HAL open science

Modélisation et calcul du flot de scène stéréoscopique par une méthode variationnelle

Frédéric Huguet

To cite this version:

Frédéric Huguet. Modélisation et calcul du flot de scène stéréoscopique par une méthode variationnelle. Modélisation et simulation. Université Joseph-Fourier - Grenoble I, 2009. Français. NNT : . tel-00421958

HAL Id: tel-00421958

https://theses.hal.science/tel-00421958

Submitted on 5 Oct 2009

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.

Université Joseph Fourier - Grenoble I

N° attribué par la bibliothèque

THÈSE

pour obtenir le grade de

Docteur de l'Université Joseph Fourier

Spécialité : **Mathématiques Appliquées**

préparée au laboratoire **INRIA Rhône-Alpes**

dans le cadre de l'École Doctorale **Mathématiques, Sciences et Technologies de l'Information, Informatique**

présentée et soutenue publiquement
par

Frédéric Huguet

le 30 avril 2009

Titre:
Modélisation et calcul du flot de scène stéréoscopique par une méthode variationnelle

Directeur de thèse: **Eric Bonnetier**
Coencadrant de thèse: **Frédéric Devernay**

Jury

M. Roger Mohr,	Président du jury
M. Rachid Deriche,	Rapporteur
M. Renaud Keriven,	Rapporteur
M. Stéphane Bouissou,	Examinateur
M. Eric Bonnetier,	Directeur de thèse
M. Frédéric Devernay,	Coencadrant de thèse

Avant-propos

La thèse fut pour moi l'occasion de découvrir le monde de la recherche et ses spécificités. Ce fut extrêmement enrichissant sur plusieurs points. Tout d'abord, ce fut l'apprentissage de l'autonomie dans le travail. Même si mes directeurs de thèse furent très présents et m'ont souvent aiguillés sur la bonne voie, j'ai pu apprendre à prendre des initiatives et à essayer de développer mes idées. Je remercie donc dans un premier temps Frédéric Devernay pour sa disponibilité et sa présence au quotidien, ainsi que pour ses idées et ses conseils méthodologiques dans le domaine du développement logiciel et l'algorithmique. Je tiens aussi à remercier mon directeur de thèse, Eric Bonnetier, pour sa gentillesse et sa disponibilité afin de discuter des aspects théoriques du sujet. Ensuite, la thèse m'a permis d'engager de nombreuses discussions intéressantes avec de nombreux autres thésards et chercheurs. Je voudrais à ce titre remercier l'ensemble de l'équipe PRIMA de l'Inria Grenoble. Enfin, la thèse s'est déroulée dans un environnement agréable, avec quelques amis thésards que j'ai fréquenté avec beaucoup de plaisir, je pense à Amaury et Elie notamment. Je tiens aussi à remercier mes parents pour leur soutien tout au long de la thèse et pour avoir préparé le meilleur pot de thèse que j'aie jamais goûté !

Résumé

En vision par ordinateur, le flot de scène représente le déplacement des points d'une surface située dans une scène 3D quelconque, entre deux instants consécutifs. Il s'agit donc d'un champ vectoriel 3D. Celui-ci est particulièrement utile dans l'analyse des déformations d'une surface quelconque, observée par un système d'au moins deux caméras. Cette thèse traite de l'estimation du flot de scène et d'une application dans le domaine de la géophysique. Elle s'est déroulée dans le cadre de l'ACI GEOLSTEREO, en collaboration étroite avec le laboratoire Geosciences Azur, situé à Sophia Antipolis (06, UMR 6526 - CNRS - UNSA - UPMC- IRD).

Nous proposons d'estimer le flot de scène en couplant l'évaluation du flot optique dans les séquences d'images associées à chaque caméra, à l'estimation de la correspondance stéréo dense entre les images. De plus, notre approche évalue, en même temps que le flot de scène, les occultations à la fois en flot optique et en stéréo. Nous obtenons au final un système d'EDP couplant le flot optique et la stéréo, que nous résolvons numériquement à l'aide d'un algorithme multirésolution original. Alors que les précédentes méthodes variationnelles estimaient la reconstrution 3D au temps t et le flot de scène séparément, notre méthode estime les deux simultanément. Nous présentons des résultats numériques sur des séquences synthétiques avec leur vérité terrain, et nous comparons également la précision du flot de scène projeté dans une caméra avec une méthode récente et performante d'estimation variationnelle du flot optique. Des résultats sont présentés sur une séquence stéréo réelle, se rapportant à un mouvement non rigide et à de larges discontinuités.

Enfin, nous présentons l'approche originale de modélisation physique 3D utilisée au laboratoire Geosciences Azur. Nous décrivons la mise en place du dispositif stéréoscopique associé, ainsi que le déroulement de l'expérience. Des résultats de reconstruction 3D, d'estimation du flot de scène, et de suivi de la déformation d'une surface sont montrés dans le chapitre 4 de la thèse.

Abstract

The scene flow is the displacement vector of any surface points estimated between two consecutive moments. Mathematically it is a three-dimensionnal vector field. This one is useful when any surface temporal deformation has to be studied, using two or more cameras. This thesis handles the scene flow computation and shows the use of this one for a geophysical project. In this aim, we worked with the geophysics sciences laboratory named Geosciencez Azur, which is located in Sophia Antipolis (Alpes Maritimes, UMR 6526 - CNRS - UNSA - UPMC- IRD).

This paper presents a method for scene flow estimation from a calibrated stereo image sequence. The scene flow contains the 3-D displacement field of scene points, so that the 2-D optical flow can be seen as a projection of the scene flow onto the images. We propose to recover the scene flow by coupling the optical flow estimation in both cameras with dense stereo matching between the images, thus reducing the number of unknowns per image point. Moreover our approach handles occlusions both for the optical flow and the stereo. We obtain a partial differential equations system coupling both the optical flow and the stereo, which is numerically solved using an original multi-resolution algorithm. Whereas previous variational methods were estimating the 3-D reconstruction at time t and the scene flow separately, our method jointly estimates both. We present numerical results on synthetic data with ground truth information, and we also compare the accuracy of the scene flow projected in one camera with a state-of-the-art single-camera optical flow computation method. Results are also presented on a real stereo sequence with large motion and stereo discontinuities.

We finally present the original approach developed in Geosciences Azur to study the gravitary mountain landslides, the 3D physical modelling. We describe the experimental stereo device used to track the deformations of the reduced moutain model used by the geophysicists. 3D reconstruction and scene flow results are shown, as well as the tracking of the observed surface deformations, in the fourth chapter of the thesis.

Table des matières

Résumé **5**

1 Introduction **11**
1 Définitions 12
2 Organisation de la thèse 16

2 Flot optique et stéréoscopie : principes, algorithmes **19**
1 Principes 19
1.1 Stéréoscopie et flot optique : des problèmes de mise en correspondance 19
2 La stéréoscopie 20
3 Le flot optique 24
4 Quelques algorithmes de stéréoscopie et de flot optique par minimisation d'énergie 26
4.1 Mise en correspondance stéréoscopique 26
4.2 Calcul du flot optique 31
4.3 Exemples de séquences de test 32
4.4 Occultations : quelques méthodes existantes 33
5 Quelques méthodes d'estimation du flot de scène 35
6 Notre stratégie d'évaluation du flot de scène 36

3 Modélisation du problème **39**
1 Hypothèses 39
1.1 Les hypothèses sur la scène 3D observée 39
1.2 Les hypothèses de régularité du flot de scène stéréoscopique . 41
2 Modélisation du problème 42
2.1 Notations 43
2.2 Conception d'une méthode variationnelle 45
3 Le problème des occultations 56
3.1 Quelques méthodes existantes 57
3.2 Estimation des occultations 57
4 Forme finale de l'énergie 62

4.1 Les équations d'Euler 63
5 Conclusion 64

4 Algorithme et résultats numériques 67
1 Approches multirésolution 67
2 Résolution numérique des EDP 70
2.1 Principes 70
2.2 Discrétisation des équations d'Euler Lagrange 74
3 Algorithme 76
3.1 Schéma général 76
4 Résultats numériques 78
4.1 Exemples synthétiques 79
4.2 Exemples réels 81
4.3 Evaluation quantitative 84
5 Améliorations possibles 86
5.1 Une modélisation cyclopéenne 87
5.2 Une résolution numérique optimisée 89
5.3 Parallélisme envisageable 92
6 Perspectives : une base de données de validation pour le flot de scène ? 93
7 Conclusion 94

5 Application à la géophysique 95
1 L'étude des mouvements gravitaires : GEOLSTEREO 96
1.1 Les mouvements de terrain gravitaires 96
1.2 Classification des approches numériques disponibles 97
1.3 La modélisation physique 3D 100
2 L'expérience 102
2.1 Dispositif géophysique 102
2.2 Dispositif stéréo 103
2.3 Protocole de prises de vue 103
2.4 Déroulement 106
2.5 Interprétation des résultats 107
3 Estimation du flot de scène 109
3.1 Séquences prétraitées 109
3.2 Évolution des composantes du flot de scène 109
3.3 Évolution des cartes d'occultation 110
3.4 Evolution des gradients de déplacements 122
3.5 Reconstruction 3D 122
3.6 Flot de scène 130
4 Conclusion 131

6 Conclusion 135

Bibliographie 137

A Calibration, rectification, stéréoscopie et reconstruction 3D 145
1 Calibrage fort 145
2 Calibration faible 148
3 Estimation numérique des paramètres des caméras 149
4 Rectification 151
4.1 Principes 151
4.2 Algorithme 153
4.3 Résultats 153
5 Stéréoscopie : utilisation de la disparité 154
5.1 Reconstruction 3D 154
5.2 Propriétés différentielles des surfaces et disparité 155

B Développements mathématiques 157
1 Discrétisation des équations d'Euler-Lagrange 157
1.1 Linéarisation des équations d'Euler Lagrange 159

C Quelques mots sur le code source 165
1 À propos du codage (teinte, saturation) des résultats 166

Chapitre 1

Introduction

La Vision par Ordinateur est un domaine de l'informatique balayant un large spectre d'applications. De nombreuses thématiques sont regroupées sous ce terme, par exemple :

- La reconstruction 3D de scènes à partir d'un système multi-caméras (2 caméras ou plus).
- L'analyse du mouvement d'objets dans des images (flot optique) ou dans une scène 3D.
- L'analyse simultanée du mouvement de caméras et d'objets d'une scène (motion-stereo), utile en robotique.

La plupart de ces thématiques ont été étudiées dans le passé et font donc l'objet d'une littérature abondante. La thèse présentée ici s'articule autour du flot de scène, qui constitue un domaine moins étudié et moins mature de la vision par ordinateur. Et plus particulièrement, nous nous intéressons au flot de scène stéréoscopique. En d'autres termes, il s'agit ici d' étudier et d'évaluer le déplacement des objets présents dans une scène 3D, à partir de séquences d'images stéréoscopiques. L'une des raisons qui explique le moindre intérêt pour le flot de scène est le fait que l'étude du mouvement a essentiellement été réalisée à partir de séquences d'images issues d'une seule caméra, et que la reconstruction 3D s'est restreinte à la stéréoscopie. D'autre part, dans le cas de l'étude du mouvement 3D, la difficulté à évaluer quantitativement les algorithmes à partir de bases de données de test peut être rédhibitoire. Il n'existe en effet pour le moment à notre connaissance aucun équivalent de la base de données Middlebury utilisée pour le flot optique et la stéréoscopie pour l'étude du flot de scène.

Cette thèse s'adresse tout d'abord aux étudiants et aux chercheurs en vision par ordinateur, mais peut également être lue par des étudiants et des chercheurs en mathématiques appliquées intéressés par les applications des Équations aux Dérivées Partielles (EDP) en traitement de l'image. De manière générale les prérequis nécessaires à une bonne compréhension de la thèse nécessitent une connaissance de la vision par ordinateur et de la résolution numérique d'équations aux dérivées partielles. Néanmoins nous présentons dans cette thèse une application originale

aux géosciences, la partie du manuscrit consacrée à cette application s'adresse donc également aux membres de la communauté des géosciences, notamment ceux s'intéressant aux déformations de terrains (glissements de terrain, séismes ...).

Nous présentons dans cette thèse une modélisation mathématique du problème du flot de scène stéréoscopique, ainsi qu'un algorithme pour résoudre les équations en découlant. Par la suite, nous présentons des résultats numériques obtenus sur des séquences synthétiques aussi bien que réelles afin de valider l'algorithme. C'est également l'occasion de présenter des résultats qualitatifs en se comparant à certaines méthodes de l'état de l'art. Enfin, nous présentons une application originale au domaine des géosciences, consistant à utiliser le flot de scène comme un outil de mesure des déformations d'une surface soumise à des contraintes physiques.

1 Définitions

Nous présentons ici les définitions et notions utilisées dans la suite de la thèse.

Calibrage Le calibrage est le processus d'estimation des paramètres d'une ou plusieurs caméras. Les paramètres estimés concernent à la fois les caractéristiques propres aux caméras (paramètres intrinsèques : focale, coordonnées du centre optique ..) et leurs positionnement dans l'espace (paramètres extrinsèques : matrice de rotation/translation par rapport à un repère donné). Dans toute la thèse, nous supposons disposer d'un système de caméras calibrées, dont les paramètres sont estimés à l'aide d'outils d'optimisation numérique (Matlab).

Stéréoscopie La stéréoscopie désigne dans toute la suite de la thèse le principe de mise en correspondance de points d'une paire d'images prises par un système de deux caméras observant la même scène. Cet appariement de points, s'il est possible, conduisant à estimer les coordonnées 3D du point de la scène correspondant, par triangulation, dans le cas d'un système de caméras calibrées.

Système stéréoscopique Un système stéréoscopique désigne un système de deux caméras associées pour observer une même scène 3D. Un exemple d'un tel système est illustré par l'image 1.1.

Séquences d'images stéréoscopiques Une séquence d'images stéréoscopiques désigne deux séquences d'images synchronisées issues d'un système stéréoscopique. On peut voir cette séquence comme une suite chronologiquement ordonnée de paires d'images stéréoscopique (la synchronisation nécessitant que les images d'une même paire soient prises rigoureusement en même temps).

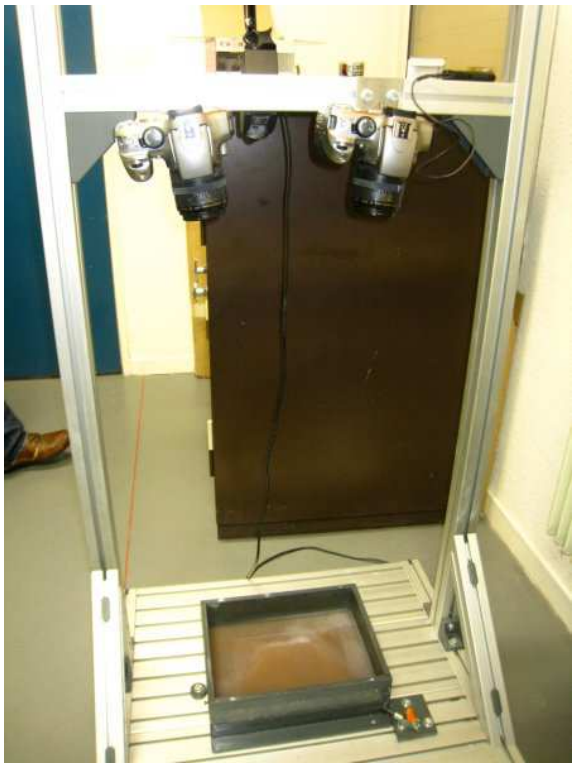

FIGURE 1.1 – Système stéréoscopique composé d'une paire d'appareils photos numériques observant un même objet. Dans ce cas les appareils ne sont pas parallèles, et la baseline correspond simplement à l'écart entre les deux appareils photos mesuré sur la barre métallique leur servant de support.

Baseline La baseline désigne la distance entre les centres optiques de caméras formant un système stéréoscopique. Sur l'image 1.1, la baseline est l'écart entre les appareils photos, mesuré sur la barre metallique leur servant de support.

Disparité Etant donné un point M d'une scène 3D quelconque observée par un système stéréoscopique, la disparité désigne l'écart, mesuré en pixels, entre les projetés de M dans les plans images des deux caméras. Elle est, pour certaines configurations de caméras, inversement proportionnelle à la profondeur Z du point M.

Carte de disparité Soit une paire d'images stéréoscopiques I_g et I_d. On choisit de définir la disparité dans le système de coordonnées attaché à une des deux images, ici par exemple, I_g, que l'on note Ω_g. Formellement, la disparité est une fonction $d : (x, y) \in \Omega_g \rightarrow d(x, y) \in \Re$, et la carte de disparité associée est la carte composée des valeurs de d en chaque pixel de I_g. Un exemple de carte de disparité est illustré par les images 1.5.

FIGURE 1.2 – L'exemple Art issu de la base de données Middlebury, disponible à l'adresse `http://vision.middlebury.edu/stereo/data/`. À gauche et au milieu : une paire d'images stéréoscopique. À droite : la carte de disparité correspondante.

Flot optique Etant donnée une séquence d'images ordonnée chronologiquement (par exemple issue d'une vidéo), et réprésentant une même scène, le flot optique désigne le mouvement apparent des objets de la scène dans les images. Dans de nombreux cas, ce mouvement apparent se confond avec la projection du mouvement 3D réel, mais ce n'est par exemple pas le cas dans les scènes ou l'illumination varie de manière sensible. L'illustration 1.3 montre un exemple de flot optique sur une séquence particulièrement utilisée pour tester les algorithmes par le passé, la séquence Yosemite.

FIGURE 1.3 – La séquence de test Yosemite disponible à l'adresse `http://www.cs.brown.edu/~black/images.html`. À gauche et au milieu : une paire d'images extraites de la séquence. À droite : le flot optique estimé par la méthode de B.Horn and B.Schunck [1981].

Flot de scène Le flot de scène est le champ vectoriel 3D définissant le déplacement des objets dans une scène 3D. Il s'agit en quelque sorte de l'extension 3D du flot optique. L'illustration 1.4 représente le flot de scène estimé sur un visage en mouvement.

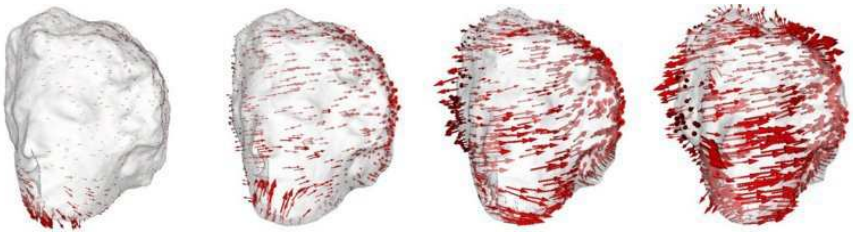

FIGURE 1.4 – Flot de scène sur la séquence Yannis par la méthode de Pons et al. [2005]

Flot de scène stéréoscopique Le flot de scène stéréoscopique est le flot de scène estimé à partir de séquences d'images stéréoscopiques fournies par un système de caméras.

Occultation Un objet présent dans une scène donnée est dit partiellement ou totalement occulté lorsque une partie ou l'ensemble de cet objet n'est pas visible par l'observateur. En général dans une scène dynamique les occultations apparaissent lorsque des objets passent les uns devant les autres.

Warp et warping Beaucoup de problèmes en vision se posent sous la forme d'un problème de mise en correspondance entre deux pixels m_1 et m_2 appartenant respectivement à une image I_1 et une image I_2, en général tel que $I_1(m_1) = I_2(m_2)$. Le warp W est la transformation définie entre I_1 et I_2, qui à m_1 fait correspondre m_2. Dans de nombreux problèmes, W peut ne pas être définie localement (présence d'occultations), ou bien être une fonction à valeurs multiples (présence d'ambiguités). Néanmoins, on peut définir un sous ensemble de warps admissibles lesquels admettent les qualités suivantes :

- Un warp admissible doit être défini en tous points de l'image source I_1.
- Il doit être continu et dérivable.
- Règle de composition : un warp suivi par un second warp reste un warp admissible.
- L'inverse d'un warp admissible est aussi un warp admissible.

Les warps définis de cette manière sont des difféomorphismes, c'est à dire des transformations différentiables et inversibles pour lesquelles les inverses sont également différentiables.

Vérité terrain La vérité terrain désigne le résultat exact à atteindre pour un algorithme donné sur des séquences d'images réelles. Dans le cas du flot optique, la vérité terrain est ainsi une carte du champ de déplacement réel, auxquels les résultats des algorithmes doivent se comparer afin de mesurer leur fiabilité.

FIGURE 1.5 – L'exemple Dimetrodon issu de la base de données Middlebury, disponible à l'adresse `http://vision.middlebury.edu/flow/data/`. À gauche et au milieu : une paire d'images issues d'une séquence vidéo. À droite : la vérité terrain du flot optique, encodée en format (teinte, saturation) par Baker et al. [2007]. Le lecteur se réfèrera à l'Annexe C pour plus de détails à ce sujet.

Scène lambertienne Une scène 3D est dite lambertienne lorsque l'intensité lumineuse des points de la scène projetés dans les plans images de caméras observant la scène ne varie pas avec le point de vue. En particulier, les surfaces des objets présents dans une telle scène ne comportent pas de spécularités.

Scène fronto-parallèle Une scène 3D est dite fronto-parallèle lorsque les surfaces des objets présents dans la scène sont parallèles aux plans images des caméras.

Image cyclopéenne Une image cyclopéenne est une image virtuelle issue de l'interpolation d'images d'une paire stéréoscopique. Elle représente une image observée par une caméra virtuelle située à mi-distance entre les deux caméras d'un système stéréoscopique.Ceci est illustré par la figure 1.6.

2 Organisation de la thèse

Dans un premier temps, dans le chapitre 2, nous proposons une modélisation du problème d'estimation du flot de scène stéréoscopique à partir de la connaissance du flot optique et de la mise en correspondance stéréoscopique d'une scène 3D donnée. Après avoir exposés les principes généraux d'estimation de ces éléments, et après avoir fait un rapide point sur les grands types de méthodes existantes pour évaluer ceux ci, nous présentons une formulation mathématique du problème. Nous avons choisi pour résoudre ce dernier une méthode variationnelle consistant à minimiser une énergie. Cette énergie est construite de telle manière qu'elle prend en compte le maximum d'informations issues de la modélisation et diverses contraintes liées à la scène 3D observée, en particulier l'apparition et l'évolution d'occultations et les changements d'éclairage dans la scène. Nous détaillons la construction d'une telle énergie ainsi que la manière de la minimiser. En effet, nous montrons par construction que le flot de scène à estimer est le minimum global, s'il existe, de cette énergie.

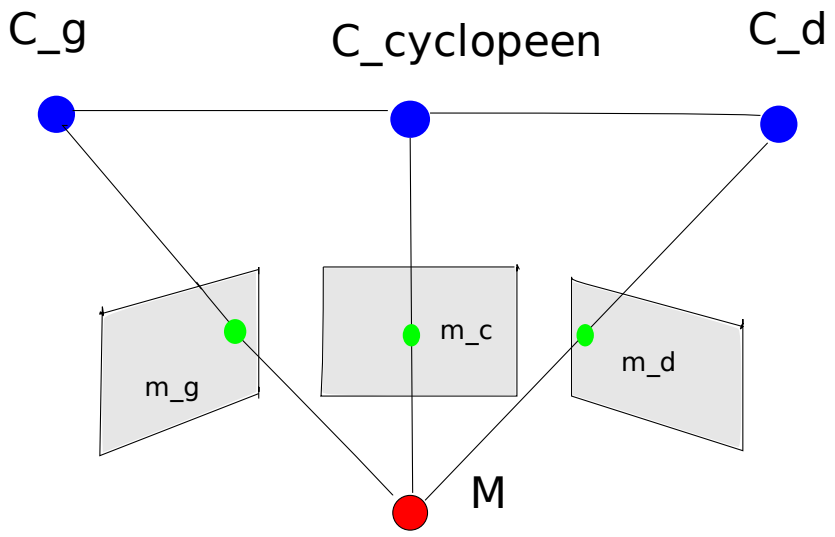

FIGURE 1.6 – Un point M d'une scène observé par deux caméras de centres optiques C_g et C_d. La caméra cyclopéenne a son centre situé au milieu du segment joignant C_g et C_d, et l'image virtuelle produite (au milieu) est déduite par interpolation entre les images prises par les caméras gauche et droite.

Une fois la modélisation effectuée, nous avons cherché à résoudre numériquement les équations découlant du processus de minimisation de l'énergie modélisant notre problème. Nous énoncons dans le chapitre 3 les difficultés numériques que posent ces équations, ainsi que l'approche adoptée pour les traiter. Nous présentons ensuite la structure de l'algorithme, les contraintes qui ont aiguillé nos choix, ainsi que des résultats numériques qualitatifs et quantitatifs sur des séquences de synthèse mais aussi sur des séquences réelles. Enfin, nous donnons quelques pistes d'amélioration du code, notamment du point de vue du temps de calcul.

Le chapitre 4 présente une application originale du flot de scène à la géophysique. Cette application est issue d'une collaboration avec le laboratoire Geosciences Azur, situé à Sophia Antipolis, et a pour but d'analyser les déformations d'une surface au cours d'expériences physiques. Le flot de scène estimé permet alors de retrouver les déplacements de la surfaces au cours du temps, et ses discontinuités permettent de localiser les fractures se formant sur la surface. Le protocole expérimental et des résultats numériques sur les séquences stéréo obtenus à Geosciences Azur sont présentés.

Chapitre 2

Flot optique et stéréoscopie : principes, algorithmes

Ce chapitre explique les principes de base des techniques utilisées par la suite pour estimer le flot de scène. Connaissant ces bases, nous pourrons alors présenter la stratégie envisagée pour estimer le flot de scène.

Nous dressons ensuite un état de l'art des méthodes potentiellement utiles pour notre problème : flot optique et stéréocopie essentiellement, ainsi que la combinaison des deux, usuellement appellée "motion stereo" dans la littérature (notamment en robotique). Nous verrons quelles sont les grandes catégories de méthodes, leurs avantages et leurs inconvénients. Cet état de l'art n'a pas pour vocation d'être complètement exhaustif, étant donné le très grand nombre d'articles parus dans ces sous-domaines de la vision par ordinateur que représentent l'étude du flot optique et celle de la stéréoscopie. D'autre part, nous verrons par la suite qu'il est extrêmement important de prendre en compte les endroits d'une scène donnée où l'on ne peut évaluer le flot optique ou réaliser une mise en correspondance stéréoscopique : les occultations. Nous donnerons quelques références récentes et importantes au sujet de méthodes d'estimation de ces dernières.

Enfin, le flot de scène étant un sujet d'étude très jeune en vision par ordinateur, nous examinerons les quelques méthodes d'intérêt parues dans la littérature depuis quelques années.

1 Principes

1.1 Stéréoscopie et flot optique : des problèmes de mise en correspondance

Le calcul du flot optique consiste à évaluer le déplacement de points dans une séquence d'images au cours du temps. L'évaluation du flot optique consiste à mettre en correspondance un point entre deux images successives, correspondant au même point de la scène 3D se déplacant dans une séquence. En règle générale, il s'évalue

entre deux images consécutives (et donc 2 instants de prise d'image successifs), et non entre une image de départ ($t = 0$) et une image prise à un instant suivant t quelconque. Un exemple concret de cette approche consiste en des séquences d'images issues de caméras embarquées dans une automobile, lesquelles illustrent des scènes variant rapidement, comme le montre la figure 2.1.

Dans ce cas la, le flot optique ne peut etre évalué entre l'instant initial de la prise de vue et un instant quelconque puisqu'on ne peut rapidement plus mettre assez de points en correspondance pour obtenir un flot optique dense.

La stéréoscopie quand à elle, consiste à mettre en correspondance des paires de points entre les images d'une même paire stéréoscopique, correspondant au même point de la scène 3D. Lorsque cela n'est pas possible (dans le cas ou certains points de la scène sont visibles dans une caméra mais pas dans l'autre), il s'agit d'une occultation et on ne peut pas retrouver la profondeur Z de la scène à cet endroit comme l'illustre la figure 2.2.

Flot optique et stéréo sont donc deux problèmes de mise en correspondance de points entre deux images, avec parfois l'impossibilité de réaliser ces appariements de points, dans un sens ils sont donc similaires, mais nous verrons, et cela a amené quelques difficultés numériques par la suite, qu'ils sont très différents par certains aspects.

2 La stéréoscopie

Avant d'aborder la problématique exposée dans la suite de la thèse, il est important d'exposer les principes généraux de la stéréoscopie.

Comme illustré sur la figure 2.3, nous disposons d'un système de deux caméras calibrées (dont on connaît les caractéristiques internes ainsi que la position dans l'espace), observant une scène 3D donnée. Nous faisons l'hypothèse que les caméras sont disposées de manière à ce qu'une majorité de points de la scène soient visibles dans les deux caméras. Cela nécessite une orientation appropriée ainsi qu'une baseline raisonnable. Supposons qu'un point M de la scène soit visible dans les deux caméras. Celui-ci se projette dans les plans images des deux caméras, chaque projeté correspondant à un point m_1 et m_2 dans les images gauche I_g et droite I_d. Notons C_1 et C_2 les centres optiques des caméras gauche et droite, respectivement. Suppons que l'on connaisse m_1 et que l'on cherche le pixel m_2 correspondant au même point 3D M. Il se trouve que par construction, ce pixel se situe obligatoirement sur la droite intersection du plan image de la caméra droite et du plan formé par C_1, C_2 et m_1. Cette droite est nommée droite épipolaire. A priori, la recherche du correspondant de m_1 est donc une recherche le long de cette droite.

FIGURE 2.1 – Un exemple de séquence stéréo automobile, avec disparition d'éléments présents dans les images initiales. La séquence est fournie par le projet LOVe (`http://love.univ-bpclermont.fr/`). Le flot optique ne peut être estimé entre la première paire d'image et une paire quelconque suivante car l'ensemble des zones où le flot optique est défini se rétrecit au fur et à mesure de l'avancement temporel dans les séquences. Pour cette raison le flot optique est calculé entre deux images consécutives.

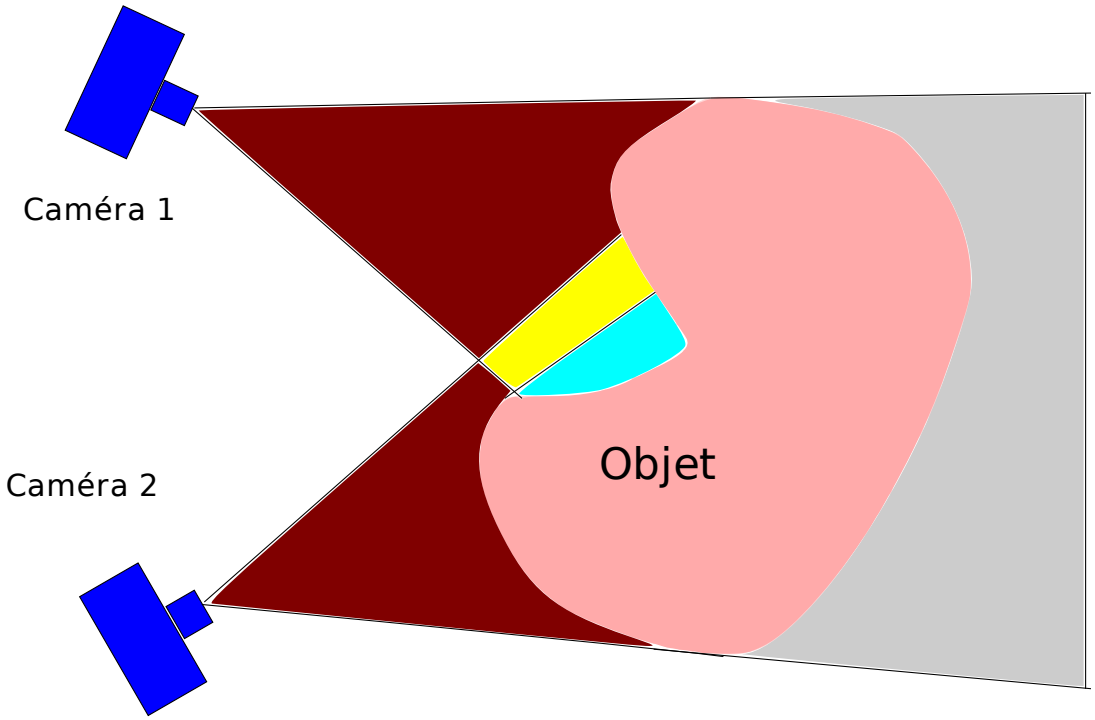

FIGURE 2.2 – Une surface quelconque observée par deux caméras. En bordeaux : les cônes de visibilité des caméras. En jaune : zone visible depuis les 2 caméras. En bleu : une zone visible pour la caméra 1 mais occultée pour la caméra 2. En gris : zone occultée par l'objet pour les 2 caméras.

La rectification

Afin de simplifier ce problème de mise en correspondance, on effectue une transformation homographique. Comme illustré en figure 2.4, il s'agit de projeter les plans images des caméras sur un plan parallèle à la droite joignant les centres optiques de ces dernières. Une fois cette transformation accomplie, les droites épipolaires sont horizontales, ce qui signifie que les pixels m_g et m_d ont la meme ordonnée. Ainsi, la mise en correspondance stéréoscopique se réduit à une recherche monodimensionnelle le long d'une droite horizontale.

Reconstruction 3D

Une fois les images rectifiées, la mise en correspondance stéréo s'effectue donc en parcourant une des deux images, puis en cherchant pour chaque pixel (x, y) le pixel correspondant dans l'autre image sur la droite horizontale d'ordonnée y. Notons (x_g, y), (x_d, y) les coordonnées de pixels correspondant au meme point 3D $M(x_m, y_m, z_m)$. On introduit la disparité d telle que $d = x_d - x_g$, mesurant l'ecart des abscisses des points mis en correspondance entre les images d'une même paire

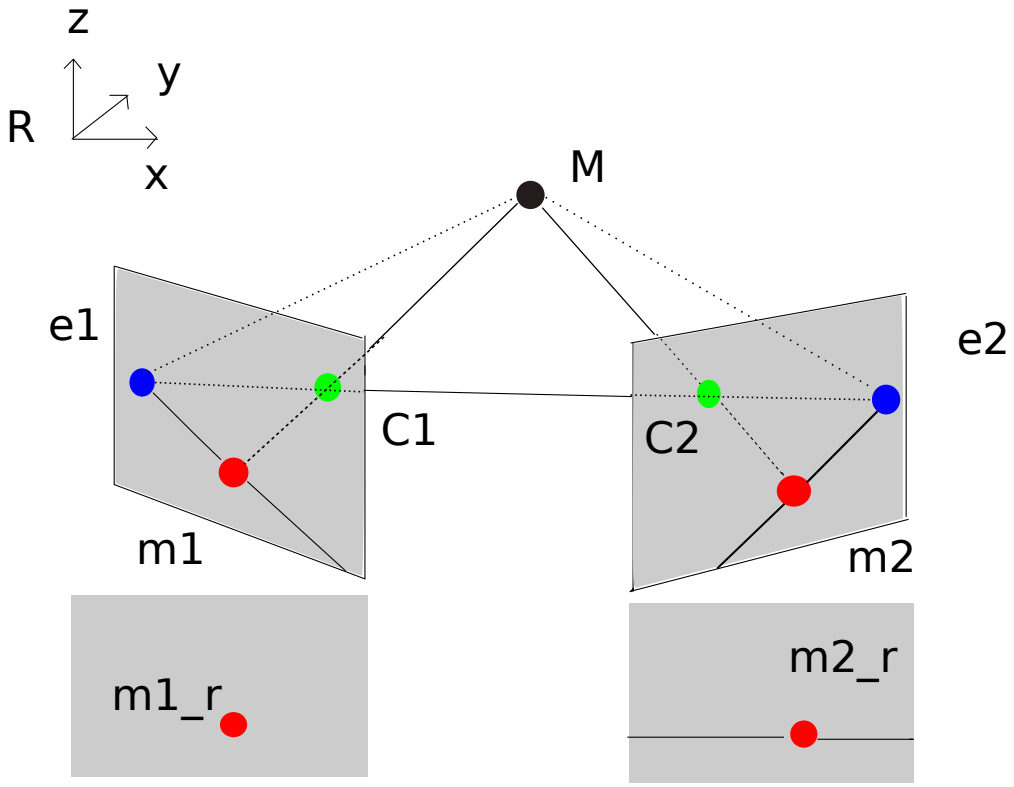

FIGURE 2.3 – Géométrie épipolaire d'un système de deux caméras.C_1 et C_2 désignent les centres optiques des caméras, M un point de la scène 3D, m_1 et m_2 ses projetés dans les plans images de chaque caméra, e_1 et e_2 les épipoles respectifs associés à chaque caméra. Enfin, les droites épipolaires correspondant à m_1 et m_2 sont indiquées.

stéréo. On peut démontrer que cette fonction peut sous certaines conditions être mise sous la forme f/z_m où f est un facteur dépendant des paramètres du système de caméras. Ainsi, si le système de caméras est calibré, la disparité nous donne accès à la profondeur de la scène et va permettre d'effectuer une reconstruction 3D (voir le travail de Devernay [1997]). Si on ne connaît pas les paramètres des caméras, il faut alors se contenter de la disparité, qui est déjà en soit une reconstruction projective de la scène.
Plusieurs situations peuvent empêcher la mise en correspondance pour un pixel donné :

- Présence d'une occulation : ce pixel est visible pour une caméra mais pas par l'autre.
- Présence d'une ambiguité : l'existence d'un motif répétitif (texture périodique ...) ne permet pas de faire un choix parmi différents pixels candidats. Ou bien au contraire, absence d'une texture discriminante : une zone trop homogène ne permet pas de trouver un correspondant sans ambiguité.
- La scène n'est pas lambertienne où il existe une spécularité : impossibilité

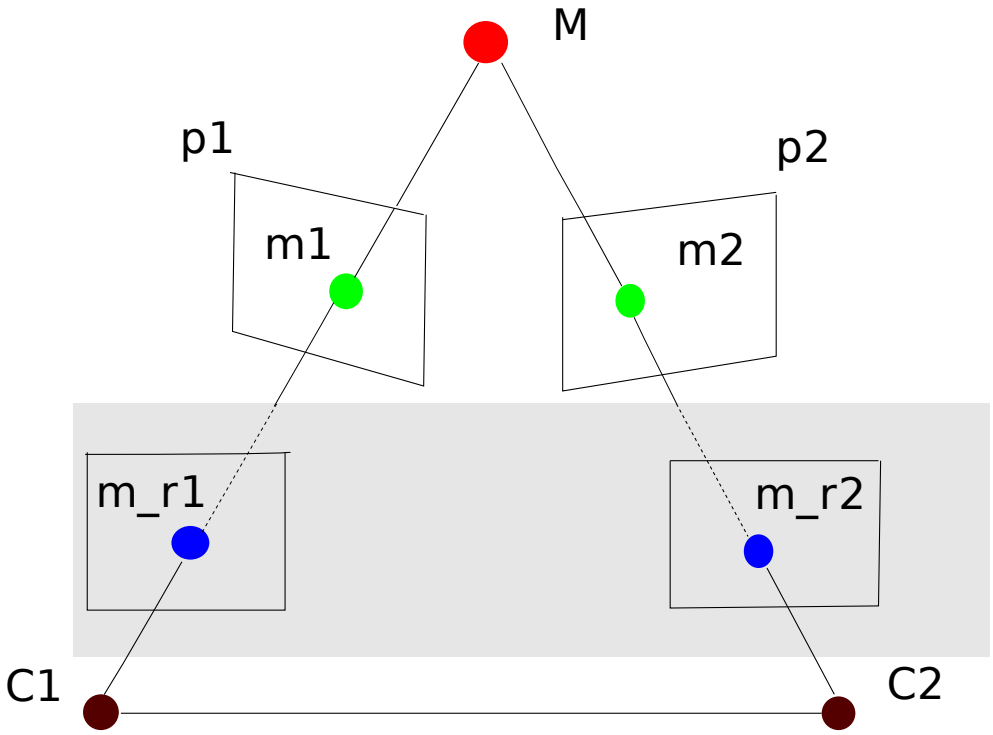

FIGURE 2.4 – La rectification consiste à reprojeter les images des plans rétinaux $p1$ et $p2$ sur un même plan P, dit « plan de rectification », parallèle à la droite $(C1C2)$ joignant les deux centres optiques. La reprojection revient à appliquer une homographie à chacune des images. Après rectification, les pixels m_r1 et m_r2 associés au point M de la scène 3D observée possèdent la même ordonnée.

d'utiliser un critère de mise en correspondance basé sur l'intensité lumineuse.
- Présence d'un flou de bougé ou de mise de point lors des prises de vue d'une scène.

En pratique cette impossibilité de réaliser la mise en correspondance se traduit par un trou sur la surface à reconstruire. Certaines méthodes (variationnelles par exemple) utilisent un terme de lissage local pour propager les valeurs des disparités des pixels voisins au pixel n'ayant pas de disparité définie. Ceci permettant une extrapolation, plus ou moins judicieuse, de la valeur de la disparité.

3 Le flot optique

Depuis l'approche pionnière de B.Horn and B.Schunck [1981], de très nombreux travaux concernant le flot optique ont été présentés. Il s'agit d'un des domaines les plus étudiés en vision par ordinateur, du fait du grand nombre d'applications. Le but de cette thèse n'est néanmoins pas de réaliser une étude exhaustive des algorithmes

existants, mais de présenter les limitations des hypothèses fondamentales du calcul du flot optique et quelques travaux importants dans ce domaine .

Limitations

Le but de l'estimation du flot optique est de réaliser un suivi d'objets dans une séquence d'images, obtenue généralement à partir d'une vidéo. En général il n'y pas de limitations sur le nombre d'objets ou sur la nature de ceux-ci car on cherche à estimer un vecteur de déplacement par pixel. Les contraintes fortes du problème proviennent essentiellement de la dynamique de la scène, du sous-échantillonnage des vidéos aboutissant à la séquence d'images à utiliser, et des conditions d'illumination. En effet, dans le cas de séquences où des objets apparaissent et disparaissent rapidement (cas de la séquence automobile précédente), un intervalle de sous échantillonnage trop large ne permet pas de faire la mise en correspondance. Il est donc nécessaire d'obtenir des séquences correctement échantillonées dans le temps par rapport aux mouvements mis en jeu dans la scène.
Tous les algorithmes de flot optique actuels se basent sur une même hypothèse fondamentale : la conservation au cours du temps du niveau d'intensité lumineuse d'un pixel associé à un objet en mouvement dans une séquence. En notant $I(\mathbf{X}, t)$ la fonction de l'espace et du temps représentant la séquence en question, et $t \rightarrow \mathbf{X}(t)$ la trajectoire d'un point de la scène dans cette séquence, cette contrainte se traduit mathématiquement par l'équation $\frac{dI(\mathbf{X}(t),t)}{dt} = 0$ ce qui s'écrit encore $\frac{\partial I}{\partial t} + \frac{d\mathbf{X}}{dt} \cdot \nabla I = 0$. La plupart du temps, cette contrainte est utilisée sous forme linéarisée, par exemple dans les travaux de B.Horn and B.Schunck [1981] et de Kornprobst [1998]. Cette linéarisation est valable dans les cas d'un flot optique de petite amplitude, mais il a été montré par Brox et al. [2004] que la forme non linéaire permet de diminuer l'erreur angulaire moyenne. Les limitations d'une telle hypothèse sont néanmoins immédiates : en présence d'ombre ou de spécularités, elle n'est plus valide. Certains travaux, en particulier le dernier cité, utilisent les gradients des images pour introduire une certaine robustesse par rapport aux variations d'illumination de la scène, mais la compensation de cette limitation demeure partielle.

D'autre part, l'équation ci-dessus est une équation a deux inconnues : les 2 composantes du flot optique $\frac{d\mathbf{X}}{dt}$. Ce qui implique que le problème est mathématiquement mal posé (désigné par "problème d'ouverture" dans la littérature). Nous allons voir que certaines méthodes, en particulier les méthodes de minimisation d'énergie, proposent d'ajouter un terme supplémentaire permettant de régulariser le problème.
Les limitations des méthodes actuelles d'estimation du flot optique proviennent donc essentiellement des scènes elles même, à la fois du point de vue de l'obtention des données et de leurs caractéristiques intrinsèques.

Exemple : analyse de la dynamique de fluides

Les applications du flot optique sont extrêmement nombreuses. Néanmoins une des applications récentes suscitant un grand intérêt dans la communauté de la vision

est l'application à l'analyse de la dynamique de fluides dans le cadre de l'assimilation de données.

Des travaux récents utilisent la théorie du contrôle optimal pour estimer des champs de vitesses contenus dans une séquence d'images.Papadakis et al. [2007a,b] minimisent par exemple une énergie parallèlement à l'estimation d'une fonction d'état par une formulation adjointe. Cette fonction d'état représente l'état d'un système obéissant à une loi dynamique (par exemple une loi de comportement de fluides, ou un modèle d'atmosphère dans le cas de l'assimilation de données issues d'images satellitaires) étant donné des mesures bruitées et parfois incomplètes de cet état (les images). L'utilisation d'une loi de la dynamique permet ici de renforcer la consistance temporelle du flot optique estimé en résolvant alternativement le problème de minimisation de l'énergie et le problème adjoint correspondant.

4 Quelques algorithmes de stéréoscopie et de flot optique par minimisation d'énergie

4.1 Mise en correspondance stéréoscopique

Les algorithmes de mise en correspondance stéréoscopique sont parmi les méthodes les plus étudiées en vision. En faire une étude exhaustive relève de l'impossible étant donné le grahd nombre de publications sur le sujet chaque année. Notre but est plutot ici d'indiquer quelles sont les grandes classes d'algorithmes couramment utilisés en stéréo. L'une des études les plus complètes des algorithmes existants a été réalisée par Scharstein and Szeliski [2002]. Nous présentons ici les bases de deux grandes classes de méthodes, que nous utilisons par la suite dans le cadre d'un système à deux caméras : les méthodes utilisant une formulation à base de champs de Markov, plus particluièrement les coupes de graphes (voir à ce sujet Kolmogorov and Zabih [2002]) mais on peut également citer la propagation de croyances, dont l'utilisation est illustrée par Sun et al. [2003], et les méthodes basées sur des EDP, à l'instar de la méthode développée par Strecha et al. [2003]. Dans cette catégorie on pourra distinguer différentes techniques, qui sont la minimisation d'une énergie construite de telle facon qu'un minimum corresponde à la solution du problème de mise en correspondance, comme dans le cas du travail de Slesareva et al. [2005], ou bien encore les techniques utilisant les level set, utilisées par Faugeras and Keriven [1998].

Coupe de graphes et propagation de croyances

Les algorithmes de vision basés sur les coupes de graphes modélisent les images sous forme de graphes, c'est à dire sous forme d'un ensemble de noeuds ν et d'un ensemble d'arrêtes orientées ϵ connectant les noeuds par paires 2.5. Il existe deux noeuds spéciaux et terminaux : la source et le puits. Chaque arête du graphe (p, q) se voit affectée d'un poids positif ou nul $w(p, q)$. Il faut noter qu'inverser l'orientation

d'une arête ne change pas obligatoirement seulement le signe du poids ($w(q,p)$ peut être différent de $-w(p,q)$). On appelle coupe de graphe une partition de l'ensemble des noeuds ν en deux parties ν_{source} contenant le noeud source et ν_{puit} contenant le noeud puit. Le coût d'une coupe de graphe est la somme des poids des arêtes traversées par la coupe, c'es-à-dire des arêtes (p,q) telles que $p \in \nu_{source}$ et $q \in \nu_{puit}$. Dans de très nombreux problèmes, on cherche à estimer la coupe de coût minimal. Étant donné la structure des images, les graphes les plus employés en vision sont naturellement des grilles bidimensionnelles. Dans ce cas là, le problème de min cut est un problème NP complet (on peut vérifier la validité d'une solution au problème avec un algorithme en temps polynomial, mais la résolution exacte du problème en lui même exige un temps considérable). En conséquence, les algorithmes de la littérature estiment des solutions approchées au problème. Il faut signaler que dans le cas de graphes monodimensionnels, le problème se résout directement par programmation dynamique. Un exemple d'application de la programmation dynamique en vision est montré par Amini et al. [1990]. Il existe de nombreux algorithmes pour trouver une solution approchée au problème de min cut, répartis dans deux grandes classes principales Goldberg and Tarjan [1988],Boykov and Kolmogorov [2001]. Dans ce cas les algorithmes de cette dernière classe sont généralement les plus efficaces.

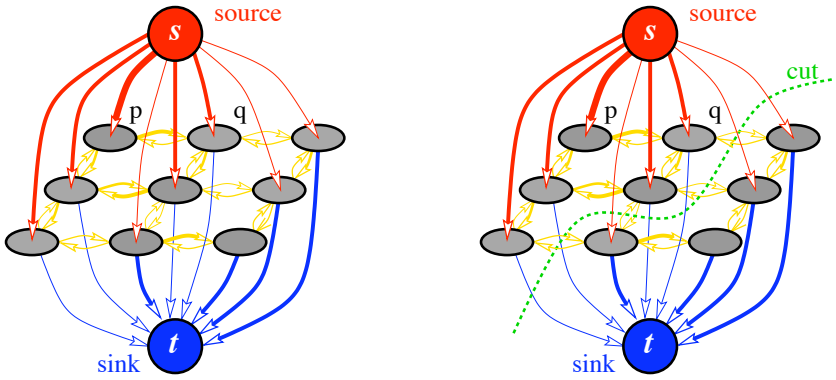

FIGURE 2.5 – Illustration issue du travail de Boykov and Veksler [2005]. A gauche : exemple d'un graphe. En rouge, la source du graphe, en bleu, le puit, et en grisé, les noeuds intermédiaires. À droite, et en vert, une coupe effectuée sur le graphe.

Comment utiliser les coupes de graphe pour la stéréoscopie ? On peut montrer que celles-ci peuvent avoir une interprétation géométrique sous la forme d'un hypersurface située dans une variété de dimension n, comme le montre l'illustration 2.6.

En pratique, certains pixels d'une image, correspondant à un objet donné, sont définis comme sources d'un graphe 2D. Des pixels du fond de l'image sont eux définis comme des puits du même graphe. Une hypersurface séparatrice correspond alors à la frontière de l'objet auquel les pixels source appartiennent. Étant donné que les contours des objets sont définis la plupart du temps par de hauts gradients en in-

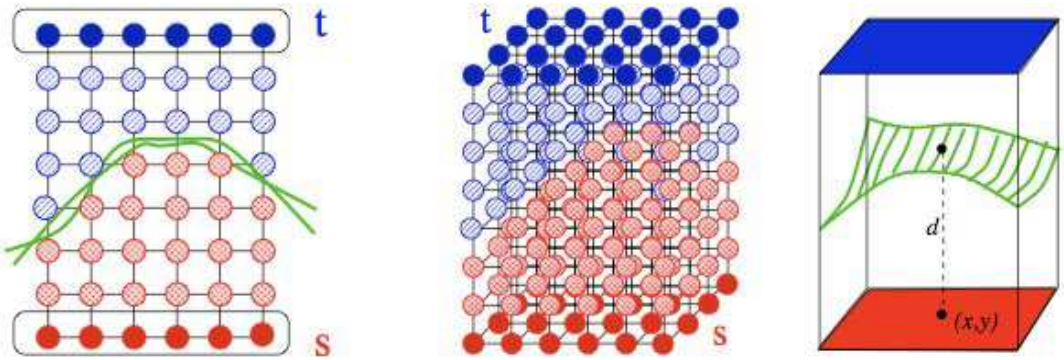

FIGURE 2.6 – Illustration issue du travail de Boykov and Veksler [2005]. À gauche : une coupe sur un graphe 2D, visualisée comme une hypersurface partitionnant l'espace auquel la grille appartient. Il faut noter que plusieurs hypersurfaces peuvent correspondre au même graph cut. Au milieu : un graphe 3D. À droite, une coupe sur ce graphe 3D, représentée par une hypersurface.

tensité, on affecte aux arêtes du graphe des poids fonction des gradients d'intensité. Un pixel possédant un fort gradient d'intensité ayant une forte probabilité de se situer au voisinage d'un contour d'objet, on affecte aux arêtes correspondantes un poids faible, et vice versa. Le graphe de cout minimal correspond ainsi au contour de l'objet. La stéréoscopie fut l'une des premières applications de la vision pour laquelle les algorithmes de coupes de graphe furent appliqués. Roy and Cox [1998], ainsi que Ishikawa and Geiger [1998] ont proposé une formulation de la stéréoscopie dans laquelle les cartes de disparité sont interprétées comme des hypersurfaces séparatrices dans des variétés 3D, et ainsi calculées efficacement avec les graphs cut. Dans le travail de Roy and Cox [1998], les cartes de disparité sont des hypersurfaces séparatrices dans une grille 3D appartenant à un espace de disparité 3D, correspondant à une paire d'images rectifiées. Les points (x, y, d) de cet espace correspondent à un point (x, y) d'une des images de la paire, choisie comme référence, et au point $(x + d, y)$ de la seconde image. La solution du problème stéréo est alors une surface séparatrice optimale $d = f(x, y)$ estimée par coupe de graphe, illustrée par la 3ème image de l'illustration 2.6. Parmi les approches très récentes de la stéréo par , et dans un contexte multi-caméras, il faut aussi signaler le travail de Sinha et al. [2007]. Il faut également noter qu'il existe des implémentations des algorithmes de coupes de graphe sur GPU (graphical processing unit) permettant une considérable accélération des temps de calcul, comme l'ont montré Vineet and Narayanan [2008].

Enfin, nous devons citer les approches de propagation de croyances qui se basent elles aussi sur une formulation par graphes des problèmes de vision. Une description de cette méthode de minimisation d'énergie définie sur des graphes peut être trouvée dans l'article de Felzenszwalb and Huttenlocher [2006]. Le code source de l'algorithme présenté dans ce travail étant disponible sur Internet à l'adresse `http://people.cs.uchicago.edu/~pff/bp/`, et les résultats présentés assez convain-

cants, nous avons utilisé cet algorithme à des fins d'initilisation de la mise en correspondance stéréoscopique dans l'algorithme d'estimation du flot de scène stéréoscopique que nous présentons par la suite. Comme pour les coupes de graphe, des implémentations sur GPU permettant de diminuer le temps de calcul sont apparues récemment, en particulier pour résoudre le problème stéréo (voir la méthode de Brunton et al. [2006]). Néanmoins, cette méthode reste gourmande en coût mémoire, puisque pour chaque pixel d'une image appartenant à une paire stéréo donnée, il faut stocker toutes les probabilités associées aux valeurs de disparité potentielles. Pour donner un ordre de grandeur, une machine dotée de 2 Go de mémoire vive était nécessaire pour utiliser l'algorithme de Felzenszwalb and Huttenlocher [2006] sur les images présentées au chapitre 4.

Méthodes variationnelles

L'intérêt des méthodes variationnelles est de fournir des résultats denses, et de pouvoir intégrer les différentes contraintes d'un problème dans une formulation mathématique unique. Le principe général est de définir une fonctionnelle paramétrée par les inconnues du problème. Cette fonctionnelle est généralement définie de telle sorte De telles méthodes présentent l'avantage de permettre une estimation de la disparité aux endroits des images où les informations viennent à manquer. En général, l'énergie à minimiser est composée d'un terme de fidélité aux données faisant intervenir les images, et d'un terme de régularisation propageant l'information des voisins aux pixels où l'information manque. Étant donné que les images peuvent être rectifiées, et la recherche de la disparité monodimensionnelle, la recherche d'une carte de disparité consiste à minimiser une énergie dont le terme de fidélité aux données est de la forme $J(d) = \int_\Omega \Phi(I_d(x+d,y) - I_g(x,y))$, Φ étant une fonction à choisir suivant les contraintes que l'on souhaite imposer à l'énergie. Il en résulte ainsi des cartes de disparité denses comme en proposent les méthodes de Alvarez et al. [2002], Kim and Sohn [2003], Mansouri et al. [1998], Robert and Deriche [1996], Scharstein and Szeliski [1998, 2002].

Un exemple récent d'estimation de cartes de disparité denses provient du travail de Slesareva et al. [2005] qui intègre la contrainte épipolaire à la formulation présentée par Brox et al. [2004]. L'énergie construite dans ce travail est composée d'un terme d'attaches aux données non linéaire intégrant la contrainte épipolaire et d'un terme de régularisation correspondant à une variation totale régularisée pour être dérivable en 0. La disparité est ici décomposée dans une base orthogonale composée d'une droite épipolaire et d'une droite perpendiculaire à cette dernière : $d(x,y) = p(x,y)\cdot v(x,y) + q(x,y)\cdot v(x,y)^\perp$, $v(x,y)$ désignant un vecteur directeur de la droite épipolaire associée au pixel de coordonnées (x,y) dans l'image choisie comme référence (la droite sur laquelle se trouve le pixel à mettre en correspondance avec le pixel (x,y)). La composante normale q est connue, il s'agit de la distance du pixel (x,y) à la droite épipolaire de direction v. L'énergie à minimiser dépend donc ici d'un seul paramètre, la composante épipolaire de la disparité, et se trouve être de la forme : $\int_\Omega \Phi((I_d(x+d(p),y) - I_g(x,y))^2 + \gamma\cdot(\nabla I_d(x+d(p),y) - \nabla I_g(x,y))^2) + \int_\Omega \Psi((\nabla p)^2)$. La

minimisation de cette énergie est réalisée en résolvant les équations d'Euler-Lagrange associées via une stratégie multirésolution. Les résultats présentés montrent des cartes de disparité denses avec une erreur moyenne en disparité plus basse que celles de Alvarez et al. [2002] (voir figure 2.7). Nous verrons dans la suite de la thèse que la formulation mathématique du problème du flot de scène peut se ramener à la formulation étudiée dans Slesareva et al. [2005] dans les cas où l'on voudra estimer seulement la disparité.

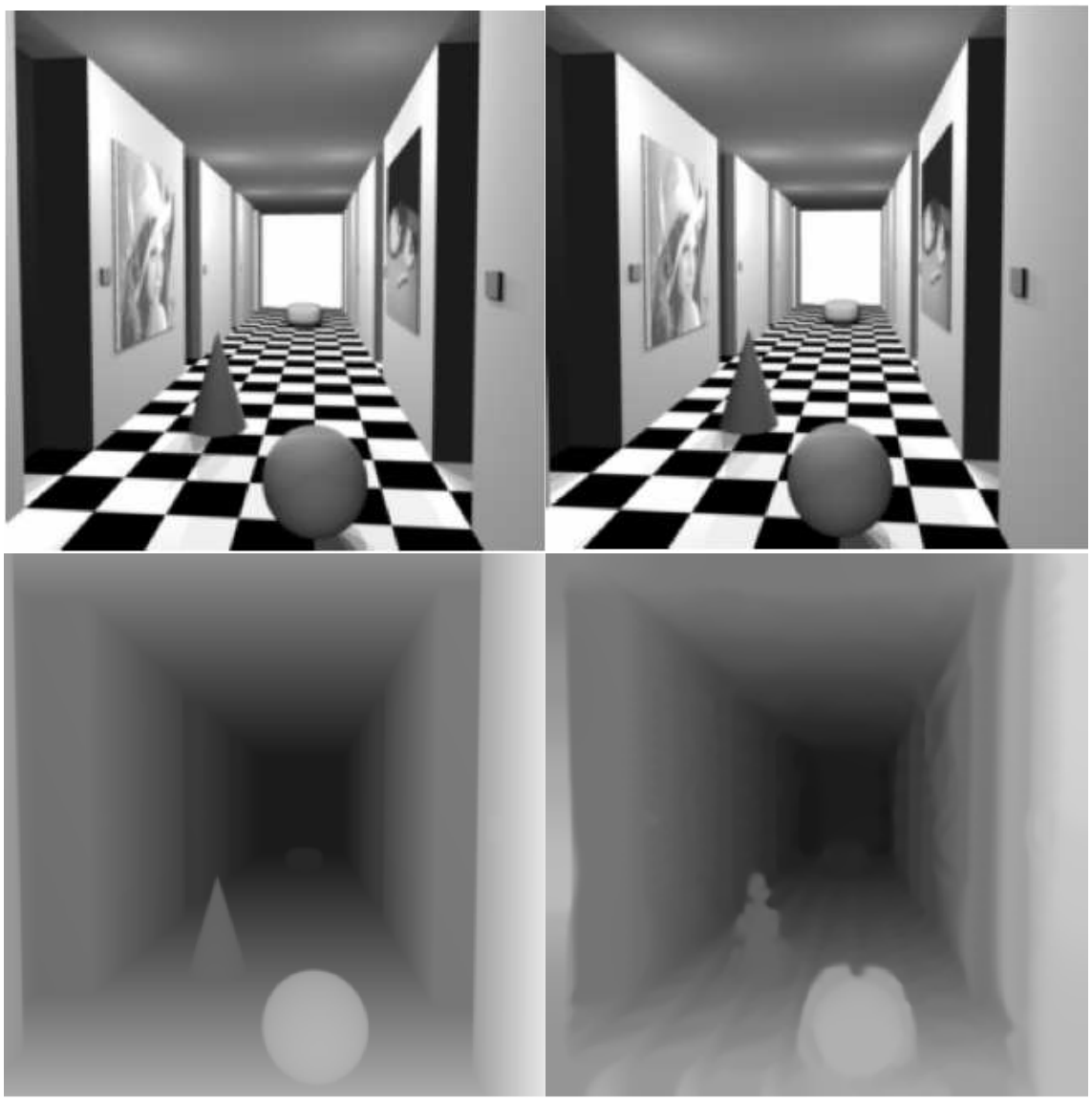

FIGURE 2.7 – En haut : une paire stéréo de test, dite séquence Corridor, disponible sur `www-dbv.cs.uni-bonn.de/`. En bas à gauche : la vérité terrain pour la disparité. En bas à droite : la disparité estimée par l'algorithme de Alvarez et al. [2002]

Parmi les approches mathématiques basées sur les EDP et développées pour ré-

soudre les problèmes de vision, on peut compter en outre les méthodes de suivi de l'évolution de courbes et de surfaces : on peut mentionner Kass et al. [1988] ou encore Caselles et al. [1997]. Parmi celles-ci, et pour le problème de la stéréoscopie, certains travaux ont utilisé les level set, en particulier celui de Faugeras and Keriven [1998]. Bien connus en mécanique des fluides, les level set, introduits par Sethian [1999] sont une technique particulièrement utilisée dans le suivi d'interfaces plongées dans un fluide. Il s'agit d'une approche géométrique qui a l'avantage de gérer automatiquement les changements de topologie des objets à suivre dans une scène donnée. Dans leur travail, Faugeras *et al.* présentent une description du problème de mise en correspondance stéréoscopique (pour un nombre quelconque de caméras supérieur ou égal à 2) sous la forme d'un problème de minimisation d'énergie. L'énergie décrite prend en compte les contraintes que doivent vérifier les surfaces des objets à détecter dans une scène donnée. Les équations d'Euler-Lagrange associées, donnant une condition nécessaire à l'obtention d'un minimum de cette énergie, sont alors utilisées pour déformer un ensemble initial de surfaces, lequel évolue ensuite vers les surfaces à reconstruire par level set. Il faut noter que certaines améliorations récentes de la technique des level set, comme par exemple le suivi de points sur les interfaces en évolution introduit dans le travail de Pons et al. [2006], pourraient être éventuellement intégrée aux méthodes existantes pour la stéréo.

4.2 Calcul du flot optique

Comme dans le cas de la stéréo, il existe différentes classifications des méthodes existantes comme par exemple la taxonomie de Barron et al. [1992] ou celle de Galvin B.Galvin et al. [1998]. Dans cette catégorie des travaux de comparaison d'algorithmes on peut également citer Nandhakumar and Aggarwal [1988], Otte and Nagel [1994], Mitiche and Bouthemy [1996] ou encore Stiller et al. [1999]. Certaines méthodes ont modifié le terme de régularisation afin de prendre en compte les discontinuités du flot optique : c'est le cas du travail de Deriche et al. [1995] et de Terzopoulos [1986]. De récents travaux se sont consacrés à la réduction du coût en temps de ces méthodes, aboutissant à des algorithmes temps réel comme l'algorithme de Bruhn et al. [2005b, 2006] ou parallélisés, en utilisant des méthodes de décomposition de domaines, par exemple dans le travail de Bruhn et al. [2005c].

Néanmoins, les meilleurs résultats en terme de précision furent obtenus par Brox et al. [2004] :ils évitent toute linéarisation des différents termes d'énergie dans leur formulation variationnelle en warpant l'image à l'instant $t+1$ vers l'image à l'instant t, et l'énergie globale n'est linéarisée qu'au moment de la résolution numérique. Dans la plupart des travaux de flot optique, l'hypothèse de conservation de l'intensité lumineuse s'écrit $\frac{\partial I}{\partial t} + \frac{d\mathbf{X}}{dt} \cdot \nabla I = 0$, et afin d'estimer $\frac{d\mathbf{X}}{dt}$, on cherche à minimiser une énergie dont le terme d'attache aux données est de la forme $\int_\Omega |\frac{\partial I}{\partial t} + \frac{d\mathbf{X}}{dt} \cdot \nabla I|^2$. Dans la méthode de Brox, ce terme d'attache aux données est remplacé par une distance quadratique entre les intensités des pixels devant se correspondre d'une image à l'autre, et devient de la forme $\int_\Omega |I(\mathbf{X}+\mathbf{W}, t+1) - I(\mathbf{X}, t)|^2 d\mathbf{X}$. Ainsi, les imprécisions numériques dues a l'approximation linéaire de la contrainte de conservation du niveau

d'intensité lumineuse, sont évitées . D'autre part cette méthode utilise les gradients d'intensité des images, inclus dans le terme de fidélité aux données. L'utilisation de ces derniers permet non seulement une plus grande robustesse aux variations d'illuminations dans une scène donnée, mais aussi de réduire les ambiguités en utilisant l'information locale et non plus simplement ponctuelle comme dans le cas de l'utilisation de l'information d'intensité seule. De plus, par l'utilisation d'une fonction convexe englobant le terme de fidélité aux données, cet algorithme est également relativement robuste numériquement aux occultations et aux discontinuités (mais ne les traite pas explicitement).Une justification théorique des performances de cette méthode est présentée par les mêmes auteurs dans Papenberg et al. [2006]. Slesareva et al. [2005] ont adapté cette formulation variationnelle au problème d'estimation de cartes de disparité denses, en ajoutant la contrainte épipolaire au terme d'attaches aux données de l'énergie définie par l'équipe de Brox. Du fait de ses avantages du point de vue numérique, nous avons étendu cette méthode à l'estimation du flot de scène comme nous le détaillerons au chapitre 3.

4.3 Exemples de séquences de test

Les séquences d'images utilisées pour tester les algorithmes doivent en principe mettre en lumière les points forts et faibles des algorithmes à tester. Une étude exhaustive des caractéristiques des différentes méthodes à tester nécessite ainsi une grande variété dans les propriétés des séquences utilisées. L'utilisation de textures de différentes taille et nature va permettre par exemple de mesurer la robustesse des algorithmes par rapport au type d'information mis en jeu. L'utilisation de grandes textures homogènes permet ainsi d'évaluer le comportement des algorithmes d'estimation du flot optique en l'absence d'informations (il est bien connu que l'estimation du flot optique est difficile en présence de grandes zones homogènes dans les images).

La base de données Middlebury, accessible à l'adresse `http://middlebury.vision.edu`, constitue la référence des bases de données de test pour la stéréoscopie (voir notamment la classification des algorithmes de stéréoscopie par Scharstein and Szeliski [2002]. Baker et al. [2007] ont récemment mis à jour cette base de données pour proposer des cas tests de flot optique. Ceux ci consistent en des séquences d'images réelles et leur vérité terrain associée obtenues en filmant des objets imprégnés d'une substance fluorescente. Ils ont ainsi généré des cas tests très divers du point de vue de la nature des objets et des mouvements observés. Quelques exemples issus de cette base de données, à la fois en stéréoscopie et en flot optique, sont montrés par les images 2.8,2.9, 2.10, ainsi que des exemples de résultats numériques en flot optique pour la séquence 2.10, montrés en figure 2.11.

FIGURE 2.8 – Paire stéréo Teddy, disponible sur `http://vision.middlebury.edu/stereo/data/`

4.4 Occultations : quelques méthodes existantes

Le problème des occultations étant extrêmement important, de nombreux travaux en vision stéréoscopique ont fait l'effort de les traiter simultanément à l'estimation de cartes de disparité. Il faut noter par ailleurs que la gestion des occultations est un des tests effectués par les auteurs de la base de donnée Middlebury pour effectuer un classement des algorithmes disponibles, ce qui disqualifie pratiquement d'office les algorithmes qui ne tenteraient pas de les gérer. Parmi les travaux récents permettant d'estimer des occultations ou des frontières d'occultations, on peut citer Williams et al. [2005] qui utilisent des champs de Markov 3D. Les pixels occultés se voient ici affecter un label particulier indiquant leur état. La modélisation des états des pixels étant réalisée par un champ de Markov aléatoire, un algorithme de belief propagation est alors utilisé pour reconstruire les cartes de disparité et retrouver les pixels occultés. Hoiem et al. [2007] retrouvent les contours d'occultation à partir d'une seule image. Ils utilisent des modèles d'apprentissage sur les frontières d'occultation en utilisant un certain nombre de critères : couleur, position, alignement de régions dans les images, longueur, estimation de l'orientation des surfaces 3D de la scène. L'algorithme utilise une sursegmentation initiale de l'image, et évolue vers une segmentation finale de celle-ci en utilisant l'apprentissage réalisé précédemment afin de fusionner progressivement les petites régions qui ont une certaine probabilité d'appartenir à un même objet de la scène. Gargallo et al. [2007] estiment les variations temporelles d'une fonction de visibilité par rapport aux variations d'une surface observée par une caméra. Un algorithme d'estimation très récent du flot optique prenant en compte les occultations est donné par Ince and Konrad [2008]. Il s'agit d'une méthode variationnelle qui estime simultanément le flot optique hors des zones occultées, détecte les occultations et extrapole le flot optique sur les zones occultées via une énergie composée de termes aux rôles spécifiques. Le terme de fidélité aux données de l'énergie définie dans l'article est désactivé dans

FIGURE 2.9 – En haut : paire stéréo Moebius composée des images 1 et 5. En bas : disparités associées. En bas à gauche : disparité entre les images 1 et 2 de la séquence. En bas à droite : disparité entre les images 5 et 6 de la même séquence. La séquences est téléchargeable sur `http://vision.middlebury.edu/flow/data/`

les zones occultées, laissant seule la contribution de diffusion anisotrope utilisée. La désactivation du terme d'attache aux données est effectuée en affectant un poids fonction d'une mesure de consistance du flot optique à ce terme.

5 Quelques méthodes d'estimation du flot de scène

Le flot de scène est un domaine jeune en vision par ordinateur comparativement au flot optique et à la stéréoscopie. L'état de l'art est par conséquent beaucoup moins dense que dans le cas de ces domaines. Il fut introduit par S.Vedula et al. [1999], S.Vedula and S.Baker [2005] comme étant le champ de vecteurs 3D défini sur les surfaces présentes dans une scène, décrivant le mouvement de chaque point 3D entre 2 instants consécutifs. Il peut être vu comme une extension 3D du flot optique.

Il existe actuellement deux grandes classes de méthodes d'évaluation du flot de

FIGURE 2.10 – Séquence Mequon (`http://vision.middlebury.edu/flow/data/`). En haut : images 7 et 8. En bas : images 9 et 10.

scène, réparties en deux grandes classes.

- Les méthodes utilisant les dérivées des images en temps et en espace : Zhang and Kambhamettu [2000], Neumann and Aloimonos [2001], Carceroni and Kutulakos [2002]. Sans terme de régularisation, l'estimation du flot de scène à partir de ces dérivées est un problème mal posé comme l'ont démontré Vedula et Baker. En général, il s'agit dans ce cas d'une généralisation 3D du "problème d'ouverture" rencontré dans le cas du flot optique (problème sous-contraint composé d'une équation à deux inconnues, les composantes du flot optique). Dans les travaux de Neumann and Aloimonos [2001] et Carceroni and Kutulakos [2002], les dérivées sont combinées afin de surcontraindre le flot de scène, alors que Zhang and Kambhamettu [2000] utilisent une régularisation de Tikhonov. Ceci étant, le caractère local des dérivées ainsi que l'hypothèse sous-jacente d'illumination constante restreignent ces méthodes principalement à des scènes variant lentement sous une illumination constante. Ce qui n'est la plupart du temps pas le cas des scènes réelles.
- La seconde famille de méthodes parmis lesquelles celles de S.Vedula and S.Baker [2005], Zhang and Kambhamettu [2000], utilise les flots optiques observés dans une ou plusieurs caméras afin de reconstruire le flot de scène. Néanmoins ces flots optiques obtenus avec plusieurs caméras peuvent ne pas être consistants entre eux, et de plus les contraintes de régularisation appliquées aux flots op-

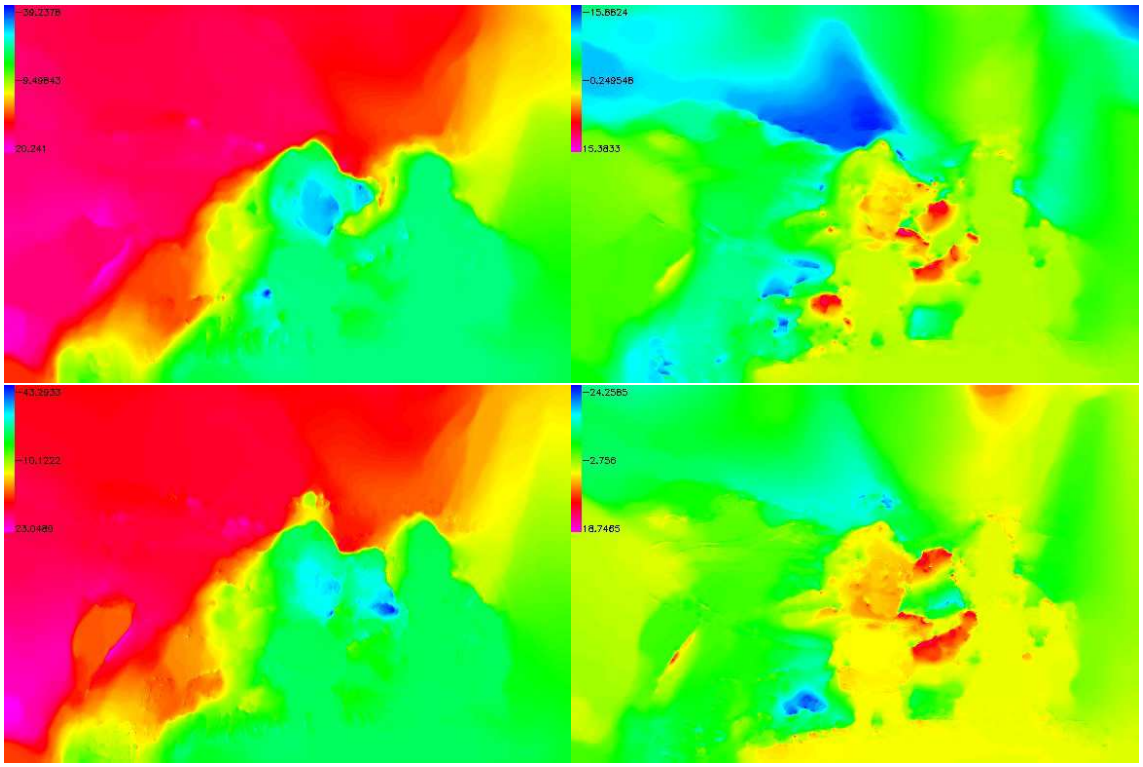

FIGURE 2.11 – Résultats numériques de quelques algorithmes sur la séquence Mequon. En haut : flot optique évalué par la méthode de Brox. En bas : flot optique issu de la projection 2D du flot de scène estimé par notre algorithme

tiques séparément peuvent éventuellement altérer l'estimation du flot de scène.

Concernant l'estimation du *flot de scène* dans le cadre variationnel, la seule méthode qui traitait à la fois de reconstruction et d'estimation du flot de scène au moment où la thèse a débuté, était celle proposée par Pons et al. [2005]. Elle ne calcule aucun flot optique et n'utilise pas non plus des estimations de dérivées spatio-temporelles des images. Il s'agit néanmoins d'une méthode de minimisation d'une énergie. Une énergie est minimisée pour estimer la reconstruction de la scèné, puis une seconde énergie est minimisée pour estimer le flot de scène. Classiquement celles-ci comprennent un terme d'attache aux données et un terme de régularisation rendant les problèmes bien posés. L'estimation du flot de scène est donc réalisée en optimisant alternativement la reconstruction et le champ de déplacement 3D.

- Dans le cas de la reconstruction stéréo, le terme de fidélité aux données est la somme pour toutes les paires d'images d'une mesure de similarité entre ces images. Pons *et al.* comparent une image prise par une caméra i avec la projection de l'image j vers la caméra i par l'intermédiaire d'une estimation de la surface à reconstruire. Celà revient donc à utiliser des prédictions sur les images trouvées par chacun des caméras en fonction de l'évolution de la surface à reconstruire.
- Le terme d'attache aux données de l'énergie minimisée par le flot de scène

est une somme sur toutes les caméras d'une mesuruse e de similarité entre les images à un instant t et les images prises à l'instant suivant $t+1$ backwarpées dans le temps en utilisant le flot de scène.

Les deux énergies sont minimisées alternativement par une méthode de descente de gradient.

De récents travaux proposèrent quant à eux une estimation jointe de la disparité et du flot optique : Min and Sohn [2006], qui néanmoins ne traitent pas les variations d'illumination et les occultations, et par conséquent obtiennent des résultats peu précis sur des séquences réelles, ainsi que Isard and MacCormick [2006], qui estiment une disparité et un flot optique à valeurs entières.

6 Notre stratégie d'évaluation du flot de scène

Nous proposons quand à nous une méthode qui estime le flot de scène par évaluation couplée de la surface reconstruite et du champ de déplacement, à partir de séquences stéréoscopiques issues de caméras calibrées. Cette méthode prend en compte la contrainte épipolaire entre les images d'une paire stéréo prise à un instant t, aboutissant à une paramétrisation minimale du flot de scène. Seulement 4 variables sont cherchées comme minimum d'une certaine énergie. Elles sont définies pour chaque pixel dans une image de référence de la manière suivante : la disparité au temps t, la disparité au temps $t+1$ et le flot optique.Le système de caméras étant calibré, le flot de scène est calculable directement à partir de ces fonctions comme indiqué sur la figure 3.5. Connaissant le flot optique et la disparité pour deux instants successifs, le flot de scène se déduit par simple différence des positions des points 3D de la scène.

Nous résolvons numériquement, via un algorithme multirésolution, un système d'EDP fortement non linéaires et couplées. Contrairement à la plupart des méthodes variationnelles d'estimation du flot optique, nous avons évité de linéariser la contrainte du flot optique et les termes de stéréo correspondant. En effet, Brox *et al.* ont démontré expérimentalement que cela permettait d'améliorer la précision des résultats numériques. Ce principe est étendu à toutes les contraintes issues de la modélisation du problème du flot de scène. D'autre part, un terme de régularisation adapté nous permet de préserver les discontinuités à la fois dans la reconstruction et dans le champ de déplacement 3D, permettant ainsi à des fractures d'apparaître sur une surface régulière au cours du temps.

En résumé, l'objectif de la thèse est donc de concevoir un algorithme d'estimation du flot de scène :

- qui soit robuste aux variations d'illumination dans la mesure du possible.
- qui permette de gérer des surfaces non lambertiennes et non fronto parallèles
- qui puisse gérer des mouvements rigides et non rigides, quelque soit la nature des objets observés

La modélisation mathématique du problème ainsi que sa mise en oeuvre algorithmique sont exposées dans les deux chapitres suivants.

Chapitre 3

Modélisation du problème

Ce chapitre traite spécifiquement de la modélisation du problème posé par l'estimation du flot de scène stéréoscopique. Partant de la donnée d'une séquence d'images stéréoscopiques, le but de l'algorithme présenté dans la thèse est d'estimer le flot de scène stéréoscopique pour les objets présents dans la scène observée. Il semble naturel d'examiner dans un premier temps les hypothèses du problème, relatives tant à la nature des scènes observées qu'à la régularité du champ de vitesses à estimer. Nous verrons quelles sont les hypothèses utilisées usuellement dans le cadre de l'estimation du flot optique et de la mise en correspondance stéréoscopique, et comment celles ci peuvent se retrouver dans l'étude du flot de scène. Nous examinerons en outre les hypothèses de régularité concernant le flot de scène. En effet, il nous faut prendre en compte les disconinuités présentes dans une scène donnée (occultations, contours). Dans un second temps, nous examinerons la modélisation du problème en elle même ainsi que la formulation mathématique associée. Celle-ci sera détaillée et justifiée terme par terme afin de montrer comment notre méthode prend en compte les diverses contraintes imposées par le problème d'estimation du flot de scène.

1 Hypothèses

1.1 Les hypothèses sur la scène 3D observée

Les hypothèses que l'on envisage habituellement sur les scènes 3D observées par un système mono- ou stéréoscopique sont généralement de plusieurs sortes.

- Les hypothèses liées aux objets de la scène. Celles ci concerne leurs propriétés physiques (élasticité, albédo ...) aussi bien que la nature de leurs mouvements (rigide ou non). De manière générale il n'y a pas d'hypothèses particulières formulées quand au nombre maximal d'objets gérables dans la scène, ni quant à la taille des objets.
- Les hypothèses sur l'éclairage de tout ou partie de la scène : présence d'ombres, de reflexions spéculaires ... autant de particularités pouvant poser problème aux méthodes d'estimation des champs de déplacement des objets dans les

séquences d'images.

Prenons l'exemple du flot optique, avec un exemple bien connu concernant l'évaluation du déplacement de voitures, la séquence du taxi de Hambourg, disponible à l'adresse `http://i21www.ira.uka.de/image_sequences/` et illustrée en figure 3.1.

FIGURE 3.1 – Séquence du taxi de Hambourg.

Sur cette séquence on observe la présence de diverses ombres, d'objets en mouvements de nature différente (voitures, piétons). La présence des ombres permet ici d'évaluer la robustesse des différentes méthodes de flot optique par rapport aux variations d'illumination, ce qui est une caractéristique importante à prendre en compte lorsqu'on analyse des séquences d'images réelles.

Concernant la stéréoscopie, l'hypothèse primaire à faire dans le cas de deux caméras consiste à supposer que l'écartement entre les caméras n'est pas excessif. En effet, il est nécessaire qu'une grande partie de la scène 3D soit visible dans les deux caméras afin de pouvoir mettre en correspondance un nombre suffisant de points de chaque paire stéréoscopique. Les systèmes comprenant N caméras, avec N assez grand, s'affranchissent évidemment de cette contrainte.

D'autre part, on peut constater qu'un grand nombre de méthodes stéréoscopiques présupposent que les surfaces à reconstruire sont lambertiennes et/ou frontoparallèles (parallèles au plan rétinien des caméras). Nous allons essayer, dans cette thèse, de gérer des surfaces qui ne le sont pas nécessairement. Enfin, comme pour le flot optique, nous ne formulons pas d'hypothèses spécifiques sur le nombre, la taille, la nature des objets à reconstruire.

Etant donné que nous étudions le flot de scène issu de séquences stéréoscopiques, les hypothèses présentées ici à la fois pour le flot optique et pour la stéréoscopie se combinent naturellement pour l'estimation du flot de scène.

1.2 Les hypothèses de régularité du flot de scène stéréoscopique

La méthode que nous proposons pour évaluer le flot de scène combine le flot optique et la stéréoscopie dans un formalisme unifié. Or, ces deux derniers ne sont pas des fonctions continues et ne sont à fortiori pas dérivables sur tout leur ensemble de définition, ce qui implique que le flot de scène stéréoscopique ne l'est pas non plus sur l'ensemble de la surface sur laquelle il est défini.La plupart du temps, les contours des objets dans les images constituent tout d'abord une discontinuité en intensité, mais aussi pour le flot optique ou la profondeur Z. En effet, il suffit de prendre l'exemple basique d'un objet unique observé sur un fond uniforme (mur ...) pour s'apercevoir que les bords de l'objet constituent une discontinuité pour le flot optique (figure 3.2) mais aussi dans le cas stéréoscopique (figure 3.3).

D'autre part les contours d'objets peuvent aussi délimiter des zones d'occultations d'autres objets situés en arrière plan, que ce soit pour le flot optique ou la stéréoscopie. Il semble donc raisonnable de supposer que les contours des différents objets vont également correspondre à des discontinuités pour le champ de vitesse 3D des objets dans la scène reconstruit à partir du flot optique et de la stéréoscopie, ainsi qu'a des contours d'occultations (qui formeront des trous dans les surfaces reconstruites).

Concernant la régularité du flot optique,Kornprobst [1998] a prouvé l'existence de solutions dans les cas ou les images sont lipschitziennes ainsi que dans le cas ou elles sont considérées comme des fonctions à variations bornées (BV). Nous nous restreindrons dans la suite au cas lipschitzien pour la raison suivante : lorsque qu'une caméra quelconque prend une image, les discontinuités sont toujours plus ou moins lissées lors du processus de formation de l'image, du fait du comportement des fonctions de transfert de modulation (FTM) des objectifs des caméras. En effet, la qualité des images produites par les objetifs diminue dans les hautes fréquences spatiales, ce qui se remarque généralement par l'atténuation du contraste dans les zones fortement hétérogènes en intensité, et donc par un lissage des intensités au voisinage des discontinuités. Dans ce cadre-là, il est donc raisonnable de supposer que les images sont uniformément lipschitziennes, pour une constante de Lipschitz dépendant des caméras (et non des séquences d'images).

L'hypothèse de régularité que nous adoptons pour le flot de scène est par conséquent celle d'un champ de vitesses régulier (c'est à dire au minimum continu ou dérivable) par morceaux. Les parties de surface sur lesquelles le flot de scène est régulier étant séparées par des discontinuités ou des trous correspondant au bord des objets ou bien à des occultations en flot optique ou en stéréoscopie.

FIGURE 3.2 – L'exemple RubberWhale issu de la base de données Middlebury, disponible à l'adresse `http://vision.middlebury.edu/flow/data/`. Les deux premières images constituent deux images consécutives d'une séquence. La troisème image donne la vérité terrain du flot optique. On remarque immédiatement que les contours des objets coincident avec les discontinuités du flot optique.

2 Modélisation du problème

Afin d'estimer un flot de scène dense, nous devons estimer le mouvement de chaque point de la scène 3D observée, lorsque celà est possible. Soit ainsi M un point 3D de cette scène, nous supposons dans cette partie qu'il n'est pas occulté pour une des deux caméras, à t et $t+1$. Il se projete alors dans chacune des 4 images. L'estimation du flot de scène stéréoscopique consiste alors a évaluer la position de

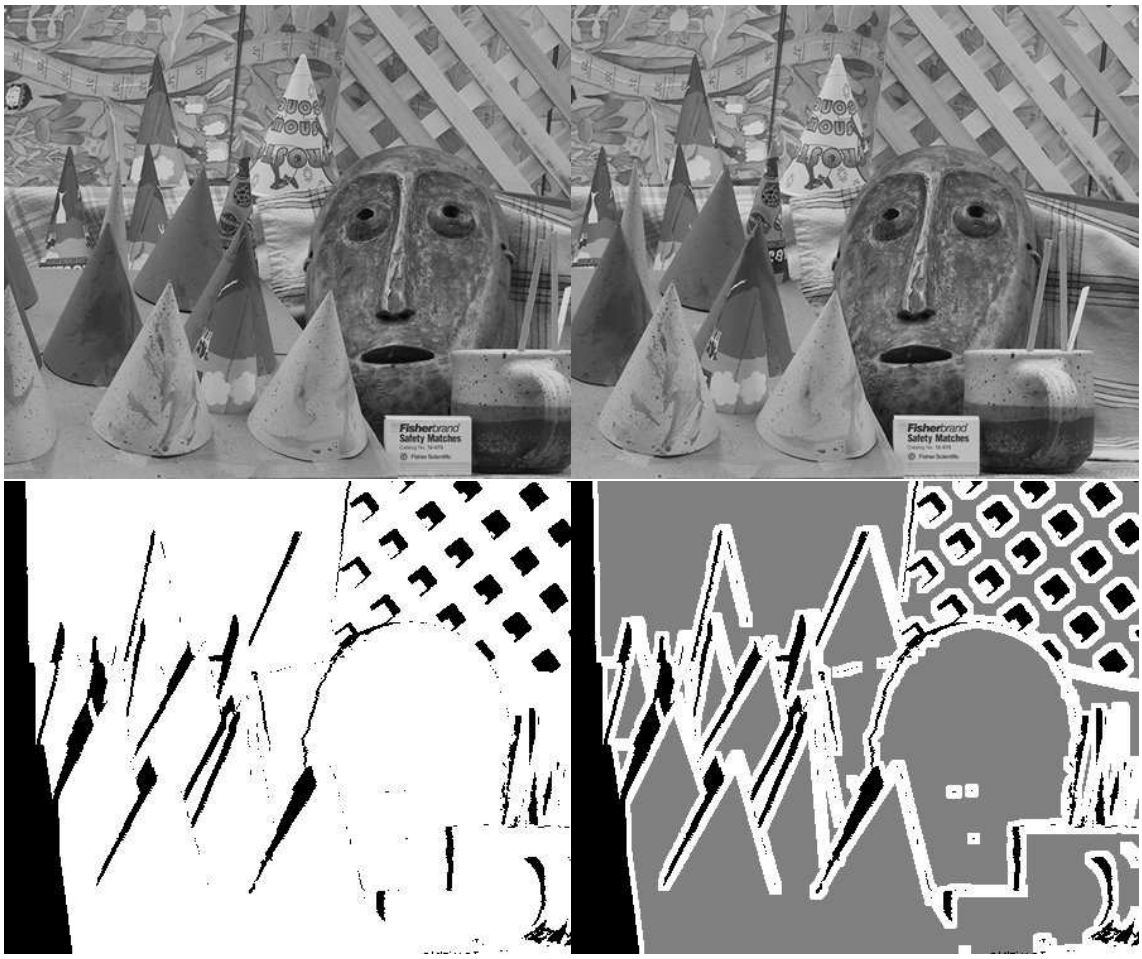

FIGURE 3.3 – L'exemple Cones issu de la base de données Middlebury, disponible à l'adresse `http://vision.middlebury.edu/stereo/data/`. En haut : une paire stéréo. En bas : a gauche, les occultations, a droite, les occultations et les discontinuités de la disparité. On constate que les bords des objets constituent à la fois des discontinuités en intensité et pour la disparité, et délimitent clairement des zones d'occultations.

chaque pixel correspondant à M dans les images. On peut envisager deux approches à ce sujet, l'approche directe, et l'approche eulérienne. Dans les deux cas, nous utilisons une méthode variationnelle consistant à minimiser une énergie fonction des inconnues du problème, ici, les fonctions permettant de retrouver les positions des pixels correspondant à M dans les séquences d'images.

2.1 Notations

Nous introduisons dans cette partie les notations que nous utilisons dans toute la suite de la thèse. Comme indiqué en figure 3.4, l'estimation du flot de scène

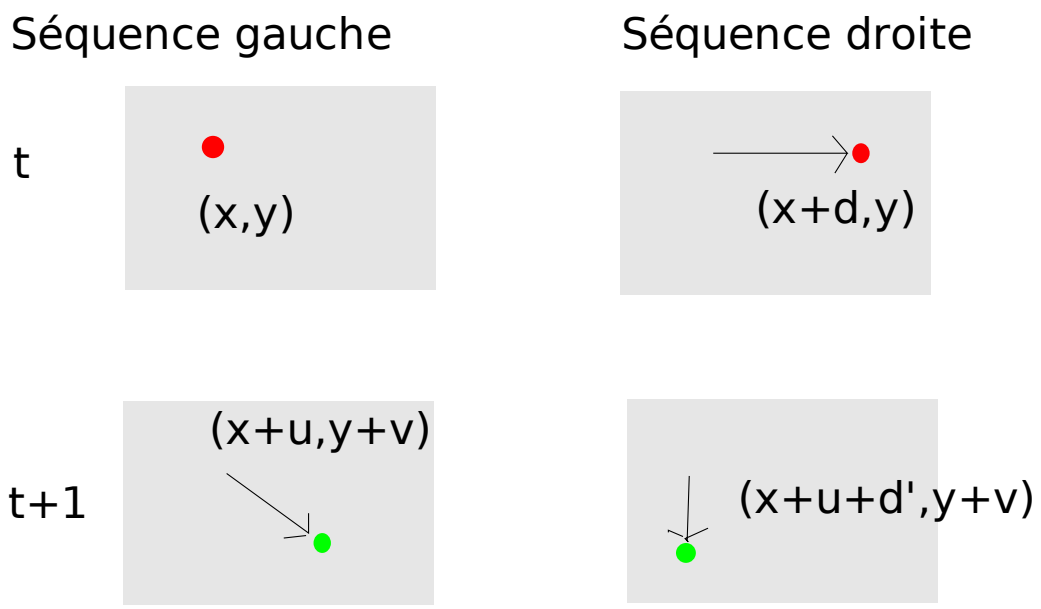

FIGURE 3.4 – Suivi d'un point 3D M dans les séquences d'images rectifiées à travers ses différents projetés dans les images. L'image I_g à l'instant t est prise comme référence pour l'expression des coordonnées des points correspondant à M. Les positions des points s'expriment en utilisant le flot optique $\mathbf{W} = (u, v)$ et les disparités d et $d^{'}$

est réalisée entre deux instants consécutifs t et $t + 1$. Pour cette raison, le temps n'apparaîtra par la suite pas de manière explicite dans la plupart des notations et dans les équations. Le flot de scène cherché étant dense, nous avons besoin d'un domaine de définition continu. Celui-ci est un rectagle noté Ω (correspondant à la forme des images). Nous introduisons par la suite deux approches différentes pour la formulation mathématique du problème.

- Une formulation directe faisant intervenir une surface $\mathbf{S_t}(p, q)$ définie à l'instant t sur Ω par un double paramétrage (p, q).
- Une formulation eulérienne dans laquelle Ω constitue le domaine de définition de plusieurs grandeurs, et dans laquelle Ω coincide avec le domaine de définition d'une des images de la figure 3.4 choisie comme système de coordonnées de référence. Les grandeurs définies sur Ω seront alors le flot optique $\mathbf{W}$(x,y) = $(u(x, y), v(x, y))$, ainsi que la disparité $d(x, y)$ à l'instant t. Nous introduirons également une grandeur $d^{'}$. Pour plus de commodités dans les notations, nous utiliserons les notations vectorielles $\mathbf{X} = (x, y)$, $\mathbf{D} = (d(x, y), 0)$ et $\mathbf{D}^{'} = (d^{'}(x, y), 0)$.

Supposons que les intensités des pixels appartenant aux images de la figure 3.4 soient définies sur un domaine Ω'. Nous définirons ces intensités comme des fonctions définies à t et $t+1$ sur Ω' soit :

$$\mathbf{X} \in \Omega' \rightarrow I_g(\mathbf{X}, t) \tag{3.1}$$

$$\mathbf{X} \in \Omega' \rightarrow I_g(\mathbf{X}, t+1) \tag{3.2}$$

$$\mathbf{X} \in \Omega' \rightarrow I_d(\mathbf{X}, t) \tag{3.3}$$

$$\mathbf{X} \in \Omega' \rightarrow I_d(\mathbf{X}, t+1) \tag{3.4}$$

Dans l'approche eulérienne présentée, les coordonnées des pixels correspondant à un même point de la scène 3D sont exprimées dans un système de coordonnées unique. Il faut donc choisir un tel système de coordonnées afin d'estimer les positions. Par convention et dans tout ce qui suit, nous choisirons l'image de gauche à l'instant t, $I_g(., t)$ comme image de référence. Une formalisation peut également être envisagée en utilisant une image cyclopéenne, comme nous le montrons plus tard.

2.2 Conception d'une méthode variationnelle

Les méthodes variationnelles en vision par ordinateur sont utilisées depuis très longtemps (on peut remonter au travail de B.Horn and B.Schunck [1981] pour le flot optique) car elles ont l'avantage de fournir une grande souplesse dans la modélisation des problèmes ainsi que des résultats naturellement denses (ce qui est moins évident avec les approches bayésiennes). De manière générale, il s'agit de minimiser une énergie $J(\Phi)$ fonction des inconnues Φ du problème. Il s'agit donc d'un problème d'optimisation par rapport à des variables fonctionnelles définies sur un espace donné. Dans la pratique, les espaces dans lesquels on cherche les solutions sont fonction des contraintes du problème : Kornprobst [1998] étudie par exemple le cas où le flot optique recherché est dans l'espace des fonctions a variations bornées $BV(\Omega)$, ce qui correspond au cas d'un flot optique discontinu sur des contours mesurables dans une image.

Comment construire l'énergie J ? En règle générale, en vision par ordinateur, cette énergie est la somme de deux termes $J_{data}(\Phi)$ et $J_{smooth}(\Phi)$. Le terme $J_{data}(\Phi)$ prend en compte les contraintes du problème dans un formalisme mathématique donné. Dans les problèmes de flot optique, il s'agit généralement de la norme quadratique de la contrainte de flot optique (Kornprobst [1998], Bruhn et al. [2003]) ou de la distance euclidienne entre les intensités des pixels correspondants entre deux images consécutives (Brox et al. [2004]). On retrouve le même type de termes d'attache aux données dans certains travaux récents sur la stéréo variationnelle. Les problèmes variationnels sont très souvent mal posés en vision : en général il n'existe pas de solution unique, ou encore de minima unique, au problème d'optimisation posé. Il convient pour y remédier numériquement d'ajouter un terme de régularisation $J_{smooth}(\phi)$. Celui ci, en général de la forme $J_{smooth} = \int_\Omega \Psi(\nabla\phi)$ ou $J_{smooth} = \int_\Omega \Psi(|\nabla\phi|^2)$, utilise en général l'information locale de l'image, ce qui permet de propager l'information d'un pixel vers ses pixels voisins.

Construction du terme d'attache aux données

Approche directe

Nous cherchons donc à reconstruire la dynamique des objets d'une scène 3D au cours du temps, tout en préservant les discontinuités qui apparaissent. Supposons que nous observons une surface $\mathbf{S}$ et son évolution au cours du temps. La surface étant paramétrée sur Ω , on note $\mathbf{S_t} = \mathbf{S}(p, q, t)$ la surface à un instant t quelconque. Le flot de scène recherché est donc par définition le champ vectoriel 3D $\mathbf{F}$ défini sur la surface $\mathbf{S}$ tel que $\mathbf{S_{t+1}}(p, q) = \mathbf{S_t}(p, q) + \mathbf{F}(\mathbf{S_t})$. Etant donné que l'on cherche un champ $\mathbf{F}$ dense, l'idée naturelle qu'on voudrait mettre en œuvre est donc d'appliquer une méthode variationnelle au champ vectoriel $\mathbf{F}$, c'est-à-dire de l'estimer comme minimum d'une énergie J.
Comme indiqué précédemment, $J(\mathbf{F}) = J_{data}(\mathbf{F}) + \alpha J_{smooth}(\mathbf{F})$, ou α désigne un terme de pondération de la régularisation par rapport à la fidélité aux données.Il est à déterminer numériquement. L'approche faisant intervenir directement la surface à estimer fait intervenir une énergie définie de la manière suivante :

$$J(\mathbf{F}) = J_{flot}(\mathbf{F}) + J_{stereo} + \alpha \cdot J_{smooth}(\mathbf{F}) \tag{3.5}$$

$$J_{flot}(\mathbf{F}) = \int_\Omega (I_g(\mathbf{S_t}(p, q) + \mathbf{F}, t + 1) - I_g(\mathbf{S_t}(p, q), t))^2 + (I_d(\mathbf{S_t}(p, q) + \mathbf{F}, t + 1) - I_d(\mathbf{S_t}(p, q), t))^2 dpdq \tag{3.6}$$

$$J_{stereo}(\mathbf{F}) = \int_\Omega (I_g(\mathbf{S_t}(p, q), t) - I_d(\mathbf{S_t}(p, q), t))^2 + (I_g(\mathbf{S_t}(p, q) + \mathbf{F}, t + 1) - I_d(\mathbf{S_t}(p, q) + \mathbf{F}, t + 1))^2 dpdq \tag{3.7}$$

$$J_{smooth}(\mathbf{F}) = \int_\Omega \Psi(|\nabla \mathbf{F}(p, q, t)|) dpdq \tag{3.8}$$

La notation $I_g(\mathbf{S_t}(p, q), t)$ désigne ici l'intensité du pixel de l'image $I_g(., t)$ correspondant au point de la surface $\mathbf{S_t}$ projeté dans la caméra de gauche. Les notations sont similaires pour la caméra de droite et pour l'instant $t + 1$. Le terme d'attache aux données J_{data} est la somme de deux termes :

- $J_{dataflow}$ est la distance euclidienne entre les intensités des pixels se correspondant entre deux images consécutives respectivement des séquences d'images gauche et droite. Autrement dit, ce terme modélise la contrainte de flot optique pour les deux séquences d'images.
- $J_{datastereo}$ représente la distance euclidienne entre les intensités des pixels correspondant au mêeme point de la surface observée entre les images des paires stéréo à t et à $t + 1$. Ce terme modélise donc la contrainte stéréoscopique.

Enfin, le terme de régularisation pénalise le gradient du flot de scène, tout en préservant les discontinuités de celui-ci grace aux propriétés de la fonction Ψ.

Une fois posée cette énergie, il est nécessaire de trouver un minimum qui convienne. Rien ne garantit à priori que cette énergie est convexe, il faut donc s'attendre à l'existence de minima locaux multiples. D'autre part, il faut estimer $\nabla_{\mathbf{F}} J$, soit, si $\mathbf{F} = (F_x, F_y, F_z)$, estimer $(\partial_{F_x} J, \partial_{F_y} J, \partial_{F_z} J)$. Or, l'énergie contient des termes non

linéaires en $I(\mathbf{S}+\mathbf{F})$. Celui-ci s'estime en projetant le point de la surface $\mathbf{S_{t+1}}$ dans le plan image de la caméra générant l'image I. Pour estimer les coordonnées pixel de cette projection, on multiplie les coordonnées 3D du point de la surface par la matrice de projection de la caméra. Etant dans le cas d'un système de caméras calibrées, on dispose des matrices de projection P et Q associées à chaque caméra. Les dites matrices permettent de passer en coordonnées pixel dans les images obtenues. On va noter dans la suite, en utilisant un système de coordonnées homogènes :

$$\begin{pmatrix} r.x_{g_pixel} \\ r.y_{g_pixel} \\ r \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{pmatrix} \begin{pmatrix} x_{\mathbf{S_{t+1}}} \\ y_{\mathbf{S_{t+1}}} \\ z_{\mathbf{S_{t+1}}} \\ 1 \end{pmatrix} \tag{3.9}$$

et

$$\begin{pmatrix} s.x_{d_pixel} \\ s.y_{d_pixel} \\ s \end{pmatrix} = \begin{pmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \end{pmatrix} \begin{pmatrix} x_{\mathbf{S_{t+1}}} \\ y_{\mathbf{S_{t+1}}} \\ z_{\mathbf{S_{t+1}}} \\ 1 \end{pmatrix} \tag{3.10}$$

$(r.x_{g_pixel}, r.y_{g_pixel}, r)$ représentant les coordonnées pixel dans la caméra de gauche d'un point M de la surface $\mathbf{S_{t+1}}$ de coordonnées $(x_{\mathbf{S_{t+1}}}, y_{\mathbf{S_{t+1}}}, z_{\mathbf{S_{t+1}}}, 1)$.

Or, $(x_{\mathbf{S_{t+1}}}, y_{\mathbf{S_{t+1}}}, z_{\mathbf{S_{t+1}}}, 1) = (x_{\mathbf{S_t}}, y_{\mathbf{S_t}}, z_{\mathbf{S_t}}, 1) + (F_x, F_y, F_z, 1)$. La présence des termes en $I(\mathbf{S}+\mathbf{F})$ dans l'énergie J nécessite, lors du calcul de $\nabla_{\mathbf{F}} J$, de dériver ces expressions par rapport aux coordonnées pixel des images, ce qui introduit d'après ce qui précède de nombreuses non linéarités ainsi que les termes des matrices de projection. Les équations obtenues sont ainsi très délicates à traiter (voir l'Annexe B pour les développements calculatoires) et il nous a semblé plus judicieux d'adopter une approche nous permettant de ne pas introduire ces difficultés dans nos équations.

Approche eulérienne

Traditionnellement, les approches eulériennes de suivi consistent à estimer l'évolution d'une grandeur donnée en chaque point de son domaine de définition au cours du temps. L'idée à mettre en oeuvre pour simplifier notre problème de formulation mathématique est de décomposer le flot de scène en composantes définies sur un domaine homéomorphe aux images, et d'évaluer l'évolution de ces grandeurs pour chaque pixel. Dans ce but, nous allons reconstruire le flot de scène à partir du flot optique et de la stéréoscopie, le tout dans un formalisme unifié. Nous utilisons ainsi une paramétrisation minimale du flot de scène à partir du flot optique et de la disparité d'une séquence d'images stéréo. Cette description du flot de scène est parfois appelée *flot de disparité* (Gong and Yang [2006]). Puisque cette paramétrisation est réalisée dans l'espace des images, le problème devient proche de celui de l'estimation du flot optique, avec plus d'inconnues et de mesures en chaque pixel.

La figure 3.4 illustre le suivi d'un point dans deux paires stéréo consécutives. Toutes les coordonnées et les fonctions à estimer sont ici définies dans le systèmes de coordonnées associé à l'image $I_g(., t)$, comme cela fut précisé précédemment. On peut signaler que nous avons aussi étudié une variante cyclopéenne, dans laquelle

nous prenons comme image de référence une image virtuelle issue d'une interpolation entre $I_g(.,t)$ et $I_d(.,t)$. les détails à ce sujet sont donnés en annexe. Le point M se projete dans $I_g(.,t)$ au pixel $m_{g,t}$ de coordonnées $\mathbf{X}$. Etant donné que ce pixel n'est pas occulté, nous pouvons estimer la position des pixels correspondants dans les 3 autres images, $m_{g,t+1}$, $m_{d,t}$ et $m_{d,t+1}$. Les coordonnées des points correspondants à M s'évaluent de la manière suivante :

- $m_{g,t}$ de coordonnées $\mathbf{X}$
- $m_{g,t+1}$ de coordonnées $\mathbf{X}+\mathbf{W}$
- $m_{d,t}$ de coordonnées $(\mathbf{X}+\mathbf{D})$
- $m_{d,t+1}$ de coordonnées $(\mathbf{X}+\mathbf{W}+\mathbf{D}^{'})$

Il est important de noter que $\mathbf{D}^{'}$ n'est pas la disparité définie à l'instant $t+1$, car cette fonction, donnant l'écart en pixels entre les points $m_{g,t+1}$ et $m_{d,t+1}$, est définie dans le système de coordonnées de $I_g(.,t)$ La disparité entre les images stéréo à $t+1$ est quant à elle exprimée dans le système de coordonnées de $I_g(.,t+1)$). En quelque sorte, $D^{'}$ est la transportée de la disparité à $t+1$ par l'inverse de la transformation définie par le flot optique (quand cela est possible toutefois, ce dernier n'étant pas toujours bijectif ni même défini).

Les images sont préalablement rectifiées donc $m_{g,t}$ et $m_{d,t}$ ont la meme ordonnée y, et il en est de meme pour $m_{g,t+1}$ et $m_{d,t+1}$. $m_{g,t+1}$ se déduit de $m_{g,t}$ par l'intermédiaire du flot optique calculé sur la séquence d'images gauche, $m_{d,t}$ se déduit de $m_{g,t}$ à l'aide de la disparité à l'instant t, et $m_{d,t+1}$ se déduit de $m_{g,t+1}$ avec $\mathbf{D}^{'}$. Il en résulte que si l'on connait $\mathbf{W}$ et $\mathbf{D}$, et étant donné que le système de caméras est calibré, on peut estimer le flot de scène, ce qui est illustré par la figure 3.5.

Il s'agit donc de formaliser cette description en l'intégrant dans une formulation variationnelle. Plusieurs exigences ont guidé notre recherche d'une formulation robuste. Tout d'abord, une exigence de précision : la plupart des algorithmes de calcul du flot optique utilisent une contrainte linéarisée. Dans ce cas, le terme d'attache aux données est de la forme $J_{data} = \int_\Omega |\mathbf{W}\cdot\nabla I + \frac{\partial I}{\partial t}|^2 dx$. Il s'agit simplement de la norme $L2$ de la contrainte de flot optique. Le principal défaut de ce genre de terme linéaire est d'etre une approximation valable en principe seulement pour les flot optique de petite amplitude. La plupart du temps, les algorithmes pallient à ce défaut en adoptant une approche multirésolution leur permettant de gérer non seulement les fonctionnelles non convexe, mais aussi les flots optiques de grande amplitude. Lefébure and Cohen [2001] donnent une justification à cette approche, permettant de montrer qu'une stratégie multirésolution permet de convexifier le problème. Néamoins, les travaux de Brox précédemment cités ont démontré que des approches non linéaires permettent de gagner en précision, tout du moins en terme d'erreur angulaire par rapport à la vérité terrain. Par conséquent, nous adoptons dans ce qui suit une approche non linéaire pour le terme d'attache aux données J_{data}. Les problèmes de flot optique et de stéréo étant tous deux des problèmes de mise en correspondance de pixels, nous pouvons définir de manière similaire les deux problèmes dans le terme de fidélité aux données. Nous choisirons donc :

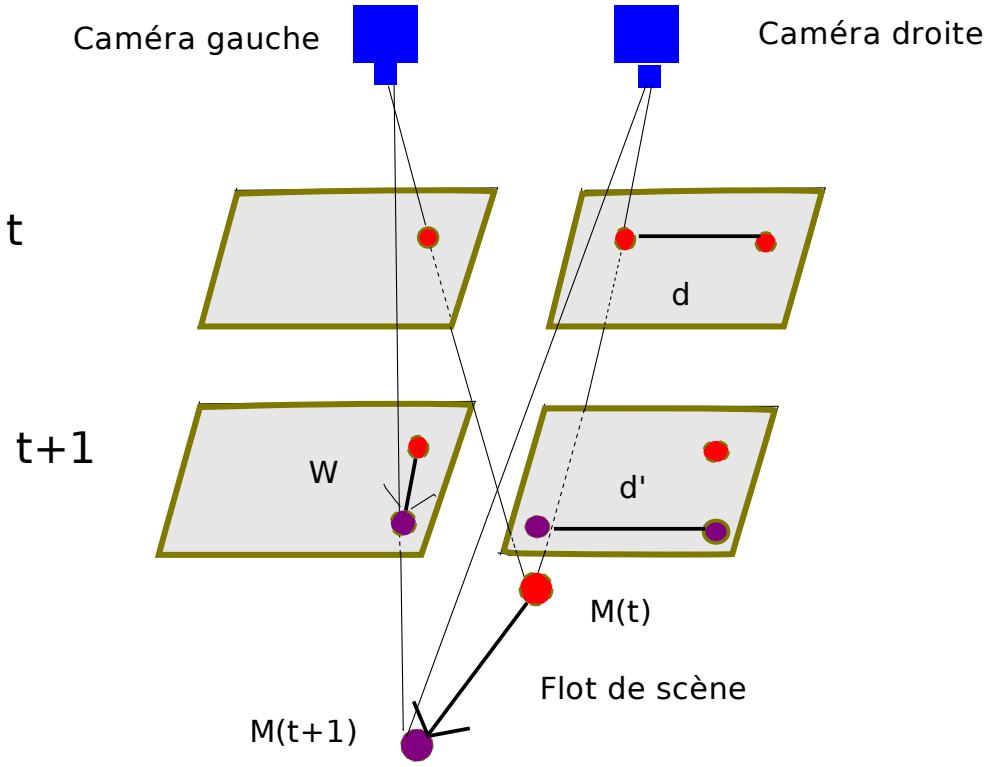

FIGURE 3.5 – Scène 3D observée par un système stéréoscopique rectifié. Reconstruction du flot de scène en utilisant le flot optique $\mathbf{W}$ et les disparités d et $d^{'}$. La connaissance de ces grandeurs permet de trianguler la position de M à t et $t+1$, le flot de scène se déduisant par différence des positions successives de M.

– pour le flot optique un terme du type

$$\int_{\Omega} |I_g(\mathbf{X}+\mathbf{W}, t+1) - I_g(\mathbf{X}, t)|^2 d\mathbf{X} \tag{3.11}$$

– pour la stéréo un terme du type

$$\int_{\Omega} |I_d(\mathbf{X}+\mathbf{D}, t) - I_g(\mathbf{X}, t)|^2 d\mathbf{X} \tag{3.12}$$

Les intégrales étant estimées sur le domaine associé au système de coordonnées de référence défini précédemment. Ces termes permettent de chercher $\mathbf{W}$ et $\mathbf{D}$ telles qu'un maximum de points soient mis en correspondance correctement, ce qui permet d'obtenir un minimum de l'énergie. En principe, la forme de ces termes ne permet pas de conclure quand à une convexité en $\mathbf{W}$ et $\mathbf{D}$ du terme d'attache aux données, et par conséquent de l'énergie totale, ce qui implique que nous serons amenés à essayer d'éviter des minima locaux incorrects.

Une seconde contrainte pour définir J_{data} est une des contraintes habituellement présente dans les scènes réelles : les changements d'illumination. Le flot de scène

doit être le plus robuste possible aux variations de l'éclairage de la scène, ce qui nécessite que ce soit aussi le cas pour l'estimation de ses composantes flot optique et stéréo. L'information sur l'intensité seule ne suffit pas : lorsqu'un objet de la scène passe dans une zone ombragée entre deux instants d'échantillonnage vidéo, il sera impossible de mettre en correspondance les pixels se situant avant et après passage dans l'ombre avec seulement l'information d'intensité. Il est donc nécessaire d'ajouter une information locale permettant de mieux discriminer les pixels dont l'intensité varie entre deux images. Lorsque l'illumination d'un objet varie, cela se répercute sur les pixels qui le représentent dans les images : etant donné un de ces pixels, ses voisins les plus proches subissent la même variation d'intensité. Par conséquent, le gradient d'intensité calculé dans cette région varie peu. Nous introduisons, comme ce fut le cas dans Brox et al. [2004], une distance quadratique entre les gradients des images :

– pour le flot optique un terme du type

$$\int_{\Omega} |\nabla I_g(\mathbf{X}+\mathbf{W}, t+1) - \nabla I_g(\mathbf{X}, t)|^2 d\mathbf{X} \tag{3.13}$$

– pour la stéréo un terme du type

$$\int_{\Omega} |\nabla I_d(\mathbf{X}+\mathbf{D}, t) - \nabla I_g(\mathbf{X}, t)|^2 d\mathbf{X} \tag{3.14}$$

Les images étant rectifiées, la stéréo est une mise en correspondance monodimensionnelle sur les droites épipolaires devenues horizontales. Ce terme de gradient permet en outre une meilleure mise en correspondance stéréo. Lors de la recherche du correspondant d'un pixel de $I_g(., t)$, par exemple, l'algorithme va avoir tendance à chercher, parmi les pixels de $I_d(., t)$ de meme intensité, celui dont le gradient d'intensité est le plus proche du gradient calculé au pixel à mettre en correspondance. Celà permet donc de réduire le nombre d'outliers pour la stéréo. L'introduction de cette distance entre gradients permet donc d'une part d'améliorer la robustesse aux variations d'éclairage de la scène, mais aussi de diminuer le nombre d'outliers en présence d'ambiguités sur les intensités.

Enfin, lorsqu'une mise en correspondance ne peut pas être faite, le terme d'attache aux données peut causer une instabilité numérique en créant un saut, ou pic d'énergie. Afin de remédier à ce problème et d'introduire une robustesse numérique aux occultations, nous choisissons de relativiser le poids du terme d'attache aux données par une fonction. Un certain nombre de travaux utilisent une minimisation $L1$, en utilisant une valeur absolue. Néanmoins, nous préférons opter pour une fonction dérivable en 0 : celà nous garantit de ne pas introduire de singularité dans l'énergie. Le comportement attendu d'une telle fonction correspond à une fonction linéaire à l'infini (pour des raisons de coercivité de l'énergie, ce qui facilite l'existence d'un minimum global si cela est possible) et qui "écrase" les valeurs autour de 0 (tout du moins les valeurs raisonnablement petites). La fonction d'Aubert présentée en figure 3.9 convient parfaitement pour cet usage, c'est donc celle que nous adoptons dans la suite. Dans toute la suite, nous la notons Ψ. La formulation du terme d'attache aux données combine donc plusieurs éléments simultanément :

- Une meilleure précision grace à la non linéarisation de la contrainte du flot optique
- L'introduction d'une distance sur les gradients permettant à la fois une certaine robustesse aux variations d'illuminations et la diminution d'ambiguités sur les mises en correspondance à partir des intensités
- Une relative robustesse aux occultations et aux grandes variations du terme de fidélité aux données

La formulation du terme d'attache aux données prend en compte tous les liens possibles entre deux paires stéréo consécutives : celà permet de contraindre le problème de manière optimale et d'exploiter toute l'information disponible. J_{data} est donc estimé de la manière suivante :

$$J_{data} = J_{fg} + J_{fd} + J_{s1} + J_2 \tag{3.15}$$

$$J_{fg} = \int_\Omega \Psi(|I_g(\mathbf{X}+\mathbf{W},t+1) - I_g(\mathbf{X},t)|^2 + \gamma \cdot |\nabla I_g(\mathbf{X}+\mathbf{W},t+1) - \nabla I_g(\mathbf{X},t)|^2) d\mathbf{X} \tag{3.16}$$

$$J_{fd} = \int_\Omega \Psi(|I_d(\mathbf{X}+\mathbf{W}+\mathbf{D}',t+1) - I_d(\mathbf{X}+\mathbf{D},t)|^2 + \gamma \cdot |\nabla I_d(\mathbf{X}+\mathbf{W}+\mathbf{D}',t+1) - \nabla I_d(\mathbf{X}+\mathbf{D},t)|^2) d\mathbf{X} \tag{3.17}$$

$$J_{s1} = \int_\Omega \Psi(|I_d(\mathbf{X}+\mathbf{D},t) - I_g(\mathbf{X},t)|^2 + \gamma \cdot |\nabla I_d(\mathbf{X}+\mathbf{D},t) - \nabla I_g(\mathbf{X},t)|^2) d\mathbf{X} \tag{3.18}$$

$$J_{s2} = \int_\Omega \Psi(|I_d(\mathbf{X}+\mathbf{W}+\mathbf{D}',t+1) - I_g(\mathbf{X}+\mathbf{W},t)|^2 + \gamma \cdot |\nabla I_d(\mathbf{X}+\mathbf{W}+\mathbf{D}',t+1) - \nabla I_g(\mathbf{X}+\mathbf{W},t+1)|^2) d\mathbf{X} \tag{3.19}$$

Le paramètre γ pondère l'influence des gradients par rapport a l'intensité. On le détermine expérimentalement en examinant le rapport entre la contribution due à l'intensité et celle due au gradient dans le cas où la minimisation est effectuée en utilisant les données de vérité terrain lorsque celles-ci sont disponibles. Dans le cas où les données de vérité terrain sont indisponibles, il n'y a à priori aucune raison de préférer une contribution à l'autre, par conséquent γ est tel que le rapport des contributions des intensités et des gradients est proche de 1 (du moins en début de minimisation). On notera que chaque inconnue u, v, d, d' est bien évaluée en chaque pixel de l'image de référénce, au cours du temps.

La régularisation du flot de scène

La plus forte contrainte concernant le design du terme de régularisation provient directement du type de régularité attendu pour le flot de scène. Celui-ci devant être régulier par morceaux, ses discontinuités doivent être préservées durant le processus de régularisation. Les premiers termes de régularisation employés (régularisation quadratique de Tikhonov and Arsenine [1977]) avaient le même effet qu'une équation de la chaleur isotrope (B.Horn and B.Schunck [1981]), jusqu'a ce que Terzopoulos [1986] ne s'intéresse à la préservation de discontinuités. De nombreux travaux ont suivi, permettant de dégager une famille de fonctions permettant d'effectuer une regularisation avec préservation des discontinuités, parmi lesquels Alvarez et al. [2000] ou Deriche and Faugeras [1997].

On peut démontrer qu'une telle régularisation est équivalente à une diffusion isotrope dans les zones de l'image homogènes, pauvres en textures, et à une diffusion anisotrope dans les zones comportant des discontinuités. Soit $\phi(u,v)$ une fonction quelconque et Ψ la fonction de régularisation à caractériser. Supposons que le terme de régularisation soit de la forme $J_{smooth} = \int_\Omega \Psi(|\nabla\phi|)$. Le terme correspondant dans les équations d'Euler-Lagrange, issu du calcul du gradient de l'énergie, est de la forme $\operatorname{div}\left(\Psi'(|\nabla\phi|)\frac{\nabla\phi}{|\nabla\phi|}\right)$. D'après la formule de la divergence :

$$\begin{aligned}\operatorname{div}\left(\Psi'(|\nabla\phi|)\frac{\nabla\phi}{|\nabla\phi|}\right) &= \frac{\Psi'(|\nabla\phi|)}{|\nabla\phi|}\operatorname{div}(\nabla\phi) + \nabla\phi\cdot\nabla\left(\frac{\Psi'(|\nabla\phi|)}{|\nabla\phi|}\right)\\ &= \frac{\Psi'(|\nabla\phi|)}{|\nabla\phi|}\triangle\phi + \nabla\phi\cdot\nabla\left(\frac{\Psi'(|\nabla\phi|)}{|\nabla\phi|}\right)\end{aligned} \tag{3.20}$$

Or on a :

$$\frac{\partial}{\partial u}\left(\frac{\Psi'(|\nabla\phi|)}{|\nabla\phi|}\right) = \frac{\frac{\partial}{\partial u}(\nabla\phi).\nabla\phi}{|\nabla\phi|^2}\left(\Psi''(|\nabla\phi|) - \frac{\Psi'(|\nabla\phi|)}{|\nabla\phi|}\right)$$

et symétriquement en $\frac{\partial}{\partial v}$. Par conséquent, on a :

$$\begin{aligned}\operatorname{div}\left(\Psi'(|\nabla\phi|)\frac{\nabla\phi}{|\nabla\phi|}\right) &= \frac{\Psi'(|\nabla\phi|)}{|\nabla\phi|}\triangle\phi + \frac{\partial\phi}{\partial u}\frac{\partial}{\partial u}\left(\frac{\Psi'(|\nabla\phi|)}{|\nabla\phi|}\right)\\ &\quad + \frac{\partial\phi}{\partial v}\frac{\partial}{\partial v}\left(\frac{\Psi'(|\nabla\phi|)}{|\nabla\phi|}\right)\\ &= \frac{\Psi'(|\nabla\phi|)}{|\nabla\phi|}\left(\triangle\phi - \frac{\nabla\phi\cdot\nabla|\nabla\phi|^2}{2|\nabla\phi|^2}\right)\\ &\quad + \Psi''(|\nabla\phi|)\frac{\nabla\phi\cdot\nabla|\nabla\phi|^2}{2|\nabla\phi|^2}.\end{aligned} \tag{3.21}$$

Nous allons interpréter notre terme de divergence comme expression d'un terme de diffusion dans une base locale formée de $\frac{\nabla\phi}{|\nabla\phi|}$ et du rotationnel $\frac{\nabla\times\phi}{|\nabla\phi|}$, et examiner les coefficients de cette décomposition.

Dans les zones de régularité de ϕ un développement de Taylor donne :

$$\forall h \in \mathbf{R}^3, \phi(x+th) = \phi(x) + t\nabla\phi(x)\cdot h + \frac{t^2}{2}h^T D^2\phi(x)h$$

On a alors, pour $h = \frac{\nabla\phi}{|\nabla\phi|}$, et en notant $d^2\phi(x)(h)$ la dérivée directionnelle d'ordre deux de ϕ suivant la direction h :

$$d^2\phi(x)(\frac{\nabla\phi}{|\nabla\phi|}) = \frac{\nabla^T\phi}{|\nabla\phi|}D^2\phi(x)\frac{\nabla\phi}{|\nabla\phi|}$$

. En développant puis simplifiant, on obtient :

$$d^2\phi(x)(\frac{\nabla\phi}{|\nabla\phi|}) = \frac{\nabla\phi\cdot\nabla|\nabla\phi|^2}{2|\nabla\phi|^2}$$

. De la meme manière on trouve pour l'autre terme que

$$d^2\phi(x)(\frac{\nabla \times \phi}{|\nabla\phi|}) = \triangle\phi - \frac{\nabla\phi \cdot \nabla|\nabla\phi|^2}{2|\nabla\phi|^2}$$

. Par conséquent, on obtient notre terme de divergence sous la forme :

$$\operatorname{div}\left(\Psi^{'}\left(|\nabla\phi|\right)\frac{\nabla\phi}{|\nabla\phi|}\right) = d^2\phi(x)\left(\frac{\nabla\phi}{|\nabla\phi|}\right)\Psi^{''}\left(|\nabla\phi|\right) + d^2\phi(x)\left(\frac{\nabla\times\phi}{|\nabla\phi|}\right)\frac{\Psi^{'}\left(|\nabla\phi|\right)}{|\nabla\phi|}. \tag{3.22}$$

Cette expression est utilisée pour trouver les conditions nécessaires sur ϕ afin d'obtenir un processus de régularisation stable. En effet, on peut remarquer les conditions suivantes :

$$\text{si } \frac{\Psi^{'}\left(|\nabla\phi|\right)}{|\nabla\phi|} = \Psi^{''}\left(|\nabla\phi|\right), \tag{3.23}$$

la régularisation s'effectue de manière isotrope dans tout l'espace et Ψ est dans ce cas une fonction quadratique. Cette diffusion étant isotrope, les discontinuités sont lissées, ce qui ne convient pas à notre problème. C'est la raison pour laquelle les fonctions de régularisation quadratiques de Tikhonov ne peuvent être employées dans notre cas.

On veut que la régularisation s'effectue de manière isotrope dans les zones de faible gradient $|\nabla\phi|$ ce qui nous amène directement aux conditions suivantes sur les coefficients du terme de divergence :

$$\lim_{|\nabla\phi|\to 0}\frac{\Psi^{'}\left(|\nabla\phi|\right)}{|\nabla\phi|} = \lim_{|\nabla\phi|\to 0}\Psi^{''}\left(|\nabla\phi|\right) = \Psi^{''}(0) > 0. \tag{3.24}$$

la condition sur la limite en 0 de $\Psi^{'}$ donne nécessairement que $\Psi^{'}(0) = 0$. Pour les zones homogènes, $|\nabla\phi|$ est petit et l'approximation suivante est valable :

$$\operatorname{div}\left(\Psi^{'}\left(|\nabla\phi|\right)\frac{\nabla\phi}{|\nabla\phi|}\right) \approx \Psi^{''}(0)\triangle\phi$$

et donc $\frac{\partial\phi}{\partial t} \approx \Psi^{''}(0)\triangle\phi$ ce qui correspond, avec $\Psi^{''}(0) > 0$, à une diffusion isotrope classique. Lorsque$|\nabla\phi| \to +\infty$, on détecte une discontinuité. Du point de vue numérique, on supposera en effet qu'il existe un seuil ϵ sur les valeurs de $|\nabla\phi|$ au delà duquel on considèrera que ϕ est discontinue.

Au voisinage des discontinuités, les lignes de niveau de ϕ se ressèrent, et on considère que l'orientation de ces lignes ressérées, donnée par $\frac{\nabla\phi}{|\nabla\phi|}$, donne la direction globale de la discontinuité. Pour préserver celles-ci, on considère traditionnellement que la composante de la diffusion le long du gradient, au voisinage d'une discontinuité, doit être nulle ou du moins décroître rapidement vers 0, tout en conservant un processus de diffusion stable le long de la direction tangent à la discontinuité,

soit

$$\lim_{|\nabla\phi|\to+\infty} \Psi''(|\nabla\phi|) = 0 \tag{3.25}$$

$$\text{et} \quad \lim_{|\nabla\phi|\to+\infty} \frac{\Psi'(|\nabla\phi|)}{|\nabla\phi|} = \beta > 0. \tag{3.26}$$

Mais ces deux dernières conditions sont incompatibles. On peut les affaiblir de manière à ce que le terme le long du gradient décroisse plus rapidement que le terme suivant la perpendiculaire au gradient, en imposant que les deux termes tendent vers 0 quand $|\nabla\phi| \to +\infty$ et que leur quotient tendent vers 0. Ces dernières conditions regroupées avec celles concernant les zones de continuité de ϕ sont celles prises en compte dans l'évaluation des fonctions Ψ préservant les discontinuités. Dans la littérature, plusieurs fonctions, ont été proposées. En voici un tableau récapitulatif.

Auteur	$\Psi(s)$
Geman-Reynolds	$\frac{s^2}{1+s^2}$
Malik-Perona	$\log(1+s^2)$
Green	$2\log(\cosh(s))$
Aubert	$2\sqrt{1+s^2}-2$

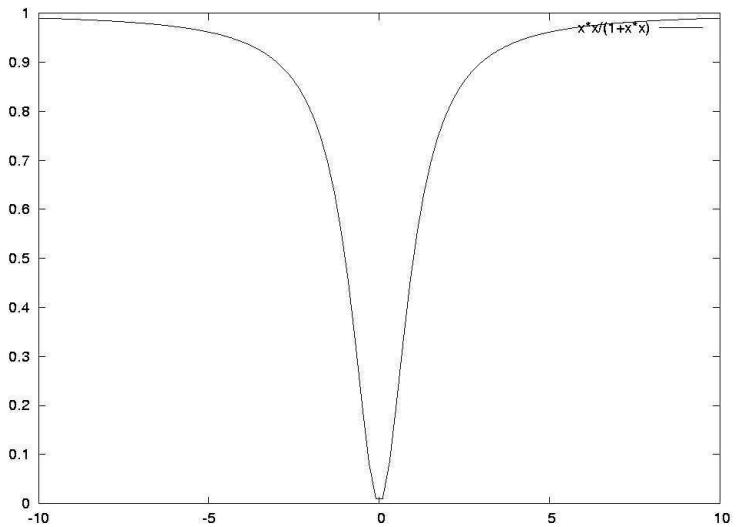

FIGURE 3.6 – Fonction de Geman-Reynolds

Les graphes de ces fonctions se trouvent sur les figures 3.6,3.7,3.8, 3.9. On observe bien sur ceux-ci qu'un processus de lissage est bel et bien effectué au voisinage de

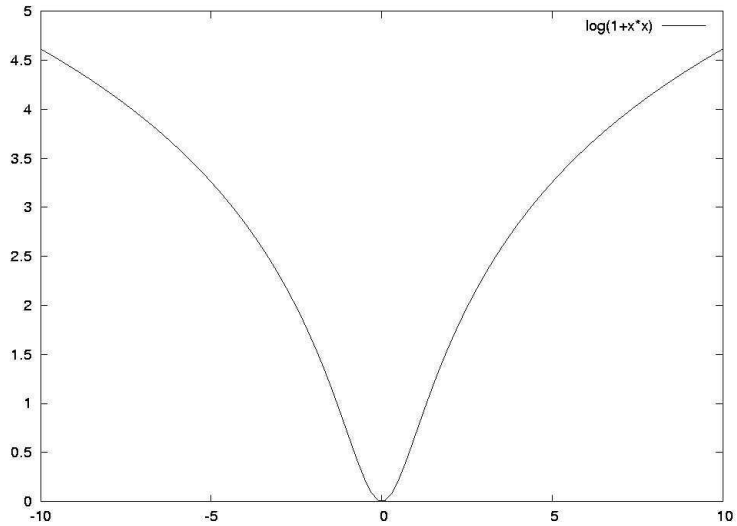

FIGURE 3.7 – Fonction de Malik-Perona

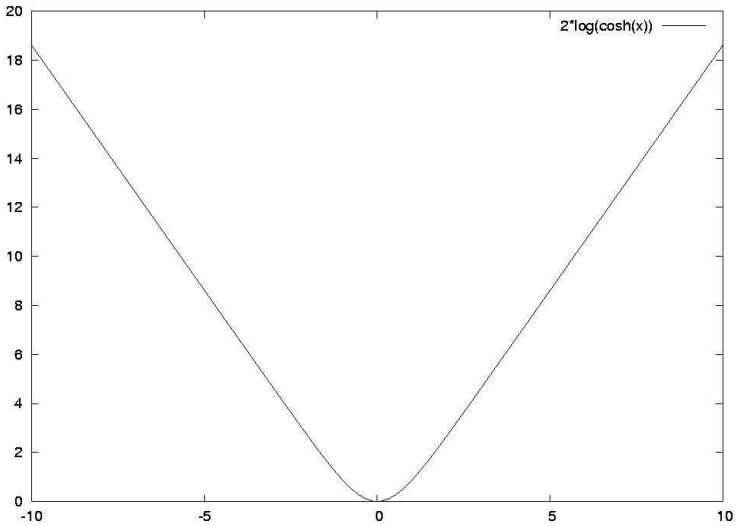

FIGURE 3.8 – Fonction de Green

0. Nous devons choisir une de ces fonctions pour obtenir des résultats numériques optimaux. Il se trouve que, numériquement, la fonction de Malik-Perona donne de bons résultats sur la reconstitution du flot optique (Deriche and Faugeras [1997]). Le terme de régularisation que nous avons choisi s'exprime alors de la manière suivante :

$$J = \int_{\Omega} \Psi(|\nabla \mathbf{W}|^2 + \lambda \cdot |\nabla(d^{'} - d)|^2 + \mu \cdot |\nabla d|^2) d\mathbf{X}. \tag{3.27}$$

En réduisant l'influence des hauts gradients du flot optique ou de la disparité sur l'énergie totale, la fonction Ψ a un rôle différent ici : elle permet de préserver les discontinuités de $\mathbf{W}$, et d, on trouve une application similaire dans le travail de Bruhn et al. [2005b]. Contrairement au cas de J_{data}, Ψ est appliquée à la somme

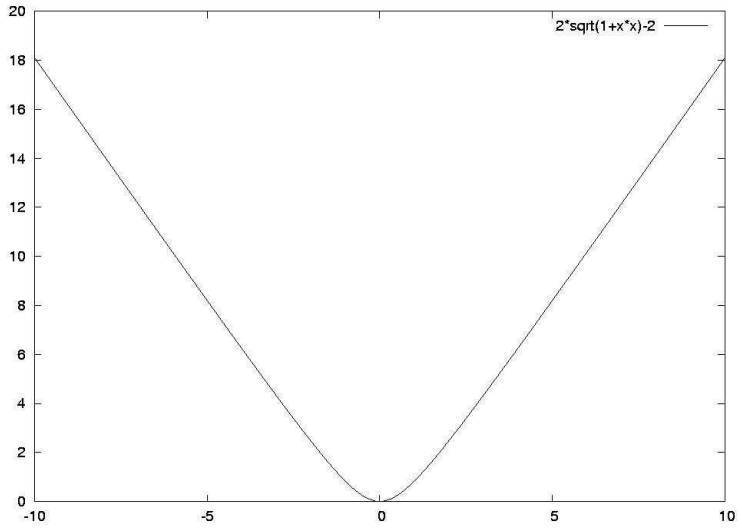

FIGURE 3.9 – Fonction d'Aubert

des normes des gradients, car habituellement les discontinuités apparaissent simultanément dans la disparité d, le flot optique $\mathbf{W}$, et le flot de disparité $d^{'} - d$ (sauf dans quelques cas spéciaux, comme dans l'exemple synthétique de la boule en rotation présenté dans la partie Résultats numériques).

L'effet de la régularisation sur le flot de scène ne devrait pas dépendre de l'orientation du champ de déplacement par rapport à la caméra, c'est pourquoi le paramètre λ devrait être ajusté pour rescaler l'influence de la régularisation sur le flot optique par rapport à la régularisation du flot de disparité, mais ne devrait pas être supérieur à μ pour éviter des oscillations durant l'optimisation : $\lambda < h/b$, où h est la distance moyenne des caméras à la scène, et b est la baseline du dispositif stéréo. L'effet de ce paramètre sera un flot de disparité $d^{'} - d$ plus régulier et des discontinuités plus petites lorsque la baseline est petite. Le paramètre μ règle l'influence relative de la disparité à t par rapport au flot optique. Puisque les discontinuités typiques dans les deux termes observées dans une scène devraient avoir le même effet sur J_{smooth}, une bonne estimation devrait être $\mu = hs/bS$ où s est l'amplitude attendue (en unités du monde) du flot de scène, et S une taille caractéristique de la scène : si le mouvement entre t et $t+1$ est petit par rapport à la taille de la scène, alors μ devrait être petit lui aussi.

3 Le problème des occultations

Le problème des occultations est fondamental en Vision, et intervient dans tous les problèmes liés au flot optique et à la stéréoscopie. En effet, il s'agit dans les deux cas de problèmes de mise en correspondance, et celle-ci ne peut parfois pas etre effectuée, soit parce qu'un pixel a une durée de vie limitée dans une séquence d'images (pour le flot optique), soit parce qu'il n'est pas visible dans toutes les caméras (dans

le cas de la stéréo). Ce genre de situations se produit en permanence dans les scènes 3D. De manière générale, les occultations sont utilisées pour mesurer la robustesse des algorithmes de mise en correspondance. Dans notre cas, les occultations peuvent se produire à la fois pour le flot optique dans les séquences gauche et droite, et la stéréo. Une occultation en flot optique ou en stéréo implique dans notre cas la non existence du flot de scène, puisqu'il nous manque alors l'information nécessaire à la reconstruction 3D à t ou à $t+1$. La figure 3.3 montre un exemple d'occultations en stéréoscopie pour la séquence Cones,

3.1 Quelques méthodes existantes

Le problème des occultations étant extrêmement importants, de nombreux travaux en vision stéréoscopique ont fait l'effort de les traiter simultanément à l'estimation de cartes de disparité. Il faut noter par ailleurs que la gestion des occultations est un des tests effectués par les auteurs de la base de donnée Middlebury pour effectuer un classement des algorithmes disponibles, ce qui disqualifie pratiquement d'office les algorithmes qui ne tenteraient pas de les gérer. Parmi les travaux récents permettant d'estimer des occultations ou des frontières d'occultations, on peut citer Williams et al. [2005] qui utilisent des champs de Markov 3D. Les pixels occultés se voient ici affecter un label particulier indiquant leur état. La modélisation des états des pixels étant réalisée par un champ de Markov aléatoire, un algorithme de belief propagation est alors utilisé pour reconstruire les cartes de disparité et retrouver les pixels occultés. Hoiem et al. [2007] retrouvent les contours d'occultation à partir d'une seule image. Ils utilisent des modèles d'apprentissage sur les frontières d'occultation en utilisant un certain nombre de critères : couleur, position, alignement de régions dans les images, longueur, estimation de l'orientation des surfaces 3D de la scène. L'algorithme utilise une sursegmentation initiale de l'image, et évolue vers une segmentation finale de celle-ci en utilisant l'apprentissage réalisé précédemment afin de fusionner progressivement les petites régions qui ont une certaine probabilité d'appartenir à un même objet de la scène. Gargallo et al. [2007] estiment les variations temporelles d'une fonction de visibilité par rapport aux variations d'une surface observée par une caméra. Un algorithme d'estimation très récent du flot optique prenant en compte les occultations est donné par Ince and Konrad [2008]. Il s'agit d'une méthode variationnelle qui estime simultanément le flot optique hors des zones occultées, détecte les occultations et extrapole le flot optique sur les zones occultées via une énergie composée de termes aux rôles spécifiques. Le terme de fidélité aux données de l'énergie définie dans l'article est désactivé dans les zones occultées, laissant seule la contribution de diffusion anisotrope utilisée. La désactivation du terme d'attache aux données est effectuée en affectant un poids fonction d'une mesure de consistance du flot optique à ce terme.

3.2 Estimation des occultations

We do not want that occulted pixels energy contributions induce any numerical instabilitues during the optimization process. The data term E_{data} is the sum of four contributions E_{fl}, E_{fr}, E_s, and E_{st}, each of these corresponding to a link between images extracted from consecutives stereoscopic pairs. Each of this link corresponds to either an optical flow either a stereo disparity and then can be affected by occlusions. Our idea is to weight each energy component to take into account the occlusions for each link. The weights used here are binary functions defined on Ω which values are 0 for occluded pixels and 1 for those being visible everywhere in two consecutive stereoscopic pairs. We could imagine to use smoothed functions (smoothed heavyside) or a probabilistic approach like Strecha's one Strecha et al. [2004b].

We use the following functions defined on Ω :

- β_{fl} weights E_{fl}
- β_{fr} weights E_{fr}
- β_s weights J_s
- β_{st} weights J_{st}

The occlusions are computed by estimating the functions β at the beginning of every high level fixed point iteration. The occlusions computation takes into account every update of the current scene flow estimation $(\mathbf{W}, \mathbf{D}, \mathbf{D}^{'})$ during the optimization process.

Les occultations sont gérées en estimant les fonctions $\beta_{g_flot}, \beta_{d_flot}, \beta_{stereo_t}, \beta_{stereo_nextt}$ au début de chaque itération de point fixe de haut niveau. Nous prenons ainsi en compte dans cette estimation des occultations chaque mise à jour de la solution $(u, v, d(., t), d(., t+1))$ au cours de l'optimisation. Les fonctions β valent 1 pour les pixels visibles et 0 pour les pixels occultés, de telle sorte que pour les pixels occultés partout, seul le terme de régularisation est pris en compte. Dans ce cas là, la régularisation aide à stabiliser la solution numérique obtenue en remplissant les pixels ou une mise en correspondance ne peut etre effectuée.

Nous décrivons les étapes nécessaires à l'estimation de la fonction β_{stereo_t} associée à la mise en correspondance stéréo à la date t, les autres fonctions étant évaluées similairement. Le principe général réside en la localisation des pixels de $I_g(., t)$ pour lesquels la transformation $\Phi_{\mathbf{D}} : \mathbf{X} \in \Omega \rightarrow \Phi(\mathbf{X}) = \mathbf{X} + \mathbf{D} \in \Omega^{'}$ n'est pas bijective ($\Omega^{'}$ étant ici associé au domaine de définition de $I_d(., t)$). En effet, lorsque 2 points correspondent effectivement au même point de la scène 3D, et que D est estimée correctement pour cette paire, $\Phi_{\mathbf{D}}$ est naturellement inversible pour cette paire de points. Pour celà, nous utilisons 2 cartes de disparité intermédiaires, d_1 et d_2. Une première étape consiste à estimer la disparité d_1 en transportant les valeurs de d vers le système de coordonnées associé à $\Omega^{'}$, en utilisant $\Phi_{\mathbf{D}}$. Pour $\mathbf{X}$ à valeurs entières, $\Phi_{\mathbf{D}}(\mathbf{X})$ n'est pas forcément à valeurs entières, le transport des valeurs de d se traduit donc par :

- $\forall (i, j) \in \Omega, (i, j) \in N^2, d_1(\lfloor i + d(i, j) \rfloor, j) = d(i, j)$
- $\forall (i, j) \in \Omega, (i, j) \in N^2, d_1(\lfloor i + d(i, j) \rfloor + 1, j) = d(i, j)$

Dans le cas où deux couples (i, j) et $(i^{'}, j)$ sont tels que $\lfloor i + d(i, j) \rfloor + 1 = \lfloor i + d(i^{'}, j) \rfloor$, par exemple, la valeur de disparité choisie pour $d_1(\lfloor i + d(i^{'}, j) \rfloor, j)$ est la plus petite des deux (à l'instar de ce qui se passe pour la profondeur dans un processus de Z buffering).

Une fois d_1 estimée, nous obtenons la carte de disparité d_2 en transportant d_1 vers le système de coordonnées de $I_g(., t)$, à l'aide de $\Phi_{\mathbf{D}}^{-1}$ lorsque celle-ci existe. Compte tenu des approximations numériques pouvant apparaître durant cette opération, nous ajoutons un seuil de tolérance (en pratique nous choisissons 1.5 pixel) pour la disparité obtenue. Aux pixels pouvant être mis en correspondance, et dans le cas où l'estimation de d est correcte, d_2 et d doivent être identiques au seuil de tolérance près. Lorsque ce seuil est dépassé, on peut conclure que l'on a détecté une occultation. Nous résumons cette approche dans l'algorithme 1.

Algorithme 1 Estimation de β_{stereo_t}

Sorties:

Entrées: Valeurs de la disparité d définie dans le système de coordonnées associé à $I_g(., t)$. Images $I_g(., t)$ et $I_d(., t)$. 2 Cartes de disparité intermédiaires à estimer d_1 et d_2. Seuil de tolérance sur la disparité τ.

Initialiser d_1 avec la plus grande valeur de disparité possible (la dimension horizontale de l'image exprimée en pixels).

pour $\forall(i, j)$ **faire**
  pour $p \in [\lfloor i + d(i, j) \rfloor, [\lfloor i + d(i, j) \rfloor + 1]$ **faire**
    si $d(i, j) < d_1(p, j)$ **alors**
      $d_1(p, j) = d(i, j)$
    finsi
  fin pour
fin pour
$d_2 = \Phi_{\mathbf{D}}^{-1}(d_1) + \tau$
Détection des pixels occultés
pour $\forall(i, j)$ **faire**
  si $|d_2(i, j) - d(i, j)| > 0$ **alors**
    $\beta_{stereo_t}(i, j) = 1$
  sinon
    $\beta_{stereo_t}(i, j) = 0$
  finsi
fin pour

La mise à jour de J_{data} pour chaque pixel de l'image de référence $I_g(., t)$ peut alors être effetuée, connaissant les occultations pour chaque composante de J_{data}. Ainsi, plus un pixel de $I_g(., t)$ connait des difficultés à trouver un correspondant dans les autres images, moins son apport à l'énergie totale est important. Dans le cas où aucune correspondance ne peut etre trouvée, J_{smooth} assure la stabilité numérique de la solution en propageant l'information des pixels voisins. Nous présentons un

exemple de cartes d'occultations obtenues sur l'exemple d'une boule en rotation, celle-ci étant texturée pour permettre une mise en correspondance correcte à la fois pour le flot optique et la stéréo.

4 Forme finale de l'énergie

Nous avons précédemment défini l'allure globale du terme de fidélité aux données et le terme de régularisation. Il reste à inclure la gestion des occultations. L'énergie modélisant le problème du flot de scène stéréoscopique se présente donc sous la forme suivante :

$$J(\mathbf{W},\mathbf{D},\mathbf{D}') = J_{data}(\mathbf{W},d,d') + \alpha \cdot J_{smooth}(\mathbf{W},\mathbf{D},\mathbf{D}') \tag{3.28}$$

$$J_{data}(\mathbf{W},\mathbf{D},\mathbf{D}') = \beta_{g_flot} \cdot J_{g_flot} + \beta_{d_flot} \cdot J_{d_flot} + \beta_{stereo_t} \cdot J_{stereo_t} + \beta_{stereo_nextt} \cdot J_{stereo_nextt} \tag{3.29}$$

$$J_{g_flot}(\mathbf{W},\mathbf{D},\mathbf{D}') = \int_\Omega \Psi(|I_g(\mathbf{X}+\mathbf{W},t+1) - I_g(\mathbf{X},t)|^2 + \gamma \cdot |\nabla I_g(\mathbf{X}+\mathbf{W},t+1) - \nabla I_g(\mathbf{X},t)|^2) d\mathbf{X} \tag{3.30}$$

$$J_{d_flot}(\mathbf{W},\mathbf{D},\mathbf{D}') = \int_\Omega \Psi(|I_d(\mathbf{X}+\mathbf{W}+\mathbf{D}',t+1) - I_d(\mathbf{X}+\mathbf{D},t)|^2 + \gamma \cdot |\nabla I_d(\mathbf{X}+\mathbf{W}+\mathbf{D}',t+1) - \nabla I_d(\mathbf{X}+\mathbf{D},t)|^2) d\mathbf{X} \tag{3.31}$$

$$J_{stereo_t}(\mathbf{W},\mathbf{D},\mathbf{D}') = \int_\Omega \Psi(|I_d(\mathbf{X}+\mathbf{D},t) - I_g(\mathbf{X},t)|^2 + \gamma \cdot |\nabla I_d(\mathbf{X}+\mathbf{D},t) - \nabla I_g(\mathbf{X},t)|^2) d\mathbf{X} \tag{3.32}$$

$$J_{stereo_nextt}(\mathbf{W},\mathbf{D},\mathbf{D}') = \int_\Omega \Psi(|I_d(\mathbf{X}+\mathbf{W}+\mathbf{D}',t+1) - I_g(\mathbf{X}+\mathbf{W},t+1)|^2 + \gamma \cdot |\nabla I_d(\mathbf{X}+\mathbf{W}+\mathbf{D}',t+1) - \nabla I_g(\mathbf{X}+\mathbf{W},t+1)|^2) d\mathbf{X} \tag{3.33}$$

$$J_{smooth}(\mathbf{W},\mathbf{D},\mathbf{D}') = \int_\Omega \Psi(|\nabla \mathbf{W}|^2 + \lambda \cdot |\nabla(d' - d)|^2 + \mu \cdot |\nabla d|^2) d\mathbf{X} \tag{3.34}$$

Il faut noter qu'un travail très récent de Wedel et al. [2008] a repris une énergie quasi identique.

En pratique, α dépend de la nature des séquences d'images utilisées. Mais en règle générale, on observe numériquement que la plage de valeur donnant une solution convenable au final est relativement étroite. Une valeur trop petite donne une solution parfois chaotique, alors qu'une valeur trop grande aboutit à une solution trop lissée, dont les discontinuités sont gommées. Le choix de ce paramètres est donc essentiellement expérimental. Cette énergie comporte beaucoup de non linéarités, notamment parce que nous avons choisi délibéremment d'éviter de linéariser la contrainte de flot optique et celle correspondant à la stéréo. Les images et les inconnues (u, v, d) n'étant à priori pas des fonctions convexes, l'énergie n'a aucune raison de l'être non plus. Définie comme telle, J possède plusieurs minima locaux, ce que

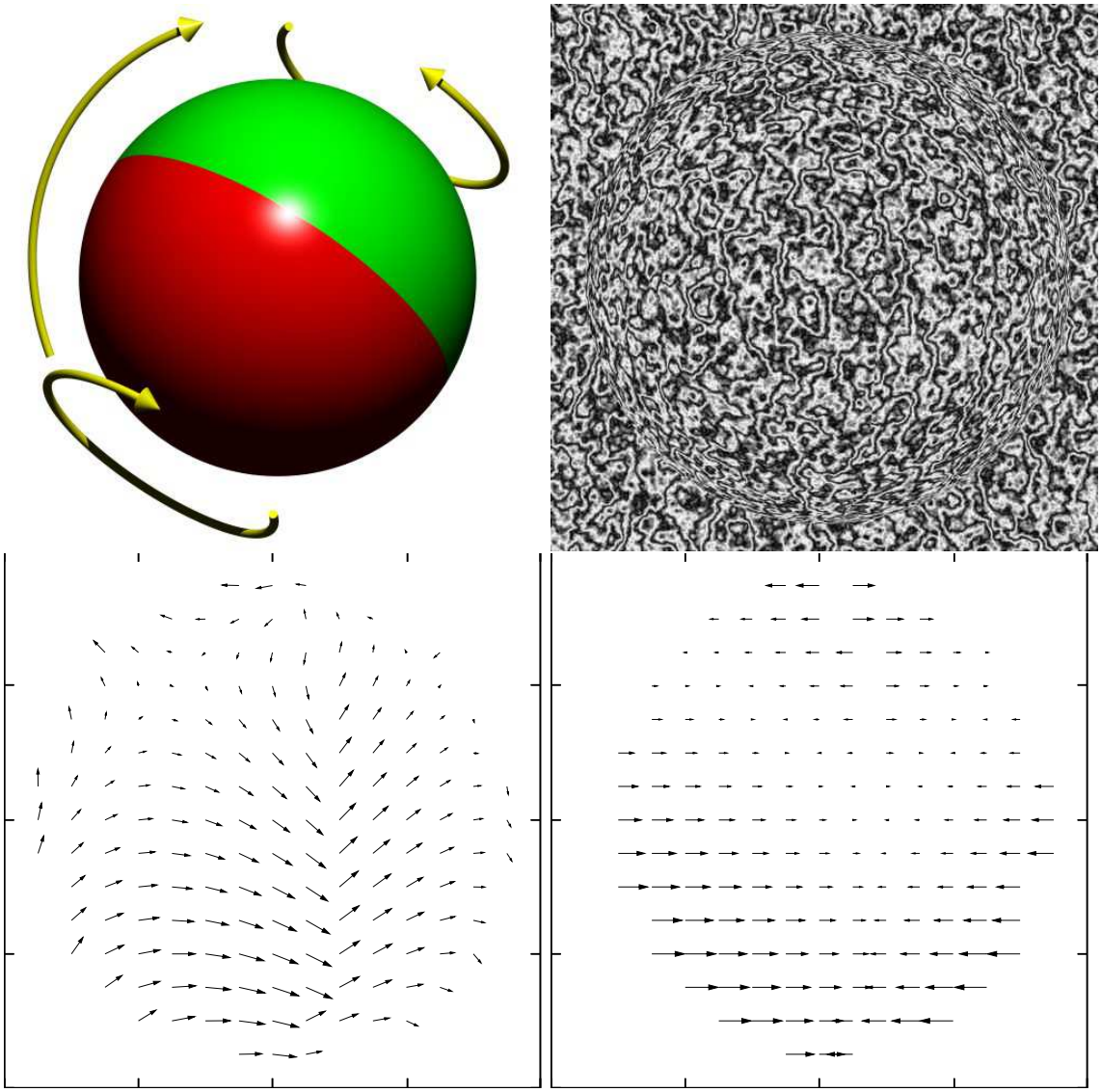

FIGURE 3.10 – La scène synthétique présentée est une sphère texturée en rotation, pour laquelle les deux hémisphères tournent indépendamment dans des sens opposés (haut-bas, image en haut a droite).

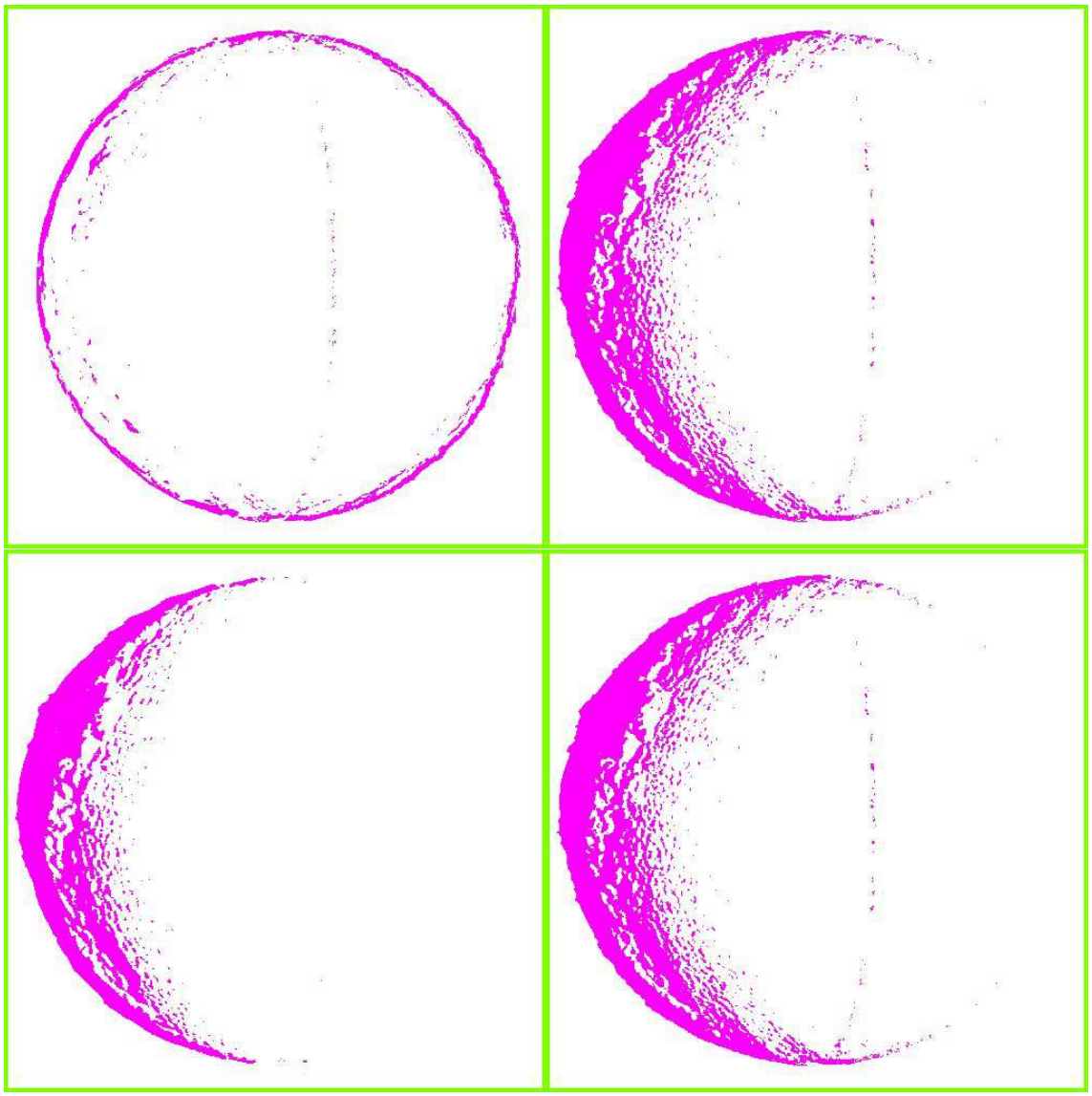

FIGURE 3.11 – Les cartes d'occultation pour les termes d'attache aux données correspondant au flot optique gauche, au flot optique droit, et aux disparités aux temps t et $t+1$. Les pixels occultés sont colorés en rose.

confirment les expériences numériques. Le problème de minimisation de J est ainsi mal posé. Afin de le résoudre numériquement, nous écrivons le système d'équations $\nabla J(\mathbf{W}, d^{'}, d) = 0$. Ce système représente un ensemble de conditions nécessaires pour que J ait un minimum. Ce minimum est unique si et seulement si la fonctionnelle est strictement convexe. A priori ce n'est pas notre cas.

4.1 Les équations d'Euler

La minimisation de l'énergie alors décrite s'effectue alors en utilisant les conditions d'Euler. Lorsque une fonctionnelle J admet un minimum en ϕ, alors nécessairement $\nabla J(\phi) = 0$. Néanmoins, celà n'est une condition suffisante que dans le cas ou J est strictement convexe, auquel cas l'existence et l'unicité d'un minimum sont garanties. Si cette condition n'est pas remplie, J peut admettre plusieurs minima locaux. Celà implique la nécessité d'une initialisation de l'algorithme de minimisation proche du minimum, ce qui est souvent délicat à estimer, ou bien, et c'est la méthode que nous emploierons, cela nécessite l'utilisation d'une stratégie multirésolution (Lefébure and Cohen [2001]). Le système d'équations $\nabla J(\phi) = 0$ est dans la très grande majorité des problèmes concrets, non trivial et non linéaire. Il est parfois nécessaire d'adapter des schémas numériques existants ou bien d'essayer de linéariser le système. Pour simplifier les expressions qui vont suivre introduisons quelques notations :

$$I_g(\mathbf{X}, t) = I_1 \tag{3.35}$$
$$I_g(\mathbf{X} + \mathbf{W}, t + 1) = I_2 \tag{3.36}$$
$$I_d(\mathbf{X} + \mathbf{D}, t) = I_3 \tag{3.37}$$
$$I_d(\mathbf{X} + \mathbf{W} + \mathbf{D}^{'}, t + 1) = I_4 \tag{3.38}$$
$$\tag{3.39}$$

Lors du calcul des dérivées de J on voit apparaître des coefficients pondérateurs liés à chaque sous partie du terme d'attache aux données. Ces coefficients permettent, grace aux propriétés de la fonction Ψ, 'être robuste aux grandes variations des composantes du terme d'attache aux données, en lissant les énergies trop élevées.

$$\Psi^{'}_{g_flot} = \Psi^{'}((I_2 - I_1)^2 + \gamma \cdot (\nabla I_2 - \nabla I_1)^2) \tag{3.40}$$
$$\Psi^{'}_{d_flot} = \Psi^{'}((I_4 - I_3)^2 + \gamma \cdot (\nabla I_4 - \nabla I_3)^2) \tag{3.41}$$
$$\Psi^{'}_{stereo_nextt} = \Psi^{'}((I_4 - I_2)^2 + \gamma \cdot (\nabla I_4 - \nabla I_2)^2) \tag{3.42}$$
$$\Psi^{'}_{stereo-t} = \Psi^{'}((I_3 - I_1)^2 + \gamma \cdot (\nabla I_3 - \nabla I_1)^2) \tag{3.43}$$
$$\Psi^{'}_{div} = \Psi^{'}(|\nabla \mathbf{W}|^2 + \lambda |\nabla (d^{'} - d)|^2 + \mu \cdot |\nabla d|^2) \tag{3.44}$$

L'équation d'Euler-Lagrange correspondant à $\frac{\partial J}{\partial u} = 0$ est de la forme suivante :

$$\Psi'_{g_flot}(\frac{\partial I_1}{\partial x}(I_2 - I_1) + \gamma \cdot (\nabla I_2 - \nabla I_1)\frac{\partial \nabla I_2}{\partial x}) + \Psi'_{d_flot}(\frac{\partial I_3}{\partial x}(I_4 - I_3) + \gamma \cdot (\nabla I_4 - \nabla I_3)$$
$$\frac{\partial \nabla I_4}{\partial x}) + \Psi'_{stereo}((I_4 - I_2)(\frac{\partial I_4}{\partial x} - \frac{\partial I_3}{\partial x}) + \gamma \cdot ((\nabla I_4 - \nabla I_3)\frac{\partial(\nabla I_4 - \nabla I_3)}{\partial x}))-$$
$$\omega \operatorname{div}(\Psi'_{div}\nabla u) = 0 \tag{3.45}$$

L'équation correspondant à la condition $\frac{\partial J}{\partial v} = 0$ est tout à fait similaire à l'équation précédente :

$$\Psi'_{g_flot}(\frac{\partial I_1}{\partial y}(I_2 - I_1) + \gamma \cdot (\nabla I_2 - \nabla I_1)\frac{\partial \nabla I_2}{\partial y}) + \Psi'_{d_flot}(\frac{\partial I_4}{\partial y}(I_4 - I_3) + \gamma \cdot (\nabla I_4 - \nabla I_3)$$
$$\frac{\partial \nabla I_4}{\partial y}) + \Psi'_{stereo}((I_4 - I_2)(\frac{\partial I_4}{\partial y} - \frac{\partial I_2}{\partial y}) + \gamma \cdot ((\nabla I_4 - \nabla I_2)\frac{\partial(\nabla I_4 - \nabla I_2)}{\partial y}))-$$
$$\omega \operatorname{div}(\Psi'_{div}\nabla v) = 0 \tag{3.46}$$

L'équation correspondant à $\frac{\partial J}{\partial d'} = 0$ est de la forme :

$$\Psi'_{d_flot}(\frac{\partial I_4}{\partial x} \cdot (I_4 - I_3) + \gamma \cdot (\nabla I_4 - \nabla I_3) \cdot \frac{\partial \nabla I_4}{\partial x}) + \Psi'_{stereo_nextt}((I_4 - I_2) \cdot \frac{\partial I_4}{\partial x}+$$
$$\gamma \cdot ((\nabla I_4 - \nabla I_2) \cdot \frac{\partial \nabla I_4}{\partial x})) - \omega \cdot \lambda \operatorname{div}(\Psi'_{div}\nabla(d' - d)) = 0 \tag{3.47}$$

Enfin, l'équation provenant de $\frac{\partial J}{\partial d} = 0$ est très similaire, au changement de signe près :

$$\Psi'_{d_flot}(\frac{\partial I_3}{\partial x} \cdot (I_3 - I_4) + \gamma \cdot (\nabla I_3 - \nabla I_4) \cdot \frac{\partial \nabla I_3}{\partial x}) + \Psi'_{stereo_nextt}((I_3 - I_1) \cdot \frac{\partial I_3}{\partial x}+$$
$$\gamma \cdot ((\nabla I_3 - \nabla I_1) \cdot \frac{\partial \nabla I_3}{\partial x})) - \omega \cdot \lambda \operatorname{div}(\Psi'_{div}\nabla(d' - d) + \mu \cdot \nabla d) = 0 \tag{3.48}$$

Résoudre ces équations permet donc de trouver un minimum de J. Expérimentalement les valeurs des différents paramètres ω, γ notamment influent sur la nature du minimum. Certaines valeurs amènent à un minimum local qui ne convient pas au problème (solution trop lisse ou trop irrégulière), il est donc important de trouver expérimentalement les plages de valeurs des paramètres pour accéder à un minimum correct. Cela est précisément l'objet du chapitre 4, lequel développe l'algorithme mis en oeuvre pour résoudre ces équations.

5 Conclusion

Nous avons présenté dans ce chapitre une formulation variationnelle permettant de prendre en compte diverses contraintes dans le problème d'estimation du flot

de scène. L'originalité de notre formulation, ainsi que sa difficulté, proviennent de l'utilisation de distances quadratiques et non plus de contraintes linéarisées comme ce fut le cas précédemment avec l'estimation du flot optique.

D'autre part, l'utilisation de distances quadratiques entre les gradients des images permet à l'algorithme d'être robuste numériquement aux variations d'illumination dans une scène donnée. Nous avons également utilisé une fonction Ψ nous permettant à la fois d'introduire de la robustesse aux occultations et de préserver les discontinuités du flot de scène.

Enfin, notre méthode est la première , à notre connaissance, à gérer simultanément le calcul du flot de scène et celui des occultations. Au final, nous avons à résoudre un système d'équations aux dérivées partielles non linéaires et couplées. Le chapitre suivant introduit un algorithme de résolution original afin de déterminer numériquement les composantes du flot de scène.

Chapitre 4

Algorithme et résultats numériques

Partant de la modélisation présentée dans le chapitre précédent, le cœur de la thèse a constitué en l'élaboration d'un algorithme répondant aux critères précédemment cités :

- être robuste aux variations d'illumination dans la mesure du possible.
- permette de gérer des surfaces non lambertiennes et non fronto parallèles
- pouvant gérer des mouvements rigides et non rigides, quelque soit la nature des objets observés

Nous avons vu précédemment que la modélisation adoptée intègre les variations d'illumination et la capacité à gérer des surfaces non lambertiennes, et que nous n'avons pas d'hypothèses spécifiques sur la rigidité ou non des différents déplacements mis en jeu. Il s'agit maintenant d'analyser l'implémentation de la minimisation de l'énergie modélisant le problème du flot de scène stéréoscopique.

Dans un premier temps, nous allons étudier la manière de déterminer une solution numérique de nos EDP. Nous nous sommes inspirés des travaux de Brox et al. [2004] sur le flot optique afin de traiter les non linéarités, et nous nous sommes également intéressés aux approches multirésolution dans le cas des fonctionnelles non convexes, afin d'éviter de converger vers des solutions inadéquates. Nous faisons apparaître différentes difficultés numériques qui constituent autant de contraintes sur l'algorithme et son implémentation. Celà conduit à présenter un schéma général pour l'algorithme de minimisation. Classiquement, nous montrons ensuite des résultats numériques sur des exemples synthétiques et réels, d'un point de vue quantitatif mais aussi qualitatif pour les cas ou l'on ne dispose pas de la vérité terrain. Enfin, nous évoquons diverses idées d'améliorations possible pour l'algorithme, pour pallier au principal défaut de celui-ci, à savoir un coût relativement élevé en temps de calcul.

1 Approches multirésolution

Il est nécessaire pour comprendre l'algorithme mis en oeuvre de résumer le principe des approches multirésolution dans le cadre de l'optimisation de fonctionnelles. Comme nous l'avons vu dans le chapitre 2, la fonctionnelle J à minimiser n'est

FIGURE 4.1 – Exemple de pyramide gaussienne

certainement pas convexe. Dans ce cas, il existe plusieurs minima locaux dont la plupart ne donneront pas une solution satisfaisante. D'après nos expérimentations numériques, certains minima locaux donnent par exemple une disparité trop lisse, sans discontinuités, et d'autres donnent un flot optique chaotique.

Le principe des approches multirésolution pour la minimisation de fonctionnelles est résumé et justifié par Lefébure and Cohen [2001]. La fonctionnelle J n'est pas convexe si l'on considère les images à leur résolution originale. On peut néanmoins montrer que pour les images de moindre résolution obtenues par sous-échantillonnages successifs des images originales, certains minima locaux de J disparaissent. Le processus de sous-échantillonnage est donc logiquement itéré jusqu'à ce que J n'admette plus qu'un seul minimum global. Lefébure et Cohen montrent ainsi que ce processus revient à "convexifier" l'énergie J. Notons N le nombre de sous-échantillonages nécessaires pour convexifier J. Dans le cas du flot de scène stéréoscopique, celà revient donc à créer $N-1$ paires stéréo de résolution décroissante à partir des paires sétéro originales. L'ensemble des N paires d'images ainsi constituées constituent une pyramide gaussienne de paires stéréo. La figure 4.1 illustre un exemple de pyramide destinée par exemple au calcul du flot optique. La dénomination "gaussienne" provient du fait que chaque sous-échantillage est suivi d'un lissage des images obtenues par une gaussienne, afin d'éliminer le bruit introduit.

De manière générale et dans toute la suite, les étages de la pyramide, correspondant à un niveau de détails, sont numérotés de 1 à N de telle sorte que le

niveau N correspond à la résolution la plus grossière. L'un des paramètres essentiels pour la construction d'une pyramide gaussienne est le choix du facteur de sous-échantillonnage η. Tradionnellement dans la littérature η est compris entre 0.7 et 0.95. Le nombre de niveaux de détails constituant la pyramide dépend bien sûr directement de la valeur de ce paramètre. De manière générale un η élevé permet de mieux affiner les estimations successives de la solution au prix d'un temps de calcul augmenté. Le schéma global de la minimisation de J par une approche multirésolution est décrit par l'algorithme 2.

Algorithme 2 Minimisation de J par une approche multirésolution

Sorties:

Entrées: Une paire d'images stéréo $P_1 = (I_g(.,t), I_d(.,t)$ initiale, un facteur de sous-échantillonnage η, la dimension de la pyramide gaussienne N.

pour $l = 2$ à N **faire**

Sous-échantillonner P_{l-1} d'un facteur η et obtenir la paire d'images stéréo P_l

Appliquer un lissage gaussien à P_l

fin pour

Initialiser les valeurs de $(U_N, V_N, D_N(.,t), D_N(.,t+1))$, composantes du flot de scèné à la résolution la plus grossière N.

pour $l = N$ à 1 **faire**

Estimer les valeurs de la fonctionnelle J_l définie au niveau l en utilisant P_l

Minimiser J_l et trouver $(U_l, V_l, D_l(.,t), D_l(.,t+1))$ estimation de la solution au niveau de résolution l

Interpoler les valeurs $(U_l, V_l, D_l(.,t), D_l(.,t+1))$ au niveau $l-1$ en utilisant le facteur $\frac{1}{\eta}$. Appliquer un lissage gaussien sur la solution interpolée notée S_{l-1}.

Initialisation de la résolution des EDP au niveau $l-1$: $(U_{l-1}, V_{l-1}, D_{l-1}(.,t), D_{l-1}(.,t+1)) = S_{l-1}$.

fin pour

La minimisation débute par la construction de la pyramide gaussienne d'images. Une fois ceci réalisé, les inconnues du problème sont initialisée et la fonctionnelle J est estimée au niveau de résolution le plus grossier. Son estimation à ce niveau est notée J_N. D'apres le travail de Lefébure and Cohen [2001], si N est suffisamment grand, J_N est convexe. En pratique, on choisit N pour que le flot optique au dernier niveau de la pyramide soit d'amplitude inférieure au pixel : $||\mathbf{W}||_\infty < 0.5$. Dans ce cas là, l'initialisation importe peu, une descente de gradient suffit pour trouver son minimum global. Une fois ceci réalisé, on itère en parcourant la pyramide en direction du niveau de résolution original, et en initialisant la résolution des EDP avec la solution estimée au niveaux précedents. Dans ces conditions, le minima obtenu en fin de processus est un ou le minimum global cherché. Il est important de noter que cette stratégie multirésolution peut être combinée à différentes approches multigrilles.

– Les approches de "cascading multigrid" , ou méthode multigrille monodirec-

tionnelle, correspondent aux méthodes multirésolutions basiques (Kornprobst [1998]).
- les approches bidirectionnelles, avec notamment des applications au calcul temps réel du flot optique. On peut citer les travaux de Bruhn et al. [2005a, 2003, 2005b]. A chaque niveau de résolution, on redescend au niveau de détail inférieur afin de procéder àune estimation de l'erreur sur la solution, erreur par la suite interpolée au niveau de résolution courant, ce qui permet de corriger la solution estimée. Ainsi, á chaque niveau de résolution, la solution est rendue plus robuste à l'aide de ces allers-retours entre niveaux de détails (dont la profondeur peut être en fait arbitrairement fixée. Il faut noter qu'une implémentation sur GPU des méthodes multigrilles a été réalisée récemment par Grossauer and Thoman [2008].

2 Résolution numérique des EDP

L'énergie J n'est malheureusement pas convexe, ceci étant dû à la non linéarité des termes de l'énergie. Celà rend le problème mal posé au sens où il n'y pas existence d'un minimum global unique. Dans le but de contourner cette difficulté numérique, nous utilisons une approche multirésolution incrémentale, avec des itérations de point fixe imbriquées sur l'estimation courante de la solution (u, v, d, d') afin de l'affiner à chaque niveau de résolution. Les pyramides d'images stéréo sont estimées avec un facteur de sous-échantillonnage η, $0.5 < \eta < 1$ pour obtenir une transition régulière entre les niveaux des pyramides (nous utilisons en général $\eta = 0.9$). L'approche multirésolution assure en pratique la convergence vers un minimum global (pas forcément unique), comme cela est montré par Lefébure and Cohen [2001]. Cette méthode a prouvé son efficacité sur de nombreux problèmes et a été récemment améliorée pour obtenir des performances temps réel sur des séquences de résolution raisonnable, par une implémentation GPU avec le langage CUDA par Pauwels and Van Hulle [2008].

2.1 Principes

Le terme de fidélité aux données dans les équations d'Euler-Lagrange est composé de valeurs d'images et de gradients d'images évalués par rapport à l'image de référence $I_g(., t)$. Ceci est équivalent à warper les 3 autres images ($I_g(., t+1)$, $I_d(., t)$ et $I_d(., t+1)$) issues du même niveau dans les pyramides, vers l'image $I_g(., t)$, en utilisant l'estimation courante de la solution (u, v, d, d'). Les termes de l'énergie s'évaluent ainsi à partir de ces images warpées et de leurs gradients.

À un niveau de résolution donné, nous traitons les non linéarités des équations en utilisant des itérations de point fixe imbriquées, en l'occurence au nombre de 2 (voir Annexe B pour les développements formels justificatifs), obtenues en réalisant un développement de Taylor au premier ordre des équations d'Euler Lagrange. Ceci aboutit à la résolution d'un système linéaire pour chaque itération de l'itération

de point fixe de bas niveau. Cette dernière évalue ainsi de petits incréments de la solution $(\delta u, \delta v, \delta d, \delta d^{'})$, et les images sont rewarpées vers l'image de référence en utilisant la transformation $(u + \delta u, v + \delta v, d + \delta d, d' + \delta d^{'})$ à chaque itération. L'itération de point fixe de haut niveau met à jour la solution totale $(u, v, d, d^{'})$, dès que l'itération de bas niveau a convergé, et permet aussi de rewraper les images en utilisant la nouvelle estimation $(u, v, d, d^{'})$. L'itération de point fixe de bas niveau utilise une méthode SOR modifiée pour résoudre le système linéaire final.

Dans la méthode SOR traditionnelle, décrite par l'algorithme 3, la matrice du système est séparée en trois parties : diagonale, triangulaire supérieure et inférieure. Par conséquent, des ordonnancements différents des lignes et des colonnes du système (tout dépend de la manière dont on organise les pixels de l'image pour l'écriture du système) peuvent aboutir à des résultats différents à chaque itération. Notre implémentation, résumée par l'algorithme 4, utilise alternativement 4 ordonnancements différents, dans lesquels les pixels sont parcourus dans 4 différentes directions, dans le but de réduire l'asymétrie introduite par chaque itération de SOR. Etant donné l'orientation des artefacts numériques, nous avons choisi de parcourir la matrice du système suivant une orientation à 45° : les lignes et colonnes sont balayées diagonalement du haut vers le bas puis dans le sens inverse, pour les deux orientations diagonales. Il en résulte 4 sous-itérations de SOR consécutives pour aboutir à une itération minimisant la dépendance à l'organisation des données des images dans la structure matricielle. Ce problème d'asymétrie n'est pas visible dans le cas de l'estimation du flot optique seul, mais tend à induire des ondes orientées dans la solution numérique du flot de scène. Les systèmes linéaires à résoudre dans le cas du flot de scène sont, pour un niveau de détail donné, 4 fois plus grands et tout aussi creux que les systèmes apparaissant dans le problème du flot optique. Celà suggère un conditionnement probablement encore plus médiocre et par conséquent une plus grande sensibilité à la méthode de résolution.

Les conditions d'arrêt pour les itérations de point fixe sont estimées à partir des normes L^2 relatives entre des incréments consécutifs. Nous avons utilisé 0.05 comme condition d'arrêt pour les itérations de bas niveau et 0.01 pour celles de haut niveau. Une fois que la minimisation de J est effectuée à un niveau de résolution donné, la solution est multipliée par $1/\eta$, suréchantillonnée au niveau de résolution suivant, et le même processus est itéré jusqu'à ce que la résolution finale des images soit atteinte. Le pseudo-code de la méthode de résolution est présenté à l'aide de l'algorithme **??**.

2.2 Discrétisation des équations d'Euler Lagrange

Pour résoudre nos équations, il est nécessaire de trouver un schéma numérique robuste et si possible inconditionnellement stable. Chaque équation contient un terme de fidélité aux données et un terme diffusif anisotrope. Pour un point de la scène observée donné, le terme d'attache aux données est approximé par un développe-

Algorithme 3 Méthode SOR originale

Sorties:

Entrées: Matrice $A = (a_{ij}) \in M_{N,N}(\Re)$ du système linéaire, vecteur second membre $b = (b_i) \in \Re^N$. $x \in \Re^N$ vecteur inconnue du système. ϵ étant l'erreur sur la norme à obtenir, K_{max} le nombre maximal d'itérations. r^k désigne l'erreur à la $k$ième itération. ω désigne le paramètre de relaxation de la méthode.

Initialiser x.

k = 0 ;

$r^0 = A \cdot x^0 - b$

tantque $r^k < \epsilon$ et $k < K_{max}$ **faire**

  pour $i = 1$ à N **faire**

    $x_i^{k+1} = (1-\omega) \cdot x_i^k + \frac{\omega}{a_{ii}} \cdot (b_i - \sum_{j=1}^{\hat{i}-1} a_{ij} \cdot x_j^{k+1} - \sum_{j=i+1}^{N} Na_{ij} \cdot x_j^k)$

  fin pour

  $r^k = A \cdot x^k - b$

  $k = k + 1$

fin tantque

Algorithme 4 Méthode SOR modifiée

Sorties:

Entrées: Matrice $A = (a_{ij}) \in M_{N,N}(\Re)$ du système linéaire, vecteur second membre $b = (b_i) \in \Re^N$. $x \in \Re^N$ vecteur inconnue du système. ϵ étant l'erreur sur la norme à obtenir, K_{max} le nombre maximal d'itérations. r^k désigne l'erreur à la $k$ième itération. ω désigne le paramètre de relaxation de la méthode. 4 orientations de parcours des termes de la matrice A : Nord Ouest - Sud Est (NO-SE), Sud-Est - Nord Ouest (SE-NO), Nord Est - Sud Ouest (NE-SO), Sud Ouest - Nord Est (SO-NE).

Initialiser x.

k = 0 ;

$r^0 = A \cdot x^0 - b$

tantque $r^k < \epsilon$ et $k < K_{max}$ **faire**

  pour Orientation $\in$ NO-SE, SE-NO, NE-SO, SO-NE **faire**

    Initialisation des index de la matrices suivant les orientations : ligne-debut, ligne-fin, colonne-debut, colonne-fin

    pour $p =$ ligne-debut à ligne-fin **faire**

      pour $q =$ colonne-debut à colonne-fin **faire**

        i = p + q

        $x_i^{k+1} = (1-\omega) \cdot x_i^k + \frac{\omega}{a_{ii}} \cdot (b_i - \sum_{j=1}^{\hat{i}-1} a_{ij} \cdot x_j^{k+1} - \sum_{j=i+1}^{N} Na_{ij} \cdot x_j^k)$

      fin pour

    fin pour

  fin pour

  $r^k = A \cdot x^k - b$

  $k = k + 1$

fin tantque

Algorithme 5 Résolution des EDP à l'aide d'une stratégie multirésolution

Sorties:

Entrées: 4 pyramides gaussiennes issues de deux paires stéréo consécutives : $I_g(.,t)$, $I_d(.,t)$, $I_g(.,t+1)$, $I_d(.,t+1)$. Le nombre de niveaux de ces pyramides, N. Le facteur de sous-échantillonnage correspondant η. Critères d'arrêt : erreurs ϵ_1, ϵ_2, ϵ_3, nombres d'itérations maximums N_1, N_2,N_3. Constante $C > 0$ telle que : $C > \epsilon_1$, $C > \epsilon_2$, $C > \epsilon_3$. Erreurs sur les itérations successives : externe e_1, interne e_2, sor e_3. Nombres d'itérations effectuées correspondants : i_1,i_2, i_3.

Initialisation de $(u^0, v^0, d^0, d^{',0})$

pour $l = N$ à 1 **faire**

  $e_1 = C$, $i_1 = 0$

  tantque $e_1 > \epsilon_1$ et $i_1 < N_1$ **faire**

    $e_2 = C$, $i_2 = 0$

    Initalisation des incréments

    $\delta u^0 \leftarrow 0$, $\delta v^0 \leftarrow 0$, $\delta d^0 \leftarrow 0$, $\delta d^{',0} \leftarrow 0$

    tantque $e_2 > \epsilon_2$ et $i_2 < N_2$ **faire**

      Résolution du système linéaire induit par les EDP discrétisées

      $e_3 = C$, $i_3 = 0$

      tantque $e_3 > \epsilon_3$ et $i_3 < N_3$ **faire**

        Itération de SOR modifié sur la matrice du système linéaire

        $i_3 \leftarrow i_3 + 1$

      fin tantque

      Update de δu^{i_2}, δv^{i_2}, δd^{i_2}, $\delta d^{'i_2}$

      $i_2 \leftarrow i_2 + 1$

    fin tantque

    Ajout des incréments aux estimations courantes des inconnues

    $u^{i_1} \leftarrow u^{i_1-1} + \delta u^{i_2}$, $v^{i_1} \leftarrow v^{i_1-1} + \delta v^{i_2}$, $d^{i_1} \leftarrow d^{i_1-1} + \delta d^{i_2}$, $d^{'i_1} \leftarrow d^{'i_1-1} + \delta d^{'i_2}$

    $e_1 = ||(u^{i_1}, v^{i_1}, d^{i_1}, d^{'i_1}) - (u^{i_1-1}, v^{i_1-1}, d^{i_1}, d^{'i_1-1})||_2$

    $i_1 \leftarrow i_1 + 1$

  fin tantque

  Interpolation par un facteur $\frac{1}{\eta}$ de l'estimation $(u, v, d, d^{'})$ au niveau de résolution suivant

fin pour

ment de Taylor d'ordre 1 au voisinage du projeté de ce point dans l'image de référence (noté dans ce paragraphe $P_{i,j}$), ce qui aboutit à des expressions fonctions de $(\delta u, \delta v, \delta d, \delta d^{'})$. Par exemple, $I_g(x+u+\delta u, y+v+\delta v)$ est remplacé par l'expansion de Taylor $I_g(x+u, y+v) + \partial_X I_l \cdot \delta u + \partial_Y I_l \cdot \delta v$.

En ce qui concerne le terme de diffusion, nous utilisons un schéma à 4 points utilisant les voisins gauche, droit, haut et bas de $P_{i,j}$ ou bien un schéma à 8 points utilisant les voisins diagonaux. Ces schémas utilisent les valeurs des fonctions $(u, v, d, d^{'})$ interpolées au milieu des segments joignant $P_{i,j}$ à ses plus proches voisins ($P_{i+\frac{1}{2},j}$, $P_{i-\frac{1}{2},j}$, $P_{i,j+\frac{1}{2}}$ and $P_{i,j-\frac{1}{2}}$ dans le cas du schéma 4 points). En utilisant les opérateurs aux différences finies concernant ces points, notés $\delta_x^{+\frac{1}{2}}$, $\delta_x^{-\frac{1}{2}}$, $\delta_y^{+\frac{1}{2}}$, $\delta_y^{-\frac{1}{2}}$, nous obtenons une approximation numérique pour le terme de divergence estimé au pixel $P_{i,j}$:

$$\begin{aligned} div(b\nabla\phi)|_{i,j} \approx \frac{1}{h^2}(&b_{i+\frac{1}{2},j}\phi_{i+1,j} + b_{i-\frac{1}{2},j}\phi_{i-1,j} + b_{i,j+\frac{1}{2}}\phi_{i,j+1} + b_{i,j-\frac{1}{2}}\phi_{i,j-1} \\ &- (b_{i+\frac{1}{2},j} + b_{i-\frac{1}{2},j} + b_{i,j+\frac{1}{2}} + b_{i,j-\frac{1}{2}})\phi_{i,j}) \end{aligned} \tag{4.1}$$

Cette expression est alors utilisée pour discrétiser les expressions de la forme $\mathrm{div}(\phi+\delta\phi)$ pour chaque variable fonctionnelle $(u, v, d, d^{'})$. Au final, nous regroupons les expressions utilisant les incréments inconnus $(\delta u, \delta v, \delta d, \delta d^{'})$ et nous aboutissons (voir les détails en annexe) à un grand et extrêmement creux système linéaire $Ax = b$ dont la matrice principale possède la structure par blocs suivante :

$$A = \begin{array}{|c|c|c|} \hline A_1 & -A_1 & 0_{4\cdot N_X} \\ \hline A_3 & A_2 & A_4 \\ \hline 0_{4\cdot N_X} & A_1 & -A_1 \\ \hline \end{array}$$

Notons N_X et N_Y les dimensions (en pixels) de l'image de référence. $0_{4\cdot N_X}$ est la matrice carrée nulle de dimension $4 \cdot N_X$. Alors, parce qu'il y a 4 équations par pixel de l'image de référence, $A \in M_{4\cdot N_X\cdot N_Y, 4\cdot N_X\cdot N_Y}(\Re)$. $A_1 = \frac{1}{h} Id_{4\cdot N_X}$, où $Id_{4\cdot N_X}$ est la matrice identité de dimension $4 \cdot N_X$, est la sous-matrice associée aux conditions aux bords sur les cotés horizontaux de l'image, et A_2 est la sous-matrice associéeà tous les autres pixels de l'image de référence. A_2 possède la structure par blocs suivante :

$$A_2 = \begin{array}{|c|}\hline B \\ \hline C \\ \hline \vdots \\ \hline C \\ \hline B \\ \hline C \\ \hline \vdots \\ \hline \end{array}$$

Les sous-matrices B et C alternent à l'intérieur de la structure de A_2. B est associée aux pixels situés sur les bords verticaux de l'image de référence, et appartient à l'espace $M_{4,4\cdot N_X\cdot(N_Y-2)}(\Re)$ (4 lignes car il y'a 4 équations par pixel, et $4\cdot N_X\cdot(N_Y-2)$ est le nombre total d'inconnues associées aux pixels situés ailleurs que sur les bords horizontaux). B a la forme suivante :

$$B = \begin{array}{|c|c|c|c|c|c|c|c|}\hline 0 & \cdots & 0 & \frac{1}{h}Id_4 & -\frac{1}{h}Id_4 & 0 & \cdots & 0 \\ \hline \end{array}$$

si le pixel pour lequel B représente les équations discrétisées est situé sur le coté gauche ou

$$B = \begin{array}{|c|c|c|c|c|c|c|c|}\hline 0 & \cdots & 0 & -\frac{1}{h}Id_4 & \frac{1}{h}Id_4 & 0 & \cdots & 0 \\ \hline \end{array}$$

si le pixel est sur le coté droit.

La matrice C représente les équations discrétisées aux pixels localisés à l'intérieur de l'image. Pour un pixel intérieur donné, $C \in M_{4,4\cdot N_X\cdot(N_Y-2)}(\Re)$, comme B. Cette matrice est composée des coefficients associés aux inconnues $(\delta u, \delta v, \delta d, \delta d^{'})$, estimés lors de l'approximation de Taylor des équations d'Euler Lagrange.

Pour un pixel intérieur donné $P_{i,j}$, $C = (C_{i,j})$ a la structure par blocs suivante :

$$C_{i,j} = \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|}\hline 0 & \cdots & D_{1,i,j} & 0 & \cdots & D_{2,i,j} & D_{5,i,j} & D_{3,i,j} & 0 & \cdots & D_{4,i,j} & 0 & \cdots \\ \hline \end{array}$$

où

$$D^T_{1,i,j} = \begin{pmatrix} -\omega b^1_{i-1,j} & -\omega b^1_{i-1,j} & -\omega\lambda b^1_{i-1,j} & -\omega(\lambda+\mu) b^1_{i-1,j} \end{pmatrix}$$

$$D^T_{2,i,j} = \begin{pmatrix} -\omega b^1_{i,j-1} & -\omega b^1_{i,j-1} & -\omega\lambda b^1_{i,j-1} & -\omega(\lambda+\mu) b^1_{i,j-1} \end{pmatrix}$$

$$D_{3,i,j}^T = \begin{pmatrix} -\omega b_{i,j+1}^1 & -\omega b_{i,j+1}^1 & -\omega\lambda b_{i,j+1}^1 & -\omega(\lambda+\mu) b_{i,j+1}^1 \end{pmatrix}$$

et

$$D_{4,i,j}^T = \begin{pmatrix} -\omega b_{i+1,j}^1 & -\omega b_{i+1,j}^1 & -\omega\lambda b_{i+1,j}^1 & -\omega(\lambda+\mu) b_{i+1,j}^1 \end{pmatrix}$$

ce qui correspond, pour les 4 équations définies en $P_{i,j}$, aux coefficients du terme de divergence discrétisé et estimé aux plus proches voisins (voir eq. (4.1)).

$D_{5,i,j}$ est estimée en $P_{i,j}$ en évaluant les coefficients des équations relatifs à $(\delta u, \delta v, \delta d, \delta d')$ et en leur ajoutant les termes coefficients du terme diffusif en $P_{i,j}$. Cette matrice a la forme générique :

$$D_{5,i,j} = \begin{pmatrix} a_{\delta u}^1 & a_{\delta v}^1 & a_{\delta d'}^1 & a_{\delta d}^1 \\ a_{\delta u}^2 & a_{\delta v}^2 & a_{\delta d'}^2 & a_{\delta d}^2 \\ a_{\delta u}^3 & a_{\delta v}^3 & a_{\delta d'}^3 & a_{\delta d}^3 \\ a_{\delta u}^4 & a_{\delta v}^4 & a_{\delta d'}^4 & a_{\delta d}^4 \end{pmatrix}$$

Les sous-matrices A_3 et A_4 ont l'élément 0 presque partout excepté sur les lignes associées aux pixels de l'image de référence possédant des voisins sur les bords horizontaux. Dans ce cas, les éléments non nuls de ces sous-matrices sont $D_{1,i,j}$ ou $D_{4,i,j}$. Cette sous-matrice n'est pas symétrique (par exemple dans le cas général, $a_{\delta v}^1$ et $a_{\delta u}^2$ ne sont pas égaux) et c'est la raison pour laquelle A n'est pas symétrique. Chaque ligne de A comporte tout au plus 8 éléments non nuls, dans le cas des pixels à l'intérieur de l'image de référence, et seulement 2 pour les pixels du bord. Ainsi, le système linéaire final à résoudre pour estimer les composantes du flot de scène, appartient à la classe de systèmes linéaires les plus difficiles à résoudre : la matrice est extrêmement creuse et non symétrique définie positive.

3 Algorithme

Nous présentons tout d'abord le schéma global de l'algorithme, puis nous discutons des difficultés numériques rencontrées, des choix effectués pour palier à ces dernières, et nous présentons ensuite des résultats numériques.

3.1 Schéma général

Introduisons tout d'abord quelques notations :

- Soit $\mathbf{W_d} = (u_d, v_d)$ le flot optique estimé sur les séquences $I_d(., t)$. Les images étant rectifiées, il faut noter que v_d est identique à la composante verticale v du flot optique estimé sur les séquences $I_g(., t)$, nous noterons donc $\mathbf{W_d} = (u_d, v)$.
- Soit $J_g = \int_\Omega |I_g(\mathbf{X}+\mathbf{W}, t+1) - I_g(\mathbf{X}, t)|^2 d\mathbf{X} + \alpha \cdot \int_\Omega \Psi(|\nabla \mathbf{W}|^2) d\mathbf{X}$ la fonctionnelle J réduite à la contrainte du flot optique évaluée entre les images $I_g(., t)$ et $I_g(., t+1)$. Le terme de régularisation ne s'appliquant ici qu'au gradient du flot optique.

- Soit $J_d = \int_\Omega |I_d(\mathbf{X}+\mathbf{W}, t+1) - I_d(\mathbf{X}, t)|^2 d\mathbf{X} + \alpha \cdot \int_\Omega \Psi(|\nabla \mathbf{W_d}|^2) d\mathbf{X}$ la fonctionnelle J réduite à la contrainte du flot optique évaluée entre les images $I_d(., t)$ et $I_d(., t+1)$.
- Soit $J_{stereo} = \int_\Omega |I_d(\mathbf{X}+\mathbf{D}, t) - I_g(\mathbf{X}, t)|^2 d\mathbf{X} + \alpha \cdot \int_\Omega \Psi(|\nabla \mathbf{D}|^2) d\mathbf{X}$ la fonctionnelle J réduite à la contrainte de stéréo entre les images à l'instant t.

L'algorithme proposé a la forme décrite en 6.

Algorithme 6 Estimation du flot de scène stéréoscopique

Sorties: Calculer les composantes du flot de scène stéréoscopique $(u, v, d(., t), d(., t+1))$ entre t et $t+1$ en utilisant une pyramide gaussienne de paires stéréo (chaque pyramide possède N_1 niveaux)

Entrées: $N_1, N_2, N_3 \in \mathbb{N}, N_1 > N_2 >= 1, N_1 > N_3 >= N_2 >= 1$

$u_g \leftarrow 0$, $v \leftarrow 0$, $u_d \leftarrow 0$

pour $l = N_1$ à N_2 **faire**

Minimiser $J_g(u_g, v)$ au niveau l

Minimiser $J_d(u_d, v)$ au niveau l

fin pour

u_g, u_d, v constituent une estimation du flot optique sur les séquences I_g et I_d à la résolution correspondant au niveau N_2 de la pyramide gaussienne.

Initialisation de la disparité d avec un algorithme externe par exemple la méthode de Felzenszwalb and Huttenlocher [2006], suivie d'un sous-échantillonage jusqu'au niveau N_3 de la pyramide.

pour $l = N_3$ à N_2 **faire**

Minimisation de J_{stereo} avec initialisation de l'optimisation par la valeur de d calculée à l'étape précédente. Permet d'affiner l'estimation de $d(., t)$

fin pour

$d^{'}$ est estimée au niveau de résolution N_2 en utilisant la relation $d^{'} = d + u_d - u_g$.

pour $l = b$ à 1 **faire**

Minimisation de J_l estimation de J au niveau l. On en déduit les composantes du flot de scène $(u, v, d, d^{'})$ au niveau l.

On initialise la minimisation de J_{l-1} avec l'estimation $(u_l, v_l, d_l, d_l^{'})$ interpolée au niveau $l-1$

fin pour

Reconstruction du flot de scène en utilisant ses composantes estimées et les paramètres du système stéréo.

Le problème étant fortement non linéaire et non convexe, l'algorithme de résolution doit, dans certains cas, être soigneusement initialisé afin d'éviter des minima locaux éventuellement présents au niveau de résolution le plus grossier. Dans le cas de l'estimation du flot optique, et spécifiquement lorsqu'on utilise un algorithme multirésolution, la résolution minimale utilisée peut être aussi petite que possible, et le flot optique est alors habituellement initialisé à 0. La raison de ce choix est que le flot optique est généralement petit rapporté aux dimensions des images, et il

est aisé de trouver une résolution d'image de départ telle que l'amplitude maximale du flot optique sous-échantillonné soit inférieure à 0.5 pixel, ce qui est en pratique suffisant pour assurer la convergence vers le minimum global.

Dans le problème du flot de scène, nous avons un problème mixte : si nous considérons chaque caméra séparément, cette situation est analogue au problème du flot optique, mais nous essayons de résoudre simultanément un problème de mise en correspondance stéréo entre les images de gauche et de droite. Les caractéristiques d'un problème de stéréo sont très différentes de celles du flot optique : l'amplitude de la disparité est généralement comparable à la taille des images (et même plus grande que la taille des objets de la scène vus dans les images), et de plus il existe beaucoup de zones occultées. Pour cette raison, les approches multirésolution sont peu efficaces sur les problèmes stéréo si elles démarrent l'optimisation à un niveau de résolution trop grossier. Notre méthode souffrirait également de cet inconvénient dans cette situation.

Par conséquent, nous avons choisi de démarrer l'estimation du flot de scène à un niveau de résolution intermédiaire, et d'initialiser les inconnues (u, v, d, d') avec des valeurs judicieuses. Tout d'abord, dans les cas où il n'est pas possible d'initialiser la stéréo à 0, nous initialisons d avec un algorithme de l'état de l'art (Felzenszwalb and Huttenlocher [2006] ou bien suivant les cas la méthode par corrélation de Devernay [1997]) qui calcule la disparité aux images de pleine résolution (niveau 1 des pyramides d'images). L'erreur commise sur la disparité d'un algorithme stéréo donné peut être aisément estimée en utilisant les benchmarks standards (Scharstein and Szeliski [2002]), et nous estimons ensuite le niveau de pyramide b tel que l'erreur RMS (root mean square, moyenne quadratique) en disparité sous-échantillonnée se situe en dessous de 0.5 pixels. Nous estimons aussi un niveau de pyramide a, plus grand que b (correspondant donc à un niveau de résolution plus grossier), tel que l'amplitude maximale du flot optique à ce niveau soit en dessous de 0.5 pixels. Nous résolvons alors le problème du flot optique (en gardant les termes de l'énergie J en relation avec le flot optique) séparément sur les séquences gauche et droite, du niveau a au niveau b, et nous obtenons une estimation initiale pour les flots optiques gauche (u, v) et droite (u', v'). La disparité d à t est alors affinée entre un certain niveau de pyramide c et le niveau b (c supérieur ou égal à b)., utilisant la même méthode et gardant seulement les termes de E relatifs à la stéréo à la date t. $d^{'}$ est initialisée en additionnant à d la différence entre u' et u, et en warpant le résultat vers I_l au temps t (les détails sont donnés dans Algorithme 6). Enfin, l'algorithme d'estimation du flot de scène est appliqué aux 4 images, du niveau de pyramide b au niveau 1 de la pyramide (résolution originale des images). L'algorithme d'estimation du flot de scène, incluant la phase d'initialisation, est détaillé dans Algorithme 6. L'implémentation de l'algorithme est brièvement décrite en Annexe C.

4 Résultats numériques

Alors qu'il existe de nombreuses séquences d'images de validation avec vérité terrain pour divers algorithmes en vision par ordinateur, le problème du flot de scène ne bénéficie pas encore de benchmarks reconnus. Néanmoins, de telles bases de données de validation existent pour les sous-problèmes du flot de scène : le flot optique et la stéréo. Le benchmark standard pour le flot optique fut pendant longtemps la séquence Yosemite (`http://www.cs.brown.edu/~black/images.html`), une séquence de vol sur un paysage rendu par ray tracing, avec les vérités terrain en flot optique et en profondeur. Malheureusement, pour le moment la séquence n'est associée qu'à une seule caméra. Une séquence obtenue par une seconde caméra pourrait être générée en utilisant l'information de profondeur pour warper la première séquence d'images, mais la qualité ne serait pas très bonne et toutes les zones occultées seraient manquantes. Néanmoins la base de données Middlebury s'est etoffée de nouvelles séquences de tests pour les algorithmes d'estimation du flot optique, grâce au travail de Baker et al. [2007]. Les séquences de tests générées, utilisent divers objets et diverses textures, en essayant de fournir un jeu de données permettant d'évaluer les algorithmes pour un maximum de critères (robustesse à l'absence d'informations, robustesse aux occultations, gestion de mouvements non rigides). La vérité terrain associée à ces séquences a été générée en utilisant des textures fluorescentes. Elles sont disponibles à l'adresse `http://vision.middlebury.edu/flow/data/` et font aujourd'hui référence dans le domaine de l'évaluation des algorithmes de flot optique.

Pour le problème stéréo, plusieurs séquences de validation sont disponibles, chacune consistant en 8 vues d'une même scène, dans lesquelles tous les centres optiques sont alignés et espacés régulièrement, et les images sont rectifiées (voir à ce sujet Scharstein and Szeliski [2002]). Il se trouve que ces images peuvent être utilisées pour évaluer un algorithme de flot de scène, en imaginant 2 caméras rectifiées observant une scène statique, et translatées dans la direction du segment joignant leurs centres optiques. Toutes ces images sont présentes dans les bases de données de validation stéréo. Néanmoins, elles représentent des configurations spéciales pour le flot de scène, puisque la composante flot optique est rigoureusement horizontale ($v = 0$), et la disparité reste constante ($d' = d$), mais puisque notre algorithme n'a aucune connaissance à priori de la nature de ce mouvement, celà reste encore un bon benchmark. Nous avons pris les images 2 et 6 des séquences Venus 4.8, Teddy 4.4 et Cones **??** comme paire stéréo à la date t, et les images 4 et 8 comme paire stéréo au temps $t + 1$. La vérité terrain est donnée pour la disparité entre les images 2 et 6, et la vérité terrain du flot optique est celle de la disparité divisée par 2.

4.1 Exemples synthétiques

Afin d'évaluer notre algorithme sur un flot de scène plus général, nous avons également généré des images synthétiques d'une sphère en rotation (Fig. 4.2). Cette scène représente un cas extrême où une reconstruction 3D ne donnera aucune information sur ce qu'il se passe dans la scène, et toute l'information est contenue dans

le flot de scène : puisque la sphère est en rotation, la reconstruction reste identique au fil du temps. En outre, les hémisphères tournent dans des sens opposés, ce qui génère une forte discontinuité dans le flot de scène, et nous vérifions que l'algorithme est capable de retrouver cette discontinuité avec précision.

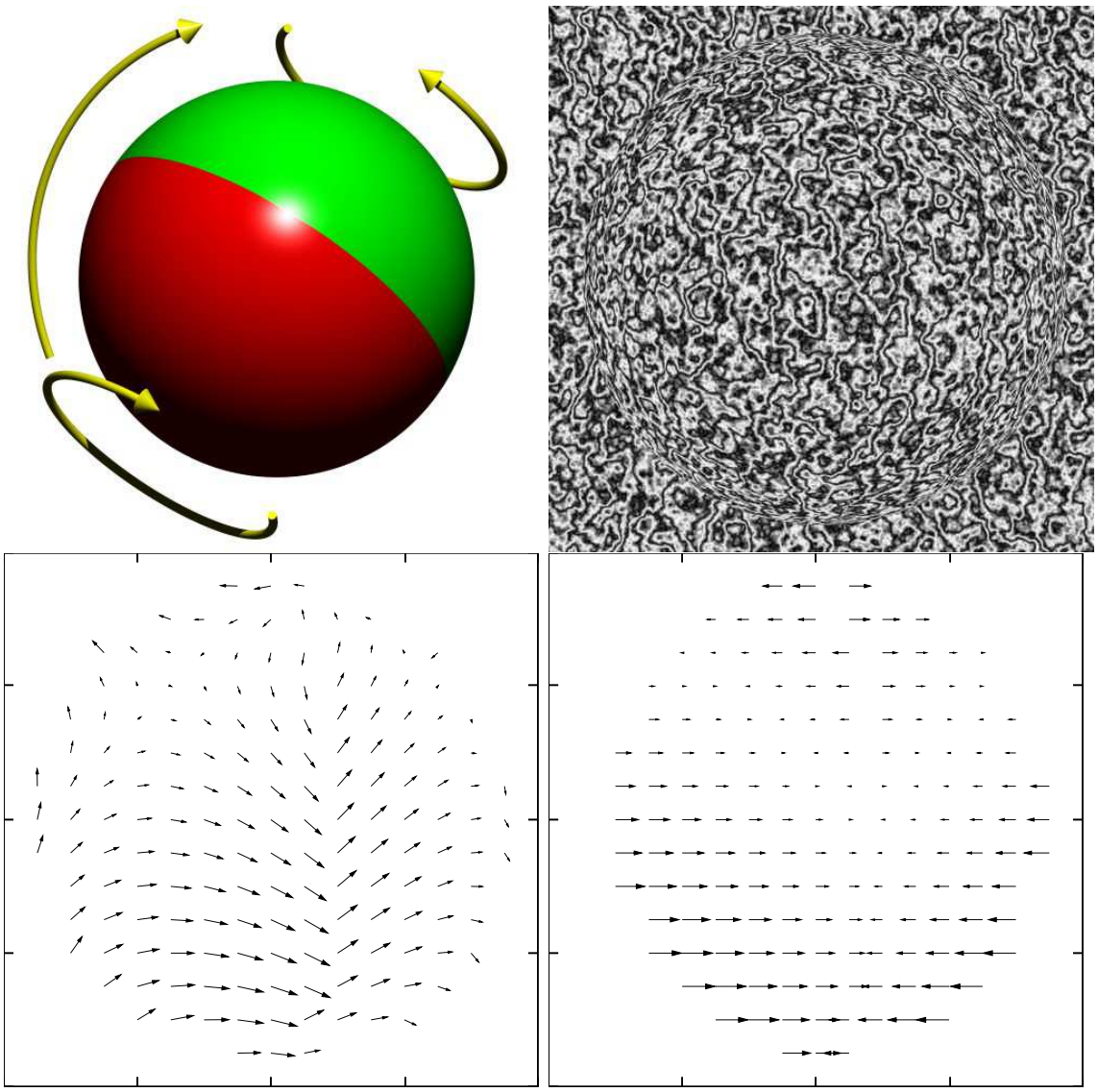

FIGURE 4.2 – En haut à gauche :La scène synthétique présentée est une sphère texturée en rotation, pour laquelle les deux hémisphères tournent indépendamment dans des sens opposés. En haut à droite : image texturée pour diminuer les ambiguités dans les mises en correspondance. La reconstruction 3D est invariante par rotation. L'information du mouvement 3D est mesurable seulement à partir du flot de scène : le flot optique (u, v) (en bas à gauche) et le flot de disparité $d' - d$ (en bas à droite) montrent une discontinuité du flot de scène le long du méridien vertical.

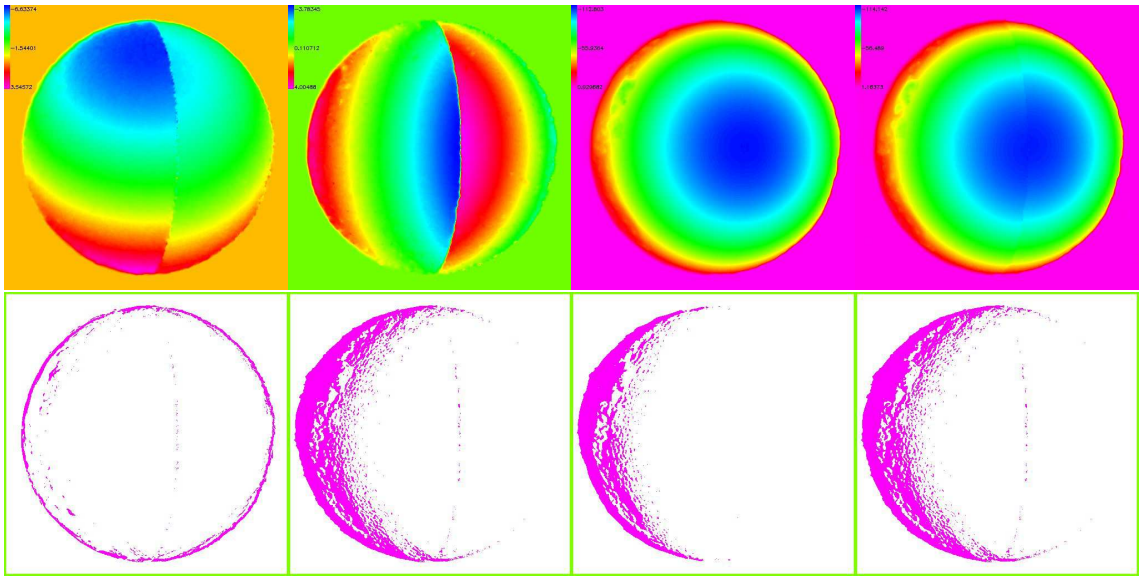

FIGURE 4.3 – En haut, les composantes du flot de scène. De gauche à droite : le flot optique horizontal U, le flot optique vertical V, puis les disparités à t et $t+1$. En bas, les cartes d'occultatio. De gauche à droite : les occultations pour le flot optique (U puis V), et pour la stéréo à t et $t+1$.

Fig. 4.3 expose les valeurs finales de u, v, d et d' pour l'exemple de la sphère, montrant que la discontinuité a été correctement retrouvée par notre algorithme, et montrant les cartes d'occultation générées. Il faut noter que cet exemple a été repris par **??** afin de tester leur algorithme du flot de scène (qui est extrêmement proche du notre).

4.2 Exemples réels

Nous avons utilisé plusieurs types de séquences réelles. Le premier type est constuit à base des images de la base de donnée Middlebury comme indiqué plus haut. Le second type consiste en des séquences réelles prises à l'INRIA, mais sans vérité terrain associée. Cela aurait nécessité en effet l'utilisation d'un grand nombre de capteurs. Néanmoins nous présentons une de ces séquences et une manière d'évaluer qualitativement la précision du flot de scène estimé. Les figures 4.4,4.6,4.8 montre les paires stéréo pour lesquelles nous calculons le flot de scène associé. Les résultats numériques correspondant sont illustrés respectivement en figures 4.5,4.7,4.9. Le déplacement étant ici horizontal, le flot optique vertical est nul et n'est donc pas

FIGURE 4.4 – Exemple Teddy. En haut, les images 2 et 6 de la séquence constituent une première paire stéréo. En bas, les images 4 et 8 constituent la seconde paire.

indiqué.

L'exemple réel filmé à l'INRIA, dénommé "séquence Diana" par la suite, est présenté en figure 4.10. Nous avons réalisé deux séquences vidéo à l'aide d'un système stéréo et en avons extrait des séquences d'images stéréo. L'intervalle de temps entre les deux paires stéréo est d'1.5 secondes. Cette séquence fait intervenir des variations d'illuminations, et de grands mouvements non rigides (rotation de la tête, déformation de la bouche). Les résultats sont exposés en figure 4.11 et 4.12. L'image 4.11 présente les valeurs numériques des composantes du flot de scène, accompagnées des cartes d'occultation. Il faut noter que les pixels en verts sur ces cartes sont désactivés au cours du processus d'optimisation. Les pixels en rose sur les cartes d'occultation sont quand à eux occultés soit pour le flot optique, soit pour la disparité. Comme pour les exemples précédents, la disparité est représentée sous forme de surface $\mathbf{X} \in \Omega \rightarrow d(\mathbf{X})$. On remarque la bonne préservation de discontinuités

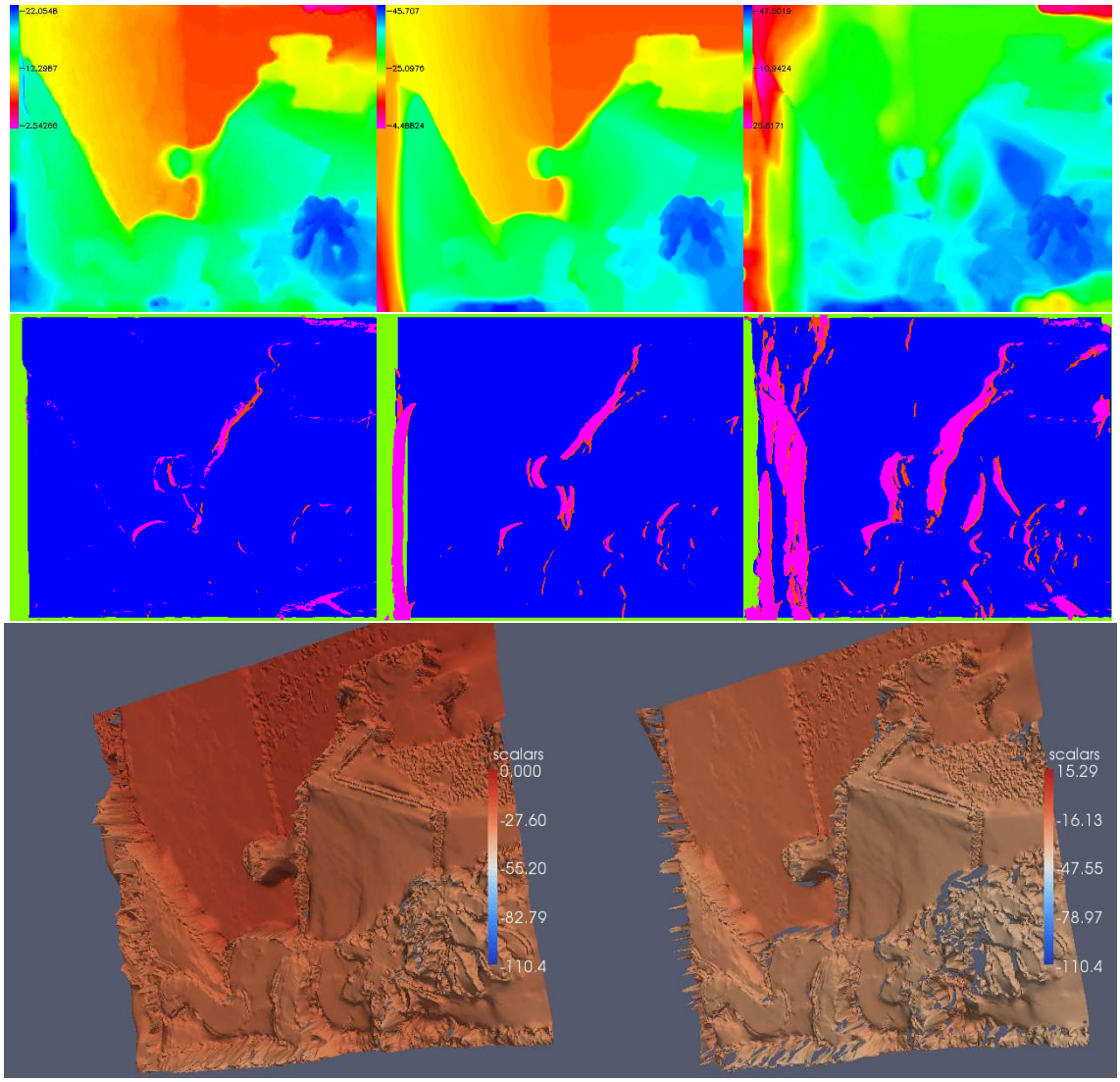

FIGURE 4.5 – Résultats numériques sur la séquence Teddy. En haut, les composantes du flot de scène. De gauche à droite : le flot optique horizontal U, puis les disparités à t et $t+1$. Au milieu, les cartes d'occultation. De gauche à droite : les occultations pour le flot optique gauche, et pour la stéréo à t et $t+1$. En bas : les disparités d et d' représentées comme surfaces. Les trous visibles correspondent aux zones d'occultation indiquées par les cartes du milieu.

au niveau de la tête et de la bouche. L'image 4.12 présente l'image de référence et les trois autres images $I_d(.,t)$, $I_g(.,t+1)$ et $I_d(.,t+1)$ warpées vers le système de coordonnées de $I_g(.,t)$ en utilisant les inverses des composantes du flot de scène estimé, quand celà est possible. Celà peut se formuler de la manière suivante :

- $I_d(.,t) \xrightarrow{d^{-1}} I_g(.,t)$
- $I_g(.,t+1) \xrightarrow{(u,v)^{-1}} I_g(.,t)$

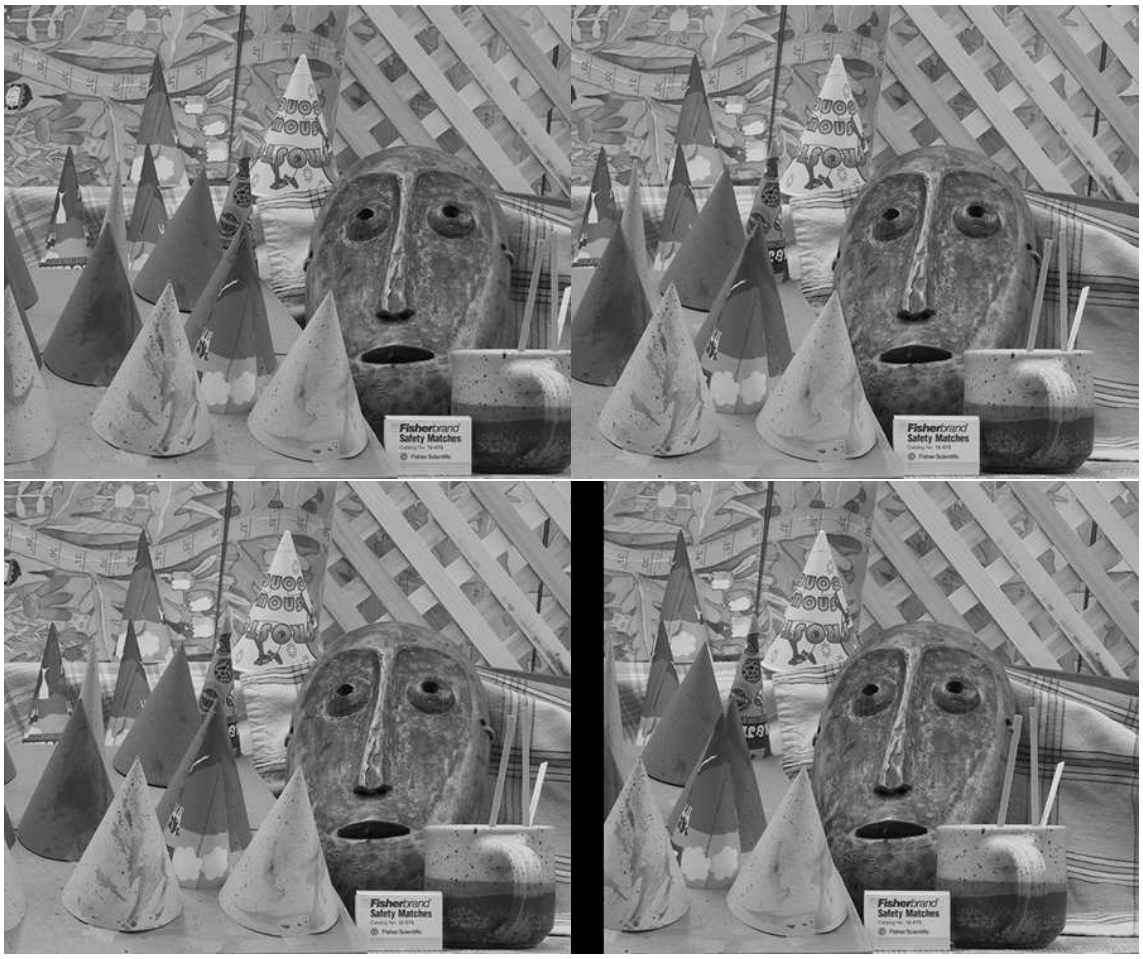

FIGURE 4.6 – Exemple Cones. En haut, les images 2 et 6 de la séquence constituent une première paire stéréo. En bas, les images 4 et 8 constituent la seconde paire.

– $I_d(., t+1) \overset{(u+d',v)^{-1}}{\longrightarrow} I_g(., t)$

Chaque image est ramenée à l'image de référence $I_g(.t)$, quand c'est possible, par la transformée inverse du flot optique et de la disparité. Ainsi, les images produites (dites images backwarpées) sont comparées à l'image de référence originale, ce qui indique les endroits où le flot de scène est correctement estimé. On peut en voir une zone en bord de tête où l'estimation du flot de scène n'est pas précise . Cette zone correspond à un ensemble de pixels où les composantes du flot de scène ne sont pas des transformations injectives. D'autre part, on voit clairement au niveau de la bouche que le flot de scène n'est pas à cet endroit d'une précision subpixélique. Néanmoins, partout ailleurs, l'estimation du flot de scène est visuellement satisfaisante. L'algorithme peut donc gérer dans une certaine mesure des scènes comportant un certain nombre de difficultés : variations d'éclairage, mouvements non rigides ...

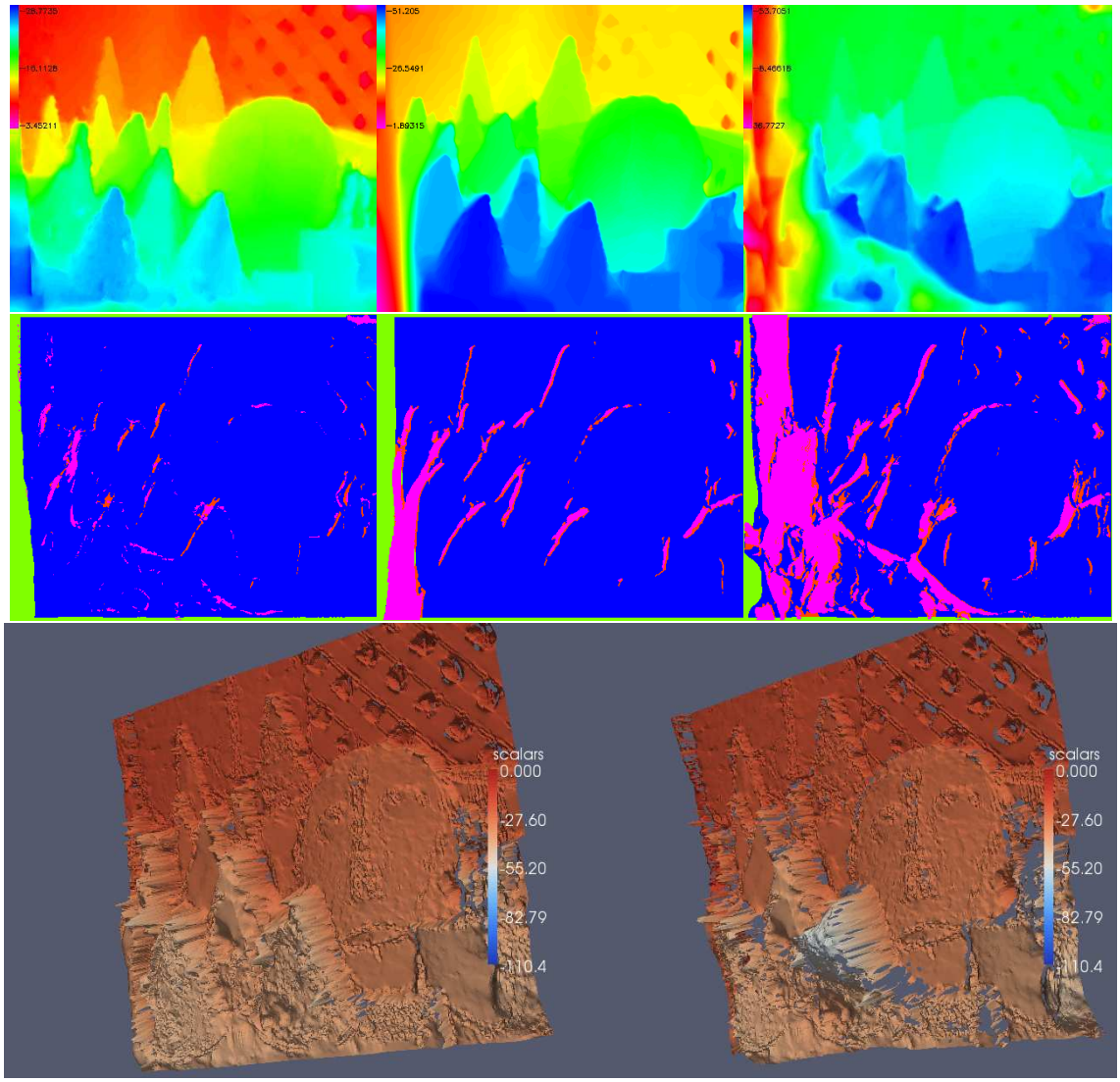

FIGURE 4.7 – Résultats numériques sur la séquence Cones. En haut, les composantes du flot de scène. De gauche à droite : le flot optique horizontal U, puis les disparités à t et $t+1$. Au milieu, les cartes d'occultation. De gauche à droite : les occultations pour U , et pour la stéréo à t et $t+1$. En bas : les disparités d et d' représentées comme surfaces. Les trous visibles correspondent aux zones d'occultation indiquées par les cartes du milieu.

4.3 Evaluation quantitative

L'évaluation quantitative des algorithmes de flot optique consiste traditionnellement dans le calcul de l'erreur angulaire moyenne. On mesure généralement l'écart moyen angulaire entre le champ de déplacement réel (vérité terrain) et le flot optique estimé. Concernant la stéréo, une mesure d'erreur en norme est utilisée (RMS, root mean square error). L'évaluation de la précision de l'algorithme est réalisée ici en estimant l'erreur RMS (root mean square) pour les 4 cartes de valeurs de u, v, $d(.,t)$,

FIGURE 4.8 – Exemple Venus. En haut, les images 2 et 6 de la séquence constituent une première paire stéréo. En bas, les images 4 et 8 constituent la seconde paire.

$d(., t+1)$. Les précisions des cartes de flot optique (u, v) sont évaluées ensemble, et les cartes de disparité sont évaluées séparément : bien qu'exprimées également en pixels, les disparités sont plus difficiles à mesurer du fait de leur plage de valeur et de la présence d'occultations. Les résultats de nos évaluations sont montrés en Fig. 4.13. Fig. 4.14 compare l'erreur angulaire de la composante flot optique du flot de scène avec le flot optique estimé en utilisant l'algorithme de Brox et al. [2004].

5 Améliorations possibles

Un certain nombre d'optimisations peuvent etre envisagées sur le code de résolution numérique, portant à la fois sur la modélisation et sur l'implémentation. En

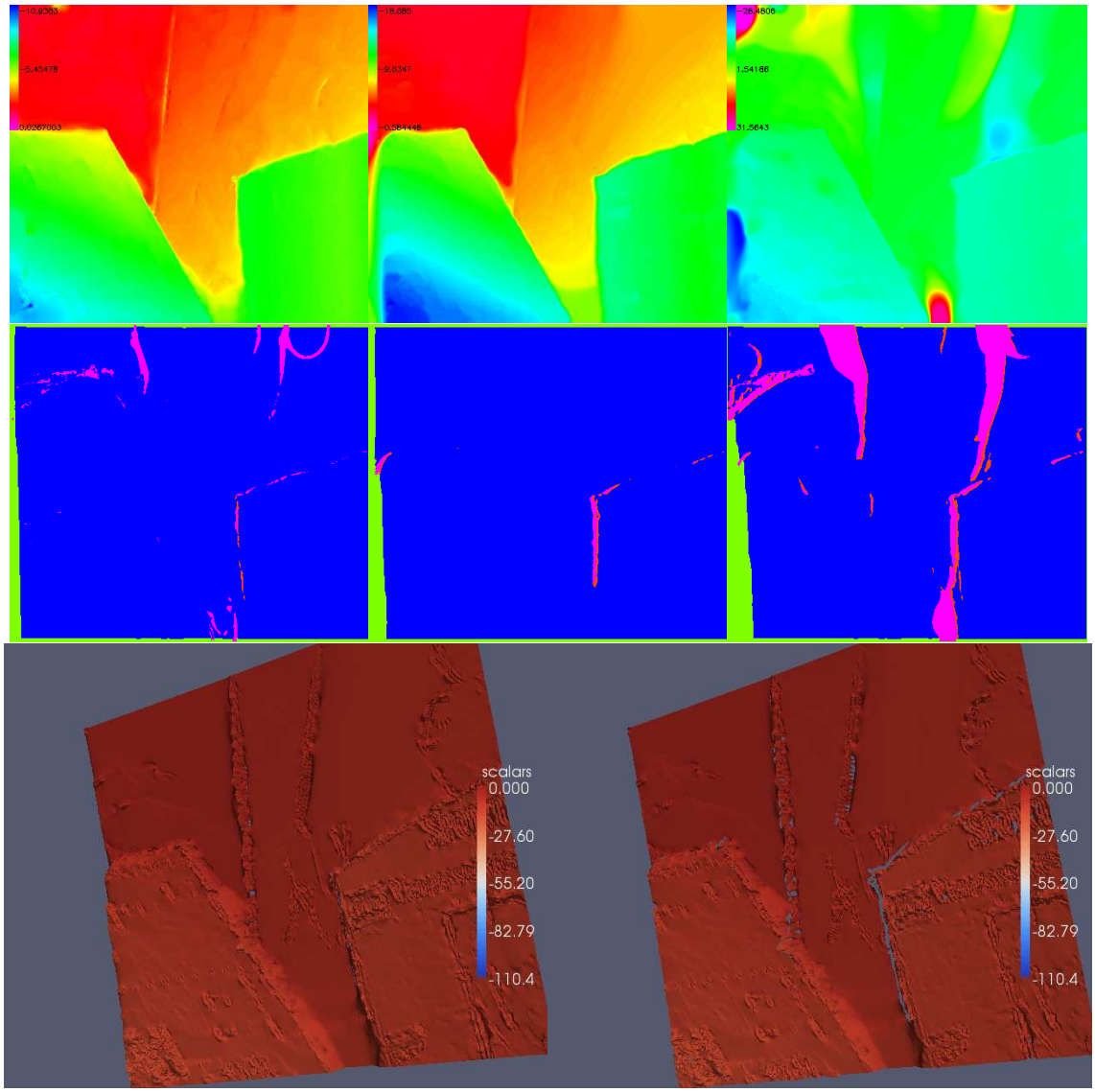

FIGURE 4.9 – Résultats numériques sur la séquence Venus. En haut, les composantes du flot de scène. De gauche à droite : le flot optique horizontal U, puis les disparités à t et $t+1$. Au milieu, les cartes d'occultation. De gauche à droite : les occultations pour U, et pour la stéréo à t et $t+1$. En bas : les disparités d et d' représentées comme surfaces. Les trous visibles correspondent aux zones d'occultation indiquées par les cartes du milieu.

effet, les temps de calcul deviennent relativement importants lorsque la taille des images devient conséquente.

FIGURE 4.10 – Un exemple avec des données réelles (images de 854×854 pixels). L'intervalle de temps entre les paires stéréo du haut et du bas est de 1.5s. On observe des variations d'illumination, de grands mouvements (en translation et rotation), et une claire discontinuité du mouvement dans la région de la bouche.Les plages de valeurs en pixel pour les composantes du flot de scène sur cet exemple sont $u \in [-131, 1]$, $v \in [-49, 33]$, $d(., t), d(., t+1) \in [-122, -39]$.

5.1 Une modélisation cyclopéenne

La modélisation présentée est asymétrique, du fait du choix de $I_g(., t)$ comme image de référence pour l'écriture de l'énergie modélisant le problème du flot de scène stéréoscopique. Afin de conférer un rôle symétrique aux flots optiques gauche et droit nous changeons d'image de référence : nous utilisons les images cyclopénnes aux instants t et $t+1$.a Il n'y a ainsi plus qu'un flot optique (u, v) correspondant a

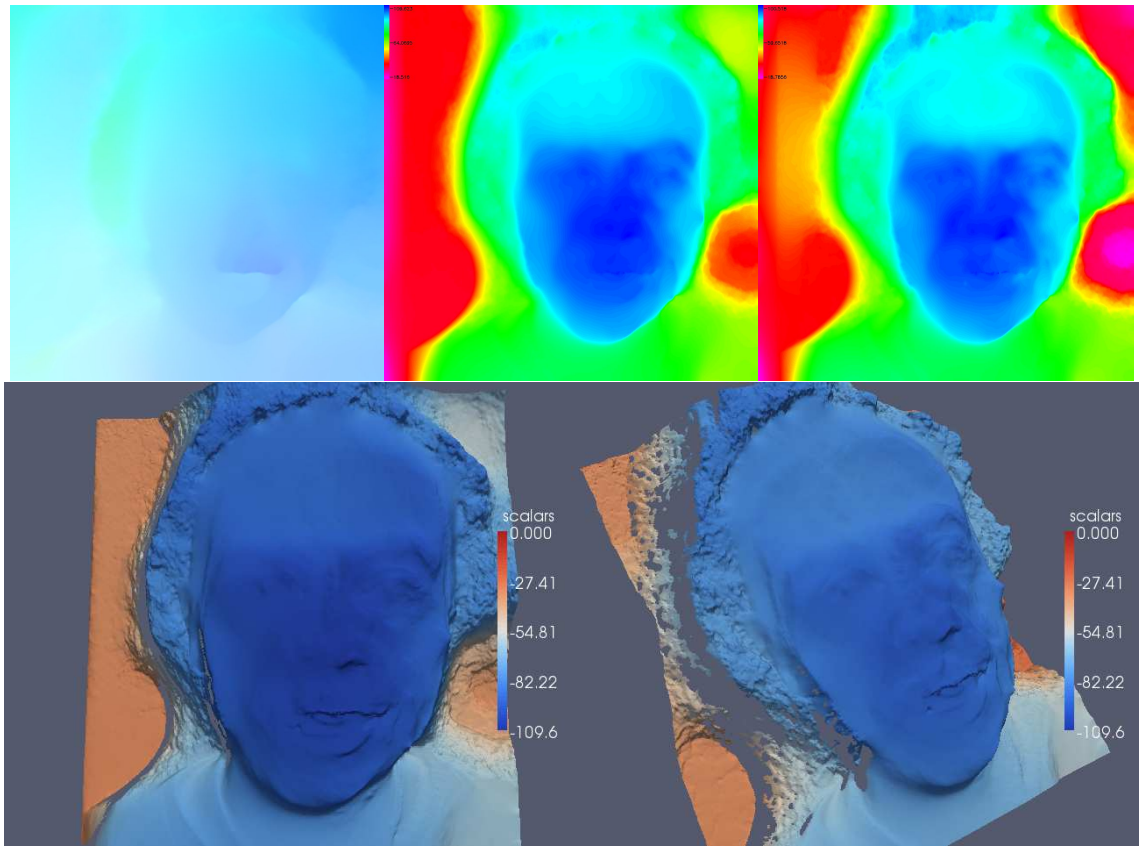

FIGURE 4.11 – Résultats sur des données réelles : en haut à gauche, le flot optique. En haut, au milieu et à droite : les disparités d et d'. En bas, les disparités représentées sous forme de surfaces.

la trajectoire des points dans les images cyclopénnes, et les disparités apparaissent de manière symétrique (et divisées par 2) dans les images gauche et droite. Par conséquent les images ont un rôle symétrique dans la formulation de l'énergie. Les composantes du terme de fidélité aux données J_{data} exprimé dans le système de coordonnées cyclopéen s'expriment désormais de la manière suivante :

- Contrainte de flot optique gauche
- $J_1 = \iint_\Omega \Psi((I_g(x+u-\frac{d(x,y,t+1)}{2}, y+v, t+1) - I_g(x-\frac{d(x,y,t)}{2}, y, t))^2 + \gamma \cdot (\nabla I_g(x+u-\frac{d(x,y,t+1)}{2}, y+v, t+1) - \nabla I_g(x-\frac{d(x,y,t)}{2}, y, t))^2)$
- Contrainte de flot optique droit
- $J_2 = \iint_\Omega \Psi((I_d(x+u+\frac{d(x,y,t+1)}{2}, y+v, t+1) - I_d(x+\frac{d(x,y,t)}{2}, y, t))^2 + \gamma \cdot (\nabla I_d(x+u+\frac{d(x,y,t+1)}{2}, y+v, t+1) - \nabla I_d(x+\frac{d(x,y,t)}{2}, y, t))^2)$
- Contrainte stereoscopique temporelle
- $J_3 = \iint_\Omega \Psi((I_d(x+u+\frac{d(x,y,t+1)}{2}, y+v, t+1) - I_g(x+u-\frac{d(x,y,t+1)}{2}, y+v, t+1))^2 + \gamma \cdot (\nabla I_d(x+u+\frac{d(x,y,t+1)}{2}, y+v, t+1) - \nabla I_g(x+u-\frac{d(x,y,t+1)}{2}, y+v, t+1))^2)$
- Contrainte stereoscopique initiale
- $J_4 = \iint_\Omega \Psi((I_d(x+\frac{d(x,y,t)}{2}, y, t) - I_g(x-\frac{d(x,y,t)}{2}, y, t))^2 + \gamma \cdot (\nabla I_d(x+\frac{d(x,y,t)}{2}, y, t) -$

FIGURE 4.12 – Résultats sur la séquence Diana : en haut à gauche, l'image de référence, dans le système de coordonnées de laquelle s'expriment les composantes du flot de scène. L'image de droite au temps 0 (en haut a droite) et la paire stéréo à 1.5s ont été warpées vers l'image de gauche à l'instant 0, montrant les endroits où le flot de scène a été correctement estimé. On remarque d'une part une large discontinuité sur le côté de la tête, correspondant à une zone où les composantes du flot de scène ne sont pas des applications injectives, et d'autre part que la précision du flot optique n'est pas subpixélique au niveau de la bouche.

$\nabla I_g(x - \frac{d(x,y,t)}{2}, y, t))^2)$

Dataset	(u, v)	d	d'
Venus	0.31	0.97	1.48
Teddy	1.25	2.27	6.93
Cones	1.11	2.11	5.24
Sphere	0.69	3.73	3.81

FIGURE 4.13 – Erreur RMS en pixels sur les 4 composantes du flot de scène estimées par notre algorithme sur les différents jeux de données de validation.

Dataset	μ_{of}	σ_{of}	μ_{sf}	σ_{sf}
Venus	1.06	1.17	0.98	0.91
Teddy	0.43	0.49	0.51	0.66
Cones	0.66	1.21	0.69	0.77
Sphere	1.50	5.65	1.75	6.07

FIGURE 4.14 – Espérance μ_{sf} et écart type σ_{sf} de l'erreur angulaire en degrés de la composante flot optique (u, v) du flot de scène, comparées a l'erreur angulaire (μ_{of}, σ_{of}) du flot optique estimé séparément

5.2 Une résolution numérique optimisée

L'inconvénient majeur de l'algorithme mis en oeuvre est son cout en temps de calcul. Plusieurs raisons expliquent ce défaut :

- Le résultat recherché est dense, les EDP sont donc résolues en chaque pixel de l'image de référence $I_g(., t)$. Soit n_0 le nombre de pixels d'une image à sa résolution originelle. Les images sont sous échantillonnées d'un facteur η pour construire les pyramides gaussiennes. Le nombre de pixels de l'image à la résolution i est donc $n_i = \eta^{2i} \cdot n_0$. Soit K_i le cout de la résolution des EDP pour un pixel donné à la résolution i. Le cout de l'algorithme est de l'ordre de $\Sigma_{i=1}^{i=niveau-max} K_i \cdot n_i$ soit $n_0 \cdot \Sigma_{i=1}^{i=niveau-max} K_i \cdot \eta^{2i}$. $\Sigma_{i=1}^{i=niveau-max} K_i \cdot \eta^{2i}$ étant une constante indépendante du nombre de pixels dans les images, l'algorithme a donc un cout en $O(n_0)$. Les images issues d'appareils photos numériques comportant actuellement plusieurs millions de pixels, il est donc clair que l'algorithme est très onéreux en temps de calcul pour des séquences réelles de taille optimale.
- L'emploi de la méthode SOR pour résoudre les systèmes linéaires issus de la discrétisation des équations d'Euler Lagrange possède aussi un inconvénient majeur en terme de rapidité. Cette méthode a pour caractéristique d'éliminer en quelques itérations l'essentiel de l'erreur entre la solution approchée et la solution exacte d'un système linéaire. En terme d'espace de Fourier, elle élimine rapidement les hautes fréquences de l'erreur. Néanmoins, lorsque ces hautes fréquences sont éliminées, SOR est à la peine avec ce qui reste de l'erreur et de nombreuses itérations sont nécessaires pour converger assez proche de la solution. Etant donné que l'on doit utiliser SOR un grand nombre de fois

au cours des itérations de l'algorithme, l'impact sur la rapidité du code est considérable. D'autre part, les systèmes linéaires résolus sont, dans le cas du flot de scènes, grands et très fortement creux (moins d'un pour cent d'éléments non nuls dans la matrice du système pour les grandes résolutions), ce qui amène le problème à etre numériquement mal conditionné.

Des travaux récents, comme ceux de Bruhn et al. [2005a] ont permis d'accélérer considérablement le calcul du flot optique. Ceux-ci sont basés sur des approches multigrilles bidirectionnelles (Briggs et al. [2000]) qui surmontent les problèmes de rapidité de SOR ou de Gauss-Seidel en créant une hiérarchie de systèmes d'équations possédant de très bonnes propriétés de diminution des erreurs. Les basses fréquences des erreurs sur les hautes résolutions induisent des hautes fréquences sur les erreurs estimées aux résolutions plus grossières, ou elles peuvent etre rapidement éliminées par SOR ou Gauss-Seidel. Cette stratégie permet ainsi d'estimer des résultats précis de manière bien plus rapide qu'avec des solvers itératifs non hiérarchiques. Bruhn et al. [2005a] décrivent en détail l'écriture d'une telle hiérarchie de systèmes pour un 2-cycle (dont un aller retour entre deux niveaux de résolution consécutifs) et présentent des résultats pour des V-cycles (3 niveaux de résolution) et des W-cycles. Les résultats présentés permettent de constater l'efficacité de cette approche du point de vue de la précision des résultats et de la rapidité de l'obtention de ceux ci. Une telle approche est tout à fait envisageable dans le cas du flot de scène, étant donné qu'un grand système linéaire est résolu à chaque sous-itération de l'algorithme. Et compte tenu du grand nombre de résolutions de système à effectuer pour estimer le flot de scène entre deux paires stéréo consécutives, on peut raisonnablement envisager un gain en temps et en précision sensible.

5.3 Parallélisme envisageable

Les problèmes rencontrés aujourd'hui dans différents domaines (physique notamment) mettent souvent en jeu de grands domaines de calcul, et nécessitent la plupart du temps pour leur résolution numérique un temps de calcul important sur du matériel puissant. Néanmoins l'avènement des processeurs multicores, notamment dans les machines grand public, ainsi que du développement d'algorithmes sur GPU, permettent dans un certain nombre de cas de concevoir une solution parallèle. De manière générale, les méthodes par décomposition de domaines sont utilisées afin de découper de grands domaines de calcul en sous domaines, chacun de ces derniers étant affecté à un processeur (ou à un core). Les solutions étant ensuite raccordée avec des conditions au bord spécifiques. Les images étant des domaines homéomorphes à des formes géométriques basiques, en l'occurence des rectangles, leur découpage en sous domaine est trivial. Les méthodes par décomposition de domaine sont donc naturellement adaptées aux problèmes de résolution d'EDP en traitement d'images et en vision. Le travail de Kohlberger et al. [2003] introduit une méthode de décomposition de domaine pour le flot optique, dans le cadre variationnel. Il semble donc, au vu de la qualité des résultats proposés dans l'article, possible de l'adapter au problème de l'estimation du flot de scène tel que nous l'avons étudié.

La difficulté principale étant de trouver une condition au bord des sous domaines assez convenables pour raccorder de manière continue les composantes du flot de scène évaluées séparément sur chaque sous-domaine. Il est clair que pour des soucis de raccordement, l'équation de raccordement doit etre résolue pour chaque itération principale de l'algorithme (et non pas attendre d'avoir estimé une solution finale sur chaque sous-domaine avant de les raccorder).

D'autre part, nous assistons à l'essort du calcul scientifique sur GPU depuis quelques années. Un bon exemple de diverses applications se trouve à l'adresse `http://www.nvidia.com/object/cuda_home.html#`. Nous avons cité précédement des méthodes de flot optique et des méthodes multigrilles implémentées sur GPU, notamment les travaux de Pauwels and Van Hulle [2008] et de Grossauer and Thoman [2008]. L'implémentation sur GPU de notre algorithme est possible en adaptant une méthode de décomposition de domaine et en changeant le solveur itératif que nous utilisons pour résoudre les systèmes linéaires correspondant aux EDP. En effet, chaque itération SOR (algorithme 3) est telle qu'un terme du vecteur d'inconnues à évaluer est mis à jour en utilisant les valeurs de ses prédecesseurs (dans l'ordre de rangement des éléments du vecteur) estimés à la même itération. Cette méthode n'est donc pas parallélisable. On peut penser par exemple à remplacer SOR par le solveur à base de gradient conjugué implémenté sur GPU par Buatois et al. [2007]. On peut également noter que la méthode GMRES, beaucoup utilisée notamment dans les cas où les matrices des systèmes linéaires sont creuses et non définies positives, est implémentable sur GPU en utilisant les outils développés par Buatois *et al.*.

6 Perspectives : une base de données de validation pour le flot de scène ?

Il existe depuis un certain temps déjà des bases de données de test publiques pour évaluer la robustesse des algorithmes de stéréo. La base de données de référence est actuellement celle de Middlebury (`http://vision.middlebury.edu/stereo/`), évaluée à partir de la classification de Scharstein et Szeliski, et propose un classement de divers algorithmes de la littérature. Le travail de construction de séquences d'images avec leur vérité terrain de Baker et al. [2007] a permis de créer une base de données de test pour les algorithmes d'évaluation du flot optique, accessible sur le site de la base de données Middlebury. La combinaison des deux amène naturellement à se poser la question de la possibilité de créer une base de données similaire de validation pour le flot de scène. Le flot de scène n'étant pas un domaine d'étude en Vision aussi mature que la stéréo ou le flot optique, l'absence d'une telle base de données de test se fait ressentir cruellement. Comment dans ces conditions obtenir une évaluation valable des qualités d'un algorithme donné ? Pour obtenir une vérité terrain, nécessaire à l'obtention de statistiques (l'erreur angulaire et l'erreur en norme, comme dans le cas du flot optique), dans le cas de l'estimation d'un flot de scène dense, il est nécessaire de placer un très grand nombre de capteurs sur

les objets de la scène étudiée. Bien sur, celà n'est pas possible ne pratique, il faut donc envisager de réaliser des mesures d'erreur sur un sous ensemble du flot de scène estimé, ou bien se contenter de résultats purement qualitatifs, comme ceux présentés précédemment avec la séquence Diana. Une autre manière de procéder, spécifique au flot de scène stéréocopique, consiste à combiner simplement les bases de données de test stéréo et flot optique afin d'évaluer les composantes 2D du flot de scène séparément. Cette approche a aussi l'avantage de comparer les algorithmes d'estimation du flot de scène stéréoscopique avec les méthodes de stéréo et de flot de scène, ce qui peut permettre d'évaluer l'impact de la quantité d'information apportée par l'estimation du flot de scène par rapport à l'estimation de la stéréo ou du flot optique seul. Il est donc clairement envisageable de constitiuer une base de données tests dans le cas du flot de scène stéréoscopique.

7 Conclusion

Nous avons présenté un algorithme original de résolution numérique des équations présentées au chapitre précédent. Celui-ci utilise des pyramides gaussiennes afin de résoudre des systèmes d'équations plus simples sur des domaines de résolutions progressives. Si l'on part d'un niveau de résolution suffisamment grossier, les expériences montrent que l'on peut espérer converger initialement vers un minimum acceptable. Le raffinement des estimations numériques au fur et à mesure de la montée en résolution dans les pyramides gaussiennes nous permet au final d'aboutir à de bonnes estimations numériques.

Pour valider notre approche nous avons utilisé des exemples à la fois synthétiques (la boule, plus tard reprise dans certains travaux sur le flot de scène postérieurs au nôtre) et réels (exemples issus de Middlebury, séquence Diana), nous permettant d'évaluer la méthode dans différentes conditions (illumination, mouvements rigides ou non ...). Nous avons dû créer nos propres séquences de test, mais comme nous l'avons vu, nous pouvons envisager la création de bases de données de test spécifiques au flot de scène. Le dernier chapitre présente l'application de notre algorithme à l'estimation de la dynamique d'une surface continue par morceaux observée par un système stéréoscopique.

Chapitre 5

Application à la géophysique

Bien que n'étant pas encore un domaine d'étude totalement mature en vision par ordinateur, le flot de scène possède plusieurs applications directes évidentes et généralisant les applications du flot optique traditionnel :

- Le "tracking 3D" des objets d'une scène, généralisant le tracking utilisé dans les systèmes de vidéo surveillance
- Applications en robotique
- Intégration possible dans une méthode d'assimilation de données : analyse 3D du mouvement de fluides

Néanmoins, dans cette thèse, nous nous attachons à développer une application originale du flot de scène. Le travail présenté dans ce chapitre a été réalisé en étroite collaboration avec le laboratoire Geosciences Azur, situé à Sophia Antipolis. L'une des problématiques étudiées au sein de ce laboratoire est l'étude des mouvements de terrain dus à la gravité : ceci comprend les éboulements et les glissements de terrain. Les modèles numériques développés par les géophysiciens se limitent à une approche 2D et en petites déformations (cas de l'initation de ruptures). En effet, chaque massif montagneux possède une structure interne et une composition propre, ce qui oblige pratiquement à refaire une modélisation lorsque l'on change de massif d'études. Pour palier à ces difficultés, les géophysiciens de Geosciences Azur ont développé une nouvelle méthode, la modélisation physique 3D. Nous exposerons les grands principes de celle-ci au cours de ce chapitre. Lors de la réalisation de ces expériences, nous disposons un système stéréo calibré afin de suivre l'évolution de la déformation à la surface des modèles au cours du changement gravitaire. Partant des séquences stéréo fournies, nous pouvons grâce à notre algorithme, évaluer le flot de scène. Celui-ci doit alors permettre de mesurer les champs de déplacements et le tenseur de déformation associé, ce qui fournit de précieuses informations sur les processus en jeu dans les mouvements gravitaires. Nous présentons ici une série de résultats numériques sur des séquences stéréo obtenues à Geosciences Azur.

1 L'étude des mouvements gravitaires : GEOLSTEREO

1.1 Les mouvements de terrain gravitaires

De nombreuses catastrophes naturelles liées aux mouvements de terrain sont dues aux séismes (chevauchement de plaques tectoniques). Néanmoins, les mouvements gravitaires, ou déstabilisations de versants, moins connus, sont également à l'origine de mouvements de terrain destructeurs : glissements de terrain, éboulis, effondrements de falaise, coulées de boue (1870, volcan Huascaran, Pérou, 18000 morts). Dans notre projet, nous étudions la modélisation de ces mouvements gravitaires sur un seul massif.

Un mouvement gravitaire peut mettre en jeu plusieurs phénomènes causés par la gravité. La diversité de ces phénomènes rend difficile une classification complète de ces mouvements. Néanmoins, les géophysiciens regroupent généralement ces mouvements en deux sous-groupes, les mouvements de comportement fluide visqueux et ceux affectant les massifs rocheux :

- Les mouvements gravitaires de comportement fluide visqueux. Ils affectent les surfaces meubles (argile, sols) pouvant être saturés en eau suite à d'importantes précipitations. Ils sont différenciés par la morphologie des dépôts et leur comportement mécanique. Le comportement de l'écoulement est alors conditionné par la rhéologie (l'étude des propriétés des matériaux), l'hydrologie et la structure du versant déstabilisé. Un exemple en est montré en figure 5.1 avec le glissement-coulée de Super-Sauze.

FIGURE 5.1 – Glissement-coulée situé à Super-Sauze (image Geosciences Azur)

– Les mouvements gravitaires affectant les massifs rocheux. Ils sont d'ampleur très variée. Parmi ceux-ci, on distingue les chutes de blocs (quelques milliers de mètres cubes de roche), les grands mouvements de versant, comme sur le massif de La Clapière présenté précédemment, représentant plusieurs kilomètres cubes, et enfin les effondrements d'édifices volcaniques.

Le cadre de notre étude concerne le second type de mouvements gravitaires. Parmi ces derniers, on distingue 3 sous-types dont voici une description.

Les mouvements de blocs. Ils affectent généralement les falaises. Ils semblent être déclenchés par l'existence de fracturations au sein du massif ainsi que par des paramètres extérieurs tels que la circulation d'eau et les variations de température. En général, on cherche à trouver les relations entre les déformations et les conditions thermo-hydromécaniques, sur des sites-laboratoires (falaise de Valabre dans les Alpes Maritimes).

Les effondrements d'édifices volcaniques. Mouvements de très grande ampleur, ils semblent affecter tous les grands volcans au cours de leur évolution. Des preuves d'occurence de ces effondrements ont été mises en lumière sur diverses chaînes de volcans à travers le monde (Canaries, Andes, Massif Central). Les causes supposées de ces effondrements sont liées à l'activité volcanique (intrusions de magma), ou à des jeux de failles. Un exemple de ce type de mouvements est montré par l'exemple du volcan de la Soufrière en figure 5.2.

Les effondrements de versants. Ils affectent des massifs montagneux de grande taille, et un bon exemple en est celui du massif de La Clapière représenté en figure 5.3. Ces glissements affectent, en zone superficielle, des massifs cristallins fracturés et altérés. Cette altération est due aux circulations d'eau à l'intérieur du massif. Circulation d'eau, répartition des contraintes liée à la structure interne du massif, ainsi que discontinuités structurales apparaissent comme fondamentales quand à la déstabilisation des versants.

L'étude des mouvements gravitaires affectant les massifs rocheux est donc importante dans la mesure où elle pourrait à terme permettre d'effectuer des calculs de risques dans des zones habitées et ainsi prévenir des catastrophes.

1.2 Classification des approches numériques disponibles

Il existe différents types d'analyse numérique utilisés suivant le but recherché. Les analyses de stabilité et les analyses cinématiques déterminent la stabilité d'un versant en connaissant les surfaces de rupture. La modélisation paramétrique détermine les paramètres mécaniques conduisant à la déstabilisation d'un massif. Résumons brièvement ces approches.

Les analyses de stabilité

Utilisées en particulier dans le cadre d'études sur des formations meubles (sols, moraines). Elles consistent à calculer le rapport de la force de résistance sur la

FIGURE 5.2 – Dépôt d'avalanche de débris du 26 décembre 1997 à la Soufrière. Image issue des expériences menées par l'Institut de Physique du Globe de Paris, disponible à l'adresse `http://www.ipgp.jussieu.fr/pages/0218021201.php?version=print`

FIGURE 5.3 – Massif de la Clapière, St Etienne de Tinée, Alpes Maritimes (image Geosciences Azur)

force motrice (le poids). Le versant est stable si ce rapport excède 1. On utilise les analyses par équilibres limites, ou méthodes de calcul à la rupture. Celles ci étudient la stabilité des mouvements de translation ou de rotation sur une zone donnée. Elles consistent en un bilan des forces et des moments sur une zone discrétisée en tranches. Les contraintes prises en compte sont alors le poids et le cisaillement le long de la surface de rupture (là où débute le mouvement). Les analyses de stabilité sont généralement utilisées pour l'étude des glissements en milieu continu (sols) mais elles sont moins adaptées aux massifs rocheux dont le comportement mécanique est beaucoup plus complexe. En effet, dans un massif rocheux, l'évolution d'une instabilité dépend non seulement de la résistance à la base mais aussi de la capacité de la masse rocheuse à se déformer et à se rompre. Il est donc nécessaire de prendre en compte les caractéristiques internes du massif étudié, ce que ne font pas les analyses de stabilité.

Les analyses cinématiques

Elles sont utilisées pour l'analyse de la stabilité des falaises. Lorsque sur le terrain, les géophysiciens observent l'orientation des fractures, ils peuvent en déduire la géométrie des différents blocs. Connaissant celà, ils en déduisent les plans de glissement potentiels. Néanmoins les travaux de Cappa [2001-2002] ont inclus dans ce type d'analyse le comportement mécanique des blocs, en prenant en compte le frottement entre blocs et leur élasticité. Des codes de calcul récents incluent une analyse de stabilité couplée à cette analyse mécano-cinématique.

Les modèles mécaniques

Ces modèles traitent un versant comme un milieu continu, discontinu ou mixte. Dans le cas continu, on utilise des codes aux éléments finis. Le versant est caractérisé par des paramètres d'élasticité et par un critère de rupture. L'inconvénient de ce type d'approche continue est que pour le moment, ces valeurs sont attribuées au massif de manière homogène, éludant ainsi les hétérogénéités structurales constatées sur le terrain. Le comportement modélisé correspondrait plutôt à celui d'un tas de sable.

Dans le cas de la modélisation en milieu discontinu on utilise des codes à éléments distincts. Le milieu est considéré comme un assemblage de blocs séparés par des interfaces. Sur celles ci sont définies des valeurs de raideur normales et tangentielles. Ces valeurs caractérisent la mobilité relative de chaque bloc par rapport aux autres. Des coefficients d'élasticité sont également affectés aux blocs pour prendre en compte leur capacité à se déformer. Ces codes sont utilisés pour analyser les changements de comportement d'un massif suite à l'action de la circulation d'eau aux interfaces. L'inconvénient de ces codes est de ne pouvoir prendre en compte une trop grande complexité structurale. En effet, le nombre d'orientations possibles de fractures introduites dans le modèle est limité, ainsi que les propriétés des interfaces.

Les approches mixtes couplent les deux approches précédentes. Le milieu est divisé en blocs comme dans la méthode des éléments distincts, cependant ces blocs sont maillés comme dans un code aux éléments finis traditionnels. Dans les zones où les critères de rupture sont satisfaits, des fractures sont initiées et le maillage s'adapte au cours du temps pour autoriser la propagation des failles. Néanmoins, le problème majeur de ce type d'approche réside dans la difficulté de représenter des structures 3D hétérogènes complexes.

Ce récapitulatif concernant les méthodes numériques employées nous permet de mieux cerner les difficultés rencontrées par la simulation numérique dans l'étude de la déstabilisation de versants. Un bon exemple de simulation numérique appliqué au massif de La Clapière est décrit par Cappa [2001-2002].

1.3 La modélisation physique 3D

Principe

Pour pallier à la difficulté de représenter une structure 3D aussi hétérogène et complexe qu'un massif montagneux, les chercheurs du laboratoire Géosciences Azur travaillent sur des maquettes de massif à l'échelle réduite. La forme de celle ci reproduit de manière simplifiée la forme du massif (maquette de forme prismatique). Le problème essentiel de la modélisation physique consiste à trouver les matériaux constitutifs de la maquette qui puissent permettre à celle ci de reproduire le comportement de la montagne étudiée.

Pour celà, les matériaux sont élaborés en laboratoire. Il s'agit de matériaux composites à base d'hydrocarbures solides et liquides. Les propriétés de ces matériaux peuvent dépendre fortement des conditions de température et pression. Celà nécessite de grandes précautions au niveau expérimental. En général, les modèles sont

conservés dans une chambre à la température étroitement contrôlée par un système de ventilation.

Les propriétés des matériaux utilisés sont ajustées de manière à ce que, par exemple, un glissement de la surface de la maquette s'effectue sur une distance compatible avec les observations sur le site réel. On veut, à partir de ce modèle réduit, simuler les mouvements gravitaires affectant la surface du modèle et simulant les mouvements de terrain. Pour celà, on impose à la maquette des accélérations cycliques à l'issue duquel on observe l'effondrement d'une partie de la surface (dans le cas présenté à Géosciences Azur, une cinquantaine d'accélérations étaient nécessaires pour l'observer). Les accélérations successives provoquent l'apparition de faiblesses structurelles dans le modèle comme les conditions naturelles précédemment citées le font avec les massifs, mais à l'échelle de milliers d'années (voire de millions d'années). Le dispositif expérimental est entièrement décrit par Bachmann [2007].

Expérimentalement, les géophysiciens créent dans la maquette des hétérogénéités simulant des zones de faiblesse dans le massif, avant l'expérience, afin de déterminer quel type de fracture tend à accroître les risques de glissement en surface, ou bien tend à favoriser la propagation du mouvement. Une fois ces différents types de fractures identifiés, les formes de glissement obtenus sur la maquette dans différentes configurations de défauts sont comparées aux glissements observés sur le terrain. On en déduit le degré de réalisme de l'expérience et l'influence des paramètres de départ. L'évolution d'une maquette homogène au cours de l'expérience est représentée sur la figure 5.4.

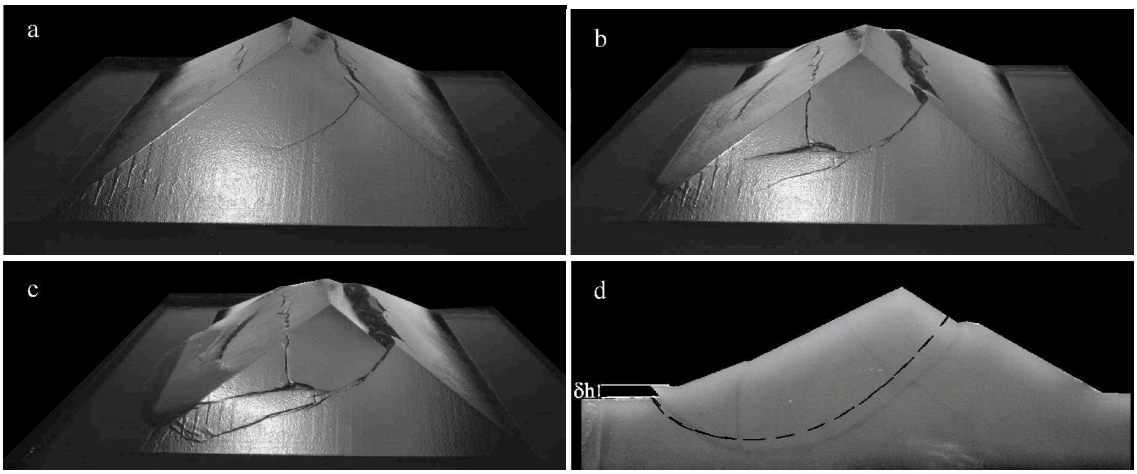

FIGURE 5.4 – En haut à gauche : début de rupture après 100 cycles d'accélérations. En haut à droite : après 105 cycles d'accélérations. En bas à gauche : après 110 cycles. En bas à droite : la ligne pointillées représente la surface de rupture obtenue.

2 L'expérience

L'expérience mise en oeuvre comporte une partie dédiée à la reconstitution de glissements de terrain à l'échelle réduite, et une partie destinée à la reconstruction du flot de scène associé, constituée d'un système stéréo à base d'appareils photos numériques identiques. L'expérience débute par la fabrication de la maquette. Le matériau est d'abord découpé manuellement, jusqu'à atteindre la forme voulue, comme l'illustre la figure 5.5.

FIGURE 5.5 – À gauche : découpe du matériau. À droite : la forme finale de la maquette.Les dimensions de celle-ci sont d'environ 30 cms sur 20 cms.

Une fois la maquette réalisée, nous la saupoudrons avec du sable peint à l'aide d'encre de Chine, afin de texturer la surface et de permettre à l'algorithme d'estimation du flot de scène de disposer de suffisamment d'informations. La maquette texturée se trouve en image 5.6.

2.1 Dispositif géophysique

Afin de soumettre la maquette à des contraintes gravitaires, on lui fait subir un cycle d'accélérations successives en lui imposant une chute libre et un amortissement progressif en fin de chute. La maquette est ainsi disposée sur un plateau que l'on élève progressivement grâce à des rails. Un accéléromètre fixé sur le plateau mesure l'acceleration du modele pendant la phase de chargement. Compte tenu de la course de l'amortisseur et de la hauteur de chute , on impose une acceleration de 500 m/s^2 (figure 5.8). Dans notre cas, il est nécessaire d'imposer une quarantaine d'accélérations afin d'observer les premières failles sur la maquette. Il faut noter que l'observateur est protégé par une grille entourant le dispositif. La figure 5.7 représente le dispositif de manière légendée, et la figure 5.8 donne un exemple de données d'accélération.

FIGURE 5.6 – La maquette texturée avant l'expérience.

2.2 Dispositif stéréo

Deux appareils photo numériques sont placés au dessus de la maquette afin de suivre l'évolution de celle-ci au cours du temps. Le pas de temps minimal entre deux prises de vue est défini par un cycle de chargement. Nous disposons donc d'un flot temporel de paires d'images prises à des intervalles arbitraires (et donc aussi petits que l'on veut). Le dispositif est montré tel quel en figure A.3.

2.3 Protocole de prises de vue

Du fait de la sensibilité du paramétrage des appareils photos, il a fallu établir avant toute expérimentation un protocole de prises de vue tout à a fait strict. En effet, nous avions constaté certains défauts sur les images prises lors d'expériences précédentes. D'une part, nous n'avions pas exactement les memes conditions d'éclairage pour une paire stéréo donnée, ce qui est très pénalisant lorsqu'il s'agit de mettre en correspondance des points. D'autre part, nous avions observé des zones floues sur les zones des images correspondant aux Z minimum et maximum de la scène : il s'agit typiquement d'un problème de profondeur de champ. Nous avons par consé-

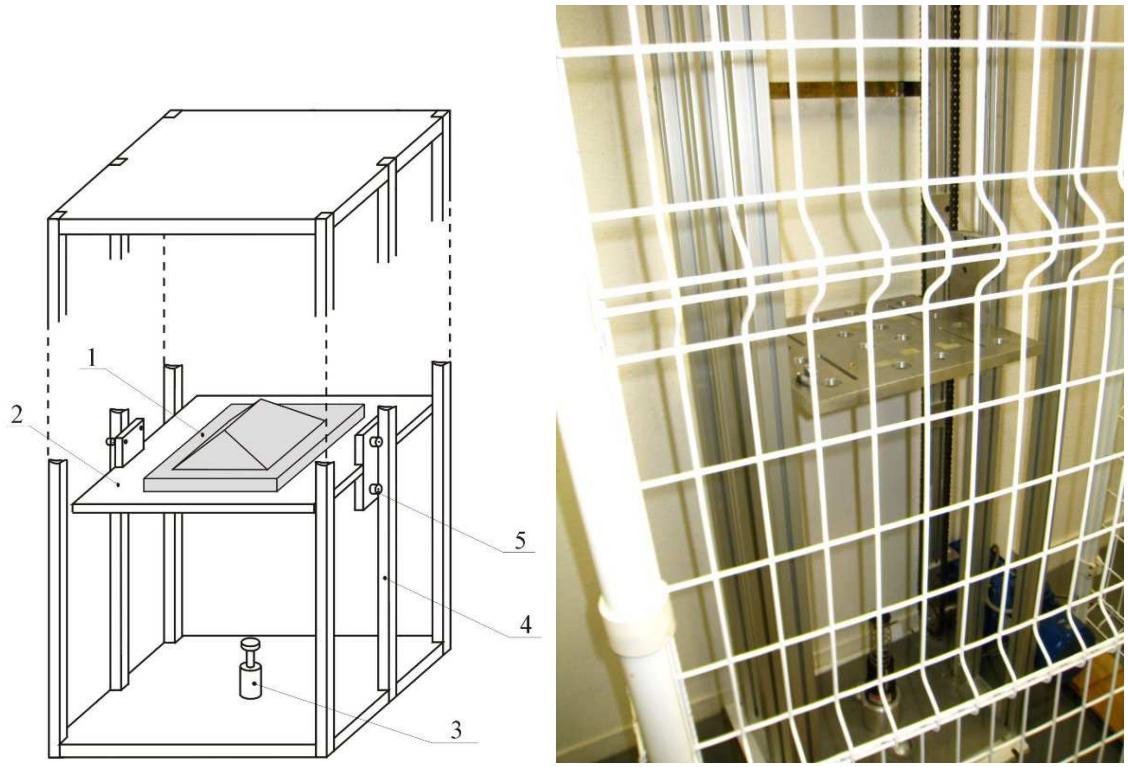

FIGURE 5.7 – À gauche, le montage expérimental. Le modèle (1) est placé sur un plateau (2) qui peut être translaté verticalement avec un guidage avec des rails (4) et des galets (5). Pendant l'expérience, le plateau est monté à une hauteur de 2 mètres et tombe en chute libre. b) Le modèle est alors stoppé avec un amortisseur progressif sur 5 cm (3). A droite, la cage englobant le dispositif de chute de la maquette.

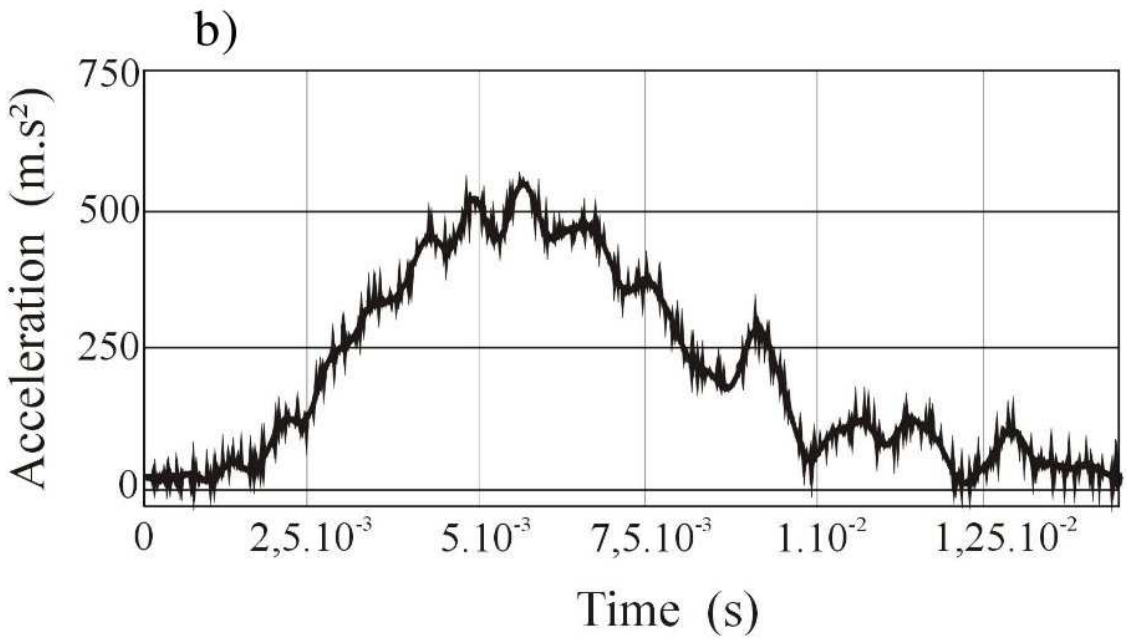

FIGURE 5.8 – L'accélération que subit le modèle est enregistrée par un capteur et un système informatique d'acquisition des données. Plusieurs cycles sont nécessaires pour faire évoluer la rupture. On peut ainsi photographier plusieurs stades d'évolution entre les cycles de chargement.

FIGURE 5.9 – A gauche, Le dispositif stéréo de GEOLSTEREO, constitué de deux appareils photos numériques identiques, disposés au dessus de la maquette. Il est important de paramétrer correctement les appareils, notamment du point de vue du temps d'exposition et de la profondeur de champ, afin d'obtenir des images sans saturation lumineuse ni flou. A droite, l'éclairage de la maquette est réalisé non seulement par l'éclairage ambiant mais aussi à l'aide d'une source lumineuse située au dessus du système stéréo.

quent adopté les règles suivantes :

- Vérifier l'illumination globale de la maquette et essayer de minimiser les trop grandes variations locales d'éclairage.
- Utiliser un ISO minimal et faire en sorte que le temps d'exposition soit de l'ordre d'une seconde, afin d'éviter les scintillements des néons.
- Vérifier visuellement la profondeur de champ. En cas d'images floues, fermer le diaphragme. Régler le focus en manuel.
- Vérifier les différences d'illumination entre deux images des memes paires.

2.4 Déroulement

L'expérience se déroule comme un cycle de chutes libres successives, accompagnées d'un relevé de mesures sur l'accélération subie par le modèle au moment de l'amortissement. Il est clair que chaque chute libre contribue à déstabiliser la structure interne du modèle. Néanmoins, dans le cas des expériences menées à Geos-

ciences Azur, on obsèrve généralement les toutes premières failles à la surface du modèle aux alentours du 40ème cycle. Ce qui implique que la structure interne du modèle commence probablement à etre déstabilisée dans les cycles précédents. Les figures 5.10,5.11,5.12,5.13 et 5.14 montrent des paires stéréo prises respectivement aux cycles 0 (avant chargement), 40, 50, 60 et 70. Les failles apparaissent visiblement sur la deuxieme paire présentée, puis celles-ci évoluent et déclenchent un glissement de terrain. Il faut noter que ces images comportent une zone ombragée sur leur bord supérieur, ce qui rend plus délicate l'estimation du flot de scène en ces points.

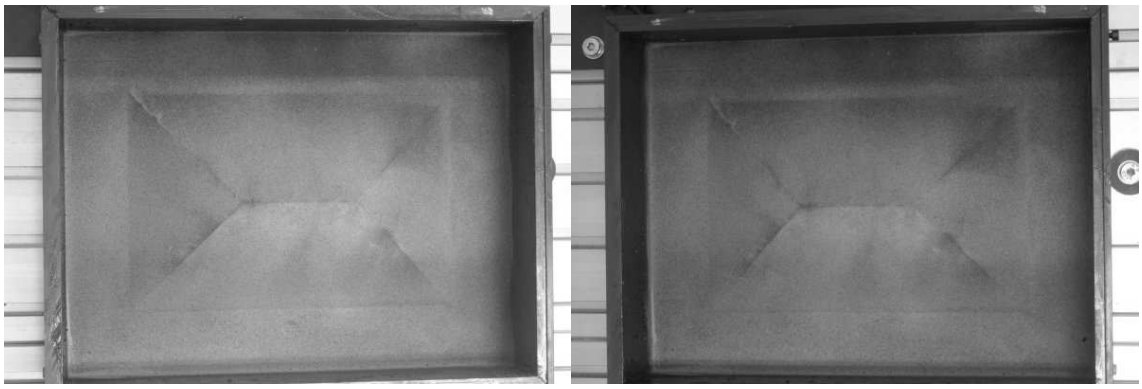

FIGURE 5.10 – Séquence stéréo représentant l'état initial du modèle, prise avant chargement (cycle 0)

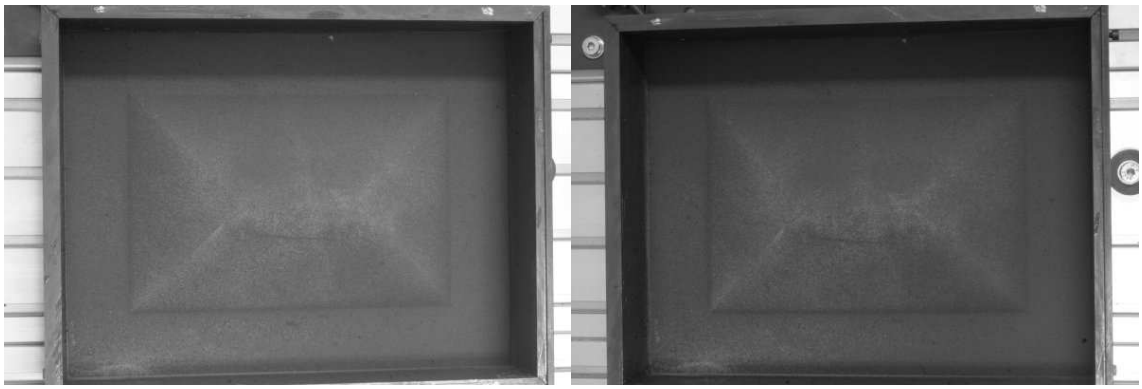

FIGURE 5.11 – Apparition des failles en surface , cycle 40

2.5 Interprétation des résultats

Concernant l'interprétation des résultats de l'algorithme de calcul du flot de scène sur les séquences GEOLSTEREO, il est clair que la reconstruction du champ de

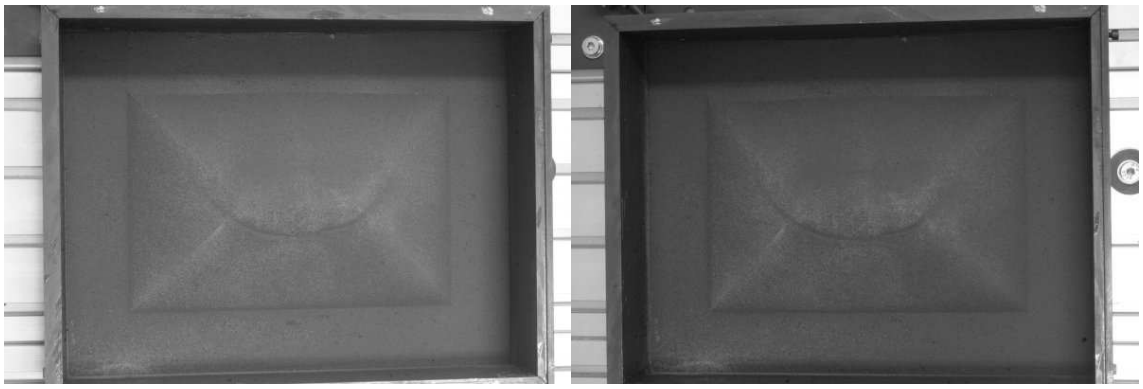

FIGURE 5.12 – Evolution des failles, cycle 50

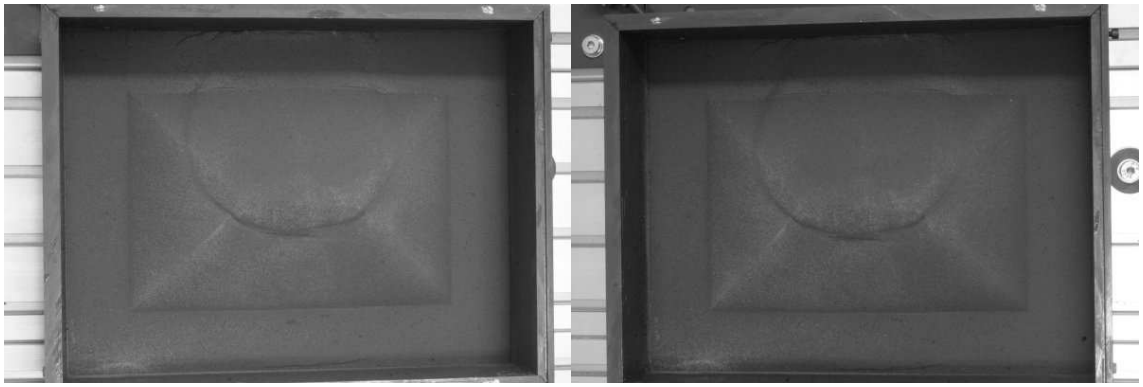

FIGURE 5.13 – Evolution des failles, cycle 60

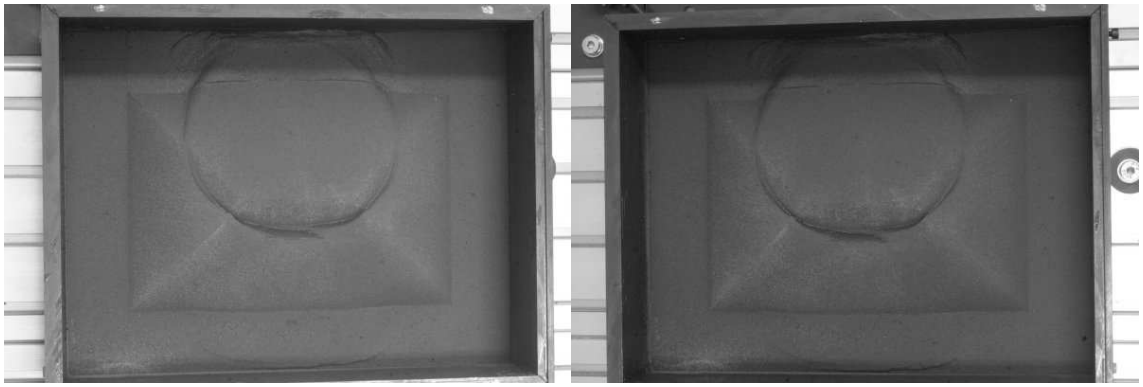

FIGURE 5.14 – Etat final du modéle, glissement de terrain initié au cycle 70

déplacement 3D permet d'analyser les mouvements de diverses parties de la surface qui ne se déplacement pas forcement de manière homogène les unes par rapport aux autres lors d'un glissement de terrain (certaines parties du modele peuvent se désolidariser de leurs voisines, par exemple). En outre, les occultations englobent aussi les fractures apparaissant à la surface du modèle, puisque en ces points de fracture, le flot de scène n'est clairement pas défini. Ainsi, l'algorithme fournit les

cartes de localisation des fractures, ce qui est extremement utile aux géophysiciens, notamment au début de leur apparation, quand leur taille avoisine celle d'un pixel et ne sont donc pas facilement analysables à l'oeil nu. D'autre part, l'analyse des tenseurs de déformations permet la localisation des zones de la maquette subissant les plus fortes contraintes. Dans ce but, l'algorithme génère les cartes de valeurs associées au gradient de flot optique et au gradient de disparité. Ce dernier étant corrélé à la variation en Z subie par la maquette, nous obtenons une analyse des déformations subies par la maquette dans les 3 dimensions spatiales.

3 Estimation du flot de scène

Une fois l'acquisition des séquences réalisée, nous sommes prêts à utiliser notre algorithme afin de déterminer le champ de déplacement 3D des points du modèle initial, ainsi que les occultations, correspondant aux fissures créées par les cycles d'accélérations. Connaissant les paramètres extrinsèques et intrinsèques des caméras, nous pouvons reconstruire le modèle en 3D associé à chaque paire stéréo. La reconstruction du flot de scène est ci essentielle car elle permet l'analyse des mouvements de la surface en détails, et donc l'analyse du glissement de terrain.

3.1 Séquences prétraitées

Les séquences d'images Geolstereo sont d'abord, afin d'atténuer le bruit, lissées par une gaussienne de taille 3*3 pixels, puis sont rectifiées afin de simplifier la mise en correspondance stéréo. La figure 5.15 montre une paire d'images brute puis rectifiée.

D'autre part, les distorsions dues aux capteurs optiques sont éliminées en utilisant Matlab. Le processus est détaillé en annexe A.

Sachant que la maquette est de dimensions 30 cms en longueur et 20 cms en largeur, un pixel représente ici, en rapportant les dimensions réelles aux images rectifiées, un rectangle de longueur 0.125 mm et de largeur 0.102 mm. Il est important de connaître ces dimensions car les résultats qui suivent, notamment les cartes de déplacement liées au flot optique et les cartes de disparité, sont exprimés en pixels.

3.2 Évolution des composantes du flot de scène

Nous présentons dans cette section des résultats obtenus sur la séquence d'images stéréoscopiques présentée en figures 5.10,5.11,5.12,5.13 et 5.14. Comme décrit précédemment, un prétraitement est effectué sur ces images en éliminant les distorsions présentes et en les rectifiant. Les images obtenues sont présentée en figure 5.16 et 5.17. Nous avons choisi ces images afin d'illustrer à la fois l'initiation d'un glissement de terrain et son évolution jusqu'à effondrement d'une partie de la surface. Nous utilisons 5 paires d'images stéréo pour lesquelles nous présentons les résultats du calcul de flot de scène. Les paires stéréo sont simplement désignées par la suite par leur numéro : 1, 2, 3, 4 et 5. 4 étapes de calcul du flot de scène sont évaluées ici.

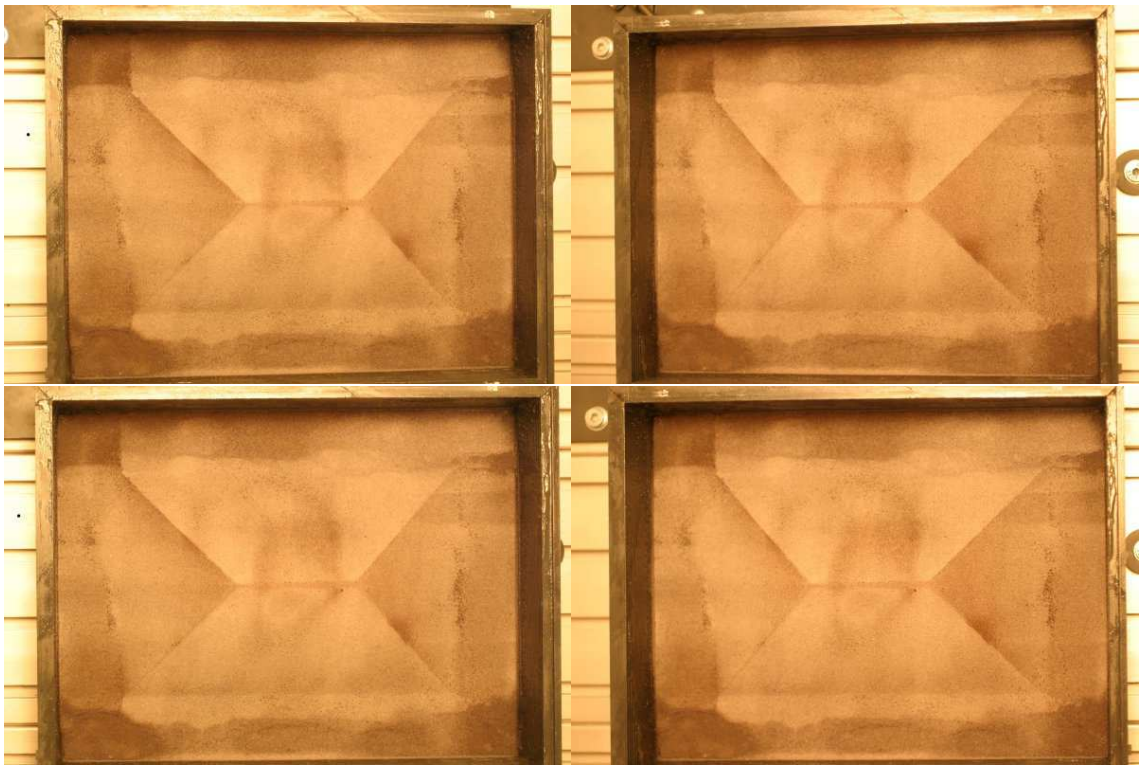

FIGURE 5.15 – En haut : paire stéréo brute. En bas : la meme paire rectifiée.

Les composantes du flot de scène sont présentées sous forme de cartes de valeurs numériques. Les images 5.18, 5.19, 5.20, 5.21 et 5.22 représentent les cartes de valeurs associées aux composantes du flot de scène. Les plages de valeurs sont indiquées en mms dans les légendes. Il faut noter que la plupart du temps les bornes inférieures et supérieures de ces plages de valeurs sont atteintes au bord, et peuvent correspondre à des zones occultées ou bien au bord supérieur des images, situé dans l'ombre et pour lequel l'estimation du flot de scène est ainsi moins fiable.

3.3 Évolution des cartes d'occultation

L'évolution temporelle des cartes d'occultation estimées par l'algorithme est présentée par les images 5.23, 5.24, 5.25 et 5.26. Comme on peut voir l'évolution des fractures sur les images 5.16 et 5.17, on peut observer une évolution similaire sur les cartes d'occultation. Les cartes d'occultations du flot de scène obtenues constituent

FIGURE 5.16 – Paires d'images stéréo 1, 2, 3. L'initiation du glissement est clairement visible, suivie de l'effondrement d'une partie de la surface.

ainsi un premier indicateur de l'évolution de l'effondrement de la surface du modèle.

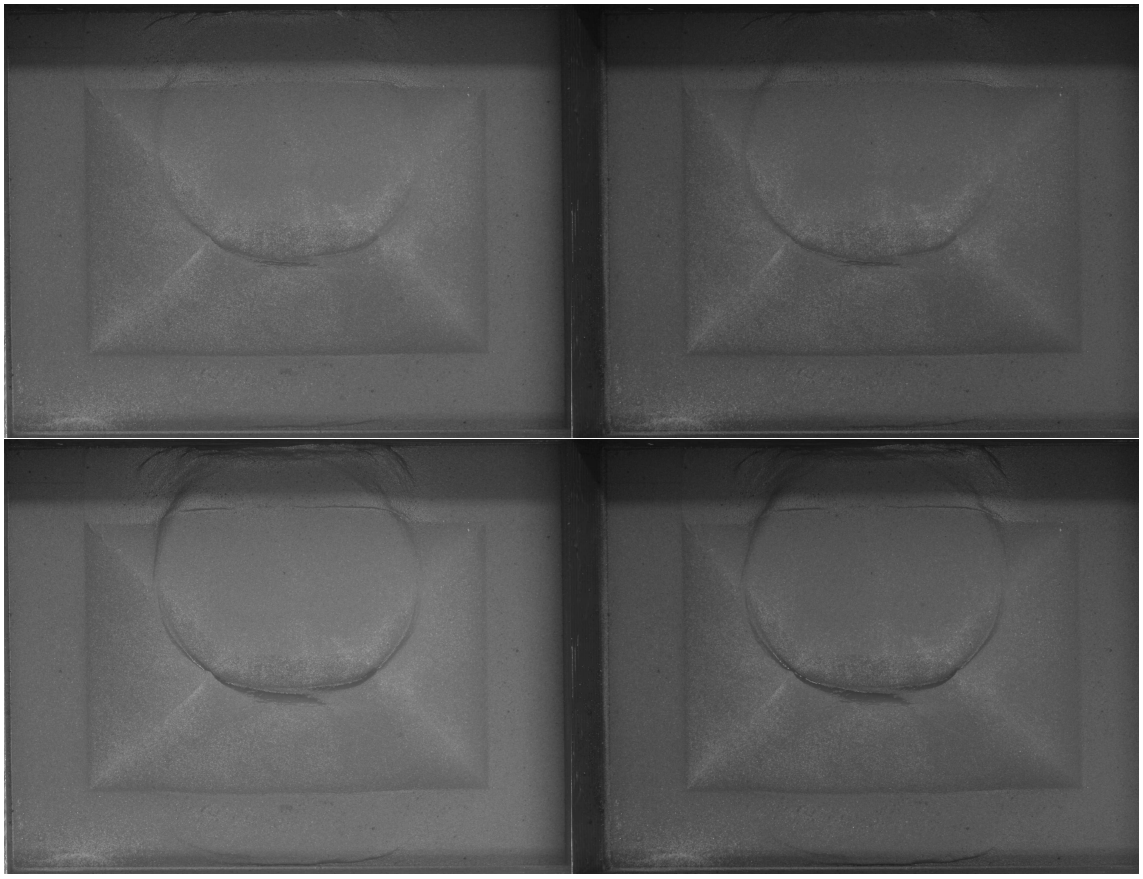

FIGURE 5.17 – Paires d'images stéréo 4 et 5. Le glissement évolue jusqu'à effondrement d'une partie de la surface.

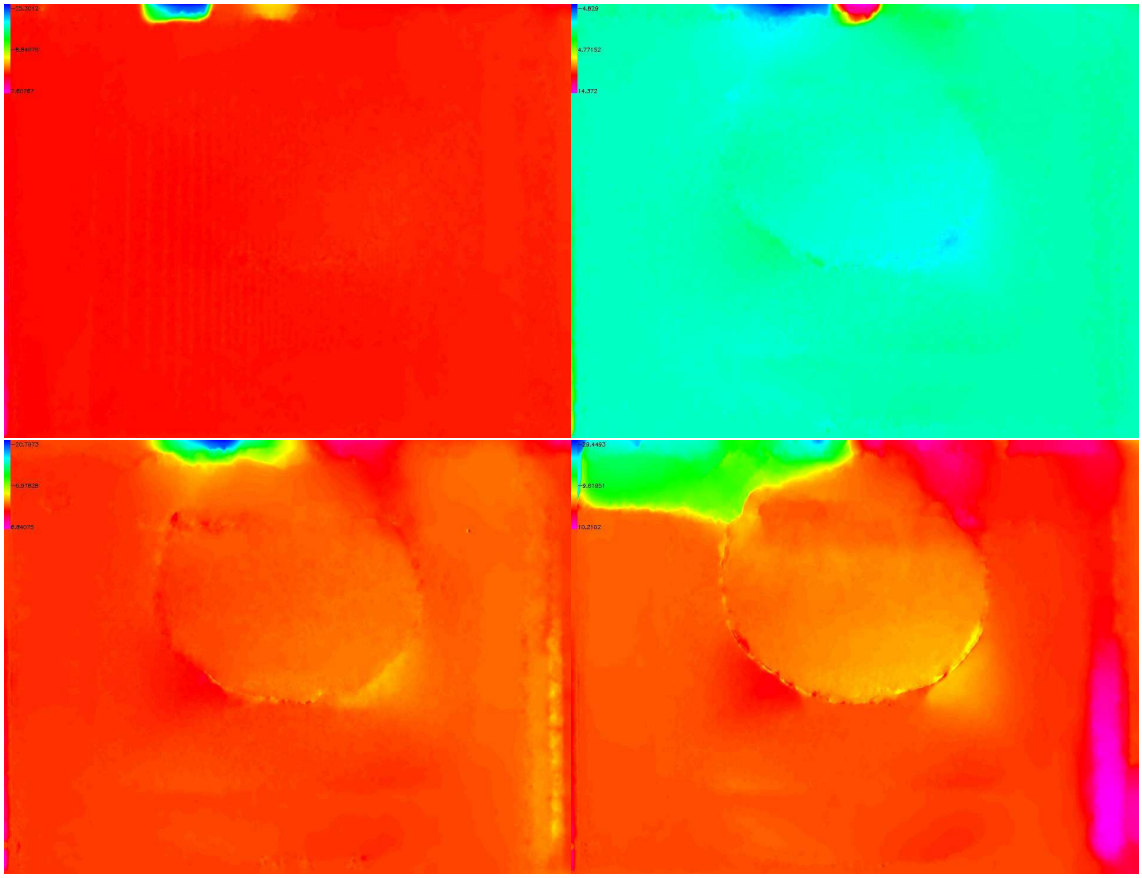

FIGURE 5.18 – Évolution de la composante flot optique horizontal U du flot de scène. En haut à gauche : entre les paires 1 et 2. $U \in [-25.3, 7.6]$ ce qui correspond à un déplacement réel [-7.11mm, 2.13mm] .En haut à droite : entre les paires 2 et 3, [3.13mm, 4.05mm]. En bas à gauche : entre les paires 3 et 4, [-5.85mm, 1.92mm]. En bas à droite : entre les paires 4 et 5, [-8.28mm, 2.86mm]. On constate le déplacement horizontal progressif d'un des versants de la maquette. D'autre part, sur les images du bas, le déplacement est plus important sur la droite de la faille (zone jaune), impliquant un mouvement non isotrope.

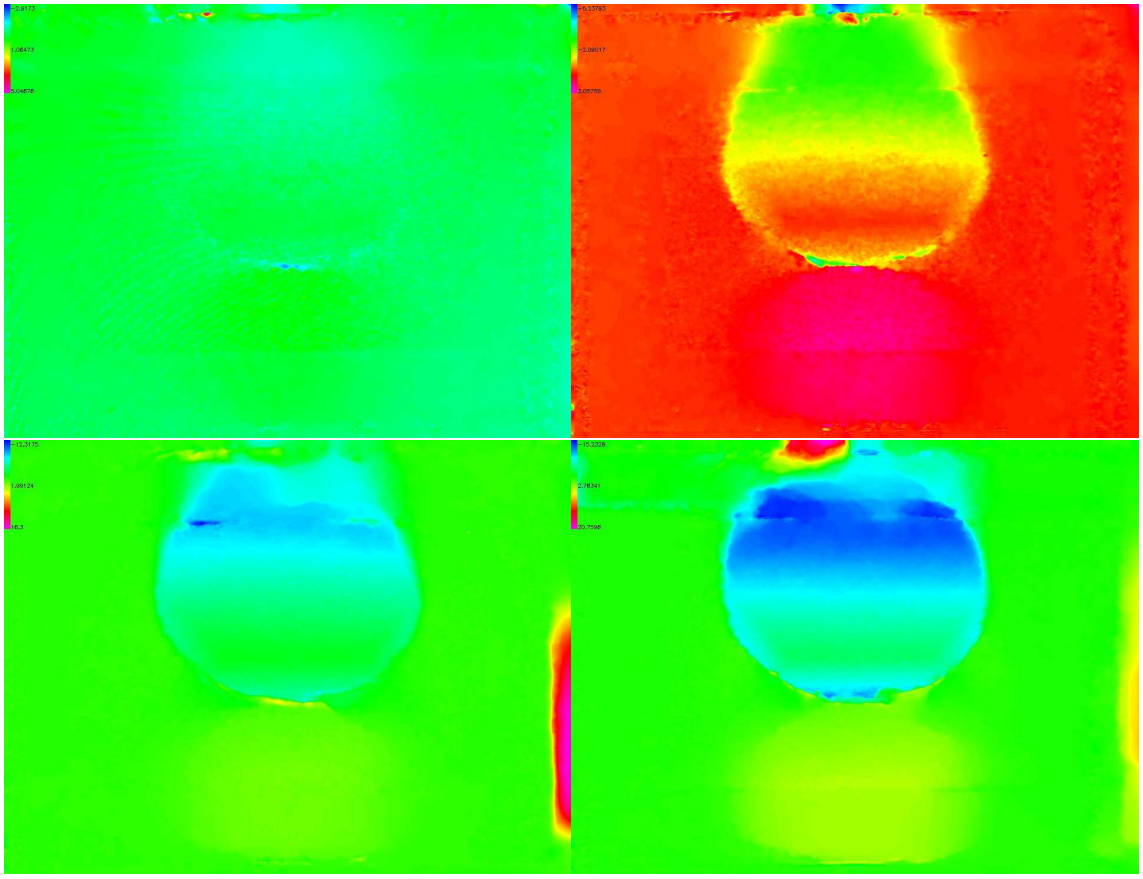

FIGURE 5.19 – Évolution de la composante flot optique vertical V du flot de scène. En haut à gauche : entre les paires 1 et 2, [-0.666mm, 1.15mm]. En haut à droite : entre les paires 2 et 3, [-1.42mm, 0.47mm]. En bas à gauche : entre les paires 3 et 4, [-2.84mm, 3.73mm]. En bas à droite : entre les paires 4 et 5, [-3.49mm, 4.75mm]. Le déplacement vertical du versant de la maquette s'effondrant est progressif. On observe également l'apparition d'un mouvement symétrique, de moins grande amplitude, sur l'autre versant. La partie sommitale de la maquette s'effondre. L'interprétation conjointe des valeurs numériques de U et V indique en pratique un mouvement rotationnel.

FIGURE 5.20 – Codage (teinte, saturation) de l'évolution de la composante flot optique (U, V) du flot de scène. En haut à gauche : entre les paires 1 et 2. En haut à droite : entre les paires 2 et 3. En bas à gauche : entre les paires 3 et 4. En bas à droite : entre les paires 4 et 5. Outre l'effondrement d'un des versants, on constate bien l'apparition d'un déplacement symétrique sur l'autre versant.

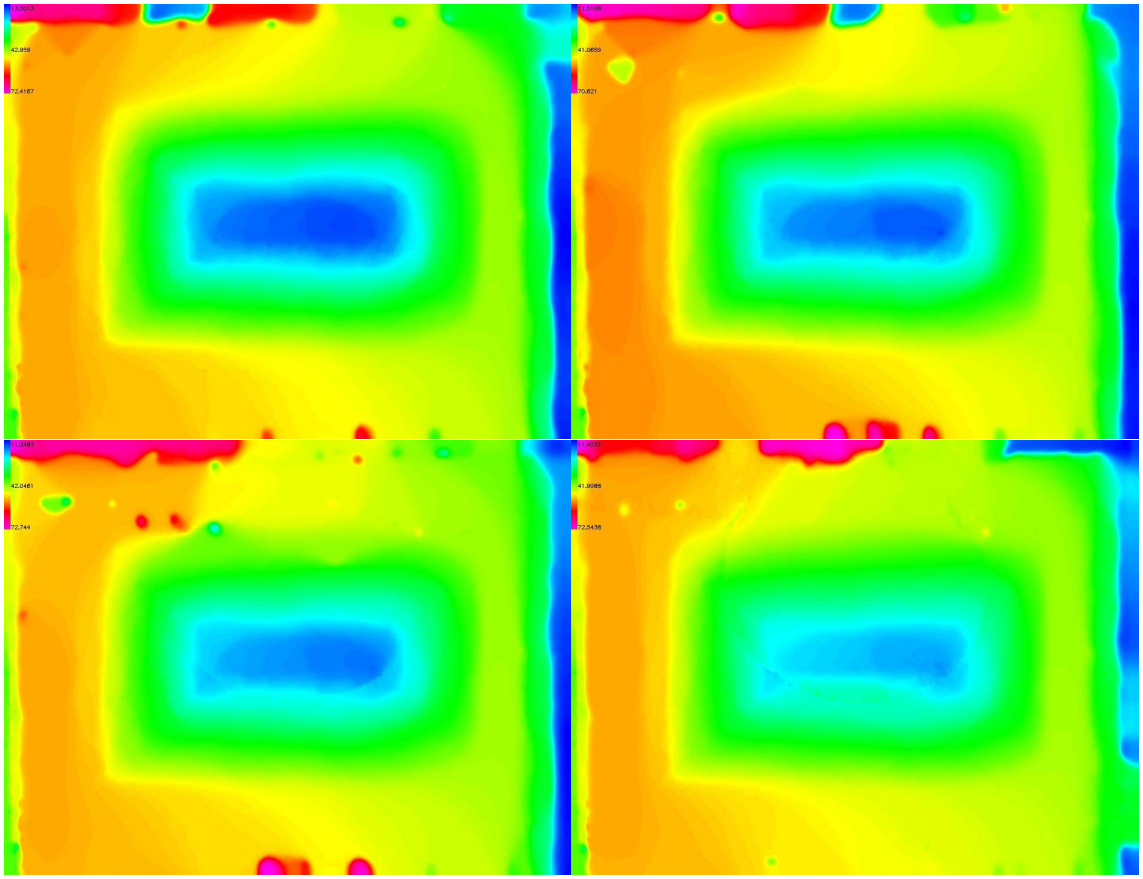

FIGURE 5.21 – Évolution de la composante disparité D du flot de scène. En haut à gauche : entre les paires 1 et 2, [3.8mm, 20.36mm]. En haut à droite : entre les paires 2 et 3, [3.24mm, 19.87mm]. En bas à gauche : entre les paires 3 et 4, [3.2mm, 20.45mm]. En bas à droite : entre les paires 4 et 5, [3.21mm, 20.4mm]. On constate l'apparition de la faille sur cette dernière image, permettant de constater que la rupture en surface de la maquette entraîne aussi un effondrement en Z.

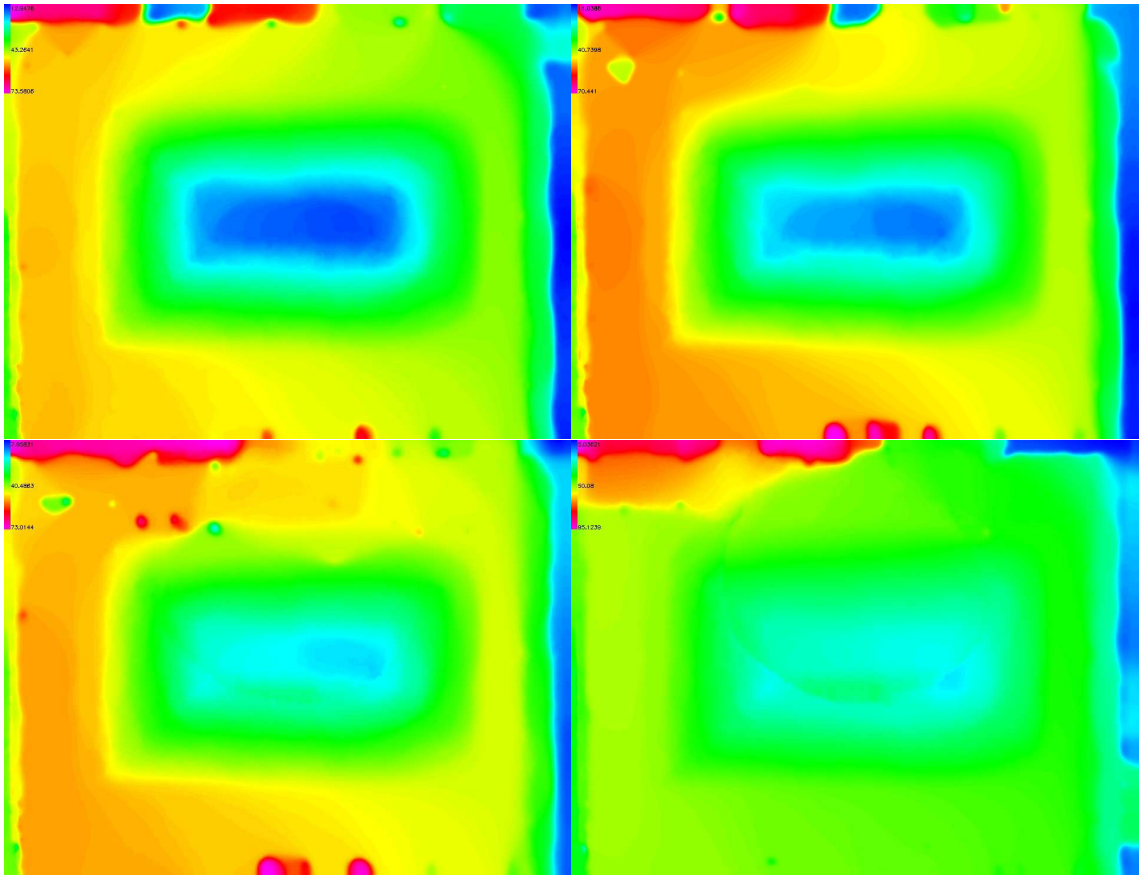

FIGURE 5.22 – Évolution de la composante disparité D' du flot de scène. En haut à gauche : entre les paires 1 et 2, [3.65mm, 20.7mm]. En haut à droite : entre les paires 2 et 3, [3.105mm, 19.8mm]. En bas à gauche : entre les paires 3 et 4, [2.3mm, 20.52mm]. En bas à droite : entre les paires 4 et 5, [1.42mm, 26.7mm]. De même que dans le cas de D, on observe l'apparition d'une discontinuité de profondeur correspondant à l'apparition de la faille et à un effondrement de la maquette dans le direction des Z.

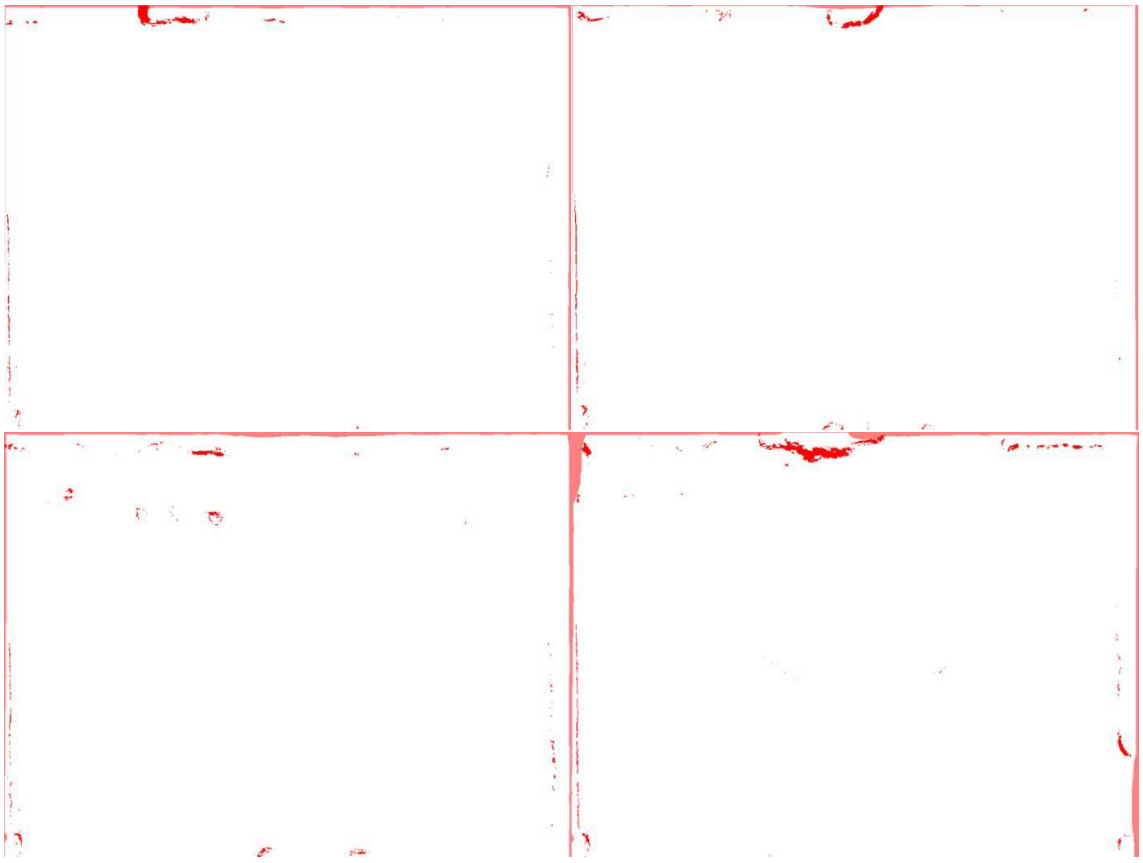

FIGURE 5.23 – Évolution des occultations associées à la composante flot optique gauche du flot de scène. En haut à gauche : entre les paires 1 et 2. En haut à droite : entre les paires 2 et 3. En bas à gauche : entre les paires 3 et 4. En bas à droite : entre les paires 4 et 5.

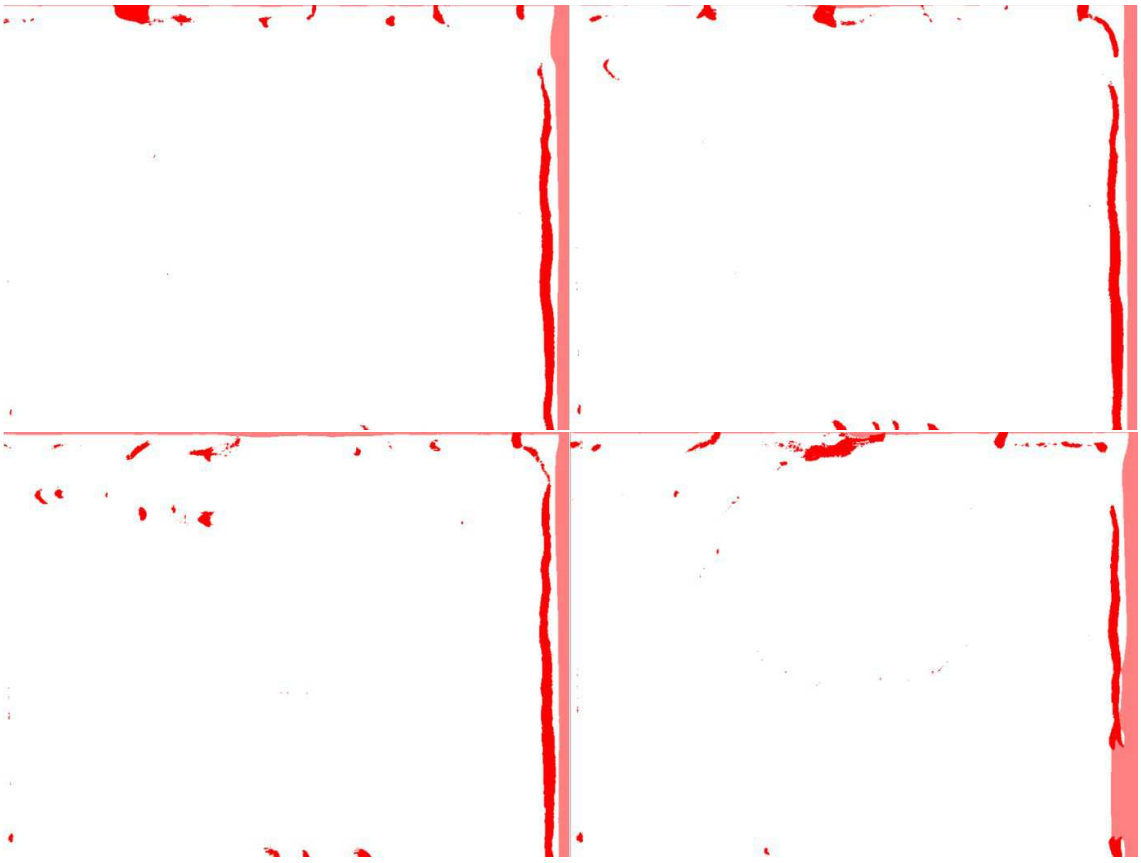

FIGURE 5.24 – Évolution des occultations associées à la composante flot optique droit du flot de scène. En haut à gauche : entre les paires 1 et 2. En haut à droite : entre les paires 2 et 3. En bas à gauche : entre les paires 3 et 4. En bas à droite : entre les paires 4 et 5.

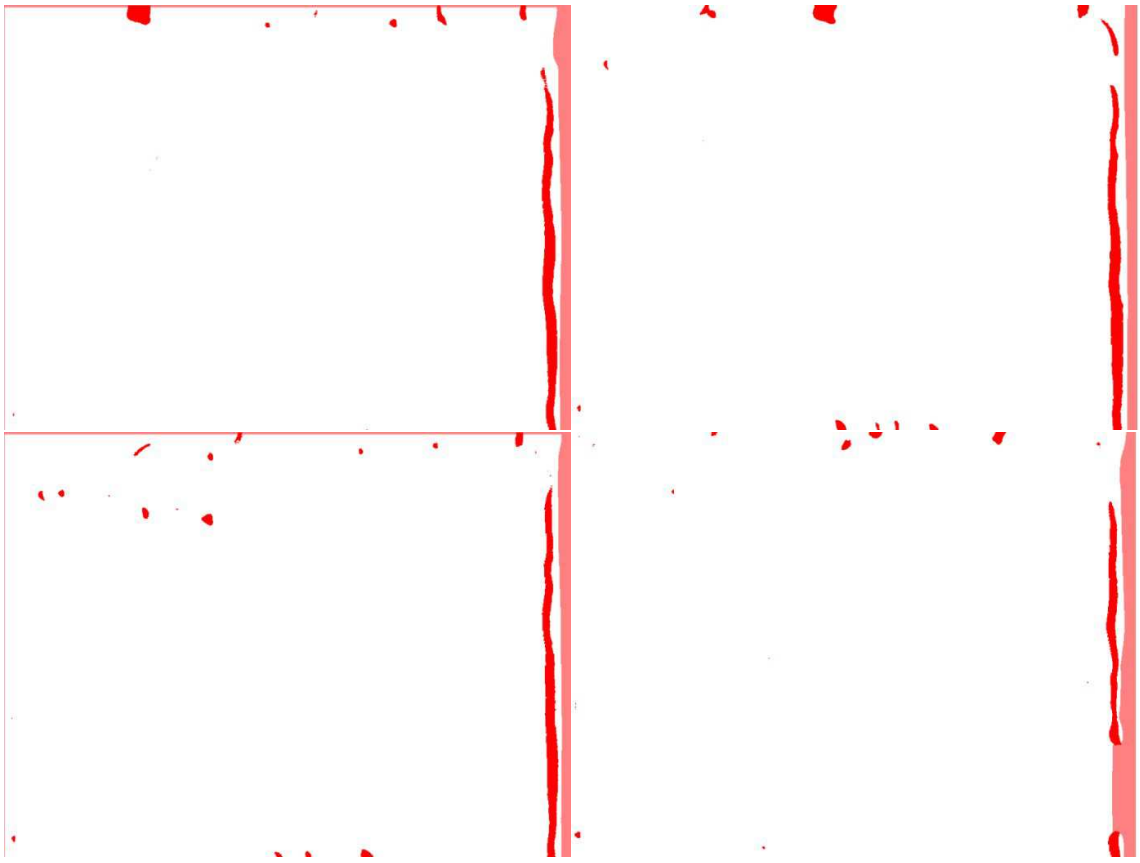

FIGURE 5.25 – Évolution des occultations associées à la composante disparité d du flot de scène. En haut à gauche : entre les paires 1 et 2. En haut à droite : entre les paires 2 et 3. En bas à gauche : entre les paires 3 et 4. En bas à droite : entre les paires 4 et 5.

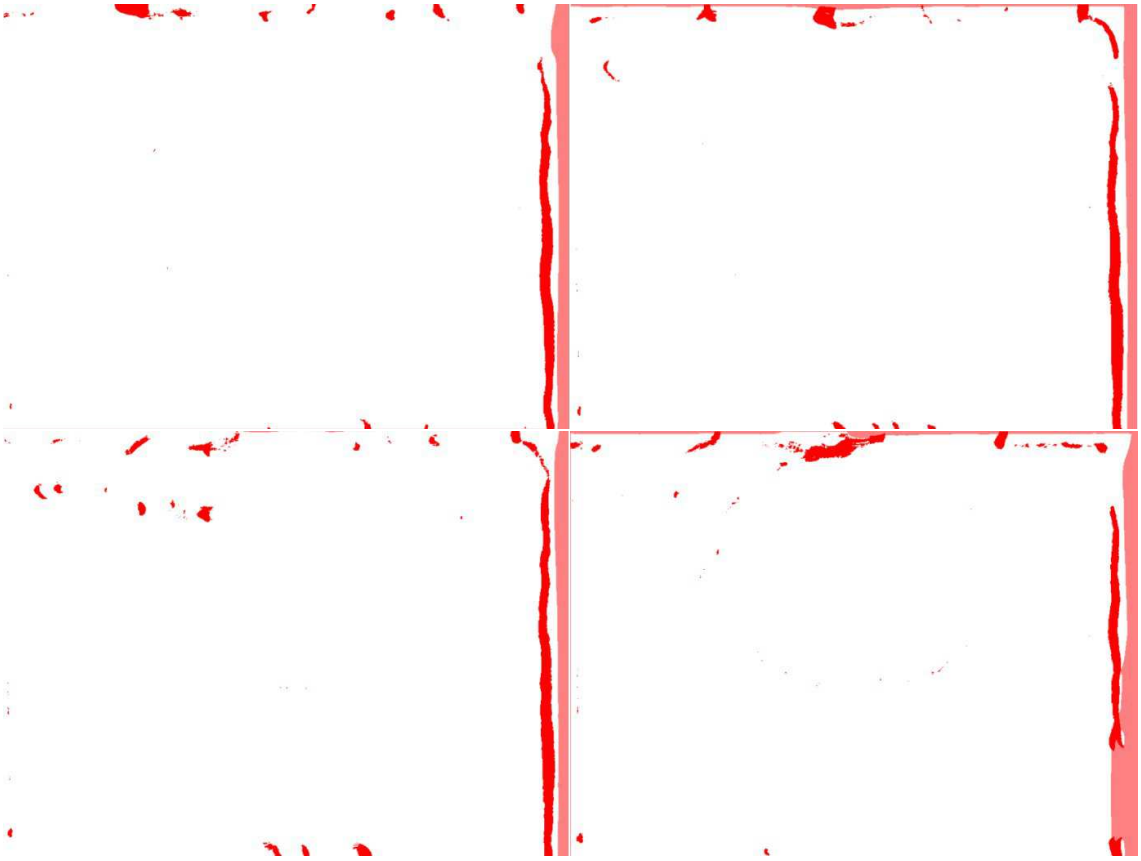

FIGURE 5.26 – Évolution des occultations associées à la composante d' du flot de scène. En haut à gauche : entre les paires 1 et 2. En haut à droite : entre les paires 2 et 3. En bas à gauche : entre les paires 3 et 4. En bas à droite : entre les paires 4 et 5.

3.4 Evolution des gradients de déplacements

L'algorithme estime aussi la norme des gradients de déplacements liés à la projection 2D du flot de scène (ce qui revient à estimer un tenseur de déformations), ce qui permet d'évaluer les endroits ou les déformations sont les plus fortes. On doit ainsi en théorie détecter comme pour les occultations du flot de scène les fractures à la surface du modèle. Les résultats numériques confirment cet à priori sur les figures 5.27. D'autre part, on peut aussi évaluer la norme du gradient du flot de disparité. Cette dernière étant liée à la profondeur des points de la scène 3D, dénotée Z, cette norme est étroitement corrélée avec les variations de profondeur durant l'expérience. Comme dans le cas du gradient de la composante 2D du flot de scène, et étant donné que les discontinuités de flot optique et de disparité sont liées dans la formulation de l'énergie J, les fractures du modèles doivent également être révélées et on constate que c'est bien le cas sur la figure illustrant l'évolution du gradient de disparité 5.28. Enfin on constate que le plan de rupture est assez complexe, ce qui est conforme aux observations menées sur le terrain par les géophysiciens.

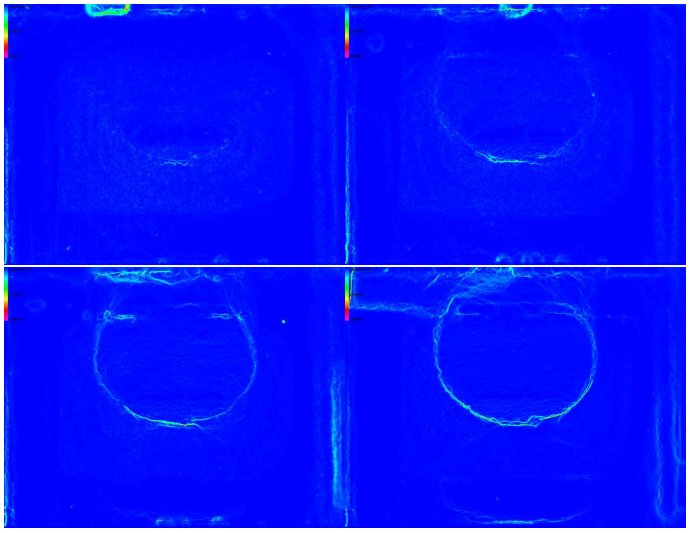

FIGURE 5.27 – Évolution de la norme du gradient de la projection du flot de scène. En haut à gauche : entre les paires 1 et 2. En haut à droite : entre les paires 2 et 3. En bas à gauche : entre les paires 3 et 4. En bas à droite : entre les paires 4 et 5.

Les résultats expérimentaux montrent ainsi que le calcul des tenseurs de déformation liés au flot de scène permet un suivi précis des fractures apparaissant au cours de l'expérience sur la maquette.

3.5 Reconstruction 3D

Afin de retrouver les paramètres intrinsèques et extrinsèques des appareils photos, nous avons utilisé le module MATLAB Calibration Toolbox. Celui-ci utilise

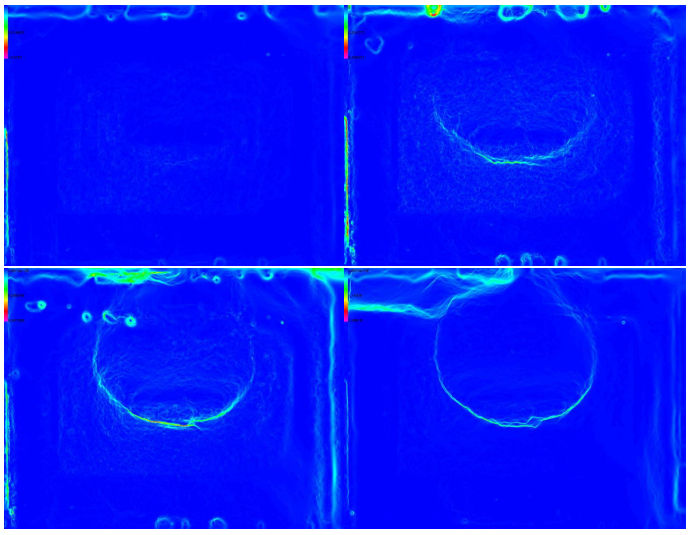

FIGURE 5.28 – Évolution de la norme du gradient du flot de disparité. En haut à gauche : entre les paires 1 et 2. En haut à droite : entre les paires 2 et 3. En bas à gauche : entre les paires 3 et 4. En bas à droite : entre les paires 4 et 5.

differentes images d'une mire afin de déterminer les paramètres des caméras. Cet outil utilise une approche d'optimisation aux moindres carrés pour évaluer ces paramètres en utilisant des points sur la mire désignés par l'utilisateur. Le processus de calibration est décrit et illustré en annexe A. Une fois ces paramètres connus, nous sommes en mesure d'estimer la matrice de reonctruction, de restituer ainsi le modèle en 3D, et d'évaluer le flot de scène par différence des positions des points 3D entre deux instants consécutifs de prises de vue.

L'algorithme effectue la reconstruction 3D et stocke le maillage de la surface du modèle ainsi que les données vectorielles au format de la librairie graphique VTK, disponible à l'adresse `www.vtk.org`. Nous utilisons le logiciel de visualisation open source Paraview, téléchargeable à l'adresse `www.paraview.org`, afin de visualiser les reconstructions 3D obtenues. Les images 5.29, 5.30, 5.31 et 5.32 montrent la reconstruction brute de la surface, ainsi qu'une représentation couleurs où les valeurs du module du flot de scène sont représentées en chaque point des surfaces reconstruites. On observe bien l'apparition et le déroulement d'un glissement de terrain à la surface, ce qui permet alors d'effectuer des mesures sur ce glissement. Nous suivons l'évolution des déformations subies par le sommet de la maquette au cours de l'expérience. L'image 5.33 montre le sommet au début de l'apparition du glissement. Les images 5.34 et 5.35 montrent l'évolution de la déformation subie jusqu'à l'état

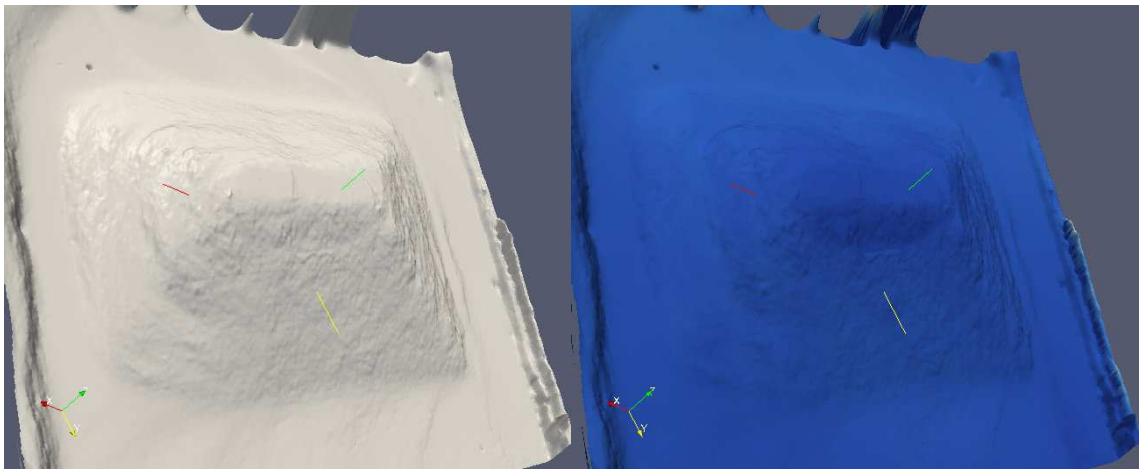

FIGURE 5.29 – Reconstruction de la surface estimée avec la paire d'images 1. A gauche : reconstruction brute. à droite, les couleurs indiquent le module du flot de scène en chaque point de la surface, estimé entre les paires stéréo 1 et 2. Le flot de scène est quasi nul presque partout sur la surface.

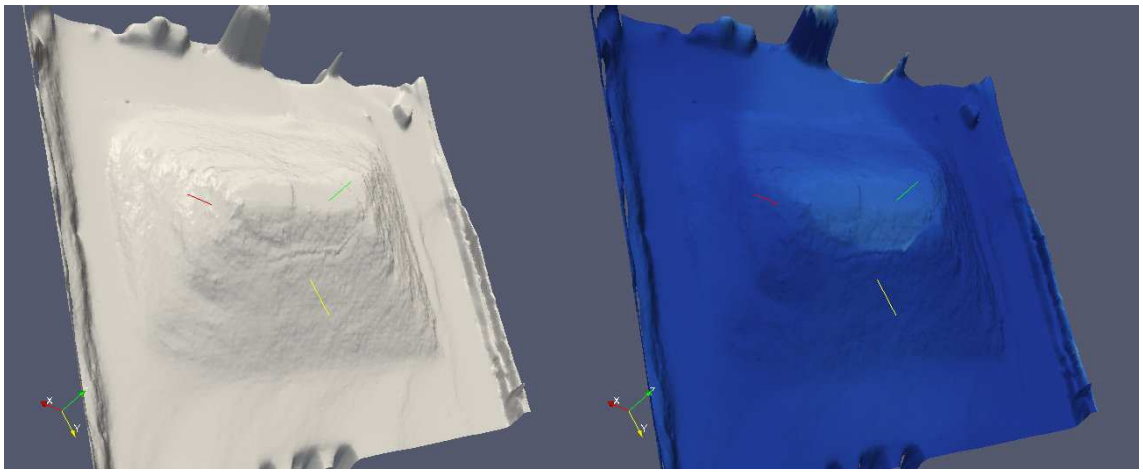

FIGURE 5.30 – Reconstruction de la surface estimée avec la paire d'images 2. A gauche : reconstruction brute. à droite, les couleurs indiquent le module du flot de scène en chaque point de la surface, estimé entre les paires stéréo 2 et 3. Le flot de scène devient non nul sur le versant commençant à s'effondrer.

d'effondrement final du sommet illustré par la figure 5.36.

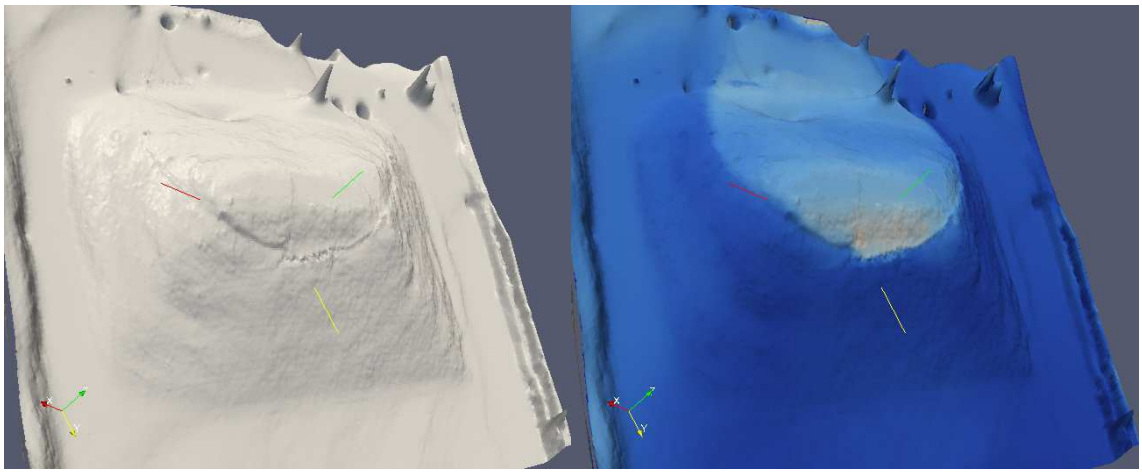

FIGURE 5.31 – Reconstruction de la surface estimée avec la paire d'images 3. A gauche : reconstruction brute. À droite, les couleurs indiquent le module du flot de scène en chaque point de la surface, estimé entre les paires stéréo 3 et 4. Le flot de scène voit son amplitude croître sur le versant glissant.

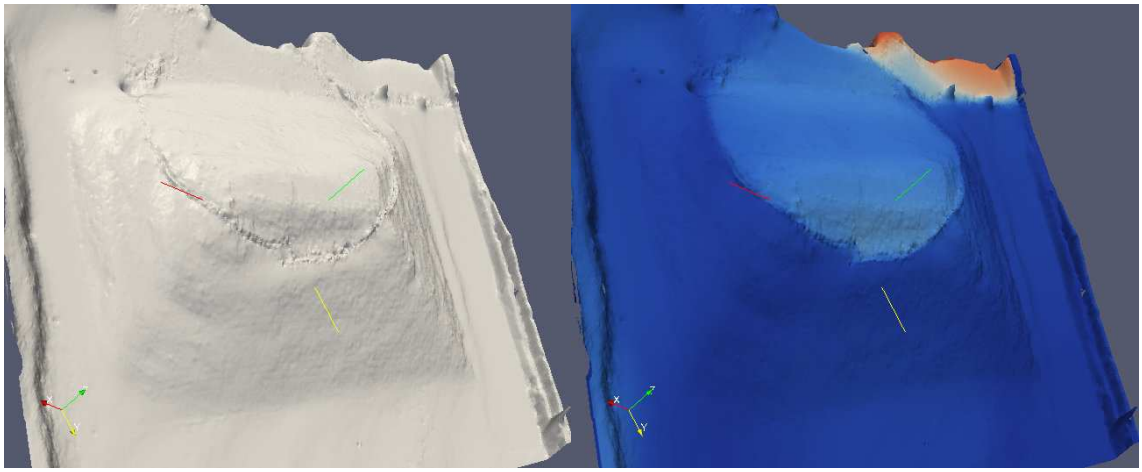

FIGURE 5.32 – Reconstruction de la surface estimée avec la paire d'images 4. A gauche : reconstruction brute. À droite, les couleurs indiquent le module du flot de scène en chaque point de la surface, estimé entre les paires stéréo 4 et 5. Le flot de scène est d'amplitude maximum en fin d'expérience.

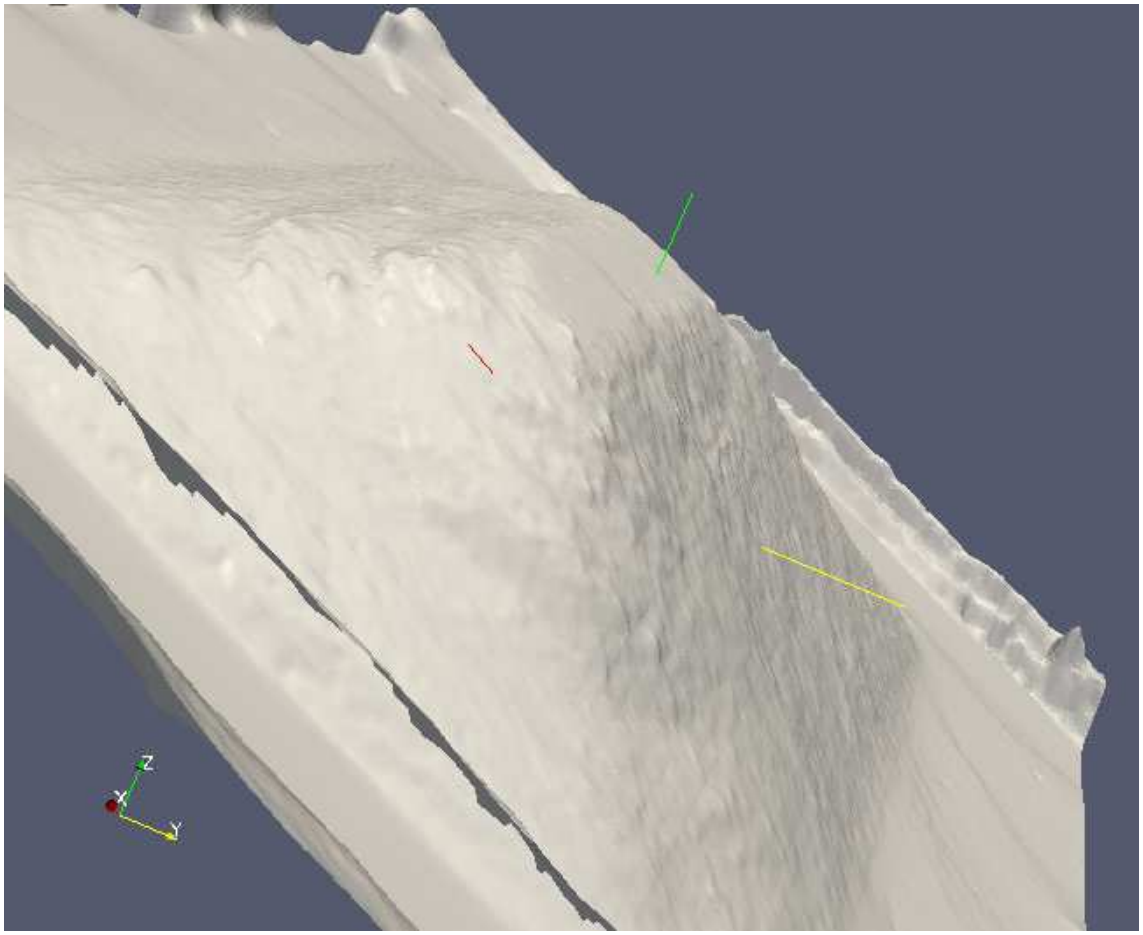

FIGURE 5.33 – Comportement du sommet de la surface au moment de l'initialisation du glissement de terrain.

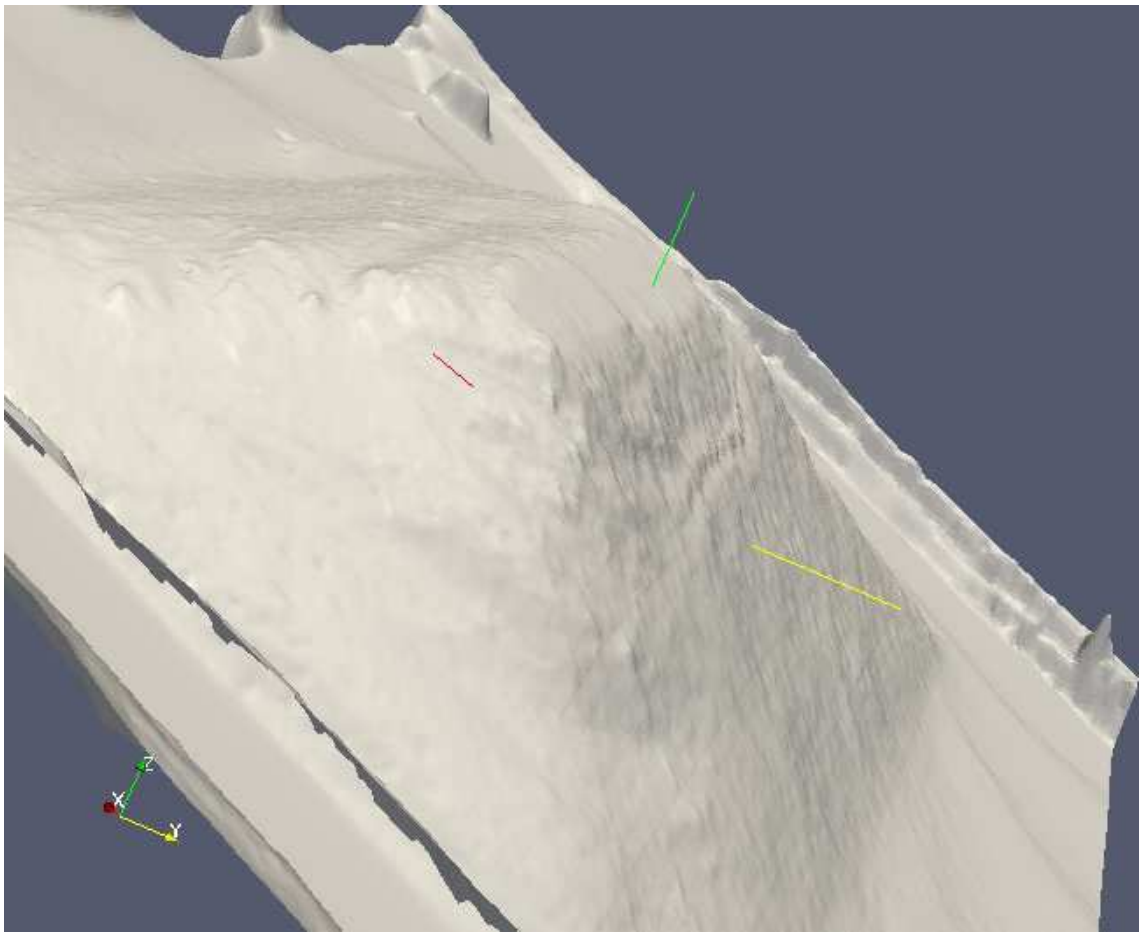

FIGURE 5.34 – Comportement du sommet de la surface au début du glissement de terrain.

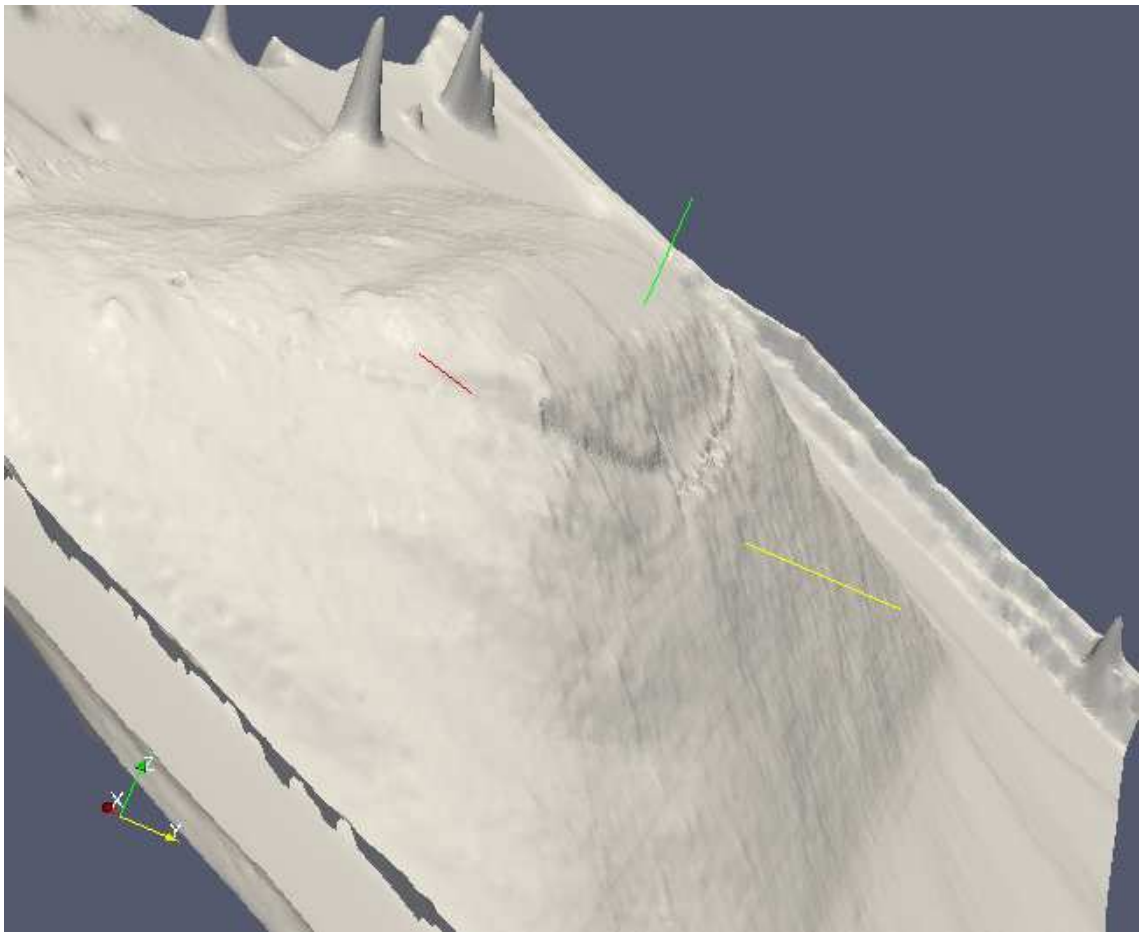

FIGURE 5.35 – Comportement du sommet de la surface au début du glissement de terrain.

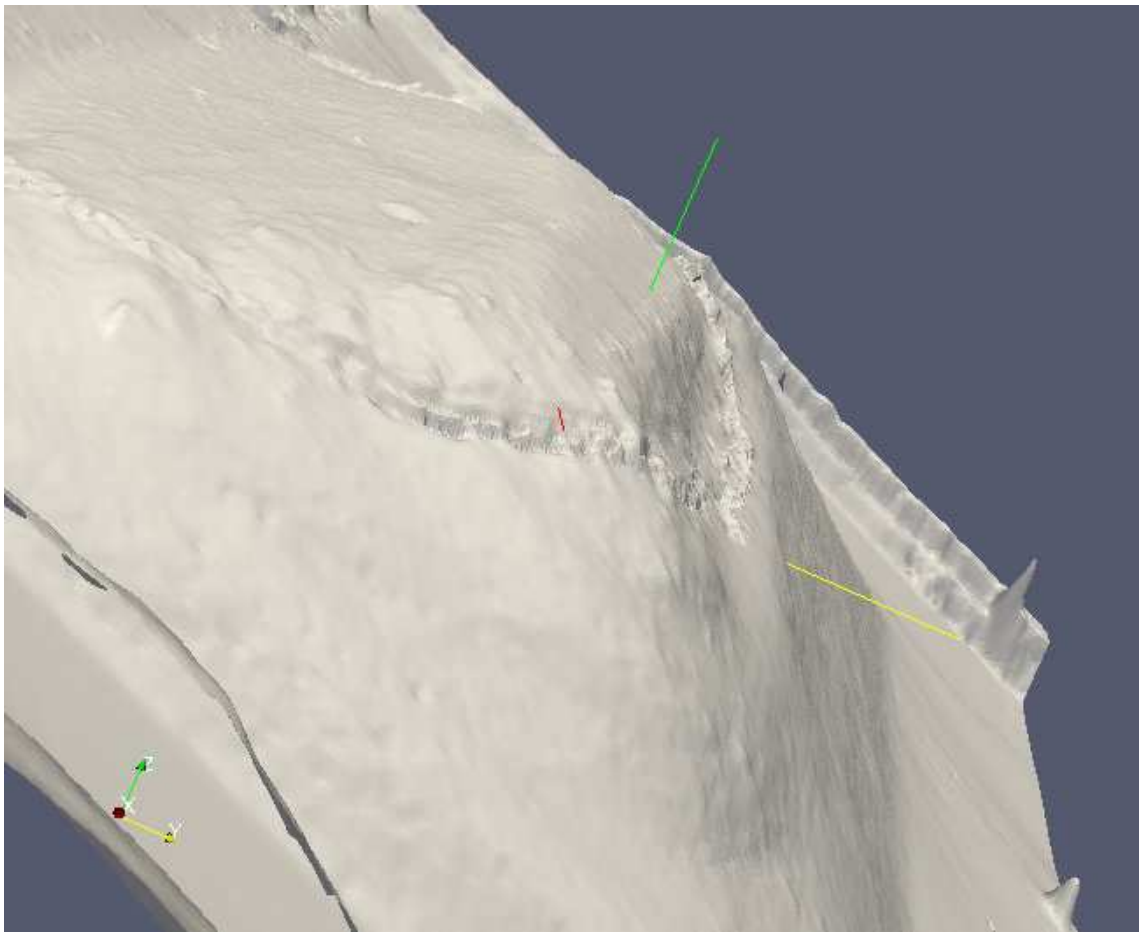

FIGURE 5.36 – État du sommet de la surface suite au glissement de terrain.

3.6 Flot de scène

Enfin, nous montrons quelques exemples numériques d'estimation du flot de scène, et nous examinons plus particulièrement le versant de la maquette subissant le plus fort déplacement au cours des expériences. Pour plus de lisibilité sur les figures, nous avons multiplié l'amplitude du flot de scène en chaque point de la surface par 2. D'autre part, l'orientation du champ de déplacement nous a imposé de visualiser la maquette "par dessous". Les images 5.37, 5.38, 5.39, 5.40 montrent l'évolution du flot de scène estimé sur le versant effondré au cours de l'expérience. Les géophysiciens sont donc capables, à l'aide du flot de scène, d'analyser le mouvement de la surface d'un point de vue numérique et de détecter les endroits de la surface où se produisent les ruptures (ce qui correspond aux zones où le flot de scène devient subitement non nul). La visualisation de l'orientation du champ de vecteurs permet de mettre en évidence le caractère rotationnel du flot de scène en fin d'expérience, ce qui était prédit au vu du comportement de la composante flot optique, et ce qui correspond aux observations sur le terrain.

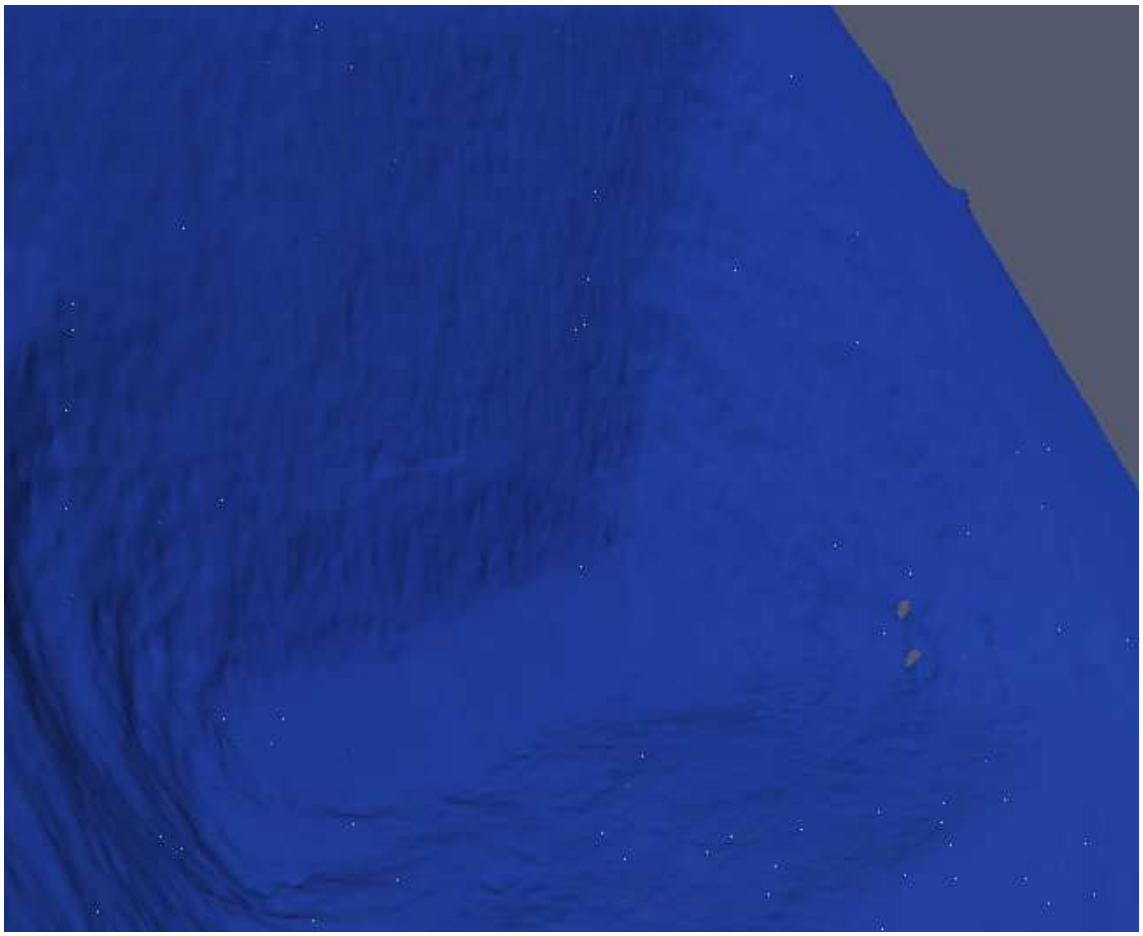

FIGURE 5.37 – Flot de scène, quasi nul, au moment de l'apparition de la fracture dans les séquences d'images (paires d'images 1 et 2)

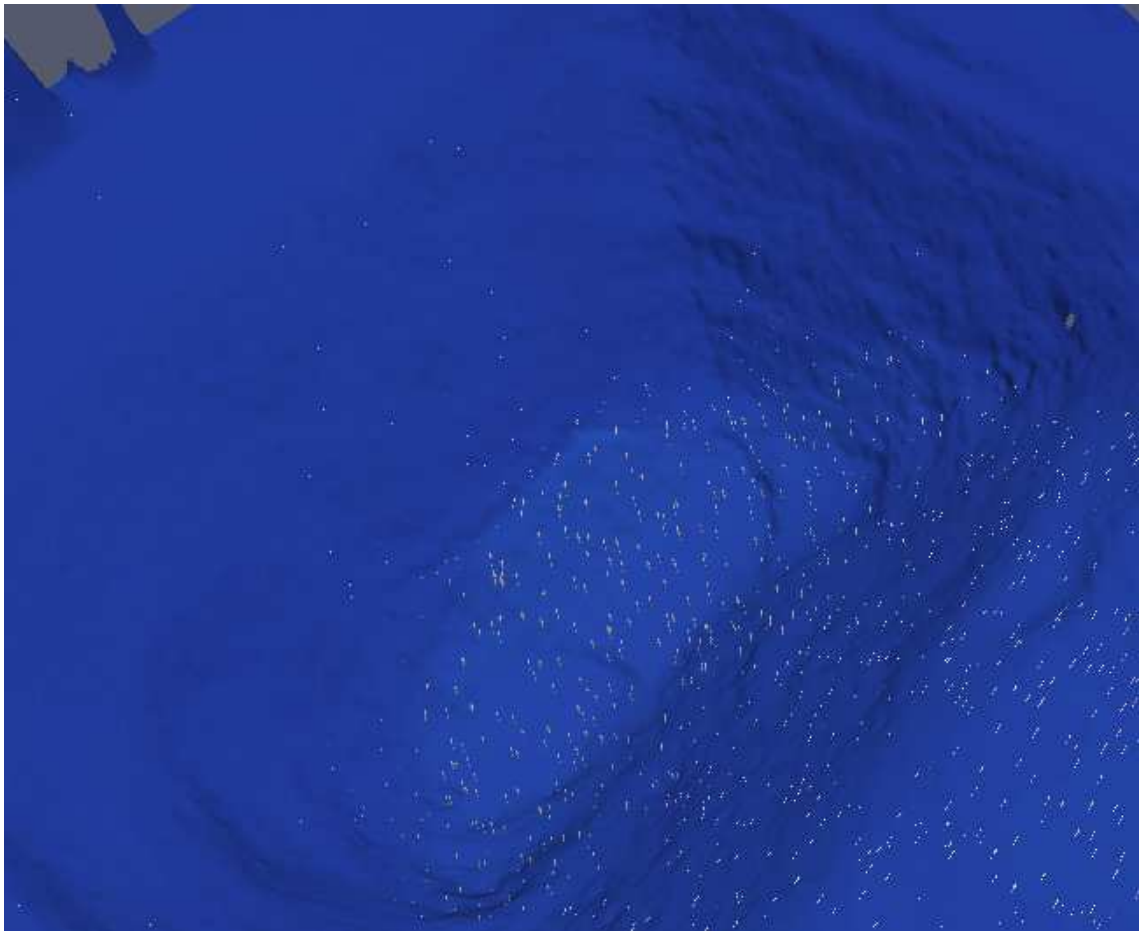

FIGURE 5.38 – Flot de scène au tout début du glissement de terrain (paires d'images 2 et 3)

4 Conclusion

Nous avons vu dans ce chapitre une application originale du flot de scène à la géophysique. Notre approche aide les physiciens à éviter les problèmes liés aux simulations numériques délicates dans le cadre de l'analyse des glissements de terrain en montagne. Nous avons prouvé que l'on peut détecter et suivre finement les fractures apparaissant au cours du temps sur une maquette à l'échelle réduite soumise à des contraintes physiques.

Nos résultats concernent à la fois la reconstruction 3D et l'estimation du flot de scène en chaque point, ou presque, de la reconstruction surfacique en jeu dans une expérience donnée. Nous avons vu que le glissement attendu par les géophysiciens est en translation ou bien en rotation. Le caractère rotationnel du flot de scène reconstruit concorde donc tout à fait avec les attentes des physiciens.

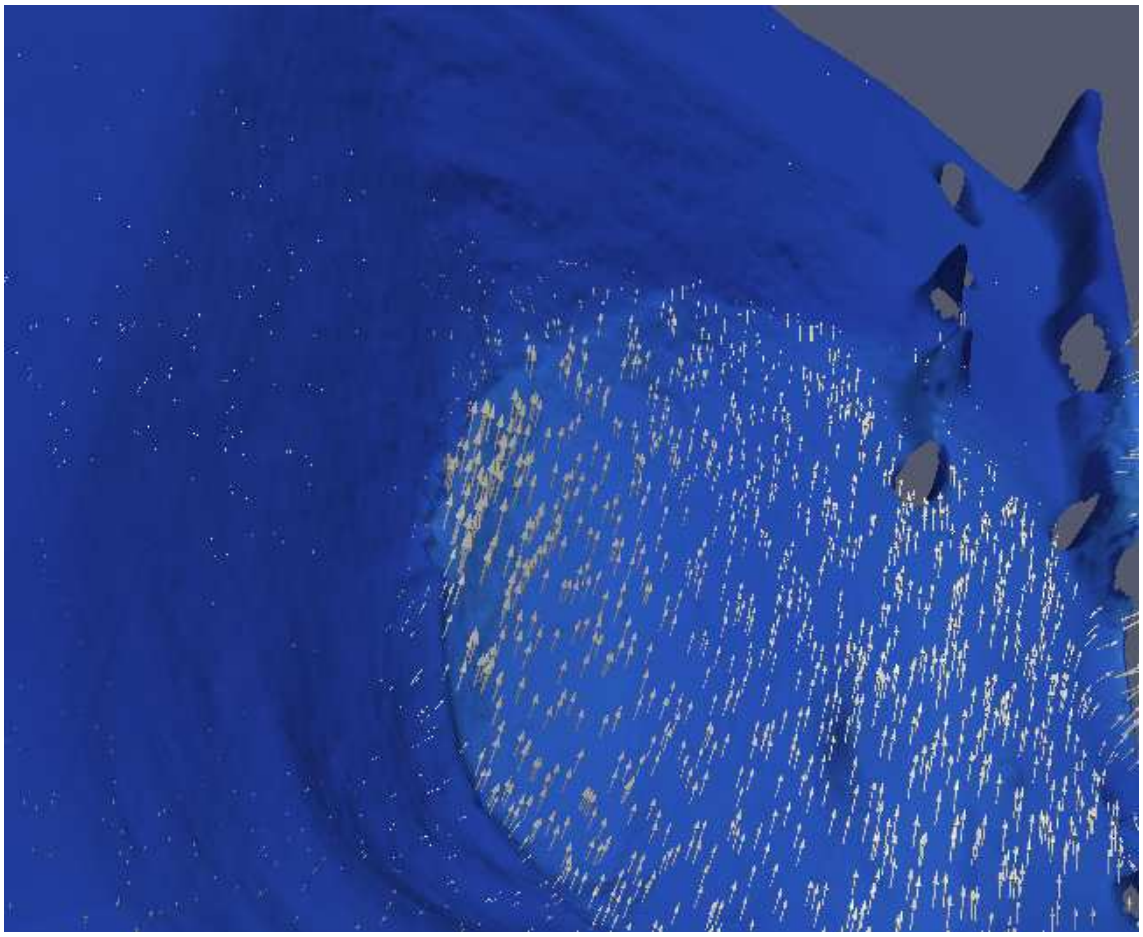

FIGURE 5.39 – Flot de scène pendant le glissement de terrain (paires d'images 3 et 4). Son amplitude croît sur la partie de la surface en plein effondrement. On constate l'apparition d'un flot de scène non nul sur l'autre versant, mais d'amplitude négligeable par rapport au déplacement subi par le versant qui s'effondre.

Enfin, notre algorithme donne également aux physiciens des valeurs annexes, telles que le module du champ de vitesse ou bien les cartes de déformations, sous forme de carte des modules des tenseurs.

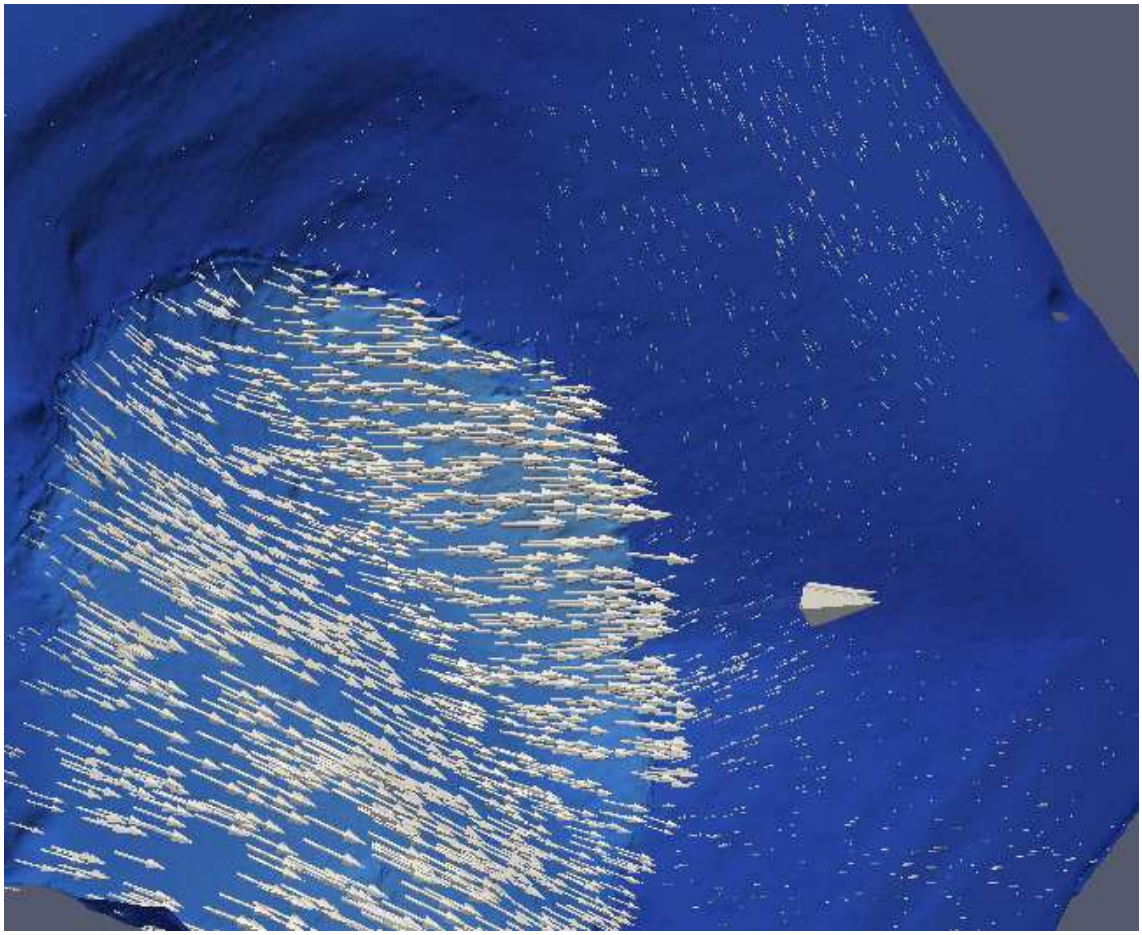

FIGURE 5.40 – Flot de scène en fin d'expérience (paires d'images 4 et 5). Le caractère rotationnel du champ de vecteurs est clairement visible et correspond à ce qui est observé dans la nature.

Chapitre 6

Conclusion

Dans cette thèse, nous avons présenté une méthode variationnelle pour estimer le flot de scène et les occultations associées simultanément à partir de séquences stéréoscopiques. Cette méthode couple l'estimation du flot optique avec la mise en correspondance stéréo dense, en minimisant une énergie construite à partir des relations entre les images des séquences stéréo. Notre algorithme gère également les discontinuités de la géométrie 3D et du champ de déplacement 3D, et est robuste aux variations d'illumination.

Notre algorithme généralise le travail de Brox *et al.* Brox et al. [2004] sur l'estimation précise du flot optique, en ajoutant des contraintes issues de la géométrie épipolaire, et nous avons montré que le même type de schéma numérique peut être utilisé pour résoudre les deux problèmes. Néanmoins, la nature de la disparité est différente de celle du flot optique, dans le sens où les occultations sont plus larges, et que la plage de valeur en disparité est comparable à la taille des objets dans les images, ce qui cause beaucoup de difficultés à de nombreux algorithmes stéréo multirésolution. Nous avons ainsi proposé un algorithme qui, dans les cas difficiles pour la stéréo, évalue tout d'abord les composantes du flot de scène par une estimation séparée puis raffine cette estimation par une estimation conjointe de toutes les composantes. Il s'agit du premier travail sur le flot de scène qui présente une évaluation quantitative de celui-ci, en comparant la composante de flot optique de notre flot de scène (en quelque sorte la projection dans les images de celui-ci) avec les résultats d'une des méthodes variationnelles d'estimation du flot optique les plus précises à notre connaissance. De plus, nos expériences ont également démontré que l'algorithme présenté est capable de traiter des séquences stéréo réelles difficiles comportant des mouvements non rigide, des discontinuités en mouvement et en stéréo, ainsi que des variations d'illumination. Nous avons également présenté des pistes pour l'amélioration de l'algorithme notamment du point de vue du temps de calcul, en adaptant des travaux récents sur les méthodes variationnelles de flot optique en temps réel, ou bien en parallélisant l'algorithme de manière à pouvoir l'adapter aux GPUs. En outre, nous pourrions remplacer les cartes d'occultations binaires par des fonctions déterministes continues. Des travaux précédents utilisent une formulation

probabiliste, à l'instar de Strecha et al. [2004b,a], mais une formulation déterministe et continue pourrait être intégrée plus naturellement à notre formulation variationnelle.

Enfin, nous avons présenté une collaboration originale entre des géophysiciens et des chercheurs en vision par ordinateur. D'une part il a fallu comprendre les besoins exprimés par les chercheurs du laboratoire Geosciences Azur, et d'autre part nous avons participé pleinement à la mise en place du dispositif expérimental. Une telle collaboration fut extrêmement enrichissante, et ce d'autant que les résultats obtenus permettent effectivement une analyse des déformations subies par les modèles en laboratoire. Il faut noter que les résultats présentés dans la thèse corroborent les hypothèses faites par les géophysiciens au sujet de la nature du mouvement de glissement.

Bibliographie

L. Alvarez, R. Deriche, J. Sanchez, and J. Weickert. Dense disparity map estimation respecting image discontinuities : a pde and scalespace based approach. Technical Report RR-3874, INRIA, January 2000. URL `citeseer.ist.psu.edu/article/alvarez00dense.html`.

L. Alvarez, R. Deriche, J. Sanchez, and J. Weickert. Dense disparity map estimation respecting image discontinuities : A pde and scale-space based approach. *JVCIR*, 13(1/2) :3–21, March 2002.

Amir A. Amini, Terry E. Weymouth, and Ramesh C. Jain. Using dynamic programming for solving variational problems in vision. *PAMI*, 12(9) :855–867, Sep 1990.

Damien Bachmann. *Modélisation physique tridimensionnelle des mouvements gravitaires de grande ampleur en milieu rocheux*. PhD thesis, Université de Sophia-Antipolis, 2007.

S. Baker, S. Roth, D. Scharstein, M.J. Black, J.P. Lewis, and R. Szeliski. A database and evaluation methodology for optical flow. In *ICCV07*, pages 1–8, 2007.

J.L. Barron, D.J. Fleet, S.S. Beauchemin, and T.A. Burkitt. Performance of optical flow techniques. In *Proc. IEEE CVPR*, pages 236–242, 1992. URL `citeseer.ist.psu.edu/barron92performance.html`.

B.Galvin, B.McCane, K.Novins, D.Mason, and S.Mills. Recovering motion fields : An evaluation of eight optical flow algorithms. In *BMVC98*, pages xx–yy, 1998.

B.Horn and B.Schunck. Determining optical flow. *Artifical Intelligence*, 17 :185–203, 1981.

Yuri Boykov and Vladimir Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. In *In Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 359–374, 2001.

Y.Y. Boykov and O. Veksler. Graph cuts in vision and graphics : Theories and applications. In *MMCV05*, pages 79–96, 2005.

William L. Briggs, Van Emden Henson, and Steve F. McCormick. *A multigrid tutorial : second edition*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000. ISBN 0-89871-462-1.

A. Brox, N. Bruhn, J. Papenberg, and T. Weickert. High accuracy optical flow estimation based on a theory for warping. In *Proc. 8th ECCV*, volume 3024 of *LNCS*, pages 25–36, Prague, Czech Republic, 2004. Springer-Verlag.

A. Bruhn, J. Weickert, C. Feddern, T. Kohlberger, and C. Schnorr. Real-time optic flow computation with variational methods. In *CAIP03*, pages 222–229, 2003.

A. Bruhn, J. Weickert, C. Feddern, T. Kohlberger, and C. Schnorr. Variational optical flow computation in real time. *IP*, 14(5) :608–615, May 2005a.

A. Bruhn, J. Weickert, T. Kohlberger, and C. Schnorr. Discontinuity preserving computation of variational optic flow in real-time. In *ScaleSpace05*, pages 279–290, 2005b.

Andrés Bruhn, Joachim Weickert, Christian Feddern, Timo Kohlberger, and Christoph Schnörr. Variational optical flow computation in real time. *IEEE Trans. Image Processing*, 14(5) :608–615, 2005c.

Andrés Bruhn, Joachim Weickert, Timo Kohlberger, and Christoph Schnörr. A multigrid platform for real-time motion computation with discontinuity-preserving variational methods. *Int. J. Comput. Vision*, 70(3) :257–277, 2006. ISSN 0920-5691. doi : http://dx.doi.org/10.1007/s11263-006-6616-7.

Alan Brunton, Chang Shu, and Gerhard Roth. Belief propagation on the gpu for stereo vision. In *CRV '06 : Proceedings of the The 3rd Canadian Conference on Computer and Robot Vision*, page 76, Washington, DC, USA, 2006. IEEE Computer Society. ISBN 0-7695-2542-3. doi : http://dx.doi.org/10.1109/CRV.2006.19.

Luc Buatois, Guillaume Caumon, and Bruno Lévy. Concurrent number cruncher : An efficient sparse linear solver on gpu. In R. Perrott et al., editor, *High Performance Computation Conference (HPCC-07), http ://www.tlc2.uh.edu/hpcc07/*, volume 4782 of *Lecture Notes in Computer Science 4782*, pages 358–371. Springer, 2007.

Frédéric Cappa. Apports de la modélisation numérique 2d à la compréhension du comportement hydro-mécanique couplé d'un grand versant rocheux instable. application au glissement de terrain de la clapière (alpes-maritimes). Master's thesis, UFR Sciences et Techniques de Franche-Comté, 2001-2002.

Rodrigo L. Carceroni and Kiriakos N. Kutulakos. Multi-view scene capture by surfel sampling : From video streams to non-rigid 3D motion, shape and reflectance. *Int. J. Comput. Vision*, 49(2-3) :175–214, 2002. ISSN 0920-5691.

Vicent Caselles, Ron Kimmel, and Guillermo Sapiro. Geodesic active contours. *Int. J. Comput. Vision*, 22(1) :61–79, 1997. ISSN 0920-5691. doi : http://dx.doi.org/10.1023/A:1007979827043.

R. Deriche and O. Faugeras. Les EDP en traitement des images et vision par ordinateur. Technical Report RR-2697, INRIA, November 1997.

Rachid Deriche, Pierre Kornprobst, and Gilles Aubert. Optical-flow estimation while preserving its discontinuities : A variational approach. In *Proc. ACCV*, pages 71–80, 1995. URL `citeseer.ist.psu.edu/deriche95opticalflow.html`.

Frédéric Devernay. *Vision stéréoscopique et propriétés différentielles.* PhD thesis, École Polytechnique, 1997.

Olivier Faugeras and Renaud Keriven. Variational principles, surface evolution, PDEs, level set methods and the stereo problem. *IEEE Transactions on Image Processing*, 7 :336–344, 1998.

Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Efficient belief propagation for early vision. *International Journal of Computer Vision*, 70(1), oct 2006.

Pau Gargallo, Emmanuel Prados, and Peter Sturm. Minimizing the reprojection error in surface reconstruction from images. In *Proceedings of the International Conference on Computer Vision, Rio de Janeiro, Brazil.* IEEE Computer Society Press, 2007. URL `http://perception.inrialpes.fr/Publications/2007/GPS07`.

Andrew V. Goldberg and Robert E. Tarjan. A new approach to the maximum-flow problem. *J. ACM*, 35(4) :921–940, 1988. ISSN 0004-5411. doi : http://doi.acm.org/10.1145/48014.61051.

Minglun Gong and Yee-Hong Yang. Disparity flow estimation using orthogonal reliability-based dynamic programming. In *Proc. 18th ICPR*, pages 70–73. IEEE, 2006. ISBN 0-7695-2521-0. doi : http://dx.doi.org/10.1109/ICPR.2006.456.

H. Grossauer and P. Thoman. Gpu-based multigrid : Real-time performance in high resolution nonlinear image processing. In *CVS08*, pages xx–yy, 2008.

D. Hoiem, A.N. Stein, A.A. Efros, and M. Hebert. Recovering occlusion boundaries from a single image. In *ICCV07*, pages 1–8, 2007.

S. Ince and J. Konrad. Occlusion-aware optical flow estimation. *IP*, 17(8) :1443–1451, August 2008.

M. Isard and J.P. MacCormick. Dense motion and disparity estimation via loopy belief propagation. In *ACCV06*, pages II :32–41, 2006.

Hiroshi Ishikawa and Davi Geiger. Occlusions, discontinuities, and epipolar lines in stereo. In *ECCV '98 : Proceedings of the 5th European Conference on Computer Vision-Volume I*, pages 232–248, London, UK, 1998. Springer-Verlag. ISBN 3-540-64569-1.

M. Kass, A.P. Witkin, and D. Terzopoulos. Snakes : Active contour models. *IJCV*, 1(4) :321–331, January 1988.

H.S. Kim and K.H. Sohn. Hierarchical disparity estimation with energy-based regularization. In *ICIP03*, pages I : 373–376, 2003.

J.J. Koenderink and A.J. van Doorn. Geometry of binocular vision and a model for stereopsis. *Biological Cybernetics*, 1976.

T. Kohlberger, C. Schnorr, A. Bruhn, and J. Weickert. Domain decomposition for parallel variational optical flow computation. In *DAGM03*, pages 196–203, 2003.

Vladimir Kolmogorov and Ramin Zabih. Multi-camera scene reconstruction via graph cuts. In *ECCV '02 : Proceedings of the 7th European Conference on Computer Vision-Part III*, pages 82–96, London, UK, 2002. Springer-Verlag. ISBN 3-540-43746-0.

Pierre Kornprobst. *Contributions à la restauration d'Images et à l'analyse de séquences : Approches Variationnelles et solutions de viscosité*. PhD thesis, Université de Nice-Sophia-Antipolis, 1998.

Martin Lefébure and Laurent D. Cohen. Image registration, optical flow and local rigidity. *J. Math. Imaging Vis.*, 14(2) :131–147, 2001. ISSN 0924-9907. doi : http://dx.doi.org/10.1023/A:1011259231755.

A. Mansouri, A. Mitiche, and J. Konrad. Selective image diffusion : Application to disparity estimation. In *ICIP-C 98*, pages 284–288, 1998.

D. Min and K. Sohn. Edge-preserving simultaneous joint motion-disparity estimation. In *ICPR06*, pages II : 74–77, 2006.

Amar Mitiche and Patrick Bouthemy. Computation and analysis of image motion : a synopsis of current problems and methods. *Int. J. Comput. Vision*, 19(1) :29–55, 1996. ISSN 0920-5691. doi : http://dx.doi.org/10.1007/BF00131147.

N. Nandhakumar and J.K. Aggarwal. On the computation of motion from sequences of images : A review. In *Univ. of Texas*, 1988.

J. Neumann and Y. Aloimonos. Spatio-temporal stereo using multi-resolution subdivision surfaces. *Stereo and Multi-Baseline Vision, IEEE Workshop on*, 0 :0103, 2001. doi : http://doi.ieeecomputersociety.org/10.1109/SMBV.2001.988768.

M. Otte and H.-H. Nagel. Optical flow estimation : advances and comparisons. In *ECCV '94 : Proceedings of the third European conference on Computer vision (vol. 1)*, pages 51–60, Secaucus, NJ, USA, 1994. Springer-Verlag New York, Inc. ISBN 3-540-57956-7.

Nicolas Papadakis, Thomas Corpetti, and Etienne Memin. Dynamically consistent optical flow estimation. In *ICCV07*, pages 1–7, 2007a.

Nicolas Papadakis, Patrick Héas, and Etienne Mémin. Memin image assimilation for motion estimation of atmospheric layers with shallow-water model. In *In 8th Asian Conference on Computer Vision (ACCV 07)*, 2007b.

Nils Papenberg, Andrés Bruhn, Thomas Brox, Stephan Didas, and Joachim Weickert. Highly accurate optic flow computation with theoretically justified warping. *Int. J. Comput. Vision*, 67(2) :141–158, 2006. ISSN 0920-5691. doi : http://dx.doi.org/10.1007/s11263-005-3960-y.

K. Pauwels and M.M. Van Hulle. Realtime phase-based optical flow on the GPU. In *CVGPU08*, pages 1–8, 2008.

J.-P. Pons, G. Hermosillo, R. Keriven, and O. Faugeras. Maintaining the point correspondence in the level set framework. *Journal of Computational Physics*, 220 :339–354, dec 2006. doi : 10.1016/j.jcp.2006.05.036.

Jean-Philippe Pons, Renaud Keriven, and Olivier Faugeras. Modelling dynamic scenes by registering multi-view image sequences. In *Proc. IEEE CVPR*, volume 2, pages 822–827, 2005. ISBN 0-7695-2372-2. doi : http://dx.doi.org/10.1109/CVPR.2005.227.

L. Robert and R. Deriche. Dense depth map reconstruction : A minimization and regularization approach which preserves discontinuities. In *ECCV96*, pages I :439–451, 1996.

Sébastien Roy and Ingemar J. Cox. A maximum-flow formulation of the n-camera stereo correspondence problem. In *ICCV '98 : Proceedings of the Sixth International Conference on Computer Vision*, page 492, Washington, DC, USA, 1998. IEEE Computer Society. ISBN 81-7319-221-9.

Daniel Scharstein and Richard Szeliski. Stereo matching with nonlinear diffusion. *Int. J. Comput. Vision*, 28(2) :155–174, 1998. ISSN 0920-5691. doi : http://dx.doi.org/10.1023/A:1008015117424.

Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3) :7–42, 2002.

J. A. Sethian. *Level Set Methods and Fast Marching Methods/ J. A. Sethian.* Cambridge University Press, Cambridge, 1999. ISBN 0-521-64557-3. Includes bibliographical references and index.

S.N. Sinha, P. Mordohai, and M. Pollefeys. Multi-view stereo via graph cuts on the dual of an adaptive tetrahedral mesh. In *ICCV07*, pages 1–8, 2007.

N. Slesareva, A. Bruhn, and J. Weickert. Optic flow goes stereo : A variational method for estimating discontinuity-preserving dense disparity maps. In *DAGM05*, page 33, 2005.

Christoph Stiller, Janusz Konrad, and Robert Bosch. Estimating motion in image sequences : A tutorial on modeling and computation of 2d motion. *IEEE Signal Processing Magazine*, 16 :70–91, 1999.

C. Strecha, R. Fransens, and L.J. Van Gool. Wide-baseline stereo from multiple views : A probabilistic account. In *Proc. IEEE CVPR*, volume 1, pages 552–559, 2004a.

Christoph Strecha, Tinne Tuytelaars, and Luc Van Gool. Dense matching of multiple wide-baseline views. In *ICCV '03 : Proceedings of the Ninth IEEE International Conference on Computer Vision*, page 1194, Washington, DC, USA, 2003. IEEE Computer Society. ISBN 0-7695-1950-4.

Christoph Strecha, Rik Fransens, and Luc J. Van Gool. A probabilistic approach to large displacement optical flow and occlusion detection. In *ECCV Workshop SMVP*, pages 71–82, 2004b.

Jian Sun, Nan-Ning Zheng, and Heung-Yeung Shum. Stereo matching using belief propagation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(7) :787–800, 2003. ISSN 0162-8828. doi : http://dx.doi.org/10.1109/TPAMI.2003.1206509.

S.Vedula and S.Baker. Three-dimensional scene flow. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(3) :475–480, 2005. ISSN 0162-8828. doi : http://dx.doi.org/10.1109/TPAMI.2005.63.

S.Vedula, S.Baker, P.Rander, R.Collins, and T.Kanade. Three-dimensional scene flow. In *Proc. IEEE ICCV*, pages 722–729, 1999. URL `citeseer.ist.psu.edu/vedula99threedimensional.html`.

D. Terzopoulos. Regularization of inverse visual problems involving discontinuities. *IEEE Trans. PAMI*, 1986.

A. Tikhonov and V. Arsenine. *Méthodes de résolution de problèmes mal posés.* Editions de Moscou, 1977.

V. Vineet and P.J. Narayanan. Cuda cuts : Fast graph cuts on the gpu. In *CVGPU08*, pages 1–8, 2008.

A. Wedel, C. Rabe, T. Vaudrey, T. Brox, U. Franke, and D. Cremers. Efficient dense scene flow from sparse or dense stereo data. In *ECCV08*, pages I : 739–751, 2008.

R.P. Wildes. Direct recovery of three-dimensional scene geometry from binocular stereo disparity. *T-PAMI*, 13 :761–774, 1991.

Oliver Williams, Michael Isard, and John MacCormick. Estimating disparity and occlusions in stereo video sequences. In *CVPR '05 : Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, pages 250–257, Washington, DC, USA, 2005. IEEE Computer Society. ISBN 0-7695-2372-2. doi : http://dx.doi.org/10.1109/CVPR.2005.146.

Y. Zhang and C. Kambhamettu. Integrated 3D scene flow and structure recovery from multiview image sequences. In *Proc. IEEE CVPR*, pages II : 674–681, 2000.

Annexe A

Calibration, rectification, stéréoscopie et reconstruction 3D

La calibration d'une caméra consiste à trouver analytiquement la fonction de projection qui à un point de $\Re^3$ associe sa projection dans l'image (qui se forme dans le plan rétinien de la caméra). Cette fonction de projection dépend du modèle de caméra choisi. On fait la distinction entre deux types de calibrage :

- le calibrage fort
- le calibrage faible

Ces types de calibrage sont explicités brièvement dans les 2 sous sections suivantes.

1 Calibrage fort

Pour estimer la fonction de projection, on utilise une mire. Sur celle ci sont marqués des points dont on connaît la position 3D. Connaissant les positions dans l'image de ces mêmes points, on détermine via une méthode d'optimisation les paramètres de la caméra réalisant la meilleure correspondance entre les points 3D et leurs projections. On regroupe les modèles de caméras de la manière suivante :

- le modèle projectif linéaire
- le modèle avec distorsion non linéaire : on ajoute les paramètres de distorsion non linéaires au modèle projectif linéaire

La figure A.1 donne une description du modèle projectif linéaire

La fonction de projection est une composée de deux transformations :

- une rotation/translation du repère choisi dans $\Re^3$ vers le repère attaché à la caméra. Cette transformation est décrite par la matrice des paramètres extrinsèques de la caméra.
- la transformation des coordonnées d'un point dans le repère de la caméra vers les coordonnées de la projection du point dans l'image. Cette transformation est décrite par la matrice des paramètres intrinsèques de la caméra.

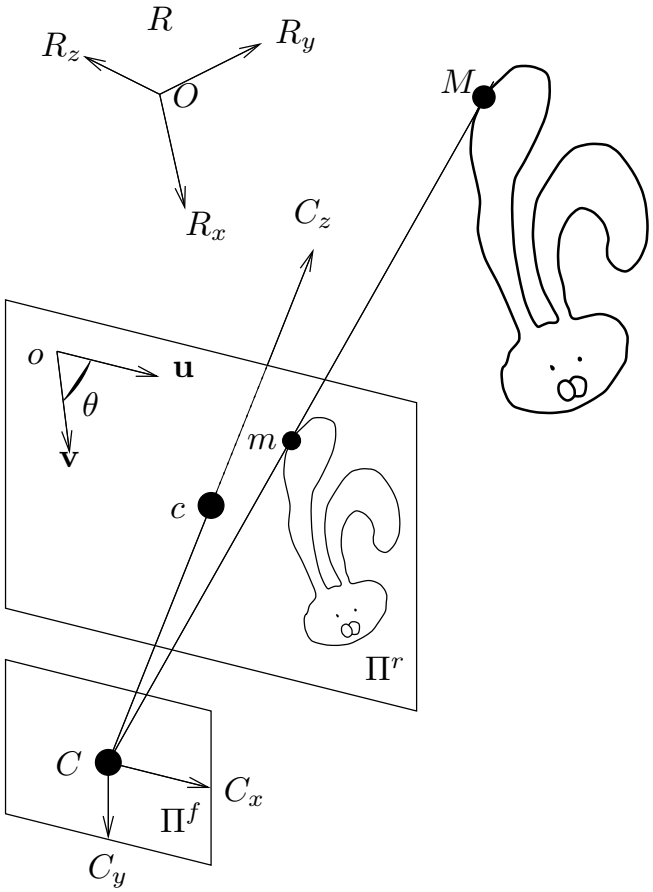

FIGURE A.1 – Illustration issue du travail de **??** représentant le modèle de caméra projectif linéaire.

Les paramètres intrinsèques de la caméra comportent dans le cas du modèle projectif linéaire les paramètres suivants :

- la distance focale
- les coordonnées du point principal (c sur la figure 2.1)
- l'angle θ entre les axes de coordonnées du repère attaché à l'image

auxquels se rajoutent les paramètres de distorsion non linéaire dans le cas où le modèle projectif linéaire est non Dans le projet Geolstereo, les distorsions liées aux appareils photos numériques utilisés pour obtenir les paires d'images sont prises en compte. Les évaluations des paramètres extrinsèques, intrinsèques et de distorsions sont réalisées au laboratoire Géosciences Azur avec l'outil "Matlab Calibration Toolbox disponible à l'adresse `http://www.vision.caltech.edu/bouguetj/calib_doc/`.

Expression analytique de la projection et des distorsions

Soient, dans le cas de l'utilisation d'un modèle projectif linéaire :
- $R = (O, R_x, R_y, R_z)$ un repère de $\Re^3$.
- R_c un repère attaché à la caméra
- M un point de l'espace
- M_R ses coordonnées dans R
- M_C ses coordonnées dans R_c
- K la matrice de rotation, t le vecteur de translation
- θ l'angle entre les axes du repère attaché au plan image
- (p_0, p_1) les coordonnées du point principal dans le plan image, et (α_0, α_1) une fonction linéaire de la distance focale.

On a comme expliqué précédemment, en coordonnées homogènes (utilisées en géométrie projective) :

$M_C = \begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = KM_R + t$ et la projection sur le plan image a pour coordonnées dans R_C : $\begin{pmatrix} f\frac{X_c}{Z_c} \\ f\frac{Y_c}{Z_c} \\ f \end{pmatrix}$.

On passe ensuite en coordonnées dans le repère image (coordonnées pixels) :

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \alpha_0 & -\alpha_0 cot(\theta) \\ 0 & \alpha_1 sin(\theta) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} p_0 \\ p_1 \end{pmatrix} \tag{A.1}$$

Finalement, soit $M = r \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$ et son projeté $m = s \begin{pmatrix} u \\ v \\ 1 \end{pmatrix}$ dans le plan image vérifie la relation :$m = \beta PM$ avec P notre matrice 3×4 de projection telle que

$$P = \begin{pmatrix} \alpha_0 & -\alpha_0 cot(\theta) & p_0 \\ 0 & \alpha_2 sin(\theta) & p_1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} K & t \\ \mathbf{0} & 1 \end{pmatrix} \tag{A.2}$$

À celà on doit rajouter des distorsions qui sont de plusieurs types. Pour formuler celà analytiquement, on pose :

$$x_d = x + \delta_x(x, y) \tag{A.3}$$
$$y_d = y + \delta_y(x, y) \tag{A.4}$$

les coordonnées mesurées de la projection sur le plan image, où (x, y) sont les coordonnées dans le cadre du modèle projectif linéaire. La fonction de correction δ est

somme de plusieurs termes correctifs provenant de termes de distorsions d'origines différentes. Ceux-ci sont détaillés dans Devernay [1997]. La distorsion totale s'écrit :

$$\delta(x,y) = \begin{pmatrix} k_1 x(x^2+y^2) + p_1(3x^2+y^2) + 2p_2 xy + s_1(x^2+y^2) \\ k_1 y(x^2+y^2) + p_2(x^2+3y^2) + 2p_1 xy + s_2(x^2+y^2) \end{pmatrix} \tag{A.5}$$

Les k_i et les p_i sont les paramètres de distorsion. On applique ensuite la transformation affine pour passer en coordonnées pixel comme réalisé précédemment et on obtient les coordonnées de la projection tenant compte des distorsions.

2 Calibration faible

La calibration faible consiste à calibrer deux ou plusieurs caméras sans aucune connaissance à priori sur les caractéristiques de celles-ci ou sur la scène observée. On peut montrer quà partir de correspondances de points entre les images des deux caméras, on peut calibrer la géométrie épipolaire du système. Dans le cas où on ne connaît pas les caractéristiques des caméras, la géométrie épipolaire est décrite par la matrice fondamentale. Cette matrice contient les informations nécessaires à la reconstruction de la structure de la scène à une transformation projective de l'espace près.

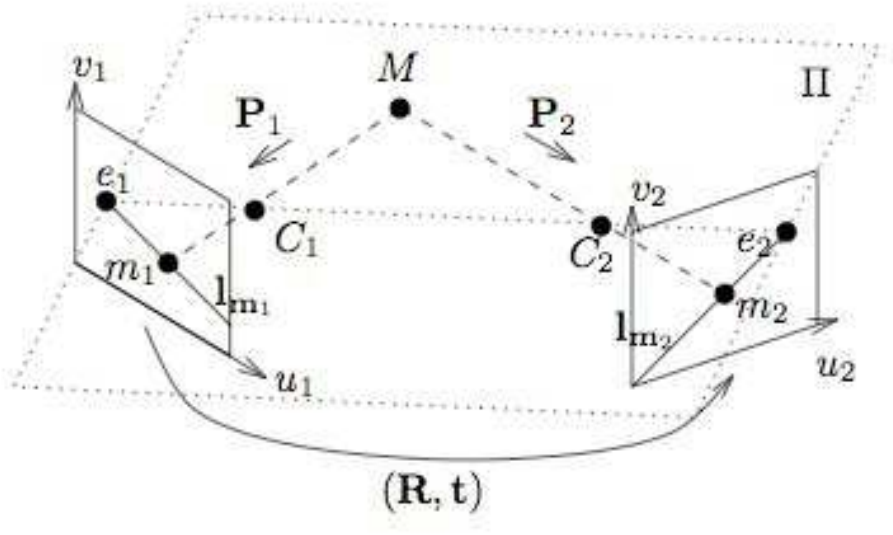

FIGURE A.2 – Géométrie épipolaire d'un système de deux caméras.C_1 et C_2 désignent les centres optiques des caméras, M un point de la scène 3D, m_1 et m_2 ses projetés dans les plans images de chaque caméra, e_1 et e_2 les épipoles respectifs associés à chaque caméra. Enfin, les droites épipolaires correspondant à m_1 et m_2 sont indiquées.

Comme illustré sur la figure A.2, nous disposons d'un système de deux caméras, observant une scène 3D donnée. Nous faisons l'hypothèse que les caméras sont disposées de manière à ce qu'une majorité de points de la scène soient visibles dans les deux caméras. Cela nécessite une orientation appropriée ainsi qu'une baseline raisonnable. Supposons qu'un point M de la scène soit visible dans les deux caméras. Celui-ci se projette dans les plans images des deux caméras, chaque projeté correspondant à un pixel m_1 et m_2 dans les images gauche I_g et droite I_d. Notons C_l1 et C_2 les centres optiques des caméras gauche et droite, respectivement. Suppons que l'on connaisse m_1 et que l'on cherche le pixel m_1 correspondant au même point 3D M. Il se trouve que par construction, ce pixel se situe obligatoirement sur la droite intersection du plan image de la caméra droite et du plan formé par C_1, C_2 et m_1. Cette droite est nommée droite épipolaire. A priori, la recherche du correspondant de m_g est donc une recherche le long de cette droite. Celle-ci étant sans particularité à priori, cette recherche est bidimensionnelle.

Concernant la figure A.2, la droite épipolaire $\mathbf{l}_{\mathbf{m}_2}$ est également l'intersection du plan Π qui passe par m_1, C_1, et C_2, appelé *plan épipolaire*, avec le plan rétinien de la deuxième caméra. On remarque ainsi que toutes les droites épipolaires passent dans la deuxième image par le point e_2, appelé *épipôle*, intersection de la droite (C_1C_2) avec le plan rétinien, puisque la droite (C_1C_2) est nécessairement incluse dans tous les plans épipolaires. De plus, la géométrie épipolaire est symétrique : étant donné un point m_2 dans la seconde image, son correspondant m_1 se trouve nécessairement sur la droite épipolaire $\mathbf{l}_{\mathbf{m}_1}$, et toutes les droites épipolaires de la première images passent par l'épipôle e_1, intersection de (C_1C_2) avec le plan image de la première caméra. Les points (m_1, m_2, C_1, C_2) sont donc dans un même plan, c'est la contrainte de coplanarité, utilisée notamment pour calculer le mouvement et la structure 3-D lorsque les paramètres intrinsèques sont connus. La matrice fondamentale F_{12} est l'expression algébrique de cette contrainte de coplanarité, et vérifie la propriété :

$$m_1^T F_{12} m_2 = 0. \tag{A.6}$$

Cette matrice exprime les caractéristiques de la géométrie épipolaire du système stéréoscopique lorsque les paramètres intrinsèques des caméras sont inconnus.

3 Estimation numérique des paramètres des caméras

Dans notre cas nous voulions effectuer une reconstruction 3D des scènes observées dans le cas des expériences du chapitre 4. Afin de calibrer les caméras, nous avons pris des images stéréo d'une mire avec le système décrit par l'image A.3.

Nous avons utilisé le Matlab Calibration ToolBox, plug-in logiciel développé pour Matlab et disponible à l'adresse `http://www.vision.caltech.edu/bouguetj/calib_doc/`. Les étapes de calibration du système de caméras à partir des images de la mire, à l'aide de ce logiciel, sont les suivantes :

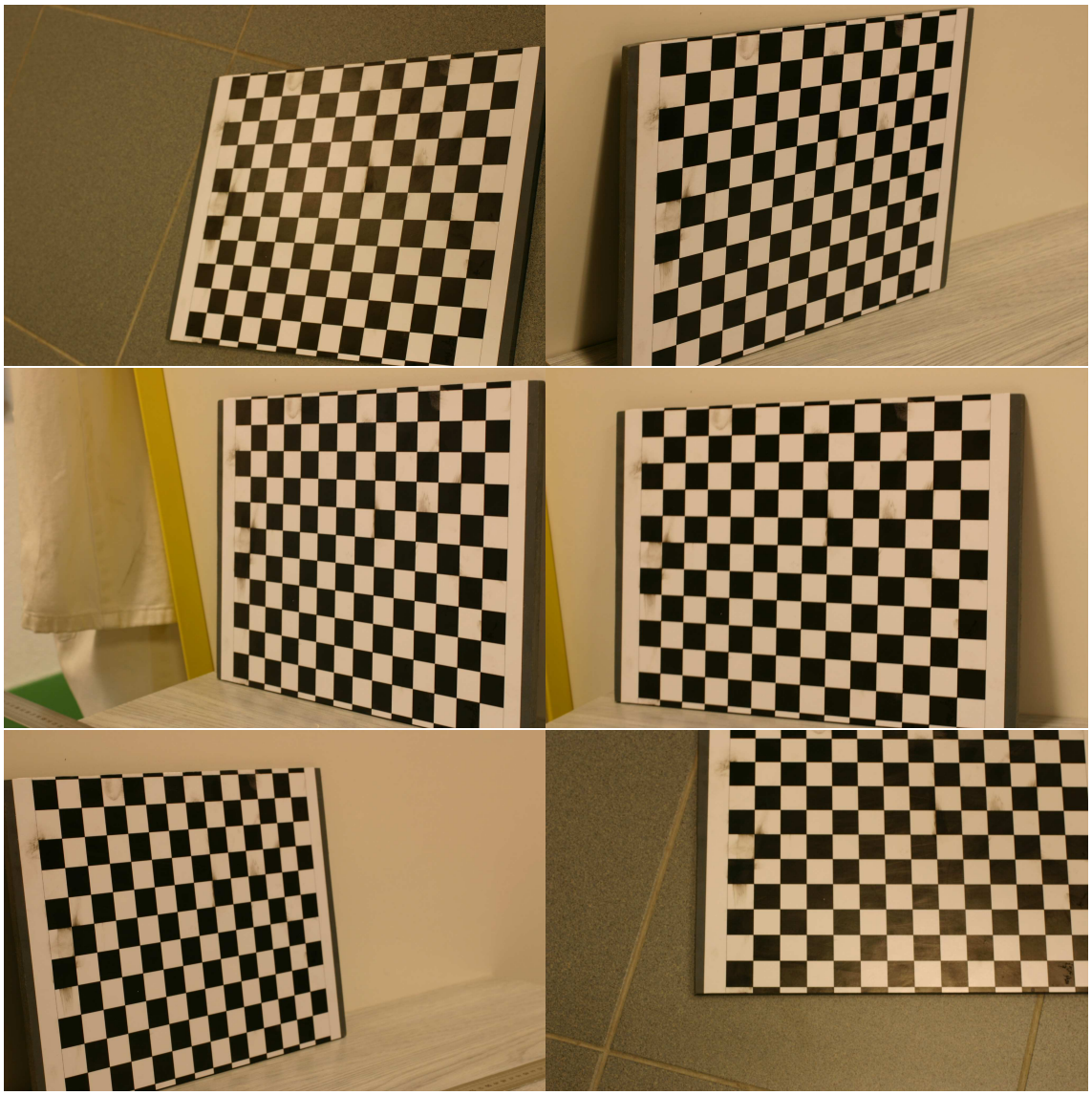

FIGURE A.3 – Quelques images de la mire utilisée pour calibrer le système stéréoscopique utilisé dans les expériences décrites dans le chapitre 4.

- Chargement des images de la mire (figure A.4)
- L'utilisateur choisit un pattern rectangulaire afin de délimiter une zone d'extractions de points d'intérêt, en l'occurence les coins du damier (figure A.5).
- Extraction des points d'intérêt du damier (figure A.6)
- Minimisation aux moindres carrés pour obtenir une estimation des paramètres intrinsèques et extrinsèques (figure A.7)

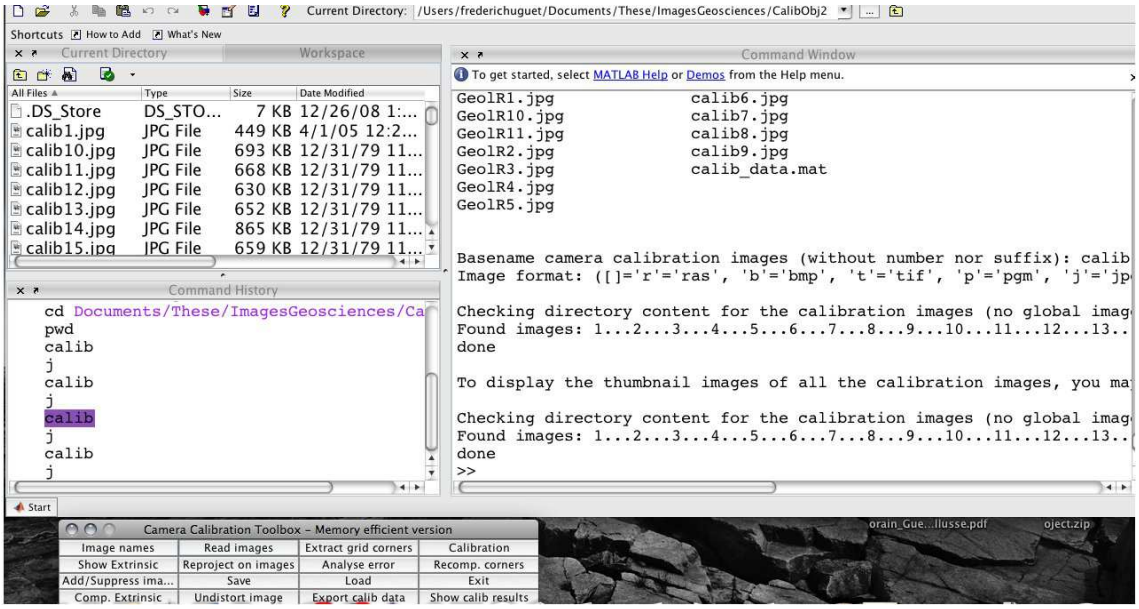

FIGURE A.4 – Étape 1 : chargements des images de la mire

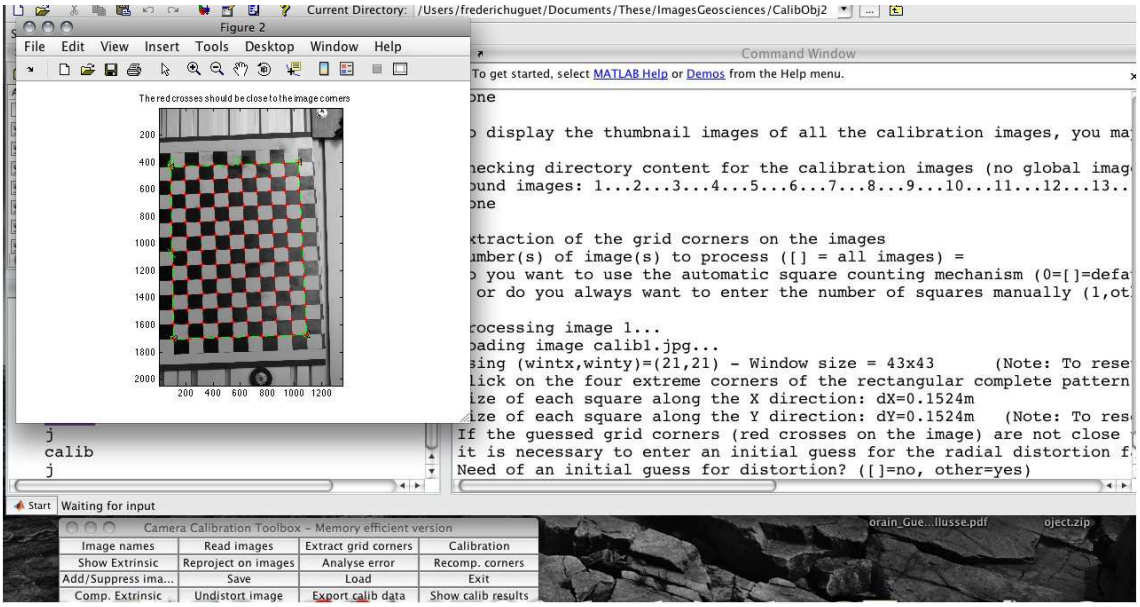

FIGURE A.5 – Étape 2 : choix des coins de base

4 Rectification

4.1 Principes

Afin de simplifier le problème de mise en correspondance stéréo, on effectue une transformation homographique. Comme illustré en figure A.8, il s'agit de projeter

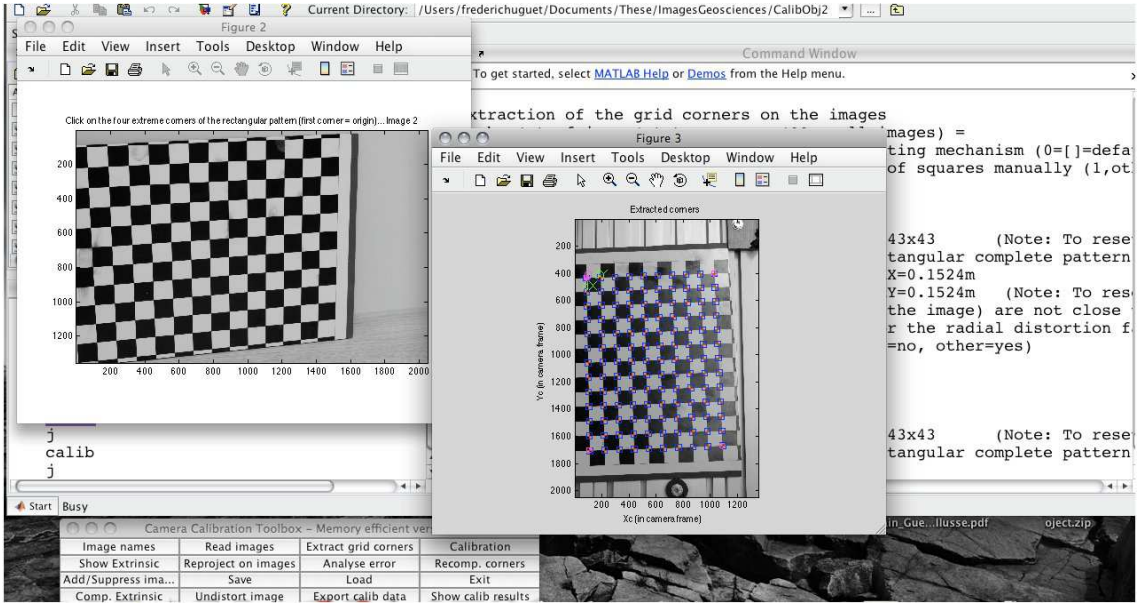

FIGURE A.6 – Étape 3 : extraction de coins

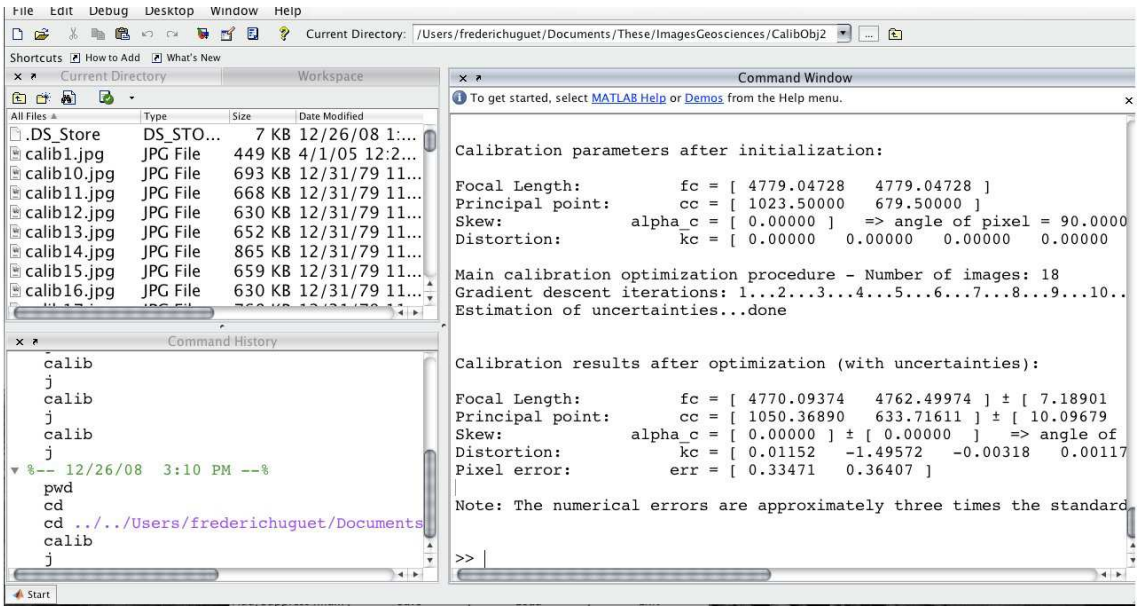

FIGURE A.7 – Étape 4 : minimisation aux moindres carrés et obtention des paramètres

les plans images des caméras sur un plan parallèle à la droite joignant les centres optiques de ces dernières. Une fois cette transformation accomplie, les droites épipolaires sont horizontales, ce qui signifie que les pixels m_g et m_d ont la meme ordonnée. Ainsi, la mise en correspondance stéréoscopique se réduit à une recherche monodimensionnelle le long d'une droite horizontale.

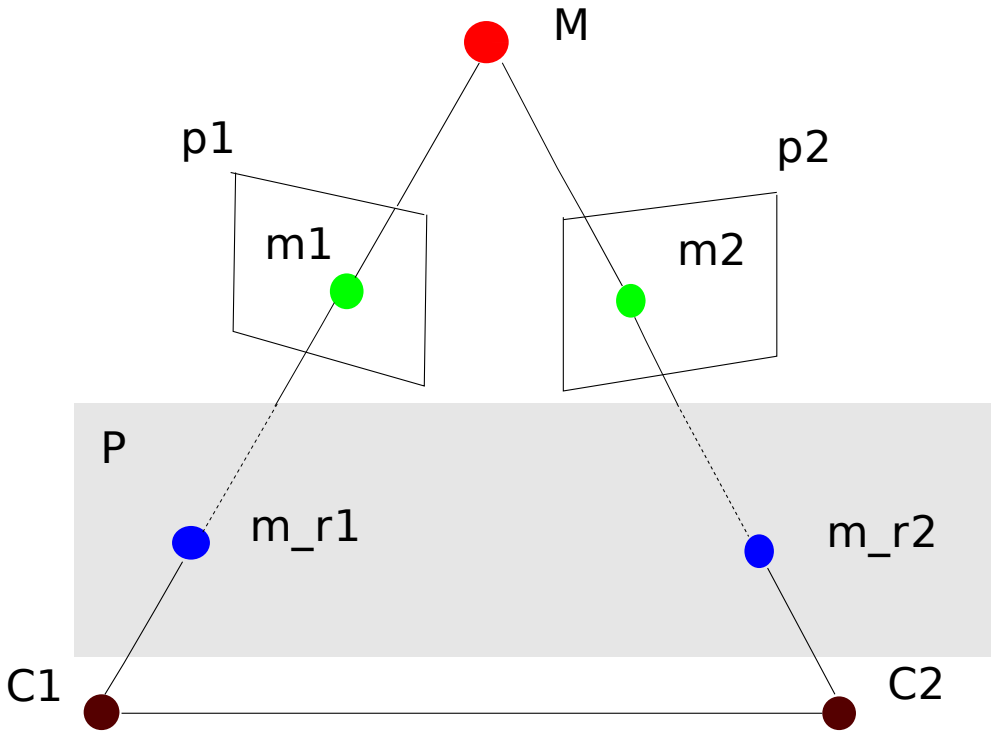

FIGURE A.8 – La rectification consiste à reprojeter les images des plans rétinaux Π^r et $\Pi^{r'}$ sur un même plan Π, dit « plan de rectification », parallèle à la droite (CC') joignant les deux centres optiques. La reprojection revient à appliquer une homographie à chacune des images.

4.2 Algorithme

L'algorithme de rectification utilisé dans la thèse étant utilisé par une société nous nous contentons de décrire brièvement les grandes étapes de la méthode :

- Pour une paire d'images données, découpage des images en secteurs
- Pour chaque secteur, faire une extraction de points disposés selon une loi probabiliste
- Mettre en correspondance les points pour chaque secteur en utilisant des estimateurs statistiques robustes (ou encore M-estimators)
- En deduire une paire de matrices de rectification par une méthode d'optimisation numérique (moindres carrés ...)

4.3 Résultats

Quelques résultats sur les images issues du système stéréoscopique de Geolstereo sont illustrés en figure A.9.

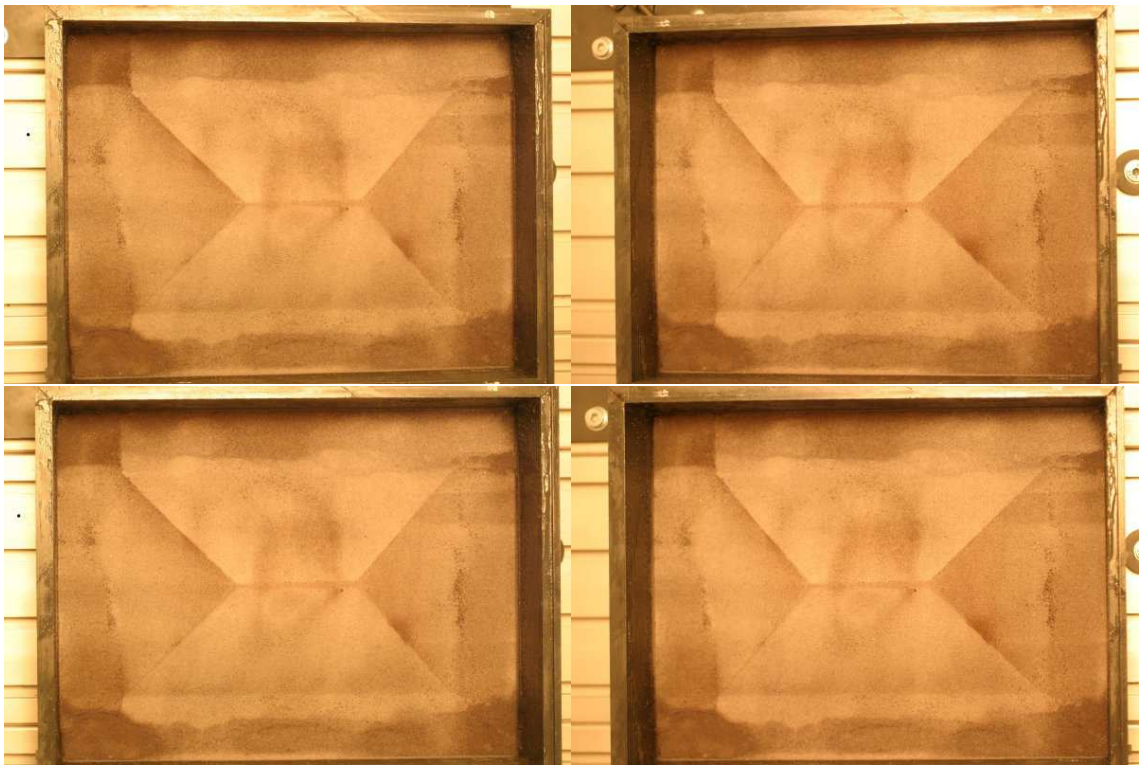

FIGURE A.9 – En haut : paire stéréo brute. En bas : la meme paire rectifiée.

5 Stéréoscopie : utilisation de la disparité

La disparité est, comme nous l'avons vu dans les chapitres centraux de la thèse, utilisée pour estimer la reconstruction 3D d'une scène observée par un système stéréoscopique fortement calibré, mais il existe aussi un lien entre la disparité et les propriétés différentielles des surfaces reconstruites.

5.1 Reconstruction 3D

La reconstruction 3D consiste à calculer la géométrie de la scène observée par le système de vision, à partir des résultats des algorithmes de mise en correspondance stéréoscopique. Dans le cas qui nous intéresse, nous disposons d'une carte de disparité et nous devons donc pour chaque paire de points mis en correspondance entre les deux images stéréoscopiques calculer la position dans l'espace du point 3-D correspondant.

Considérons les matrices de projection correspondant aux caméras après l'étape de rectification, $P = \begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix}$ et $Q = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix}$. Après rectification, les projetés d'un même point ont même ordonné, c'est pourquoi les matrices ne diffèrent que par leur première ligne.

On démontre que l'on peut directement retrouver la position 3D du point cherché à partir des coordonnées pixels de son projeté dans une image et de la disparité en ce projeté. Notant $R = \begin{pmatrix} p_1 \\ p_2 \\ q_1 - p_1 \\ p_3 \end{pmatrix}^{-1}$, que l'on nomme matrice de reconstruction, on obtient après quelques calculs de projection :

$$M = \begin{pmatrix} s.x \\ s.y \\ s.z \\ s \end{pmatrix} = R \begin{pmatrix} u \\ v \\ d(u,v) \\ 1 \end{pmatrix}. \tag{A.7}$$

Cette dernière formule, établissant un lien direct et simplement matriciel entre une carte de disparité et la reconstruction 3D correspondante, montre que la disparité permet de reconstruire la scène 3D. La carte de disparité est donc une reconstruction projective de la scène.

5.2 Propriétés différentielles des surfaces et disparité

Un lien entre les dérivées de la disparité par rapport aux coordonnées pixel et les propriétés différentielles de la surface à reconstruire a été établi par Koenderink and van Doorn [1976] puis par Wildes [1991]. Les dérivées de la disparité sont liées, à l'ordre 1, au plan tangent à la surface, et à l'ordre 2, à la courbure de celle-ci. Il existe plusieurs méthodes d'évaluation de ces dérivées :

- Méthodes de calcul par ajustage de surface, où l'on minimise des fonctions splines.
- Minimisation non convexe avec discontinuités autorisées à l'intérieur d'une surface lisse.
- Dérivation point à point des données reconstruites
- Méthodes bayésiennes et probabilistes : calcul de la distribution des dérivées de la disparité dans le cas oú la distribution des orientations de surface est aléatoire.
- Calcul par régression bilinéaire sur les données situées dans une fenêtre centrée autour du pixel courant. Les dérivées premières sont alors les coefficients du plan issu de l'approximation des données aux moindres carrés. Cette approche permet de prendre en compte l'absence d'informations lorsque la disparité et ses dérivées ne sont pas définies.

Devernay [1997] présente une méthode de corrélation fine pour l'estimation des dérivées de la disparité. Pour exprimer l'équation du plan tangent à une surface donnée, en un point M de coordonnées $((x, y, z, 1))$, il suffit d'après les travaux cités de trouver l'équation du plan tangent au point de la carte de disparité correspondant puis d'utiliser la reconstruction projective que constitue la carte de disparité.

En un point d'une carte de disparité $(x_0, y_0, d(x_0, y_0)$, l'équation du plan tangent s'écrit sous la forme $z = d(x_0, y_0) + (x - x_0)\frac{\partial d}{\partial x}(x_0, y_0) + (y - x_0)\frac{\partial d}{\partial y}(x_0, y_0)$.

Ce qui s'écrit

$$(a, b, c, d)^T \cdot (x, y, z, 1) = 0 \tag{A.8}$$

avec

$$a = \frac{\partial d}{\partial x}(x_0, y_0) \tag{A.9}$$

$$b = \frac{\partial d}{\partial y}(x_0, y_0) \tag{A.10}$$

$$c = -1 \tag{A.11}$$

$$d = d(x_0, y_0) - x_0 a - y_0 b \tag{A.12}$$

$p_{\text{disparite}} = (a, b, c, d)$ sont les coordonnées du plan tangent à la carte de disparité.

R étant inversible, on obtient les coordonnées du plan tangent à la surface en M :

$$p = R^{-T} p_{\text{disparite}}. \tag{A.13}$$

Les dérivées secondes de la disparité, quand elles existent, sont liées à la courbure de la surface. Pour plus de précisions à ce sujet on pourra se reporter au travail de Devernay [1997].

Annexe B

Développements mathématiques

Cette annexe est consacrée aux développements mathématiques associés aux EDP à résoudre pour estimer le flot de scène. Nous détaillons ici notamment la discrétisation des équations ainsi

1 Discrétisation des équations d'Euler-Lagrange

On considère une discrétisation de notre domaine spatial Ω avec h comme pas en u et en v. On notera $\triangle t$ le pas de temps à choisir. Pour le moment nous restons le plus général possible concernant les valeurs choisies pour ces pas, même si en règle générale, le pas vaut la largeur d'un pixel.

Calcul pratique des dérivées des images

Pour calculer les dérivées spatiales des images $\frac{\partial I}{\partial x}$ et $\frac{\partial I}{\partial y}$ en un point quelconque on pourra simplement utiliser, à chaque itération en temps, une convolution par un filtre de dérivation c'est-à-dire : $\frac{\partial I}{\partial x} = \frac{1}{2}\begin{pmatrix}-1 & 0 & 1\end{pmatrix} * I$ et $\frac{\partial I}{\partial y} = \frac{1}{2}\begin{pmatrix}-1\\0\\1\end{pmatrix} * I$.

À chaque itération en temps de notre algorithme, on va donc devoir appliquer le filtre de dérivation à nos deux images d'une même paire stéréo.

Discrétisation du terme de divergence

Le second problème lié à la discrétisation des EDP consiste à choisir une discrétisation de notre terme de divergence qui soit assez robuste. Parmi les différents schémas aux différences finies actuellement usités , plusieurs pourraient convenir **?**. Nous présentons ici brièvement les schémas possibles et leurs défauts du point de vue de la prise en compte de l'information.

On notera δ_x et δ_y les schémas aux différences finies centrés approximant respectivement $\frac{\partial}{\partial x}$ et $\frac{\partial}{\partial y}$. On a donc en un noeud $N_{i,j}$ du maillage de Ω, pour une fonction

$\Phi(u, v)$:

$$\delta_x(\Phi_{i,j}) = \frac{\Phi_{i+1,j} - \Phi_{i-1,j}}{2h} \tag{B.1}$$

$$\text{et } \delta_y(\Phi_{i,j}) = \frac{\Phi_{i,j+1} - \Phi_{i,j-1}}{2h} \tag{B.2}$$

On sait que l'on a :

$$\text{div}(b\nabla\Phi) = \frac{\partial}{\partial x}(b\frac{\partial\Phi}{\partial x}) + \frac{\partial}{\partial y}(b\frac{\partial\Phi}{\partial y}) \tag{B.3}$$

qui nous amène a essayer plusieurs combinaisons des opérateurs δ_x et δ_y.

En combinant de la manière la plus simple les opérateurs, on obtient

$$\text{div}(b\nabla\Phi)|_{i,j} \approx \delta_x(b_{i,j}\delta_x(\Phi_{i,j})) + \delta_y(b_{i,j}\delta_y(\Phi_{i,j})) \tag{B.4}$$

En développant, il vient finalement :

$$\begin{aligned}\text{div}(b\nabla\Phi)|_{i,j} \approx \frac{1}{4h^2}(&b_{i+1,j}\Phi_{i+2,j} + b_{i-1,j}\Phi_{i-2,j} + b_{i,j+1}\Phi_{i,j+2} + b_{i,j-1}\Phi_{i,j-2} \\ &- (b_{i+1,j} + b_{i-1,j} + b_{i,j+1} + b_{i,j-1})\Phi_{i,j})\end{aligned}$$

On constate dans cette formulation, que concernant notre inconnue Φ, n'interviennent pas les points immédiatement voisins du noeud courant $N_{i,j}$, soit les points $N_{i-1,j}$, $N_{i+1,j}$, $N_{i,j-1}$ et $N_{i,j+1}$. Cela pose problème au voisinage des discontinuités, où une forte variation locale de Φ associée au manque d'informations dû à l'absence de ces points peut entraîner bien sûr un biais dans notre terme de divergence.

Pour faire apparaître les points du voisinage manquants, on pense à combiner des opérateurs aux différences finies du premier ordre non centrés, c'est-à-dire :

$$\delta_x^+ = \frac{\partial\Phi}{\partial x}|_{i,j} = \frac{\Phi_{i+1,j} - \Phi_{i,j}}{h} \tag{B.5}$$

$$\delta_x^- = \frac{\partial\Phi}{\partial x}|_{i,j} = \frac{\Phi_{i,j} - \Phi_{i-1,j}}{h} \tag{B.6}$$

et en notant similairement δ_y^+ et δ_y^- les approximations de $\frac{\partial\Phi}{\partial x}|_{i,j}$ il vient :

$$\text{div}(b\nabla\Phi)|_{i,j} \approx \delta_x^+(b_{i,j}\delta_x^-(\Phi_{i,j})) + \delta_y^+(b_{i,j}\delta_y^-(\Phi_{i,j})),$$

soit finalement :

$$\begin{aligned}\text{div}(b\nabla\Phi)|_{i,j} \approx \frac{1}{h^2}(&b_{i+1,j}\Phi_{i+1,j} + b_{i,j}\Phi_{i-1,j} + b_{i,j+1}\Phi_{i,j+1} + b_{i,j}\Phi_{i,j-1} \\ &- (b_{i+1,j} + b_{i,j+1} + 2b_{i,j})\Phi_{i,j}).\end{aligned} \tag{B.7}$$

Dans ce cas, concernant les noeuds manquants dans l'approximation de Φ, le problème a disparu. Cependant, $N_{i-1,j}$ et $N_{i,j-1}$ n'interviennent pas dans l'utilisation de b dans le schéma. Pour avoir un schéma robuste parfaitement symétrique dans

l'utilisation du maillage on utilise, et c'est la méthode que nous choisissons pour notre système, un schéma aux différences finies centré plus fin, faisant intervenir des points intermédiaires calculés par interpolation : $N_{i+\frac{1}{2},j}$, $N_{i-\frac{1}{2},j}$, $N_{i,j+\frac{1}{2}}$ et $N_{i,j-\frac{1}{2}}$. En effet, notant les opérateurs centrés faisant intervenir ces points : $\delta_x^{+\frac{1}{2}}$, $\delta_x^{-\frac{1}{2}}$, $\delta_y^{+\frac{1}{2}}$, $\delta_y^{-\frac{1}{2}}$ on obtient :

$$\begin{aligned} div(b\nabla\Phi)|_{i,j} \approx \frac{1}{h^2}(&b_{i+\frac{1}{2},j}\Phi_{i+1,j} + b_{i-\frac{1}{2},j}\Phi_{i-1,j} + b_{i,j+\frac{1}{2}}\Phi_{i,j+1} + b_{i,j-\frac{1}{2}}\Phi_{i,j-1} \\ &- (b_{i+\frac{1}{2},j} + b_{i-\frac{1}{2},j} + b_{i,j+\frac{1}{2}} + b_{i,j-\frac{1}{2}})\Phi_{i,j}) \end{aligned} \tag{B.8}$$

Malgré l'inconvénient de réaliser une interpolation pour calculer les points intermédiaires, le schéma présenté fait maintenant intervenir de manière symétrique tous les points au voisinage le plus proche de $N_{i,j}$. D'autres possibilités pourraient être testées, comme la prise en compte des voisins situés en diagonale de $N_{i,j}$, et celà pourra être envisagé dans l'avenir dans le cadre de l'optimisation plus approfondie du schéma. Nous allons dans un premier temps expliciter un premier schéma implémentable, le coder et le tester numériquement, et par la suite nous essaierons d'améliorer le conditionnement du problème. Notre terme d'attache aux données n'étant pas convexe en l'inconnue recherchée, on ne peut prouver l'unicité d'un minimum à priori. Par un prétraitement sur les images (filtrage gaussien ...) et l'utilisation d'une analyse multi-échelles on va essayer de convexifier le problème. Les approches de résolution numérique multi-grilles pour la reconstruction de flot optique sont quant à elles décrites dans Bruhn et al. [2005b]. .

1.1 Linéarisation des équations d'Euler Lagrange

Le système étant non convexe, la solution nécessite d'être correctement initialisée, en particulier concernant la disparité initiale d^0. On peut envisager l'écriture d'un modèle dans lequel cette disparité initiale serait considŕée comme une variable. Bien initialisé avec un algorithme de stéréo existant, son estimation serait affinée au fur et à mesure des estimations ce qui rendrait la résolution des systèmes moins sensibles a une erreur faite eventuellement au départ pour une d^0 qui serait fixée une bonne fois pour toutes. De plus, celà permet d'exploiter tous les liens entre les images et donc le maximum d'informations. Ce modèle s'ecrit de la maniere suivante : $J(u, v, d, d^0) = J_1 + J_2 + J_3 + J_4 + \omega J_{\text{reg}}$

avec

- Contrainte de flot optique gauche
- $J_1 = \int_\Omega \Psi((I_L^t(x+u, y+v) - I_L^0(x, y))^2 + \beta_1(\nabla I_L^t(x+u, y+v) - \nabla I_L^0(x, y))^2)$
- Contrainte de flot optique droit
- $J_2 = \int_\Omega \Psi((I_R^t(x+u+d', y+v) - I_R^0(x+d, y))^2 + \beta_1(\nabla I_R^t(x+u+d', y+v) - \nabla I_R^0(x+d, y))^2)$
- Contrainte stereoscopique temporelle
- $J_3 = \int_\Omega \Psi((I_R^t(x+u+d', y+v) - I_L^t(x+u, y+v))^2 + \beta_1(\nabla I_R^t(x+u+d', y+v) - \nabla I_L^t(x+u, y+v))^2)$

- Contrainte stereoscopique initiale
- $J_4 = \int_\Omega \Psi((I_R^0(x+d,y) - I_L^0(x,y))^2 + \beta_4(\nabla I_R^0(x+d,y) - \nabla I_L^0(x,y))^2)$
- Contrainte de regularisation de la solution
- $J_{\text{reg}} = \int_\Omega \Psi(|\nabla u|^2 + |\nabla v|^2 + \lambda|\nabla(d^{'} - d)|^2 + \mu|\nabla d|^2)$

Adoptons les notations suivantes :

- $\Psi^{'}_{l_f low} = \Psi^{'}((I_L^t(x+u,y+v) - I_L^0(x,y))^2 + \beta_1(\nabla I_L^t(x+u,y+v) - \nabla I_L^0(x,y))^2)$
- $\Psi^{'}_{r_f low} = \Psi^{'}((I_R^t(x+u+d^{'},y+v) - I_R^0(x+d,y))^2 + \beta_1(\nabla I_R^t(x+u+d^{'},y+v) - \nabla I_R^0(x+d,y))^2)$
- $\Psi^{'}_{stereo} = \Psi^{'}((I_R^t(x+u+d^{'},y+v) - I_L^t(x+u,y+v))^2 + \beta_1(\nabla I_R^t(x+u+d^{'},y+v) - \nabla I_L^t(x+u,y+v))^2)$
- $\Psi^{'}_{stereo-init} = \Psi^{'}((I_R^0(x+d,y) - I_L^0(x,y))^2 + \beta_1(\nabla I_R^0(x+d,y+v) - \nabla I_L^0(x,y))^2)$
- $\Psi^{'}_{div} = \Psi^{'}(|\nabla u|^2 + |\nabla v|^2 + \lambda|\nabla(d^{'} - d)|^2 + \mu|\nabla d|^2)$

Ces coefficients apparaissent dans les équations d'Euler obtenues en appliquant la condition nécessaire de minimalité e l'énergie : $\nabla J = 0$ ce qui sécrit encore $(\partial_u J, \partial_v J, \partial_d J, \partial_{d'} J) = (0,0,0,0)$. Ces coefficients permettent, grace aux propriétés de la fonction Ψ, 'être robuste aux grandes variations des composantes du terme d'attache aux données. Ecrivons les équations d'Euler-Lagrange nous obtenons :

$\frac{\partial J}{\partial u} = 0$

$$\Psi^{'}_{l_f low}(\partial_X I_L^t(I_L^t - I_L^0) + \beta_1(\nabla I_L^t - \nabla I_L^0).\partial_X \nabla I_L^t) + \Psi^{'}_{r_f low}(\partial_X I_R^t(I_R^t - I_R^0) + \beta_2(\nabla I_R^t - \nabla I_R^0).\partial_X \nabla I_R^t) + \Psi^{'}_{stereo}((I_R^t - I_L^t)(\partial_X I_R^t - \partial_X I_L^t) + \beta_3((\nabla I_R^t - \nabla I_L^t).\partial_X(\nabla I_R^t - \nabla I_L^t))) - \omega \operatorname{div}(\Psi^{'}_{div}\nabla u) = 0 \quad \text{(B.9)}$$

$\frac{\partial J}{\partial v} = 0$

$$\Psi^{'}_{l_f low}(\partial_Y I_L^t(I_L^t - I_L^0) + \beta_1(\nabla I_L^t - \nabla I_L^0).\partial_Y \nabla I_L^t) + \Psi^{'}_{r_f low}(\partial_Y I_R^t(I_R^t - I_R^0) + \beta_2(\nabla I_R^t - \nabla I_R^0).\partial_Y \nabla I_R^t) + \Psi^{'}_{stereo}((I_R^t - I_L^t)(\partial_Y I_R^t - \partial_Y I_L^t) + \beta_3((\nabla I_R^t - \nabla I_L^t).\partial_Y(\nabla I_R^t - \nabla I_L^t))) - \omega \operatorname{div}(\Psi^{'}_{div}\nabla v) = 0 \quad \text{(B.10)}$$

(B.11)

$\frac{\partial J}{\partial d'} = 0$

$$\Psi^{'}_{r_f low}(\partial_X I_R^t(I_R^t - I_R^0) + \beta_2(\nabla I_R^t - \nabla I_R^0).\partial_X \nabla I_R^t) + \Psi^{'}_{stereo}((I_R^t - I_L^t)\partial_X I_R^t + \beta_3((\nabla I_R^t - \nabla I_L^t).\partial_X(\nabla I_R^t))) - \omega \cdot \lambda \operatorname{div}(\Psi^{'}_{div}\nabla(d^{'} - d)) = 0 \quad \text{(B.12)}$$

(B.13)

$\frac{\partial J}{\partial d} = 0$

$$\Psi^{'}_{r_f low}(\partial_X I_R^0(I_R^0 - I_R^t) + \beta_2(\nabla I_R^0 - \nabla I_R^t).\partial_X \nabla I_R^0) + \Psi^{'}_{stereo-init}((I_R^0 - I_L^0)\partial_X I_R^0 + \beta_4((\nabla I_R^0 - \nabla I_L^0).\partial_X(\nabla I_R^0))) + \omega \operatorname{div}(\Psi^{'}_{div}(\lambda\nabla(d^{'} - d) + \mu\nabla d)) = 0 \quad \text{(B.14)}$$

(B.15)

Afin d'écrire le système linéaire correspondant, on linéarise les inconnues :

$$\begin{aligned} u^{k,l+1} &= u^{k,l} + \delta u^{k,l} \\ v^{k,l+1} &= v^{k,l} + \delta v^{k,l} \\ d^{k,l+1} &= d^{k,l} + \delta d^{k,l} \end{aligned}$$

On linéarise les équations afin d'aboutir à un système linéaire d'inconnues $\delta u^{k,l}, \delta v^{k,l}, \delta d^{',k,l}, \delta d^{k,l}$ Pour celà, on utilise des DL d'ordre 1 sur les termes faisant intervenir les images et leurs gradients :

$$\begin{aligned} I_L^{t,k}(x+u^{t,k,l+1}, y+v^{t,k,l+1}) = I_L^{t,k}(x+u^{t,k,l}, y+v^{t,k,l}) + \delta u^{t,k,l}\nabla_X I_L^{t,k} + \delta v^{t,k,l}\nabla_Y I_L^{t,k} \\ I_R^{t,k}(x+u^{t,k,l+1}+d^{t,k,l+1}(x+u^{t,k,l}, y+v^{t,k,l}), y+v^{t,k,l+1}) = I_R^{t,k}(x+u^{t,k,l} \\ +d^{t,k,l}(x+u^{t,k,l}, y+v^{t,k,l}), y+v^{t,k,l}) + (\delta u^{t,k,l}+\delta d^{t,k,l})\nabla_X I_R^{t,k} + \delta v^{t,k,l}\nabla_Y I_R^{t,k}) \end{aligned} \tag{B.16}$$

On peut ensuite écrire le système obtenu sous forme matricielle :
$Ax = b$ avec

$$A = \begin{pmatrix} 1/h & 0 & \cdots & 0 & -1/h & 0 & \cdots & 0 \\ 0 & 1/h & 0 & \cdots & 0 & -1/h & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 1/h & 0 & \cdots & 0 & -1/h & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & 0 & (a_{ij}) & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 1/h & 0 & \cdots & 0 & -1/h \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 1/h & 0 & \cdots & 0 & -1/h \end{pmatrix}$$

Ou les coefficients a_{ij} representent les coefficients en $(\delta u, \delta v, \delta d^{'}, \delta d)_{i,j}$ pour chacune des 4 équations d'Euler écrite en chaque pixel de l'image de référence. Pour un pixel de coordonnées (i,j) donnï¿½ la sous matrice des coefficients (a_{ij}) s'écrit :

$$\begin{pmatrix} 0 & \cdots & -\omega b^1_{i-1,j} & \cdots & -\omega b^1_{i,j-1} & a^1_{\delta u} & a^1_{\delta v} & a^1_{\delta d'} & a^1_{\delta d} & -\omega b^1_{i,j+1} & \cdots \\ 0 & \cdots & -\omega b^1_{i-1,j} & \cdots & -\omega b^1_{i,j-1} & a^2_{\delta u} & a^2_{\delta v} & a^2_{\delta d'} & a^2_{\delta d} & -\omega b^1_{i,j+1} & \cdots \\ 0 & \cdots & -\omega\lambda b^1_{i-1,j} & \cdots & -\omega\lambda b^1_{i,j-1} & a^3_{\delta u} & a^3_{\delta v} & a^3_{\delta d'} & a^3_{\delta d} & -\omega\lambda b^1_{i,j+1} & \cdots \\ 0 & \cdots & -\omega(\lambda+\mu) b^1_{i-1,j} & \cdots & -\omega(\lambda+\mu) b^1_{i,j-1} & a^2_{\delta u} & a^2_{\delta v} & a^2_{\delta d'} & a^2_{\delta d} & -\omega(\lambda+\mu) b^1_{i,j+1} & \cdots \end{pmatrix}$$

où les coefficients b^1 représentent les coefficients de divergence. Dans le cadre de l'utilisation d'un schéma à 8 points on doit aussi inclure les coefficients correspondants aux points en diagonale du pixel courant (i,j).

Equation 1 :

- $\delta u : \Psi^{'}_{l_f low}((\partial_X I^t_L)^2 + \beta_1((\partial^2_{XX} I^t_L)^2 + (\partial^2_{XY} I^t_L)^2)) + \Psi^{'}_{r_f low}((\partial_X I^t_R)^2 + \beta_2((\partial^2_{XX} I^t_R)^2 + (\partial^2_{XY} I^t_R)^2)) + \Psi^{'}_{stereo}((\partial_X I^t_R - \partial_X I^t_L)^2 + \beta_3((\partial^2_{XX} I^t_R - \partial^2_{XX} I^t_L)^2 + (\partial^2_{XY} I^t_R - \partial^2_{XY} I^t_L)^2)) + \omega(\Psi^{'}_{div_{i+\frac{1}{2},j}} + \Psi^{'}_{div_{i-\frac{1}{2},j}} + \Psi^{'}_{div_{i,j+\frac{1}{2}}} + \Psi^{'}_{div_{i,j-\frac{1}{2}}})$

- $\delta v : \Psi'_{l_flow}(\partial_X I^t_L \partial_Y I^t_L + \beta_1 \partial^2_{XY} I^t_L(\partial^2_{XX} I^t_L + \partial^2_{YY} I^t_L)) + \Psi'_{r_flow}(\partial_X I^t_R \partial_Y I^t_R + \beta_2 \partial^2_{XY} I^t_R(\partial^2_{XX} I^t_R + \partial^2_{YY} I^t_R)) + \Psi'_{stereo}((\partial_X I^t_R - \partial_X I^t_L)(\partial_Y I^t_R - \partial_Y I^t_L) + \beta_3((\partial^2_{XX} I^t_R + \partial^2_{YY} I^t_R - \partial^2_{XX} I^t_L - \partial^2_{YY} I^t_L)(\partial^2_{XY} I^t_R - \partial^2_{XY} I^t_L)))$
- $\delta d : \Psi'_{r_flow}((\partial_X I^t_R)^2 + \beta_2((\partial^2_{XX} I^t_R)^2 + (\partial^2_{XY} I^t_R)^2)) + \Psi'_{stereo}((\partial_X I^t_R - \partial_X I^t_L)\partial_X I^t_R + \beta_3((\partial^2_{XX} I^t_R - \partial^2_{XX} I^t_L)\partial^2_{XX} I^t_R + (\partial^2_{XY} I^t_R - \partial^2_{XY} I^t_L)\partial^2_{XY} I^t_R))$
- $\delta d^0 : -\Psi'_{r_flow}(\partial_X I^t_R \partial_X I^0_R + \beta_2(\partial^2_{XX} I^t_R \partial^2_{XX} I^0_R + \partial^2_{XY} I^t_R \partial^2_{XY} I^0_R))$
- Reste : $\Psi'_{l_flow}(\partial_X I^t_L(I^t_L - I^0_L) + \beta_1(\nabla I^t_L - \nabla I^0_L).\partial_X(\nabla I^t_L)) + \Psi'_{r_flow}(\partial_X I^t_R(I^t_R - I^0_R) + \beta_2(\nabla I^t_R - \nabla I^0_R).\partial_X(\nabla I^t_R)) + \Psi'_{stereo}((\partial_X I^t_R - \partial_X I^t_L)(I^t_R - I^t_L) + \beta_3((\partial_X I^t_R - \partial_X I^t_L)(\partial^2_{XX} I^t_R - \partial^2_{XX} I^t_L) + (\partial_Y I^t_R - \partial_Y I^t_L)(\partial^2_{XY} I^t_R - \partial^2_{XY} I^t_L))) - \omega \operatorname{div}(\Psi'_{div} \nabla u^k)$

Equation 2 :

- $\delta u : \Psi'_{l_flow}(\partial_X I^t_L \partial_Y I^t_L + \beta_1 \partial^2_{XY} I^t_L(\partial^2_{XX} I^t_L + \partial^2_{YY} I^t_L)) + \Psi'_{r_flow}(\partial_X I^t_R \partial_Y I^t_R + \beta_1 \partial^2_{XY} I^t_R(\partial^2_{XX} I^t_R + \partial^2_{YY} I^t_R)) + \Psi'_{stereo}((\partial_X I^t_R - \partial_X I^t_L)(\partial_Y I^t_R - \partial_Y I^t_L) + \beta_3(\partial^2_{XY} I^t_R - \partial^2_{XY} I^t_L)(\partial^2_{XX} I^t_R + \partial^2_{YY} I^t_R - \partial^2_{XX} I^t_L - \partial^2_{YY} I^t_L))$
- $\delta v : \Psi'_{l_flow}((\partial_Y I^t_L)^2 + \beta_1((\partial^2_{XY} I^t_L)^2 + (\partial^2_{YY} I^t_L)^2)) + \Psi'_{r_flow}((\partial_Y I^t_R)^2 + \beta_2((\partial^2_{XY} I^t_R)^2 + (\partial^2_{YY} I^t_R)^2)) + \Psi'_{stereo}((\partial_Y I^t_R - \partial_Y I^t_L)^2 + \beta_3((\partial^2_{XY} I^t_R - \partial^2_{XY} I^t_L)^2 + (\partial^2_{YY} I^t_R - \partial^2_{YY} I^t_L)^2)) + \omega(\Psi'_{div_{i+\frac{1}{2},j}} + \Psi'_{div_{i-\frac{1}{2},j}} + \Psi'_{div_{i,j+\frac{1}{2}}} + \Psi'_{div_{i,j-\frac{1}{2}}})$
- $\delta d : \Psi'_{r_flow}(\partial_X I^t_R \partial_Y I^t_R + \beta_2 \partial^2_{XY} I^t_R(\partial^2_{XX} I^t_R + \partial^2_{YY} I^t_R)) + \Psi'_{stereo}((\partial_Y I^t_R - \partial_Y I^t_L)\partial_X I^t_R + \beta_3((\partial^2_{XY} I^t_R - \partial^2_{XY} I^t_L)\partial^2_{XX} I^t_R + (\partial^2_{YY} I^t_R - \partial^2_{YY} I^t_L)\partial^2_{XY} I^t_R))$
- $\delta d^0 : -\Psi'_{r_flow}(\partial_Y I^t_R \partial_X I^0_R + \beta_2(\partial^2_{XY} I^t_R \partial^2_{XX} I^0_R + \partial^2_{YY} I^t_R \partial^2_{XY} I^0_R))$
- Reste : $\Psi'_{l_flow}(\partial_Y I^t_L(I^t_L - I^0_L) + \beta_1(\nabla I^t_L - \nabla I^0_L).\partial_Y(\nabla I^t_L)) + \Psi'_{r_flow}(\partial_Y I^t_R(I^t_R - I^0_R) + \beta_2(\nabla I^t_R - \nabla I^0_R).\partial_Y(\nabla I^t_R)) + \Psi'_{stereo}((\partial_Y I^t_R - \partial_Y I^t_L)(I^t_R - I^t_L) + \beta_3((\partial_X I^t_R - \partial_X I^t_L)(\partial^2_{XY} I^t_R - \partial^2_{XY} I^t_L) + (\partial_Y I^t_R - \partial_Y I^t_L)(\partial^2_{YY} I^t_R - \partial^2_{YY} I^t_L))) - \omega \operatorname{div}(\Psi'_{div} \nabla v^k)$

Equation 3 :

- $\delta u : \Psi'_{r_flow}((\partial_X I^t_R)^2 + \beta_2((\partial^2_{XX} I^t_R)^2 + (\partial^2_{XY} I^t_R)^2)) + \Psi'_{stereo}((\partial_X I^t_R - \partial_X I^t_L)\partial_X I^t_R + \beta_3((\partial^2_{XX} I^t_R - \partial^2_{XX} I^t_L)\partial^2_{XX} I^t_R + (\partial^2_{XY} I^t_R - \partial^2_{XY} I^t_L)\partial^2_{XY} I^t_R))$
- $\delta v : \Psi'_{r_flow}(\partial_X I^t_R \partial_Y I^t_R + \beta_2 \partial^2_{XY} I^t_R(\partial^2_{XX} I^t_R + \partial^2_{YY} I^t_R)) + \Psi'_{stereo}(\partial_X I^t_R(\partial_Y I^t_R - \partial_Y I^t_L) + \beta_3(\partial^2_{XX} I^t_R(\partial^2_{XY} I^t_R - \partial^2_{XY} I^t_L) + \partial^2_{XY} I^t_R(\partial^2_{YY} I^t_R - \partial^2_{YY} I^t_L)))$
- $\delta d : \Psi'_{r_flow}((\partial_X I^t_R)^2 + \beta_2((\partial^2_{XX} I^t_R)^2 + (\partial^2_{XY} I^t_R)^2)) + \Psi'_{stereo}((\partial_X I^t_R)^2 + \beta_3((\partial^2_{XX} I^t_R)^2 + (\partial^2_{XY} I^t_R)^2)) + \omega \cdot \lambda(\Psi'_{div_{i+\frac{1}{2},j}} + \Psi'_{div_{i-\frac{1}{2},j}} + \Psi'_{div_{i,j+\frac{1}{2}}} + \Psi'_{div_{i,j-\frac{1}{2}}})$
- $\delta d^0 : -\Psi'_{r_flow}(\partial_X I^t_R \partial_X I^0_R + \beta_2(\partial^2_{XX} I^t_R \partial^2_{XX} I^0_R + \partial^2_{YXY} I^t_R \partial^2_{XY} I^0_R)) - \omega \cdot \lambda(\Psi'_{div_{i+\frac{1}{2},j}} + \Psi'_{div_{i-\frac{1}{2},j}} + \Psi'_{div_{i,j+\frac{1}{2}}} + \Psi'_{div_{i,j-\frac{1}{2}}})$
- Reste : $\Psi'_{r_flow}(\partial_X I^t_R(I^t_R - I^0_R) + \beta_2(\nabla I^t_R - \nabla I^0_R).\partial_X(\nabla I^t_R)) + \Psi'_{stereo}(\partial_X I^t_R(I^t_R - I^t_L) + \beta_3(\nabla I^t_R - \nabla I^t_L).\partial_X(\nabla I^t_R))$

Equation 4 :

- $\delta u : \Psi'_{r_flow}(-\partial_X I^0_R \partial_X I^t_R - \beta_2(\partial^2_{XX} I^t_R \partial^2_{XX} I^0_R + \partial^2_{XY} I^t_R \partial^2_{XY} I^0_R))$
- $\delta v : \Psi'_{r_flow}(-\partial_X I^0_R \partial_Y I^t_R - \beta_2(\partial^2_{XY} I^t_R \partial^2_{XX} I^0_R + \partial^2_{YY} I^t_R \partial^2_{XY} I^0_R))$
- $\delta d : \Psi'_{r_flow}(-\partial_X I^0_R \partial_X I^t_R - \beta_2(\partial^2_{XX} I^t_R \partial^2_{XX} I^0_R + \partial^2_{XY} I^t_R \partial^2_{XY} I^0_R)) - \omega \cdot \lambda(\Psi'_{div_{i+\frac{1}{2},j}} + \Psi'_{div_{i-\frac{1}{2},j}} + \Psi'_{div_{i,j+\frac{1}{2}}} + \Psi'_{div_{i,j-\frac{1}{2}}})$
- $\delta d^0 : \Psi'_{r_flow}((\partial_X I^0_R)^2 - \beta_2((\partial^2_{XX} I^0_R)^2 + (\partial^2_{XY} I^0_R)^2)) + \Psi'_{stereo-init}((\partial_X I^0_R)^2 + \beta_4((\partial^2_{XX} I^0_R)^2 +$

$(\partial^2_{XY} I^0_R)^2)) + \omega \cdot (\lambda + \mu)(\Psi'_{div_{i+\frac{1}{2},j}} \Psi'_{div_{i-\frac{1}{2},j}} + \Psi'_{div_{i,j+\frac{1}{2}}} + \Psi'_{div_{i,j-\frac{1}{2}}})$

– Reste : $\alpha \Psi'_{r_f low}(\partial_X I^0_R(I^0_R - I^t_R) + \beta_2(\nabla I^0_R - \nabla I^t_R).\partial_X(\nabla I^0_R)) + \gamma \Psi'_{stereo-init}(\partial_X I^0_R(I^0_R - I^0_L) + \beta_4(\nabla I^0_R - \nabla I^0_L).\partial_X(\nabla I^0_R))$

Annexe C

Quelques mots sur le code source

Le code complet développé durant la thèse est disponible à l'adresse `http://devernay.free.fr/vision/varsceneflow/`. L'implémentation de l'algorthme d'estimation du flot de scène a été réalisée en C/C++ sous Linux et sous Mac OS X, en utilisant la librairie de traitement d'images OpenCV, disponible à l'adresse `http://sourceforge.net/projects/opencvlibrary/`. Le code réalisé permet de :

- Lire et prétraiter des séquences stéréo quelconques. Lissage gaussien, suppression des distorsions et rectification sont possibles.
- Estimer le flot optique sur les séquences gauche et droite séparément
- Estimer les cartes de disparité
- Estimer le flot de scène
- Générer des images en fausses couleurs des composantes du flot de scène, et les statistiques associées au flot de scène (Erreur angulaire, erreur RMS) et à la stéréo (Erreur RMS)
- Dans le cas où le système de caméras utilisé est calibré, permet la construction du flot de scène et la reconstruction de la scène 3D. L'ensemble des points et vecteurs 3D obtenus est stocké dans un fichier au format VTK, pouvant etre visualisé par le logiciel ParaView (`http://www.paraview.org/New/index.html`).
- Dans le cas où le système de caméras n'est pas calibré, le code permet la visualisation de la disparité sous forme de surface, constituant en soi une reconstruction projective de la scène.

Le code possède un nombre relativement restreint de fichiers, s'organisant essentiellement autour du moteur de calcul.

- *Resolution.cpp* implémente les fonctions de calcul du flot optique et du flot de scène, ainsi que des utilitaires de sauvegarde des images résultats en fausses couleurs.
- *GLVisu3D.cpp* implémente les routines de reconstruction 3D et d'estimation du flot de scène pour un système stéréo calibré, ainsi que les calculs des matrices de projection des caméras rectifiées.
- *main.cpp* contient le point d'entrée du code, permet de prétraiter les images,

et contient la boucle de traitement des paires stéréoscopiques : évaluation du flot optique et du flot de scène en appellant les fonctions implémentées dans *Resolution.cpp*. D'autre part les fichiers de résultats (images fausses couleurs des composantes du flot de scène et maillage de la scène 3D) sont écrits en fin de boucle pour deux paires stéréo consécutives données.
- *Classes.cpp* et *Utility.cpp* implémentent diverses structures et utilitaires nécessaires au code.
- D'autres fichiers implémentent un parseur d'analyse de fichiers de cas tests ou bien le calcul des statistiques d'erreur comparée à la vérité terrain.

Le code utilise les structures de base d'OpenCV pour les matrices (cvMat), ainsi que les opérateurs associés (cvAdd, cvMult, ...). Etant donné le nombre d'opérations sur les matrices l'utilisation d'une version GPU d'OpenCV permettrait une considérable accélération du temps de calcul. On pourrait par exemple envisager d'utiliser GPU CV, disponible à l'adresse `https://picoforge.int-evry.fr/cgi-bin/twiki/view/Gpucv/Web/`.

1 À propos du codage (teinte, saturation) des résultats

Il faut noter que pour générer les images du flot optique en codage (teinte, saturation) nous avons utilisé le code mis à disposition par Baker et al. [2007] à l'adresse `http://vision.middlebury.edu/flow/data/`. Les valeurs du champ de vecteur 2D sont rapportées à une grille de couleurs 2D, permettant de coder à la fois la norme et l'orientation des vecteurs, comme illustré en figure C.1.

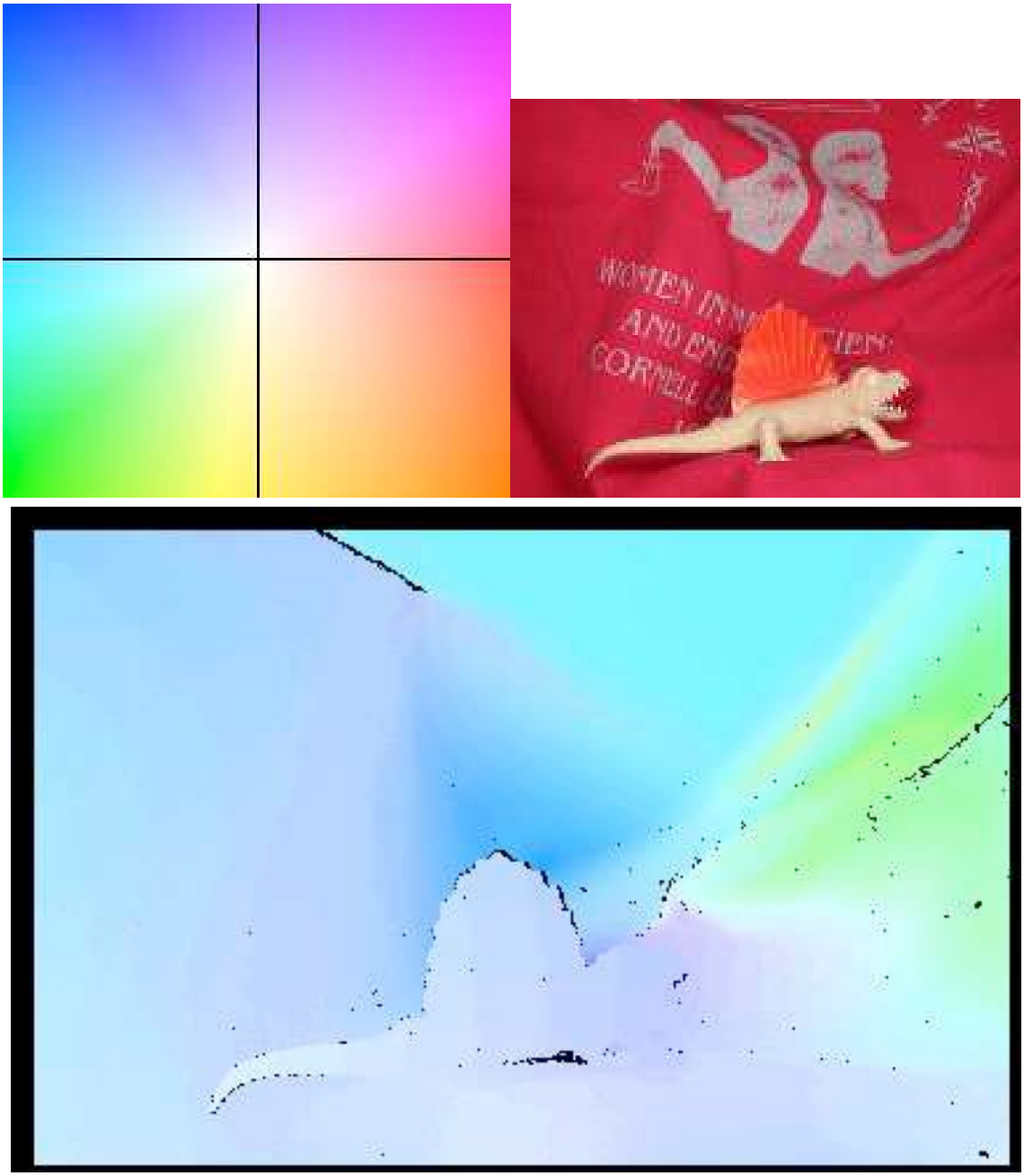

FIGURE C.1 – En haut à gauche : grille de couleurs utilisée pour représenter les valeurs (norme et angle) d'un champ vectoriel 2D, ou d'un champ scalaire. Dans ce dernier cas les valeurs sont encodées sur l'axe des abscisses. En haut à droite : la première image de la séquence Dimetrodon de la base de données Middlebury. En bas : la vérité terrain en codage (teinte, saturation)