “Correction des effets de relief en spectro-imagerie aéroportée”. In: *GDR-ISIS - Co-conception : capteurs hybrides et algorithmes pour des systèmes innovants*. 2017.

[Bar+17b] C. Barbanson, A. Almansa, Y. Ferrec, and P. Monasse. “Reconstruction 3D par Images des Plans Epipolaires : Application à un interférométre imageur hyperspectral aéroporté”. In: *Gretsi 2017 - XXVIeme colloque*. 2017.

[Bar+18] C. Barbanson, A. Almansa, Y. Ferrec, and P. Monasse. “Digital Elevation Model estimation with an airborne hyperspectral imager”. In: (2018).

[BBM87] R. Bolles, H. H. Baker, and D. H. Marimont. “Epipolar-Plane Image Analysis : An Approach to Determining Structure From Motion”. In: *International Journal of Computer Vision* 1.1 (1987), pp. 7–55.

[BBM90] P. A. Beardsley, J. M. Brady, and D. W. Murray. “Prediction of stereo disparity using optical flow”. In: *In British Machine Vision Conf.* 1990, pp. 259–264.

[BD06] J. Berent and P. L. Dragotti. “Segmentation of Epipolar-Plane Image Volumes with Occlusion and Disocclusion Competition”. In: *2006 IEEE Workshop on Multimedia Signal Processing*. 2006, pp. 182–185. DOI: 10.1109/MMSP.2006.285293.

[Ber09] D. Bertsekas. *Convex Optimization Theory*. Athena Scientific optimization and computation series. Athena Scientific, 2009.

[Bri+06] X. Briottet, Y. Boucher, A. Dimmeler, A. Malaplate, A. Cini, M. Diani, H. Bekman, P. Schwering, T. Skauli, I. Kasen, I. Renhorn, L. Klasén, M. Gilmore, and D. Oxford. “Military applications of hyperspectral imagery”. In: *Proc. SPIE 6239, Targets and Backgrounds XII : Characterization and Representation*. Vol. 6239. May 2006, 62390B. DOI: 10.1117/12.672030.

[BT98] S. Birchfield and C. Tomasi. “A pixel dissimilarity measure that is insensitive to image sampling”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.4 (1998), pp. 401–406. DOI: 10.1109/34.677269.

[BT99] S. Birchfield and C. Tomasi. “Depth Discontinuities by Pixel-to-Pixel Stereo”. In: *International Journal of Computer Vision* 35.3 (1999), pp. 269–293. DOI: 10.1023/A:1008160311296.

[Bua+10] A. Buades, T. M. Le, J.-M. Morel, and L. A. Vese. “Fast Cartoon + Texture Image Filters”. In: *IEEE Transactions on Image Processing* 19.8 (2010).

[BV04] S. Boyd and L. Vandenberghe. *Convex Optimization*. New York, NY, USA: Cambridge University Press, 2004.

[BV06] Y. Boykov and O. Veksler. “Graph Cuts in Vision and Graphics : Theories and Applications”. In: *Handbook of Mathematical Models in Computer Vision*. Ed. by N. Paragios, Y. Chen, and O. Faugeras. Boston, MA: Springer US, 2006, pp. 79–96. DOI: 10.1007/0-387-28831-7_5.

[BVZ01] Y. Boykov, O. Veksler, and R. Zabih. “Fast approximate energy minimization via graph cuts”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23.11 (2001), pp. 1222–1239. DOI: 10.1109/34.969114.

[BVZ98] Y. Boykov, O. Veksler, and R. Zabih. “Markov random fields with efficient approximations”. In: *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No.98CB36231)*. 1998, pp. 648–655. DOI: 10.1109/CVPR.1998.698673.

[Ca15] C. Coudrain and al. “SIELETERS, an airborne infrared dual-band spectro-imaging system for measurement of scene spectral signatures”. In: *Opt. Express* 23 (2015).

[Car12] J. Caron. “Restauration en échantillonnage irrégulier. Théorie et applications aux signaux et images satelliatires.” PhD thesis. Université de Picardie Jules Verne, 2012.

[CCK91] C. Chang, S. Chatterjee, and P. R. Kube. “On an analysis of static occlusion in stereo vision”. In: *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 1991, pp. 722–723. DOI: 10.1109/CVPR.1991.139799.

[Cha+09] M. Chamberland, V. Farley, P. Lagueux, A. Villemaire, P. Dubois, and J.-P. Gagnon. "Recent Instrument Development at Telops". In: *Advances in Imaging*. Optical Society of America, 2009, FThA2. DOI: 10.1364/FTS.2009.FThA2.

[CK16] M. L. Clark and N. E. Kilham. "Mapping of land cover in northern California with simulated hyperspectral satellite imagery". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 119 (Sept. 2016), pp. 228–245. DOI: 10.1016/j.isprsjprs.2016.06.007.

[Con17] L. Condat. "Discrete Total Variation : New Definition and Minimization". In: *SIAM Journal on Imaging Sciences* 10.3 (2017), pp. 1258–1290. DOI: 10.1137/16M1075247. eprint: https://doi.org/10.1137/16M1075247.

[Cox94] I. J. Cox. "A maximum likelihood N-camera stereo algorithm". In: *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 1994, pp. 733–739. DOI: 10.1109/CVPR.1994.323889.

[CP04] P. L. Combettes and J. C. Pesquet. "Image restoration subject to a total variation constraint". In: *IEEE Transactions on Image Processing* (2004).

[CP10] P. L. Combettes and J.-C. Pesquet. "Proximal Splitting methods in signal processing". In: (2010).

[Cri05] A. Criminisi. "Extracting layers and analysing their specular properties using epipolar-plane-image analysis". In: *Computer Vision and Image Understanding* (2005).

[CS01] T. Chan and J. Shen. "Mathematical models of local non-texture inpaintings". In: *SIAM Journal on Applied Mathematics* 62.3 (2001), pp. 1019–1043.

[CW05] P. L. Combettes and V. R. Wajs. "Signal Recovery by Proximal Forward-Backward Splitting". In: *Multiscale Modeling & Simulation* 4.4 (2005), pp. 1168–1200. DOI: 10.1137/050626090. eprint: https://doi.org/10.1137/050626090.

[CW98] T. Chan and C.-K. Wong. "Total blind deconvolution". In: *IEEE Transactions on Image Processing* (1998).

[DBG08] M. Dalponte, L. Bruzzone, and D. Gianelle. "Fusion of Hyperspectral and LIDAR Remote Sensing Data for Classification of Complex Forest Areas". In: *IEEE Transactions on Geoscience and Remote Sensing* 46.5 (May 2008), pp. 1416–1427. DOI: 10.1109/TGRS.2008.916480.

[DC11] M. Deseilligny and I Clery. "APERO, an open source bundle adjusment software for automatic calibration and orientation of set of images". In: XXXVIII-5/W16 (Jan. 2011).

[Des+02] A. Desolneux, S. Ladjal, L. Moisan, and J.-M. Morel. "Dequantizing Image Orientation". In: *IEEE Transactions on Image Processing* 11.10 (2002).

[DG13] M. Diebold and B. Goldluecke. "Epipolar Plane Image Refocusing for Improved Depth Estimation and Occlusion Handling". In: *Vision, Modeling Visualization*. Ed. by M. Bronstein, J. Favre, and K. Hormann. The Eurographics Association, 2013. DOI: 10.2312/PE.VMV.VMV13.145-152.

[DMM08] A. Desolneux, L. Moisan, and J.-M. Morel. *From gestalt to image analysis : a probabilistic approach*. Springer, 2008.

[DP14] A. Delaunoy and M. Pollefeys. “Photometric Bundle Adjustment for Dense Multi-view 3D Modeling”. In: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. 2014, pp. 1486–1493. DOI: 10.1109/CVPR.2014.193.

[DR07] J. Delon and B. Rougé. “Small Baseline Stereovision”. In: *Journal of Mathematical Imaging and Vision* 28.3 (2007), pp. 209–223. DOI: 10.1007/s10851-007-0001-1.

[Eis+15] A. Eisele, S. Chabrillat, C. Hecker, R. Hewson, I. C. Lau, C. Rogass, K. Segl, T. J. Cudahy, T. Udelhoven, P. Hostert, and H. Kaufmann. “Advantages using the thermal infrared (TIR) to detect and quantify semi-arid soil properties”. In: *Remote Sensing of Environment* 163 (June 2015), pp. 296–311. DOI: 10.1016/j.rse.2015.04.001.

[FEK03] I. Feldmann, P. Eisert, and P. Kauff. “Extension of epipolar image analysis to circular camera movements”. In: *Proceedings 2003 International Conference on Image Processing (Cat. No.03CH37429)*. Vol. 3. 2003, III–697–700 vol.2. DOI: 10.1109/ICIP.2003.1247340.

[Fer08] Y. Ferrec. “Spectro-Imagerie aéroportée par transformatoin de Fourier avec un interféromètre statique à décalage latéral : Réalisation et mise en oeuvre”. PhD thesis. Institut d’Optique - Université Paris Sud XI, 2008.

[FM05] P. E. Forssen and A. Moe. “View matching with blob features”. In: *The 2nd Canadian Conference on Computer and Robot Vision (CRV’05)*. 2005, pp. 228–235. DOI: 10.1109/CRV.2005.88.

[Fos+16] A. P. Fossi, Y. Ferrec, N. Roux, O. D’almeida, N. Guerineau, and H. Sauer. “Miniature and cooled hyperspectral camera for outdoor surveillance applications in the mid-infrared”. In: *Opt. Lett.* 41.9 (2016), pp. 1901–1904. DOI: 10.1364/OL.41.001901.

[FPB18] J. Fade, E. Perrotin, and J. Bobin. “Polarizer-free two-pixel polarimetric camera by compressive sensing”. In: *Appl. Opt.* 57.7 (2018), B102–B113. DOI: 10.1364/AO.57.00B102.

[FZB02] H. Foroosh, J. B. Zerubia, and M. Berthod. “Extension of Phase Correlation to Subpixel Registration”. In: *Trans. Img. Proc.* 11.3 (Mar. 2002), pp. 188–200. DOI: 10.1109/83.988953.

[Gio+12] R. Grompone von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall. “LSD : a Line Segment Detector”. In: *Image Processing On Line* 2 (2012), pp. 35–55. DOI: 10.5201/ipol.2012.gjmr-lsd.

[Gor+96] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen. “The Lumigraph”. In: Association for Computing Machinery Inc., 1996.

[GPK07] J. Gorski, F. Pfeuffer, and K. Klamroth. “Biconvex sets and optimization with biconvex functions : a survey and extensions”. In: *Mathematical Methods of Operations Research* 66.3 (2007), pp. 373–407. DOI: 10.1007/s00186-007-0161-1.

[Hir05] H. Hirschmüller. “Accurate and efficient stereo processing by semi-global matching and mutual information”. In: *In Proc. CVRP*. IEEE Computer Society, 2005, pp. 807–814.

[Hir07] H. Hirschmüller. “Stereo Processing by Semiglobal Matching and Mutual Information”. In: *IEEE Transactions on Pattern Analysis Machine Intelligence* 30 (June 2007), pp. 328–341. DOI: 10.1109/TPAMI.2007.1166.

[HS09] H. Hirschmuller and D. Scharstein. “Evaluation of Stereo Matching Costs on Images with Radiometric Differences”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31.9 (2009), pp. 1582–1599. DOI: 10.1109/TPAMI.2008.221.

[HZ04] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[Ish03] H. Ishikawa. “Exact Optimization for Markov Random Fields with Convex Priors”. In: *IEEE Transactions on Pattern Analysis Machine Intelligence* 25 (2003), pp. 1333–1336. DOI: doi.ieeecomputersociety.org/10.1109/TPAMI.2003.1233908.

[Jel+17] R. A. Jellal, M. Lange, B. Wassermann, A. Schilling, and A. Zell. “LS-ELAS : Line segment based efficient large scale stereo matching”. In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. 2017, pp. 146–152. DOI: 10.1109/ICRA.2017.7989019.

[Jor+15] L. Jorissen, P. Goorts, S. Rogmans, G. Lafruit, and P. Bekaert. “Multi-camera epipolar plane image feature detection for robust view synthesis”. In: *2015 3DTV-Conference : The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON)*. 2015, pp. 1–4. DOI: 10.1109/3DTV.2015.7169362.

[Kan+95] T. Kanade, H. Kano, S. Kimura, A. Yoshida, and K. Oda. “Development of a video-rate stereo machine”. In: *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*. Vol. 3. 1995, 95–100 vol.3. DOI: 10.1109/IROS.1995.525868.

[KMS16] S. Kim, D. Min, and K. Sohn. “ANCC flow : Adaptive normalized cross-correlation with evolving guidance aggregation for dense correspondence estimation”. In: *2016 IEEE International Conference on Image Processing (ICIP)*. 2016, pp. 3454–3458. DOI: 10.1109/ICIP.2016.7533001.

[KMT14] V. Kolmogorov, P. Monasse, and P. Tan. “Kolmogorov and Zabih’s Graph Cuts Stereo Matching Algorithm”. In: *Image Processing On Line* 4 (2014), pp. 220–251. DOI: 10.5201/ipol.2014.97.

[KS00] K. Kutulakos and S. M. Seitz. “A theory of shape by space carving”. In: *IJCV* (2000).

[KZ01] V. Kolmogorov and R. Zabih. “Computing visual correspondence with occlusions using graph cuts”. In: *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*. Vol. 2. 2001, 508–515 vol.2. DOI: 10.1109/ICCV.2001.937668.

[KZ02a] V. Kolmogorov and R. Zabih. “Multi-camera Scene Reconstruction via Graph Cuts”. In: *Computer Vision — ECCV 2002 : 7th European Conference on Computer Vision Copenhagen, Denmark, May 28–31, 2002 Proceedings, Part III*. Ed. by A. Heyden, G. Sparr, M. Nielsen, and P. Johansen. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 82–96. DOI: 10.1007/3-540-47977-5_6.

[KZ02b] V. Kolmogorov and R. Zabih. “What Energy Functions Can Be Minimized via Graph Cuts ?” In: *Computer Vision — ECCV 2002*. Ed. by A. Heyden, G. Sparr, M. Nielsen, and P. Johansen. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 65–81.

[KZP13] C. Kim, H. Zimmer, and Y. Pritch. “Scene reconstruction from High Spatio-Angular Resolution Light Fields”. In: *Disney Reasearch* (2013).

[Low04] D. G. Lowe. “Distinctive Image Features from Scale-Invariant Keypoints”. In: *International Journal of Computer Vision* 60.2 (2004), pp. 91–110. DOI: 10.1023/B:VISI.0000029664.99615.94.

[Luc+12] P. G. Lucey, M. Wood, S. T. Crites, and J. Akagi. *A LWIR hyperspectral imager using a Sagnac interferometer and cooled HgCdTe detector array*. 2012. DOI: 10.1117/12.918970.

[Man+95] Mann, Picard, S. Mann, and R. W. Picard. “On Being ‘undigital’ With Digital Cameras : Extending Dynamic Range By Combining Differently Exposed Pictures”. In: *Proceedings of IST*. 1995, pp. 442–448.

[MAP11] P. Maurel, J.-F. Aujol, and G. Peyré. “Locally Parallel Texture Modeling”. In: *SIAM Journal on Imaging Sciences* 4.1 (2011), pp. 413–447. DOI: 10.1137/090770795. eprint: https://doi.org/10.1137/090770795.

[Mar+02] A. Marino, G. Ludovisi, A. Moccaldi, and S. Bellagamba. “Imaging spectrometry and GIS techniques application for the study of seawater quality and pollution hazard”. In: *Proc. SPIE 4488, Ocean Optics : Remote Sensing and Underwater Imaging*. Vol. 4488. 2002, pp. 264–268. DOI: 10.1117/12.452822.

[MBM97] B. Marcel, M. Briot, and R. Murrieta. “Calcul de Translation et Rotation par la Transformation de Fourier”. In: *Colloque Gretsi Saint Martin d’Hères*. 1997.

[MG01] F. Malgouyres and F. Guichard. “Edge Direction Preserving Image Zooming : A Mathematical and Numerical Analysis”. In: *SIAM Journal on Numerical Analysis* 39.1 (2001), pp. 1–37. DOI: 10.1137/S0036142999362286.

[Moi07] L. Moisan. “How to discretize the Total Variation of an image ?” In: *PAMM* 7.1 (2007), pp. 1041907–1041908. DOI: 10.1002/pamm.200700424.

[Mon11] P. Monasse. “Quasi-Euclidean Epipolar Rectification”. In: *Image Processing On Line* 1 (2011). DOI: 10.5201/ipol.2011.m_qer.

[Mor62] J. J. Moreau. “Fonctions convexes duales et points proximaux dans un espace hilbertien.” In: *Comptes Rendus de l’Académie des Sciences (Paris), Série A* 255 (1962), pp. 2897–2899.

[MSS16] N. Monzón, A. Salgado, and J. Sánchez. “Robust Discontinuity Preserving Optical Flow Methods”. In: *Image Processing On Line* 6 (2016), pp. 165–182. DOI: 10.5201/ipol.2016.172.

[MTE02] J. Mellor, S. Teller, and T. As Lozano-p Erez. *Dense Depth Maps from Epipolar Images*. Tech. rep. MASSACHUSETTS INSTITUTE OF TECHNOLOGY ARTIFICIAL INTELLIGENCE LABORATORY, Aug. 2002.

[Mus+06] P. Musé, F. Sur, F. Cao, Y. Gousseau, and J.-M. Morel. “An a contrario decision method for shape element recognition”. In: *International Journal of Computer Vision* 69.3 (2006), pp. 295–315.

[Ng+05] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan. *Light Field Photography with a Hand-held Plenoptic Camera*. Tech. rep. Stanford University, Apr. 2005.

[Nik00] M. Nikolova. "Local Strong Homogeneity of a Regularized Estimator". In: *SIAM Journal on Applied Mathematics* 61.2 (2000), pp. 633–658. DOI: 10.1137/S0036139997327794.

[NJ18] NASA-JPL. *U.S. Releases Enhanced Shuttle Land Elevation Data*. 2018. URL: https://www2.jpl.nasa.gov/srtm/ (visited on 04/05/2018).

[NS14] M. Nikolova and G. Steidl. "Fast Ordering Algorithm for Exact Histogram Specification". In: *IEEE Transactions on Image Processing* 23.12 (2014), pp. 5274–5283. DOI: 10.1109/TIP.2014.2364119.

[NT17] M. Nikolova and P. Tan. "Alternating proximal gradient descent for nonconvex regularised problems with multiconvex coupling terms". working paper or preprint. Aug. 2017.

[NWC13] M. Nikolova, Y.-W. Wen, and R. Chan. "Exact Histogram Specification for Digital Images Using a Variational Approach". In: *Journal of Mathematical Imaging and Vision* 46.3 (2013), pp. 309–325. DOI: 10.1007/s10851-012-0401-8.

[OK85] Y Ohta and T. Kanade. "Stereo by intra - and inter - scanline search". In: (Jan. 1985).

[OK90] M. Okutomi and T. Kanade. "A locally adaptive window for signal matching". In: *[1990] Proceedings Third International Conference on Computer Vision*. 1990, pp. 190–199. DOI: 10.1109/ICCV.1990.139519.

[OK93] M. Okutomi and T. Kanade. "A multiple-baseline stereo". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15.4 (1993), pp. 353–363. DOI: 10.1109/34.206955.

[OSV03] S. Osher, A. Solé, and L. A. Vese. "Image Decomposition and Restoration using Total Variation Minimization and the H1 norm". In: *Multiscale Modeling and Simulation* 1.3 (2003), pp. 349–370.

[PB13] N. Parikh and S. Boyd. "Proximal Algorithms". In: *Foundations and Trends in Optimization* (2013).

[Per+10] R. Perrier, E. Arnaud, P. Sturm, and M. Ortner. "Satellite image registration for attitude estimation with a constrained polynomial model". In: *2010 IEEE International Conference on Image Processing*. 2010, pp. 925–928. DOI: 10.1109/ICIP.2010.5650997.

[Pey10] G. Peyré. *Toolbox for non-smooth convex optimization using first order schemes.* 2010. URL: www.numerical-tours.com (visited on 2018).

[PKF07] J.-P. Pons, R. Keriven, and O. Faugeras. "Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score". In: *The International Journal of Computer Vision* 72.2 (2007), pp. 179–193.

[RH06] P. M. Rojo and J. Harrington. "A Method to Remove Fringes from Images Using Wavelets". In: *The Astrophysical Journal* 649.1 (2006), p. 553.

[ROF92] L. I. Rudin, S. Osher, and E. Fatemi. "Nonlinear total variation based noise removal algorithms". In: *Physica D* (1992).

[Roq+04] S. Roques, L. Jahan, B. Rouge, and C. Thiebaut. “Satellite attitude instability effects on stereo images”. In: *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 3. 2004, iii–477–80 vol.3. DOI: 10.1109/ICASSP.2004.1326585.

[SAA12] M. Santoro, G. AlRegib, and Y. Altunbasak. “Misalignment correction for depth estimation using stereoscopic 3-D cameras”. In: *2012 IEEE 14th International Workshop on Multimedia Signal Processing (MMSP)*. 2012, pp. 19–24. DOI: 10.1109/MMSP.2012.6343409.

[Sal+] R. d. R. Salles, C. R. de Souza Filho, T. Cudahy, L. E. Vicente, and L. V. S. Monteiro. “Hyperspectral remote sensing applied to uranium exploration : A case study at the Mary Kathleen metamorphic-hydrothermal U-REE deposit, NW, Queensland, Australia”. In: *Journal of Geochemical Exploration* (). DOI: 10.1016/j.gexplo.2016.07.002.

[SAM11] N. Sabater, A. Almansa, and J.-M. Morel. “Meaningful Matches in Stereovision”. In: *IEEE Trans. PAMI* 34.5 (2011), pp. 930–42. DOI: 10.1109/TPAMI.2011.207. arXiv: 1112.1187.

[SAM12] N. Sabater, A. Almansa, and J.-M. Morel. “Meaningful matches in stereovision”. In: *IEEE Transactions on pattern analysis and machine intelligence* (2012).

[Sei+06] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. “A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms”. In: *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1*. CVPR ’06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 519–528. DOI: 10.1109/CVPR.2006.19.

[SO08] J. Shi and S. Osher. “A Nonlinear Inverse Scale Space Method for a Convex Multiplicative Noise Model”. In: *SIAM Journal on Imaging Sciences* 1.3 (2008), pp. 294–321. DOI: 10.1137/070689954. eprint: https://doi.org/10.1137/070689954.

[Son+17] D. C. Soncco, C. Barbanson, M. Nikolova, A. Almansa, and Y. Ferrec. “Fast and Accurate Multiplicative Decomposition for Fringe Removal in Interferometric Images”. In: *IEEE Transactions on Computational Imaging* 3.2 (2017), pp. 187–201. DOI: 10.1109/TCI.2017.2678279.

[Spi+08] C. Spinetti, V. Carrère, M. F. Buongiorno, A. J. Sutton, and T. Elias. “Carbon dioxide of Pu‘u‘O‘o volcanic plume at Kilauea retrieved by AVIRIS hyperspectral data”. In: *Remote Sensing of Environment* 112.6 (June 2008), pp. 3192–3199. DOI: 10.1016/j.rse.2008.03.010.

[SS02] D. Scharstein and R. Szeliski. “A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms”. In: *International Journal of Computer Vision* 47.1 (2002), pp. 7–42. DOI: 10.1023/A:1014573219977.

[SSS06] N. Snavely, S. M. Seitz, and R. Szeliski. “Photo tourism : Exploring photo collections in 3D”. In: *In Proc. ACM SIGGRAPH*. 2006.

[STM06] T. Stich, A. Tevs, and M. Magnor. “Global depth from epipolar volumes - A general framework for reconstructing non-lambertian surfaces”. In: *Proceedings of the Third International Symposium on 3D Data Processing, Visualization and Transmission* (2006).

[Sze] R. Szeliski. “Computer Vision : Algorithms and Applications”. In: Springer London. Chap. 11.

[Tan16] P. Tan. “Précision de modèle et efficacité algorithmique : exemples du traitement de l’occultation en stéréovision binoculaire et de l’accélération de deux algorithmes en optimisation convexe”. Thèse de doctorat dirigée par Chambolle, Antonin et Monasse, Pascal Mathématiques appliquées Paris Saclay 2016. PhD thesis. 2016.

[Tav+08] F. Tavin, A. Roman, S. Mathieu, F. Baret, W. Liu, and P. Gouton. “Comparison of Metrics for the Classification of Soils Under Variable Geometrical Conditions Using Hyperspectral Data”. In: *IEEE Geoscience and Remote Sensing Letters* 5.4 (Oct. 2008), pp. 755–759. DOI: 10.1109/LGRS.2008.2005212.

[TH86] Q. Tian and N. Huhns. “Algorithms for subpixel registration”. In: *Computer Vision, Graphics, and Image Processing* 35 (1986), pp. 220–233.

[Tri+00] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. “Bundle Adjustment – A Modern Synthesis”. In: *VISION ALGORITHMS : THEORY AND PRACTICE, LNCS*. Springer Verlag, 2000, pp. 298–375.

[UWI11] C. Unger, E. Wahl, and S. Ilic. “Efficient Stereo and Optical Flow with Robust Similarity Measures”. In: *Pattern Recognition : 33rd DAGM Symposium, Frankfurt/Main, Germany, August 31 – September 2, 2011. Proceedings*. Ed. by R. Mester and M. Felsberg. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 246–255. DOI: 10.1007/978-3-642-23123-0_25.

[Vog+07] G. Vogiatzis, C. Hernández Esteban, P. H. S. Torr, and R. Cipolla. “Multiview Stereo via Volumetric Graph-Cuts and Occlusion Robust Photo-Consistency”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 29.12 (Dec. 2007), pp. 2241–2246. DOI: 10.1109/TPAMI.2007.70712.

[Wan+15] F. Wang, J. Zhou, J. Jing, Q. Wu, and W. Cheng. *Research on LASIS interferogram processing*. 2015. DOI: 10.1117/12.2205859.

[Wet+12] G. Wetzstein, I. Ihrke, D. Lanman, W. Heidrich, K. Akeley, and R. Raskar. “Computational Plenoptic Imaging”. In: *SIGGRAPH Course* (2012).

[WG12] S. Wanner and B. Goldluecke. “Globally Consistent depth labeling of 4D light fields”. In: *IEEE* (2012).

[WZC15] J. Wen, J. Zhao, and W. Cailing. “Improved MCA–TV algorithm for interference hyperspectral image decomposition”. en. In: *Optics and Lasers in Engineering* 75 (Dec. 2015), pp. 81–87. DOI: 10.1016/j.optlaseng.2015.07.001.

[XTA87] G. Xu, S. Tsuji, and M. Asada. “A Motion Stereo Method Based on Coarse-to-Fine Control Strategy”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-9.2 (1987), pp. 332–336. DOI: 10.1109/TPAMI.1987.4767908.

[YBVRZ01] Y Y. Boykov, O Veksler R., and R Zabih. “Efficient Approximate Energy Minimization via Graph Cuts”. In: 20 (Jan. 2001).

[YL08] H. Yan and J. G. Liu. “Robust phase correlation based feature matching for image co-registration and DEM generation”. In: *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 37.PART B7 (2008), pp. 1751–1756.

[Zha+95] Z. Zhang, Z. Zhang, R. Deriche, R. Deriche, O. Faugeras, O. Faugeras, Q. tuan Luong, Q. tuan Luong, P. Robotique, and P. Robotvis. “A Robust Technique for Matching Two Uncalibrated Images Through the Recovery of the Unknown Epipolar Geometry”. In: *Artificial Intelligence* 78 (1995), pp. 87–119.

[Zho+12] X. Zhou, A. G. Podoleanu, Z. Yang, T. Yang, and H. Zhao. “Morphological operation-based bi-dimensional empirical mode decomposition for automatic background removal of fringe patterns”. In: *Opt. Express* 20.22 (2012), pp. 24247–24262. DOI: 10.1364/OE.20.024247.

[ZN96] W. Zhao and N. Nandhakumar. “Effects of camera alignment errors on stereoscopic depth estimates”. In: *Pattern Recognition* 29.12 (1996), pp. 2115 –2126. DOI: https://doi.org/10.1016/S0031-3203(96)00051-9.

[Tan+16] T. Taniai, Y. Matsushita, Y. Sato, and T. Naemura. “Continuous 3D Label Stereo Matching using Local Expansion Moves”. In: *ArXiv e-prints* (Mar. 2016). arXiv: 1603.08328 [cs.CV].

TABLE DES FIGURES

1.1 **Imagerie hyperspectrale. -** (a) Cube de données hyperspectrales avec les coordonnées spatiales (x,y) et la coordonnée spectrale (z). Dans ce cube, tous les plans perpendiculaires à $\vec{z}$ correspondent à une image de la scène à une longueur d'onde donnée. (b) Spectre associé à un pixel. *Source : doi :10.1038/ nphoton.2009.205* 8

1.2 **Transmittance de l'atmosphère en fonction de la longueur d'onde. -** L'absorption par la vapeur d'eau et par le dioxide de carbone définit les fenêtres atomsphériques. *Source : https:// fr.wikipedia.org/ wiki/ Fenêtre_atmosphérique* 9

1.3 **Superposition des interférences. -** Chaque onde génère son propre système d'interférences. A la différence de marche nulle, toutes les ondes sont en phase donc les franges sont contrastées. Lorsque l'on s'éloigne, les figures ne se superposent plus, et l'on ne distingue plus de franges. Elles sont brouillées. Il peut arriver que certaines interférences coïncident à nouveau, ce qui peut fournir des franges légèrement contrastées au-delà de l'origine. C'est ce que l'on voit sur la dernière oscillation du schéma. . . . 10

1.4 **Sieleters BIII monté sur plateforme stabilisée.** 11

1.5 **Schéma optique de l'interféromètre de Sieleters. -** Il s'agit d'un interféromètre de Michelson à dièdres de décalage latéral t (par rapport à une configuration symétrique des deux miroirs selon la séparatrice). Le rayon entrant est divisé par la séparatrice, chaque rayon résultant est réfléchi par un dièdre et une lentille en sortie les fait converger en un même point du détecteur. 12

1.6 **Figure de calcul de la différence de marche. -** Le décalage entre les deux dièdres est $\frac{t}{2}$, ce qui après réflexion sur le dièdre donne un décalage t entre les deux branches de l'interféromètre. δ est la différence de marche. 13

1.7 **Image issue de Sieleters. -** Les franges du carré rose, autour de $\delta = 0$ sont très contrastées. Elles sont brouillées sur le carré bleu en raison de la grande ouverture spectrale de l'instrument. Puis, vers le haut de l'image, dans le carré vert, les franges sont à nouveau contrastées mais plus légèrement. 13

1.8 **Principe d'acquisition hyperspectrale de Sieleters. -** Le déplacement de l'avion (indiqué par la flèche bleue) permet aux éléments de la scène d'être vus sous différents états d'interférence. En associant les différentes valeurs prises par un objet (désigné par une croix rouge) sur chaque image, on accède à son interférogramme. Le spectre est obtenu ensuite en appliquant la transformée de Fourier. 14

1.9 **Effet du relief. -** Le sommet de prise de vue (S) observe le point M et lui attribue la position M' au lieu de sa position réelle M". La zone grisée est invisible pour le capteur, on parle d'occlusion. 15

1.10 **Interférogramme mal reconstruit sur un bâtiment. -** (a) et (b) montrent des détails d'images du début et de la fin de la séquence acquise par Sieleters. Ces images ont été recalées en faisant l'hypothèse d'un sol plat. Les interférogrammes en (c) correspondent aux croix rouge et verte sur les imagettes. Ces croix sont situées sur le haut d'un bâtiment et chacune pointe le même pixel dans les deux images. La croix rouge, en bord de toit, pointe sur le bord noir dans (a) et sur l'intérieur blanc du toit en (b) en raison de la parallaxe. L'interférogramme rouge est donc incorrect, et c'est effectivement ce que l'on observe en le traçant. L'interférogramme vert est correct car il se trouve sur une zone homogène. 16

2.1 **Schéma du modèle de caméra sténopé. -** *Source : [HZ04]* 20

2.2 **Schéma de principe pour la stéréovision après prise en compte du déplacement de la caméra de C1 à C2. -** C1 et C2 sont les sommets des images , B la base, H la hauteur de vol, f la focale, d la disparité et h la hauteur du point M que l'on cherche à déterminer. M' correspond à la reprojection au sol de M via C1. 21

2.3 **(a) Géométrie epipolaire -** Les centres des deux clichés, C et C', et un point X de la scène constituent un plan épipolaire π. L'intersection entre ce plan et chacune des images est une ligne : les lignes épipolaires l et l'. **(b) Axe des épipoles -** Les épipoles e et e' définissent un axe. L'ensemble des plans contenant cet axe forment l'ensemble des plans épipolaires de la paire d'images. *Source : [HZ04]* 23

2.4 **Illustration du phénomène d'adhérence. -** Les points P et Q étant radiométriquement similaires sur les images, ils sont appariés. Le point Q prend alors la même disparité que P. L'emprise du bâtiment est alors étendue. *Source : [DR07]* 25

2.5 **Algorithme de block matching sur un couple d'images Sieleters. -** Les images (a) et (b) sont deux images d'une séquence Sieleters dont on a enlevé les franges (cf. Partie 2). (c) est la carte de disparité correspondante obtenue par l'algorithme de [Mon11]. L'intervalle de disparités est [-142 -137] pixels au quart de pixel près. 26

2.6 **Algorithme de flot optique sur un couple d'images Sieleters. -** Les images Sieleters utilisées sont les mêmes que sur la figure 6.1. L'image (a) est l'image de référence et (b) montre le flot optique correpondant, calculé avec l'algorithme [MSS16]. 27

2.7 **Correspondances en programmation dynamique -** Les abscisses représentent une ligne épipolaire de l'image de gauche, les ordonnées sa correspondante sur l'image de droite. Les lettres en minuscules associées sont les valeurs d'intensité de chaque pixel de la ligne. L'intervalle de disparités autorisées est signalé par les cases en blanc. L'appariement de deux pixels est signifié par un **M**. **L** et **R** indiquent que le pixel est occulté dans l'image de gauche et de droite respectivement. Ces lettres constituent donc un chemin. *Source : [SS02]* 28

2.8 **Construction d'une image des plans épipolaires. -** A droite, deux images issues d'une séquence acquise par un déplacement rectiligne vers la droite. Les lignes épipolaires correspondantes sont surlignées en rouge, et la ligne est prolongée pour montrer leur position sur l'image des plans épipolaires à gauche. s est l'axe des numéros d'images, u est l'axe spatial de la direction du mouvement et v est l'axe spatial transversal au mouvement. Source : [KZP13] 31

2.9 **Construction du cube spatio-temporel. -** Les trois clichés, obtenus par un déplacement rectiligne de la caméra vers la gauche, observent un même point p. A partir de ces images on peut construire un cube, dont l'EPI est une coupe transversale qui décrit par une ligne le déplacement du point selon les différents points de vue. Crédits : [BBM87] 32
2.10 **Schéma de deux images en géométrie épipolaire. -** Le point M de la scène (en haut) est observé par les images (en bas) de centres C1 et C2, de focale f et base B. D'après : [BBM87] 33

3.1 **Schéma de calcul des images simulées de franges, panchromatique et interférométrique.** 36
3.2 **Images simulées pour une même scène.** 37
3.3 **Images simulées interférométriques. -** L'image (j) a été suréchantillonnée pendant les traitements, ce qui explique qu'elle soit plus nette que les autres. 38
3.4 **Images du centre ONERA de Le Fauga-Mauzac. -** Les points rouges sont les points de contrôle utilisés au chapitre 8. *Source : Géoportail* 39
3.5 **Photographies du centre ONERA de Le Fauga-Mauzac. -** Les photographies ont été prises en septembre 2017. 41
3.6 **Image de la scène Canjuers. -** *Source : Géoportail* 42

4.1 **Résultat de l'algorithme de block-matching. -** (c) montre le résultat de l'algorithme de block-matching [SAM11] pour le couple d'image (a) et (b). Les pixels colorés en cyan correspondent à des points pour lesquels l'algorithme n'a pas trouvé d'appariemment. 46
4.2 **Image des plans épipolaires obtenue à partir d'images Sieleters. -** On observe des franges d'interférence obliques sur l'ensemble de l'EPI, et en particulier sur la droite, où elles sont très contrastées. Cela se répercute sur les orientations des gradients de l'image (en degrés), avec des nombreuses valeurs proches de la verticale ($\pm 180^\circ$). . 48
4.3 **Transformée de Fourier d'une image Sieleters -** Dans le voisinage des basses fréquences verticales, deux lignes quasi verticales correspondent aux fréquences propres des franges. L'une d'elles est désignée par une flèche verte. 49
4.4 **Image Sieleters et sa transformée de Fourier. -** La flèche verte montre les fréquences des franges pour une scène homogène. La flèche rouge indique les fréquences propres à la scène, particulièrement au bâtiment de la recette en bas au milieu de l'image. Enfin, en jaune sont désignées les répliques des franges dues à la texture de la scène. 50

5.1 **Décomposition cartoon-texture -** (a) Image originale. (b) Composante lisse (*cartoon*). (c) Composante contenant la texture. Crédits : [OSV03] 53
5.2 **Illustration de l'estimation de l'opérateur de proximité. -** Les lignes noires désignent les lignes de niveau de la fonction, la ligne en gras est la frontière de son domaine. Les estimations sur les points bleus donnent un résultat en rouge. Tous tendent vers le minimum de la fonction. *Source : [PB13]* 55
5.3 **Approximation lisse de la norme l_1. -** En bleu : $\phi(t) = |t|$. En vert pointillé : $\phi_\alpha = |t| - \alpha \log(1 + \frac{|t|}{\alpha})$. On observe que ϕ_α est une bonne approximation dérivable en 0 de ϕ. 57
5.4 **Résultat sur une image simulée additive.** 62
5.5 **Résultat sur une image simulée physique.** 63

5.6 **Détails de la scène en figure 5.5 sur les résultats obtenus à partir des images simulées physiques et additives. -** Sur les résultats des images simulées physiques, des franges demeurent autour des zones à fort contraste. 64

5.7 **Résultats sur une image Sieleters. -** Les transformées de Fourier sont obtenues après prolongement symétrique de l'image et apodisation par une fenêtre de Hanning. On remarque sur l'image (d) que la zone $\tilde{\Omega}$, bien que discernable, n'est pas vide, et qu'elle a bien été complétée par des fréquences de l'image. 65

5.8 **Détails des images. -** (a) et (b) sont des imagettes issues de l'image Sieleters, dans une zone où les franges sont très contrastées, (c) et (d) sont ces mêmes détails après régularisation. 66

5.9 **Extrait de l'image de la figure 5.7 avec des résidus de franges -** Dans (a), les résidus de franges correspondent aux lignes horizontales. On remarque dans (b) que l'imagette contient des fréquences de franges dans Ω, mais aussi dans Ω^C (entourées en jaune). 66

6.1 **Isolignes de fonctions biconvexes -** (a) La fonction présente des minima locaux sur les courbes des deux hyperboles $xy = 1$. (b) La fonction présente un minium et plusieurs points stationnaires ne correspondant pas à des minima. 70

6.2 **Comparaison des transformées de Fourier d'une image simulée panchromatique et sa correspondante avec franges -** En vert, valeurs pour l'image panchromatique et en bleu l'image avec franges. Les courbes sont réalisées à partir de la moyenne des lignes ou colonnes d'une image. (a) Les transformées de Fourier selon les lignes présentent des valeurs très proches pour les deux images. (b) Pour les colonnes, les courbes diffèrent autour des fréquences propres des franges. 71

6.3 **Oracle pour les algorithmes multiplicatifs. -** 75

6.4 **Influence des variations des paramètres. -** Les valeurs données correspondent à la médiane des résultats sur images simulées. L'influence est mesurée par la différence entre le score de PSNR pour la meilleure valeur du paramètre et le score pour la valeur de paramètre donnée en abscisse. Les valeurs résultantes sont donc nulles (en particulier lorsque la valeur testée est celle qui a été choisie) ou négatives. Les algorithmes ne sont pas sensibles à des variations d'un facteur 2 autour de la valeur choisie. . . . 76

6.5 **Histogrammes des résultats de la comparaison des deux algorithmes multiplicatifs par calcul du PSNR. -** Ces résultats sont obtenus par comparaison de 30 images non redondantes issues de Sieleters. 79

6.6 **Résultats de l'algorithme 3 sur une image Sieleters. -** 80

6.7 **Spectres sur l'infrarouge thermique. -** Obtenu avec Sieleters sur diverses cibles. *Source : ONERA.* 81

6.8 **Spectres sur le moyen infrarouge. -** Obtenu avec Sieleters sur diverses cibles. *Source : ONERA.* 81

6.9 **Spectres dans le domaine visibles. -** Obtenus avec l'instrument CaHyD. *Source : ONERA.* 82

6.10 **Détail d'une image Sieleters. -** Les traces horizontales des champs ont une fréquence et orientation proche de celle des franges. Pour cela, elles ont tendance à être effacées. 82

6.11 **Détails de la scène en figure 6.12 avant et après traitement. -** Les imagettes de la première colonne sont issues de l'image d'origine (avec franges). Celles du milieu du résultat du modèle additif et celles de la troisième colonne du modèle multiplicatif. La première ligne montre les lacunes du modèle additif sur les motifs à haute fréquence. Les deux autres montrent des scènes plus lisses, où peu de différences sont observées entre les deux modèles d'image. 87

6.12 **Comparaison des résultats sur le spectre de Fourier. -** L'algorithme utilisé pour l'image (c) est celui par filtrage non-linéaire. Les fréquences des franges et leurs répliques sont enlevées par le modèle multiplicatif mais conservées par le modèle additif. 88

7.1 **(b) Image des plans épipolaires lorsque l'orientation de la caméra varie selon ses trois axes. -** Le schéma (a) montre, pour plusieurs images successives, la position d'une même ligne épipolaire. La variation d'orientation a pour résultat de modifier la position des épipoles, et donc des lignes épipolaires. Extraire les lignes épipolaires de chaque image pour construire l'EPI devient plus difficile. Le traitement de l'EPI est lui aussi compliqué. Source : [BBM87] 92

7.2 **Angles de roulis, tangage et lacet.** 93

7.3 **Effet de la non uniformité de la vitesse de vol sur l'EPI -** (a) Chaque point correspond à un échantillon d'un chemin épipolaire. La ligne horizontale en haut de la figure montre les positions successives de l'avion le long de la ligne de vol, correspondant à chacune des colonnes de l'EPI. Sur les points rouges, la vitesse du porteur a doublé. L'échantillonnage de l'EPI ne rend toutefois pas compte de cette variation et reste constant. Dans ce cas, le chemin dévie de sa direction attendue (en pointillés) et sa pente augmente brusquement. Les estimations de disparité sont alors mauvaises. (b) Sur cette EPI, la partie centrale a été obtenue en n'utilisant qu'une image sur cinq, ce qui permet de simuler le cas d'une variation (exagérée) de la base. On voit bien que sur la portion centrale, la pente est différente du reste de l'EPI, ce qui correspond bien au cas schématisé en (a). 94

7.4 **Effet d'une translation transversale. -** Le schéma de gauche montre deux images rectangulaires prises à des instants différents et ayant un décalage latéral t_y. La même colonne du détecteur est tracée sur chacune d'elles. Sur le dessin de droite, on retrouve ces même colonnes qui composent l'EPI résultante d'une séquence d'images dont font partie celles de gauche. L'objet rouge qu'observe la caméra n'apparaît pas sur la bonne colonne de l'image en pointillé en raison du décalage t_y. Sur l'EPI on observe que le chemin correspondant à cet objet est alors discontinu. 94

7.5 **Influence de l'orientation de l'objet sur l'erreur de disparité. -** A nouveau, on observe à droite deux images successives et avec un décalage latéral t_y. A droite est dessinée l'EPI pour la ligne épipolaire correspondant aux colonnes surlignées. Si l'objet observé, en rouge, est incliné par rapport à l'axe de vol, alors l'erreur de disparité est accrue. En effet, le point gris, au lieu d'être associé à son correspondant de l'autre image qui devrait se trouver au point noir, est associé à l'autre point de même couleur, soit le point rouge. Le chemin de l'EPI sera continu, mais sa pente sera modifiée. . . 95

7.6 **Trajectoire du vol canjuers. -** La ligne rouge correspond à la trajectoire réelle, tandis que la bleue correspond à la trajectoire attendue, qui sert de référence à la correction. Les trajectoires sont données en coordonnées planes, en mètres. 96

7.7 **Histogrammes des écarts en translation de la trajectoire réelle par rapport à la trajectoire corrigée (en mètres).** 96

7.8 **Ecarts en attitude par rapport à la valeur moyenne en vol (en radians).** 97

7.9 **Schéma de calcul de l'erreur sur la disparité δd_x. -** La reprojection du point I pour corriger la déviation δB_x selon l'axe de vol devrait se faire en J. Comme on ne connaît pas l'élévation de la scène, elle se fait toutefois en I'. 98

7.10 **(a) Schéma de calcul de l'erreur sur la position de la reprojection δr d'un point M pour une déviation transversale δB_y. -** La reprojection du point I se fait en I' au lieu de J. **(b) Schéma de la relation entre le décalage transversal de l'avion par rapport à son axe de visée δr et l'erreur sur la mesure de disparité δd_y. -** Le schéma représente l'image reprojetée. Le trait noir vertical en pointillé dessine une colonne quelconque (une ligne épipolaire). Le trait gris continu désigne la reprojection d'un objet surélevé. Le trait gris en pointillé correspond à la position correcte de l'objet. L'erreur faite sur la disparité est donc le segment rouge. 98

7.11 **(a) Erreur de reprojection pour une variation d'altitude. -** La reprojection du point I pour corriger la déviation δB_z devrait se faire en J. Comme on ne connaît pas l'élévation de la scène, elle se fait toutefois en I'. **(b) Décomposition de l'erreur due à l'altitude. -** Le schéma représente l'image reprojetée. L'erreur de disparité se décompose en une erreur selon la ligne de vol et une erreur transversale. 99

7.12 **Evolution de l'erreur de disparité. -** Chaque ligne correspond à la déviation de l'avion selon une direction en particulier, respectivement l'axe de direction de vol X, l'axe transverse Y et l'axe vertical Z. Pour les valeurs maximales de déviation données en 7.7, les courbes (a) et (e) dépendent des variations de la hauteur du point au sol h et la courbe (c) des variations de l'orientation θ pour plusieurs valeurs de h. Les courbes de la deuxième colonnes sont calculées avec $h = 20\text{m}$, $\theta = 45°$. C'est la valeur de la déviation de la base qui varie. 101

8.1 **Construction des EPI avec les images Sieleters. -** La ligne rouge en (a) correspond à une ligne épipolaire, à partir de laquelle est construite l'EPI en (b). En gris, le plan épipolaire associé. La flèche indique la direction de vol. Sur l'EPI, nous appelons les lignes que l'on observe des chemins. 104

8.2 **Gestalt. -** Les points sont agencés de telle sorte qu'ils forment un cercle. Un telle structure aurait peu de chances de se produire dans une image de bruit. Il s'agit donc d'un gestalt. 105

8.3 **Extraction de segments avec l'algorithme LSD. -** Les segments extraits sont en bleu. La direction des chemins de l'EPI est différente de celle de la figure 8.1, pour laquelle le déplacement de l'avion n'a pas été compensé. L'utilisation de l'un ou l'autre est indifférente. 106

8.4 **Carte éparse obtenue par EPI pour la scène CFM. -** La carte, en couleur, désigne les hauteurs relatives en mètres. Elle est superposée à l'image Sieleters correspondante (en gris) pour faciliter l'interprétation. 107

8.5 **Carte éparse obtenue par EPI pour la scène Canjuers. -** La carte, en couleur, désigne les hauteurs relatives en mètres. Elle est superposée à l'image Sieleters correspondante (en gris) pour faciliter l'interprétation. En haut à gauche en rouge foncé, désigné par une flèche, les valeurs de hauteurs anormalement élevées sont causées par un véhicule qui se déplace au cours de l'acquisition. 108

8.6 **Illustration du manque d'information sur les bords verticaux. -** L'élément central est un extrait de la carte éparse en 8.4, entourée de la moitié de l'EPI des colonnes directement à gauche et à droite du trait rouge. La carte éparse est élargie afin de la rendre identifiable. 108

9.1 **Graphe. -** Les liens entre noeuds sont à la fois arêtes jaunes gérant les relations de voisinage, et les liens rouges et bleus correspondants à des propriétés des noeuds. Le coût de la coupure de graphe en vert est la somme des liens qu'elle brise. p et q sont deux noeuds quelconques différents de la source et du puits. Source : [BV06] 112

9.2 **(a) Résultat de la minimisation sans régularisation sur image CFM. -** Le résultat est dicté par le terme d'attache aux données seul. Les pixels verts sont des pixels occultés. **(b) Résultat avec tous les termes d'énergie.** Les pixels occultés sont en jaune. Les échelles sont en mètres. 116

9.3 **Deux cas de gestion des ombres. -** On montre ici le résultat du graphcut en niveaux de gris sur lequel est superposée en couleur la carte éparse. Les vignettes correspondent aux zones dans les rectangles de la figure 9.2(b). (a) En général, sur les arbres isolés, l'algorithme parvient à séparer le feuillage et son ombre, en suivant les contours donnés par la carte éparse. (b) Entre deux alignements de disparité différente, l'algorithme a des difficultés dans les ombres, où le terme d'attache aux données est trop faible pour être déterminant. 117

9.4 **Toit de la recette. -** On montre ici le résultat du graphcut en niveaux de gris sur lequel est superposée en couleur la carte éparse. 117

9.5 **Résultat du *graphcut* pour Canjuers. -** Les pixels occultés sont en bleu foncé. L'échelle est en mètres. 118

9.6 **Interférogrammes sur un bâtiment. -** L'interférogramme rouge a été calculé après recalage avec un MNE plat. On observe qu'il est incorret par les déviations de la valeur moyenne. Celui en vert est issu d'images recalées avec un MNE obtenu par les traitements présentés dans cette thèse. 119

Correction des effets de relief en spectro-imagerie aéroportée

Clara BARBANSON

RESUME : Cette thèse présente une série de traitements pour des spectro-imageurs aéroportés à trame défilante. Ces instruments acquièrent des séquences denses d'images qui forment un cube hyperspectral. Toutefois, en présence de relief sur la scène, l'erreur de parallaxe entre les différentes images de la séquence génère des erreurs dans le cube si les images ne sont pas corrigées avec un modèle numérique d'élévation (MNE). Ce dernier étant en général inconnu, on peut se servir du cube hyperspectral, doté de nombreuses images de la scène, à des fins de stéréoscopie.

L'objectif de cette thèse est donc de calculer le MNE de la scène survolée à partir des images de l'instrument. Cette tâche présente deux principales difficultés. La première concerne les disparités observées entre deux clichés successifs, qui sont inférieures au centième de pixel. La seconde concerne l'instrument auquel sont appliqués ces traitements. Il s'agit d'un spectro-imageur à transformée de Fourier statique, qui restitue sur chaque image une figure d'interférences fixe.

Nous proposons donc d'abord de décomposer l'image afin d'enlever les franges d'interférence, qui perturbent les traitements stéréoscopiques. Pour cela deux modèles de décomposition sont traités : l'un additif, et l'autre multiplicatif. Nous utilisons ensuite les images des plans épipolaires (EPI) pour obtenir une carte de disparité éparse. Cette méthode nécessite que tous les sommets de prise de vue soient colinéaires, ce qui n'est pas tout à fait vérifié en imagerie aéroportée. Nous étudions donc l'impact de ces erreurs de trajectoire sur la mesure de disparité. Enfin, pour obtenir un modèle numérique de surface continu, nous densifions la carte éparse par une minimisation d'énergie par graphcut adaptée au cas multi-vues et sous-pixellique.

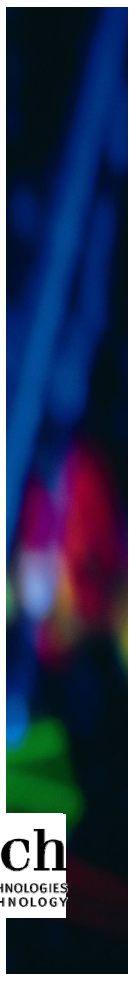
MOTS-CLEFS : stéréoscopie – décomposition d'image – optimisation biconvexe – imageur hyperspectral – multi-vues – navigation

ABSTRACT : We present a pipeline of algorithms applied to the images of an imaging static Fourier transform spectrometer (ISFTS), and more generally to windowing hyperspectral instruments. Those instruments measure hyperspectral cubes that can become flawed if the induced parallax error between the images is not corrected using a digital elevation model (DEM). The latter is generally not known. We surmise that the many images of the interferometric cube could be used in order to compute the DEM.

What we aim to achieve in this thesis is therefore to compute the DEM of a given scene from the images of an ISFTS. We are faced with several difficulties. Firstly, the instrument provides images of the scene on which a fixed interference figure is superimposed. Secondly, the observed disparities between two consecutive images of the sequence are less than a hundredth of a pixel, which calls for sharp subpixel processing.

We tackle the problem of interference fringes by ridding ourselves of them through image decomposition. To this purpose we propose two models : an additive model and a multiplicative one. In order to take advantage of the redundancy of information on the images and handle subpixel disparities, we use the epipolar plane images to obtain a precise sparse disparity map. One important assumption to be able to use this method is that all camera centers should be aligned. This is not a given in airborne acquisitions, and so we study the influence of trajectory fluctuations on the disparity measure. Finally, in order to obtain a dense height map, we densify the sparse values by minimising a functional using graphcuts.

KEY-WORDS : stereoscopy – image decomposition – biconvex optimisation – hyperspectral imager – multi-view – navigation



TELECOM ParisTech

ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY