HAL open science

Recalage déformable, jeux de données et protocoles d'évaluation pour la chirurgie mini-invasive abdominale augmentée

Richard Modrzejewski

▶ To cite this version:

Richard Modrzejewski. Recalage déformable, jeux de données et protocoles d'évaluation pour la chirurgie mini-invasive abdominale augmentée. Vision par ordinateur et reconnaissance de formes [cs.CV]. Université Clermont Auvergne [2017-2020], 2020. Français. NNT : 2020CLFAC044 . tel-03158762

HAL Id: tel-03158762

https://theses.hal.science/tel-03158762v1

Submitted on 4 Mar 2021

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.

Recalage déformable, jeux de données et protocoles d'évaluation pour la chirurgie mini-invasive abdominale augmentée

Richard Modrzejewski

Examinatrice :	Géraldine Morin, Professeure des Universités INPT, IRIT
Rapporteur :	Yohan Payan, Directeur de Recherche CNRS, TIMC
Rapporteur :	Caroline Essert, Maître de Conférences Universitaire et HDR Unistra, ICube
Directeur de Thèse :	Adrien Bartoli, Professeur des Universités UCA, Institut Pascal
Co-Encadrant :	Toby Collins, Directeur de Recherche, IRCAD France

THÈSE

Soutenue le 28 Août 2020

Présentée à l'Université Clermont Auvergne

pour l'obtention du grade de Docteur

(Décret du 5 juillet 1984)

Spécialité

VISION PAR ORDINATEUR

INSTITUT DE CHIRURGIE GUIDÉE PAR L'IMAGE

Résumé

Cette thèse concerne les techniques de recalage déformable de données pré-opératoires dans la scène per-opératoire en tant qu'étape indispensable à la réalisation de réalité augmentée pour la chirurgie abdominale. De telles techniques sont ainsi discutées, de même que les méthodologies d'évaluation associées à ces dernières. Deux contextes sont considérés : le recalage pour la chirurgie cœlioscopique assistée par ordinateur et le recalage postural de patient sur la table d'opération. Pour ces deux contextes, les besoins auxquels doivent répondre les algorithmes de recalage considérés sont discutés, ainsi que les principales limitations des solutions existantes. Des algorithmes réalisés au cours de cette thèse, permettant de répondre à ces besoins sont ainsi proposés et discutés. Une attention toute particulière est alors accordée à leur évaluation. Différents jeux de données permettant une évaluation quantitative de la précision des algorithmes de recalage, créés eux aussi durant cette thèse, et rendu publics, sont ainsi présentés. De telles données sont extrêmement importantes car elles répondent à un manque de données standardisées permettant d'évaluer l'erreur de recalage de façon quantitative, et ainsi de comparer les différents algorithmes. La modélisation de l'éclairage de la scène cœlioscopique, permettant d'extraire des contraintes fortes sur les données à recaler et la surface de l'organe observé, et ainsi d'être utilisée pour contraindre ces problématiques de recalage, est aussi discutée. Ce manuscrit est séparé en sept parties. La première traite du contexte de la thèse. La chirurgie mini-invasive est présentée ainsi que différents problèmes de vision par ordinateur généraux qui, une fois appliqués au contexte médical permettent de définir la chirurgie assistée par ordinateur. La seconde partie traite des prérequis à la lecture de la thèse. Le prétraitement des données pré-opératoires et per-opératoires, avant utilisation par les algorithmes de recalage présentés, est ainsi discuté. La troisième partie correspond au recalage de données hépatiques en cœlioscopie, et de l'évaluation associée à cette méthode. La quatrième partie correspond quant à elle à la problématique du recalage postural. La cinquième partie propose une modélisation de l'éclairage en cœlioscopie pouvant être utilisée pour obtenir des contraintes fortes entre la surface observée et les images cœlioscopiques. La sixième partie propose une utilisation des modèles de lumière discutés dans la partie précédente afin de raffiner et densifier des reconstructions de la scène cœlioscopique. Enfin, la septième et dernière partie correspond à nos conclusions vis-à-vis des problématiques abordées au cours de la thèse, et aux travaux futurs.

Mots clés : Réalité augmentée, Chirurgie assistée par ordinateur, Cœlioscopie, Recalage non-rigide, Modélisation d'éclairage, Shading, Multi-view stereo

Abstract

This thesis deals with deformable registration techniques of preoperative data to the intra-operative scene as an indispensable step in the realisation of augmented reality for abdominal surgery. Such techniques are thus discussed as well as evaluation methodologies associated with them. Two contexts are considered : the registration for computer-assisted laparoscopic surgery and the postural registration of the patient on the operating table. For these two contexts, the needs to be met by the registration algorithms considered are discussed, as well as the main limitations of the existing solutions. Algorithms developped during this thesis, allowing to meet these needs are thus proposed and discussed. Special attention is given to their evaluation. Different datasets allowing a quantitative evaluation of the accuracy of the registration algorithms, also realized during this thesis, and made public, are also discussed. Such data are extremely important because they respond to a lack of evaluation data needed in order to evaluate the registration error in a quantitative way, and thus to compare the different algorithms. The modeling of the illumination of the laparoscopic scene, allowing one to extract strong constraints between the data to be registered and the surface of the observed organ, and thus to be used to constrain these registration problems, is also discussed. This manuscript has seven parts. The first deals with the context surrounding this thesis. Minimally invasive surgery is presented as well as various general computer vision problems which, when applied to the medical context, allow the definition of computer-assisted surgery. The second part deals with the prerequisites for reading the thesis. The pre-processing of pre-operative and per-operative data, before their use by the presented registration algorithms, is thus discussed. The third part corresponds to the registration of hepatic data in laparoscopy, and the evaluation associated with this problems. The fourth part deals with the problem of postural registration. The fifth part proposes a modelling of the lighting in laparoscopy which can be used to obtain strong constraints between the observed surface and the laparoscopic images. The sixth part proposes a use of the light models discussed in the previous part in order to refine and densify reconstructions of the laparoscopic scene. Finally, the seventh and last part corresponds to our conclusions regarding the issues addressed during this thesis, and to future work.

Mots clés : Augmented reality, Computer assisted surgery, Laparoscopy, Deformable registration, Light Modelling, Shading, Multi-view stereo

Table des matières

Acronymes **8**

Liste des symboles utilisés **10**

1 Introduction **12**

1.1 Contexte clinique : chirurgie mini-invasive et cœlioscopie 13

1.2 Vision par ordinateur pour assister la cœlioscopie 14

1.3 Recalage de données pré-opératoires dans la scène per-opératoire 16

1.3.1 Motivations et difficultés principales 16

1.3.2 Pipeline général de recalage 17

1.4 Contributions et plan du manuscrit 19

2 Prérequis et outils utilisés **22**

2.1 Introduction 23

2.2 Construire le modèle de l'organe à partir d'imagerie 3D 23

2.2.1 Reconstruction du modèle géométrique 24

2.2.2 Construction du modèle biomécanique 26

2.2.3 Réduction de l'espace des déformations 28

2.3 Reconstruire la surface de l'organe considéré à partir d'images cœlioscopiques 29

2.3.1 Différentes techniques de reconstruction 31

2.3.2 Détection d'erreur de reconstruction par validation croisée stéréo 38

2.4 Détection de la silhouette de l'organe 40

2.4.1 Différencier contours occultants et d'occultation 41

2.4.2 Détection de contours 42
2.5 Conclusion 43

3 Recalage initial de modèles hépatiques pour la cœlioscopie augmentée et jeu de données d'évaluation 46
3.1 Introduction 47
3.2 État de l'art 48
3.2.1 Recalage initial mono-image 49
3.2.2 Recalage initial multi-images 52
3.2.3 Évaluation du recalage 54
3.3 Formulation par un problème d'optimisation 55
3.3.1 Généralités 55
3.3.2 Énergies à minimiser 56
3.4 Gestion des collisions internes 62
3.4.1 Prévenir les auto-intersections par la modélisation de collisions . 62
3.4.2 Énergie additionnelle de répulsion 66
3.5 Résolution 68
3.6 Évaluation du recalage en cœlioscopie 70
3.6.1 Jeu de données 70
3.6.2 Méthodologie chirurgicale 72
3.6.3 Des images à l'évaluation 74
3.7 Résultats expérimentaux 77
3.8 Conclusion 80

4 Recalage postural 82
4.1 Introduction 83
4.2 État de l'art 84
4.3 Méthode de recalage proposée 85
4.4 Un modèle biomécanique hétérogène 86
4.5 Gérer les collisions externes 88
4.5.1 Modèle paramétrique de l'objet 88
4.5.2 Énergie correspondante 89
4.6 Énergie de correspondance de surface 89
4.7 Résolution 90

4.8 Expériences 90
4.8.1 Méthodologie d'évaluation 90
4.8.2 Résultats expérimentaux 92
4.9 Automatisation du recalage rigide initial 99
4.10 Conclusion 100

5 Modélisation de l'éclairage en cœlioscopie 102
5.1 Introduction 103
5.2 État de l'art 105
5.3 Équation de rendu 111
5.4 Modèle de caméra 111
5.4.1 Modèle géométrique 111
5.4.2 Modèle photométrique 113
5.5 Modèle de réflectance 116
5.6 Modèles de lumière 118
5.6.1 Modèles ponctuels 118
5.6.2 Modèle étendu 120
5.7 Étalonnage de la source lumineuse 121
5.7.1 Méthode générale 121
5.7.2 Extraction du motif 124
5.8 Résultats expérimentaux 130
5.9 Conclusion 132

6 Densification de la reconstruction cœlioscopique en utilisant la photométrie 134
6.1 Introduction 135
6.2 État de l'art 136
6.2.1 Paramétrisations 136
6.2.2 Photo-consistance 138
6.2.3 Éclairages 139
6.3 Vue de référence, modèle de surface et propriétés émergentes 140
6.3.1 Description générale 140
6.3.2 Caractère C^1 de la surface considérée 143
6.3.3 Expression de la normale 148

6.4 Formulation par un problème d'optimisation 149
6.4.1 Énergies à considérer 149
6.4.2 Résolution 150
6.5 Méthodologie d'évaluation 152
6.5.1 Données d'évaluation 152
6.5.2 Métriques utilisées 154
6.6 Résultats expérimentaux 156
6.7 Conclusion 158

7 Conclusion et travaux futurs 160
7.1 Conclusions 161
7.1.1 Recalage de données hépatiques en cœlioscopie 162
7.1.2 Recalage postural 162
7.1.3 Modélisation de l'éclairage en cœlioscopie 163
7.1.4 Raffinage de reconstruction de surface de la scène per-opératoire en cœlioscopie 164
7.2 Perspectives 164
7.2.1 Utilisation de modèle biomécanique hétérogène de foie et évaluation sur un modèle clinique 165
7.2.2 Utilisation de données ultrasons en tant que contraintes supplémentaires pour le recalage 165
7.2.3 Fusion de surface pour l'algorithme de Shading Aware Multi-View Stereo (SAMVS) 166
7.2.4 Transfert de carte d'albédo au modèle et recalage temps réel . . 166

A Réduction de l'espace des déformations 168
A.1 Calcul simple des sous-espaces de faibles déformations 169
A.2 Centrage des données 172
A.3 Optimisation numérique par Gauss-Newton 174
A.4 Utilisation des espaces réduits dans le problème de minimisation 176

B Détection d'erreurs de reconstruction par validation croisée stéréo 178
B.1 Méthode 179
B.2 Résultats expérimentaux 180

Acronymes

ALS	Area Light Source.
BRIEF	Binary Robust Independent Elementary Features.
CRF	Fonction de Réponse de la Caméra.
DSSIM	Dissymétrie Structurelle.
FALS	Fixed Area Light Source.
FAST	Features from Accelerated Segment Test.
FEM	Méthode des Éléments Finis.
FPLS	Fixed Point Light Source.
FPSLS	Fixed Polynomial Spot Light Source.
FRE	Fiducial Registration Error.
FSLS	Fixed Spot Light Source.
GMM	Gaussian Mixture Model.
ICP	Iterative Closest Point.
IRM	Imagerie par Résonance Magnétique.
LEDs	Diodes Électro-Luminescentes.

MVS	Multi-View Stereo.
NCC	Normalized Cross Correlation.
ORB	Oriented FAST and Rotated BRIEF.
PLS	Point Light Source.
PSLS	Polynomial Spot Light Source.
RA	Réalité Augmentée.
RMN	Résonance Magnétique Nucléaire.
SAMVS	Shading Aware Multi-View Stereo.
SfC	Shape from Color.
SfM	Structure from Motion.
SfS	Shape from Shading.
SIFT	Scale-Invariant Feature Transform.
SLAM	Simultaneous Localization And Mapping.
SLS	Spot Light Source.
SSIM	Similarité Structurelle.
SURF	Speeded Up Robust Features.
TDM	Tomodensitométrie.
TRE	Target Registration Error.
VSLAM	Visual Simultaneous Localization And Mapping.

Liste des symboles utilisés

$X^\top$	Transposée de X
X^{-1}	Inverse de X
X^+	Pseudo-inverse de X
$\dim(E)$	Dimension de E
$\mathrm{vec}(X_1, \cdots, X_n)$	Espace vectoriel induit par les vecteurs $X_1, \cdots, X_n$
$\langle X \vert Y \rangle$	Produit scalaire entre X et Y
$\lVert X \rVert$	Norme 2 de X
$\mathrm{proj}(p, I)$	Projection du point p dans l'image I

Chapitre 1

Introduction

Le but de ce chapitre est d'introduire le contexte de cette thèse ainsi que les différentes problématiques abordées. La principe de la cœlioscopie, une technique chirurgicale, est donc présenté, et l'accent est mis sur la difficulté d'une telle procédure. Différentes techniques de vision par ordinateur, pouvant être utilisées afin d'assister le chirurgien pendant cette procédure, sont ainsi présentées. Parmi ces techniques, l'attention est portée tout particulièrement sur le recalage de données pré-opératoires dans la scène per-opératoire, qui représente le cœur de cette thèse.

1.1 Contexte clinique : chirurgie mini-invasive et cœlioscopie

La cœlioscopie est une technique de chirurgie dite *mini-invasive*. Au cours de celle-ci plusieurs incisions de quelques centimètres sont réalisées au niveau de la paroi abdominale. De petits tubes creux, appelés trocarts sont ensuite insérés à travers celles-ci, permettant ainsi d'introduire une fibre optique, reliée à une caméra appelée cœlioscope, ainsi que les différents outils nécessaires à l'opération. Du gaz carbonique est alors insufflé à travers l'un de ces tubes afin de soulever la paroi abdominale et de repousser l'intestin, ce qui permet de visualiser l'intérieur de la cavité abdominale et de pouvoir y manipuler correctement les outils (Figure 1.1). Différents organes sont ainsi facilement accessibles comme le foie, l'utérus, les ovaires, et les reins. Cette technique, inventée en France en 1946 par le docteur Palmer fut d'abord utilisée à des fins diagnostiques. Mais ce n'est qu'en 1972 que la cœlioscopie chirurgicale fut réellement créée à Clermont-Ferrand par le docteur Mouret. L'invention de cette technique fut une véritable révolution en médecine, permettant de réduire grandement les difficultés post-opératoires telles que les infections et la fragilisation de la paroi abdominale, dues à l'ouverture de cette dernière lors de chirurgie classique. Le temps de rémission du patient se trouve ainsi grandement diminué. Cependant, cette technique n'est pas sans inconvénient, le principal étant la complexité de la procédure en elle-même. Ainsi, contrairement à la chirurgie ouverte, la scène n'est pas vue directement mais par l'intermédiaire de la caméra du cœlioscope et de l'écran externe, associé à ce dernier. Cela limite ainsi fortement le champ de vision du chirurgien qui doit alors déplacer la caméra afin de se repérer correctement, mouvement lui-même limité par le trocart dans lequel est inséré le cœlioscope. De plus, les outils chirurgicaux sont eux aussi utilisés à travers les trocarts. Leurs déplacements sont ainsi limités et leur maniements demande alors une plus grande dextérité et une synchronisation avec le mouvement de la caméra. Finalement, la palpation de l'organe permettant, lors de la chirurgie ouverte, de localiser des structures internes telles que les vaisseaux sanguins et les tumeurs n'est bien entendu, ici pas possible. Afin de surmonter ces difficultés et d'aider le chirurgien dans sa tâche, différentes techniques basées sur la vision par ordinateur ont été proposées afin d'assister la chirurgie.

FIGURE 1.1 – Chirurgie cœlioscopique en pratique. Illustration tirée de la banque d'images shutterstock.com.

1.2 Vision par ordinateur pour assister la cœlioscopie

La vision par ordinateur est l'art d'extraire, d'analyser, de comprendre et d'utiliser l'information utile, présente dans les images, de façon automatique. Elle est en général divisée en trois grands domaines. Le premier regroupe la reconnaissance et la segmentation d'images. Il consiste à analyser la composition de l'image, les différents objets présents, certaines spécificités de ceux-ci, mais aussi l'emplacement de ces données dans l'image. Le second est le domaine de la reconstruction 3D : une ou plusieurs images sont associées, avec ou sans connaissance supplémentaire sur l'environnement dans lequel elles ont été prises, afin de retrouver l'information de profondeur présente dans celles-ci. Le troisième regroupe tous les problèmes dits de recalage, qui consistent à mettre en correspondance plusieurs ensembles d'images, de modalités différentes ou non, afin de pouvoir utiliser conjointement les informations présentes dans ces ensembles.

Tous ces domaines sont ainsi extrêmement importants pour la cœlioscopie assistée par ordinateur, permettant d'aider le chirurgien dans sa tâche à l'aide d'informations dé-

tectées de façon automatique dans les images. Parmi différentes applications réunies sous cette dénomination, nous avons par exemple la reconstruction de surface per-opératoire de l'organe, pouvant se faire à des buts diagnostiques [199, 133, 117, 140, 55, 34, 33, 143]. Le chirurgien est ainsi capable d'observer la surface en question en 3D, facilitant la vision de cette dernière. Un autre type d'application inclus sous cette dénomination comporte toutes les méthodes d'aide au diagnostic à l'aide d'images. Ce genre de méthodes ayant, de plus, connu un intérêt croissant avec l'avènement du *deep-learning* [67, 218]. Et enfin, on trouve également sous cette dénomination toutes les techniques permettant la cœlioscopie guidée par Réalité Augmentée (RA), aussi appelée cœlioscopie augmentée, consistant à recaler des images volumiques de l'organe considéré, contenant les informations sur ses structures internes (vaisseaux sanguins et tumeurs), sur des images cœlioscopiques 2D du même organe [125, 69, 20, 27]. Ces images volumiques sont obtenues par Tomodensitométrie (TDM), ou Imagerie par Résonance Magnétique (IRM) et doivent être préalablement segmentées afin de localiser l'organe considéré et ses structures internes. La prise d'image par TDM consiste à mesurer l'absorption des rayons X par les différents tissus et d'afficher ces différences d'absorption dans les images en sortie, permettant alors de distinguer les différents tissus présents. L'IRM est quant à elle une technique non-irradiante, et donc moins nocive pour l'organisme. Elle se base sur le principe de Résonance Magnétique Nucléaire (RMN). Un champ magnétique puissant et stable est produit, permettant une magnétisation des tissus. Des champs magnétiques plus faibles dit *radiofréquences* sont ensuite appliqués afin de faire varier cette magnétisation. Cela a ainsi pour effet d'engendrer un champ électromagnétique produit par les tissus, détectable, et dont les caractéristiques dépendent de la structure chimique de ces derniers. Il devient alors possible de produire en sortie, une image volumique dans laquelle les différents tissus sont différenciés. Le recalage de telles images dans la scène cœlioscopique peut être rigide, si la capture de ces images est réalisée en per-opératoire [138], ou déformable dans le cas contraire [98, 131, 70]. À noter que la plupart de hôpitaux ne possédant pas de scanner interventionnel TDM ou IRM, un recalage déformable est nécessaire dans la plupart des cas. Même lorsqu'un tel matériel est disponible, il est à rappeler que le traitement de ces données après acquisition (segmentation des structures que l'on souhaite mettre en valeur dans la scène augmentée) prend un certain temps. Un recalage déformable de données pré-opératoires, traitées préalablement est ainsi souvent privilégié, même dans ce cas. De plus, dans

le cadre des images TDM, le risque de surexposition aux radiations est aussi présent. Le nombre d'acquisitions d'images doit ainsi être limité. Des déformations de l'organe étant toujours induites par le chirurgien pendant l'opération, et une acquisition d'image n'étant en pratique pas possible après chacune d'elles, un recalage déformable est ainsi toujours préférable. Remarquons aussi que la cœlioscopie augmentée peut aussi être utilisée dans le cadre de la chirurgie assistée par la robotique, dans laquelle le chirurgien opère par l'intermédiaire d'un robot, permettant de rendre ces mouvements plus précis [202, 214, 147]. En effet, les difficultés sont ici les mêmes, et une augmentation des images avec les structures internes de l'organe peut ainsi toujours faciliter la procédure.

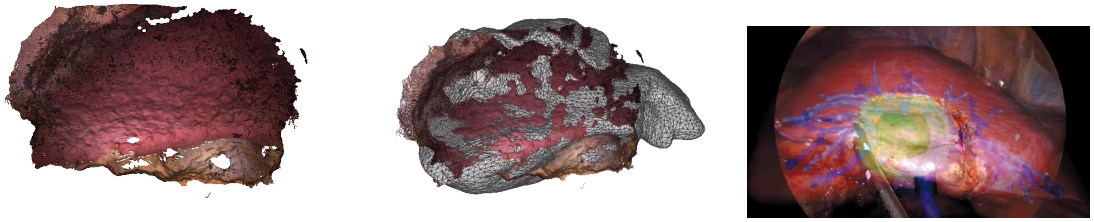

FIGURE 1.2 – Quelques exemples d'utilisation de la vision par ordinateur pour le médical : reconstruction (gauche), recalage (centre) et réalité augmentée (droite). La reconstruction est réalisée avec ORBSLAM2 [118], méthode de SLAM discutée dans ce chapitre. Le recalage a quant-à lui été réalisé à l'aide de notre méthode décrite dans le chapitre 3. La réalité augmentée n'est ici qu'une illustration et provient de [125].

1.3 Recalage de données pré-opératoires dans la scène per-opératoire

1.3.1 Motivations et difficultés principales

Nous nous intéressons dans cette thèse à la chirurgie assistée par ordinateur. En particulier, nous nous intéressons au problème de recalage de modèles, construits à partir d'images volumiques obtenues par TDM ou IRM, sur des images 2D ou 2.5D (des images RGB-D). Le premier cas étudié est ainsi le cas du recalage adapté au contexte de la chirurgie cœlioscopique hépatique et est discuté dans le chapitre 3. Le second, quant à lui considère le problème de recalage de posture du patient sur la

table d'opération. Les images utilisées en tant que cible du recalage sont ainsi des images RGB-D. Différentes solutions à ces problèmatiques de recalage ont déjà été proposées [98, 70, 38, 73, 31, 5, 10]. Cependant, aucune de ces techniques n'est sans défaut ni n'est adaptée pour tout organe. Ce problème reste ainsi un problème non résolu de façon satisfaisante. L'une des principales applications de ce recalage est la cœlioscopie augmentée. En effet, une fois les images volumiques recalées aux images endoscopiques, les structures internes de l'organe considéré peuvent être affichées dans ces dernières, augmentant ainsi la vue du chirurgien.

On étudiera dans ce manuscrit le cas particulier du foie, s'agissant d'un des cas les plus difficiles à traiter. En effet, plusieurs difficultés sont présentes. Elles ne sont pas propres à ce cas mais sont tout du moins exacerbées par ce dernier. Ainsi, contrairement à d'autres organes tels que les reins et l'utérus, pouvant quant-à eux, être considérés comme relativement rigides pendant la majorité de la procédure chirurgicale, ce n'est pas le cas pour le foie qui est un organe extrêmement élastique. S'ajoute à cela une qualité d'image moindre par rapport à des images utilisées dans d'autres applications de vision par ordinateur dans des contextes plus classiques (avec présence de fumées, de sang, et de fort changements d'éclairages). De plus, ce cas est aussi très pertinent d'un point de vue clinique car peu de travaux existent à l'heure actuelle dans ce contexte.

1.3.2 Pipeline général de recalage

Plusieurs possibilités existent lorsque l'on considère le recalage de données pré-opératoires sur une vue 2D ou 2.5D. On se limite dans ce manuscrit au cadre étudié durant la thèse, c'est-à-dire au recalage de données hépatiques ainsi que de données posturales. Dans cette optique, nous nous limitons aussi au seul pipeline de recalage considéré dans nos travaux que nous allons décrire dans la suite de cette partie [38, 35]. Nous nous intéressons ici plus en détail au cas du recalage en chirurgie cœlioscopique, le cas du recalage postural étant très similaire sur le principe. Le recalage est séparé en trois étapes. 1) En pré-opératoire, une acquisition d'images TDM du patient est tout d'abord réalisée. Les images TDM ou IRM correspondantes sont alors segmentées (comme illustré en Figure 1.3) et utilisées pour générer un modèle biomécanique de l'organe.

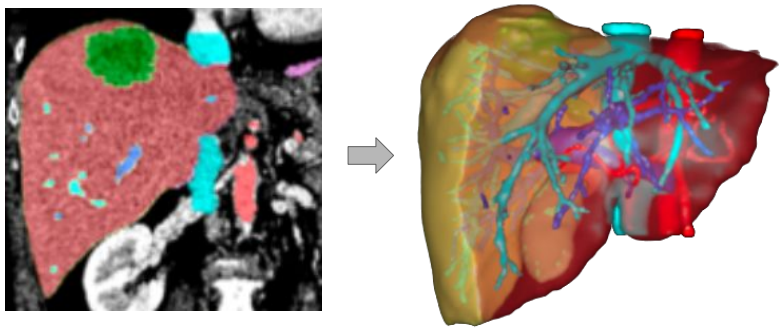

FIGURE 1.3 – Segmentation d'images TDM de foies et récupération du modèle géométrique correspondant contenant les structures internes segmentées. Illustration provenant de Visible Patient.

Ensuite, le jour de l'opération, le chirurgien réalise au tout début de la chirurgie cœlioscopique, une vidéo d'exploration de la scène ou il essaie de voir le plus possible de la surface de l'organe. Lors de cette vidéo, le patient est mis artificiellement en apnée afin de limiter les mouvement induits par la respiration dans la scène qui peut alors être considérée comme rigide. 2) Le modèle est alors tout d'abord recalé à cette vidéo, ce qui permet alors de texturer le modèle biomécanique de l'organe. 3) Une fois cette étape réalisée, les informations de texture peuvent être utilisées pour induire des déformations spécifiques de l'organe en fonction de nouvelles images arrivant en entrée. C'est l'étape de *tracking* pendant laquelle le chirurgien peut manipuler l'organe tout en étant capable de déformer l'organe en temps réel conformément à la scène observée.

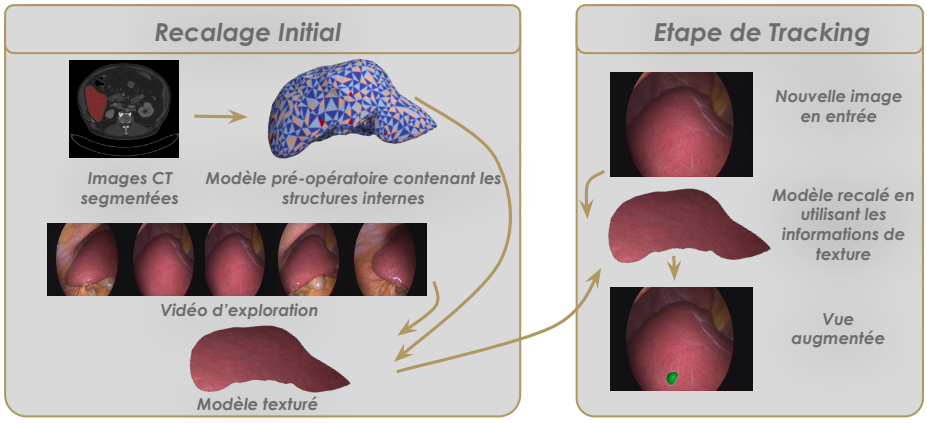

FIGURE 1.4 – Illustration du pipeline général de recalage considéré.

Différents défis sont à relever lors de l'utilisation d'un tel pipeline. Ainsi, une modélisation réaliste des déformations de l'organe doit être réalisée. De plus, différents indices visuels permettant de bien définir la cible du recalage doivent être extraits des images. L'étape de texturation du modèle doit, quant à elle, prendre en compte les forts changements d'éclairage de la scène. S'ajoute à cela des contraintes sur le temps de résolution, devant être compatible avec le *workflow* chirurgical.

1.4 Contributions et plan du manuscrit

Dans ce manuscrit, nous nous concentrons sur le recalage initial d'un modèle pré-opératoire à la vidéo d'exploration, et nous utilisons en tant que cible de notre recalage une reconstruction de la surface observée. Nous nous intéressons aussi à la façon dont on peut améliorer ce recalage en détectant les erreurs dans notre surface cible et en raffinant cette dernière en tirant parti du contexte particulier de l'éclairage en cœlioscopie. Le chapitre 2 traite ainsi de différentes techniques générales utilisées dans les autres chapitres parmi lesquels la construction de modèles biomécaniques, la reconstruction de surfaces per-opératoires, la détection d'erreurs dans celles-ci, et la réduction de l'espace de recherche des déformations (permettant de réduire le temps de calcul tout en ne gardant que des déformations lisses). Le chapitre 3 décrit le problème du recalage initial dans le cadre de la cœlioscopie. La principale nouveauté décrite dans ce chapitre concerne la gestion des collisions internes au sein du modèle utilisé. De plus,

dans ce chapitre, nous décrivons aussi un jeu de données *in vivo* d'évaluation, créé afin de valider notre méthode mais permettant aussi de la comparer avec d'autres. Ce jeu de données est public, et est construit à partir de données hépatiques porcines. Dans ces données, des marqueurs métalliques sont injectés dans l'organe et fixés à la surface de ce dernier. Ils permettent ainsi d'obtenir l'erreur véritable en chacun de ces points. Cette base de donnée est importante car elle est la première base publique à permettre une telle évaluation, comblant ainsi un manque véritable. Elle pousse ainsi la recherche en avant dans ce domaine, permettant la comparaison de différents algorithmes de recalage sur des données communes. Le chapitre 4 propose, d'utiliser une extension de l'algorithme décrit dans le chapitre 3 pour résoudre le problème de recalage de posture. Le problème est ainsi de recaler un modèle pré-opératoire de l'abdomen du patient, à ce dernier, allongé sur la table, le jour de l'opération. Dans ces travaux, la modélisation de la collision du patient avec la table d'opération est réalisée afin d'améliorer la précision du recalage. Une fois encore, une attention toute particulière est portée à l'évaluation de ce recalage. Une base de données porcine est utilisée ici aussi. Des marqueurs placés à la surface de la peau de porcs sont utilisés et permettent de calculer l'erreur en chacun de ces points. Dans le chapitre 5, nous étudions la façon de modéliser l'éclairage dans la scène cœlioscopique. Une telle modélisation est importante car elle permet d'induire de fortes et denses contraintes entre les images et la surface de l'organe observé. Généralement, le pipeline de réalité augmentée pour la cœlioscopie peut être divisé en plusieurs étapes : la reconstruction de la surface de la scène per-opératoire, le recalage du modèle pré-opératoire à cette reconstruction, la texturation de ce même modèle en utilisant les images ainsi que le recalage réalisé, et enfin l'étape de *tracking* temps réel utilisant l'information de texture pour se recaler. Ces contraintes provenant de l'éclairage de la scène peuvent ainsi être utilisées à différents stades de ce pipeline de recalage afin d'améliorer la précision de celui-ci. Ces contraintes peuvent alors être utilisées directement par le recalage, mais aussi pour obtenir de meilleures reconstructions per-opératoire de l'organe, utilisées en tant que cible du recalage. De plus, l'albédo de l'organe peut, lorsque l'éclairage est modélisé, être directement extrait et utilisé pour texturer le modèle pré-opératoire. Ceci permet alors d'obtenir une texture ne dépendant pas de l'éclairage utilisé. Cela permet ainsi, pour des applications de cœlioscopie augmentée d'améliorer la robustesse de l'étape finale de *tracking*. Différents modèles de lumière, plus ou moins simples, sont alors évalués et comparés. Une évaluation est

ainsi réalisée en utilisant des images de mires de recalage, et en évaluant la différence entre les blancs perçus et ceux attendus conformément aux modèles. Puis dans le chapitre 6 nous utilisons ces modèles de lumière afin de densifier les surfaces cibles du recalage, en décrivant un algorithme de SAMVS. La surface reconstruite contient ainsi l'information sur l'albédo de la surface et n'est ainsi pas dépendante de l'éclairage de la scène. L'évaluation de la surface est réalisée à l'aide de données hépatiques *ex-vivo*. La méthodologie concernant cette évaluation est une fois encore discutée dans le chapitre correspondant. Le chapitre 7 donne les conclusions de ce manuscrit et présente les travaux futurs. Enfin deux annexes terminent le manuscrit. La première décrit plus en détail le problème de la réduction de l'espace des déformations que nous utilisons dans nos travaux. La seconde, décrit une méthode permettant de détecter des erreurs dans les reconstructions utilisées en tant que cible du recalage lorsqu'un cœlioscope stéréo a été utilisé pour la prise d'image. Pour chacun de ces chapitres, un état de l'art détaillé, spécifique aux problématiques discutées dans le chapitre considéré, est réalisé. Quatre publications, évaluées par les pairs, ont étés réalisées pendant cette thèse et sont associées à ce manuscrit. La première a été réalisée pour Surgetica 2017, et traite de la détection d'erreur dans des reconstructions en présence d'information stéréo. Elle est ainsi associée aux travaux décrits dans l'annexe B. La seconde a été publiée dans IJCARS, à l'occasion de la conférence internationale IPCAI 2019 [131] et traite de la gestion des collisions internes au sein du modèle biomécanique ainsi que de la base de données *in vivo* publique réalisée initialement pour évaluer notre travail. Cette publication correspond aux travaux décrits dans le chapitre 3. La troisième fut réalisée pour la conférence internationale MICCAI 2018 [128] et traite de la question du recalage postural. Elle est associée aux travaux décrits dans le chapitre 4. Enfin, la quatrième et dernière contribution fut publiée dans IJCARS, pour la conférence IPCAI 2020 [130] et traite de la modélisation de l'éclairage dans la scène cœlioscopique et de son utilisation pour la densification de surface. Elle est associée aux chapitres 5 et 6 de ce manuscrit.

Chapitre 2

Prérequis et outils utilisés

Ce chapitre a pour objectif de définir les prérequis à la lecture de ce manuscrit. Dans celui-ci, nous nous concentrons sur le problème du recalage de données pré-opératoires hépatiques à la scène cœlioscopique, qui correspond à la principale application de nos travaux. Cependant, les concepts présentés sont généraux et il est ainsi possible d'étendre ceux-ci au recalage déformable d'autres organes. Nous verrons ainsi tout d'abord comment le modèle biomécanique de l'organe est construit à partir des images volumiques (provenant d'imagerie TDM ou IRM). C'est ce modèle qui est ensuite recalé à la scène cœlioscopique per-opératoire. Afin de réaliser ce recalage, une méthode énergétique est utilisée. Une fonction objectif, associée au recalage, et ainsi créée et minimisée. Pour réaliser cette minimisation, des méthodes d'optimisation numérique telles que l'algorithme de Gauss-Newton ou de Levenberg-Marquardt sont utilisées [17]. Afin de réduire le temps d'exécution de ces minimisations, une méthode permettant de réduire les dimensions du problème est employée. Cette méthode est ainsi discutée dans ce chapitre. En tant que cible du recalage, des reconstructions de la surface observée sont utilisées. Un coût associé au recalage, et représentant la proximité entre surface du modèle biomécanique et surface reconstruite est ainsi défini et ajouté à la fonction objectif. Cela permet ainsi de s'assurer de la correspondance entre ces deux surfaces en fin de recalage. Différentes méthodes de reconstruction utilisées dans dans nos travaux sont ainsi elles aussi discutées dans ce chapitre. À cause de la qualité des images utilisées dans ce contexte, ces reconstructions peuvent cependant contenir des erreurs. Nous expliciterons ainsi une méthode permettant de détecter d'éventuelles erreurs dans ces reconstructions. Finalement, dans une dernière sous-partie, une technique permettant de détecter la silhouette de l'organe est elle aussi présentée. Cette dernière fait ainsi la distinction entre contours occultant (la silhouette), et contours d'occultation. Une telle distinction est importante car elle permet de différencier des contours permettant de contraindre le recalage (la silhouette) et ceux devant être ignorés (les contours occultés). Cette méthode permet ainsi d'extraire des contraintes fortes, utilisées dans notre méthode de recalage, entre le modèle et la vue cœlioscopique.

2.1 Introduction

Afin de résoudre le problème du recalage de modèle 3D de foie dans la scène cœlioscopique per-opératoire, différents problèmes relativement généraux doivent être résolus. Ce chapitre propose ainsi un aperçu des différentes techniques utilisées dans les chapitres principaux de ce manuscrit. Chacune d'elles est ainsi présentée, discutée, et les différentes publications en lien sont référencées. Pour les techniques peu explicitées dans la littérature mais dont la compréhension est capitale lorsque l'on souhaite répliquer les méthodes et résultats présentés dans ce document, une description plus détaillée de la technique en question est proposée. Ainsi, la méthode de réduction de l'espace des déformations, utilisée par chacun de nos algorithmes de recalage afin de permettre une résolution en un temps raisonnable, tout en ne gardant que des déformations lisses, est explicitée. D'autres problèmes complexes dont la résolution est importante pour le bon déroulement du *pipeline* de recalage sont eux aussi explicités. Ainsi une méthode permettant de détecter la présence d'erreurs dans les reconstructions est proposée dans ce chapitre. De même, la méthode de détection des silhouettes de l'organe, utilisée afin d'induire des contraintes sur le recalage, complémentaires aux contraintes de correspondance surface-à-surface entre reconstruction de la scène per-opératoire et modèle pré-opératoire, est elle aussi présentée. Ces méthodes correspondent à des travaux préparatoires, et sont ainsi exclues des chapitres principaux. Elles sont néanmoins présentées dans ce chapitre.

2.2 Construire le modèle de l'organe à partir d'imagerie 3D

Afin de recaler les images volumiques pré-opératoires de l'organe considéré sur la scène cœlioscopique per-opératoire, un modèle biomécanique de l'organe doit tout d'abord être créé. Ce dernier est créé à partir des images volumiques pré-opératoires et contient les informations sur les structures internes de l'organe. C'est lui qui est ainsi recalé à la scène cœlioscopique. Cette section propose d'expliciter en détail sa création.

2.2.1 Reconstruction du modèle géométrique

La première étape nécessaire à la construction du modèle mécanique est la localisation de l'organe et de ses structures internes dans les images volumiques provenant d'un scan pré-opératoire du patient (TDM ou IRM). Pour ce faire, différentes possibilités existent. La première est d'utiliser un prestataire externe pour réaliser cette tâche. Des sociétés spécialisées dans ce domaine, telles que Visible Patient [1] ont été créées au cours des dernières années et proposent un tel service. De plus, les segmentations proposées par ces entreprises étant certifiées médicalement, il est alors possible de les utiliser sans problème dans le *workflow* chirurgical. Lorsque nos expérimentations impliquaient des patients, cette solution fut alors retenue et les segmentations ainsi effectuées par Visible Patient. Une autre solution consiste à utiliser des méthodes entièrement automatiques, basées sur des techniques de *deep learning* [81, 13, 215, 107]. Néanmoins, bien que ces méthodes aient connu un intérêt grandissant ces dernières décennies, résultant en une fiabilité grandissante pour ces techniques, celles-ci ne sont pas encore totalement prêtes à être utilisées au sein d'applications cliniques. Une dernière solution est d'utiliser l'un des nombreux logiciels existants permettant une segmentation semi-automatique de l'organe [209, 149, 159, 220], et offrant ainsi une plus grande liberté et fiabilité vis-à-vis de la segmentation finale obtenue. Dans notre cas, nous utilisons, lorsque nos expérimentations sont réalisées sur des données animales, le logiciel MITK [2], logiciel libre et open-source. Ce logiciel n'est cependant pas certifié et son utilisation sur des données humaines doit donc en pratique être restreint. L'organe, ainsi que ses différentes structures internes telles que les vaisseaux sanguins et les tumeurs, sont alors segmentées dans les images.

1. https://www.visiblepatient.com
2. http://mitk.org

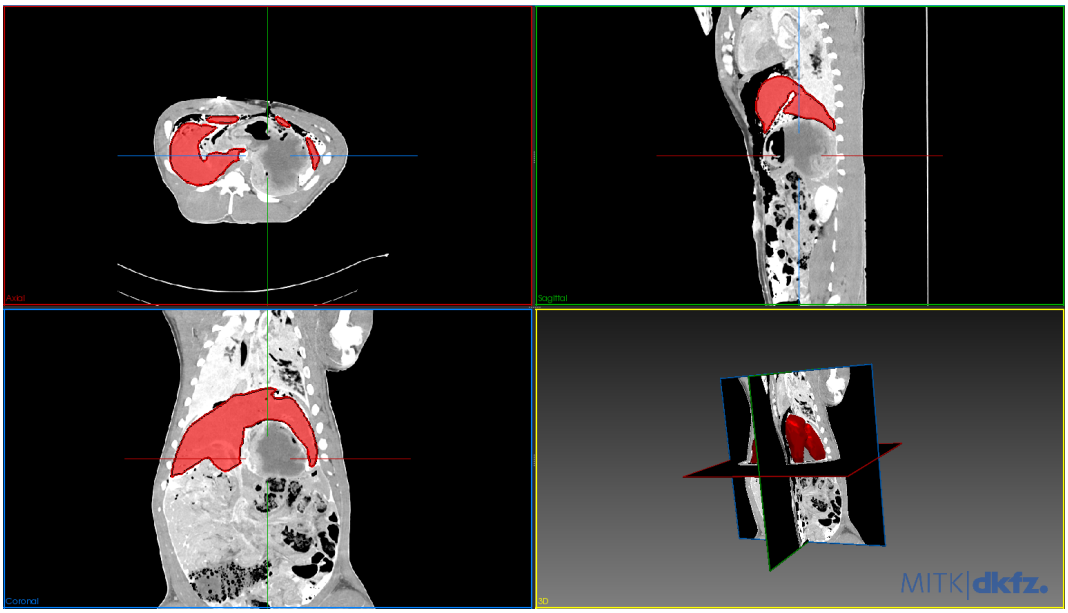

FIGURE 2.1 – Segmentation d'un foie porcin en utilisant MITK.

La surface de l'organe est ainsi repérée, et à partir de celle-ci, un maillage correspondant est construit en utilisant l'algorithme des marching cubes [112]. Le maillage de surface de l'organe comporte cependant un grand nombre d'irrégularités. Ce maillage doit être lissé dans un premier temps afin de supprimer différents artefacts dus à une segmentation imprécise et/ou un résultat de l'algorithme des marching cubes trop grossier. Pour cela, on utilise en général un algorithme de lissage de Taubin [185] : pour chaque point du maillage, on considère les n plus proches voisins de ce dernier et on met à jour sa valeur à la moyenne de ces points. Le maillage est ensuite corrigé afin de refermer d'éventuels trous dans ce dernier et de le rendre plus régulier [6]. De plus, d'éventuelles auto-intersections dans le maillage doivent, elles aussi, être supprimées [172, 213]. Ces corrections sont ainsi réalisées avec le logiciel Meshlab [3]. A partir de ce maillage de surface corrigé, une tétraèdrisation de Delaunay [19] est réalisée à l'aide du logiciel Gmsh [4] afin d'obtenir notre modèle géométrique de l'organe final sous la forme d'un maillage tétraédrique.

3. https://www.meshlab.net
4. https://gmsh.info

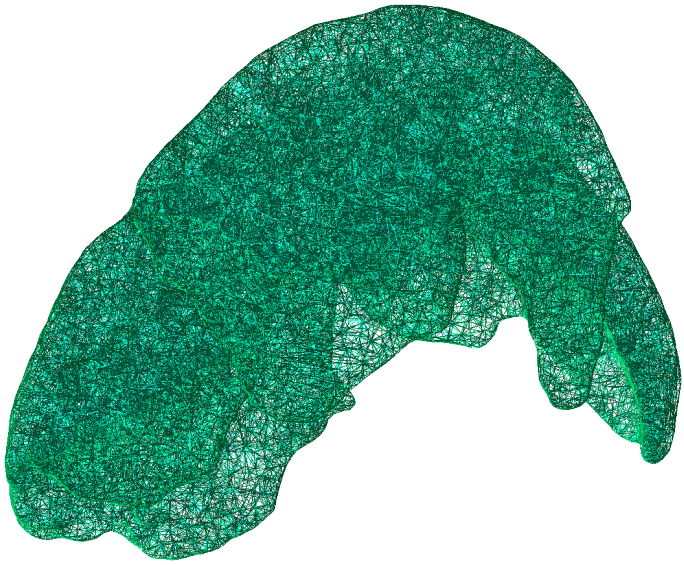

FIGURE 2.2 – Maillage volumique de foie porcin obtenu à partir de nos données pré-opératoires.

2.2.2 Construction du modèle biomécanique

A partir du modèle tétraédrique, le but est maintenant de construire un modèle biomécanique correspondant à la géométrie de l'organe extrait. Par modèle biomécanique, on entend modèle contenant de l'information renseignant sur la déformation courante de l'objet. Pour réaliser cette modélisation, la première étape consiste tout d'abord à choisir le type de comportement que l'on souhaite modéliser pour la déformation de l'organe. Plusieurs choix sont possibles. Si l'on considère que la réponse à la déformation est directement proportionnelle à la quantité de déformation en jeu alors on utilise une loi de comportement *élastique*. Cette considération n'est cependant réaliste que pour de faibles déformations. Si la déformation est trop importante, la réponse n'est plus linéaire en la quantité de déformation et on utilise plutôt une loi *hyper-élastique*. Pour les deux premières lois discutées, les déformations restent réversibles. Lorsque la déformation n'est plus réversible, on utilise un comportement dit *plastique*. Dans nos expérimentations, on se limite au cas des déformations *élastiques*. Les déformations considérées restent en effet relativement faibles. De plus, les travaux de cette thèse sont à considérer en tant que travaux préliminaires, se limitant à des modèles relative-

ment simples concernant la modélisation mécanique, quitte à les complexifier dans des travaux futurs. En effet, les algorithmes présentés dans ce manuscrit sont entièrement adaptables à tout types de modèles. Une fois la loi de comportement choisie, un modèle permettant de rendre compte de cette loi doit être choisi pour les déformations de l'organe. Nombre de ces modèles ont été proposés dans la littérature et sont de complexité variable. Certains sont ainsi très simples, tels le modèle *masse-ressort* [132] lorsque l'on considère un comportement *élastique*, tandis que d'autres sont bien plus complexes comme les modèles *Néo-Hokéens* [139, 30, 124, 25], les modèles de *Mooney-Rivling* [187], ou encore les modèles de *Fung* [179] lorsque l'on considère des comportements *hyper-élastiques*. Néanmoins, bien que certains de ces modèles complexes aient été testés pour des problèmes de recalage [124, 25, 30, 187], l'avantage réel apporté par leur utilisation, par rapport à des modèles plus simples, n'a jamais été réellement démontré. De plus, capturer précisément les caractéristiques physiques de l'organe est en pratique complexe à réaliser [166, 222, 68, 164, 144]. Enfin, la précision du recalage ne dépend pas uniquement de la modélisation physique des propriétés de l'organe, mais aussi de la qualité des données auxquelles on souhaite se recaler (avec notamment la non-reconstruction de certaines parties de la surface de l'organe ou du bruit dans ces données, pouvant induire des erreurs dans le recalage). Un compromis doit ainsi être trouvé dans la complexité du modèle biomécanique utilisé. Un modèle trop complexe, n'améliore en effet, après une certaine complexité déjà atteinte, que peu la précision du recalage final. Ainsi, une modélisation biomécanique simple de l'organe, en plus de permettre une résolution plus rapide grâce à la faible complexité engendré par celle-ci, est en général préférable.

Pour réaliser une telle modélisation, une méthode directe est de considérer la loi de Hooke, modélisant le comportement des solides élastiques [175]. On applique alors la Méthode des Éléments Finis (FEM) à cette loi sur les éléments unitaires de notre modèle géométrique [224, 136]. Ces éléments unitaires peuvent être les arêtes, les faces ou les tétraèdres eux-mêmes. Utiliser directement les tétraèdres en tant qu'éléments unitaires aboutit néanmoins à un modèle très complexe. Un tel modèle n'est utilisé en général que dans des simulateurs (tel abaqus [5]) se voulant très réalistes [211]. L'utilisation de tels modèles ne permet pas, lorsque l'on considère, pour le modèle, des déformations conséquentes, d'obtenir des résultats en temps réels, ni dans des temps adaptés à leur utilisation au sein du *workflow* chirurgical. L'utilisation de faces en tant qu'éléments à

5. https://www.3ds.com/fr/produits-et-services/simulia/produits/abaqus

considérer dans la FEM n'est quant à elle jamais utilisée dans ce contexte. L'utilisation des arêtes du modèle tétraédrique aboutit à des modèles 1D de type masse-ressort [132]. Bien que de tels modèles ont, de par leurs simplicités, des difficultés à rendre compte de certains phénomènes (comme la conservation du volume observé sur la plupart des tissus mous du vivant), leur utilisation permet néanmoins d'obtenir en général une bonne approximation du comportement d'organes tels que le foie. Dans un tel modèle, la masse de l'objet est répartie dans les différents nœuds du modèle et chaque arête se comporte comme un ressort ayant une certaine constante de raideur. La valeur de cette constante peut alors être la même pour le modèle entier. On parle alors de modèle homogène (à noter cependant que cela ne garanti pas un comportement homogène du modèle). Il est aussi possible d'utiliser les différentes segmentations des structures internes précédemment réalisées afin d'associer à chaque ressort une constante de raideur dépendant de la segmentation à laquelle il appartient. On parle alors d'un modèle hétérogène [69]. Une fois construit, ce modèle doit alors être recalé aux données per-opératoires. La façon dont ce recalage est réalisé est l'un des points principaux sur lequel cette thèse s'est focalisée. Les différentes méthodes étudiées sont ainsi développées en détail dans les chapitres 3 et 4.

2.2.3 Réduction de l'espace des déformations

Comme évoqué dans l'introduction, nous considérons dans nos travaux une méthode énergétique pour résoudre le recalage. Une fonction objectif, représentant l'état du recalage est ainsi créée à partir des différentes contraintes appliquées au modèle biomécanique (contraintes de correspondance avec la surface per-opératoire reconstruite, contraintes de correspondance avec les contours des images cœlioscopiques per-opératoires, contraintes sur la quantité de déformation acceptable afin que le recalage corresponde à une déformation réaliste d'un point de vue physique, et contraintes d'autres types). Cette fonction prend comme argument la déformation actuelle du modèle (c'est-à-dire la position de tous ses sommets), et retourne une valeur dans $\mathbb{R}^+$, représentant le respect de ces différentes contraintes au vue de la déformation actuelle. Plus cette valeur sera faible, mieux les contraintes seront respectées. Cette fonction non convexe doit ainsi être minimisée. Pour cela on utilise alors des méthodes numériques telles que l'algorithme de Gauss-Newton ou l'algorithme de Levenberg-Marquardt [17].

Néanmoins, si le modèle biomécanique est de grande résolution, alors le nombre d'inconnues du problème associé est élevé. La résolution d'un tel problème est alors coûteuse en temps. Afin d'éviter cela, l'espace de recherche doit alors être réduit. Différentes approches existent dans la littérature pour cela. La plupart de ces techniques proposent ainsi de considérer directement des paramétrisations du modèle adaptées ne représentant pas tous les états possibles de la déformation. Des paramétrisations plus ou moins complexes peuvent ainsi être utilisées : du modèle linéaire de déformation [165] à des modèles plus complexes [142, 146, 18]. De telles solutions nécessitent néanmoins une réécriture du problème d'optimisation considéré, en utilisant la paramétrisation choisie. Pour nos travaux, une solution permettant de s'affranchir d'une telle réécriture fut préférée, s'appuyant sur les travaux décrits dans [36]. Dans ces travaux, une forme quadratique semi définie positive est utilisée pour caractériser la régularité des déformations considérées. Une décomposition spectrale de la matrice associée à cette forme est ainsi utilisée afin de définir un sous-espace des déformations de faible intensité, c'est-à-dire des transformations "proches" de transformations localement affines. Le nombre de vecteurs de bases retenus pour ce sous-espace influe ainsi sur la complexité des déformations retenues. On souhaite ainsi résoudre le problème de minimisation non plus sur l'ensemble des déformations possibles, mais sur les décompositions possibles suivant cette base. La problème de minimisation et sa Jacobienne sont ainsi directement modifiés afin de correspondre à la recherche de déformations suivant cette base. Cette méthode est ainsi utilisée dans [36] afin de rendre temps réel un algorithme de recalage déformable. Cependant, dans cette publication, cette méthode n'est que très peu explicitée. Afin de permettre une meilleure ré-implémentation et reproductibilité de nos travaux, nous avons ainsi explicité et discuté cette approche dans l'Annexe A.

2.3 Reconstruire la surface de l'organe considéré à partir d'images cœlioscopiques

Avant de réaliser le recalage de notre modèle sur nos données cœlioscopiques, une étape supplémentaire est nécessaire (sauf dans le cas du recalage mono-image qui sera évoqué dans le chapitre 3). Cette étape consiste à organiser entre elles les différentes images cibles du recalage, c'est-à-dire à évaluer, pour chaque paire d'images, les posi-

tions relatives des caméras correspondant aux vues en question. Cela permet alors de pouvoir lier les contraintes du recalage associées à ces images de façon cohérente. Pour cela, on réalise, à partir de ces images une reconstruction de la surface observée comme présenté dans la Figure 2.3.

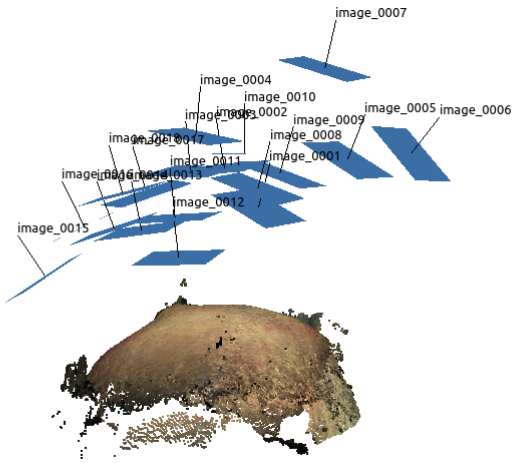

FIGURE 2.3 – Organisation des différentes images entre elles à l'aide d'une reconstruction de surface, ici provenant de photoscan.

Cette reconstruction se compose alors non seulement des points 3D reconstruits de la surface mais aussi d'un certain sous ensemble des images fournies, appelées images clés ou *keyframes*. Ces images sont sélectionnées automatiquement à partir d'un certain nombre de critères (absence de bruit, nombre important de points caractéristiques, pertinence des nouvelles informations qu'apporte l'image en question par rapport aux vues préalablement retenues). Pour recaler ces reconstructions, deux principaux types d'algorithmes existent : les algorithmes basés photométrie et les algorithmes basés sur des associations entre les images. Les techniques basées sur la photométrie se basent sur l'analyse du rayonnement lumineux dans la scène, et en particulier la façon dont celui-ci se comporte, depuis la source, en passant par la surface observée jusqu'à son arrivée sur le capteur de la caméra [79, 210, 104]. Cette analyse permet alors de déterminer la géométrie et la réflectance de l'objet observé, expliquant l'image obtenue par le capteur. Ce type de technique est décrit plus en détail dans les chapitres 5 et 6 de ce manuscrit. Le deuxième type de reconstruction se base sur des associations entre les images. Ces

associations peuvent alors être soit denses soit éparses. Pour des associations denses, on considérera pour des images voisines, directement des associations entre tous les pixels, en étudiant les différences entre les intensités colorimétriques entre ces derniers. Pour des associations éparses, seuls un certain nombre de points d'intérêts, aussi appelés points clés sont considérés dans les images, et associés entre eux.

2.3.1 Différentes techniques de reconstruction

Méthodes de reconstruction éparses : utilisation de points clés

Les points d'intérêt considérés sont des points se remarquant particulièrement dans l'image par rapport à d'autres, dont la spécificité est d'être invariants par rotation et translation. Pour plusieurs vues où un tel point est repéré, les informations définissant celui-ci en tant que point d'intérêt (sans prendre en compte sa position dans l'image) sont ainsi les mêmes dans les vues. Il est alors possible de repérer ce point et de le mettre en correspondance dans toutes les vues, créant ainsi des associations dans les images. Pour définir ces points, deux classes d'algorithmes sont utilisées : les descripteurs, qui permettent de définir en tout point de l'image un vecteur, représentation de celui-ci, et les détecteurs permettant de détecter dans les images des candidats pour les points caractéristiques à extraire. Bien qu'un certain nombre de ces algorithmes existe, leur fonctionnement est très similaire. Pour chaque point d'une image retenue par le détecteur, on considère dans une fenêtre autour de ce point les différents points présents. Une fonction, définie par le descripteur utilisé, prend alors en entrée ces points et calcule un vecteur représentant les informations du point considéré, relatif au descripteur que l'on utilise pour calculer nos associations. Une fonction de distance entre ces vecteurs représentatifs est aussi définie au sein du descripteur, permettant ainsi de comparer les points clés entre eux et les associer si cette distance est sous un certain seuil. Nombre de ces détecteurs et descripteurs ont été créés et étudiés au cours des dernières décennies. Parmi les plus connus et utilisés, on dénote l'algorithme Scale-Invariant Feature Transform (SIFT) [113] décrivant son détecteur et son descripteur associé, et l'algorithme Speeded Up Robust Features (SURF) [11], dont le détecteur est plus rapide à extraire les points clés que SIFT et dont le descripteur est plus robuste en terme de qualité de points extraits pour certaines images. On dénote aussi l'algorithme KAZE [3], étant libre et moins sensible aux bruits que les autres méthodes. Des méthodes mixtes

utilisant des détecteurs et descripteurs non associés initialement sont aussi proposées. Ainsi, récemment l'algorithme Oriented FAST and Rotated BRIEF (ORB) [162] a été proposé. Il s'agit d'une alternative gratuite d'utilisation à SIFT et SURF. Il possède des performances similaires à ces derniers tant sur le point rapidité que sur la qualité des points extraits. Ce dernier est basé sur un détecteur Features from Accelerated Segment Test (FAST) [160] et un descripteur Binary Robust Independent Elementary Features (BRIEF) [23].

Une première technique de reconstruction basée sur des points clés : le Structure from Motion (SfM)

Deux approches sont alors possibles pour obtenir notre reconstruction en fonction de la situation envisagée. La première de ces méthodes considère que la prise d'images a été réalisée une fois pour toutes, avant que la reconstruction n'ait lieu. Elle utilise ainsi toutes les images pour réaliser sa reconstruction. Elle n'a alors en général pas besoin d'être temps réel. Les algorithmes de ce type se nomment ainsi algorithmes SfM. Les points clés discutés dans la section précédente sont ainsi extraits des images que l'on souhaite utiliser en entrée de notre reconstruction et mis en correspondance dans les différentes images. On considère n images et k points clés uniques (c'est-à-dire ayant une signature différente pour le descripteur considéré), associés entre eux dans les différentes images et on note $Y_{i,j}$ le j-ième point clé, présent dans la i-ième image (image préalablement rectifiée, et donc supposée sans distorsion), exprimé en coordonnées homogènes et correspondant à l'image d'un point 3D X_k de position inconnue. Si l'on considère (R_i, T_i) la pose de caméra associée à la i-ième image, on note alors $\widetilde{P}_{i,j}$ la position en coordonnées homogènes de la projection du point X_j dans la i-ième image :

$$\widetilde{P}_{i,j} = K(R_i X_j + T_i), \tag{2.1}$$

avec K matrice de projection de la caméra. On note $P_{i,j}$ la projection de ce point en coordonnées cartésiennes (correspond au vecteur composé des deux premières lignes de $\widetilde{P}_{i,j}$ divisé par la troisième ligne de $\widetilde{P}_{i,j}$). La reconstruction correspond ainsi à un

problème de minimisation qui est le suivant :

$$\min_{R_i \in SO_3(\mathbb{R}), T_i \in \mathbb{R}^3, X_j \in \mathbb{R}^3} \sum_{i=1}^{n} \sum_{j=1}^{k} \delta_{i,j} \left\| P_{i,j} - Y_{i,j} \right\|^2 , \tag{2.2}$$

avec $\delta_{i,j}$ valant 1 si le point X_j est visible dans l'image i, et 0 sinon. On considère en général que le travail ayant introduit cette problématique, et étant à l'origine du SfM, est l'algorithme des 8 points [110], publié en 1981. Dans ce dernier, le problème à deux vues est évoqué et il est montré que 8 points en correspondance dans les deux images suffisent à récupérer la matrice fondamentale [115] F du système. Cette matrice fondamentale lie ainsi les positions Y et Y', en coordonnées homogènes, des points des images en correspondance présents dans les images, par la relation suivante :

$$Y'^{\top} \; F \; Y = 0 \tag{2.3}$$

Si la matrice des paramètres intrinsèques K de la caméra utilisée est connue (si la caméra est calibrée), la position des points dans le repère lié à chacune des deux vues (X et X'), en coordonnées images normalisées (c'est à dire la position des points 3D au facteur d'échelle près) ainsi que la matrice essentielle [110] E du système peuvent être obtenues. On a ainsi pour expression des points :

$$\begin{aligned} X &= K^{-1} Y \\ X' &= K^{-1} Y', \end{aligned} \tag{2.4}$$

avec Y et Y' exprimés en coordonnées homogènes, et pour expression de la matrice essentielle :

$$E = K^{\top} F K. \tag{2.5}$$

Cette matrice lie alors la position des points en coordonnées images normalisées X et X' dans les deux vues par la relation suivante :

$$Y'^{\top} \; F \; Y = 0. \tag{2.6}$$

Cette matrice essentielle peut alors être décomposée [110] pour obtenir la pose relative entre les deux vues. Pour ce qui est de la généralisation du problème du SfM à n vues

tel que nous le connaissons, considérant des caméras perspective de poses inconnues, il faut alors attendre 1996 pour qu'une solution soit proposée [177]. Parmi les approches modernes, on dénote deux types de méthodes pour le SfM : le SfM incrémental [167] et le SfM global [192, 41]. Pour le premier, on commence notre reconstruction à partir de deux paires. Pour cela, on calcule pour toutes les paires d'images celles ayant le plus grand nombre de points clés en commun et on initialise alors la reconstruction avec ces deux vues. On cherche ensuite la troisième image, possédant le plus de points clés en commun avec l'une deux images déjà dans la reconstruction, on calcule les deux poses possibles pour cette dernière, relatives aux deux autres images déjà présentes dans la reconstruction. On moyenne alors les poses trouvées et on initialise la pose de la nouvelle caméra à cette moyenne. On ré-optimise ensuite l'ensemble des poses avec un ajustement de faisceaux [196], problème non-algébrique (au contraire de la méthode des huits points) et non-convexe, dont le but est de minimiser l'erreur de re-projection de l'ensemble des points de la reconstruction dans les différentes images clés en fonction de l'ensemble des poses de caméra de la reconstruction. On itère ensuite la méthode en sélectionnant une quatrième image, puis une cinquième... tout en ré-optimisant les poses de caméra à chaque nouvelle image. Ce problème est cependant extrêmement coûteux en terme de quantité de calcul à effectuer (et donc aussi en temps d'exécution). Le SfM global quant à lui propose de ne faire qu'un seul ajustement de faisceaux [196] en considérant toutes les poses en même temps. Pour ce faire, on calcule, une nouvelle fois, une initialisation des poses relatives de caméra associées aux images, en considérant pour chaque image un certain nombre de paires d'images la contenant et ayant un nombre de points clés en commun avec celle-ci, dépassant un certain seuil (on ne calcule pas toutes les paires possibles, mais on tire plutôt des paires de façon aléatoire jusqu'à en obtenir suffisamment). On regroupe ensuite les poses entre elles, en utilisant les images communes entre celles-ci, jusqu'à obtenir un système de poses entièrement défini. On réalise finalement un ajustement de faisceaux sur l'ensemble des poses trouvées. Les principaux algorithmes de SfM utilisés dans cette thèse furent Theia [182] et celui présent dans le software Photoscan [6]. La première librairie a pour avantage d'être open-source, de n'avoir que très peu de dépendances, et de fournir des interfaces faciles d'utilisation, ce qui en fait une candidate de choix pour réaliser des tests préliminaires. Cependant, bien que les résultats obtenus par cette méthode soient

6. https://www.agisoft.com

assez robustes, les reconstructions finales ont tendance à être très peu denses. Photoscan de son côté propose une architecture fermée, mais puisqu'il est directement couplé à un algorithme Multi-View Stereo (MVS) permettant de densifier la surface à la sortie du SfM, il permet d'obtenir des reconstructions bien plus denses et d'une très grande précision.

Une seconde technique de reconstruction basée sur des points clés : le Visual Simultaneous Localization And Mapping (VSLAM)

Le second type d'algorithme de reconstruction se basant sur des points clés propose, au contraire, de reconstruire la surface observée de façon incrémentale, au fur et à mesure que les images sont capturées. On parle alors de VSLAM. Cette méthode fait partie des différents algorithmes de type Simultaneous Localization And Mapping (SLAM) développés initialement pour la robotique, dont l'origine remonte à 1990 [174]. Originellement, ces algorithmes de SLAM n'étaient pas basés sur la vision mais sur d'autres données capteurs tels que des sonars ou des capteurs lasers. Le principe de la méthode n'était alors pas basé sur des mises en correspondances de points 2D mais plutôt sur de l'utilisation plus directe de données 3D. Ce n'est que plus tard que fut réellement développé le principe de SLAM visuel (VSLAM), basé sur des images, tel que nous l'utilisons en vision par ordinateur [168, 42, 90, 99, 118]. Dans ces méthodes, la reconstruction est ainsi initialisée avec un nombre réduit d'images, puis on calcule la pose de caméra associée à la nouvelle image en calculant les correspondances entre les points clés dans la nouvelle image et ceux déjà présents dans la reconstruction. Ensuite, si cette nouvelle image ajoute un nombre de points suffisant à la reconstruction (si elle contribue assez à cette dernière), on l'ajoute à celle-ci et on met à jour le nuage de points en fonction. Les nouvelles poses de caméra sont ainsi toujours estimées en fonction des poses préalablement calculées. L'intérêt d'une telle approche est double. Tout d'abord, comme la reconstruction est calculée de façon incrémentale, le nombre d'inconnues de chaque minimisation considérée, à chaque étape de la reconstruction est bien plus réduit que lorsque l'on considère le problème dans son entièreté. De ce fait, le problème peut alors être facilement résolu en temps réel. Il est ainsi possible de réaliser la reconstruction pendant la capture des images et de voir la reconstruction se réaliser au fur et à mesure. Cela permet alors au manipulateur d'avoir un réel retour sur ce qu'il fait, et de lui indiquer quelle zone de la surface cible n'a pas encore été reconstruite, et doit ainsi

être ciblée par la caméra. Cependant, à cause de cette construction incrémentale, un problème peut subvenir dans la reconstruction. En effet, comme les poses de caméra précédemment calculées sont utilisées pour le calcul de la pose courante, les erreurs au niveau de l'estimation de ces poses s'accumulent au cours du temps. Cela génère alors non seulement des erreurs dans les poses des images clés de la reconstruction mais aussi dans le nuage de points reconstruit, où de mauvais alignements aussi appelés dérives apparaissent. Dans les premiers travaux s'attaquant à la problématique du VSLAM, il s'agissait ici d'une véritable limitation de la méthode. Néanmoins, à partir des années 2000, une organisation des images clés de la reconstruction sous la forme d'un graphe et une méthode d'optimisation de ce graphe lorsque certains évènements sont détectés, est proposée afin de résoudre en grande partie (où tout du moins de fortement diminuer) ce problème [94, 180]. Un graphe des images clés est ainsi créé, reliant celles-ci spatialement et temporellement, en utilisant l'ordre de création de ces dernières. On relie ainsi chaque nouvelle image clé utilisée à celle ayant été créée juste avant celle-ci. Non seulement, elle est alors proche en terme de temporalité (son moment de création n'est que peu différent, bien qu'antérieur à la *keyframe* nouvellement créée), mais aussi en terme de spatialité car le mouvement de la caméra est continu dans l'espace. Le but est alors de repérer et de créer des boucles en terme de spatialité dans ce graphe, et d'optimiser les poses de caméras à l'intérieur de ces boucles lorsque celles-ci sont détectées à l'aide d'un ajustement de faisceaux. L'algorithme est alors séparé en deux taches s'exécutant en parallèle. La première tâche est la tâche principale du SLAM classique : extraire les points clés de la nouvelle image, la mettre en correspondance avec les images clés appartenant déjà à la reconstruction, calculer la pose associée, et ajouter ou non cette image à la reconstruction en fonction de la présence ou non d'informations nouvelles dans cette dernière (et une fois l'image ajoutée, mettre à jour les points de la reconstruction ainsi que le graphe des images clés). La seconde tâche est de détecter les boucles au sein du graphe et de corriger le graphe en fonction. Pour chaque nouvelle image ajoutée au graphe, on regarde si cette nouvelle image n'est pas proche spatialement d'une autre image clé ajoutée précédemment (autre que les dernières images ajoutés avant celle-ci). Pour cela on teste le nombre de mêmes points clés dans cette image et dans les images clés précédentes. Si le seuil de points clés communs entre deux images est suffisant, les deux images sont ainsi proches l'une de l'autre : on crée alors une boucle dans le graphe et à l'intérieur de cette boucle, et on ré-estime alors toutes les poses

à l'intérieur de celle-ci. Cela permet ainsi de corriger ces même poses et de supprimer une dérive éventuelle. Les autres poses du graphe sont alors corrigées en fonction des nouvelles poses dans la boucle (en utilisant les poses relatives entre les vues, préalablement calculées). Un autre algorithme de SLAM récent, méritant d'être mentionné ici, est le LSD-SLAM [50], ayant pour particularité de ne pas utiliser des points clés mais toute l'information présente dans les images. Le principal algorithme de VSLAM utilisé dans différents travaux de cette thèse fut ORBSLAM2 [135], utilisant des descripteurs ORB. Ce dernier possède l'avantage d'être open-source, et dispose d'une détection de boucle efficace qui lui permet d'avoir de très bon résultats avec peu de dérives grâce à son ajustement de faisceaux. Il utilise aussi une technique de stockage des descripteurs de ses points clés sous la forme de sac de mots [60], permettant de calculer les correspondances plus rapidement. Il est aussi disponible avec différents modes correspondant à différents types de données d'entrée, changeant quelque peu son comportement mais offrant toujours de très bonnes reconstructions : le mode monoculaire (mode classique, avec une seule caméra), et le mode stéréo.

Les inconvénients des méthodes de reconstruction monoculaires basées points clés et leurs résolutions

Les deux approches de reconstruction présentées précédemment présentent cependant quelques inconvénients majeurs à surmonter avant de pouvoir les utiliser pour des applications d'aide à la chirurgie cœlioscopique [199, 133, 117]. La première est commune à toute méthode de reconstruction monoculaire (utilisant une seule caméra), à savoir une ambiguïté sur l'échelle de la reconstruction obtenue. En effet, sans aucune contrainte à priori entre plusieurs poses de caméra, ni aucun objet de taille connue dans la scène, il existe toujours une ambiguïté d'échelle dans la reconstruction. En effet, il est impossible de savoir si l'on observe un petit objet de près ou un objet plus important de loin. Le nuage reconstruit est ainsi obtenu à un facteur d'échelle près, facteur qui se retrouve aussi dans toutes les translations entre les poses de caméra estimées. Plusieurs possibilités existent pour lever cette ambiguïté. La première est de filmer un marqueur visuel dont la taille est connue dans la scène [44]. Cela permet alors directement d'estimer le facteur d'échelle de la reconstruction à partir de la reconstruction du marqueur. Cette solution n'est néanmoins pas envisageable dans le cas de la chirurgie laparoscopique où cela nécessiterait d'introduire un tel marqueur dans le corps du patient, ce qui n'est

évidemment pas souhaitable. Les autres solutions envisagées pour répondre à ce problème nécessitent l'utilisation d'un endoscope stéréo. Un tel cœlioscope comporte ainsi deux caméra fixes l'une par rapport à l'autre dont la position relative peut être déterminé à l'aide d'une phase préalable de calibrage. À chaque instant t, on ne récupère ainsi non plus une seule image, mais une paire d'images correspondant à une observation de la scène sous deux points de vues légèrement différents. Une seconde solution pour résoudre cette ambiguïté d'échelle consiste alors à forcer, dans l'algorithme de reconstruction, les poses relatives entre les caméras associées aux paires d'images en question [135, 193]. Cela permet ainsi de fixer le rapport d'échelle à 1. Cette méthode n'est cependant pas implémentée dans de nombreux softwares de reconstruction. Une troisième méthode est possible pour récupérer l'échelle de la reconstruction de la scène cœlioscopique, à l'aide d'un cœlioscope stéréo, en utilisant cette fois-ci une méthode de reconstruction pensée pour le monoculaire. On choisit tout d'abord une caméra de référence (gauche ou droite). Ensuite, pour chaque paire d'images, on calcule une carte de profondeur en utilisant un algorithme stéréo [16, 76]. On donne ensuite à l'entrée d'un algorithme de reconstruction monoculaire (SfM ou VSLAM) l'ensemble des images correspondant à la caméra de référence. La reconstruction en sortie comporte donc une échelle. Pour chaque image clé de la reconstruction, on peut alors reprojeter le nuage de points reconstruit dans le plan image correspondant, et pour chaque point ainsi reprojeté, comparer la valeur de la profondeur correspondante du point dans le plan image et la profondeur obtenue grâce à l'algorithme stéréo pour la paire d'image associée. Le rapport de ces deux valeurs nous donne ainsi le facteur d'échelle attendu. Pour rendre cette estimation plus robuste, on considère ce rapport pour tous les points de la reconstruction projetés dans toutes les images clés. On retient ensuite la médiane de toutes ces valeurs en tant que facteur d'échelle de la reconstruction.

2.3.2 Détection d'erreur de reconstruction par validation croisée stéréo

Les reconstructions de surface obtenues avec des méthodes telles que celles décrites précédemment sont toujours sujettes à des erreurs importantes principalement dues à de mauvaises estimations de poses de caméra. Cela peut ainsi être problématique lorsque de telles reconstructions ont pour vocation d'être utilisées en tant que cible du recalage.

Les erreurs de reconstruction se répercutent ainsi sur la précision de celui-ci. Pour des recalages présents au sein d'applications à risque telles que de l'aide à la chirurgie, il est alors impératif de détecter ces erreurs afin de pouvoir les supprimer, ou même lorsque cela est possible, les corriger. Dans notre cas, le cœlioscope que nous considérons pour nos expérimentation est un cœlioscope stéréo. La méthode de détection d'erreur dans les reconstructions que nous utilisons est alors spécifiquement adaptée à ce type de matériel. Elle présente l'intérêt d'être relativement simple. Son principe consiste à vérifier la cohérence de notre surface reconstruite à tester, avec des reconstructions obtenues pour chaque vue, à l'aide d'un algorithme de reconstruction stéréo : on parlera de validation croisée stéréo. On considère donc comme reconstruction à tester des reconstructions obtenues à l'aide de méthodes monoculaires pour lesquelles seule l'une des deux caméras du cœlioscope en question est à l'origine des images d'entrée de l'algorithme utilisé (SfM ou VSLAM). La reconstruction générée consiste alors en un ensemble de points ayant étés reconstruits et des vues associées à cette reconstruction, appelées vues clés. Pour chacune de ces vues, la pose de caméra de l'image correspondante à cette vue est ainsi estimée. Comme expliqué dans la sous-section précédente, on peut ensuite utiliser chaque image clé de la reconstruction générée, ainsi que son image clé associée (l'image étant capturée au même instant par la seconde caméra du cœlioscope) afin de générer une carte de profondeur pour la vue en question en utilisant un algorithme de reconstruction stéréo. On suppose ici, comme pour la section précédente, que la caméra a été préalablement calibrée et que le changement de repère entre la caméra droite et la caméra gauche a été estimé. On projette alors chaque point de la reconstruction dans chacune de ces vues clés, et on calcule le rapport entre la profondeur associée à notre nuage de point et la profondeur associée à la reconstruction stéréo, correspondant à des valeurs candidates pour le facteur d'échelle de la reconstruction monoculaire testé. Contrairement à la méthode décrite dans la section précédente, considérant directement l'extraction de ce facteur d'échelle en question, on ne considère plus maintenant uniquement la médiane de ces valeurs mais on s'intéresse à la distribution de l'entièreté de celles-ci. Comme précisé dans la section précédente, si le bruit dans les images était nul et que les reconstructions étaient parfaites (la reconstruction SfM ou VSLAM testée mais aussi les différentes reconstructions stéréo calculées pour chaque paire d'image), alors toutes les valeurs calculées pour ces rapports de profondeur correspondraient à une seule et unique valeur : le facteur d'échelle en question. Cependant, dans le cas réel, et

pour des reconstructions sans erreurs importantes (autre que du bruit), ces valeurs ont tendance à toutes être différentes, bien que très proches d'une valeur unique correspondant au facteur recherché. Lorsque l'on regarde alors la distribution de ces valeurs, on observe alors un unique mode dans celle-ci. La variance associée à ce mode unique est alors assimilable à la quantité globale de bruit dans la reconstruction. Si cette variance est trop importante, il convient alors de rejeter le résultat. De plus, des erreurs bien plus importantes dans la reconstruction dues à une dérive progressive des poses de caméra estimées, sont elles aussi repérables dans cette distribution : elles correspondent à des modes secondaires. Les points associés à ces modes sont ainsi facilement repérables dans le nuage de points, et retirés de la reconstruction. Si la reconstruction provient d'une méthode de type VSLAM, et si cette méthode de vérification et suppression est réalisée sur un thread parallèle, il est alors possible de donner à l'utilisateur un retour concernant cette partie de la surface qui a besoin d'être recalculée à l'aide de nouvelles données. Cette méthode est explicitée plus en détail dans l'Annexe B.

2.4 Détection de la silhouette de l'organe

Les reconstructions de la surface observée décrites précédemment, après remise à l'échelle et éventuelles corrections, servent ainsi à ordonner nos images entre elles. En effet, non seulement un nuage de points correspondant à la surface reconstruite est accessible, pouvant servir à contraindre le recalage d'une première façon, mais les différentes poses de caméra exprimées dans le système de coordonnées lié à ce même nuage de points, sont elles aussi disponibles. Cela permet ainsi d'avoir, pour chaque paire de vues, les déplacements relatifs des caméras. Lorsque l'on souhaite ainsi réaliser un recalage de données préopératoires sur ces images, il est alors possible, à l'aide des poses calculées entre les vues correspondantes, de lier toutes les contraintes associées à celles-ci, et ce de façon cohérente. En effet, si par exemple, des points de repère sont localisés dans une image et associés avec une zone de notre modèle, alors il est possible de retrouver ces mêmes points dans d'autres images (si ils ne sont pas occultés) et d'affiner leur position 3D, et donc d'affiner les contraintes associées avec le modèle. D'une façon similaire, il est aussi possible d'associer les contraintes correspondantes aux contours de l'organe sur les images. En effet, chaque contour dans une image va correspondre à un ensemble de contraintes, sur une ou plusieurs lignes à la surface du modèle à recaler,

en plus des contraintes directes 3D/3D entre la surface reconstruite et le modèle. En connaissant les poses de caméra entre les vues, il est alors aussi possible d'associer ces contours à plusieurs ensembles de lignes sur le modèle, tout en restant cohérent vis-à-vis des images observées, et d'utiliser ces lignes en tant que contraintes additionnelles.

2.4.1 Différencier contours occultants et d'occultation

Les contours de l'organe doivent ainsi être détectés dans les images afin de mieux contraindre le recalage. Ainsi, il est à noter que ces contraintes sont complémentaires des contraintes de correspondance surface-à-surface entre reconstruction de la scène cœlioscopique et modèle de l'organe. En effet, la reconstruction de surface apporte, en général, des contraintes localisées au niveau du centre de l'organe, tandis que les contours engendrent plutôt des contraintes sur les extrémités de l'organe, au niveau des zones n'ayant pas étés reconstruites. Cependant, il est à noter que tous les contours ne sont pas à prendre en compte. En effet, seuls des contours correspondant réellement à des lignes sur notre modèle doivent être considérés. Il faut ainsi bien distinguer la silhouette de l'organe qui nous servira à contraindre le recalage, et que l'on appellera aussi les contours occultants de l'organe, des contours occultés correspondant aux occultations de l'organe par d'autres tissus ou par des outils.

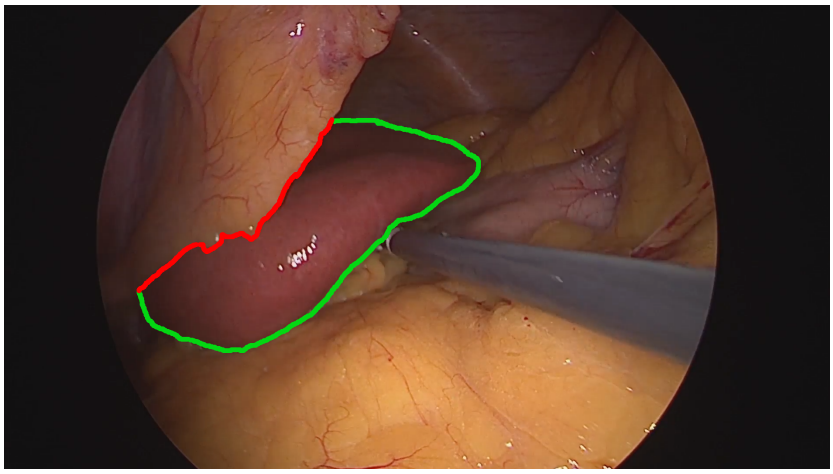

FIGURE 2.4 – Illustration des différents contours de l'organe : en vert la silhouette et en rouge les contours occultants.

2.4.2 Détection de contours

Dans la littérature, différentes techniques existent pour détecter les contours. Initialement, ces techniques étaient basées sur de la détection de zones de forts gradients, sur l'étude des distributions des couleurs, textures et illuminations locales autour de ces zones, ainsi que sur le tri des points retenus en fonction de différentes heuristques [181, 186, 127, 121]. Néanmoins, dans la plupart des publications récentes, des méthodes basées sur du *deep learning* [56, 71, 216] sont maintenant préférées. Dans notre cas, cette détection ne fait pas partie des axes majeurs de recherche étudiés dans la thèse. De ce fait, une solution plus simple n'utilisant pas de *deep learning* fut préférée. L'idée générale est ainsi de récupérer tout d'abord tous les contours de l'organe, puis de supprimer les contours occultés en s'appuyant sur l'information de profondeur obtenue grâce au cœlioscope stéréo. Pour détecter l'ensemble de tous les contours, on procède en deux étapes. Tout d'abord, nous apprenons un modèle de couleur de l'organe considéré ainsi qu'un autre de l'arrière plan de la scène laparoscopique. Les modèles utilisés sont basés sur des mélanges de gaussiennes aussi appelées Gaussian Mixture Model (GMM) [156]. Pour l'apprentissage, on demande à l'utilisateur d'annoter un certain nombre d'images de foies provenant, de préférence, d'un grand nombre de patients différents, en labellisant l'organe et l'arrière plan. Cette étape peut être réalisée avec des données ne provenant pas de l'utilisateur en question et être donc réalisée au préalable. Les images sont ensuite converties dans l'espace chromatique CIELAB, connu pour permettre de mieux caractériser la couleur de la surface observée [61]. On étudie ensuite la distribution des couleurs des différents pixels pour toutes les images. Cette fonction de distribution est ainsi une fonction de $[\![0, 255]\!]^3$ dans $\mathbb{N}$. On interprète ainsi ces modèles comme des mélanges de gaussiennes et on récupère les n modes associés aux plus grands maxima locaux. Les différents modes de telles distributions correspondent ainsi à des ellipsoïdes. Chaque modèle de couleur (un pour l'organe et un pour l'arrière plan) est ainsi associé à n modes correspondants à une valeur LAB (provenant de la conversion de l'image au format colorimétrique CIELAB), une variance suivant L, une autre suivant A et une dernière suivant B. On utilise ensuite ce modèle entraîné avec les images dans lesquelles on souhaite détecter nos contours, pour initialiser un algorithme *GrabCut* [161]. Pour cela on regarde tout d'abord pour chaque pixel de l'image si ce dernier appartient à l'une des deux classes apprises (organe ou arrière plan). Pour cela il suffit de regarder s'il appartient à l'union des ellipsoïdes du modèle considéré. Cela définit

alors sur l'image différentes zones de plus ou moins grande taille dans lesquelles tous les pixels appartiennent à la classe correspondante. On récupère ainsi, pour chacune des deux classes, les zones de plus grande taille en terme de superficie dans l'image, et on utilise celles-ci comme entrée de l'algorithme *GrabCut* qui nous donne alors en sortie une segmentation raffinée de l'organe. Les bordures de cette segmentation nous servent alors à définir l'ensemble des contours de l'organe dans l'image. Pour supprimer les contours occultant, on utilise alors l'information stéréo provenant du cœlioscope de la façon suivante. Pour chacune des images utilisées on considère l'image associée provenant de l'autre caméra et à l'aide d'un algorithme stéréo (dans nos expérimentation, on utilise l'algorithme de semi global matching implémenté dans OpenCV [77]), on récupère tout d'abord une carte de profondeur correspondant à la surface observée. On étudie ensuite, pour chaque point des contours observés, les zones situées des deux côtés de ce contour. Pour cela, on prend les tangentes aux contours en ces points et on regarde dans les deux directions orthogonales aux contours (l'une vers l'intérieur de la segmentation, et l'autre vers l'extérieur) les pixels appartenant à cette orthogonale. On considère alors n pixels dans chacune de ces directions (n choisi ni trop petit au risque de ne pas avoir assez de points, ni trop grand afin de ne pas considérer des points trop loin du contour), et on regarde la valeur médiane de la profondeur associée à ces derniers. Si la valeur à l'intérieur de la segmentation est supérieure à celle à l'extérieur, l'organe est au dessus de l'arrière-plan au niveau du pixel considéré et donc ce pixel correspond à un contour occultant. Dans le cas contraire, c'est l'arrière-plan qui se retrouve devant l'organe, et le contour associé est donc un contour occulté. De cette façon on est alors capable de ne garder que les contours occultés qui pourront alors être utilisés pour contraindre le recalage de l'organe.

2.5 Conclusion

Nous avons présenté dans ce chapitre différents prérequis nécessaires à la bonne compréhension des chapitres suivants ainsi qu'un certain nombre d'outils utilisés. Nous avons ainsi décrit plusieurs étapes préliminaires nécessaires aux algorithmes de recalage discutés dans ce manuscrit. Parmi ceux-ci, on dénote la construction de modèles mécaniques à partir d'images volumiques préopératoires, la reconstruction de surface per-opératoire cible du recalage, et la détection de la silhouette de l'organe dans les

images utilisées. Par ailleurs, une technique permettant de réduire l'espace de recherche pour les déformations impliquées dans le recalage, et permettant ainsi de réduire drastiquement le temps d'exécution de l'algorithme tout en ne gardant que les déformations réalistes, fut elle aussi discutée.

Chapitre 3

Recalage initial de modèles hépatiques pour la cœlioscopie augmentée et jeu de données d'évaluation

Nous nous intéressons dans ce chapitre au problème de recalage de modèles tétraédriques de foie à des images 2D de ce dernier, capturées à l'aide d'un cœlioscope. Le modèle à recaler correspond à une configuration pré-opératoire de l'organe et est construit à partir d'images obtenues par TDM ou par IRM. Les images 2D auxquelles on souhaite se recaler, proviennent quant à elles d'une vidéo d'exploration réalisée au début de l'intervention chirurgicale. Elles correspondent ainsi à une configuration per-opératoire. Le modèle doit ainsi être déformé et positionné dans la scène afin de correspondre aux images. Afin de simplifier le problème, nous ne considérons pas dans nos travaux les images elles-mêmes en tant que cible du recalage, mais une reconstruction de la surface observée obtenue par SfM, à partir de celles-ci. Pour le recalage en lui-même, une approche énergétique est proposée. Une énergie de minimisation, composée de différents termes est alors définie. Ces termes permettent de rendre compte des différentes contraintes s'appliquant sur le modèle telles que la correspondance du modèle à la surface et la quantité de déformation du modèle se devant de rester faible. Pour des modèles à géométrie non-convexe, les collisions pouvant subvenir au sein du modèle doivent de plus être prises en compte et empêchées lorsque l'on utilise des modèles mécaniques simples, tels que celui que nous utilisons. Afin d'évaluer l'algorithme de recalage que nous proposons, une base de données composée de données porcines *in vivo* a été créée. Celle-ci peut aussi être utilisée pour évaluer la précision d'autres recalages, et ainsi permettre une véritable comparaison des différents algorithmes, a été réalisée. Cette base de données a ainsi été rendue publique. Cette dernière est composée de deux modèles tétraédriques de foie porcin (l'un réalisé avec les lobes séparés et un autre avec les lobes fusionnés), correspondant à l'état pré-opératoire, et de différentes configurations per-opératoires à utiliser pour le recalage. Chacune de ces configurations comporte une vidéo d'exploration ainsi que des reconstructions de la surface associée. Différents marqueurs métalliques (correspondant à des sphères injectées dans les différents lobes et des clips fixés à la surface des lobes) sont repérés dans les différentes configurations à l'aide d'imagerie TDM interventionnelle, et sont utilisés pour calculer l'erreur de notre recalage en chacun de ces points. Cette base de données est la première de ce type, permettant une évaluation de l'erreur réelle du recalage (et non uniquement de l'erreur résiduelle correspondant à la distance surface-à-surface entre modèle recalé et vérité terrain) tout en considérant des données *in vivo*. Utilisant cette base de données, nous montrons ainsi l'importance de la prise en compte des collisions lorsque l'on considère de fortes déformations de l'organe et des modèles d'organes complexes (non-convexes). Ce chapitre est basé sur notre contribution [131] publiée dans IJCARS pour la conférence internationale IPCAI 2019.

3.1 Introduction

Problème général

Pour des applications telles que la cœlioscopie augmentée, des images 3D préopératoires de l'organe considéré, obtenues par TDM ou IRM, et contenant des informations sur ses structures internes (vaisseaux sanguins ou tumeurs), doivent être recalées aux images 2D de la scène opératoire provenant du cœlioscope. Ces images 3D ne sont en général pas recalées directement. Un modèle biomécanique (ou un modèle purement géométrique si aucune déformation de ce dernier n'est considérée) est ainsi généralement créé à partir de ces images préopératoires, comme décrit dans le chapitre 2, et celui-ci est alors recalé. Différentes approches existent dans la littérature pour résoudre ce problème : le recalage rigide manuel, automatique ou semi-automatique et le recalage déformable automatique ou semi-automatique.

Différents types de recalage

Lorsqu'une capture d'images 3D est possible en per-opératoire [91], les déformations en jeu entre les données volumiques et les données cœlioscopique sont alors très faibles. Des recalages rigides sont alors envisageables [138]. Une solution manuelle, simple à mettre en place pour des recalages rigides, peut alors être utilisée. Cependant, une telle méthode demande du temps et du personnel spécialisé en anatomie de l'organe que l'on souhaite recaler. Une solution automatique ou semi-automatique est souvent préférable [98, 131, 70]. Un recalage rigide peut, de plus, pour des organes peu déformables comme l'utérus, être utilisé à la suite d'un recalage déformable initial afin de suivre l'organe déjà déformé dans la scène. Dans ce cas, on suppose alors que l'organe en question n'est déformé principalement que par le pneumopéritoine (correspondant à insufflation dans la cavité abdominale) ainsi que par l'évolution de la maladie entre la capture des images IRM ou TDM et le jour de l'intervention. Les manipulations de l'organe lors de la chirurgie peuvent alors être considérées comme des déplacements rigides [38]. En revanche, le recalage initial entre le modèle pré-opératoire et la scène pré-opératoire doit être déformable. De plus, pour des organes très élastiques tels que le foie, un recalage rigide n'est jamais réellement possible car aucune déformation ne peut être totalement négligée (hormis dans le cas où l'on réalise un premier recalage déformable des données pré-opératoire dans la scène cœlioscopique, et que l'on considère

ensuite une simple exploration à vue diagnostique, sans aucune manipulation et pendant laquelle le recalage peut ainsi être considéré comme rigide). Dans tous les cas, le modèle pré-opératoire doit être déformé afin de correspondre à la configuration observée dans la scène per-opératoire. Quel que soit le type de recalage envisagé (rigide ou déformable), deux stratégies peuvent être utilisées pour sa résolution. Pour des recalages rigides ou lorsque l'aspect de l'organe déformé (per-opératoire) est proche de sa configuration préopératoire, une seule image peut être utilisée : on parle alors de recalage mono-image [98]. Une telle méthode peut être utilisée pour des organes quasi-rigides et lorsque l'on ne considère pas de manipulations importantes sur ces derniers (lors de prises de vue à visée diagnostique ou lorsque l'on souhaite planifier l'intervention par exemple). Pour des déformations plus importantes, on utilisera en général plusieurs images et on parlera alors de recalage multi-images [131]. En effet, lorsque plusieurs images sont utilisées, il est alors possible de réaliser une reconstruction de la surface per-opératoire et de l'utiliser en tant que cible indirecte du recalage. Le problème sera alors généralement décomposé en deux étapes [69, 36]. Tout d'abord, une vidéo est prise au début de l'intervention où la seule déformation à considérer est celle induite par l'insufflation (le patient est mis en apnée durant cette étape afin de pouvoir négliger les autres sources de déformation), et pendant laquelle le chirurgien essaie de voir le plus possible la surface de l'organe considéré. Bien que la seule déformation visible dans cette vidéo provienne de l'insufflation, elle ne peut pas être négligée pour des organes très élastiques tels que le foie [225]. Un recalage déformable doit donc être envisagé. Un recalage initial est ainsi effectué par rapport à ce premier ensemble d'images. Une fois ce recalage obtenu, il devient alors possible de texturer le modèle de l'organe et d'utiliser cette information de texture pour retrouver des déformations plus complexes (telles que celles induites par des manipulations avec des outils) dans la suite de l'intervention chirurgicale [35]. Seules des contributions concernant le recalage initial ont été réalisées durant cette thèse. Le reste est présenté comme potentiels travaux futurs.

3.2 État de l'art

Pour résoudre ces problèmes de recalage, plusieurs possibilités existent dans la littérature. La première est d'utiliser une simulation physique réaliste du modèle pré-opératoire et de toutes les forces s'exerçant sur lui [5, 10]. Le problème de telles réso-

lutions provient du fait qu'elles s'appuient sur une estimation de forces, qui elle aussi est difficile à réaliser correctement en pratique. Une seconde solution est d'utiliser, encore une fois une simulation des forces internes de l'organe mais d'exprimer cette fois ci les contraintes à appliquer sur l'organe non plus sous la forme de forces extérieures mais sous la forme de contraintes imposées aux nœuds du maillage [150]. Cette solution nécessite néanmoins une estimation relativement précise des forces s'exerçant dans l'organe lors de déformations. Une dernière solution est de considérer une approche énergétique [131, 128, 36, 7]. Une fonction d'énergie, composée de différents termes dont une partie représente la quantité de déformation appliquée au modèle, et une autre correspond à la vraisemblance entre l'état du recalage et les données d'observations, est alors considérée. Le principe est alors de minimiser cette énergie afin de trouver un recalage correspondant correctement aux données et dont la déformation n'est pas trop importante (la déformation étant limitée par la physique de l'objet). Chacune de ces techniques utilise un modèle biomécanique à recaler, modélisant les déformations de l'organe. Néanmoins, une approche basée sur la simulation physique nécessite que toute contrainte sur le modèle s'exprime comme une force s'appliquant sur ce dernier, ou comme une contrainte directe sur la position des points du modèle. Les approches énergétiques quant à elles sont bien plus libres concernant l'écriture de contraintes.

3.2.1 Recalage initial mono-image

Le premier type de recalage initial envisageable consiste ainsi à utiliser une seule image. Il peut être utilisé lorsque l'on considère des organes peu déformables. À partir de cette image, différents indices tels que certaines structures visibles, ou encore les contours de l'objet, sont utilisés pour contraindre le recalage [1, 98, 70]. Cependant, un problème apparaît avec une telle méthode. Bien que le recalage puisse être correctement contraint dans le cas général, il est cependant extrêmement sensible aux erreurs dans les extractions des contours utilisés. Prenons par exemple le cas des contours et illustrons le problème dans la Figure 3.1.

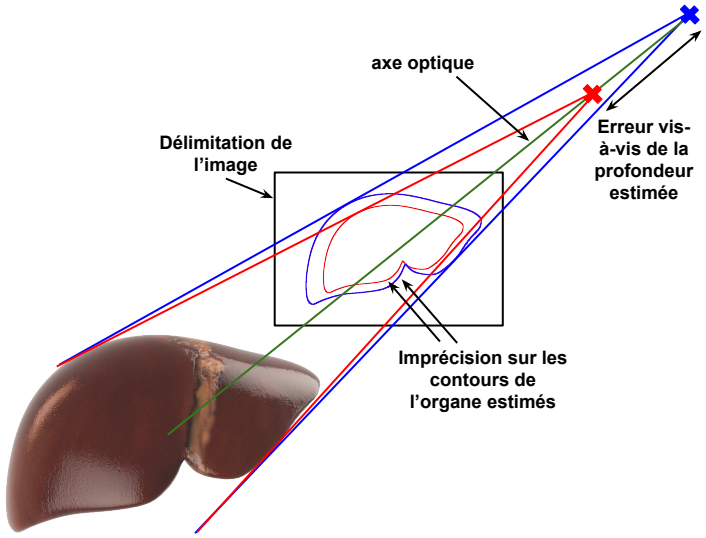

FIGURE 3.1 – Imprécision sur l'extraction des contours de l'image menant à une erreur d'estimation de profondeur.

La projection des bords du modèle correspond aux contours observés. Supposons que les contours soient détectés plus larges que dans la réalité. On observerait alors une erreur importante sur le recalage, et plus précisément sur la profondeur estimée. Un cas simple en 2D est présenté en Figure 3.2.

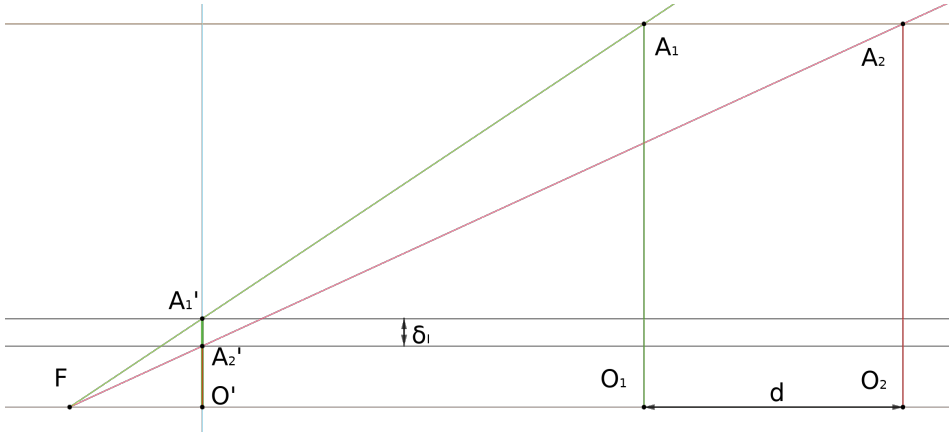

FIGURE 3.2 – Erreur sur la profondeur estimée due à la mauvaise estimation des contours : un cas simple en 2D.

Dans notre schéma, on note F le centre optique de la caméra, $O'A'_2$ et $O'A'_1$ les images de l'objet considéré (avec $A'_2A'_1$ la différence de détection au niveau du contour considéré) et O_1A_1 et O_2A_2 l'estimation de la position des objets associés. L'erreur dans l'estimation de la profondeur est ainsi égale à $A_1A_2 = O_1O_2 = d$ et est engendrée par une erreur d'estimation dans le contour égale à $A'_2A'_1 = \delta_I$ dans le plan image. Si la taille du capteur de la caméra est t_c et que la résolution de celui-ci est de n_{pixels} suivant cette dimension, alors on a $\delta_p = \frac{\delta_i \ n_{pixels}}{t_c}$, avec δ_p l'erreur en pixels lors de l'extraction de contours. On peut alors facilement exprimer d. Notons alors $A_1O_1 = A_2O_2 = L$ la taille de l'objet, $FO' = f$ la distance focale de la caméra et $z = FO_1$ la distance à l'objet. En écrivant la relation de Thalès dans les triangles FO_1A_1 et FO_2A_2, on a :

$$\frac{A_1O'}{A_1O_1} = \frac{FO'}{FO_1} \qquad \text{et} \qquad \frac{A_2O'}{A_2O_2} = \frac{FO'}{FO_2} \tag{3.1}$$

D'où

$$A_2O' = \frac{L \ f}{z+d} \qquad \text{et} \qquad A'_1O' \cdot FO_1 = A'_2O' \cdot FO_2 \tag{3.2}$$

Soit en séparant $A'_1O' = A'_1A'_2 + A'_2O'$ et en les remplaçant par leurs expressions :

$$\delta_I \ z = A'_2O' \ d \tag{3.3}$$

Par substitution de A'_2O' par son expression, on obtient finalement :

$$d = \frac{\delta_I \ z^2}{L \ f - \delta_I \ z} \tag{3.4}$$

Supposons alors un cas courant : une caméra cœlioscopique de focale 30 mm avec une taille de capteur de 15 mm et une résolution de 1000 pixels dans la dimension considérée observant un objet de 30 mm à une distance de 10 cm. Si on suppose un δ_p de deux pixels, on obtiendrait alors un d de 3.5 mm, qui est une valeur très importante au vue de la faible erreur sur la détection du contour δ_I considéré, quand on la compare à la distance d'observation (erreur de 3% de la valeur d'observation). De plus, les contraintes sur les contours n'apportent que des contraintes éparses sur le recalage, localisées au niveau des bords de l'organe observé dans les images. Des informations supplémentaires (telles que des informations de texture, si l'on considère que le modèle a été texturé précédemment) doivent ainsi être utilisées afin de mieux contraindre le problème. Un

tel recalage est donc à éviter lorsque seuls les contours de l'objet sont considérés.

Un autre type de recalage désigné parfois lui aussi comme mono-image peut aussi être considéré. Il s'agit de méthodes utilisant des données provenant d'une caméra stéréo pour se recaler [176, 4]. De l'information sur la profondeur de la scène est ainsi accessible et utilisable pour des contraintes supplémentaires à appliquer sur le modèle à recaler. Ces méthodes sont ainsi bien plus contraintes que des recalages mono-image utilisant une caméra monoculaire. Néanmoins, le qualificatif de mono-image est ici légèrement usurpé car une paire d'image, provenant d'une seule prise d'un cœlioscope stéréo, est utilisée. Ce type de méthode est ainsi très similaire aux recalages multi-images décrits plus précisément dans la section suivante.

3.2.2 Recalage initial multi-images

Le second type de recalage consiste à utiliser plusieurs images provenant de la vidéo d'exploration. Il est à privilégier lorsque plus d'une image est disponible. La déformation de l'organe est alors supposée constante durant la vidéo (la scène observée est considérée rigide pendant la durée de la prise de la vidéo). Il est alors possible d'utiliser des méthodes classiques de reconstruction (SfM, SLAM, MVS, ou autre) pour obtenir une reconstruction de la surface observée [120, 117, 195] et les poses de caméra associées aux différentes images, comme cela est discuté dans le chapitre 2. Pour des reconstructions utilisant des techniques monoculaires (n'ayant aucune contrainte a priori entres les poses de caméra), un facteur d'échelle inconnu persiste dans la reconstruction. Dans les expérimentations discutées dans ce chapitre, un cœlioscope stéréo est utilisé afin de lever l'ambiguïté, mais d'autres techniques peuvent aussi être utilisées. Des méthodes de reconstruction stéréo (où on estime que la pose de la seconde caméra est connue dans le système de caméra de la première : on suppose que le calibrage externe entre les deux caméras est connu) permettent ainsi d'obtenir des reconstructions ayant directement la bonne échelle comme expliqué dans le chapitre 2. Une autre solution, si l'on souhaite utiliser des algorithmes de reconstruction monoculaire, est d'utiliser ces méthodes avec, en entrée, les images provenant d'une des deux caméras uniquement, et d'utiliser les images de la seconde caméra (couplées à celles de la première caméra) pour estimer une carte de profondeur pour chaque vue. Il est alors possible de comparer les profondeurs de ces cartes avec celles obtenues en projetant la reconstruction dans le plan image de

chaque vue considérée. On peut ainsi estimer l'échelle à appliquer à la reconstruction pour faire correspondre ces valeurs. La fiabilité de la reconstruction peut, de plus, être évaluée conjointement en utilisant une méthode similaire à celle-ci. Tout ceci est expliqué plus en détail dans le chapitre 2. Enfin, une autre méthode permettant de récupérer le facteur d'échelle de la reconstruction est d'utiliser une structure de taille connue dans la scène et de chercher à faire correspondre la reconstruction de la structure en question avec sa taille véritable. Un marqueur de taille connue peut ainsi être utilisé et placé dans la scène. Cette stratégie n'est cependant pas utilisable en cœlioscopie. Une méthode en cœlioscopie, consiste alors à cliquer des distances sur le modèle pré-opératoire (à la bonne échelle) et dans la reconstruction et à mettre ces distances en relation. En supposant la reconstruction finale fiable, il est alors possible d'utiliser les poses de caméra pour coupler les contraintes que nous aurions eu sur une image (comme les contours) entre elles. De plus, la reconstruction de la scène elle-même peut aussi servir de contrainte supplémentaire. Se recaler à l'ensemble des images revient alors à se recaler avec la reconstruction de surface elle-même [73, 150] (Figure 3.3). En effet, une fois le modèle recalé, et toujours en supposant la reconstruction fiable, une partie de sa surface (correspondant à la partie visible dans les images) doit correspondre à la reconstruction de surface de l'organe préalablement obtenue.

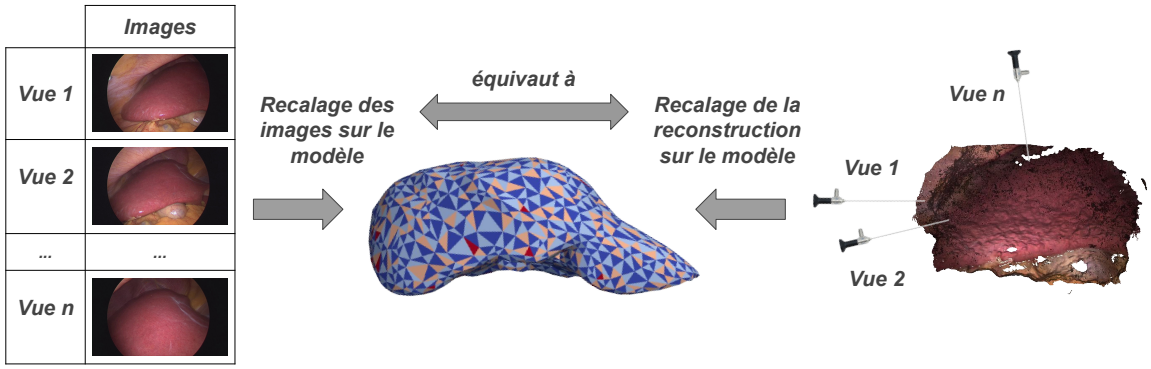

FIGURE 3.3 – Recalage multi-images : équivalence entre le recalage sur une reconstruction de surface obtenue à partir d'un certain nombre d'images cœlioscopiques et le recalage sur ces images directement.

À noter qu'avant ce recalage, un raffinement de la surface obtenue peut être réalisé en prenant en compte l'éclairage particulier de l'endoscope comme expliqué au chapitre 5.

Pour des organes peu déformables, une solution classique consiste alors à trouver la transformation rigide faisant correspondre une partie de la surface du modèle à la reconstruction obtenue. Pour ce faire, des points de repères anatomiques, localisables à la fois dans le modèle et dans la reconstruction, peuvent être utilisés pour initialiser un algorithme de type Iterative Closest Point (ICP) [31]. Cependant, pour des organes élastiques, déformés de façon non négligeable par l'insufflation, une déformation supplémentaire doit être appliquée au modèle afin de faire correspondre ces surfaces.

3.2.3 Évaluation du recalage

Le recalage initial de modèle hépatique à la scène cœlioscopique est un problème complexe. Aucune approche présente dans la littérature ne permet de résoudre ce problème de façon robuste, efficace et dans un contexte général. De grands efforts sont ainsi encore nécessaires dans ce domaine avant d'aboutir à une solution permettant de considérer ce problème comme résolu. L'un des principaux freins à l'amélioration de ces recalages provient du manque de données d'évaluation. Il est ainsi difficile d'évaluer quantitativement les réels progrès réalisés dans ce domaine au cours de ces dernières années car la comparaison des différentes méthodes proposées s'avère difficile. La raison derrière ce manque de données provient de la grande difficulté à estimer correctement la déformation des tissus en chirurgie cœlioscopique. Pour tout de même tenter d'évaluer les algorithmes de recalage dans ces conditions, plusieurs métriques et méthodes ont été proposées. Pour un tissu réel, la complète connaissance de la déformation en jeu est impossible à obtenir. De ce fait, plusieurs métriques permettant d'approximer cette déformation et permettant ainsi de définir une erreur pour le recalage, sont ainsi discutées dans la littérature. Ces métriques sont la Target Registration Error (TRE) et la Fiducial Registration Error (FRE) et sont discutées plus en détail dans les sections suivantes. De plus, différents types de données sont aussi considérés en tant que données d'évaluation. La façon la plus simple d'évaluer le recalage est ainsi d'utiliser des données simulées (*in silico*) [150, 98, 70]. Cependant l'erreur calculée dépend alors fortement de la qualité de la simulation réalisée. D'autres approches proposent d'utiliser des modèles d'organes en silicone [191, 32, 73, 92]. Néanmoins, pour réaliser de tels modèles, les caractéristiques physiques de l'organe considéré doivent être extraites et transmises au modèle créé. Répliquer ainsi le comportement *in vivo* de tels modèles

peut alors s'avérer complexe. Enfin, des données utilisant de vrais organes *ex vivo* ou *in vivo* peuvent elles aussi être utilisées [190]. Ce type de données est idéal car il permet ainsi de s'approcher au mieux de conditions réelles. Néanmoins, des jeux de données publics *in vivo* permettant de calculer l'erreur réelle d'algorithme de recalage n'existent actuellement pas dans la littérature.

3.3 Formulation par un problème d'optimisation

3.3.1 Généralités

Le recalage est considéré comme un problème de minimisation d'énergie composée de plusieurs termes. Une partie de ces termes sont appelés des termes *d'attache aux données* et correspondent à l'éloignement entre les données à recaler et les images que l'on souhaite utiliser en tant que cibles de notre recalage. On utilise par exemple des termes correspondant à la distance entre la surface du modèle et celle de la reconstruction, d'autres faisant référence à la distance entre des points de repère (comme les contours) dans le modèle et dans les images. L'autre partie de ces termes sont des termes dit de *régularisation*. Ils correspondent à la quantification de l'écart entre la configuration initiale et la configuration recalée (déformée) et sont importants afin de ne pas privilégier de la même façon toutes les configurations pour le recalage, mais plutôt celles de moindre déformation. Tous les termes utilisés dans notre cas sont définis et discutés dans la section suivante.

On considère ainsi un modèle biomécanique généré à partir d'images TDM préopératoires du foie du patient. La création d'un tel modèle est discutée dans le chapitre 2 et n'est pas réexpliquée ici. Les n sommets du modèle sont représentés par un vecteur $\mathbf{x}$ d'inconnues, exprimé dans les coordonnées du modèle per-opératoire, $\mathbf{p}$ dénotant la position initiale de ces derniers. $\mathcal{V}_s$ est le sous-ensemble correspondant aux sommets présents à la surface du modèle. Résoudre ce problème de recalage revient donc à minimiser l'énergie correspondant à celui-ci en fonction de $\mathbf{x}$. Le problème possède donc $3n$ inconnues à prendre en compte dans la minimisation, ce qui peut poser des problèmes en terme de complexité. On réduit alors les dimensions du problème en utilisant la méthode décrite dans le chapitre 2. Une fois cette minimisation réalisée, on obtient alors un compromis entre la proximité du résultat avec les données de recalage (les

images) d'une part, et la vraisemblance de la déformation estimée (en la comparant à la déformation initiale) d'autre part.

3.3.2 Énergies à minimiser

L'énergie à minimiser se compose donc de deux ensembles de termes :

$$E_{\mathrm{T}}(\mathbf{x}) = E_{\mathrm{reg}}(\mathbf{x}, \mathcal{A}_p) + E_{\mathrm{asso}}(\mathbf{x}, \mathcal{S}, \mathcal{I}), \tag{3.5}$$

où E_{asso} sont les termes d'attache aux données et E_{reg} sont les termes de régularisation. On note $\mathcal{S}$ la surface reconstruite à partir de la vidéo d'exploration et $\mathcal{I}$ les images clés associées à celle-ci. D'autres données, servant d'a priori sont notées $\mathcal{A}_p$. Dans E_{asso}, on retrouve tous les termes (pondérés) correspondant à l'accroche aux données auxquelles on souhaite se recaler. Par exemple, on considère des termes associés aux contours 2D détectés dans les images, d'autres correspondant à des points de repère 3D directement localisés dans la reconstruction et associés à des points du modèle, et des termes quantifiant la proximité du modèle avec la surface reconstruite et s'appuyant sur une méthode de type ICP [15]. E_{asso} s'écrit alors :

$$E_{\mathrm{asso}}(\mathbf{x}, \mathcal{S}, \mathcal{I}) = \lambda_{\mathrm{cont2D}} E_{\mathrm{cont2D}}(\mathcal{V}_s, \mathcal{I}) + \lambda_{\mathrm{cont3D}} E_{\mathrm{cont3D}}(\mathcal{V}_s, \mathcal{S}) + \lambda_{\mathrm{ICP}} E_{\mathrm{ICP}}(\mathcal{V}_s, \mathcal{S}). \tag{3.6}$$

Dans cette équation, les différents λ sont les poids à donner aux différentes contraintes, dépendant de la confiance dans les données ayant servir à les définir et/ou dans le degré d'importance que l'on souhaite donner à tel ou tel type de contraintes. Dans E_{reg}, on retrouve tous les termes limitant la déformation due à la physique s'appliquant à l'objet. On a ainsi un terme correspondant à l'énergie biomécanique du modèle (représentant la quantité de déformation appliquée au modèle par rapport à son état initial), des termes limitant les collisions (internes et/ou externes). D'autres termes, comme certains découlant de la gravité, pourraient être ajoutés mais ils ne sont que simplement évoqués ici. La gravité n'est ainsi par exemple pas à prendre en compte lorsque l'on ne change pas la position du patient entre la prise d'imagerie préopératoire volumique et sa position sur la table d'opération. Les différents coûts à appliquer au modèle ne sont ainsi présents que pour représenter des changements de configuration entre modèle pré-opératoire et per-opératoire. En effet, si la position du patient ne change pas et donc que la force

induite sur l'organe par la gravité ne change pas, cette dernière n'est pas à prendre en compte. On a ainsi :

$$E_{\text{reg}}(\mathbf{x}, \mathcal{A}_p) = \lambda_{\text{M}} E_{\text{M}}(\mathbf{x}) + \lambda_{\text{cols}} E_{\text{cols}}(\mathbf{x}, \mathcal{A}_p). \tag{3.7}$$

Modélisation considérée et énergie biomécanique

$E_{\text{M}}(\mathbf{x})$ correspond à l'énergie mécanique du modèle et représente la quantité de déformation existant entre la configuration initiale et la configuration courante. Pour définir une telle énergie, on utilise la méthode des éléments finis appliquée à la mécanique de l'organe, sur notre modèle. Considérer directement la 3D, et prendre comme éléments de base de notre modèle sur lesquels on réalise l'approximation mécanique (c'est-à-dire sur lesquels on applique la loi de Hooke [175]) les tétraèdres eux-mêmes, donne des expressions relativement complexes pour l'énergie correspondante. Une telle approche ne sera ainsi utilisée, en général, que dans des simulateurs se voulant extrêmement réalistes, et n'ayant pas de contrainte de temps sur leur exécution [211]. En général, une approximation 1D est préférée car elle simplifie énormément le problème. Les éléments considérés sont alors les arêtes des tétraèdres pris indépendamment. Sous cette configuration, l'application de la loi de Hooke nous amène à les assimiler à des ressorts. On parle alors de modèle *masse-ressort* : un modèle où la masse est localisée dans les sommets des tétraèdres (même si cela n'est pas utilisé puisque la gravité n'est pas prise en compte), et où les arêtes se comportent telles des ressorts [132]. Ces ressorts possèdent une constante de rigidité. Si cette constante est identique pour tous les éléments, on parle d'un modèle homogène, sinon il est question d'un modèle hétérogène. Intégrer de l'hétérogénéité dans le modèle peut s'avérer particulièrement utile en présence de zones de rigidité profondément différentes (localisables dans le modèle) [69, 128]. Ce modèle *masse-ressort* a l'avantage d'être simple, tout comme l'écriture de son énergie mécanique associée. Néanmoins, ce modèle présente des défauts. Par exemple, il est possible de voir apparaître une inversion des faces des tétraèdres sans que cela n'entraîne une augmentation de l'énergie associée au modèle. Malgré cela, ce type de modèle étant assez simple, et ce genre d'inversion se produisant assez rarement (car nécessitant une relativement grande énergie pour cela) nous le retenons pour nos expérimentations. Ces potentielles inversions sont alors gérées par un autre terme de régularisation : le terme gérant les collisions internes E_{cols}. De plus, comme la flexibilité des tissus composant

le foie est supposée la même en tout point, on utilise un modèle homogène. Soit $\mathcal{E}_d$ l'ensemble des arêtes du modèle, pour toute arête $a \in \mathcal{E}_d$, on note $a_{0,X}$ et $a_{1,X}$ les points de $\mathbf{x}$ associés à ces arêtes, $a_{0,0}$ et $a_{1,0}$ leurs positions initiales. L'énergie associée à un tel modèle peut alors s'écrire :

$$E_{\mathrm{M}}(\mathbf{x}) = \sum_{a \in \mathcal{E}_d} \left(\|a_{0,0} - a_{1,0}\| - \|a_{0,X} - a_{1,X}\| \right)^2 \tag{3.8}$$

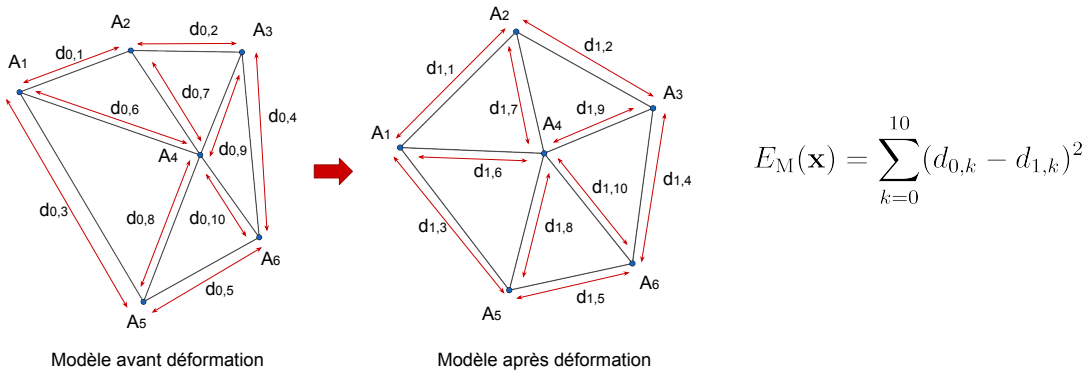

FIGURE 3.4 – Exemple d'expression de l'énergie mécanique pour un exemple simple en 2D avec 6 points $A_1, \ldots, A_6$ définissant 10 arêtes de longueur initiale $d_{0,1}, \ldots, d_{0,10}$ et de longueur courante $d_{1,1}, \ldots, d_{1,10}$.

Pour cette énergie, on considère les longueurs des arêtes comme invariantes. On peut alors étendre ceci, et considérer directement une invariance des volumes associés aux éléments 3D du modèle (les tétraèdres) [188, 158]. Une telle énergie est, cette fois-ci, robuste aux inversions de faces. Cependant, elle ne correspond pas à celle obtenue par approximation des éléments finis appliquée aux éléments 3D du modèle (aux tétraèdres). Son utilisation est donc discutable. Dans nos expérimentations, on préfère ainsi utiliser E_{M} défini à l'aide d'un modèle *masse-ressort*, ce qui simplifie l'expression de l'énergie associée au modèle et permet donc une résolution plus rapide du problème.

Énergie de correspondance de surface

E_{ICP} correspond à l'énergie dénotant la distance entre la surface du modèle et la reconstruction de surface considérée. On définit une telle énergie comme suit. Pour chaque point $P_i \in \mathcal{V}_s$ de la surface du modèle, on cherche le point le plus proche de ce dernier dans la reconstruction. Si la distance est supérieure à un certain seuil fixé, on considère que ce point n'apparaît pas dans la reconstruction. Il n'est alors pas pris en compte dans la définition de l'énergie. Pour les autres, on calcule la distance *point-plan* à ce point associé Q_i. Cette distance est définie comme suit : on considère le plan représentatif du point comme le plan passant par ce point, et de normale égale à la normale n_i du point dans la reconstruction. Cette distance s'écrit alors :

$$dist_{P2Plane}(P_i, Q_i, n_i) = n_i \cdot \overrightarrow{P_i Q_i} \tag{3.9}$$

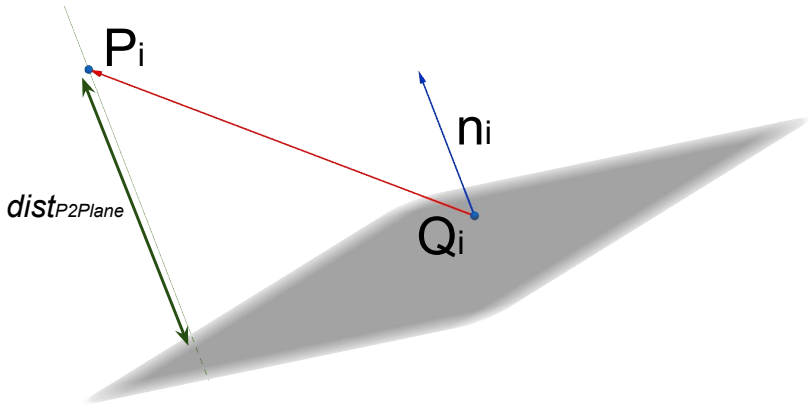

FIGURE 3.5 – Illustration de la distance point-plan.

Toutes ces distances forment alors un vecteur. Sa norme de Huber [82] notée ρ_h sert à définir notre énergie, afin d'être robuste à d'éventuels *outliers* restants dans la reconstruction (des points aberrants étant générés en grande partie par du bruit dans nos données). On écrit ainsi :

$$\begin{aligned} E_{\text{ICP}}(\mathcal{V}_s, \mathcal{S}) = \rho_h \big(&\delta_0\ dist_{P2Plane}(P_0, Q_0, n_0)\ ,\ \delta_1\ dist_{P2Plane}(P_1, Q_1, n_1)\ , \\ &\cdots\ ,\ \delta_i\ dist_{P2Plane}(P_i, Q_i, n_i)\ ,\ \cdots\ ,\ \delta_n\ dist_{P2Plane}(P_n, Q_n, n_n)\big) \end{aligned}, \tag{3.10}$$

Le calcul de correspondance préliminaire à la définition de cette énergie doit être mis à jour tout au long de la minimisation.

Énergie de correspondance 3D

Certains points de repère anatomiques peuvent être facilement repérés à la fois sur le modèle et sur la reconstruction de la surface, cible du recalage. On peut ainsi considérer des points à la surface de la vésicule biliaire, certaines zones jonction au niveau du ligament falciforme, ou même des points facilement identifiables au niveau des séparations entre les différents lobes du foie. Ces points peuvent être cliqués manuellement par un intervenant humain ou être détectés automatiquement lorsque c'est possible. Différents travaux existent dans la littérature concernant la détection automatique de tels points de repère dans les images [153], mais il s'agit ici d'un problème ouvert non abordé durant la thèse. Les points de repère utilisés sont ainsi dans nos travaux cliqués manuellement. On peut alors guider les points du modèle correspondant vers ces mêmes points repérés dans la reconstruction grâce à une énergie de correspondance 3D définie comme suit :

$$E_{\text{cont3D}}(\mathcal{V}_s, \mathcal{S}) = \sum_{(p_i, q_i) \in \mathcal{V}_s \times \mathcal{S}} \|p_i - q_i\|^2 \tag{3.11}$$

où (p_i, q_i) sont les points associés dans le modèle et dans la reconstruction. Dans notre cas, on utilise comme données pour nos expérimentations, des foies porcins (voir sections suivantes). Afin de faciliter les premières étapes de notre recalage, des points situés au niveau des séparations entre les différents lobes sont utilisés (voir Figure 3.3).

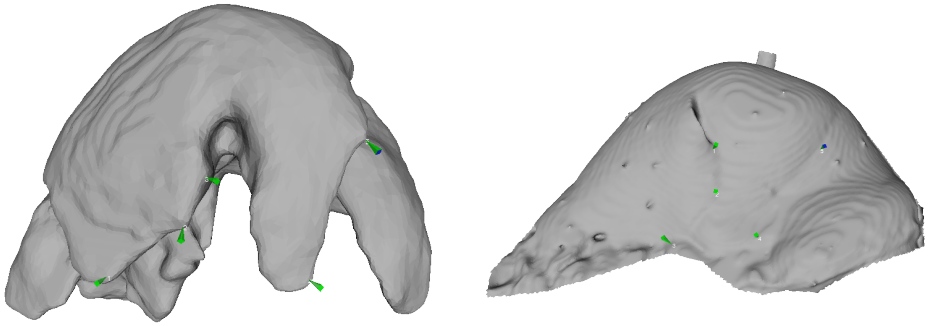

FIGURE 3.6 – Correspondances utilisées au niveau des jonctions entre les lobes pour un modèle de foie porcin. Dans le modèle à gauche et sur la surface cible à droite (extraite d'une imagerie TDM).

Ces clics peuvent ainsi être faits relativement facilement si la séparation des lobes du foie est bien visible dans la reconstruction de surface, et prendre une trentaine de secondes à être réalisé.

Énergie de correspondance aux contours

Une autre possibilité pour contraindre le problème de recalage, déjà évoquée dans l'introduction de ce chapitre, consiste à utiliser directement les contours de l'organe présents dans les images $I_k \in \mathcal{I}$. Pour cela, à chaque pas de notre algorithme, on réalise un rendu du modèle pour les différentes vues associées aux images clés de la reconstruction de la surface cible, et on extrait les points du modèle correspondant aux contours de ces rendus. On identifie ensuite les contours de l'organe dans les images, et on associe de façon automatique chaque point des contours du modèle précédemment extraits aux points des contours les plus proches dans les images. Le but est alors de rapprocher les projections des points du modèle dans les plans images, de leurs associations dans celles-ci. Pour cela, on ajoute alors un terme (qui doit être recalculé à chaque itération de l'optimisation) à l'énergie à minimiser. Si l'on note $\text{proj}(p_i, I_k)$ la projection de p_i dans la k-ième image, on a :

$$E_{\text{cont2D}}(\mathcal{V}_s, \mathcal{I}) = \sum_{k=0}^{n-1} \sum_{(p_i, \widetilde{q}_{i,k}) \in \mathcal{V}_s \times I_k} \|\text{proj}(p_i, I_k) - \widetilde{q}_{i,k}\|^2 \tag{3.12}$$

où les points $(p_i, \widetilde{q}_{i,k})$ sont les points associés du modèle et les images. La procédure pour calculer ce coût est illustrée dans la Figure 3.7.

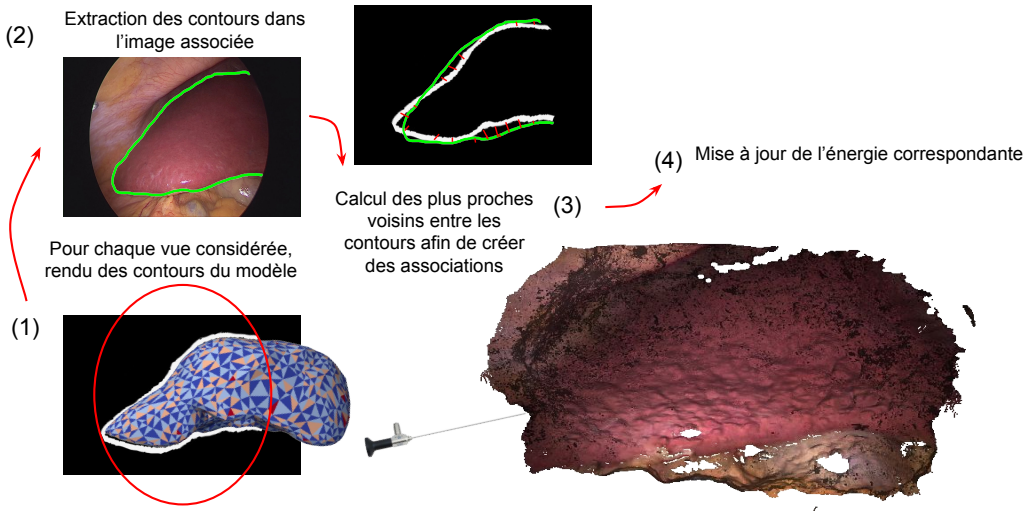

FIGURE 3.7 – Illustration de la mise à jour du coût de correspondance aux contours.

Dans nos expérimentations, deux techniques ont été utilisées pour détecter les contours dans les images cœlioscopiques. La première a ainsi été d'utiliser la méthode décrite dans le chapitre 2. Néanmoins, cette technique nécessite l'utilisation d'un cœlioscope stéréo et n'est pas toujours très précise. La deuxième technique envisagée a consisté à simplement faire annoter les silhouettes de l'organe à un opérateur humain, mais encore une fois, la précision de tels contours est limitée. Dans de futurs travaux, une solution automatique à base de *deep learning* doit être préférée [56, 71, 216].

3.4 Gestion des collisions internes

3.4.1 Prévenir les auto-intersections par la modélisation de collisions

Le besoin

Avec certaines énergies mécaniques telles que celles provenant du modèle *masse-ressort* que nous utilisons, un problème subsiste. En effet, bien que les déformations ne

soient limitées, grâce au terme d'énergie mécanique, qu'à des configurations de déformations minimales, certaines déformations non réalistes sont encore possibles et doivent être gérées. Ainsi, si le modèle possède des parties non convexes, des collisions internes entre celles-ci ne sont pas interdites, et ne sont même défavorisées par aucun terme de l'énergie. Ce problème est d'autant plus important pour des modèles de foie d'animaux tels que celui du porc composé de différents lobes pouvant glisser les uns sur les autres. Des modèles de foie porcins, recalés à la scène utilisant directement les énergies définies précédemment, ont ainsi de très grandes chances de générer des configurations dans lesquelles les lobes s'intersectent. De plus, des collisions internes peuvent apparaître même sans considérer des modèles plus simples et non convexes. En effet, avec une telle définition de l'énergie mécanique (qui a été définie de cette façon simple par choix afin de ne pas trop complexifier le problème), rien n'empêche des inversions de tétraèdres dans le modèle considéré. Par inversion de tétraèdre, on entend ici un sommet traversant la face opposée et retournant ainsi les faces de celui-ci. Pour des modèles à géométrie relativement complexe de l'organe à recaler, il est donc impératif de gérer ces collisions internes.

Approches existantes

La détection de collisions pour des modèles géométriques rigides ou déformables est un problème majeur [85]. Même en ne considérant que les auto-intersections du modèle (en prenant en compte uniquement le modèle lui-même, sans son environnement), le calcul direct de ces dernières est extrêmement coûteux en terme de nombre d'opérations à réaliser. L'essentiel de la recherche revient donc à trouver des moyens pour réduire l'espace de recherche pour les potentiels chevauchements au sein du modèle. Une première approche proposée dans la littérature, utilisée uniquement pour des modèles rigides, consiste à réaliser une découpe spatiale de l'objet et de considérer un arbre de boîtes englobantes adaptées [95, 65]. Par exemple, si on cherche les chevauchements potentiels entre deux modèles, on teste tout d'abord l'intersection des boîtes englobantes d'ordre zéro (englobant la structure entière), puis si ces boîtes se croisent, on teste les intersections des boîtes filles dans lesquelles un croisement se produit, jusqu'à arriver aux boîtes d'ordre le plus grand (correspondant aux feuilles de l'arbre, et donc aux boîtes de plus grande résolution). Si une intersection existe entre les boîtes englobantes d'ordre

maximal, on teste alors directement les collisions réelles entre les structures élémentaires (des tétraèdres en général) présentes dans ces boîtes. Cependant, cette méthode est peu adaptée aux structures déformables. En effet, la structure de l'arbre doit alors être ré-estimée à chaque étape de la déformation. Cela n'est pas facilement parallélisable et implique un grand nombre de calculs. Ainsi, intégrer une telle méthode dans notre recalage (et donc devoir recalculer cette structure à chaque pas de l'algorithme) ralentirait énormément ce dernier, ce qui n'est pas souhaitable.

Une autre approche consiste à diviser non pas le modèle de l'organe, mais l'espace dans lequel il se trouve [163, 208, 134]. Une structure d'arbre peut là aussi être utilisée pour définir des voxels de différents ordres délimitant entièrement l'espace. Cela nécessite néanmoins de pouvoir connaître avant le recalage les limites de l'espace où le modèle pourra se trouver à tout instant. Cela est possible en se donnant une marge importante. Plus on utilise, dans cette méthode, un arbre profond (et donc plus les voxels correspondant aux feuilles de l'arbre sont de petite taille), moins il y a d'éléments (tétraèdres en général) à considérer dans ces derniers, donc moins l'algorithme nécessite de calculer d'intersections réelles entre les éléments. Néanmoins, plus l'arbre est profond, plus le nombre total de voxels à stocker en mémoire est grand, ce qui peut alors être une limitation lorsque nous considérons de grandes résolutions.

Approche considérée

Au vu des limitations des deux approches présentées précédemment, nous avons développé notre propre méthode qui a comme avantage de présenter une faible complexité en termes de calculs mais aussi en termes de mémoire utilisée. Notre modèle géométrique est un modèle composé de tétraèdres. Afin de réduire l'espace de recherche, une première technique utilisée consiste à ne chercher de potentielles intersections que pour les tétraèdres à la surface du modèle. En effet, si un chevauchement existe dans le modèle alors il est entre autres présent au niveau de sa surface. Lorsqu'une telle collision est détectée, on modifie le problème de recalage pour la prendre en compte et la résorber. Cela a ainsi pour effet de déplacer les collisions les plus en profondeur vers la surface. Itérativement, toute collision interne est finalement annulée. Utiliser, comme zone de recherche pour les collisions, uniquement les éléments de surface est donc suffisant pour notre problème. Une deuxième technique que nous exploitons est

de ne pas considérer directement les chevauchements de tétraèdres mais de travailler à l'aide de structures intermédiaires. En effet, le calcul d'intersections de tétraèdres, bien qu'étant un problème résolu, reste relativement gourmand en terme de nombre d'opérations à réaliser. Pour calculer ces intersections, nous utilisons la technique décrite par [49]. Afin de vérifier si des intersections existent au sein de notre modèle, chaque tétraèdre doit ainsi être testé avec tous les autres afin de juger de la présence ou non d'entrecoupements dans le modèle. Cela peut être bien entendu massivement parallélisé, mais les calculs par thread restent relativement lourds. Pour simplifier cela, nous considérons alors les sphères circonscrites à nos tétraèdres et vérifions d'abord les intersections entre sphères. Bien évidemment, des entrecoupements de sphères existent toujours sans pour autant avoir d'intersections entre les tétraèdres considérés (comme par exemple, entre un élément et ses voisins directs) mais le but ici est de réduire l'espace de recherche en obtenant une condition nécessaire mais non suffisante. Ainsi si une intersection entre tétraèdres existe, cela signifie qu'il y a forcément entrecoupement des sphères circonscrites associées. De ce fait, on vérifie d'abord les intersections entre sphères, qui requièrent bien moins de calculs, puis on teste les croisements entre les tétraèdres si une intersection entre sphères est constatée. On réduit ainsi grandement le nombre de calculs nécessaires pour évaluer nos collisions internes. La construction du modèle est illustrée dans la Figure 3.8.

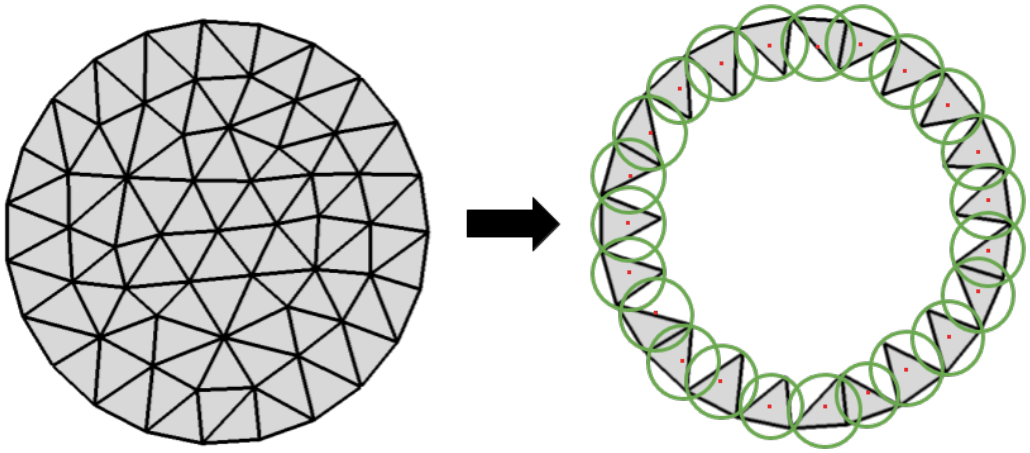

FIGURE 3.8 – Exemple simple de construction du modèle de collision en 2D.

Cette solution a été implémentée sur GPU grâce à CUDA. En pratique, on procède

de la manière suivante. Tout d'abord, on extrait du modèle original les tétraèdres de surface à prendre en compte, tout en gardant l'association entre eux et le modèle initial. On crée ensuite plusieurs buffers GPU. L'un d'eux contient la position des points des tétraèdres considérés et est mis à jour à chaque pas de déformation (c'est-à-dire à chaque pas de l'algorithme de recalage). Un autre buffer contient la structure des sphères circonscrites, c'est-à-dire les centres et les rayons. Enfin, un dernier buffer est utilisé pour récupérer le résultat des tests d'intersection et le transmettre au CPU. A chaque pas de déformation (se déroulant sur CPU), on met ainsi à jour le buffer contenant les positions de tétraèdres. On considère ensuite un thread par élément (par tétraèdre). Chaque thread recalcule tout d'abord le centre de sa sphère associée et son rayon. Une fois cette étape achevée par tous les threads, chaque thread va tester les collisions entre sa structure associée et tous les autres éléments. Pour ce faire, il vérifie tout d'abord les chevauchements potentiels entre sphères, et si une telle intersection est détectée, il vérifie l'existence ou non d'un croisement des tétraèdres associés. Le thread mettra à jour alors, conformément aux résultats obtenus, le buffer représentant la présence ou non d'intersections pour l'élément considéré. Chaque thread s'occupant alors uniquement de sa propre condition, il n'y a pas de problème de concurrence ici. Le résultat de ce buffer est alors copié sur CPU, ce qui permet finalement à l'algorithme de recalage de savoir si des collisions existent et où elles se trouvent, et de pouvoir ajouter des termes rectificatifs à l'énergie à minimiser. La définition de tels termes rectificatifs est réalisée dans la section suivante. On peut alors procéder à une itération de la minimisation avant de remettre à jour les buffers et de réitérer le processus de détection de collisions, jusqu'à l'obtention du recalage final.

3.4.2 Énergie additionnelle de répulsion

Lorsqu'une collision est repérée pendant le recalage, un terme additionnel, dépendant de la collision détectée, doit être ajouté à notre énergie à minimiser. Ainsi, lorsque l'on constate qu'un tétraèdre de surface est en collision avec un autre élément du modèle géométrique, on cherche à repousser ce dernier vers l'extérieur du modèle. D'autres stratégies que celle finalement retenue ont été envisagées mais nécessiteraient de complexifier fortement l'algorithme. Par exemple, une approche alternative est de considérer pour chaque tétraèdre, le ou les autres éléments qu'il intersecte. Le but est alors

de trouver la direction et le sens vers lequel on doit déplacer l'élément pour annuler ces chevauchements. Plusieurs problèmes surviennent alors. Tout d'abord, le buffer permettant de récupérer les intersections doit être agrandi. En effet, on doit, lorsque l'on considère cette méthode alternative, récupérer l'état de notre élément (en intersection ou non) pour chaque couple de tétraèdres auquel il appartient. Le buffer en question ne contient plus n booléens mais $n(n-1)$, où n est le nombre de tétraèdres de surface. Cette augmentation de taille pour ce buffer est cependant négligeable vu qu'il s'agit ici de booléens, et donc de buffers ayant une taille relativement petite. De plus, dans ce cas, un thread ne peut s'arrêter lorsqu'il trouve une intersection mais doit continuer afin de toutes les trouver. La quantité de calculs pour chaque thread augmente alors légèrement, ainsi que le temps d'exécution. Le problème principal est cependant de déterminer la direction, le sens et l'amplitude dans laquelle doivent être déplacés les tétraèdres. Pour cela, on peut considérer le vecteur entre les sphères inscrites associées, dont la direction nous renseigne sur celle à suivre (par couple considéré), et sa norme sur l'amplitude à utiliser. Le sens de ce vecteur ne doit cependant pas être considéré directement, mais le sens pointant vers la surface doit être utilisé (le produit scalaire entre le vecteur de la direction du déplacement et la normale à la surface doit être positif). En plus de cela, des cas limites apparaissent. Il faut par exemple traiter le cas de tétraèdres s'intersectant dont les centres des sphères circonscrites coïncident. Tout cela ajoute finalement beaucoup de complexité. Une solution plus simple est alors retenue. Pour chaque élément pour lequel une intersection a été détectée (sans prendre en compte l'élément qu'il coupe) on associe un déplacement vers l'extérieur du modèle (en utilisant la normale à la surface) de norme fixe. On souhaite alors guider le recalage pour que les éléments s'intersectant soient préférentiellement dirigés conformément aux déplacements calculés. On ajoute alors à notre minimisation le terme suivant :

$$E_{\text{cols}}(\mathbf{x}, \mathcal{V}_s) = \sum_{collisions} \|x_c - (x_{ci} + \mu_{sharpness} n_{surface})\|^2 \tag{3.13}$$

avec x_{ci} et c_c dénotant les centres des sphères associées aux tétraèdres où une collision a été détectée, respectivement avant la minimisation (position initiale) et pendant la minimisation (position courante), $n_{surface}$ dénotant les normales aux surfaces respectives et $\mu_{sharpness}$ un poids contrôlant l'amplitude souhaitée pour la correction pour un pas du recalage. L'énergie de collision est ici une énergie de rectification. La solution finale

du recalage est donc un équilibre où de petites oscillations interviennent autour de faibles collisions de surface (car il n'y a pas d'énergie empêchant de futurs impacts). Des solutions obtenues avec notre méthode, permettant de visualiser les collisions (en rouge) obtenues en fin de recalage avec et sans rectification sont présentées en Figure 3.9.

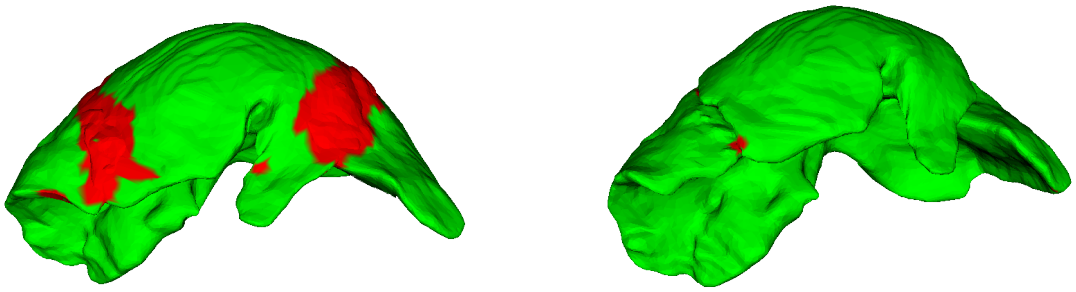

FIGURE 3.9 – Résultats sans (gauche) et avec (droite) coût additionnel permettant de gérer les collisions.

Une méthode plus globale requiert une expression analytique pour le champ de distance au modèle géométrique (comportant distances internes et externes), ce qui est bien plus complexe. Notre méthode offre une solution alternative, relativement simple où le calcul temps réel est toujours envisageable.

3.5 Résolution

Énergie finale considérée

Dans les expérimentations discutées dans la suite de ce chapitre, l'énergie finale que nous minimisons est la suivante :

$$E_{\mathrm{T}}(\mathbf{x}) = E_{\mathrm{M}}(\mathbf{x}) + \lambda_{\mathrm{cont3D}} E_{\mathrm{cont3D}}(\mathcal{V}_s, \mathcal{S}) + \lambda_{\mathrm{ICP}} E_{\mathrm{ICP}}(\mathcal{V}_s, \mathcal{S}) + \lambda_{\mathrm{cols}} E_{\mathrm{cols}}(\mathcal{V}_s) \tag{3.14}$$

On note que les termes associés aux contours E_{cont2D} ne sont pas présents. En effet, nous ne disposons pas ici de détecteur de contours précis. Les méthodes d'extraction de contours présentées dans le chapitre 2 sont imparfaites et l'erreur sur les contours obtenus est supérieure à celle discutée au début de ce chapitre ($>$ 2 pixels). L'erreur

associée à un tel coût est ainsi supérieure aux erreurs discutées. Ainsi, ces termes n'ont pas été utilisés ici.

Méthode de résolution

Le recalage est tout d'abord initialisé avec un recalage rigide réalisé manuellement, de façon relativement grossière (il n'a pas à être précis à cette étape). Afin de réduire le temps de calcul de la minimisation correspondant au recalage non rigide, nous utilisons la méthode décrite dans le chapitre 2 pour réduire l'espace de recherche pour les déformations, en exploitant le fait qu'en pratique, les déformations réalistes ont tendance à être lisses. On adopte ensuite une méthode itérative pour optimiser l'énergie. Ainsi, on minimise une première fois $E_{\mathrm{T}}(\mathbf{x})$ avec les valeurs initiales des poids associées aux différents termes expliqués précédemment à l'aide de l'algorithme de Gauss-Newton. Cette étape est réalisée à l'aide de la librairie ceres [2]. Les poids λ_{ICP} et λ_{cols} sont ensuite augmentés dans $E_{\mathrm{T}}(\mathbf{x})$ afin de diminuer la rigidité effective du modèle, tandis que $\lambda_{\mathrm{cont3D}}$ est diminué. Une seconde minimisation est alors réalisée sur cette énergie ayant été mise à jour. On itère ensuite le processus un certain nombre de fois jusqu'à obtenir le recalage final (on souhaite alors qu'entre deux minimisations, lors des étapes finales, le recalage ne soit plus qu'infimement modifié). En pratique, on fixe le nombre d'itérations à 20. Avec une telle méthode, on passe d'un modèle relativement rigide en début de résolution, à un modèle beaucoup plus flexible à la fin [37, 7]. Ceci est important pour éviter de tomber dans des minima locaux. La diminution de λ_{rep} est elle réalisée afin de ne donner aux termes correspondant à E_{rep} une importance forte qu'en début de résolution. En effet, ces points étant cliqués, les correspondances en jeu ne sont pas très précises. Les contraintes associées doivent donc n'être utilisées qu'au début de la méthode. Ainsi, le poids accordé à celles-ci doit être réduit au fur et à mesure de l'avancée de la résolution.

3.6 Évaluation du recalage en cœlioscopie

3.6.1 Jeu de données

Motivation

Afin d'évaluer notre méthode, mais aussi de permettre à d'autres recalages d'être évalués et comparés, nous avons créé une base de données publique composée de données porcines obtenues *in vivo*. Une telle base de données est importante car elle permet une évaluation plus rigoureuse des algorithmes de recalage. Cette base de donnée est de plus la première de ce type. En effet, dans la littérature actuelle, l'évaluation pour les algorithmes de recalage n'est, en général, que très peu étudiée et des solutions simples mais peu fiables, sont souvent préférées, ce qui peut alors être très discutable. Cela s'explique cependant. En effet, mesurer la précision du recalage dans le cas de la chirurgie cœlioscopique est très difficile car en pratique, la vérité terrain pour des conditions réelles, n'est pas accessible facilement. Une bonne base de donnée de validation est ainsi très précieuse car elle permet de réellement observer de façon quantitative les réels progrès étant réalisés dans le domaine. Les statistiques sur les erreurs peuvent ainsi être calculées et différents algorithmes comparées entre eux en utilisant les mêmes données et métriques.

Les métriques

Dans la littérature, deux valeurs quantitatives sont souvent discutées pour de telles évaluations : la TRE et la FRE. La FRE est la distance entre les points utilisés comme points d'accroche pour le recalage et leurs vérités terrain. Cette erreur peut être considérée comme l'erreur résiduelle du recalage. La TRE est, quant à elle, l'erreur pour les points n'ayant pas été utilisés directement en tant que points de repère. Dans les méthodes de recalage de type ICP, où de tels points ne sont pas utilisés directement (on entend ici des points de repère dont on connaît directement la position 3D dans la scène), on peut assimiler la FRE à la distance surface à surface entre le modèle et la reconstruction de surface cible. Néanmoins, lorsque l'on considère des recalages déformables, la distance surface à surface pour représenter notre erreur n'est pas une bonne métrique. En effet, comme le modèle peut être déformé pour correspondre à n'importe quel type de surface, tant que l'on considère le modèle comme assez flexible, il n'y a

aucune garantie concernant la précision du recalage final. Une surface plane peut ainsi glisser sur une autre tout en maintenant une distance nulle entre celles-ci. De plus, comme cela nous le laisse déjà penser, il a été montré qu'il n'existe que de faibles dépendances entre TRE et FRE [52, 53]. La TRE doit donc être privilégiée dans tous les cas.

Différents types de données

Différents types de données sont considérés dans la littérature. La façon la plus simple d'obtenir des résultats quantitatifs est de réaliser les recalages sur des données simulées (*in silico*) [150, 98, 70]. Avec une telle approche, la TRE peut être aisément calculée en ajoutant des marqueurs virtuels dans la scène. L'erreur correspondant à la distance surface à surface peut aussi être facilement calculée. Néanmoins, dans ce cas, une caractérisation physique des tissus de l'organe que l'on souhaite simuler doit être réalisée. Cependant, la caractérisation complète des tissus ainsi que toutes les forces s'appliquant sur ces derniers n'est pas réalisable en pratique. Un modèle simplifié doit alors être considéré pour la simulation des déformations. Le réalisme des déformations obtenues avec une telle méthode n'est alors aucunement garanti. D'autres approches proposent d'utiliser des modèles en silicone [191, 32, 73, 92] (faits d'un seul bloc homogène ou de plusieurs blocs de silicone de différents types, permettant d'avoir différentes flexibilités dans le modèle et apportant donc un réalisme plus important). Cependant, bien qu'aucune simulation n'entre en jeu ici, les caractéristiques de l'organe que l'on souhaite approcher grâce à notre modèle doivent toujours être estimées. De plus, l'utilisation de modèles en silicone ne permet pas, en pratique, de mettre en évidence toutes les caractéristiques mécaniques d'un organe réel, ni sa texture. Ajouté à cela, ce type de modèles est en général très onéreux. Pour de tels modèles, le calcul de la TRE pour un certain nombre de points peut s'avérer relativement simple, en utilisant un silicone particulier aux emplacements de ces points et en utilisant une imagerie volumique de type TDM pour chaque déformation considérée. La dernière possibilité consiste à utiliser directement des données provenant d'un véritable organe. Deux choix sont alors envisageables : prélever un organe et réaliser la prise de données *ex vivo* ou bien effectuer une opération chirurgicale sur un animal et récupérer les données nécessaires *in vivo*. Cependant, le cas *ex vivo*, bien que s'approchant des conditions réelles, n'est pas toujours le reflet exact de ce qui se passe dans la réalité. En effet, dans ce cas,

les images sont en général de meilleure qualité (pas de saignements ou de fumée par exemple), l'organe est en général isolé, et des changement de rigidité peuvent subvenir (causés entre autre par l'absence de sang circulant) [29]. De plus, il est aussi difficile de simuler les occultations naturelles (par d'autres organes ou de la graisse) présents dans le cas *in vivo*. Le cas *in vivo* correspond, quant à lui, exactement aux conditions réelles. Cependant, pour de telles données, la récupération de la vérité terrain est bien plus complexe. Pour les données *ex vivo*, une capture plus précise de la surface en question peut être réalisée en projetant un motif fortement texturé sur l'organe considéré et en réalisant la reconstruction de surface avec ces données. La surface observée ayant une plus forte texture, la reconstruction obtenue est ainsi bien plus précise (la méthode est détaillée dans le chapitre 6). Il est alors possible de calculer la distance surface à surface entre modèle recalé et surface cible. Cela correspond, comme expliqué précédemment, en quelque sorte, à la TRE. Pour calculer la FRE, une solution est d'insérer des marqueurs en surface et à l'intérieur de l'organe, dont la position est récupérable après recalage. Des marqueurs métalliques, visibles dans les images TDM, peuvent être utilisés. Une autre solution envisageable serait l'emploi de marqueurs électromagnétiques, mais leur utilisation est plus complexe car ils sont en général reliés par des fils à l'émetteur, et leur insertion (ainsi que leur manipulation en grand nombre) est plus difficile. Pour des données *in vivo*, l'insertion de tels marqueurs est encore plus difficile à effectuer. Bien qu'une telle technique soit ainsi possible, celle-ci n'a pas encore été testée. Une autre difficulté est ensuite d'associer les marqueurs entre eux à travers les différentes configurations afin de calculer les erreurs de position en ces points.

3.6.2 Méthodologie chirurgicale

Dans cette section, nous présentons le premier jeu de données public *in vivo* permettant de calculer la TRE pour des algorithmes de recalage. La prise de données est effectuée sur un grand porc anglais *Sus scrofa domesticus*, mâle de 50 kg. La procédure est illustrée dans la Figure 3.10.

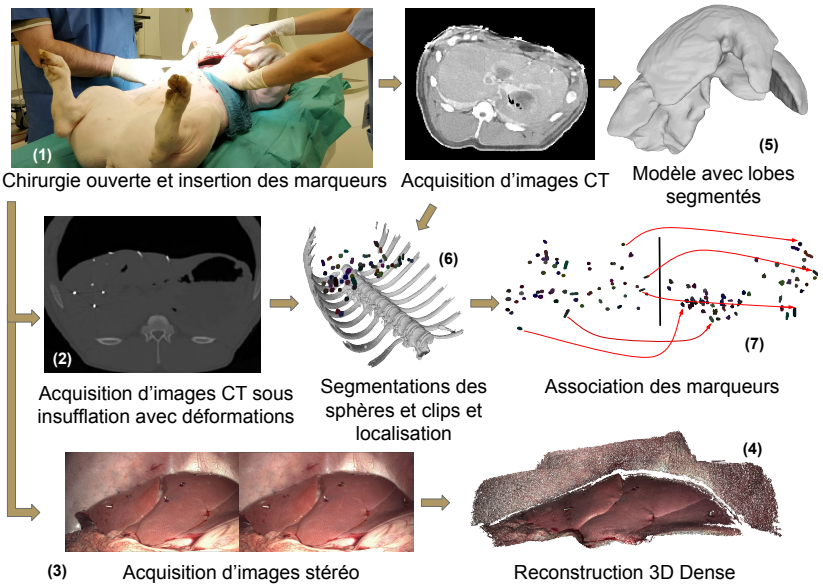

FIGURE 3.10 – Création de la base de données.

Par laparotomie, la surface du foie est rendue visible. Quinze clips métalliques de 1 cm de long sont répartis de façon homogène et fixés sur la surface ventrale des quatre lobes du foie. De plus, quarante cinq sphères métalliques de 2 mm sont introduites dans le parenchyme. Pour cela, on accède à la surface dorsale de chaque lobe et on les insère à l'aide d'un cathéter, et ce afin d'éviter des lésions indicatives de leurs positions sur la face visible (ventrale). La laparotomie est ensuite refermée et on passe à la phase cœlioscopique. Les trocarts sont insérés et la cavité est insufflée au CO2 à une pression standard (12 mmHg). On déplace ensuite les lobes suivant treize positions différentes. Pour chacune de ces configurations, une vidéo d'exploration est réalisée à l'aide d'un cœlioscope stéréo, ainsi qu'une prise de vue TDM afin d'obtenir la position des marqueurs. Deux cibles sont alors considérées pour le recalage. La première possibilité est d'utiliser une segmentation de la surface du foie provenant des images TDM. Cette première cible est considérée afin d'étudier le comportement des algorithmes de recalage en condition optimale, en utilisant l'entièreté de la surface visible. Ainsi, lors de l'évaluation, seule l'erreur de recalage est considérée (et non d'éventuelles erreurs dues au manque d'information ou à des inexactitudes dans la reconstruction). La seconde cible considérée pour le recalage provient d'une reconstruction obtenue à l'aide de la

vidéo d'exploration. Il s'agit ici du cas d'utilisation réel (le pipeline complet de cette étape). Pour obtenir cette surface, nous recourons à PhotoScan-Pro [1] dans nos tests, un logiciel de reconstruction de pointe. Cependant toute autre méthode pourrait être adoptée. Après avoir considéré ces différentes positions, le porc est ré-ouvert et des tissus imprégnés de produit de contraste sont utilisés pour séparer les différent lobes. Cela permet, lors d'une acquisition d'images TDM, de faciliter la segmentation de ces derniers et d'obtenir un modèle dans lequel les lobes sont physiquement séparés. À la fin de la procédure, le porc est euthanasié, conformément au protocole.

3.6.3 Des images à l'évaluation

À partir des images TDM obtenues, les marqueurs sont segmentés dans celles-ci, et classifiés en tant que sphères et clips en fonction de leur forme. Un recalage rigide, utilisant une ICP est réalisé entre les reconstructions de surface et celles extraites des images TDM afin de faire tous nos calculs dans le système de coordonnées lié à ces dernières. Il reste alors à associer les marqueurs dans les différentes configurations afin de permettre le calcul de la TRE. En effet, une fois cela fait, la position de chaque marqueur après déformation, interpolée à partir de la configuration de référence, peut être comparée à la position associée dans la configuration considérée : la vérité terrain. Pour cela, on analyse les distances entre marqueurs voisins. L'idée est alors de considérer la déformation comme localement isométrique. Ainsi, les distances entre marqueurs doivent être conservées dans les différentes configurations. Le problème est donc de trouver les associations, pour deux configurations données, minimisant les différences de position entre marqueurs voisins dans celles-ci. Cependant, ce problème d'association est NP-complet et pour le résoudre entièrement et de façon exacte, toutes les configurations doivent être testées. Ainsi, dans notre cas, pour deux configurations, il y aurait $45!15! \approx 10^{68}$ cas à étudier, ce qui est bien entendu impossible. Le problème doit donc être simplifié. Pour cela, un problème d'optimisation de graphe est considéré pour chaque couple de référence/configuration testée, comme illustré dans la Figure 3.11.

1. http://www.agisoft.com

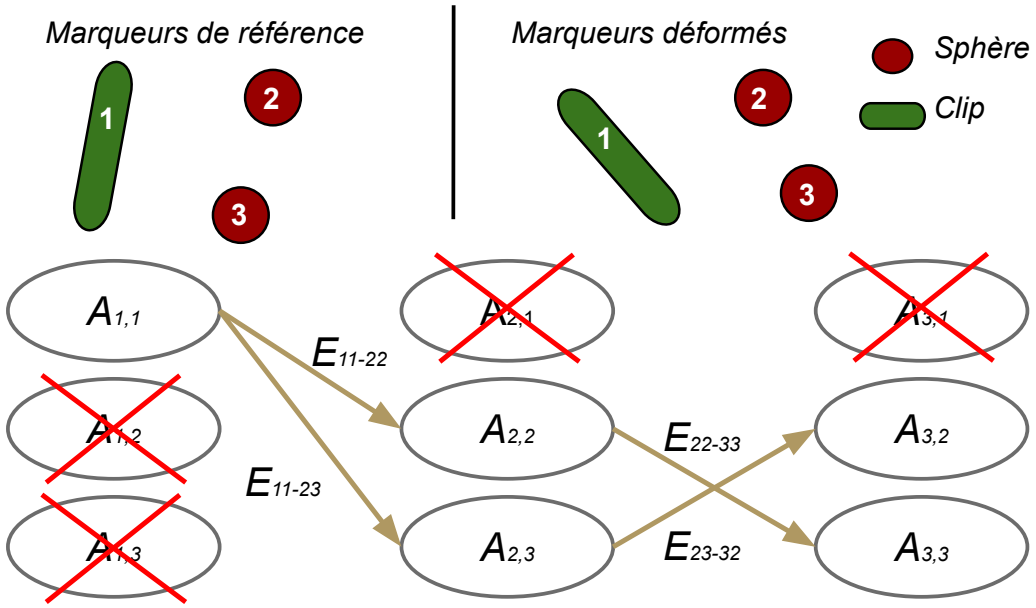

FIGURE 3.11 – Graphe représentant l'association des marqueurs.

Chaque nœud représente une association de marqueurs entre les deux configurations et chaque arc représente une probabilité d'accepter l'association suivante connaissant la précédente dans le graphe. Dans l'exemple fourni, $A_{i,j}$ représente l'association du marqueur i dans la position de référence avec le marqueur j dans la configuration testée (après déformation). E_{ij-kl} représente la probabilité d'acceptation de l'association $A_{k,l}$ en considérant l'association $A_{i,j}$ comme vraie. Une association complète entre tous les marqueurs des deux configurations correspond ainsi à un parcours de graphe. Plusieurs contraintes s'ajoutent lors de la création de ce graphe, ce qui permet de le simplifier. Ainsi, on ne considère pas les nœuds associant des marqueurs de natures différentes (sphère avec clip). De plus, pour deux associations successives, on ne conserve pas non plus les arcs ré-associant le même marqueur. Dans notre exemple, il n'y a ainsi pas d'arc entre $A_{2,2}$ et $A_{3,2}$ car cela reviendrait à associer le marqueur 2 de la configuration testée deux fois. Enfin, on ne considère que les arcs correspondant à des marqueurs voisins dans le modèle. Le poids associé à chaque chaque arc est alors :

$$E_{ij-kl} = \exp(-\lambda|dist(\texttt{Ref}_i, \texttt{Ref}_k) - dist(\texttt{Def}_j, \texttt{Def}_l)|) \tag{3.15}$$

où $\texttt{Ref}_i$ et $\texttt{Ref}_j$ sont les centres des marqueurs i et j dans la configuration de référence tandis que $\texttt{Def}_k$ et $\texttt{Def}_l$ sont les centres des marqueurs k et l dans la configuration déformée. Le problème est donc de trouver un parcours de graphe tel que la somme des arcs soit minimale et tel qu'aucun marqueur ne soit associé deux fois. Ce problème peut être résolu de différentes manières dont celle proposée dans [39]. Néanmoins, du fait que les marqueurs ne soient pas densément présents dans le foie, la solution exacte de ce problème peut ne pas coïncider avec celle attendue pour de fortes déformations. Afin de faire face à ce comportement non souhaité, nous ajoutons des contraintes supplémentaires en fixant certaines associations (cela revient à supprimer manuellement certains nœuds et arcs du graphe).

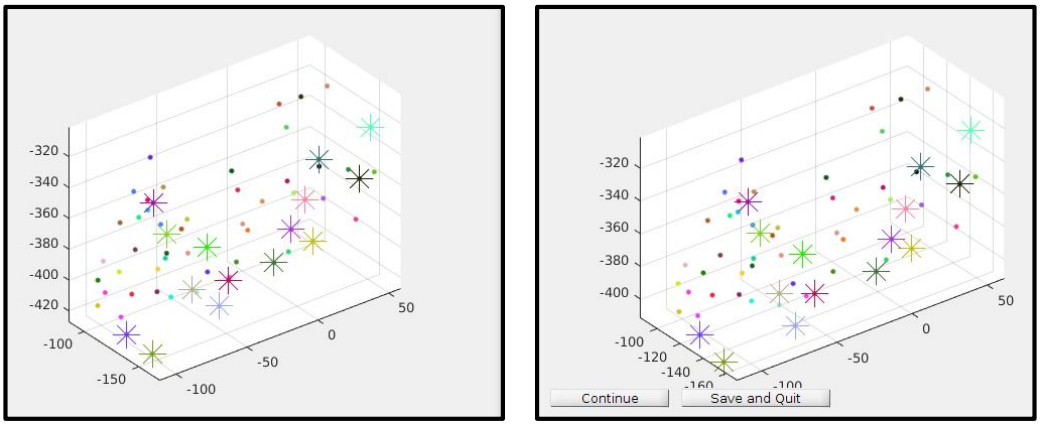

FIGURE 3.12 – Application matlab d'association de marqueurs entre deux configurations (gauche et droite).

Pour réaliser cette résolution, ayant de potentielles contraintes additionnelles, nous avons développé une interface utilisateur (Figure 3.12) résolvant le problème une première fois et proposant une solution initiale pouvant être corrigée par l'utilisateur en cliquant certaines associations reconnues. Par la suite, le problème est relancé, en prenant en compte ces nouvelles contraintes, et une autre solution est proposée. Itérativement, une solution convenable est finalement trouvée. Une fois les associations entre chaque marqueur trouvées pour chaque configuration, la TRE peut être finalement calculée. Notre base de données est ainsi prête à être utilisée. Un récapitulatif de sa composition est rappelé en Figure 3.13.

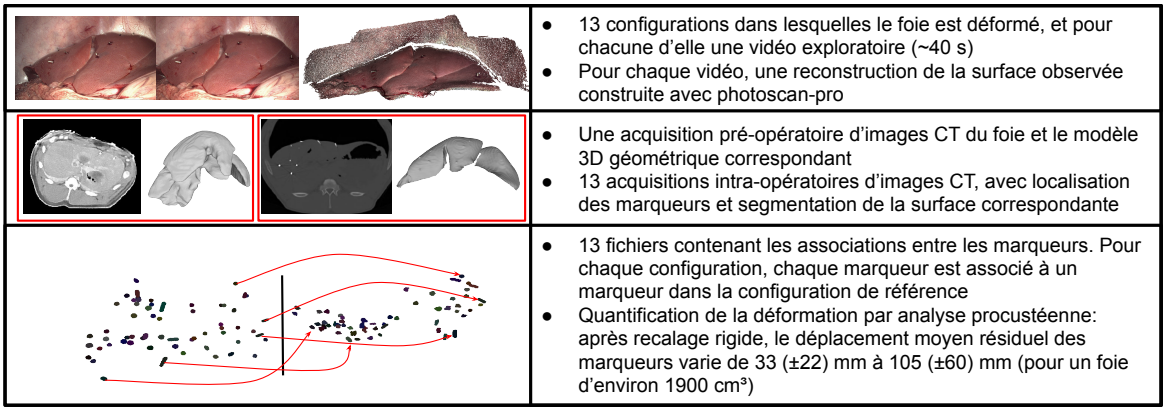

FIGURE 3.13 – Composition de la base de données.

3.7 Résultats expérimentaux

Tout d'abord, afin d'évaluer la validité de la méthode de détection de collision et sa rapidité, des tests de collisions ont étés réalisés, à l'extérieur de notre algorithme de recalage, pour un modèle de 68123 tétraèdres dans différentes configurations, proposant diverses intersections en son sein. En utilisant une GeForce GTX 1080, le temps d'exécution moyen enregistré est de 65 ms, ce qui rend alors son utilisation à l'intérieur de notre algorithme de recalage totalement acceptable.

Afin maintenant d'évaluer le recalage décrit dans ce chapitre, nous utilisons la base de données décrite dans la section précédente. En vue de juger de l'intérêt réel ou non des contraintes permettant de gérer de potentielles collisions internes, deux configurations sont alors considérées. La première (Figure 3.14) consiste à utiliser comme modèle de référence celui avec les lobes fusionnés. Cette configuration permet de s'affranchir des glissements observés sur un foie polylobé comme celui du porc. Elle permet ainsi une représentation très proche de l'anatomie de foie humain qui présente seulement deux lobes ne pouvant pas glisser l'un part rapport à l'autre. Cette approximation n'est cependant pas réaliste pour nos données à cause de différents glissements des lobes les uns sur les autres pour certaines configurations. La seconde configuration (Figure 3.15) considère un modèle ayant ses lobes séparés. Pour celle-ci, deux tests sont réalisés, l'un avec le coût de rectification (b) pour les collisions et l'autre sans (a). Pour chacun de ces tests, plusieurs erreurs sont considérées. L'une d'elles est calculée en mesurant la TRE

pour tous les marqueurs (a et b). Ce premier calcul permet alors d'évaluer le recalage sur le modèle complet et ainsi d'estimer l'erreur induite par les lobes s'intersectant. L'autre erreur n'utilise, pour son calcul, que les marqueurs se situant au niveau du lobe principal, c'est-à-dire dans la zone contenant suffisamment de données dans la vidéo d'exploration, pour pouvoir reconstruire correctement la surface de l'organe en question. Cette erreur est en pratique celle que nous devons considérer pour évaluer la précision réelle pouvant être atteinte. En effet, pendant l'opération, le chirurgien n'observe majoritairement que le lobe sur lequel est planifiée l'intervention dans la plupart des cas. Parmi les cibles du recalage, deux sont considérées. La première est une surface directement extraite des images TDM. Une telle cible permet alors d'évaluer l'erreur de recalage que nous pourrions avoir dans le cas où notre reconstruction de surface serait parfaite. Seule l'erreur induite par le recalage est ainsi représentée. La seconde cible consiste en une reconstruction obtenue à partir de la vidéo d'exploration, ayant été recalée rigidement à la scène liée aux images TDM. L'erreur observée correspond ainsi à celle de notre recalage, à laquelle s'ajoute l'erreur de reconstruction ainsi que celle du recalage rigide entre le système de coordonnées lié à la reconstruction et le système de coordonnées lié aux images TDM.

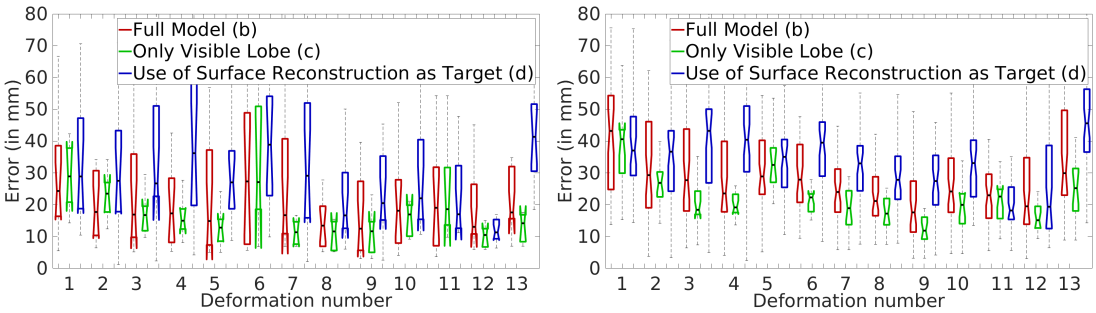

FIGURE 3.14 – Boîtes à moustaches (avec médiane, quartiles et valeurs extrêmes) montrant les erreurs de recalage à la surface (gauche) et à l'intérieur du foie (droite) pour des modèles aux lobes non-segmentées. (b,c,d) sont expliquées dans cette section.

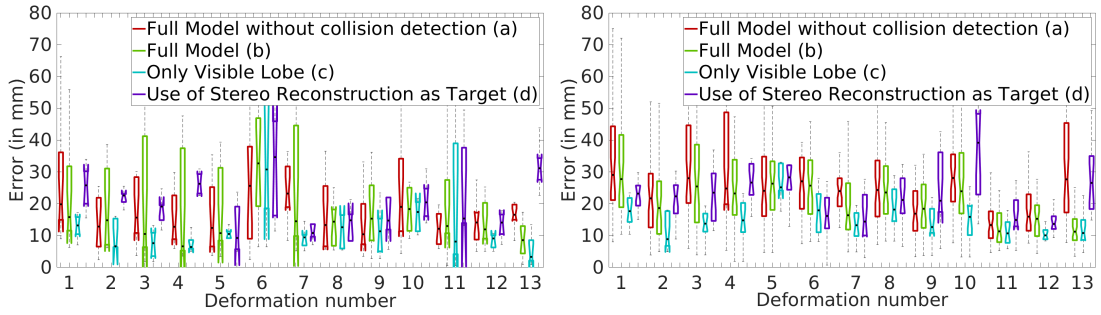

FIGURE 3.15 – Boîtes à moustaches montrant les erreurs de recalage à la surface (gauche) et à l'intérieur du foie (droite) pour des modèles aux lobes segmentés. (a,b,c,d) sont expliqués dans cette section.

Qualitativement, on observe dans la Figure 3.9 que les termes correctifs devant empêcher les collisions internes fonctionnent comme il était prévu : sans eux, on observe de profondes intersections entre les lobes tandis qu'en les prenant en compte dans le recalage, seules de faibles collisions résiduelles en surface subsistent. Les résultats quantitatifs sont montrés en Figure 3.14 et 3.15. On observe tout d'abord que les résultats sont en général meilleurs lorsqu'on utilise le modèle ayant les lobes segmentés, comme on s'y attendait (10 mm d'amélioration pour les marqueurs à l'intérieur du foie, et 5 mm pour ceux en surface). De plus, pour ce qui est des marqueurs présents à l'intérieur du modèle, l'utilisation de notre gestion des collisions améliore légèrement la qualité du recalage (1-2 mm d'amélioration), ce qui montre l'intérêt de la méthode. En considérant le cas le plus proche des conditions réelles (uniquement le lobe principal) et en utilisant notre détection de collisions internes, l'erreur médiane est alors de 20 mm environ à l'intérieur du modèle et de 15 mm en surface. L'erreur est ainsi encore relativement trop importante pour permettre la localisation précise de structures de petite taille telles que les vaisseaux sanguins. Néanmoins, elle peut être utilisée afin de visualiser des tumeurs de taille moyenne et d'aider à la planification de l'ablation tout en considérant des marges d'incertitude autour de la tumeur. Deux points sont aussi à prendre en compte. Le jeu de données utilisé pour l'évaluation de notre méthode est un jeu de données complexe, avec des foies porcins dont les lobes glissent les uns sur les autres. Un recalage sur des foies humains, où un tel glissement ne se produit pas, est ainsi plus simple à réaliser. De plus, l'erreur évaluée dans ce chapitre correspond à la véritable erreur 3D du recalage, et non à l'erreur de reprojection des structures dans

l'image qui serait alors bien plus faible. Une autre évaluation quantitative est réalisée sur un foie humain et présentée dans la Figure 3.16.

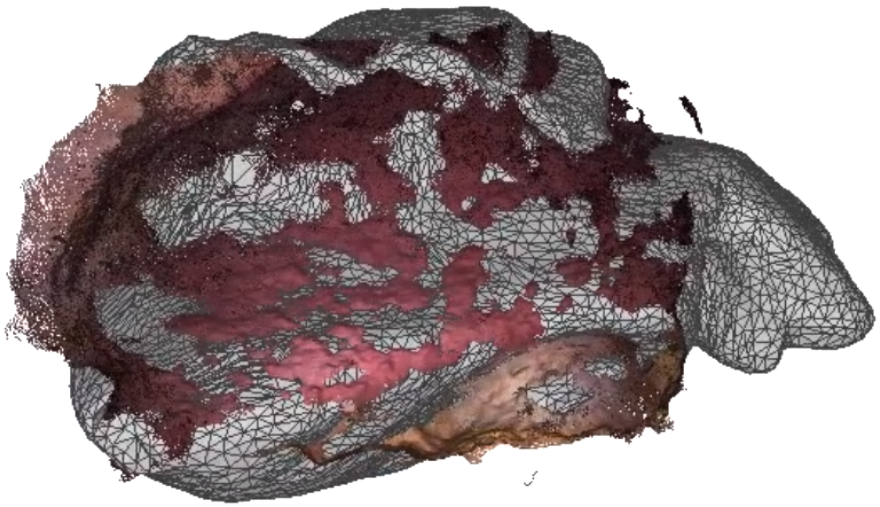

FIGURE 3.16 – Résultat de notre algorithme de recalage sur des données hépatiques humaines.

On observe ainsi le résultat du recalage et la reconstruction de surface cible (provenant d'une reconstruction réalisée avec ORBSLAM2 [118]). Cela montre ainsi la validité de notre méthode avec des données humaines. Une évaluation quantitative de notre méthode utilisant de telles données n'est cependant pas encore réalisée.

3.8 Conclusion

Nous avons présenté dans ce chapitre un algorithme de recalage déformable adapté aux données présentes dans le cadre de la cœlioscopie assistée par ordinateur. Des modèles pré-opératoires de foies construits à partir d'images TDM ou IRM peuvent ainsi être recalés à la scène cœlioscopique per-opératoire, et plus particulièrement à des vidéos d'exploration prises en début de chirurgie. Ce recalage initial est important car il constitue une étape clé du pipeline de cœlioscopie augmentée. L'algorithme que nous

proposons est, de plus, capable de prévenir d'éventuelles collisions internes pouvant survenir au sein du modèle. Ceci est indispensable lorsque nous considérons des modèles complexes d'organes comme, par exemple, des foies porcins, qui sont constitués de plusieurs lobes pouvant glisser les uns sur les autres. Pour tester notre méthode, nous avons créé une base de données publique *in vivo* servant à évaluer la précision de différents algorithmes de recalage. Grâce à celle-ci, nous avons montré que notre algorithme permet d'obtenir des précisions médianes de l'ordre de 20 mm en interne et 15 mm en surface. De plus, nous avons aussi montré que la prise en compte de collisions internes pour de telles données est importante car elle permet d'améliorer de façon significative la précision du recalage. Plusieurs améliorations de la technique présentée dans ce chapitre sont alors possibles. Dans des travaux futurs, l'automatisation du recalage rigide initial sera ainsi considérée. De plus, afin d'améliorer la précision du recalage, des contraintes sur les contours utilisant des détecteurs précis à base de *deep learning* pourront être utilisées. La sélection des points de repère anatomiques pourra elle aussi être réalisée de façon automatique, permettant ainsi une plus grande précision des clics pouvant ainsi résulter en une amélioration des contraintes sur le modèle et donc à une meilleure précision sur le recalage. Enfin, une dernière piste serait d'évaluer l'incertitude du recalage et de l'intégrer dans le résultats, permettant ainsi d'augmenter la vue avec ces données lorsque l'on considère une application complète de cœlioscopie augmentée.

Chapitre 4

Recalage postural

Dans ce chapitre, nous nous intéressons au problème de recalage de posture. Un modèle 3D de l'abdomen du patient, construit à partir d'images volumiques préopératoires (obtenues par TDM ou IRM) doit ainsi être recalé sur des images RGB-D du patient, présent le jour de l'opération, sur la table. Un tel recalage est utile pour des applications de suivi de doses d'irradiation lors de séances de radiothérapie, des applications d'échographie assistée par la robotique, ou encore pour des applications de chirurgie percutanée assistée par ordinateur comme l'aide à l'insertion d'aiguilles radiofréquence (pour l'ablation par radiofréquence). Pour résoudre ce problème, nous avons adapté l'algorithme de recalage présenté au chapitre précédent à cette situation. Afin de tenir compte de la plus grande disparité de la rigidité dans l'abdomen, due en particulier aux os, qui ne se déforment pas, un modèle biomécanique hétérogène est ainsi utilisé. De plus, une modélisation de la table est réalisée et les collisions avec cette dernière sont prises en compte afin de diminuer l'erreur de recalage au niveau de l'interface entre le patient et la table. Notre algorithme de recalage est ainsi testé avec des données animales : un porc euthanasié, positionné suivant différentes configurations, et sur la peau duquel sont disposés des marqueurs métalliques servant à calculer la vérité terrain, est ainsi utilisé en tant que cible du recalage. Cela nous permet ainsi d'évaluer l'algorithme de recalage et de montrer l'importance de la modélisation des collisions avec la table afin d'augmenter sa précision. Les travaux décrits dans ce chapitre correspondent à notre publication [128] pour la conférence MICCAI 2018.

4.1 Introduction

L'algorithme de recalage présenté au chapitre précédent, bien que développé initialement pour le contexte de la cœlioscopie, peut aussi être adapté pour d'autres situations. Un autre besoin médical auquel peut aussi répondre notre méthode est le recalage de modèle de posture du patient sur la table d'opération. Dans ce but, des données volumiques (provenant d'images TDM ou IRM préopératoires) de l'abdomen du patient doivent être recalées sur des images 2D de sa personne lorsqu'il se trouve sur la table d'opération. Un tel recalage peut ainsi être utile lors de traitements par radiothérapie afin de pouvoir suivre tout au long des différentes séances la dose de rayonnement reçue par le patient à certains endroits spécifiques de sa peau [88, 157, 200]. Pour ce faire, un modèle de la surface du patient doit être mis en correspondance pour les différentes séances. Cela permet alors d'éviter la surexposition aux radiations de certaines parties de la peau (en privilégiant d'autres accès à la zone interne à irradier) et éviter ainsi des complications futures pour le patient (comme les brûlures). Une autre utilisation de ce recalage de posture est l'aide à l'insertion d'aiguille [169] pour réaliser des biopsies [66, 93] ou dans le contexte de l'ablation par radiofréquence [22, 40]. En recalant un modèle préopératoire de l'abdomen du patient contenant les différents organes à la table d'opération, ces derniers peuvent servir à augmenter la vue du chirurgien et ainsi permettre une meilleure estimation de l'endroit où l'aiguille doit être insérée. Enfin, une dernière application du recalage postural est l'échographie assistée par robotique. On pense ainsi aux travaux décrits dans [74] où l'échographie est directement réalisée via un bras robotique de façon automatique après avoir sélectionné dans les images IRM une région d'intérêt. Pour ce faire, les images IRM doivent être recalées au patient ainsi qu'à la sonde échographique.

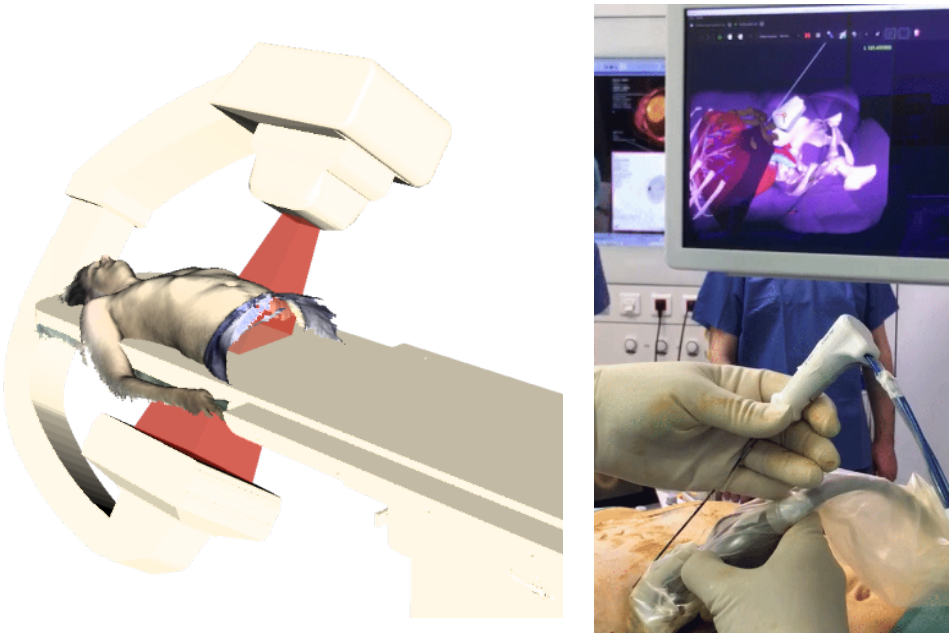

FIGURE 4.1 – Deux exemples d'applications du recalage de posture, le suivi de dose et l'aide à l'insertion d'aiguille radiofréquence.

4.2 État de l'art

De récentes approches ont été proposées pour résoudre ce problème et recaler des modèles 3D préopératoires, en capturant la posture du patient, mais aussi les structures alentours à l'aide de caméras externes RGB [119] ou RGB-D (couleurs et profondeur) [116, 106, 28, 197]. Cependant, de telles méthodes se limitent actuellement aux recalages rigides. Les caméras RGB-D présentent l'avantage d'être peu coûteuses, compactes, et elles permettent de scanner de larges régions de la surface du corps du patient. Cependant, avec de tels dispositifs, seules les informations de la surface du patient sont accessibles et peuvent être exploitées pour contraindre le recalage en question. Cela peut alors rendre ce dernier particulièrement difficile, en particulier lorsque l'on considère la région abdominale où la surface de la peau ne possède que peu de caractéristiques géométriques significatives. Une autre grande difficulté à surmonter est la présence d'importantes régions occultées dans les données. En effet, si le patient est allongé sur

le dos par exemple, alors la région dorsale ne sera pas visible, ce qui prive ainsi une zone entière du modèle à recaler de contraintes utilisables provenant directement des images. De plus, négliger les déformations du modèle (et considérer uniquement un recalage rigide) peut s'avérer très contestable, surtout lorsque ce dernier se trouve dans une position différente de celle dans laquelle il était lorsqu'ont été réalisées les images 3D, à l'origine du modèle préopératoire. Les images TDM, utilisées pour construire notre modèle, sont en effet généralement capturées en position allongée tandis que la procédure peut requérir de positionner le patient en position ventrale ou latérale. De fortes déformations de l'abdomen du patient entrent alors en jeu et doivent être prises en compte afin d'obtenir un recalage précis.

4.3 Méthode de recalage proposée

Nous proposons d'adapter l'algorithme de recalage présenté dans le chapitre précédent pour résoudre ce problème de posture. Il s'agit ainsi du premier travail proposant d'utiliser une méthode non rigide pour se recaler à un patient sur la table d'opération à l'aide d'une reconstruction de surface obtenue par fusion d'images RGB-D, les autres travaux de la littérature considérant uniquement un recalage rigide pour ce problème. Bien que le principe de la méthode de recalage discuté dans ce chapitre soit le même que celui décrit dans le chapitre précédent, des différences existent. Tout d'abord, la surface cible sur laquelle on se recale ici ne provient pas d'une vidéo d'exploration mais d'une numérisation de la surface du patient sur la table, réalisée en début d'opération à l'aide d'une caméra RGB-D (une Orbbec Astra Pro [1] dans notre cas). Une autre option possible pour obtenir des données auxquelles on souhaite se recaler est d'utiliser des caméras RGB-D fixées au plafond. Cela présente néanmoins un certain nombre de désavantages. On peut citer le coût de ces dernières, des occlusions pouvant intervenir, causées par le matériel ainsi que le personnel présent dans la scène, mais surtout le fait que ces caméras soient en général placées à une distance bien plus grande de la table d'opération. Tout ceci peut alors entraîner des problèmes de précision pour la surface cible obtenue de cette façon. Afin de fusionner les cartes de profondeur obtenues lors de la numérisation en une unique surface relativement dense à utiliser en tant que cible de notre algorithme de recalage, nous utilisons une méthode existante : Kinect Fusion [84],

1. https://orbbec3d.com

mais tout autre procédé pourrait aussi être utilisé. La procédure de numérisation de la surface est réalisée en une dizaine de secondes, nous permettant ainsi de négliger le mouvement induit par la respiration pendant ce laps de temps. Une autre solution, aurait été d'utiliser des reconstructions de surface prenant en compte la déformation de la surface à reconstruire comme ElasticFusion [207]. Une autre modification de l'algorithme provient de la prise en compte de la collision entre le modèle à recaler et la table d'opération. En effet, afin d'être résilient au manque de données présent au niveau de la zone de contact entre le patient et la table, la collision avec celle-ci doit être modélisée pendant le recalage.

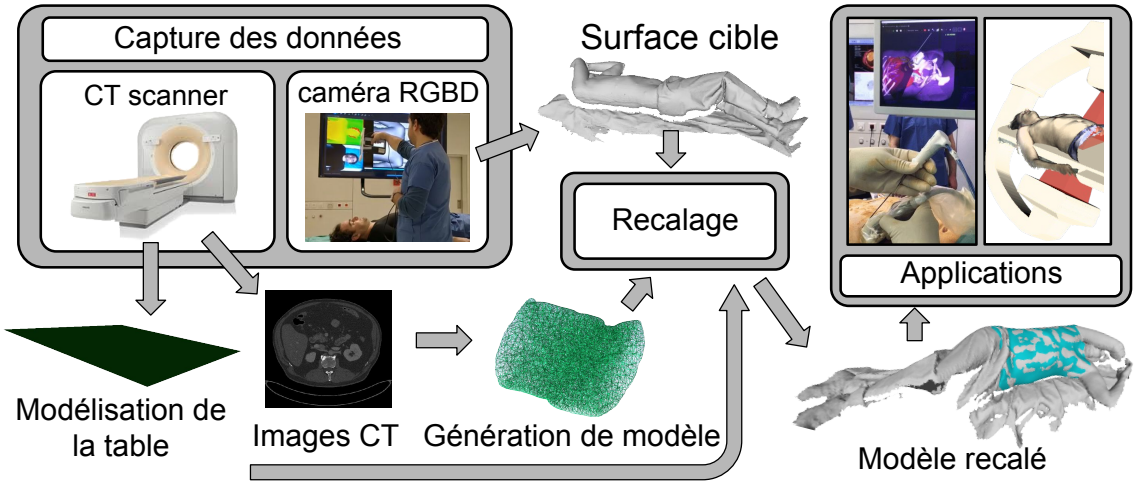

FIGURE 4.2 – Pipeline global du recalage de posture.

Enfin, on utilise ici un modèle hétérogène pour le modèle biomécanique associé au patient plutôt qu'un modèle homogène, afin de prendre en compte la rigidité plus ou moins importante de certaines parties du corps (en particulier les os). Ces deux points seront discutés dans la partie suivante.

4.4 Un modèle biomécanique hétérogène

On segmente tout d'abord dans les images TDM préopératoires les différents organes ainsi que les os et le reste des tissus. Une valeur de flexibilité est alors choisie pour les classes associées aux segmentations. Dans nos travaux, seule la classe correspondant

aux os possède une flexibilité différente, fixée à 100 fois la valeur des autres tissus (fixée à 1.0). À noter que lorsque l'on considère les rigidités réelles des organes ce facteur est plutôt de l'ordre de 10^5 à 10^6. Cependant, pour des raisons de stabilité lors de la minimisation, ce facteur est uniquement fixé ici à 100, ce qui est suffisant pour que les os ne soient pas déformés dans notre cas. Rien n'empêche néanmoins d'utiliser d'autres valeurs différentes, choisies pour chacune de ces classes. Cela aurait cependant nécessité d'apprendre ces valeurs ou d'utiliser des mesures recensées par des études réalisées sur la rigidité des différents tissus considérés. Par souci de simplicité, nous avons choisi de considérer que seuls les os possèdent une rigidité réellement différente des autres tissus, ce qui nous paraît être une supposition tout à fait raisonnable. En effet, les seules parties ne correspondant pas à des os peuvent ainsi réellement se déformer.

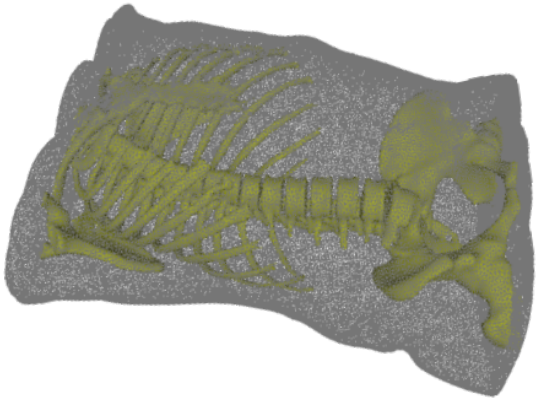

FIGURE 4.3 – Utilisation des os pour ajouter de la rigidité dans le modèle.

À partir de ces images segmentées, on construit un modèle biomécanique de l'abdomen du patient consistant en un modèle *masses-ressorts*, déjà discuté précédemment, dont les valeurs de rigidité des ressorts sont fixées à celles préalablement choisies. C'est ce modèle qui est utilisé pour notre recalage. Cela nous permet d'obtenir des déformations bien plus réalistes qu'avec un modèle homogène. La déformation du modèle est ainsi paramétrée par un vecteur d'inconnues $\mathbf{x}$, correspondant à la position des points du modèle dans le système de coordonnées lié à la numérisation du patient (c'est-à-dire la configuration déformée). À partir du modèle mécanique, on définit ainsi $E_{\mathrm{M}}(\mathbf{x})$, l'énergie biomécanique du modèle. Comme pour le chapitre précédent, si $\mathcal{E}_d$ dénote l'ensemble

des arêtes du modèle, et que pour toute arête $a \in \mathcal{E}_d$, on note $a_{0,X}$ et $a_{1,X}$ les points de $\mathbf{x}$ associés à ces arêtes, et $a_{0,0}$ et $a_{1,0}$ leurs positions initiales, on a alors :

$$E_{\mathrm{M}}(\mathbf{x}) = \sum_{a \in \mathcal{E}_d} \alpha_r \left(dist(a_{0,0}, a_{1,0}) - dist(a_{0,X}, a_{1,X}) \right)^2 , \tag{4.1}$$

avec α_r le coefficient de rigidité de l'arête associée valant 100.0 pour une arête appartenant aux os, ou 1.0 sinon.

4.5 Gérer les collisions externes

4.5.1 Modèle paramétrique de l'objet

À cause du manque de données au niveau de la partie de l'abdomen du patient en contact avec la table, le modèle recalé risque de passer à travers cette dernière pour de fortes déformations (lorsque le patient est en position latérale par exemple), ce qui aboutirait donc à une configuration recalée non réaliste. La collision avec la table opératoire doit ainsi être modélisée. Nos travaux sont ainsi les premiers à traiter ce cas. On considère ainsi un modèle paramétrique pour notre table, noté $\mathcal{T}$. Pour nos travaux, nous assimilons la table à un plan. Cependant, tout autre modèle peut être utilisé suivant la table considérée. On doit ainsi adapter la suite de la méthode de prise en compte des collisions conformément au modèle choisi. À partir de la numérisation du patient sur la table, on cherche, dans la surface obtenue, la zone plane principale, qui correspond ici à la table. D'autres indices peuvent aussi être utilisés afin de supprimer des parties planes n'appartenant pas à la table, comme la couleur des points de la reconstruction. On cherche alors les paramètres du plan correspondant aux points de la zone précédente extraite (dans notre cas, on cherche a,b,c,$d \in \mathbb{R}^4$ définissant le plan $ax+by+cz+d=0$). Cela revient à trouver le plan tel que la somme des distances entre les points de la zone extraite et le plan soit minimale. Durant le recalage, on souhaite donc que les points du modèle du patient restent dans l'un des demi-espaces engendrés par le plan correspondant à la zone située devant la table.

4.5.2 Énergie correspondante

À partir du modèle paramétrique de la table $\mathcal{T}$, on définit la fonction de distance signée relative à celle-ci d_{rel}, telle que la distance obtenue soit positive devant la table et négative derrière :

$$\begin{array}{rcl} d_{rel} & : & \mathbb{R}^3 \longrightarrow \mathbb{R} \\ & & x \longmapsto \begin{cases} dist(x, \mathcal{T}) & \text{si devant la table} \\ -dist(x, \mathcal{T}) & \text{sinon.} \end{cases} \end{array} \tag{4.2}$$

On souhaite alors ajouter un coût correctif à l'énergie à minimiser lorsque des points du modèle passent à travers la table. On définit l'énergie additionnelle de collision comme :

$$E_{\text{collision}}(\mathbf{x}) = \sigma_k \sum_{p \in \mathcal{V}} \max(d_{rel}(p, \mathcal{T}), 0), \tag{4.3}$$

avec σ_k le coefficient de pénalité associé aux collisions et dépendant du nombre d'itérations de l'algorithme de recalage déjà effectuées. Initialement, $\sigma_0 = 1$, puis, à chaque itération, il est multiplié par un facteur 10 (choisi de façon empirique). Cela permet ainsi de résorber lentement les collisions en début de résolution, afin de faciliter la mise en correspondance des points du modèle et des points de la reconstruction, puis de les résorber totalement en fin de résolution.

4.6 Énergie de correspondance de surface

Comme dans le chapitre précédent, on définit une énergie de correspondance entre la surface du modèle et la surface reconstruite. Pour chaque point $P_i \in \mathcal{V}_s$ de la surface du modèle, on cherche Q_i, le point le plus proche de ce dernier dans la reconstruction. Si la distance est supérieure à un certain seuil fixé, on ne considère pas le point dans la définition de l'énergie (car il correspond à un point non reconstruit : un *outlier*). Si n_i est la normale en Q_i, $dist_{P2Plane}$ la distance *point-plan* définie dans le chapitre précédent, et ρ_h la norme de Huber [82] on a :

$$\begin{aligned} E_{\text{ICP}}(\mathcal{V}_s, \mathcal{S}) = \rho_h \big(& \delta_0 \; dist_{P2Plane}(P_0, Q_0, n_0) \; , \; \delta_1 \; dist_{P2Plane}(P_1, Q_1, n_1) \; , \\ & \cdots \; , \; \delta_i \; dist_{P2Plane}(P_i, Q_i, n_i) \; , \; \cdots \; , \; \delta_n \; dist_{P2Plane}(P_n, Q_n, n_n) \big) \end{aligned}, \tag{4.4}$$

avec $\delta_i = 0$ si P_i est un *outlier*, ou 1 sinon.

4.7 Résolution

On considère l'énergie totale comme étant :

$$E_{\mathrm{T}}(\mathbf{x}) = E_{\mathrm{M}}(\mathbf{x}) + \lambda_{\mathrm{ICP}} E_{\mathrm{ICP}}(\mathcal{V}_s, \mathcal{S}) + E_{\mathrm{collision}}(\mathbf{x}), \tag{4.5}$$

avec λ_{ICP} le poids associé à l'énergie de correspondance ICP. Pour minimiser cette énergie, on utilise la même méthode que celle décrite dans le chapitre précédent. L'espace des solutions (c'est-à-dire l'espace des déformations) est tout d'abord réduit (avec la méthode décrite dans l'Annexe A). On clique ensuite des points correspondants sur le modèle ainsi que sur la surface reconstruite. Cela permet ainsi de calculer une transformation rigide entre la reconstruction et le modèle et d'ainsi initialiser la position des points du modèle. On procède ensuite à une méthode itérative basée sur plusieurs résolutions de la minimisation des énergies associées en utilisant l'algorithme de Gauss-Newton, tout en recalculant les associations et les coûts associés entre chaque itération. De la même façon que précédemment, le poids λ_{ICP} est modifié afin de passer d'un modèle relativement rigide aux premiers instants de la minimisation à un modèle bien plus élastique en fin de résolution, afin de permettre une meilleure convergence de l'algorithme [37, 7].

4.8 Expériences

4.8.1 Méthodologie d'évaluation

Un grand porc anglais (*Sus scrofa domesticus*) euthanasié d'environ 50 kg est utilisé pour l'évaluation de notre méthode. Pour ce faire, à la surface de sa peau, 197 marqueurs circulaires (d'environ 1 cm de diamètre) sont alors répartis et fixés. L'utilisation de ces marqueurs permet une évaluation dense de l'erreur au niveau de la surface de la peau. Il s'agit du premier travail considérant un tel calcul dense de l'erreur de surface dans ces conditions. Tout d'abord, une prise d'image TDM de référence est réalisée en position dorsale afin de construire notre modèle à recaler. Le porc est alors déplacé en

19 positions différentes, représentant les poses pouvant être prises par le patient sur la table d'opération (13 positions dorsales, 4 positions latérales gauche et 2 positions latérales droite). Pour chacune de ces postures, on réalise alors une prise d'image TDM ainsi qu'un scan RGBD. Dans les images TDM, on récupère alors la position des centres des disques marqueurs, qui nous servira à calculer l'erreur de recalage de la même façon que dans le chapitre précédent. Toutefois, les déformations étant peu importantes et les marqueurs relativement bien espacés (environ 50 mm entre deux marqueurs en moyenne), notre méthode d'association développée dans le chapitre précédent n'est pas utile ici et on peut considérer les marqueurs les plus proches (après déformation) comme étant les marqueurs associés.

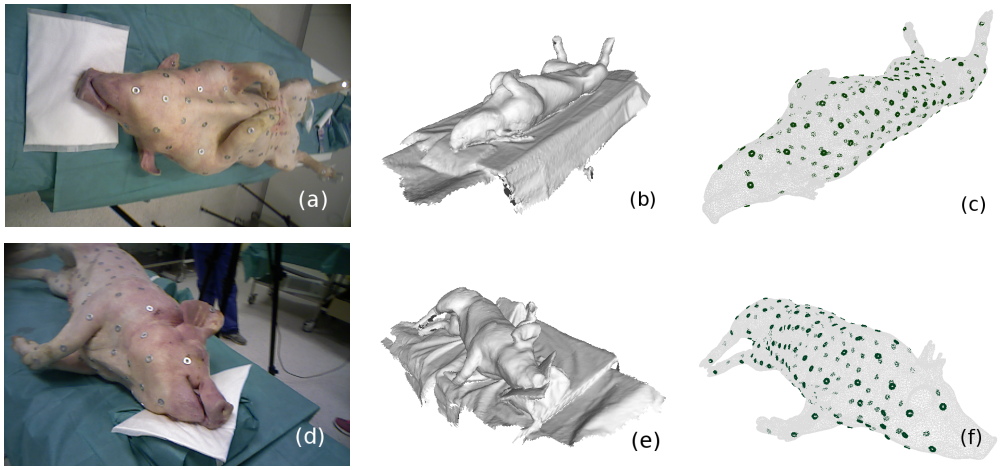

FIGURE 4.4 – Données utilisées pour l'évaluation. (a) Exemple de position dorsale, (b) Numérisation RGB-D correspondant à (a), (c) modèle CT correspondant à (a) avec les marqueurs de surface segmentés, (d-f) images équivalentes en position latérale droite.

Dans la littérature, lorsque l'on considère le recalage de posture et ses applications, seul un recalage rigide est envisagé. On compare ainsi les résultats de notre recalage déformable à ceux obtenus avec un recalage rigide basé sur un ICP. Pour notre recalage, on considère un modèle hétérogène, tel que celui décrit dans les sections précédentes, mais aussi un modèle homogène, afin de juger de l'importance ou non d'une telle considération. De plus, pour ces différentes configurations, on teste notre méthode avec et sans coût empêchant la collision avec la table, afin de juger de la pertinence de tels termes. Enfin, plusieurs sections de l'abdomen sont considérées afin de générer notre modèle

à recaler. Ceci est réalisé afin de déterminer quelle taille de section de l'abdomen est réellement utile afin d'obtenir un recalage acceptable.

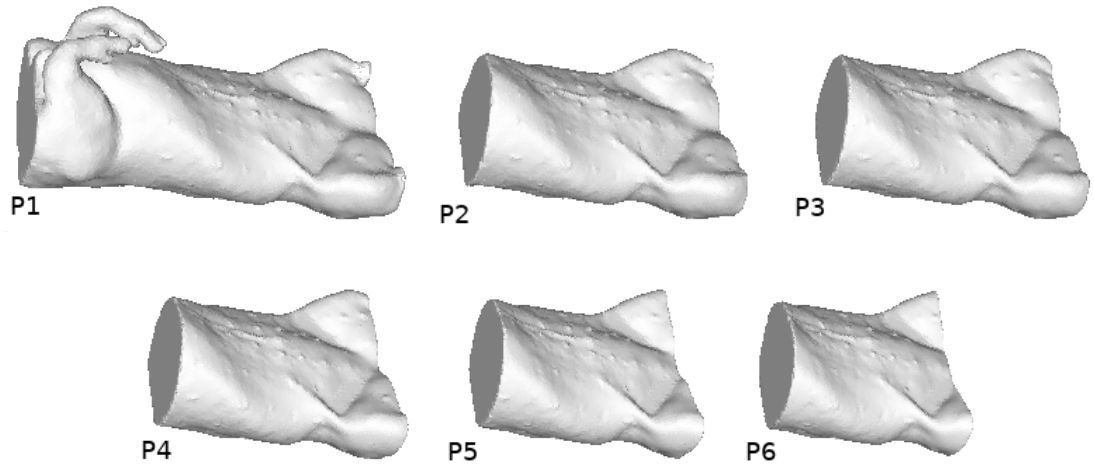

FIGURE 4.5 – Les différentes tailles de section utilisées pour le recalage.

4.8.2 Résultats expérimentaux

Résultats quantitatifs

Les résultats obtenus pour différents types de recalage considérés (recalage rigide, déformable avec modèle homogène, hétérogène, et avec prise en compte des collisions) sont présentés dans la Figure 4.6. Le premier graphique regroupe les configurations ayant des déformations de faible intensité (obtenues avec des positions dorsales), tandis que le deuxième regroupe les fortes déformations (obtenues avec des positions latérales). Tous les marqueurs sont utilisés pour obtenir cette erreur. Le troisième graphique considère quant à lui, uniquement les marqueurs présents au niveau de l'interface entre le modèle et la table. Nos travaux sont ainsi les premiers à réaliser une telle comparaison, permettant de vérifier l'importance de certaines modélisations (la prise en compte de la rigidité des os et la prise en compte des collisions avec la table).

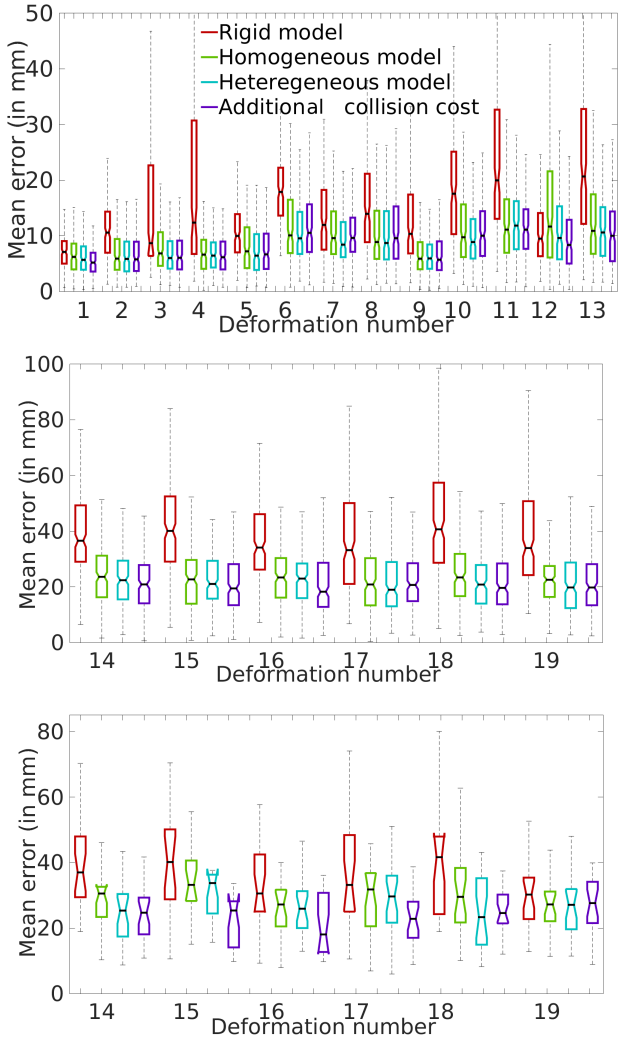

FIGURE 4.6 – Boîtes à moustaches (avec médiane, quartiles et valeurs extrêmes) présentant les erreurs obtenues pour différentes méthodes de recalage dans différentes configurations. Pour les premières figures, tous les marqueurs sont utilisés pour calculer l'erreur pour de faibles déformations (haut) et des déformations importantes (centre). Pour la figure du bas, seuls les marqueurs présents au niveau de la table sont considérés.

Pour de faibles déformations, on observe ainsi une légère amélioration du recalage lorsque nous utilisons notre méthode de recalage déformable plutôt qu'un recalage rigide. On remarque de plus une bien plus grande disparité des erreurs lorsque l'on utilise un recalage rigide. En effet, bien que faible, la déformation du modèle existe, et n'est pas négligeable en particulier au niveau de l'interface avec la table. Ainsi, l'erreur en

ces endroits pour le recalage rigide sera relativement élevée. Sur l'ensemble du modèle, une amélioration de quelques millimètres est tout de même notable entre les résultats obtenus avec notre algorithme de recalage déformable et ceux obtenus avec un recalage rigide. On observe de plus une très légère amélioration des résultats lors du passage d'un modèle biomécanique homogène à un modèle hétérogène. De façon similaire, l'utilisation d'un coût permettant de prendre en compte les collisions augmente elle aussi très légèrement la précision du recalage. Les améliorations observées sont cependant très faibles dans les deux cas et sont de l'ordre du millimètre. Pour les déformations plus importantes, les observations sont similaires mais les améliorations observées sont plus grandes. Ainsi, une différence de l'ordre du centimètre est observée entre les résultats obtenus avec notre recalage et ceux obtenus avec un recalage rigide. Les améliorations dues à l'utilisation d'un modèle biomécanique hétérogène et des coûts permettant d'empêcher les collisions sont encore une fois de l'ordre du millimètre. Cependant, dans le dernier graphique de la Figure 4.6, on observe que pour les points situés au niveau de l'interface avec la table, l'amélioration engendrée par l'utilisation d'un modèle hétérogène et par la gestion des collisions avec la table, est plus importante et est de l'ordre de quelques millimètres. Lorsque l'on utilise notre méthode avec un modèle biomécanique hétérogène et lorsque la gestion des collisions est activée, on obtient une erreur médiane d'environ 8 millimètres pour des déformations faibles et d'environ 2 centimètres pour les déformations fortes. Une telle erreur est ainsi trop importante pour des applications précises d'aide à l'insertion d'aiguilles radiofréquence ou de biopsie (bien que notre méthode permette tout de même de donner de bonnes impressions sur les zones à éviter) et pour le suivi de dose d'irradiation. Néanmoins, pour des applications d'échographie assistée par la robotique, ne nécessitant pas un recalage extrêmement précis, notre méthode est bien adaptée.

La Figure 4.7 présente quant à elle les résultats du recalage (recalage déformable utilisant un modèle biomécanique hétérogène et un coût permettant de gérer les collisions avec la table) pour des modèles construits avec les différentes sections de l'abdomen montrées en Figure 4.5. Dans ces résultats, P1 est la plus grande section mesurant 80 cm de long, et P6 est la plus petite mesurant 35 cm.

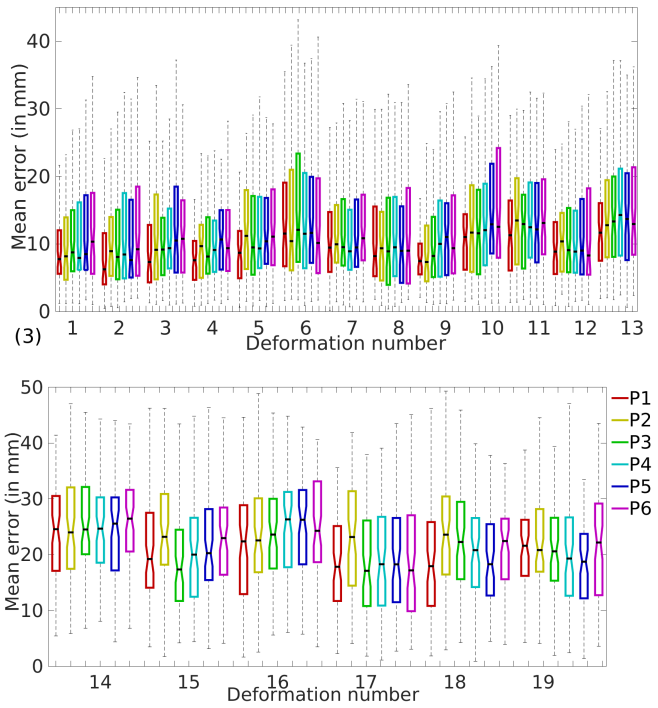

FIGURE 4.7 – Boîte à moustaches présentant les erreurs obtenues pour différents modèles générés à partir de différentes tailles de section d'abdomen. P1 est la plus grande section et P6 est la plus petite.

On remarque ici que pour la plupart des configurations, une légère détérioration de la précision du recalage (de l'ordre de quelques millimètres) apparaît lorsque l'on diminue la taille de la section de l'abdomen considérée. Cela est dû, en grande partie, à la réduction des informations géométriques (telle que la présence de pattes disparaissant entre P1 et P2) présentes dans les données et nécessaires pour rendre le problème bien contraint. Néanmoins, ces différences restent relativement faibles autant pour les déformations faibles que pour les déformations fortes. Les conclusions vis-à-vis de cette expérimentation sont donc à relativiser. Il n'est ainsi pas possible de donner une taille limite pour la section d'abdomen considérée à partir de laquelle le modèle biomécanique généré ne sera plus capable de se recaler correctement. Les seules conclusions sont ainsi que la taille de la section considérée pour générer le modèle biomécanique influe sur la précision du recalage et qu'il est toujours mieux de privilégier de grande sections, comportant plus d'informations géométriques aidant à contraindre le recalage.

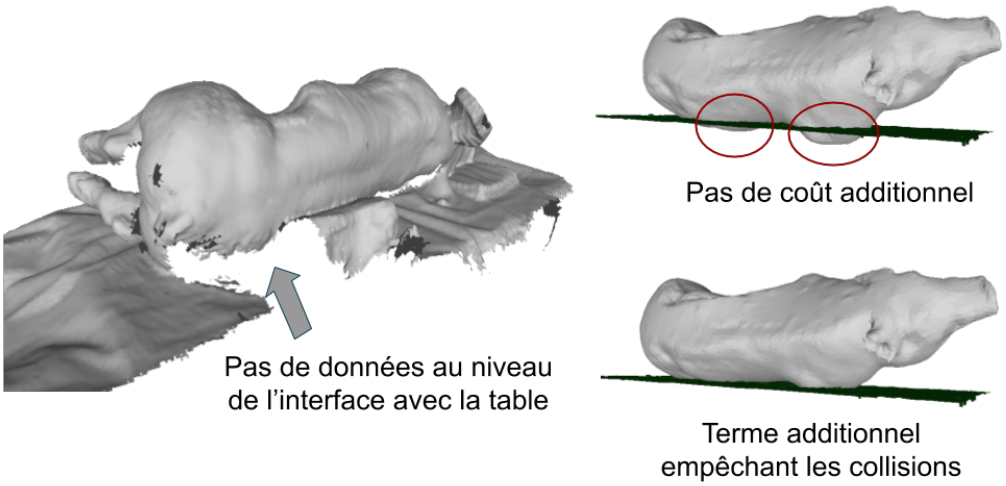

FIGURE 4.8 – Recalage final avec et sans gestion des collisions.

Présenté dans la Figure 4.8, on voit plus précisément l'effet du terme permettant de gérer d'éventuelles collisions avec la table pour un exemple donné. Sans coût additionnel, le modèle passe à travers la table, tandis qu'avec ce coût, il n'entre pas en collision avec cette dernière. La méthode fonctionne ainsi comme l'on pouvait s'y attendre. Néanmoins, on voit ici que la région passant à travers la table est de relativement petite taille, ce qui explique alors la faible amélioration observée du recalage lorsque l'on utilise cette détection de collision. L'erreur est trop localisée pour apparaître clairement dans les résultats quantitatifs.

Résultats qualitatifs : application à l'aide à l'insertion d'aiguilles radiofréquences

Afin de tester notre recalage, celui-ci a été implémenté au sein d'une application d'aide à l'insertion d'aiguilles radiofréquences. Le but de cette étude était alors de tester s'il était possible d'utiliser notre algorithme en conditions réelles en salle d'opération. Pour cela, on considère un dispositif composé d'une telle aiguille et d'une sonde échographique, servant entre autres à donner un visuel alternatif facilitant le planté d'aiguille. Ce dispositif doit alors être recalé (de façon rigide) à notre scène, en plus de notre modèle de patient. Ainsi, quelques temps avant la procédure, des acquisitions

d'images TDM sont réalisées. Les organes sont alors segmentés et un modèle biomécanique de l'abdomen du patient est construit. Le jour de l'intervention, ce modèle doit alors être recalé à la position prise par le patient sur la table d'opération comme cela est décrit dans les sections précédentes de ce chapitre. Plusieurs caméras entrent alors en jeu. Une première caméra RGB-D est tout d'abord utilisée pour scanner la surface de la peau du patient, et ainsi permettre la construction d'une surface cible à notre recalage. De plus, au niveau de la sonde échographique, des marqueurs infrarouges (réfléchissants) sont fixés et sont observés par une seconde caméra RGB-D disposée au dessus de scène. Il est alors possible de segmenter automatiquement et de façon très rapide les marqueurs infrarouges dans les images provenant de celles ci. La position de la sonde dans le repère associé à cette caméra peut ainsi être facilement calculée à partir de ces marqueurs [203]. À noter qu'ici, la caméra sert uniquement à voir les marqueurs en question et non la scène directement. Le problème de calcul de pose à partir de ces images est un problème très bien contraint et peu sensible au bruit. Les limitations exposées plus tôt dans ce chapitre concernant l'utilisation d'une caméra au plafond ne s'appliquent pas entièrement ici (la position de la sonde extraite reste très précise). Enfin, une caméra RGB positionnée au niveau du plafond est utilisée en tant que vue externe. Des points en association sont alors cliqués (dans cette première version présentée) sur les images provenant de la caméra observant le dispositif et sur la reconstruction obtenue à partir des images provenant de la numérisation réalisée avec l'autre caméra RGB-D. Tout est ainsi repositionné (de façon rigide) dans le repère de la caméra au plafond. De plus, ces points servent aussi à donner une initialisation à notre recalage. Le recalage non rigide, décrit dans ce chapitre est ensuite réalisé, et les organes segmentés au début de la procédure sont déformés en conséquence. La position de la sonde étant connue dans le monde de la caméra RGB plafond et tout étant recalé dans cette vue, il est alors possible d'obtenir les modèles dans le monde lié à la sonde échographique. De plus, l'aiguille radiofréquence étant visible dans les images échographiques, il est ainsi possible de guider cette dernière grâce au recalage du modèle dans celles-ci.

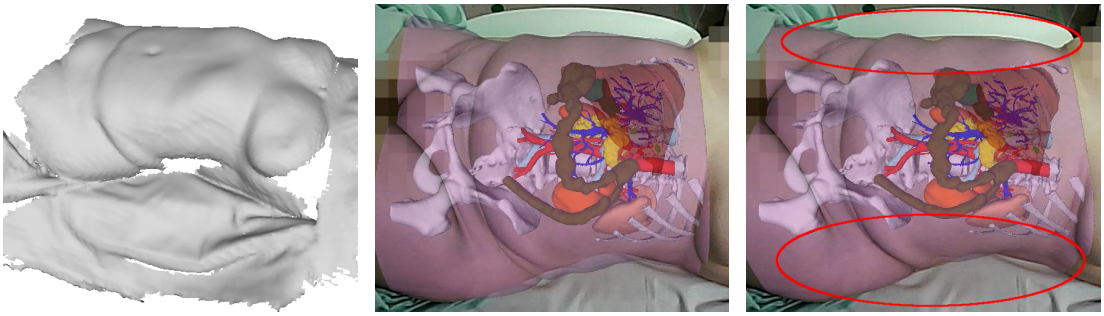

FIGURE 4.9 – Résultat du recalage obtenu avec un patient sur la table d'opération. La numérisation RGB-D (gauche), la transformation rigide (centre) et le recalage déformable (droite).

Le résultat est montré dans la Figure 4.9, où on observe bien une meilleure correspondance du contour du patient après recalage déformable (au niveau des cercles rouges).

FIGURE 4.10 – Aide à l'insertion d'aiguille radiofréquence.

Un autre rendu pour cette application est montré en Figure 4.10, où on observe à l'écran les différents organes après déformation, ainsi que l'aiguille et le plan correspondant à la coupe échographique considérée.

4.9 Automatisation du recalage rigide initial

Comme décrit dans la section précédente, des points doivent être cliqués dans les différentes images ainsi que dans le modèle afin de faire correspondre les systèmes de coordonnées des deux caméras mais aussi d'initialiser notre recalage non rigide. Cela peut alors être particulièrement gênant, ne serait-ce que pour des raisons de temps. À noter que lorsqu'on se propose de cliquer des points dans la reconstruction dans la section précédente, cela est équivalent (non pas en terme de temps de réalisation mais en terme de résultats) à repérer des points dans les différentes images à l'origine de celles-ci et à trianguler leur position 3D dans la reconstruction. Afin d'automatiser ces associations, nous utilisons OpenPose [24], une librairie permettant de récupérer la pose d'êtres humains dans des images 2D sous la forme de squelettes (Figure 4.9).

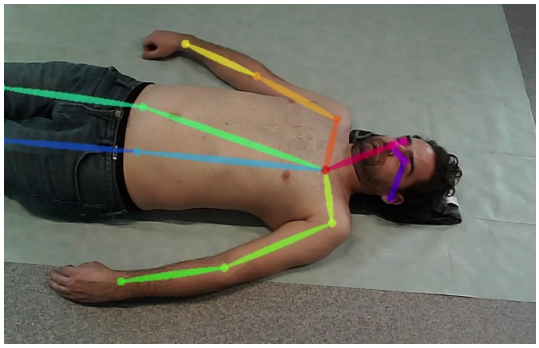

FIGURE 4.11 – Détection de squelette utilisant OpenPose.

De ces squelettes, nous extrayons cinq points d'ancrage (épaules gauche et droite, cou, hanches gauche et droite). Une telle détection dans les images des deux caméras RGB-D permet de réaliser notre alignement des deux systèmes de coordonnées de nos caméras. Cependant, cela n'est pas suffisant pour initialiser le recalage non rigide. En effet, ces mêmes points doivent être aussi repérés dans le modèle biomécanique. Cependant, contrairement au clic de points nécessaire dans notre première version, cette étape doit être réalisée une unique fois à la création du modèle (et non à chaque recalage). Ces points peuvent alors être mis en correspondance, et une pose rigide, initialisant notre recalage non rigide peut être calculée (Figure 4.12). Ces tests n'ont cependant pas été évalués rigoureusement à l'heure actuelle et seuls des résultats qualitatifs sont disponibles. Dans de futurs travaux, cette méthode devra être testée quantitativement.

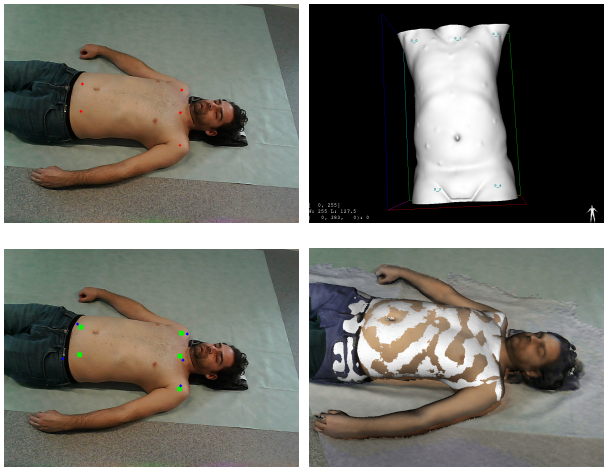

FIGURE 4.12 – Recalage de posture utilisant OpenPose. De gauche à droite et de haut en bas : points de repère dans l'image 2D, dans le modèle volumique, points en correspondance (bleu et vert) et recalage final.

4.10 Conclusion

Dans ce chapitre, nous avons présenté une solution au problème de recalage de posture. Un modèle volumique de l'abdomen du patient, construit à partir d'images préopératoires TDM ou IRM, est ainsi recalé à une reconstruction de la surface de la peau du patient sur la table d'opération. Afin d'améliorer la qualité du recalage, la table est modélisée et la collision avec celle-ci est prise en compte. La méthode proposée est testée de deux façons. Une première évaluation qualitative est ainsi réalisée sur des données patient, en utilisant notre recalage pour une application d'aide à l'insertion d'aiguille radiofréquence. Une deuxième évaluation quantitative est faite à l'aide de données animales. Un porc euthanasié sur la peau duquel ont été fixé des marqueurs métalliques, permettant d'obtenir une vérité terrain, est ainsi utilisé en tant que cible du recalage. Cette évaluation permet ainsi de mettre en évidence l'importance de la prise en compte des collisions avec la table. Elle permet aussi de renseigner l'influence de la taille de la section d'abdomen utilisée pour construire le modèle à recaler sur la précision du recalage, et donc d'avoir une idée sur la taille minimale à utiliser pour cette section, en pratique pour de telles applications. Le jeu de données utilisé pour cette seconde évaluation est très précieux car il permet de calculer densément l'erreur en surface

du modèle pour des algorithmes de recalage postural. Ces données d'évaluation seront rendues publiques. Les erreurs obtenues avec ces données pour notre algorithme de recalage sont de l'ordre de 8 millimètres pour des déformations faibles et de l'ordre de 2 centimètres pour des déformations plus importantes. Une telle précision ne rend ainsi pas possible son utilisation pour l'aide à l'insertion d'aiguilles radiofréquence, même si elle permet d'avoir une bonne idée de l'emplacement des structures à éviter. L'utilisation de données provenant de sonde échographique permettant de mettre à jour le recalage en temps réel, pourraient cependant être utilisées afin d'améliorer la précision du recalage [103, 105], le rendant ainsi utilisable dans ce contexte. La méthode présentée n'est pas non plus utilisable pour des applications de suivi de dose d'irradiation pour les mêmes raisons de précision. Cependant, elle peut être utilisée pour des applications d'échographie assistée par la robotique où une grande précision du recalage n'est pas capitale. Une limitation de ce travail correspond à la non prise en compte de la respiration du patient sur la table. Une amélioration de nos travaux consisterait ainsi à simuler la respiration de ce dernier et mettre à jour le recalage conformément à cette simulation [9, 80, 8]. Enfin, il est à noter que les résultats qualitatifs des travaux présentés dans ce chapitre ont été réalisés avec des données animales. Une meilleure évaluation sur des données patients est ainsi indispensable pour la suite de nos expérimentations.

Chapitre 5

Modélisation de l'éclairage en cœlioscopie

Dans ce chapitre, nous nous intéressons aux particularités de l'éclairage au sein de la scène cœlioscopique, ainsi qu'aux raisons pour lesquelles sa modélisation peut être extrêmement bénéfique pour la cœlioscopie assistée par ordinateur. Une connaissance de l'illumination de la scène permet en effet d'extraire des informations précieuses au niveau de l'ombrage présent dans les images. Dans la littérature, ces informations ont ainsi déjà prouvé leur efficacité pour résoudre des problème de reconstruction 3D et de recalage de données [14, 98, 34, 33, 140, 55]. Dans le contexte de la cœlioscopie, une bonne connaissance de l'illumination peut être obtenue par une modélisation de la source de lumière associée au cœlioscope. Néanmoins, bien qu'un certain nombre de modèles aient été proposés, aucun consensus n'existe concernant le meilleur modèle permettant d'estimer au mieux l'illumination de la scène. Différents modèles de lumière, plus ou moins complexes, sont ainsi présentés dans ce chapitre. Parmi ceux-ci, des modèles ponctuels sont discutés mais aussi des modèles plus spécifiques, prenant en compte la forme du motif lumineux du cœlioscope. Des méthodes de calibrage, permettant d'extraire leurs paramètres sont explicitées. Une étude comparative entre les modèles est enfin réalisée, utilisant des images de mires de calibrage, et étudiant la différence d'intensité estimée et réelle des pixels considérés. Il est ainsi montré que des modèles relativement simples peuvent s'avérer particulièrement performants. Ce chapitre est basé sur une partie de notre contribution [130] publiée dans IJCARS pour la conférence internationale IPCAI 2020.

5.1 Introduction

L'ombrage (*shading)* constitue l'un des indices fondamentaux utilisés par la vision humaine afin de ressentir la profondeur d'une scène [145]. Il correspond à la variation de l'irradiance d'un matériau en fonction de la différence entre la direction de l'éclairage principal et à la normale à la surface observée. Il n'est ainsi pas à confondre avec l'analyse des ombres portées qui bien que renseignant aussi sur la géométrie de la scène correspond à un tout autre problème. Une tâche importante en vision par ordinateur a ainsi été de s'appuyer sur ces observations de l'ombrage de la scène et de les mettre au service de la détection de relief dans les images. Les ombrages dans les images sont ainsi automatiquement extraits et analysés afin d'en déduire une information sur la profondeur dans la scène. Ce genre de problème définit ainsi tout un panel de problèmes dits photométriques, où l'on recherche le lien entre éclairage de la scène, géométrie et réflectance de la surface observée, et enfin propriétés de la caméra utilisée. Plus précisément, on s'intéresse, pour une scène filmée, à la façon dont le rayonnement lumineux se propage dans la scène : comment ce dernier se réfléchit sur la surface observée, et comment cette lumière réfléchie arrive jusqu'au capteur de la caméra. En d'autres termes, on étudie la façon dont se forme l'image de la scène sur le capteur, et on cherche à étudier le pipeline de formation de l'image. Inverser le processus de formation de l'image permet ainsi d'obtenir des informations sur la surface observée et est à l'origine de tout algorithme de reconstruction basé sur la photométrie de la scène [155, 140, 55, 33, 143]. Cette approche est ainsi très différente des approches basées sur l'étude des mouvements de la caméra dans la scène (SfM [167, 192, 41] ou SLAM [168, 42, 90, 99, 118]) qui ne modélisent pas ces caractéristiques photométriques. Ainsi, contrairement à ces méthodes basées sur le mouvement, ne pouvant être utilisées qu'avec des textures relativement importantes, et générant ainsi des reconstructions dont la densité dépend fortement de la quantité de texture détectée, les approches photométriques produisent toujours des reconstructions denses. Les premier travaux considérant cette approche furent les algorithmes de Shape from Shading (SfS) [78] (utilisant une unique image), sous les hypothèses d'un éclairage et un albédo de la surface considérée connu. Ils ont été suivis plus tard par les algorithmes de stéréo-photométrie [210], considérant le problème à plusieurs vues.

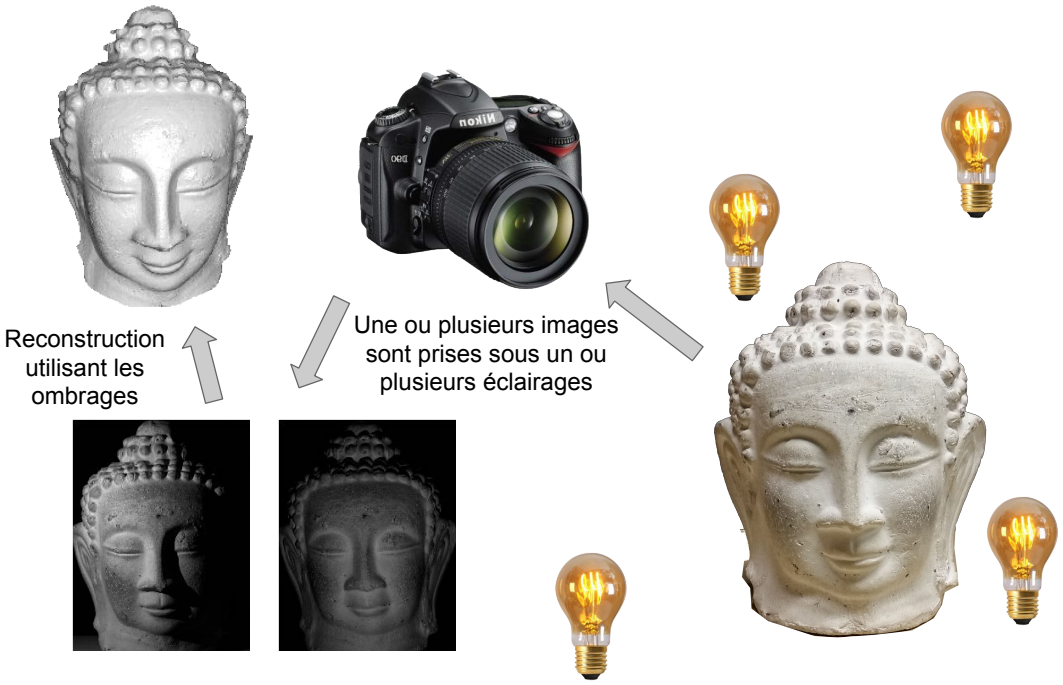

FIGURE 5.1 – Illustration du problème de reconstruction de surface en utilisant les ombrages de la scène. La reconstruction représentée est réalisée par stéréo-photométrie et correspond aux travaux présentés dans [155].

Néanmoins, l'utilisation des ombrages ne se restreint pas au domaine de la reconstruction 3D, et de nombreuses utilisations de ces ombrages sont possibles et sont discutées dans l'état de l'art associé à ce chapitre. En chirurgie cœlioscopique, les ombrages de la scène sont particulièrement marqués du fait de la proximité de la source lumineuse (présente sur le cœlioscope) et la surface observée, ce qui fait de ce contexte un candidat de choix pour l'utilisation de techniques basées sur la photométrie. Ainsi, par exemple, ces indices peuvent même être utilisés afin de contraindre fortement des problématiques de recalage déformable [14, 98, 34], en l'absence de contraintes denses présentes dans les images, est un problème sous-contraint. De plus, dans le cadre de la cœlioscopie, la source lumineuse, étant attachée à la caméra, et étant donc fixe par rapport à celle-ci, sa modélisation s'en trouve alors facilitée. L'éclairage est ainsi bien plus contraint que dans le cas général. De nombreux modèles de lumière ont été proposés dans la littérature. Cependant, aucune comparaison sérieuse et quantitative de la qualité de l'estimation de l'éclairage n'a jamais été réalisée. C'est ce constat qui a motivé les expérimentations

présentées dans ce chapitre. Des modèles de lumière sont ainsi présentés et discutés. Une méthode permettant de réaliser leur étalonnage est ensuite énoncée. Enfin, une comparaison quantitative entre les éclairages estimés par ces modèles est ensuite réalisée, permettant ainsi de donner des conclusions vis-à-vis des modèles lumineux à préférer.

5.2 État de l'art

Utilisation des informations photométriques en vision par ordinateur

Chercher directement à reconstruire la surface observée à partir d'une seule image est ce qu'on appelle le SfS [78] dont le principe a été établit par Horn en 1970. Ce problème consiste à calculer, pour chaque pixel d'une image, son information de profondeur, sous couvert de la connaissance de l'éclairage en ce point et de l'albédo. Dans le cas général, il s'agit d'un problème mal posé qui ne peut être résolu sans ajout de contraintes supplémentaires. Utiliser plusieurs images ayant des poses de caméra identiques mais des éclairages différents correspond quant à lui au problème de la stéréo-photométrie [210] qui fut, introduit par Woodham en 1980. Dans ces deux cas, le but est alors de récupérer les normales à la surface en s'appuyant sur l'ombrage présent dans la scène. Les normales sont ensuite intégrées pour récupérer la profondeur en chacun des points. Néanmoins, des ambiguïtés apparaissent lorsque l'on considère une source *à l'infini* (dont l'intensité lumineuse ne dépend pas de la distance à laquelle elle se trouve). Ainsi, une ambiguïté d'échelle persiste dans la reconstruction, ainsi aussi que ce qu'on l'on appelle une ambiguïté des bas reliefs [12] (correspondant à une variation de profondeur globale pouvant être à la fois convexe ou concave). De plus, la stéréo-photométrie suppose en général un éclairage variable, connu et contrôlé [155]. On parle de stéréo-photométrie calibrée. Elle n'est donc en pratique, utilisable que dans des conditions de laboratoire. Le cas non calibré n'est quant à lui jamais entièrement soluble. En effet, la non-connaissance des sources, leurs non-directionnalités, et les autres objets réfléchissants de la scène se comportant comme des sources secondaires, rendent le problème extrêmement mal posé. Une variante de la stéréo-photométrie existe aussi et se propose d'utiliser une seule image RGB mais plusieurs lampes colorées permettant ainsi de séparer les différents canaux de l'image et de revenir au problème de la stéréo-photométrie. Une telle

méthode s'appelle le Shape from Color (SfC) [54]. Au contraire de la stéréo-photométrie et de ses dérivées, utiliser plusieurs images, en ayant cette fois-ci des poses de caméra différentes mais un éclairage identique équivaut au problème du SAMVS [104] qui est, au contraire des techniques précédentes, une technique relativement récente. Pour cette approche, les poses relatives entre les caméras doivent cependant être connues ou déterminées préalablement. Cette problématique peut être directement résolue pour des caméras à plusieurs objectifs [96], mais de telles caméras ne sont pas courantes et sont ainsi peu utilisées. En général, une autre technique de reconstruction basée sur le calcul de correspondances entre les images (comme les algorithmes de SfM ou de SLAM) est utilisée pour obtenir ces poses. En plus de ces méthodes de reconstruction, l'analyse de ces ombrages fut aussi récemment proposée afin de mieux contraindre des problèmes de recalage de données [14, 98, 34, 58]. Cependant, toutes ces techniques présupposent en général des a priori forts vis-à-vis de l'éclairage. Ainsi, nombre de ces travaux ne considèrent que des sources lumineuses directionnelles, ou des harmoniques sphériques, correspondant au cas de sources lumineuses positionnées *à l'infini* [109, 58]. Cela équivaut au cas de la majorité des scènes de jour, en extérieur.

Utilisation de l'ombrage dans le contexte de la chirurgie assistée par ordinateur

En chirurgie cœlioscopique, la seule source primaire de lumière provient du cœlioscope lui même. Deux types de sources lumineuses sont actuellement utilisées pour ces derniers. Les cœlioscopes les plus courants disposent d'une source de lumière externe dont le rayonnement lumineux est redirigé vers l'extrémité de l'instrument grâce à une fibre optique.

FIGURE 5.2 – Source de lumière externe Storz pour endoscope.

L'intensité lumineuse de telles sources peut ainsi être très élevée et très peu variable. Le second type de cœlioscopes dispose de Diodes Électro-Luminescentes (LEDs), directement intégrées sur leurs extrémités. L'intensité lumineuse obtenue ne peut alors être aussi puissante que pour les premiers modèles discutés. Cependant, le système a comme bénéfice d'être tout-en-un, il est donc en général moins cher. Dans tous les cas, la source lumineuse située à l'extrémité du cœlioscope est fixe par rapport au centre d'optique de la caméra. Déterminer sa position relative par rapport à la caméra n'a donc à être fait qu'une seule fois puisqu'elle restera inchangée dans toutes les images considérées, prises par cette caméra. Cette forte relation entre position de la source lumineuse et position du centre optique fait de la cœlioscopie un environnement de choix pour les problèmes utilisant les ombrages.

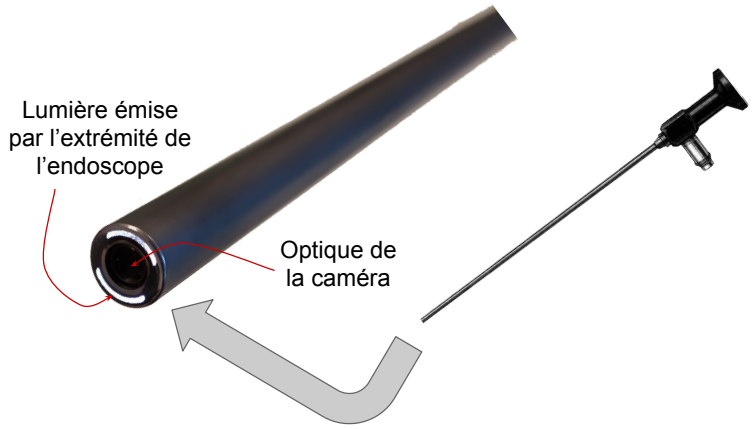

FIGURE 5.3 – Lien fort entre position de la lumière et centre optique de la caméra pour un endoscope.

De plus, dans le cas des organes faiblement texturés tel que le foie, les ombrages sont quelquefois les seuls indices visuels présents de manière dense dans l'image. Considérer ces informations apparaît de plus comme extrêmement utile pour la cœlioscopie assistée par ordinateur. Son utilité est en effet multiple. Tout d'abord, il est déjà possible d'utiliser les techniques présentées précédemment (SfS et SAMVS) pour obtenir des reconstructions per-opératoires de l'organe considéré [33, 140, 55]. Ces reconstructions peuvent ensuite être utilisées à des fin diagnostiques ou dans le pipeline de recalage présenté en introduction. Couramment, des techniques de SfM ou SLAM sont utilisées pour

ces reconstructions [117, 199, 195, 178, 118]. Néanmoins, certains organes étant faiblement texturés (le foie par exemple), les reconstructions obtenues sont souvent de faible qualité (peu denses, bruitées et/ou possédant des informations de texture peu fiables). L'utilisation des ombrages de la scène permet alors d'obtenir des reconstructions de bien meilleure qualité [33, 140, 55]. De plus, dans les méthodes de reconstruction plus classiques (basées sur les appariements de points entre images, c'est-à-dire sur des méthodes de type SfM et SLAM), une ambiguïté sur l'échelle de la reconstruction perdure. En effet, considérer l'image d'un objet revient à considérer la projection centrale de sommet F (centre optique de la caméra) de cet objet sur un plan P (appelé plan image). Multiplier la taille de l'objet par un certain facteur tout en multipliant aussi la distance entre l'objet et F par ce même facteur ne change ainsi pas la projection considérée, d'où l'existence d'une ambiguïté d'échelle. Ceci est illustré dans la Figure 5.4.

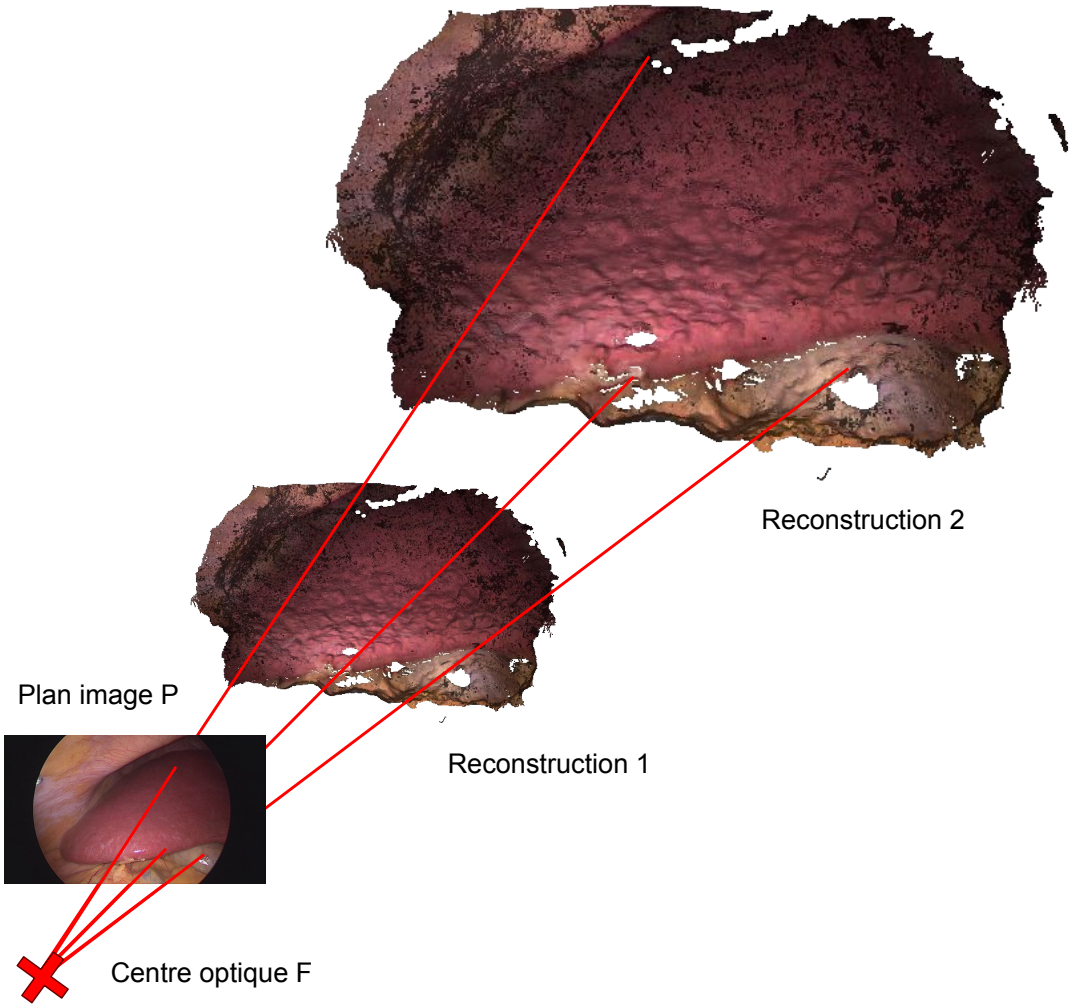

FIGURE 5.4 – Ambiguïté d'échelle pour des reconstruction monoculaires.

Énoncé plus simplement : il n'existe pas de différence entre la vue d'un grand objet vu de loin et la vue d'un plus petit objet observé de plus près. Néanmoins, il est quelquefois possible d'utiliser les informations sur l'éclairage de la scène pour obtenir des reconstructions sans ambiguïté. En particulier, certains travaux proposent d'utiliser la décroissance de la puissance lumineuse au fur et à mesure que l'on s'éloigne de la source pour obtenir de telles reconstructions [108]. Une autre utilité de ces informations d'ombrage est enfin de les utiliser pour des problèmes de recalage non-rigide mono-vue [43].

Cela correspond par exemple, dans le pipeline de réalité augmentée discuté en introduction, à la phase dite de *tracking*. En effet, pour des modèles d'objets à caractère faiblement texturé, il n'est possible d'extraire qu'un faible nombre d'appariements entre points du modèle et points observés, ce qui n'engendre au final que des contraintes de déformation éparses sur le modèle considéré. Pour de fortes déformations, il n'est cependant possible de recouvrer entièrement la forme de l'objet qu'à partir de contraintes denses sur la surface du modèle. Celles-ci peuvent ainsi provenir de l'utilisation des ombrages. De telles approches ont ainsi déjà montré leur efficacité pour le recalage de modèles extrêmement déformables et à faible texture tels que des feuilles de papier [58] ou des visages humains [109]. Cependant, bien que ces conditions (fortes déformations et faible texture) soient celles que l'on retrouve dans certaines interventions chirurgicales telles les cœlioscopies hépatiques, de telles approches ne sont que peu utilisées dans ce contexte [34]. Enfin, la modélisation de l'éclairage dans la vue cœlioscopique peut aussi être utilisée afin d'améliorer la qualité des simulateurs chirurgicaux [154, 137].

Modélisation de l'éclairage en cœlioscopie

En dépit de l'éclairage particulier en cœlioscopie, permettant d'avoir des ombrages très marqués dans les images, ces informations ne sont encore que très rarement utilisées dans le contexte de la cœlioscopie. Pour que de telles applications existent, l'éclairage de la scène, c'est-à-dire, l'éclairage induit par le cœlioscope doit être modélisé. Différentes visions s'opposent pour la modélisation de l'éclairage. Certains travaux proposent tout d'abord d'utiliser des modèles de lumière relativement simples. Ainsi par exemple, dans [14] un modèle de source ponctuel est proposé, dont le centre de lumière coïncide avec le centre optique de la caméra. Ce modèle est alors utilisé pour le recalage d'un modèle de foie, construit à partir d'images TDM per-opératoires, à des images cœlioscopiques elles aussi per-opératoires. Un modèle de lumière similaire est aussi utilisé dans [98] afin de recaler cette fois-ci un modèle pré-opératoire à des images cœlioscopiques per-opératoires. D'autres travaux proposent d'utiliser des modèles de lumière plus complexes. Certains travaux considèrent ainsi des modèles polynomiaux ou même des modèles non paramétriques et les utilisent pour reconstruire des surfaces per-opératoires de foie [33, 34]. Cependant, la préférence de l'utilisation de certains modèles, c'est-à-dire la raison pour laquelle il est préférable d'utiliser un modèle plutôt qu'un autre, n'est jamais réelle-

ment discutée dans la littérature. C'est ce constat qui motive nos travaux sur l'étude de l'éclairage en cœlioscopie.

5.3 Équation de rendu

Dans ce chapitre et dans le chapitre suivant, nous considérons des images RGB. Chaque canal apporte ainsi ses propres contraintes au problème d'ombrage considéré. Cependant, afin de ne pas trop complexifier nos équations, nous considérerons dans nos notations des images à un seul canal. On exprime ainsi le lien entre l'intensité d'un pixel de l'image $I(p)$ et les propriétés photométriques et géométriques de l'objet observé. L'équation générale sur les ombrages est alors :

$$I(p) = \mathcal{C} \circ F(\rho(x), \sigma(x), \mathbf{n}(x)). \tag{5.1}$$

Dans cette équation, le point image p correspond à un point 3D x de la surface de l'objet observé. $\rho(x)$ est l'albédo au point considéré, une propriété propre au matériel considéré correspondant à la proportion de lumière réfléchie par la surface. $\sigma(x)$ est le vecteur d'éclairage correspondant au flux lumineux reçu par la surface. $\mathbf{n}(x)$ est la normale en x. F est le modèle de réflectance donnant l'irradiance de la scène. Et finalement, $\mathcal{C}$ est la Fonction de Réponse de la Caméra (CRF). Les inconnues de ce problème et la façon de résoudre cette équation dépendent alors du problème considéré et des hypothèses posées.

5.4 Modèle de caméra

5.4.1 Modèle géométrique

La caméra est modélisée géométriquement par un modèle de caméra *en trou d'épingle*, c'est-à-dire un modèle de caméra perspectif, ainsi que 8 termes modélisant la distorsion (k_1, k_2, k_3, k_4, k_5, et k_6 les coefficient radiaux et p_1 et p_2 les coefficient tangentiels) [72]. Par modèle projectif, on entend un plan, appelé plan image et un point F appelé centre optique.

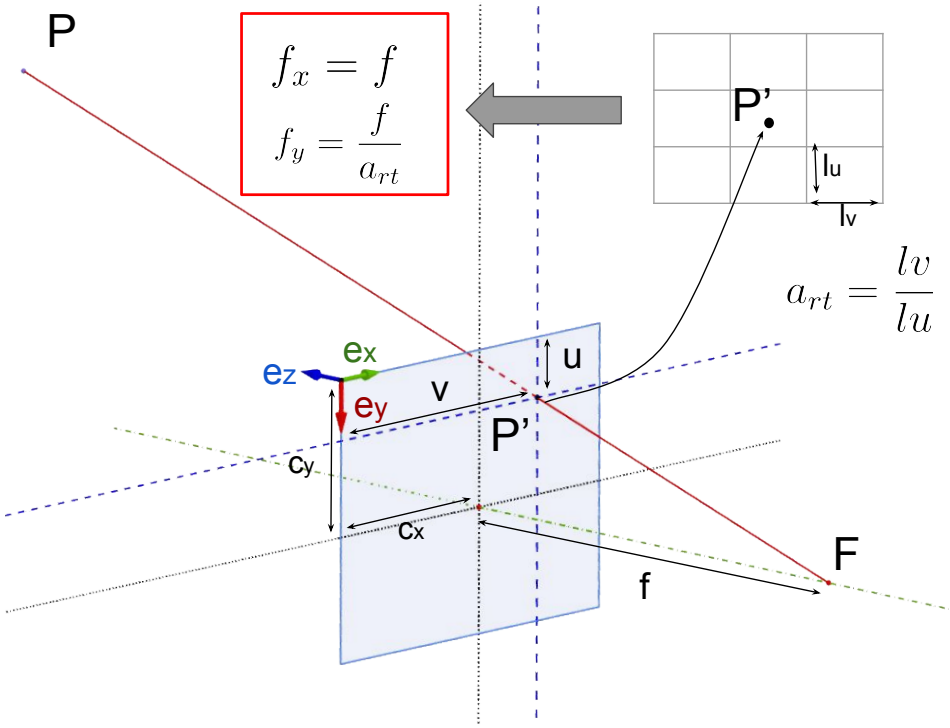

FIGURE 5.5 – Modèle de la caméra en trou d'épingle.

Tout point P du demi-espace situé *devant le plan* se projette ainsi sur le plan par projection centrale de sommet F_c. Une certaine délimitation de ce plan correspond ainsi à l'image de la caméra. Cependant, les droites de l'espace ont tendance à apparaître légèrement courbes dans les images à cause des imperfections de la lentille. Elles doivent ainsi être rectifiées grâce à des paramètres de distorsion, afin de correspondre réellement à notre modèle projectif. Les coordonnées du point exprimé en coordonnées caméra par

(x, y, z), projeté dans l'image (u, v) s'expriment alors comme suit :

$$\begin{aligned}
x' &= x/z \\
y' &= y/z \\
x'' &= x' \frac{1 + k_1 r^2 + k2r^4 + k_3 r^6}{1 + k_4 r^2 + k5r^4 + k_6 r^6} + 2p_1 x' y' + p_2(r^2 + 2x'^2) \\
y'' &= y' \frac{1 + k_1 r^2 + k2r^4 + k_3 r^6}{1 + k_4 r^2 + k5r^4 + k_6 r^6} + p_1(r^2 + 2y'^2) + 2p_2 x' y' \\
u &= f_x x'' + c_x \\
v &= f_y y'' + c_y,
\end{aligned} \tag{5.2}$$

avec f_x et f_y les distances focales suivant x et y, et (c_x, c_y) le point principal de la caméra.

5.4.2 Modèle photométrique

En terme de photométrie, la CRF doit aussi être modélisée. En effet, toute l'intensité lumineuse arrivant sur les capteurs de la caméra ne correspond pas directement aux intensités des pixels dans l'image. Il faut donc considérer une fonction de transfert entre l'intensité de l'image réellement observée I et l'irradiance de la scène L, c'est-à-dire l'intensité lumineuse qui arriverait sur les capteurs s'il n'y avait aucune perte due à la transition à travers le système optique. Cette fonction correspond à $\mathcal{C}$ dans l'équation (5.1) et est en général considérée comme étant un simple gain $\mathcal{C}'(p)$ dépendant uniquement de la position du point image [143, 34, 59]. Dans de nombreux travaux, $\mathcal{C}'(p)$ est supposée constante entre les vues [104, 109]. Néanmoins, pour la plupart des cœlioscopes, ce n'est pas le cas et $\mathcal{C}'(p)$ varie énormément entre les vues à cause de l'exposition automatique. Cette variation automatique de la CRF est importante pour le chirurgien car elle lui permet d'avoir la meilleure qualité d'image à tout moment. Néanmoins, cela ajoute une difficulté à l'utilisation des ombrages car $\mathcal{C}'(p)$ doit être déterminé pour chaque vue indépendamment. Ce gain $\mathcal{C}'(p)$ est usuellement décomposé en deux fonctions s et w :

$$\mathcal{C}' = w \circ s. \tag{5.3}$$

s correspond à l'atténuation en intensité induite par le passage des rayons lumineux à travers le système optique, c'est-à-dire la fonction de passage entre irradiance de la scène L et irradiance de l'image E. w correspond à la transformation du signal lumineux en signal électrique, puis en information utilisable : notre image. Généralement, on considère que s s'exprime comme il suit [48] :

$$E = s(L, \alpha) = L\frac{\pi}{4}\left(\frac{d}{h}\right)^2 cos^4\alpha \tag{5.4}$$

où α est l'angle entre l'axe optique et le *rayon image*, d est le diamètre de l'ouverture de l'objectif, et h la distance focale de la lentille utilisée. La Figure 5.6 illustre cela.

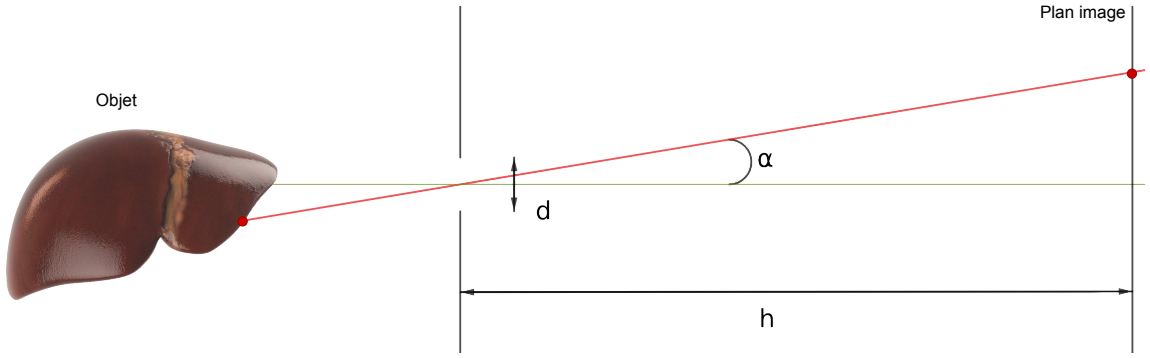

FIGURE 5.6 – Modèle de la caméra et expression de sa fonction de réponse.

On voit ainsi que le passage à travers le système optique engendre une perte linéaire d'intensité mais aussi une perte radiale. Cette atténuation radiale, aussi appelée *vignetage optique* correspond à une intensité lumineuse dans l'image lorsque l'on s'éloigne de son centre. En réalité, un autre type de *vignetage*, appelé *mécanique* existe lui aussi et est du, en particulier, à la présence de différents diaphragmes le long du système optique qui interceptent une partie des rayon lumineux [48]. Il correspond à un assombrissement bien plus abrupt mais aussi plus localisé au niveau des bords de l'image. En général, vu que cet effet reste négligeable lorsque l'on se restreint au centre de l'image, seul le vignetage optique est considéré.

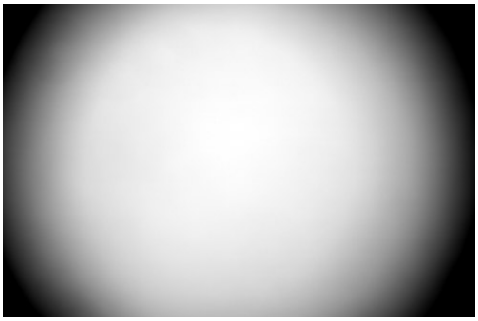

FIGURE 5.7 – Illustration du vignetage dans les images.

Concernant w, cette fonction est en général difficilement modélisable. En effet, bien que le signal électrique provenant des capteurs CCD soit analogique et directement proportionnel à l'irradiance de l'image, ce signal est ensuite converti en données numériques puis subit une normalisation pour correspondre à une image qui a visuellement du sens pour un être humain. De plus, certains filtres, comme des atténuateurs de bruits, des exhausteurs de contours, ou d'autres filtres, sont souvent présents avant ou après numérisation de l'information pour rendre l'image visuellement plus agréable. Tout ceci étant d'autant plus vrai pour des optiques de faible qualité comme ceux présents dans les téléphones portables par exemple. Cependant, pour des caméras endoscopiques, on considère ici que les effets autres que celui de normalisation peuvent être négligés. En effet, l'image étant en général occultée sur ses bords par la base de l'endoscope (aux endroits où le vignettage pourrait être important), et la région d'intérêt de l'image étant ainsi uniquement le centre de cette dernière, cet effet de vignetage n'apparaît alors que très peu. On considérera ainsi w comme linéaire, et si I est l'intensité dans l'image on a :

$$I = f(E) = KE, \tag{5.5}$$

avec K coefficient scalaire à déterminer. On considère ainsi généralement que $\mathcal{C}'(p)$ peut s'écrire :

$$\mathcal{C}'(p) = \widetilde{K} cos^4 \alpha \tag{5.6}$$

où $\widetilde{K}$ est appelée réponse linéaire et $cos^4\alpha$ réponse radiale. Dans nos travaux, nous utilisons certains modèles de lumière ayant une direction principale de propagation. Lorsque cette direction est portée par un axe similaire à l'axe optique, cette réponse

radiale est redondante avec la décroissance radiale de la source lumineuse. Pour ces raisons, et afin de bien pouvoir distinguer les différences entres les modèles comportant une direction principale de propagation et les autres, nous ne considérons ici que des réponses radiales de caméra. Si on appelle donc k l'index de l'image, on a $\mathcal{C}'_k(p) = \mathcal{C}_k$ et l'équation (5.1) peut être réécrite comme :

$$I_k(p) = \mathcal{C}_k \; L(\rho(x), \sigma(x), \mathbf{n}(x)). \tag{5.7}$$

De plus, pour des caméras couleur, on considère les trois canaux comme indépendants, et on considérera une balance des blancs (W_{br}, W_{bg}, W_{bb}) en addition de la fonction de réponse de caméra, avec $W_{br} = 1$ par défaut. Ainsi, si on considère les irradiances de la scène dans les rouges L_R, dans les verts L_G et dans les bleus L_B et les intensités des images associés I_R, I_G et I_B, on a

$$\begin{cases} I_{R,k}(p) = W_{br} \; \mathcal{C}_k \; L_R(\rho(x), \sigma(x), \mathbf{n}(x)) = \mathcal{C}_k \; L_R(\rho(x), \sigma(x), \mathbf{n}(x)) \\ I_{G,k}(p) = W_{bg} \; \mathcal{C}_k \; L_G(\rho(x), \sigma(x), \mathbf{n}(x)) \\ I_{B,k}(p) = W_{bb} \; \mathcal{C}_k \; L_B(\rho(x), \sigma(x), \mathbf{n}(x)) \end{cases} \tag{5.8}$$

5.5 Modèle de réflectance

La réflectance décrit la façon dont un matériel réfléchit la lumière incidente. La plupart des modèles considère que celle-ci peut être décomposée en deux composantes : spéculaire et diffuse. La réflexion spéculaire correspond à la lumière réémise préférentiellement autour d'une direction principale. Celle-ci crée alors des zones très localisées dans les images où l'intensité des pixels sature : des tâches spéculaires ou spécularités. La composante diffuse correspond à la lumière réfléchie de façon égale dans toutes les directions. C'est cette composante qui participe au visuel, au sens commun du terme, de la surface observée. Pour ce qui est de la composante diffuse, le modèle le plus répandu pour la modéliser est le modèle Lambertien [221]. Pour un tel modèle, la composante diffuse de l'irradiance de la scène s'écrit alors comme :

$$L_d = \rho \; \sigma \cdot \mathbf{n}, \tag{5.9}$$

où ρ est l'albédo, σ le vecteur d'éclairage, $\mathbf{n}$ la normale à la surface, et L_d la partie diffuse de l'irradiance de la scène.

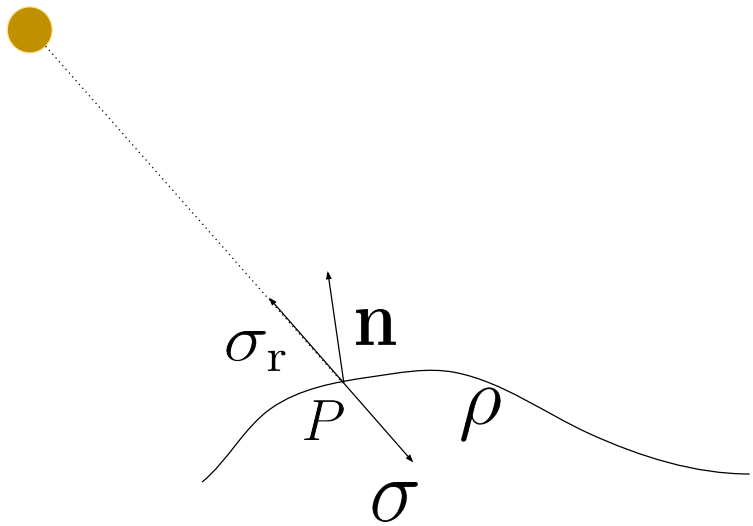

FIGURE 5.8 – Composantes en jeu pour le modèle Lambertien et le modèle de Phong.

D'autres modèles, plus complexes, tel le modèle de Oren-Nayar [141] existent, mais l'intérêt d'utiliser ceux-ci n'est présent que pour des surfaces rugueuses. Pour les réflexions spéculaires, le modèle de Phong [148] est principalement utilisé dans la littérature. Pour la partie spéculaire de l'irradiance de la scène L_s, on écrit alors

$$L_s = \|\sigma_{\mathrm{r}}\| \, \|\mathbf{v}\| \, cos^k \beta \tag{5.10}$$

où σ_{r} est le symétrique du vecteur d'éclairage par rapport à la surface considérée, $\mathbf{v}$ est le vecteur représentant le point de vue de la caméra, β est l'angle entre ces vecteurs, et k est un paramètre à déterminer, contrôlant l'apparente régularité de la surface, et appelé exposant de Phong. D'autre modèles plus complexes existent, et sont même quelquefois utilisées pour modéliser des scènes en cœlioscopie, comme le modèle de Cook-Torrance [194, 123], mais ils ne sont qu'évoqués ici. Dans notre contexte, nous ne modélisons pas directement les spécularités mais nous les détectons grâce à un test de saturation, et retirons ces données des informations à traiter. Seule la composante diffuse est ainsi prise en compte et les surfaces sont considérées comme lambertiennes. La perte

de ces informations n'est cependant pas un problème à la contrainte du problème en question. En effet, si l'on considère des tissus humains comme la surface d'organes tels que le foie, les spécularités ne sont que de taille très réduite, et ne représentent qu'une très faible part des données à disposition. Ne pas utiliser ces informations n'est donc pas très dommageable.

5.6 Modèles de lumière

5.6.1 Modèles ponctuels

La source de lumière du cœlioscope est généralement assimilée à un point P que nous appellerons *centre de lumière*. Si aucune direction préférentielle de propagation n'est considérée, on parlera de modèle Point Light Source (PLS).

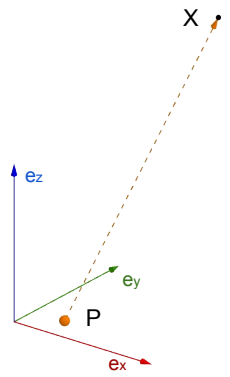

FIGURE 5.9 – Illustration du modèle PLS.

Au point x, on considère que la direction normalisée de la lumière est :

$$\mathbf{L}(x, P) = \frac{x - P}{\|x - P\|}. \tag{5.11}$$

On introduit aussi un facteur d'atténuation lumineuse dû à la loi du carré inverse. Cette atténuation est alors fonction du carré de la distance entre le point x considéré et la source P et s'écrit :

$$\mathbf{S}(x, P) = \frac{1}{\|x - P\|^2}. \tag{5.12}$$

Avec de telles notations, le vecteur d'éclairage au point x pour un modèle PLS s'écrit alors :

$$\sigma_{\text{PLS}}(x, P) = \sigma_0 \, \mathbf{S}(x, P) \, \mathbf{L}(x, P). \tag{5.13}$$

Ce modèle possède uniquement 3 paramètres (les coordonnées du *centre de lumière*). Comme présenté en introduction de ce chapitre, pour la plupart des caméras cœlioscopiques (dont celle que nous utilisons), la lumière est transmise grâce à une fibre optique. Cela donne ainsi à cette source une direction principale de propagation. Le modèle Spot Light Source (SLS) étend ainsi le modèle PLS, en incluant une telle direction.

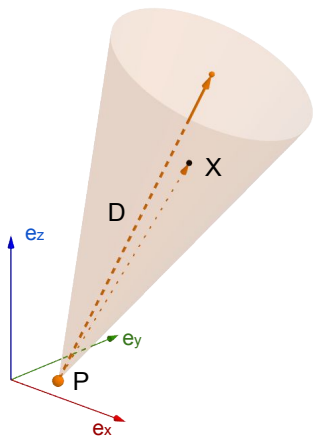

FIGURE 5.10 – Illustration du modèle SLS. Plus on s'éloigne de l'axe de révolution du cône, plus l'intensité est atténuée (en plus de l'atténuation fonction de la distance au centre lumineux).

Trois paramètres scalaires additionnels doivent ainsi être pris en compte. Les deux premiers correspondent à un vecteur D (normalisé), représentant cette direction principale de propagation. Le troisième paramètre μ correspond à la propagation autour de cette direction, c'est-à-dire la force avec laquelle l'intensité diminue lorsque l'on s'éloigne de cette direction. On définit ainsi un terme d'atténuation radiale comme étant :

$$\mathbf{R}(\mu, D, x, P) = \exp\left(-\mu \left(1 - D \cdot \mathbf{L}(x, P)\right)\right). \tag{5.14}$$

La fonction d'éclairage pour ce modèle est alors :

$$\sigma_{\mathrm{SLS}}(\mu, D, x, P) = \sigma_0 \, \mathbf{R}(\mu, D, x, P) \, \mathbf{S}(x, P) \, \mathbf{L}(x, P). \tag{5.15}$$

Plus récemment, des essais étendant ce modèle à un modèle polynomial ont été réalisés afin d'essayer d'obtenir une meilleure modélisation de l'éclairage [33]. Une plus grande flexibilité dans son expression est ainsi donnée à l'atténuation spatiale et angulaire. On appelle ce modèle Polynomial Spot Light Source (PSLS) et on utilisera la formulation suivante :

$$\sigma_{\mathrm{PSLS}}(\mu, D, x, P) = \sum_{i=0}^{p} \sum_{j=0}^{q} b_{i,j} \left(\mathbf{R}(\mu, D, x, P)\right)^i \left(\mathbf{S}(x, P)\right)^j \; \mathbf{L}(x, P). \tag{5.16}$$

Dans nos travaux, on utilise p=q=4, comme suggéré dans [33].

5.6.2 Modèle étendu

Pour la plupart des cœlioscopes utilisés aujourd'hui, la source lumineuse est difficilement assimilable à un point mais est plutôt composée d'une région plane étendue, au niveau de l'extrémité du cœlioscope. Ainsi, la modélisation de cette source en tant que source ponctuelle semble discutable. Une extension directe consiste à considérer la source comme une intégration de sources ponctuelles distribuées au niveau de l'extrémité du cœlioscope. L'expression d'une telle source est alors

$$\sigma_{extended}(X) = \iint_{P \in \psi} \sigma_p(X, P) dP, \tag{5.17}$$

où ψ correspond au motif plan lumineux correspondant à la source lumineuse. Cependant, une utilisation directe de cette expression est difficile. En conséquence, on propose d'utiliser une discrétisation ψ' de ce motif et de considérer, pour chaque point P de cette discrétisation, un modèle SLS avec une même intensité, une même déviation et une même direction de propagation. On appelle un tel modèle un Area Light Source (ALS).

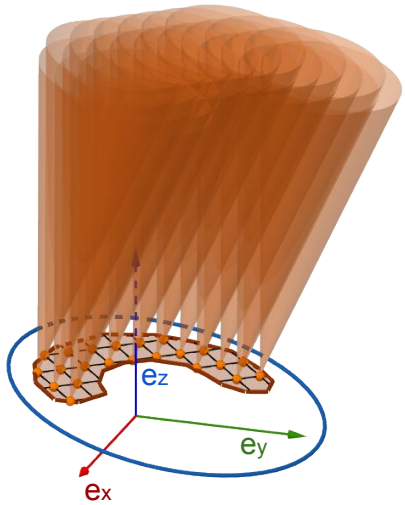

FIGURE 5.11 – Illustration du modèle ALS. Le motif lumineux est discrétisé et chaque point de cette discrétisation est ensuite considéré comme un modèle SLS.

L'expression de la source lumineuse en utilisant un tel modèle est alors :

$$\sigma_{\text{ALS}}(\mu, D, x) = \sigma_0 \sum_{P \in \psi'} \mathbf{R}(\mu, D, x, P)\, \mathbf{S}(x, P)\, \mathbf{L}(x, P). \tag{5.18}$$

5.7 Étalonnage de la source lumineuse

5.7.1 Méthode générale

Les modèles de lumière discutés précédemment doivent être calibrés. Bien que certains travaux considèrent ce problème de calibrage de lumière comme faisant partie du problème général [109] qu'ils cherchent à résoudre (reconstruction de surface, recalage, etc...), et résolvent donc ces deux problèmes simultanément (on parle alors d'*auto-calibrage*), ce problème peut être traité indépendamment. Procéder ainsi permet alors de mieux contraindre notre problème en fixant une partie des inconnues grâce à cette étape préalable. De plus, pour cette étape, les images utilisées pour le calibrage n'ont pas à être prises à l'intérieur du patient et sont prises avant la chirurgie. Celles-ci ont donc tendance à être de meilleure qualité, car étant réalisées dans un environnement pouvant être plus contrôlé, permettant ainsi une meilleure estimation des paramètres

du modèle de lumière considéré. Cependant, l'entièreté des paramètres ne peut être estimée une fois pour toute. En effet, la fonction de réponse de la caméra est propre à chaque image à cause de l'auto-exposition de la caméra. Elle doit donc être réestimée pour chaque vue. L'un des paramètres principaux à estimer est le centre principal de lumière. Tandis que certains considèrent que centre optique et centre principal de lumière coïncident pour les cœlioscopes [14, 98], certains travaux ont remarqué que cela pouvait entraîner des erreurs importantes de modélisation et ont tenté de calibrer ce centre différemment [33]. Certains de ces travaux proposent ainsi d'utiliser une mire de calibrage réfléchissante. L'image du centre lumineux *virtuel* apparaît alors dans les images. La position 3D de ce dernier peut alors être estimée en utilisant plusieurs images. En prenant le symétrique du point trouvé, il est alors possible de retrouver la position réelle du centre principal de lumière. Néanmoins, l'utilisation de la position physique réelle de ce centre est discutable. En effet, une source lumineuse plane et étendue, se comportera de façon similaire à une source ponctuelle lumineuse, située derrière cette surface, et émettant à travers une ouverture de même forme. Dans notre travail, nous calibrons ainsi ce centre, conjointement avec les autres paramètres. On considère ainsi n images I_k de mire de calibrage classique (non réfléchissantes) dans différentes positions. Ces images sont capturées à l'aide de l'une des deux boîtes réalisées pour l'occasion et dont l'utilité est de diminuer la lumière parasite présente dans la scène. Celle-ci est d'intérieur entièrement noir et ne contient que la mire de calibrage. Un orifice au sommet de cette dernière sert ainsi à passer l'extrémité de l'endoscope et permet de capturer les images. La deuxième boîte sert à capturer des images permettant d'extraire le motif lumineux de l'extrémité de l'endoscope comme expliqué dans la partie suivante. Cette boîte comporte un miroir et des ampoules afin de pouvoir correctement observer l'extrémité de l'endoscope.

FIGURE 5.12 – Boîtes permettant la capture d'images en vu de l'étalonnage des modèle lumineux.

Utilisant OpenCV [21] et son implémentation du calibrage géométrique de caméra, nous pouvons automatiquement déterminer la normale à la surface du damier, ainsi que la position 3D de chaque point x de ce dernier. Utilisant ces informations, nous cherchons les paramètres du modèle de lumière expliquant au mieux (au sens des moindre carrés) l'intensité de chaque pixel considéré $I_k(x)$ à l'intérieur des carrés blancs observés (correspondant à un ensemble de points 3D W_k). L'albédo ρ_w dans ces zones est alors inconnu mais supposé constant. On formule ainsi le problème en tant que minimisation d'une énergie, où σ est le modèle de lumière à calibrer, V les paramètres de ce dernier, n le nombre d'images considérées pour le calibrage et $\mathcal{C}_k$ les réponses de caméra :

$$E_{Calib}(V, \mathcal{C}) = \sum_{k=1}^{n} \sum_{x \in W_k} \left(I_k(x) - \frac{\rho_w \ \sigma(V, x) \cdot \mathbf{n}(x)}{\mathcal{C}_k} \right)^2 \tag{5.19}$$

Lorsque l'on considère ce problème avec ρ_w ainsi que tous les $\mathcal{C}_k$ comme étant inconnus, cela mène à une ambiguïté scalaire entre tous ces termes. Afin de lever celle-ci, une image de référence est choisie et sa réponse de caméra est fixée à 1. Nous initialisons ensuite V comme suit : pour des modèles ponctuels, le centre de lumière est initialisé au centre optique, D à la perpendiculaire au plan image, et μ à 10. Pour le modèle étendu, nous utilisons les paramètres obtenus grâce à la méthode décrite dans la section suivante. Ensuite, en inversant l'équation d'ombrage pour l'image de référence, on peut alors calculer les valeurs de l'albédo théorique pour chaque point considéré. La médiane de l'ensemble de ces valeurs sert alors à initialiser ρ_w. Faire de même en considérant cette fois-ci toutes les images et ρ_w fixe permet d'initialiser les $C(k)$. Ces paramètres sont ensuite raffinés en minimisant E_{calib} grâce à l'algorithme de Levenberg-Marquardt, et

son implémentation avec ceres [2].

5.7.2 Extraction du motif

Principe de la méthode

Afin d'utiliser le modèle ALS, le motif de lumière doit être extrait. L'approche discutée dans cette section est appropriée pour tous les cœlioscopes ayant une extrémité plane sur laquelle se trouve une source de lumière étendue, ce qui est le cas de tous les cœlioscopes Storz modernes. La méthode est illustrée dans la Figure 5.13. Utilisant un miroir plan de première surface, on prend n images du reflet de l'extrémité du cœlioscope. L'objectif est alors d'extraire la forme du motif et sa position à partir de ces images. Les contours de deux ellipses sont alors manuellement annotés dans celles-ci : l'ellipse principale, délimitant l'extrémité de la tête du cœlioscope et l'ellipse secondaire, délimitant les contours de l'objectif de la caméra. Ces deux ellipses correspondent à des disques présents sur des plans parallèles (où appartenant au même plan, suivant les configuration des endoscopes). Ces ellipses sont ainsi utilisées pour calculer la pose de l'extrémité du cœlioscope. Pour cela, on considère les poses des disques virtuels qui correspondent aux symétriques de l'extrémité du cœlioscope à travers le miroir pour les différentes images. Les poses de ces ellipses sont calculées en calculant les homographies transformant ces ellipses en disques unitaires [126]. Ces homographies sont ensuite décomposées pour obtenir les poses correspondantes en utilisant [122]. Ceci est illustré dans la Figure 5.13.

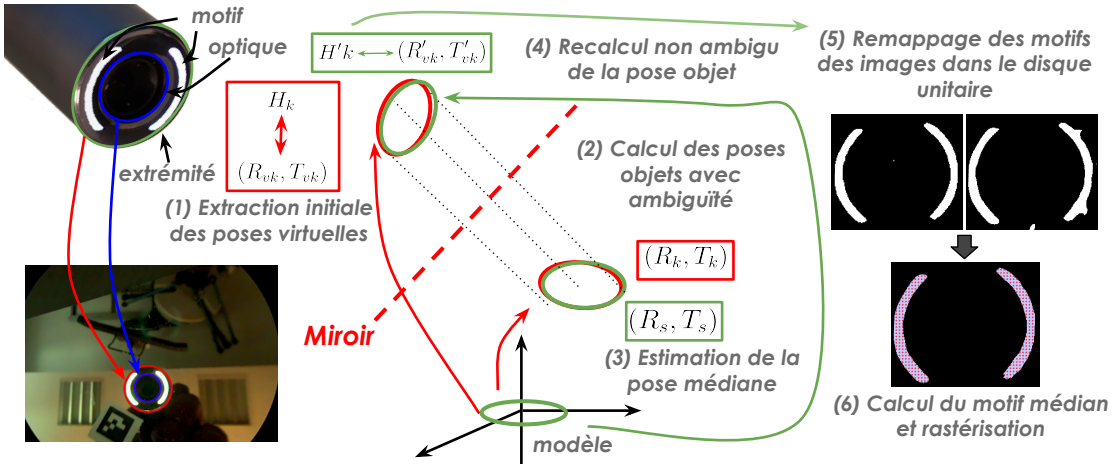

FIGURE 5.13 – Extraction du motif lumineux.

Calcul de pose

Connaissant le diamètre de la tête du cœlioscope, il est possible d'extraire, à partir des ellipses dans les images, la position pour le disque support virtuel, correspondant au symétrique du disque support par le plan du miroir considéré. Ces deux positions possèdent cependant deux ambiguïtés : une ambiguïté de perspective (Figure 5.14), ainsi qu'une ambiguïté de rotation autour de l'axe de révolution du disque virtuel considéré. Chacune de ces ambiguïtés peut cependant être résolue en tirant profit des particularités du montage utilisé.

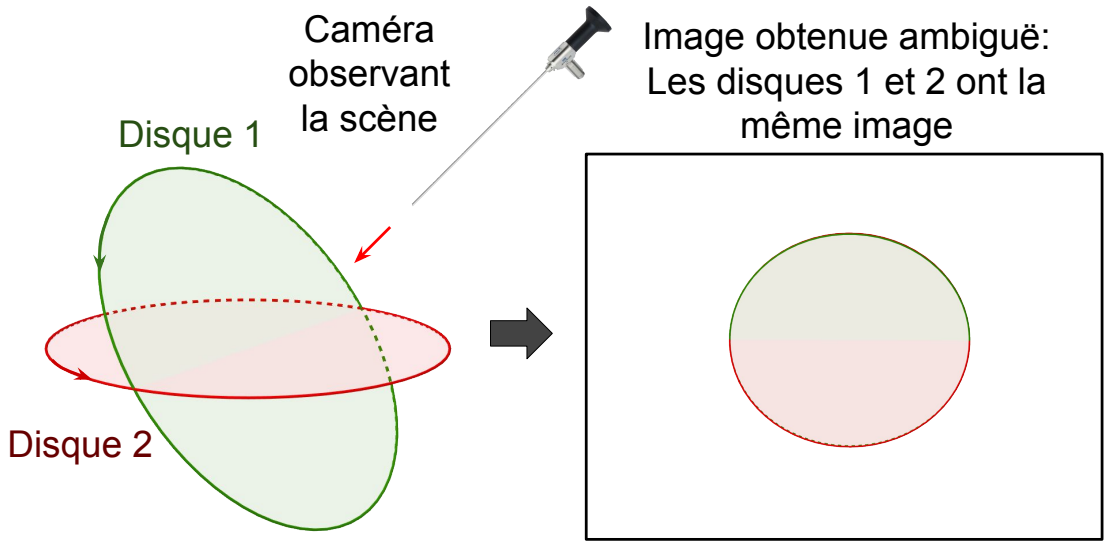

FIGURE 5.14 – Illustration des ambiguïtés de perspective et de rotation.

Notre méthode est divisée en 6 étapes. 1) Dans chaque vue, on utilise l'ellipse délimitant l'extrémité du cœlioscope pour extraire deux homographies [126] correspondant aux deux poses virtuelles (symétriques par rapport au miroir) possibles pour le cœlioscope (dues à l'ambiguïté de perspective). Utilisant l'ellipse correspondant à l'optique, on résout alors cette ambiguïté. En effet, appliquer l'homographie réellement solution du problème doit transformer cette ellipse en un cercle, tandis que l'autre homographie générera une ellipse de plus faible excentricité. On étudie ainsi la variation de l'excentricité de l'ellipse en question après application des homographies pour finalement retenir l'homographie H_k qui convient. On récupère ensuite la pose du disque associée à cette homographie (R_{vk}, T_{vk}) [122] (comportant toujours une ambiguïté de rotation). 2) On calcule la position de l'axe du disque objet, correspondant au symétrique de tous les axes des disques virtuels à travers le plan du miroir. Deux possibilités sont envisageables pour cette étape, l'une utilisant des marqueurs fixés à la surface du miroir et l'autre non. Ces deux possibilités sont discutées dans la sous-section suivante. 3) On fixe ensuite une orientation (R_s, T_s) pour le disque objet (on fixe la rotation). 4) On recalcule ensuite les poses des disques virtuels par symétrie à partir de (R_s, T_s). Comme il s'agit ici d'une symétrie, il n'y a plus d'ambiguïté de rotation (R'_{vk}, T'_{vk}). L'homographie correspondante H'_k (sans rotation) est alors recalculée. 5) Finalement,

les régions appartenant au motif lumineux dans les images sont extraites à l'aide d'un test de saturation, et sont remappées sur le disque objet à l'aide de H'_k. On récupère ensuite à la région médiane présente sur le disque objet (les pixels présents pour une majorité de vues). 6) On discrétise cette région afin d'obtenir l'ensemble de points 3D correspondant aux m sources ponctuelles du modèle ALS.

Récupération de la pose du disque support en utilisant des marqueurs

La première méthode consiste à positionner des marqueurs Aruco [63] à la surface du miroir. Ces derniers permettent alors de localiser la position du plan de symétrie correspondant au miroir. Utilisant ce plan de symétrie, il est alors possible, pour chaque image, d'obtenir la position théorique de l'axe du disque objet support du motif ainsi que son centre. Cette position n'est cependant pas exactement la même pour chaque image à cause du bruit dans l'image, de l'erreur d'extraction des ellipses et de l'erreur d'estimation finale. On considère ainsi les médianes de ces valeurs. Cela nous permet alors, après avoir fixé la rotation, de définir la pose (R_s, T_s).

Extraction du disque support sans utilisation de marqueurs

La seconde méthode décrite dans cette sous-section ne nécessite pas de marqueurs, mais uniquement la présence du miroir. Cependant, cette technique est très sensible aux bruits et aux erreurs lors de l'extraction des ellipses initiales. La méthode précédente, utilisant les marqueurs, est ainsi à préférer lorsque cela est possible. Pour chaque disque virtuel, on appelle C_i son centre et E_i sa normale. Pour chaque configuration, on note alors :

$$A_i = C_i + E_i. \tag{5.20}$$

Désigner P comme symétrique des C_i décrit alors entièrement la symétrie $\mathcal{S}_i$:

$$\mathcal{S}_i : X \to X + 2 \left\langle \frac{P - P_k}{\|P - P_i\|} \middle| X - P_i \right\rangle \frac{P - P_i}{\|P - P_i\|}. \tag{5.21}$$

On définit alors $A'_i = \mathcal{S}_i(A_i)$. Si P est le centre du disque objet recherché, alors tous les A'_i coïncident en un unique point A'. Notre problème a trois inconnues, et chaque paire

de configuration (i, j), semble nous donner trois équations indépendantes :

$$\mathcal{S}_i(A_i) = A'_i = A'_j = \mathcal{S}_j(A_j). \tag{5.22}$$

Néanmoins, comme les E_i sont normalisés, les symétriques de ces vecteurs E'_i sont aussi normalisés. Cela implique que :

$$\|A'_i - E'_i\| = \|A'_j - E'_j\|. \tag{5.23}$$

Ainsi, (5.22) ne nous donne que 2 équations indépendantes et au moins une troisième configuration k est nécessaire afin de contraindre entièrement le problème.

Résolution par optimisation non linéaire

Pour résoudre le problème, on peut alors considérer n différentes configurations avec $n > 3$, et définir l'énergie à minimiser en P suivante

$$E_{\text{center}}(P) = \sum_{(i,j)\in\{1,..,n\}^2} \|\mathcal{S}_i(A_i) - \mathcal{S}_j(A_j)\|^2. \tag{5.24}$$

Cette minimisation est réalisée en utilisant l'algorithme de Levenberg-Macquardt et son implémentation en ceres [2]. Cependant, ce problème est hautement non convexe et doit être initialisé correctement afin d'obtenir un résultat cohérent.

Initialisation : une solution algébrique

Considérer directement les équations 5.22 pour trois configurations i, j et k mène à des équations trop complexes pour être résolues directement. Afin de simplifier le problème, on considère un problème intermédiaire illustré dans la Figure 5.15.

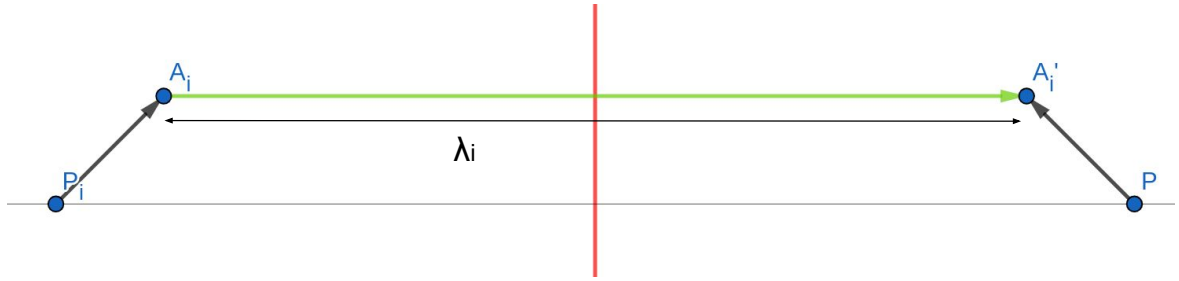

FIGURE 5.15 – Représentation du problème de géométrie associé au calcul de pose du disque objet.

Pour une configuration i, on note λ_i tel que :

$$A'_i = A' = A_i + \lambda_i \overrightarrow{P_i P}. \tag{5.25}$$

Pour un couple de configurations (i, j), on a alors :

$$A' = A_i + \lambda_i \overrightarrow{P_i P} = A_j + \lambda_j \overrightarrow{P_j P}. \tag{5.26}$$

On peut alors exprimer P :

$$P = \frac{A_i - A_j + \lambda_j I_j - \lambda_i I_i}{\lambda_j - \lambda_i}. \tag{5.27}$$

Si on considère une troisième configuration k, on peut considérer le couple (i, k) et écrire :

$$P = \frac{A_i - A_j + \lambda_j I_j - \lambda_i I_i}{\lambda_j - \lambda_i} = \frac{A_i - A_k + \lambda_k I_k - \lambda_i I_i}{\lambda_k - \lambda_i}. \tag{5.28}$$

Cela mène alors à trois équations en λ_i, λ_j et λ_k :

$$\begin{gathered} \lambda_i(A_j - A_k) + \lambda_j(A_k - A_i) + \lambda_k(A_j - A_i) + \\ \lambda_i\lambda_j(I_i - I_j) + \lambda_i\lambda_k(I_k - I_i) + \lambda_2\lambda_3(I_j - I_k) = 0. \end{gathered} \tag{5.29}$$

Cet ensemble d'équations correspond à l'intersection de trois quadriques et peut être résolu en utilisant [100]. On résout ainsi en $(\lambda_i, \lambda_j, \lambda_k)$ ce système d'équation, qui nous donne alors un point P pour chaque triplet de configurations à disposition. On récupère ensuite le point médian à ce nuage de points que nous utilisons comme centre du disque

support. Ce point est utilisé afin de définir entièrement les symétries correspondantes. Cela permet alors de calculer les axes du disque objet théoriques pour chaque configuration et de récupérer leur médiane, que l'on utilise alors pour définir entièrement l'orientation du disque support.

5.8 Résultats expérimentaux

Afin de juger de l'efficacité des modèles lumineux définis dans ce chapitre pour modéliser l'éclairage de la scène cœlioscopique, ces derniers sont comparés entre eux à l'aide de mires de calibrage. Utiliser de telles images permet ainsi de connaître directement, pour des points situés à l'intérieur de ces mires, la position 3D associée à ces derniers. Cela permet ainsi, une fois le modèle calibré, de connaître entièrement la valeur du vecteur d'éclairage en chacun des points considérés. De plus, l'albédo à l'intérieur des cases blanches est considéré comme uniforme. Seule la réponse de la caméra associée à ces vues est alors inconnue. Pour chaque image de mire, il est alors possible de simuler, à partir du modèle de lumière, et ce pour chaque point à l'intérieur des cases blanches du damier, les intensités des pixels de l'image correspondante. Ces intensités sont cependant calculées à un coefficient multiplicatif près correspondant à la CRF. Pour chaque pixel simulé, il est alors possible de considérer le quotient entre l'intensité de ce pixel simulé et celle observée afin d'obtenir des candidats pour la valeur de la CRF dans chaque image. On retient alors comme valeur la médiane à ces valeurs. Cela permet ainsi de simuler entièrement l'intensité des pixels conformément au modèle de lumière choisi et de comparer ces valeurs avec les valeurs réelles.

Deux ensembles d'images sont considérés pour l'évaluation. Le premier correspond à des images obtenues avec un damier de taille "moyenne" (30×15 mm) observé à une distance d'environ 50 mm. Le second correspond à un damier de "petite" taille observée à une distance d'environ 10-15 mm. Séparer ainsi nos données permet alors d'obtenir des informations sur le domaine de validité du modèle considéré. On sépare ensuite chacun de ces ensembles en deux sous-ensembles : l'un pour le calibrage et l'autre pour l'évaluation.

Pour chaque modèle, on considère deux configurations : l'une où le centre de lumière (ou le centre du motif pour le modèle ALS) est fixe, et l'une où ce dernier est réestimé durant le calibrage. Les modèles associés à un centre fixe sont les modèles Fixed Point

Light Source (FPLS), Fixed Spot Light Source (FSLS), Fixed Polynomial Spot Light Source (FPSLS), et Fixed Area Light Source (FALS). Les autres sont les modèles PLS, SLS, PSLS, et ALS. Pour les modèles ponctuels, ce centre est alors fixé au centre optique. Cela permet ainsi de vérifier la validité de l'hypothèse, généralement faite en cœlioscopie, confondant le *centre de lumière* et le *centre optique* de l'appareil. Pour le modèle ALS, le centre est fixé au centre précalculé comme explicité dans la section précédente. Cela de permet de vérifier si la position physique réelle de la source doit ou non être utilisée pour la modélisation, ou si une autre position doit être préférée. Les résultats sont montrés dans la Figure 5.16.

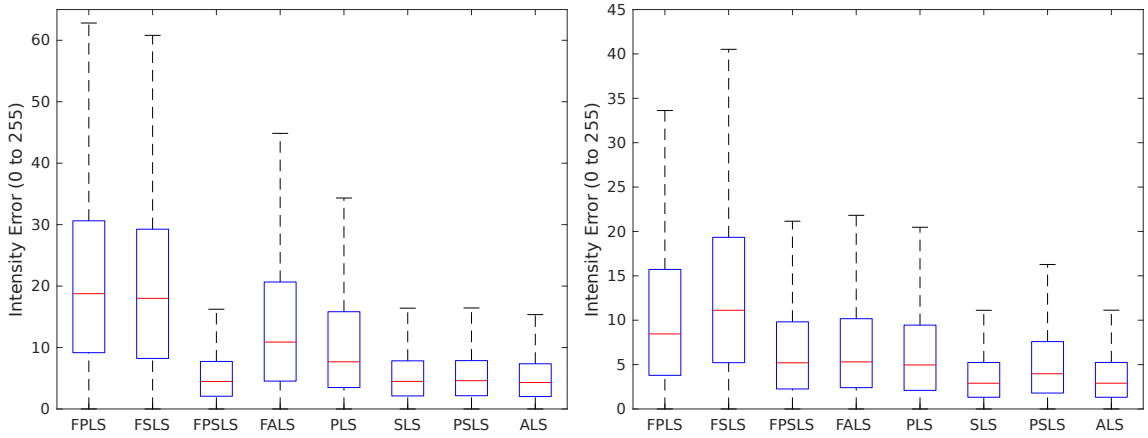

FIGURE 5.16 – Boîte à moustaches (avec médiane, quartiles et valeurs extrêmes) montrant la différence entre intensité observée et prédite pour les pixels considérés : distance de vue faible (gauche) et modérée (droite).

On constate que les modèles ponctuels où le centre est considéré comme coïncidant avec le centre optique donnent, en général, de bien moins bons résultats que leurs homologues où la position du centre est réestimée. Pour ceux-ci une amélioration de l'ordre de 50% de l'erreur d'intensité est observée. La seule exception à ce constat est le modèle SLS lorsqu'utilisé à une distance d'observation faible. Ce dernier ne bénéficie ainsi, dans ce cas bien précis, que d'une amélioration de ses performances très minime lorsque son *centre de lumière* est réestimé. Ceci nous montre ainsi que l'hypothèse consistant à faire coïncider *centre de lumière* et centre optique ne semble, en général pas valide. Le modèle ALS bénéficie lui aussi d'une relativement grande amélioration de ses estimations lorsque le centre du disque motif est réestimé. Le raffinage de tous les paramètres

(y compris les centres) des modèles de sources lumineuses décrits dans ce chapitre est ainsi fortement préférable dans tous les cas. Les centres réestimés seront ainsi en général légèrement derrière le centre optique. Le fait que le centre du disque du modèle ALS doit lui aussi être réestimé alors que les véritables positions 3D des points du motif ont préalablement été extraites peut sembler contre-intuitif mais s'explique cependant facilement. En effet, à cause de l'*ouverture de la source lumineuse* (similaire à l'ouverture optique de la caméra), le centre lumineux du modèle existera derrière le point réel d'émission. De façon similaire, pour le modèle de caméra projective, le centre optique (correspondant au centre de projection) calibré ne coïncide pas, en général avec le centre réel physique de la lentille. Le motif composant ainsi la source lumineuse du modèle ALS doit ainsi dans le modèle se trouver légèrement en retrait par rapport à sa véritable position physique. Dans la Figure 5.16, on constate aussi que le modèle PLS donne de bien moins bons résultats que les autres, il est donc à éviter. Les modèles SLS et ALS donnent quant à eux, tous les deux, de très bons résultats. Le modèle SLS semble ainsi offrir un bon compromis entre complexité et qualité de la modélisation. Le modèle PSLS offre des résultats comparables à ces deux modèles, mais semble quelquefois donner une modélisation éloignée de la réalité pouvant être due à un sur-apprentissage des données de calibrage.

5.9 Conclusion

Nous avons présenté dans ce chapitre le problème de la modélisation de l'éclairage dans la scène cœlioscopique et énoncé quelques utilisations de cette dernière pour l'amélioration de la précision de la cœlioscopie assistée par ordinateur. Différents modèles de lumière de complexités différentes ont ainsi été présentés. Une méthode permettant de calibrer ceux-ci a aussi été proposée. Les modèles ont ensuite étés comparés entre eux en évaluant l'erreur sur les intensités pixels prédites dans des images de mire de calibrage suivant ces différents modèles. Différentes observations ont ainsi été relevées. Tout d'abord, il a été montré que l'hypothèse communément utilisée, faisant coïncider *centre optique* et *centre lumineux* mène a d'importantes erreurs de modélisation de l'éclairage. Elle est ainsi à proscrire lorsqu'une certaine précision est recherchée dans des applications utilisant les ombrages en cœlioscopie. De plus, la position réelle des sources lumineuses n'est pas non plus à utiliser directement dans la modélisation. En

effet, un léger écart entre position physique des sources et position modélisée et calibrée apparaît. Les sources modélisées doivent ainsi être placées légèrement en recul par rapport à leurs positions physiques réelles. Enfin, parmi les différents modèles proposés, le modèle SLS semble, malgré sa relative simplicité, offrir de très bon résultats. Il est ainsi un très bon compromis entre simplicité et justesse de la modélisation, et son utilisation est ainsi fortement recommandée pour des applications utilisant les ombrages dans le cadre de la cœlioscopie assistée par ordinateur. Ces résultats sont cependant à nuancer légèrement et des tests supplémentaires seraient nécessaires afin de répondre à certaines questions toujours en suspens. Ainsi, les tests ayant été réalisés avec un cœlioscope utilisant une source de lumière externe, on peut alors se demander si les conclusions restent les mêmes lorsque l'on utilise un appareil avec une source de lumière intégré. Des études supplémentaires sont aussi nécessaires afin de modéliser les variations des paramètres du modèle induites par le changement d'intensité de la source externe. Ainsi, des travaux préliminaires nous amènent à penser que cette variation n'entraîne pas uniquement un changement de la norme du vecteur d'éclairage, mais aussi une modification des paramètres jouant sur l'atténuation radiale de l'intensité. Des tests additionnels sont néanmoins nécessaires afin de comprendre et estimer cette variation. L'étude de la validité du modèle de lumière calibré dans le temps est aussi un problème nécessitant d'être étudié. En effet, la fibre optique s'abîmant avec le temps, peut-on attendre que le calibrage reste valable dans le temps lorsqu'une même intensité de la source externe est toujours utilisée ? Concernant les sources secondaires induites par les réflexions de la lumière sur les différents organes et sur la paroi abdominale, des tests additionnels sont, là encore, nécessaires afin de juger de leurs contributions à l'éclairage global, et ainsi vérifier si elles peuvent réellement être négligées. Enfin, le cas de l'auto-calibrage des modèles n'est pas discuté dans ce chapitre, mais fait partie des pistes à considérer, permettant de rendre la méthode moins contraignante d'un point de vue de son utilisation au sein du *workflow* chirurgical.

Chapitre 6

Densification de la reconstruction cœlioscopique en utilisant la photométrie

Nous nous intéressons dans ce chapitre à la densification et au raffinage de reconstructions de surface de la scène cœlioscopique per-opératoire ayant été obtenues par des méthodes de type SfM ou SLAM. Pour faire cela, nous proposons ainsi d'utiliser les ombrages de la scène présents dans les images cœlioscopiques. Les modèles de lumière discutés dans le chapitre précédent sont ainsi utilisés afin de modéliser l'éclairage de la scène. Cela permet d'extraire de fortes contraintes entre les images et la surface de l'organe observé. La surface à densifier est paramétrée par une carte de profondeur composée d'une grille de patches de fonctions bicubiques. Cela permet une expression simple et directe des contraintes permettant de définir un problème de minimisation associé à la densification. Une fois résolu, ce problème nous permet ainsi d'obtenir une reconstruction continue de surface de très grande précision. Des foies porcins *ex vivo* sont utilisés pour l'évaluation de notre méthode, permettant de considérer des textures et illuminations proches de conditions réelles.

6.1 Introduction

Les modèles de lumière discutés dans le chapitre précédent permettent d'estimer l'illumination en tout point de la scène cœlioscopique. Cela permet ainsi l'expression de contraintes fortes entre les ombrages présents dans les images cœlioscopiques et la surface de l'organe observé. Une utilisation importante de ces contraintes se situe dans les problèmes de raffinage et densification de reconstructions de surface de la scène per-opératoire. Cette scène densifiée est ainsi de bien meilleure qualité que les surfaces obtenues par SfM ou SLAM [117, 199, 195, 178, 118]. Plusieurs problèmes apparaissent en effet lorsque l'on utilise directement des reconstructions basées sur l'appariement de points clés. Tout d'abord, à cause de la faible texturation de certains organes mais aussi des brusques changements de luminosité dans les images (qui ne sont pas pris en compte dans de telles méthodes), les surfaces reconstruites sont en général de faible qualité en termes de géométrie. On obtiendra ainsi en général des surfaces peu denses et très bruitées. De plus, bien que ces surfaces puissent être densifiées à l'aide d'un algorithme de MVS [170, 64, 75, 206, 201], l'information de texture des surfaces obtenues (densifiée ou non) est en général peu fiable, et ce principalement à cause de changements de luminosité non pris en compte entre les images. L'utilisation de ces informations pour texturer le modèle et réaliser ensuite un recalage temps-réel basé sur la mise en correspondance de textures comme proposé dans certains pipelines de RA [38] est ainsi contestable. L'utilisation d'une telle texture, dépendant de l'illumination de la scène, risque ainsi d'entraîner de faux appariements et de fortes imprécisions dans le recalage final. Afin de répondre à ces problèmes, nous avons développé une méthode pour raffiner ces surfaces obtenues grâce à des méthodes SfM ou SLAM, tout en étant capable d'extraire une information de texture de bien meilleure qualité, et indépendante de l'illumination de la scène. Cette texture prend alors la forme d'une carte d'albédo (correspondant à la proportion de lumière réfléchie par la surface conformément au modèle Lambertien [221], déjà discuté dans le chapitre précédent). En entrée de notre algorithme, on considère ainsi une reconstruction éparse, provenant d'une reconstruction de type SfM ou SLAM, et composée des points de cette reconstruction ainsi que des poses de caméra des images clés associées. L'idée de base de notre méthode est similaire aux algorithmes de MVS [170, 64]. L'objectif de la méthode est ainsi de créer une surface dense par vue considérée, initialisée grâce aux points de la surface éparse d'entrée, et de

reconstruire les données manquantes tout en raffinant celles existantes en utilisant les informations présentes dans les vues "proches" de ces vues de référence (et les poses des caméras associées). Ces informations permettent ainsi de définir de fortes contraintes en chaque point de la reconstruction considérée. Ces surfaces, correspondant à des vues de référence, doivent alors être fusionnées pour reconstruire la scène dans son entièreté [51, 198, 47, 217, 206]. Dans ce chapitre, seuls le raffinage et la densification de vues de référence sont considérés. La fusion de ces surfaces n'est ainsi pas traitée. Néanmoins, elle doit être traitée dans de futurs travaux afin de considérer le pipeline de densification dans son ensemble. Ainsi, dans ce chapitre, un algorithme de SAMVS, utilisant un modèle de lumière précalibré pour le cœlioscope est présenté et évalué. L'évaluation de ce dernier est alors réalisé avec les modèles de lumière discutés dans le chapitre précédent.

6.2 État de l'art

L'objectif des algorithmes de MVS est de reconstruire, de façon dense, la surface d'un objet observé par une caméra, à partir des images provenant de celle-ci et des poses de caméra associées aux vues correspondantes. Dans ces algorithmes, ces poses sont considérées comme connues. Afin d'obtenir ces poses, deux choix sont possibles. Le premier est de considérer un système de caméras calibré (dont les poses relatives entre les caméras sont connues). Si l'on se restreint aux systèmes à deux caméras, on peut utiliser des caméras stéréo. Cependant, pour un nombre de caméras plus important, un tel système est peu commun et se retrouve surtout dans des environnements de type laboratoire. Le second choix est d'utiliser une méthode existante de type SfM ou SLAM s'appuyant sur des correspondances entre points clés dans les images. Cela permet ainsi d'obtenir les poses de caméra à un facteur d'échelle près.

6.2.1 Paramétrisations

Différents types de paramétrisation existent pour les surfaces générées par de tels algorithmes. Ainsi, certaines méthodes considèrent l'espace à reconstruire sous la forme d'une grille 3D de voxels [201, 173]. Cette méthode permet ainsi une représentation relativement simple pour la reconstruction et n'entraîne pas de problème de cohérence

entre les vues vis-à-vis de la reconstruction : la reconstruction associée à chaque vue est la même et correspond à la projection de la grille de voxels dans chaque plan image. Autrement dit, la reconstruction considérée est la même pour chaque vue. Une telle méthode nécessite alors une étape de post-traitement consistant à extraire une surface correspondant à la grille de voxels obtenue [171]. Néanmoins la résolution de la reconstruction finale, et donc sa précision dépend alors directement de la résolution de la grille considérée. Cependant, cette résolution doit être limitée car des grilles volumiques de forte résolution peuvent être très lourdes à manipuler en termes de mémoire. D'autres algorithmes de MVS considèrent directement des maillages [83, 46, 57]. Encore une fois, ces méthodes permettent d'éviter de considérer des problèmes de cohérence dans la reconstruction pour différentes vues. Néanmoins ici, les contraintes à prendre en compte pour la reconstruction peuvent être bien plus complexes au niveau de leurs expressions. D'autres méthodes proposent encore d'utiliser une reconstruction basée sur une fonction de distance à la surface [152, 151, 87]. Ce type de support pour les reconstructions n'engendre encore une fois pas de problème de cohérence entre les vues. De plus, ces méthodes permettent d'obtenir une précision plus grande que les méthodes utilisant une grille de voxels comme support pour une même taille de mémoire donnée. Le stockage de telles surfaces peut, avec une telle méthode, tout de même encore se révéler être assez lourd. Enfin, une dernière solution est de stocker la reconstruction sous la forme d'un ensemble de cartes de profondeur [184, 97, 62, 89]. La formulation des contraintes en utilisant un tel support est ainsi très simple. Cependant, elle nécessite de porter une attention toute particulière à la cohérence des différentes cartes de profondeur entre elles. Cela est géré, en général, à l'aide de coûts supplémentaires permettant de maintenir la cohérence entre les différentes vues. De plus, la précision de la reconstruction est ici directement dépendante de la résolution des cartes de profondeur, choisies pour les différentes vues. En général, une étape de fusion des différentes cartes de profondeur est considérée à la suite de tels algorithmes afin d'obtenir une seule surface [51, 198, 47, 217, 206]. Dans notre étude nous proposons d'utiliser une méthode mixte.

Nous considérons ainsi les vues une à une pour la reconstruction. Chaque vue est reconstruite à partir d'une vue de référence et des vues alentours. La reconstruction n'est ici pas stockée sous la forme d'une carte de profondeur mais sous la forme d'une grille de profondeur. Cette grille est constituée d'un ensemble de patches partageant des

nœuds entre patches voisins. Chacun de ces patches représente une zone de profondeur continue représentée par une fonction bicubique en x et y (les dimensions de la vue). Une telle représentation permet ainsi d'obtenir une surface C^1 tout en n'ayant qu'une faible complexité, en termes de mémoire, à considérer.

6.2.2 Photo-consistance

Pour ces différentes méthodes de MVS, une surface dont la projection dans les différentes images considérées correspond à la photométrie de la scène, doit être estimée. La façon dont cette photo-consistance est définie dépend alors des différentes approches considérées. Certaines techniques comparent directement l'intensité des pixels dans les images en considérant chaque point inconnu de la reconstruction, et en le reprojetant dans les différentes vues [171, 101]. D'autres techniques considèrent plutôt des fenêtres de taille donnée dans les images, et ce sont ces fenêtres qui sont comparées entre elles et utilisées pour calculer la photo-consistance entre les associations. Ainsi, on considère encore une fois pour chaque point, de position inconnue, la projection de celui-ci dans les images. On étudie ensuite pour les positions des points dans les images, des fenêtres autour de ces points et y on calcule la photo-consistance [86, 51, 151]. La fonction associée à cette notion de photo-consistance peut être calculée de différentes façons. Ainsi, une méthode simple consiste à considérer la somme des différences d'intensités pixel à pixel dans les fenêtres. Afin d'être robuste à de légers mauvais alignements dans les poses de caméra, des valeurs telles que la Similarité Structurelle (SSIM) peuvent aussi être considérées [205]. D'autres méthodes telles que Normalized Cross Correlation (NCC) [219] permettent d'être moins sensible aux changements de luminosité et aux bruits dans les images. Enfin, plus récemment, des techniques ne considérant pas explicitement la photo-consistance de la scène mais plutôt basées sur le *deep learning* [114, 26, 223] ont été proposées afin de générer des cartes de profondeur, pouvant ensuite être fusionnées comme décrit dans la section précédente Dans notre méthode, nous ne considérons pas la photo-consistance entre les vues, mais la vraisemblance entre la surface à reconstruire couplée à un modèle de lumière calibré du cœlioscope, et les ombrages présents dans les images.

6.2.3 Éclairages

Pour des méthodes de MVS utilisées dans des contextes généraux [64, 75, 204], l'éclairage de la scène est soit supposé constant entre les différentes images, soit une métrique, supposée invariante aux changements d'illumination (comme NCC), et utilisée pour le calcul de la photo-consistance. Dans le cas de la chirurgie cœlioscopique, l'illumination de la scène n'est cependant pas constant entre les images car la lumière est fixée au cœlioscope dont la position évolue au cours du temps. Les contraintes à appliquer aux surfaces de référence doivent ainsi prendre en compte ce changement. Récemment, une extension des algorithmes MVS, prenant directement en compte l'éclairage, a été proposée [104]. Celle-ci utilise alors en contraintes denses, des contraintes correspondant au respect de la loi de Lambert en chaque point. On parle alors de SAMVS. Cette méthode permet ainsi de reconstruire des surfaces de faible texture, comme les organes. Cependant, dans la méthode proposée, la source de lumière est toujours supposée fixe dans les différentes images considérées, ce qui ne correspond pas à notre cas. Nous avons donc proposé une extension de cet algorithme, bien adaptée au cas de la scène cœlioscopique, et prenant en entrée additionnelle un modèle de lumière calibré (en utilisant les modèles du chapitre précédent). Ce modèle de lumière calibré, contrairement au modèle utilisé dans [104] considère ainsi une atténuation de l'intensité lumineuse en fonction du centre de lumière, créant ainsi des ombrages plus marqués, caractéristiques des images cœlioscopiques. Une source de lumière *à l'infini*, comme considérée dans [104] ne permet en effet pas de reproduire fidèlement l'éclairage en cœlioscopie. Celle-ci pourrait être source d'erreurs pour la reconstruction estimée par cet algorithme. L'utilisation d'un modèle précalibré permet alors de réduire le nombre d'inconnues du problème et d'augmenter sa robustesse. Notre implémentation est relativement générique et tout modèle de lumière peut facilement être implémenté et utilisé.

6.3 Vue de référence, modèle de surface et propriétés émergentes

6.3.1 Description générale

La reconstruction en entrée de notre algorithme se compose d'un nuage de points épars et des images clés associées à ce dernier (images et poses de caméra). Ces entrées proviennent de méthodes de type SfM ou SLAM. Dans nos expérimentations, nous utilisons la méthode de SfM implémentée dans le logiciel propriétaire PhotoScan-Pro [1] v1.1.6.2038 en utilisant le mode haute précision, avec la présélection de paires désactivée, le nombre maximum de points clés à extraire par image fixé à 40000 et un nombre de points d'accroche minimal fixé à 1000. L'étape de densification est cependant désactivée afin d'obtenir un nuage de point épars mais fiable. On considère ainsi une vue de référence associée à cette reconstruction, et la tâche est donc de calculer une surface dense correspondante à celle-ci (sous la forme d'une carte de profondeur correspondant à cette vue), ainsi qu'une valeur d'albédo pour chaque pixel. On modélise donc notre carte d'albédo par une valeur pour chaque pixel de l'image (correspondant à la texture de l'objet et étant invariante aux changements d'éclairage). La carte de profondeur est quant à elle modélisée par une grille de patches bicubiques. Contrairement à d'autres modèles de géométrie tels que les B-Splines [189] ou les surfaces de Bézier [111], ce modèle permet d'obtenir des expressions très simples pour les contraintes sur les ombrages considérés. Une illustration de cette grille est présente dans la Figure 6.1. Une représentation continue est ainsi choisie pour la géométrie de la surface car la profondeur dans la scène cœlioscopique a tendance à varier de façon régulière. Au contraire, une représentation discontinue est choisie pour la carte d'albédo car des changements abrupts de texture peuvent exister à la surface de l'organe.

1. http://www.agisoft.com

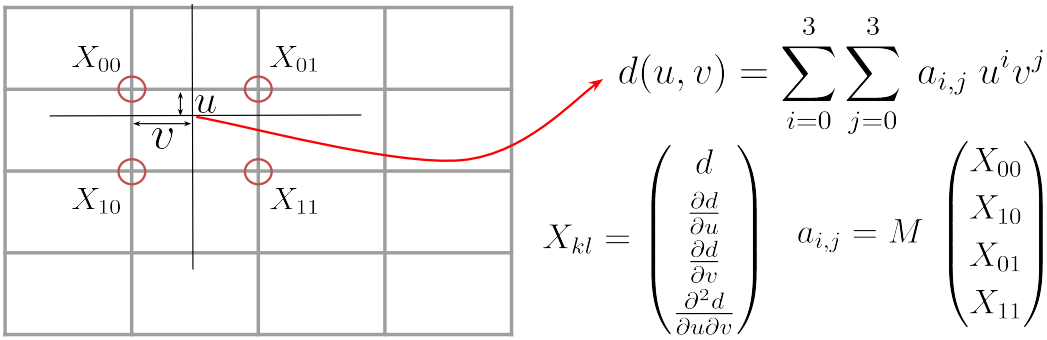

FIGURE 6.1 – Représentation de la surface reconstruite sous la forme d'une grille de profondeur.

Cette grille est alors définie par ses nœuds. Chaque nœud contient quatre valeurs : la valeur de la profondeur en ce point, les dérivées premières de la profondeur suivant les deux dimensions de l'image et la dérivée seconde mixte.

$$X_{k,l} = \begin{bmatrix} d(k,l) \\ \frac{\partial d}{\partial u}(k,l) \\ \frac{\partial d}{\partial v}(k,l) \\ \frac{\partial^2 d}{\partial u \partial v}(k,l) \end{bmatrix}. \tag{6.1}$$

Quatre nœuds définissent alors un patch bicubique, c'est-à-dire que la profondeur pour chaque point de ce patch se définit en tant que polynôme en u et v de degré total 9 dont les coefficients sont fonction des valeurs de ces nœuds. On a alors :

$$d(u,v) = \sum_{i=0}^{3}\sum_{j=0}^{3} a_{i,j} u^i v^j, \tag{6.2}$$

où les valeurs des coefficients $a_{i,j}$ sont exprimés par :

$$\begin{bmatrix} a_{3,3} \\ a_{2,3} \\ a_{1,3} \\ a_{0,3} \\ a_{3,2} \\ \vdots \\ a_{0,0} \end{bmatrix} = M \begin{bmatrix} X_{0,0} \\ X_{1,0} \\ X_{0,1} \\ X_{1,1} \end{bmatrix}, \tag{6.3}$$

pour chaque patch, avec :

$$M = \begin{bmatrix}
4 & 2 & 2 & 1 & -4 & 2 & -2 & 1 & -4 & -2 & 2 & 1 & 4 & -2 & -2 & 1 \\
-6 & -4 & -3 & -2 & 6 & -2 & 3 & -1 & 6 & 4 & -3 & -2 & -6 & 2 & 3 & -1 \\
0 & 2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 1 & 0 & 0 & 0 & 0 \\
2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
-6 & -3 & -4 & -2 & 6 & -3 & 4 & -2 & 6 & 3 & -2 & -1 & -6 & 3 & 2 & -1 \\
9 & 6 & 6 & 4 & -9 & 3 & -6 & 2 & -9 & -6 & 3 & 2 & 9 & -3 & -3 & 1 \\
0 & -3 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & -1 & 0 & 0 & 0 & 0 \\
-3 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 2 & 1 & 0 & 0 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -3 & -2 & 0 & 0 & 3 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 1 & 0 & 0 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-3 & -2 & 0 & 0 & 3 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}. \tag{6.4}$$

M est construite en évaluant l'expression en chaque nœud, de la valeur de la fonction ainsi que de ses dérivées, et en inversant le système. La surface correspondant à cette carte de profondeur est alors C^1, comme expliqué dans la section suivante.

6.3.2 Caractère C^1 de la surface considérée

Afin de montrer que la surface est C^1 (et ainsi pouvoir utiliser les coefficients de la grille $\mathcal{D}$ pour exprimer les normales à notre surface), il suffit de montrer que la carte de profondeur associée est elle aussi C^1. En effet, si on appelle S la fonction représentative de notre surface

$$\begin{array}{rccc} S & : & \mathbb{R}^2 & \longrightarrow & \mathbb{R}^3 \\ & & (u,v) & \longmapsto & S(u,v) \end{array}, \tag{6.5}$$

(λ_u, λ_v) la taille des cellules de la grille considérée et D la fonction associée à la carte de profondeur, on peut écrire :

$$\begin{array}{rccc} S & : & \mathbb{R}^2 & \longrightarrow & \mathbb{R}^3 \\ & & (u,v) & \longmapsto & p\left(\lambda_v v, \lambda_u u, D(u,v)\right) \end{array}, \tag{6.6}$$

avec :

$$\begin{array}{rccc} p & : & \mathbb{R}^3 & \longrightarrow & \mathbb{R}^3 \\ & & \begin{bmatrix} u \\ v \\ z \end{bmatrix} & \longmapsto & \begin{cases} x = & \frac{v-c_y}{f_y}\, z \\ y = & \frac{u-c_x}{f_x}\, z \\ z = & z \end{cases} \end{array}. \tag{6.7}$$

Alors, si $\mathcal{D}$ est C^1, on a $S \in C^1$ en tant que composée d'applications C^1. $\mathcal{D}$ étant définie par ses patchs, pour montrer que $\mathcal{D}$ est C^1 partout, il faut donc montrer qu'elle l'est à l'intérieur des patches et au niveau des limites entre eux. Pour commencer, les patches étant polynomiaux, ils sont C^∞ et donc C^1 en particulier. Le tout est donc d'étudier la continuité de la fonction et de ses dérivées au niveau des interfaces entre patchs.

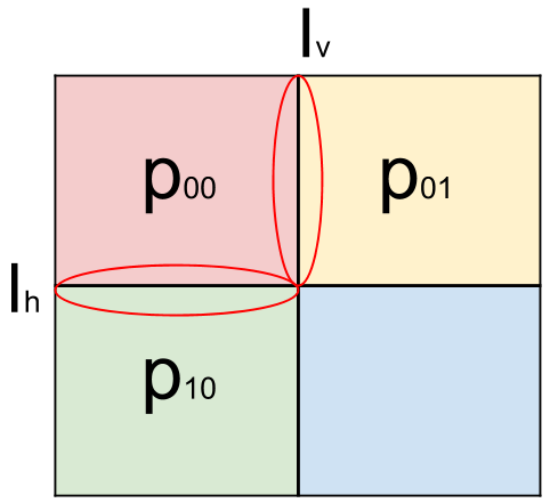

FIGURE 6.2 – Frontières entre les différents patches.

Les frontières I_v et I_h doivent donc être étudiées. Cependant ces deux frontières ayant des expressions symétriques, on ne considérera dans cette démonstration que le cas de I_v, l'autre cas s'en déduisant.

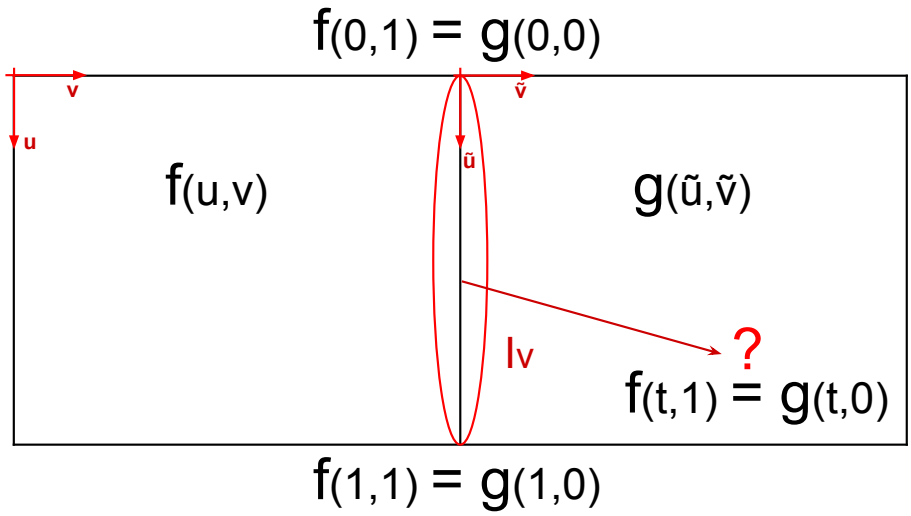

FIGURE 6.3 – Problème du raccordement.

Soient f et g les fonctions correspondant à deux patches voisins (partageant ainsi deux

nœuds et une frontière). f et g sont définies par les coefficients $a_{i,j}$ et $b_{i,j}$ (pouvant s'exprimer en fonction des valeurs des nœuds correspondants) tels que :

$$\begin{aligned} f(x,y) &= \sum_{i=0}^{3}\sum_{j=0}^{3} a_{i,j} x^i y^j \\ g(x,y) &= \sum_{i=0}^{3}\sum_{j=0}^{3} b_{i,j} x^i y^j. \end{aligned} \tag{6.8}$$

Étudions tout d'abord la continuité. Au niveau de la frontière I_v, on a :

$$\begin{aligned} f(x,1) &= \sum_{i=0}^{3}\left(\sum_{j=0}^{3} a_{i,j}\right) x^i \\ g(x,0) &= \sum_{i=0}^{3} b_{i,0}\, x^i. \end{aligned} \tag{6.9}$$

On obtient en remplaçant les coefficients par leurs valeurs obtenues grâce à l'équation (6.9) :

$$\begin{aligned} f(x,1) =& f(0,1) + \frac{\partial f}{\partial x}(1,1)\, x + \left(-3f(0,1) - 2\frac{\partial f}{\partial x}(0,1) + 3f(1,1) - \frac{\partial f}{\partial x}(1,1)\right) x^2 + \\ & \left(-2f(0,1) + \frac{\partial f}{\partial x}(0,1) - 2f(1,1) + \frac{\partial f}{\partial x}(1,1)\right) x^3 \\ g(x,0) =& g(0,0) + \frac{\partial g}{\partial x}(1,0)\, x + \left(-3g(0,0) - 2\frac{\partial g}{\partial x}(0,0) + 3g(1,0) - \frac{\partial g}{\partial x}(1,0)\right) x^2 + \\ & \left(-2g(0,0) + \frac{\partial g}{\partial x}(0,0) - 2g(1,0) + \frac{\partial g}{\partial x}(1,0)\right) x^3. \end{aligned} \tag{6.10}$$

Comme les nœuds de f et g sont partagés, on a alors :

$$\begin{aligned} f(0,1) =& g(0,0) \\ \frac{\partial f}{\partial x}(1,1) =& \frac{\partial g}{\partial x}(1,0) \\ \frac{\partial f}{\partial x}(0,1) =& \frac{\partial g}{\partial x}(0,0) \\ f(1,1) =& g(1,0). \end{aligned} \tag{6.11}$$

D'où finalement $f(x,1) = g(x,0)$. D est donc C^0. Il faut maintenant montrer la continuité des dérivées premières. On a :

$$\begin{aligned}
\frac{\partial f}{\partial x}(x,y) &= \sum_{i=0}^{2}\sum_{j=0}^{3}(i+1)a_{i+1,j}\ x^i y^j \\
\frac{\partial f}{\partial y}(x,y) &= \sum_{i=0}^{3}\sum_{j=0}^{2}(j+1)a_{i,j+1}\ x^i y^j \\
\frac{\partial g}{\partial x}(x,y) &= \sum_{i=0}^{2}\sum_{j=0}^{3}(i+1)b_{i+1,j}\ x^i y^j \\
\frac{\partial g}{\partial y}(x,y) &= \sum_{i=0}^{3}\sum_{j=0}^{2}(j+1)b_{i,j+1}\ x^i y^j.
\end{aligned} \tag{6.12}$$

Ainsi, à la frontière I_v, les dérivées sont égales à :

$$\begin{aligned}
\frac{\partial f}{\partial x}(x,1) &= \sum_{i=0}^{2}\left(\sum_{j=0}^{3}(i+1)a_{i+1,j}\right)\ x^i \\
\frac{\partial f}{\partial y}(x,1) &= \sum_{i=0}^{3}\left(\sum_{j=0}^{2}(j+1)a_{i,j+1}\right)\ x^i \\
\frac{\partial g}{\partial x}(x,0) &= \sum_{i=0}^{2}(i+1)b_{i+1,0}\ x^i \\
\frac{\partial g}{\partial y}(x,0) &= \sum_{i=0}^{3}\ b_{i,1}\ x^i.
\end{aligned} \tag{6.13}$$

En remplaçant les coefficients, on obtient :

$$\begin{aligned}
\frac{\partial f}{\partial x}(x,1) =& \frac{\partial f}{\partial x}(0,1)+ \\
& \left(-6f(0,1) - 4\frac{\partial f}{\partial x}(0,1) + 6f(1,1) - 2\frac{\partial f}{\partial x}(1,1)\right)x+ \\
& \left(4f(0,1) + 2\frac{\partial f}{\partial x}(0,1) - 4f(1,1) + 2\frac{\partial f}{\partial x}(1,1)\right)x^2 \\
\frac{\partial f}{\partial y}(x,1) =& \frac{\partial f}{\partial y}(0,1) + \frac{\partial^2 f}{\partial x \partial y}(0,1)x+ \\
& \left(-3\frac{\partial f}{\partial y}(0,1) - 2\frac{\partial^2 f}{\partial x \partial y}(0,1) + 3\frac{\partial f}{\partial y}(1,1) - \frac{\partial^2 f}{\partial x \partial y}(1,1)\right)x^2+ \\
& \left(2\frac{\partial f}{\partial y}(0,1) + \frac{\partial^2 f}{\partial x \partial y}(0,1) - 2\frac{\partial f}{\partial y}(1,1) + \frac{\partial^2 f}{\partial x \partial y}(1,1)\right)x^3
\end{aligned} \tag{6.14}$$

$$\begin{aligned}
\frac{\partial g}{\partial x}(x,0) =& \frac{\partial g}{\partial x}(0,0)+ \\
& \left(-6g(0,0) - 4\frac{\partial g}{\partial x}(0,0) + 6g(1,0) - 2\frac{\partial g}{\partial x}(1,0)\right)x+ \\
& \left(4g(0,0) + 2\frac{\partial g}{\partial x}(0,0) - 4g(1,0) + 2\frac{\partial g}{\partial x}(1,0)\right)x^2 \\
\frac{\partial g}{\partial y}(x,0) =& \frac{\partial g}{\partial y}(0,0) + \frac{\partial^2 g}{\partial x \partial y}(0,0)x+ \\
& \left(-3\frac{\partial g}{\partial y}(0,0) - 2\frac{\partial^2 g}{\partial x \partial y}(0,0) + 3\frac{\partial g}{\partial y}(1,0) - \frac{\partial^2 g}{\partial x \partial y}(1,0)\right)x^2+ \\
& \left(2\frac{\partial g}{\partial y}(0,0) + \frac{\partial^2 g}{\partial x \partial y}(0,0) - 2\frac{\partial g}{\partial y}(1,0) + \frac{\partial^2 g}{\partial x \partial y}(1,0)\right)x^3.
\end{aligned} \tag{6.15}$$

Les nœuds de f et g étant partagés, on montre l'égalité de ces expressions. $\mathcal{D}$, et donc S sont donc C^1. Il est à noter que S n'est cependant pas C^2. En effet, avec les mêmes

notations, on a pour le nœud correspondant à $(0,0)$ pour g et $(0,1)$ pour f :

$$\begin{aligned}\frac{\partial^2 f}{\partial y^2}(0,1) =& 6f(0,0) + 2\frac{\partial f}{\partial x}(0,0) + \\ & - 6f(1,0) + 2\frac{\partial f}{\partial x}(1,0) \\ \frac{\partial^2 g}{\partial y^2}(0,0) =& 6g(0,0) + 2\frac{\partial g}{\partial x}(0,0) + \\ & - 6g(1,0) + 2\frac{\partial g}{\partial x}(1,0).\end{aligned} \tag{6.16}$$

Or si on réécrit g en fonction de f on a :

$$\begin{aligned}\frac{\partial^2 f}{\partial y^2}(0,1) =& 6f(0,0) + 2\frac{\partial f}{\partial x}(0,0) + \\ & - 6f(1,0) + 2\frac{\partial f}{\partial x}(1,0) \\ \frac{\partial^2 g}{\partial y^2}(0,0) =& 6f(0,1) + 2\frac{\partial f}{\partial x}(0,1) + \\ & - 6f(1,1) + 2\frac{\partial f}{\partial x}(1,1).\end{aligned} \tag{6.17}$$

On a donc, dans le cas général, $\frac{\partial^2 f}{\partial y^2}(0,1) \neq \frac{\partial^2 g}{\partial y^2}(0,0)$. Ainsi, $\mathcal{D}$, et donc S ne sont pas C^2 au point correspondant.

6.3.3 Expression de la normale

La normale peut s'exprimer simplement en fonction des nœuds environnants. Pour cela, on exprime la Jacobienne de la fonction S définissant cette surface :

$$\begin{aligned}J_S &= \begin{bmatrix} \frac{\partial S_x}{\partial u} & \frac{\partial S_x}{\partial v} \\ \frac{\partial S_y}{\partial u} & \frac{\partial S_y}{\partial v} \\ \frac{\partial S_z}{\partial u} & \frac{\partial S_z}{\partial v} \end{bmatrix} \\ &= \begin{bmatrix} \frac{\lambda_v v - c_x}{f_x}\frac{\partial D}{\partial u} & \frac{\lambda_v D(u,v)}{f_x} + \frac{\lambda_v v - cx}{fx}\frac{\partial D}{\partial v} \\ \frac{\lambda_v D(u,v)}{fy} + \frac{\lambda_u u - c_y}{f_y}\frac{\partial D}{\partial u} & \frac{\lambda_u - c_y}{f_y}\frac{\partial D}{\partial v} \\ \frac{\partial D}{\partial u} & \frac{\partial D}{\partial v} \end{bmatrix}.\end{aligned} \tag{6.18}$$

La normale à la surface s'écrit alors :

$$\overrightarrow{N} = \begin{bmatrix} \frac{\lambda_v v - c_x}{f_x} \frac{\partial D}{\partial u} \\ \frac{\lambda_v D(u,v)}{fy} + \frac{\lambda_u u - c_y}{f_y} \frac{\partial D}{\partial u} \\ \frac{\partial D}{\partial u} \end{bmatrix} \wedge \begin{bmatrix} \frac{\lambda_v D(u,v)}{f_x} + \frac{\lambda_v v - cx}{fx} \frac{\partial D}{\partial v} \\ \frac{\lambda_u - c_y}{f_y} \frac{\partial D}{\partial v} \\ \frac{\partial D}{\partial v}. \end{bmatrix}. \tag{6.19}$$

Cette expression permet ainsi d'écrire facilement des contraintes liées aux ombrages de la scène, directement en fonction des nœuds de la grille.

6.4 Formulation par un problème d'optimisation

6.4.1 Énergies à considérer

Le problème de densification est vu comme un problème de minimisation d'énergie dont les inconnues sont les nœuds de la grille de profondeur $\mathcal{D}$, les valeurs des albédos ρ, mais aussi les différentes réponses de caméra des images $\mathcal{C}_k \in \mathcal{C}$. On décompose alors cette énergie en différents termes et on écrit

$$E_{\text{rec}}(\rho, \mathcal{D}, \mathcal{C}) = E_{\text{shad}}(\rho, \mathcal{D}, \mathcal{C}) + \lambda_{\text{pos}} E_{\text{pos}}(\mathcal{D}) + \lambda_{\text{reg}} E_{\text{reg}}(\mathcal{D}). \tag{6.20}$$

Les poids sont fixés empiriquement à $\lambda_{\text{pos}} = 50$ et $\lambda_{\text{reg}} = 40$, tandis que les différents termes sont discutés ci-dessous. Le terme $E_{\text{shad}}(\rho, \mathcal{D}, \mathcal{C})$ représente les contraintes sur les ombrages. Ce terme s'écrit alors comme la différence entre les intensités des pixels dans les images I_k et les intensités prédites par la reconstruction. Afin de ne pas prendre en compte les données provenant des tâches spéculaires, ni celles provenant de la bordure de l'image (étant occultées par le corps du cœlioscope), on utilise un masque Λ_k. De plus, ce terme utilise un poids inspiré de [104] : pour les zones de fort gradient, un poids plus faible est utilisé. À cause du manque de fiabilité de l'équation de rendu dans ces zones, une pénalisation de ce coût doit ainsi être prise en compte. En effet, conformément à l'intuition derrière la théorie rétinex [102], les ombrages ne génèrent que de faibles changements d'intensité lorsque l'on considère des surfaces relativement lisses. De plus, des points clés ont tendance à apparaître au niveau des zones de forts gradients, créant ainsi des points dans la reconstruction considérée en entrée de l'algorithme. Cela crée ainsi des contraintes de positions dans ces zones, qui n'ont alors pas besoin de contrainte

supplémentaire utilisant les ombrages. Si p_k est la projection du point x dans la k-ième image clé, $\mathbf{n}$ la normale en x, et ∇I_r le gradient de l'image de référence, on définit :

$$E_{\text{shad}}(\rho, \mathcal{D}, \mathcal{C}) = \sum_{x \in \mathcal{D}} \exp\left(-\left\|\nabla I_r(p_r)\right\|^2\right) \sum_{k=0}^{n} \Lambda_k(p_k) \left(\frac{\rho(x)\,\sigma(x) \cdot \mathbf{n}(x)}{\mathcal{C}_k} - I_k(p_k)\right)^2 . \tag{6.21}$$

$E_{\text{pos}}(\mathcal{D})$ correspond aux contraintes de profondeurs. Il lie ainsi les points de la reconstruction et le nuage de points épars d'entrée P_{sp}. Si $d(p_k)$ correspond à la valeur de la profondeur correspondant au point p_k, et $(p_k)_z$ la valeur de z pour le point p_k, on a alors :

$$E_{\text{pos}}(\mathcal{D}) = \sum_{p_k \in P_{\text{sp}}} \left((p_k)_z - d(p_k)\right)^2 . \tag{6.22}$$

$E_{\text{reg}}(\mathcal{D})$ est un terme de régularisation prévenant la présence de bruit dans la reconstruction. On suppose alors que la surface que l'on cherche à reconstruire est relativement lisse (présence de rugosités négligeables). Ce terme s'écrit comme la norme des dérivées secondes de la surface à reconstruire :

$$E_{\text{reg}}(\mathcal{D}) = \sum_{X \in \mathcal{D}} \left(\left(\frac{\partial^2 d}{\partial^2 u}(X)\right)^2 + \left(\frac{\partial^2 d}{\partial^2 v}(X)\right)^2 + \left(\frac{\partial^2 d}{\partial u \partial v}(X)\right)^2 \right) . \tag{6.23}$$

$\mathcal{D}$ n'étant pas C^2, on considérera dans cette expression les limites "à gauche".

6.4.2 Résolution

Les inconnues de cette minimisation sont initialisées comme suit. Tout d'abord, le nuage de points en entrée est projeté dans la vue de référence. Cela nous donne donc un certain nombre de points pour lesquels on dispose d'une approximation de la profondeur. On ne considère pas directement la grille de reconstruction dans une première étape mais un ensemble d'autres divisions de l'image en cellules $\{D_0, D_1, ..., D_n\}$ tel que : D_0 contient une seule cellule correspondant à la vue entière, D_1 contient $2^2 = 4$ cellules de même taille divisant la vue (en deux sur la hauteur et en deux sur la largeur), et pour tout $n > 1$ D_n divise la scène en 2^{2n} cellules (2^n en largeur et 2^n en longueur). Un schéma correspondant à une telle découpe est illustré dans la Figure 6.4.

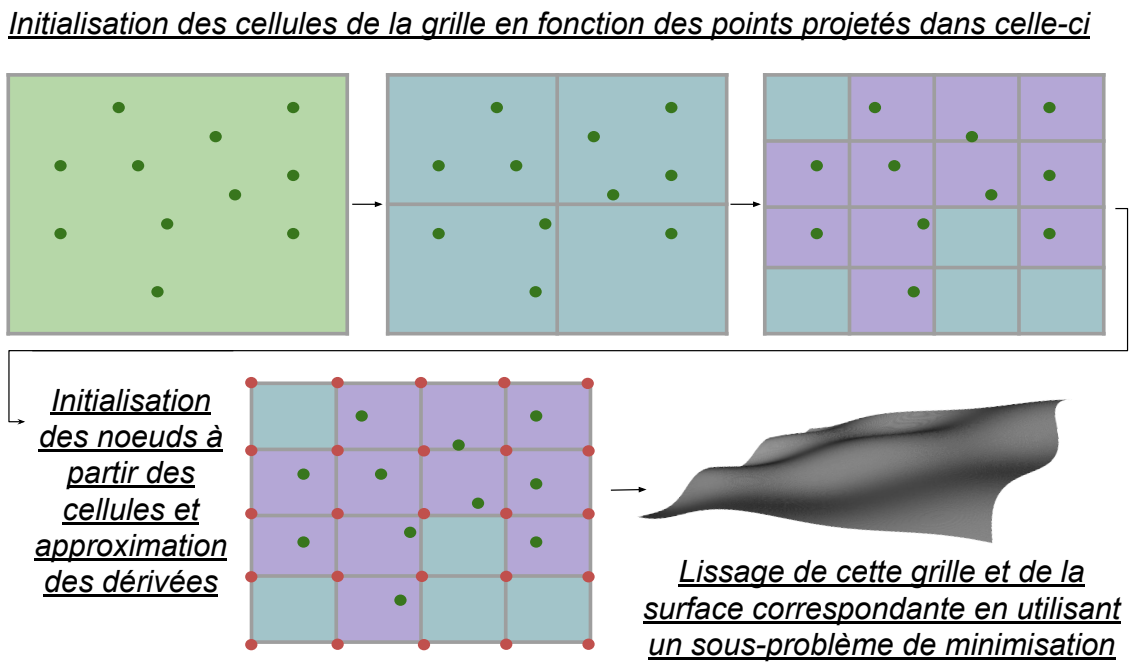

FIGURE 6.4 – Initialisation de la grille de profondeur.

Avec une telle subdivision, chaque cellule de l'ensemble D_k se divise alors en 4 cellules dans D_{k+1}. Le but est de donner une valeur de profondeur en chaque cellule de l'ensemble de plus grande subdivision, c'est-à-dire D_n, mais sans qu'aucune de ces cellules ne soit vide (non initialisée). On utilise pour cela les ensembles précédents. Ainsi, la cellule D_0 est initialisée avec la médiane des profondeurs des points se projetant dans la case. Les cases de D_1 sont ensuite mises à jour en considérant les points qui s'y projettent. Si aucun point ne s'y projette, on utilise la valeur de la case "parent" (donc ici la valeur de l'unique case de D_0). On itère ensuite le processus pour tous les ensembles jusqu'à obtenir la subdivision D_n. Cette procédure est illustrée dans la Figure 6.4. On initialise ensuite notre grille de reconstruction. Pour cela, on regarde pour chaque nœud sa profondeur (la première valeur du nœud) et on l'initialise avec la valeur de la cellule de D_n correspondante. On utilise ensuite les différences finies pour approximer les valeurs des dérivées premières et de la dérivée mixe à partir des valeurs des profondeurs de la

grille. On définit ensuite un problème de minimisation d'énergie intermédiaire comme

$$E_{\text{inter}}(\mathcal{D}) = E_{\text{pos}}(\mathcal{D}) + \lambda_{reg} E_{\text{reg}}(\mathcal{D}) \tag{6.24}$$

On utilise ensuite l'algorithme de Levenberg-Marquardt, implémenté avec ceres [2] pour minimiser cette énergie. Cela nous permet ainsi de lisser la géométrie de notre surface initiale. Les points de l'image de référence sont alors associés aux points de la surface correspondante (on récupère ainsi la position 3D de ces points, mais aussi la normale en ces points). On inverse alors l'équation de rendu dans l'image de référence pour obtenir une valeur d'albédo initiale associée à ces points. Pour chaque image clé, on considère alors chaque point ayant une association avec un point de l'image de référence et inversant l'équation de rendu à nouveau, on obtient alors un ensemble de valeurs candidates pour la réponse de caméra en question. On initialise alors la réponse de caméra correspondante avec la médiane de ces valeurs. Utilisant ces réponses de caméra, et cette fois-ci toutes les images de la reconstruction, on recalcule pour chaque point de l'image de référence les valeurs candidates pour l'albédo. On initialise finalement notre carte d'albédo avec la médiane des valeurs candidates considérées. On considère finalement le problème complet correspondant à l'énergie présentée dans l'équation (6.20) et on la minimise en utilisant encore une fois l'algorithme de Levenberg-Marquardt.

6.5 Méthodologie d'évaluation

6.5.1 Données d'évaluation

Afin de tester notre algorithme de reconstruction, nous utilisons des prises de vue, prises par le cœlioscope, de foies bovins en *ex vivo*. L'acquisition des données est illustrée dans la Figure 6.5.

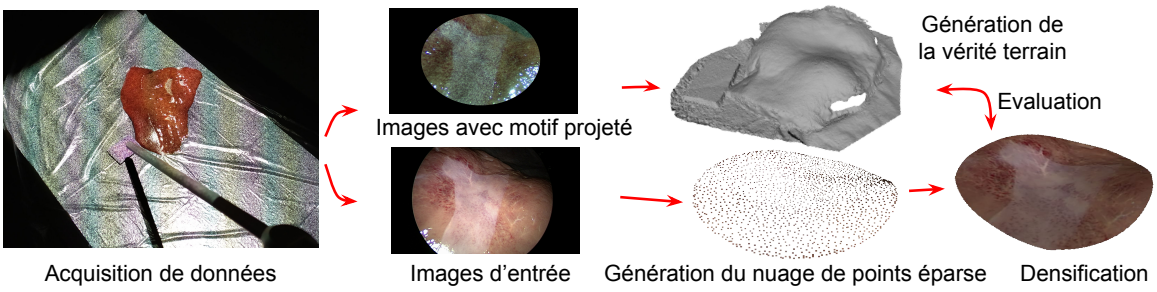

FIGURE 6.5 – Acquisition de données d'évaluation.

Afin d'acquérir la vérité terrain sur la géométrie de la scène, un projecteur numérique est installé au dessus de la scène. Ce dernier projette un motif fin et aléatoire sur le foie. Le cœlioscope est attaché à un bras mécanique et est déplacé pour voir la scène sous un grand nombre de vues. Pour chaque configuration on prend alors deux images : l'une avec le motif projeté et sans la lumière du cœlioscope et l'autre avec la lumière mais sans le motif. Les images avec motifs sont alors utilisées pour construire la vérité terrain en utilisant Photoscan [2] avec les mêmes paramètres que ceux discutés précédemment pour l'étape SfM. Cette fois-ci, cette étape est suivie d'une étape de densification en mode grande précision, avec un filtrage de la profondeur agressif, un maillage arbitraire considéré avec un grand nombre de faces et l'interpolation de profondeur activée. Les autres images (sans motif) sont quant à elles séparées en sous-ensembles plus petits d'une dizaine d'images. On utilise ensuite la méthode de SfM de Photoscan, sans activer la densification, pour générer des reconstructions à partir de ces sous-ensembles d'images. Le résultat est alors un nuage de points peu dense, et relativement bruité et un ensemble de poses associées (images et poses associées). Un facteur d'échelle est cependant présent dans cette reconstruction (comme expliqué dans le chapitre 2). Lorsqu'un tel facteur entre en jeu et que l'on considère le pipeline entier de recalage, une méthode simple pour résoudre ce problème consiste à cliquer un certain nombre de points sur le modèle et sur la reconstruction. Ces points cliqués doivent correspondre à des structures supposés rigides. Par exemple dans le cas du foie, on aura tendance à considérer des points de la vésicule biliaire. Le facteur d'échelle de la reconstruction est alors ajusté pour faire correspondre les distances entre ces points dans la reconstruction et dans la scène. Dans notre cas, ne disposant pas de modèle (car ne considérant

2. https://www.agisoft.com

pas le pipeline de recalage entier), on doit procéder différemment. Ainsi, un marqueur de taille connu est positionné dans la scène. Ce dernier n'est pas forcément vue (ni reconstruit) dans tous les sous-ensembles d'images considérés, cependant il est vu dans l'ensemble entier. Nous considérons donc la reconstruction obtenue avec les images avec motif projeté. Dans celle-ci, le marqueur est visible et reconstruit. Il est alors possible d'ajuster correctement le facteur d'échelle dans celle-ci. De plus, les images clés de cette reconstruction correspondent à des images clés des reconstructions obtenues avec les sous-ensembles d'images. Ainsi, chaque image clé des reconstructions des sous-ensembles d'images peut être associée à une image clé de la reconstruction des images à motif. Le facteur d'échelle se retrouvant aussi dans les poses (comme expliqué dans le chapitre 2), il faut ainsi aussi les mettre à jour. Ces reconstructions, remises à la bonne échelle, peuvent alors être utilisées en entrée de notre algorithme. On compare ainsi les reconstructions générées par notre algorithme en utilisant les différents modèles discutés dans le chapitre précédent, mais aussi les reconstructions obtenues en utilisant deux autres configurations. La première considère directement une reconstruction obtenue avec Photoscan avec la densification activée (DMS). La seconde considère la reconstruction obtenue avec notre algorithme mais en ne prenant en compte que les termes de régularisation et de position (pas les termes sur les ombrages). Pour chacune de ces deux configurations, la couleur des points de la reconstruction est calculée à posteriori en considérant la médiane des points dans les images correspondant aux points de la reconstruction.

6.5.2 Métriques utilisées

On utilise deux métriques pour évaluer la qualité des reconstructions. On considère tout d'abord la reconstruction obtenue avec les images sur lesquelles le motif a été projeté, que l'on projette alors dans le plan image de l'image de référence du sous-ensemble considéré. Cela nous permet d'obtenir une vérité terrain pour la profondeur de chaque pixel de l'image. La valeur absolue de la différence en millimètre entre cette profondeur et celle obtenue avec notre méthode correspond alors à ce que l'on appelle *erreur géométrique*. La deuxième métrique utilisée est ce que l'on appelle *erreur photométrique*. Ainsi, utilisant les résultats de notre algorithme de densification (positions 3D et normales pour chaque points de l'image) ainsi que le modèle de lumière calibré et

les réponses de caméra estimées, on génère, utilisant l'équation de rendu les images correspondant aux différentes images clés considérées dans notre algorithme. On compare ainsi les images générées aux images réelles correspondantes. Cependant, considérer directement l'erreur pixel à pixel n'est pas une bonne métrique pour déterminer la similarité entre deux images. En effet, pour des images ayant de fortes discontinuités (correspondant à un fort niveau de détail), un simple décalage entre ces deux images induit une forte erreur avec une telle méthode. Ainsi, si un léger désalignement existe entre les images et la reconstruction, et que cette différence pixel à pixel est utilisée, alors un score anormalement élevé est observé. Pour palier ce problème, on utilise une méthode pour calculer notre erreur photométrique bien plus représentative et largement utilisée dans la littérature : la Dissymétrie Structurelle (DSSIM) [205]. Elle compare pour des fenêtres localisées autour de chaque point à considérer, une valeur dépendante la valeur suivante :

$$DSSIM(u,v) = \frac{1 - SSIM(u,v)}{2}, \tag{6.25}$$

avec

$$SSIM(u,v) = l(u,v).c(u,v).s(u,v) = \frac{(2\mu_u\mu_v + c_1)(2\sigma_u\sigma_v + c_2)(cov_{uv} + c_3)}{(\mu_u^2 + \mu_v^2 + c_1)(\sigma_u^2 + \sigma_v^2 + c_2)(\sigma_u\sigma_v + c_3)}, \tag{6.26}$$

avec μ_u la moyenne de u, μ_v la moyenne de v, σ_u^2 la variance de u, σ_v^2 la variance de v, cov_{uv} la covariance de u et v et c_1, c_2 et c_3 des valeurs destinées à stabiliser la division quand le dénominateur est faible :

$$\begin{aligned} c_1 &= (k_1 L)^2 \\ c_2 &= (k_2 L)^2 \\ c_3 &= \frac{c_2}{2}, \end{aligned} \tag{6.27}$$

avec L la taille de la gamme des valeurs possible pour les pixels (255), $k_1 = 0.01$ et $k_2 = 0.03$. Définir l'erreur photométrique à partir de cette valeur permet ainsi de mieux rendre compte des différences visibles entre les images réelles et simulées. Un exemple illustrant cette erreur est présent dans la Figure 6.6 où la zone cerclée de vert est simulée à l'aide de la reconstruction obtenue par notre algorithme et comparée à l'image véritable en utilisant la DSSIM.

FIGURE 6.6 – Erreur photométrique obtenue en utilisant le modèle SLS pour générer la reconstruction : image cible (gauche), image générée à l'intérieur du cercle vert (centre) et DSSIM entre les deux images (droite).

6.6 Résultats expérimentaux

On considère pour les reconstructions des grilles de 65×65 nœuds associées avec des cartes de profondeur de résolution 1920×1080. Afin de calibrer les modèles de lumière à utiliser, nous avons considéré deux ensembles d'images pour le calibrage : un ensemble de faible taille (environ 15 images, amenant à un calibrage peu précis) et un ensemble de grande taille (comportant environ 60 images, menant à un calibrage plus précis). Cela a été considéré afin d'observer les effets d'un calibrage plus précis sur la reconstruction finale, et ce pour chaque modèle de lumière considéré.

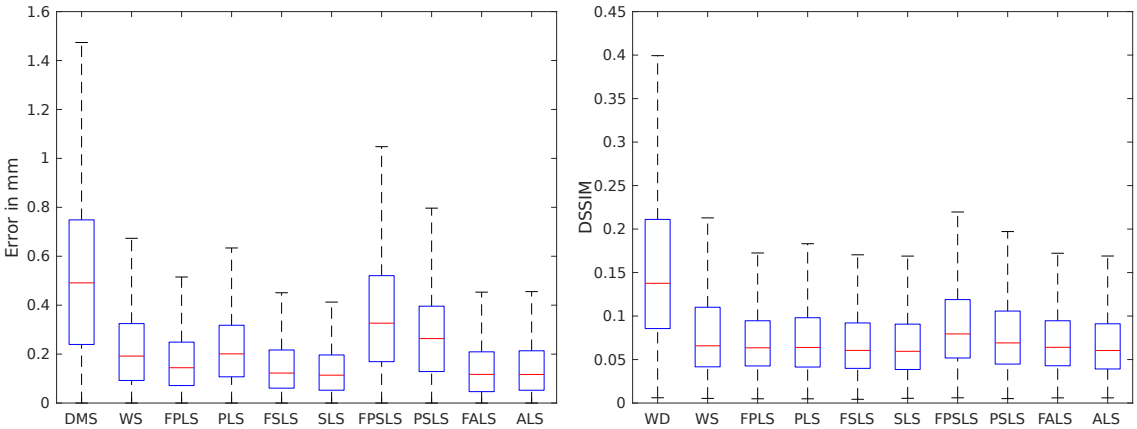

FIGURE 6.7 – Boîte à moustaches (avec médiane, quartiles et valeurs extrêmes), montrant les erreurs de reconstruction en utilisant un modèle de lumière calibré à l'aide d'un grand nombre d'images : erreur géométrique (gauche) et DSSIM (droite).

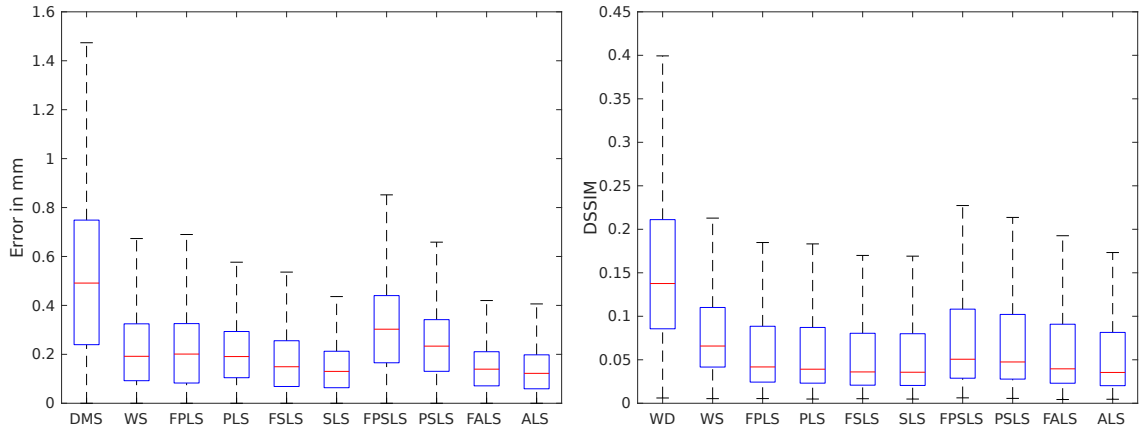

FIGURE 6.8 – Boîte à moustaches, montrant les erreurs de reconstruction en utilisant un modèle de lumière calibré à l'aide d'un faible nombre d'images : erreur géométrique (gauche) et DSSIM (droite).

Les résultats sont présentés dans les Figures 6.7 et 6.8. Les surfaces considérées pour les reconstructions sont des patches circulaires ayant des diamètres d'environ 5 cm et étant vues à une distance située entre 2 et 4 cm. Nos tests ont été réalisés avec un Intel® Xeon® E5-1620 avec du code C++ non optimisé, et 6 threads dédiés à l'optimisation non-linéaire. Chaque reconstruction générée est obtenue en un temps situé entre 55 s (pour des reconstructions obtenues avec un modèle FPLS) et 450 s (pour le modèle ALS). Si un calcul plus rapide est nécessaire, le nombre de nœuds de la grille ainsi que la résolution des cartes de profondeur et des images considérées peut être réduit, au prix d'une réduction de la précision finale de la densification. Ainsi, pour des grilles de taille 9×9, et des résolutions de 384×216, le temps d'exécution est compris cette fois-ci entre 30 s (FPLS) et 185 s (ALS).

On observe ainsi une amélioration conséquente entre nos résultats et les reconstructions générées directement avec Photoscan en termes d'erreur géométrique mais aussi en terme d'erreur photométrique et cela pour la plupart des modèles de lumière lorsque l'on utilise un calibrage précis (utilisant un grand nombre d'images). On observe ainsi, pour les meilleurs modèles de lumière testés des améliorations d'environ 0,4 mm pour l'erreur géométrique et de 0,1 unité de DSSIM pour l'erreur photométrique. On observe de plus une amélioration de la qualité des reconstructions entre des reconstructions générées en utilisant les informations sur les ombrages et d'autres générées sans ces informations.

Nos confirmons aussi les conclusions du chapitre précédent vis-à-vis des meilleurs modèles de lumière adaptés au contexte de la cœlioscopie, même si les différences entre les modèles sont moins importants que dans le chapitre précédent. Ainsi le modèle SLS semble encore une fois être un excellent choix lorsqu'il s'agit d'estimer l'illumination de la scène cœlioscopique. Il produit des résultats très similaires au modèle ALS tout en étant bien moins complexe. De plus, les modèles PLS et SLS ne produisent pas de résultats aberrants même lorsque l'on considère un calibrage de lumière peu précis. Au contraire, le modèle PSLS semble, comme dans le chapitre précédent, être peu stable. À noter que tous les modèles produisent des erreurs géométriques infra-millimétriques, démontrant ainsi le grand potentiel de la méthode décrite.

6.7 Conclusion

Nous avons présenté dans ce chapitre un algorithme de SAMVS adapté au contexte de la chirurgie cœlioscopique. Ce dernier permet, à partir d'un système de vues dont les poses de caméra sont connues, de reconstruire la surface observée, en s'appuyant sur les ombrages de la scène. Celui-ci utilise un modèle calibré de la lumière de l'endoscope afin de générer des contraintes fortes entre la surface à raffiner et les ombrages présents dans les images cœlioscopiques. Notre méthode est évaluée sur des foies bovins *ex vivo*. Les erreurs géométriques observées sont sub-millimétriques et dépassent la précision des reconstructions obtenues avec le logiciel Photoscan, connu pour la grande fiabilité de ses reconstructions. Les reconstructions obtenues en considérant différents modèles de lumière sont comparées, et encore une fois, le modèle SLS apparaît comme un excellent candidat pour estimer l'illumination de la scène cœlioscopique. Les reconstructions obtenues ne correspondent cependant qu'à une unique vue de référence. Dans de futurs travaux, la fusion de ces surfaces issues de vues de référence devra ainsi être considérée afin d'obtenir une reconstruction de la scène dans son entièreté. De plus, des améliorations sur le temps d'exécution de la méthode sont aussi à envisager, certaines résolutions nécessitant plusieurs minutes de calcul. Des études supplémentaires sont aussi nécessaires afin de pouvoir gérer les occultations se produisant dans les images. Enfin, une application principale extrêmement importante à considérer à la suite de ces travaux est le transfert d'albédo à un modèle préalablement recalé à la surface, ainsi que son utilisation afin de générer des contraintes sur les ombrages pour le recalage

déformable temps réel de ce modèle texturé à la scène cœlioscopique.

Chapitre 7

Conclusion et travaux futurs

Cette thèse a exposé nos contributions aux méthodes de recalage déformable pour la chirurgie mini-invasive abdominale augmentée, aux techniques permettant d'améliorer leur précision telles que la modélisation de l'illumination dans la scène cœlioscopique, mais aussi aux méthodologies d'évaluation de telles méthodes. Ce chapitre donne nos conclusions par rapport aux problématiques discutées à travers les différents chapitres de ce manuscrit et discute les perspectives de travaux futurs.

7.1 Conclusions

Afin de permettre la réalité augmentée pour la chirurgie mini-invasive abdominale, des données pré-opératoires doivent être recalées à la scène per-opératoire. Des déformations importantes surviennent alors et un recalage déformable doit être envisagé. Deux contextes ont été étudiés durant cette thèse : le cas du recalage de données hépatiques à la scène cœlioscopique et le cas du recalage postural pour lequel des données correspondant à une coupe de l'abdomen du patient doivent être recalées au patient sur la table d'opération. Le pipeline général de recalage envisagé et discuté dans cette thèse est cependant similaire pour ces deux cas, et seul le type de contraintes à prendre en compte est réellement différent. Le pipeline général considéré dans cette thèse est le suivant. Tout d'abord, l'organe ou la section de l'abdomen considéré ainsi que différentes structures internes, tels que les tumeurs, les vaisseaux sanguins, ou les os, sont segmentés dans les images pré-opératoires TDM ou IRM. Cela permet ainsi de générer un modèle biomécanique correspondant aux données que l'on souhaite recaler. Les différentes structures internes peuvent alors être utilisées pour créer de l'hétérogénéité dans la rigidité considérée pour le modèle. Des images RGB ou RGB-D sont ensuite capturées le jour de l'opération et utilisées afin de reconstruire une surface de la scène observée. Cette surface est utilisée en tant que cible d'un recalage initial du modèle. Une fois recalé à cette surface, le modèle peut être texturé et les informations de texture peuvent être utilisées dans la suite du recalage. Cela permet d'estimer des déformations plus complexes du modèle. Dans cette thèse, nous nous sommes ainsi intéressés à différentes étapes de ce pipeline. Notre intérêt premier a porté sur le recalage déformable initial, dans le cas de données hépatiques et dans celui de données de postures. Nous nous sommes aussi intéressés à l'estimation de l'éclairage dans la scène cœlioscopique. L'intérêt d'une telle estimation est double car il permet non seulement d'obtenir une surface sur laquelle se recaler possédant une géométrie plus précise, mais aussi des informations de texture (sous la forme d'albédo) indépendantes de l'éclairage, pouvant ainsi permettre de meilleures mises en correspondance avec les images lors de l'étape de recalage. Les conclusions détaillées de nos travaux sont listées ci-dessous.

7.1.1 Recalage de données hépatiques en cœlioscopie

Nous avons étudié le cas du recalage de données hépatiques pré-opératoires dans la scène cœlioscopique per-opératoire. En tant que cible de recalage, une reconstruction de surface obtenue avec une méthode de SfM existante a été utilisée. À cause des fortes déformations induites par l'insufflation du patient et par l'évolution de la maladie, un recalage déformable doit être considéré. Nous avons donc proposé un algorithme de recalage déformable basé sur une méthode énergétique. Un modèle biomécanique homogène de type *masse-ressort* du foie a donc été considéré et différentes contraintes ont été appliquées à ce dernier. On a ainsi utilisé des contraintes de correspondance avec la surface reconstruite basée sur une énergie de type ICP et des contraintes utilisant la silhouette de l'organe pour se repérer. Nous avons utilisé de plus des contraintes permettant de prévenir d'éventuelles collisions internes pouvant subvenir dans le modèle. Nous avons ainsi montré, pour des modèles biomécaniques simples tels que des modèles *masse-ressort*, que ce genre de considérations est important car il permet non seulement d'empêcher des collisions internes lorsque le modèle considéré n'a pas une géométrie convexe, mais aussi de potentielles inversions de tétraèdres dans le modèle. Pour de telles méthodes de recalage, nous nous sommes aussi intéressés à la façon d'évaluer de tels algorithmes au mieux et aux différentes solutions existantes dans la littérature. Les méthodes d'évaluation les plus fiables sont ainsi des méthodes considérant des données *in vivo*, correspondant ainsi aux conditions quasi-réelles, et une métrique d'erreur utilisant la TRE. Le manque de base de données publique de ce type nous a amené à créer la nôtre, à base de données porcines. Ce jeu de données est le premier de ce type, permettant de calculer la TRE de façon semi-dense à la surface mais aussi à l'intérieur de l'organe. Il permet une meilleure vision des progrès réalisés dans ce domaine en permettant un comparaison quantitative entre les différents algorithmes.

7.1.2 Recalage postural

Nous avons examiné le cas particulier du recalage de posture, c'est-à-dire du recalage de données correspondant à des sections de l'abdomen d'un patient sur la table d'opération. En tant que cible du recalage, des images provenant d'une caméra RGB-D et fusionnées en une surface unique sont utilisées. Tandis que la plupart des travaux existants considèrent un recalage rigide pour de telles applications, nous avons pro-

posé un algorithme de recalage déformable basé sur une méthode énergétique, dérivé de l'algorithme déjà discuté précédemment. Nous avons ainsi montré qu'une telle considération est importante car de fortes déformations peuvent subvenir dues au changement de position du patient entre la prise de données volumiques pré-opératoires et la capture d'images le jour de l'opération. Afin de prendre en compte la rigidité des os du patient, un modèle biomécanique hétérogène a été considéré et les bénéfices d'une telle approche ont été constatés. Enfin, les collisions avec la table, permettant d'obtenir une meilleure précision au niveau de l'interface entre le modèle et la table, ont été considérées. Un jeu de données permettant une évaluation quantitative et dense de l'erreur de recalage au niveau de la surface du modèle a été lui aussi proposé. Ce jeu de données est important car il permet de comparer différents algorithmes de recalage. En tant qu'applications utilisant le recalage, le suivi de dose d'irradiation de la peau du patient au cours de différentes séances de radiothérapie, ainsi que l'aide à l'insertion d'aiguilles radiofréquence ont été considérés. La précision actuelle de notre algorithme ne lui permet pas encore d'être utilisé dans ces contextes. Néanmoins, son utilisation est tout indiquée dans des problématiques d'échographie assistée par la robotique.

7.1.3 Modélisation de l'éclairage en cœlioscopie

Nous avons étudié le problème de la modélisation de l'illumination dans la scène cœlioscopique. Cette problématique est importante car elle permet d'obtenir des contraintes denses entre la surface de l'organe et les images, et de les utiliser dans différentes étapes du pipeline général de recalage considéré. Son utilisation permet ainsi d'obtenir de meilleures reconstructions de la scène per-opératoire, en termes de géométrie mais aussi en termes de texture, que l'on utilise ensuite pour se recaler. De plus, une fois le modèle texturé en utilisant les informations provenant d'un recalage initial, l'éclairage peut être utilisé afin de générer des contraintes denses et fiables à utiliser pour l'étape de recalage temps réel. Les conditions particulières en cœlioscopie permettent de plus une contrainte bien plus forte de l'éclairage que dans le cas général, la seule source primaire de lumière provenant du cœlioscope, et cette source étant fixe par rapport au centre optique de la caméra. Différents modèles de lumière ont été discutés et une méthode de calibrage générale de ces modèle a été proposée. Une fois calibrés, la qualité de l'illumination prédite par ces différents modèles a été comparée. De ces expérimentations, nous

avons conclu que, malgré sa simplicité, le modèle SLS, consistant en une source lumineuse ponctuelle émettant avec une direction de propagation donnée, est un candidat de choix pour modéliser l'illumination en cœlioscopie.

7.1.4 Raffinage de reconstruction de surface de la scène per-opératoire en cœlioscopie

Grâce aux modèles de lumière définis précédemment, nous avons défini un algorithme de SAMVS tirant parti des ombrages présents dans la scène. Cet algorithme permet de raffiner et densifier une reconstruction de surface en tirant parti de l'ombrage dans la scène. À partir d'une reconstruction de surface éparse de type SfM ou SLAM, composée d'un nuage de points et des images clés associées, nous générons ainsi, pour une vue de référence une surface continue sous la forme d'une grille de profondeur, composée de patches de fonctions bicubiques, et d'une valeur d'albédo par pixel. Les inconnues du problème associées à cette reconstruction sont alors les valeurs des nœuds de cette grille et les valeurs de l'albédo. Une telle représentation a l'intérêt d'être relativement simple à manier. L'expression de contraintes sur les ombrages en utilisant une telle paramétrisation est aussi simple. Notre algorithme a été évalué en prenant en compte différents modèles de lumière calibrés. Encore une fois, le modèle SLS apparaît comme le meilleur modèle à considérer. Grâce à notre méthode, une carte d'albédo de la surface considérée, indépendante de l'éclairage de la scène, est extraite. Dans le contexte du recalage, il devient alors possible de transférer cette texture au modèle recalé afin de l'utiliser ensuite pour l'étape de recalage temps réel. Enfin, il est à noter que notre méthode permet d'obtenir un précision infra-millimétrique au niveau des surfaces générées.

7.2 Perspectives

Les différentes méthodes présentées dans ce manuscrit peuvent être améliorées. De plus, une étape clé, permettant de définir entièrement de bout en bout le pipeline de recalage est encore actuellement manquante : le transfert de l'albédo au modèle, suivi de l'étape de recalage temps réel utilisant celui-ci. Les commentaires suivants permettent ainsi de définir un certain nombre de points nécessitant une étude plus approfondie.

7.2.1 Utilisation de modèle biomécanique hétérogène de foie et évaluation sur un modèle clinique

L'une des principales critiques pouvant être faites concernant nos travaux sur le recalage de données hépatiques dans la scène cœlioscopique provient du type de modèle biomécanique utilisé. En effet, durant cette thèse, l'intérêt a été porté d'avantage sur les différents types de contraintes pouvant être appliquées au modèle biomécanique que sur le modèle lui même. Le modèle utilisé est relativement simple : un modèle homogène de type *masse-ressort*. Bien que simple, des résultats de recalage assez bons ont tout de même étés observés. Une perspective pour de futurs travaux serait ainsi d'utiliser des modèles biomécaniques plus complexes, ou tout du moins prendre en compte différentes rigidités pour les différentes structures du foie et étudier l'impact de ces changements sur la qualité de recalage. De tels travaux nécessitent un moyen de déterminer les caractéristiques physiques des différentes structures composant le foie [166, 222, 68, 164]. De plus, les évaluations ayant été, durant cette thèse, majoritairement réalisées sur des données animales, une meilleure évaluation utilisant des données cliniques serait elle aussi nécessaire.

7.2.2 Utilisation de données ultrasons en tant que contraintes supplémentaires pour le recalage

Lors de nos expérimentations, nous avons été amenés à travailler avec des sondes échographiques recalées rigidement par rapport à la vue principale considérée. Cela a été utilisé afin d'observer le résultat de recalage conjointement avec le plan de l'échographe et les images dans celui-ci. Les données provenant de la sonde échographique sont riches car elles renseignent sur la position de structures internes ne pouvant être observées directement [103]. Extraire de telles informations et les utiliser pour contraindre des points à l'intérieur du modèle biomécanique pourrait alors grandement augmenter la précision du recalage. De telles contraintes pourraient être utilisées pour le recalage initial mais aussi pour l'étape de recalage temps réel afin de corriger certaines imprécisions. Les sondes échographiques étant de plus largement utilisées lors de cœlioscopies hépatiques, une telle méthode serait ainsi bien adaptée à ce contexte.

7.2.3 Fusion de surface pour l'algorithme de SAMVS

L'algorithme de SAMVS discuté précédemment ne reconstruit à l'heure actuelle que la surface associée à une vue de référence. L'un des principaux travaux futurs à considérer est ainsi de généraliser cet algorithme à la reconstruction de la scène dans son entièreté. Pour cela, différentes grilles de profondeurs doivent être construites conjointement, en prenant en compte des contraintes de cohérence entre ces dernières, puis fusionnées en une surface unique [51, 198, 47, 217, 206]. De plus, les occultations pouvant survenir dans les images doivent elles aussi être prises en compte.

7.2.4 Transfert de carte d'albédo au modèle et recalage temps réel

Une étape clé, permettant de considérer le pipeline de recalage dans son ensemble est absente de nos travaux. Cette étape consiste, une fois le recalage initial réalisé, entre le modèle pré-opératoire et la scène cœlioscopique, à transférer les informations de l'albédo de la surface considérée en tant que cible du recalage, au modèle biomécanique. Cette information doit ensuite permettre de définir un recalage temps réel s'appuyant sur ces informations d'albédo et permettant grâce aux contraintes denses pouvant alors être définies entre le modèle et les images, de capturer des déformations bien plus importantes au niveau du modèle, comme cela a déjà été proposé dans [59]. Deux types de contraintes peuvent être définis. Utilisant la configuration précédemment calculée du modèle, l'ombrage peut être en grande partie soustrait de l'image considérée. Des associations denses entre le modèle et l'image peuvent alors être calculées [45]. Ensuite, des contraintes utilisant directement l'équation de rendu, similaires à celles définies dans notre algorithme de SAMVS, peuvent être utilisées pour raffiner cette configuration.

Annexe A

Réduction de l'espace des déformations

Cette annexe décrit la méthode utilisée dans nos travaux afin de réduire les dimensions du problème de minimisation associé au recalage, et de permettre ainsi une résolution plus rapide. L'idée principale est ici de définir une forme quadratique semi-définie positive pour caractériser la complexité des déformations testées. Par complexité des déformations, on entend ici distance entre la déformation considérée et une déformation rigide. La décomposition spectrale de cette forme quadratique permet ainsi de définir un sous-espace dans lequel la complexité des déformations reste faible. Le nombre de vecteurs de base de ce sous-espace permet alors de contrôler la quantité de déformation que l'on souhaite autoriser. Il est alors possible de modifier la Jacobienne associée au problème de recalage afin que la solution ne soit recherchée que dans ce sous-espace. Deux méthodes, permettant de définir une telle forme quadratique, sont décrites dans ce chapitre. La première propose de définir une distance entre la déformation considérée et la plus proche transformation rigide correspondante. La seconde méthode discutée, propose de considérer uniquement des données centrées. Ainsi, les translations ne sont plus représentées dans la forme définie. Le noyau de celle-ci est ainsi de plus faible dimension. Les translations locales après centrage sont ainsi réparties dans l'image de cette forme. Cela permet ainsi d'être plus riche dans la caractérisation de la complexité des déformations. Ces deux méthodes sont décrites dans cette annexe, permettant une meilleure compréhension de la méthode dans son intégralité. Cependant, dans nos travaux, seule la seconde méthode est utilisée. La façon dont la Jacobienne du problème de recalage est modifiée à partir du sous-espace retenu est elle aussi décrite dans cette annexe.

A.1 Calcul simple des sous-espaces de faibles déformations

On se place ici dans le système de coordonnées lié au modèle (c'est-à-dire aux images TDM ou IRM utilisées). On considère pour chaque point du maillage à déformer les n plus proches voisins de ce dernier aux sens de la distance euclidienne. Si le maillage comporte m points, on crée ainsi m ensembles de $n+1$ points comportant le point et ses voisins. On note alors pour $i \in [\![1, m]\!]$, X_i la matrice contenant ces points, correspondant ainsi à l'état non-déformé du modèle :

$$X_i = \begin{bmatrix} x_{i,0} & y_{i,0} & z_{i,0} \\ \vdots & \vdots & \vdots \\ x_{i,n} & y_{i,n} & z_{i,n} \end{bmatrix}, \tag{A.1}$$

avec $\begin{bmatrix} x_{i,0} & y_{i,0} & z_{i,0} \end{bmatrix} = \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}$ le i-ième point du maillage considéré et $\begin{bmatrix} x_{i,k} & y_{i,k} & z_{i,k} \end{bmatrix}$ les voisins de ce point, correspondant à certains indices de points du maillage. Si l'on transforme cet ensemble de points X_i en n'importe quel autre ensemble de points Y_i, correspondant à une configuration déformée du modèle, on peut alors définir la transformation affine 3D la plus proche entre ces deux ensembles comme étant :

$$M_i, T_i \stackrel{\text{def}}{=} \underset{M \in \mathbb{R}^{3\times 3}, T \in \mathbb{R}^3}{\operatorname{argmin}} \sum_{j=0}^{n} \left\| M X_i(j,:)^\top + T - Y_i(j,:)^\top \right\|^2. \tag{A.2}$$

On écrit alors $\widetilde{X}_i$ l'ensemble de points initiaux en utilisant les coordonnées homogènes :

$$\widetilde{X}_i = \begin{bmatrix} x_{i,0} & y_{i,0} & z_{i,0} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{i,n} & y_{i,n} & z_{i,n} & 1 \end{bmatrix}. \tag{A.3}$$

On peut alors écrire :

$$\sum_{j=0}^{n} \left\| M_i X_i(j,:)^\top + T_i - Y_i(j,:)^\top \right\|^2 = \left\| \widetilde{X}_i \begin{bmatrix} M_i^\top \\ T_i^\top \end{bmatrix} - Y_i \right\|^2. \tag{A.4}$$

$\begin{bmatrix} M_i^\top \\ T_i^\top \end{bmatrix}$ est donc la solution aux moindres carrés du système suivant :

$$\min_{\Omega \in \mathbb{R}^{4\times 3}} \left\| \widetilde{X}_i \; \Omega - Y_i \right\|^2 . \tag{A.5}$$

Or, la solution d'un tel système est :

$$\Omega = \widetilde{X}_i^{\;+} Y_i, \tag{A.6}$$

ou $\widetilde{X}_i^{\;+}$ est la pseudo-inverse de $\widetilde{X}_i$. On peut alors réécrire le système et on obtient :

$$\left\| \widetilde{X}_i \begin{bmatrix} R_i^\top \\ T_i^\top \end{bmatrix} - Y_i \right\|^2 = \left\| \left(\widetilde{X}_i \widetilde{X}_i^{\;+} - I \right) Y_i \right\|^2 \tag{A.7}$$

On note alors :

$$F = \widetilde{X}_i \widetilde{X}_i^{\;+} - I, \tag{A.8}$$

une matrice intermédiaire permettant de calculer la forme définissant la quantité de déformation d'une transformation, et :

$$Y_i = \begin{bmatrix} Y_{i,x} & Y_{i,y} & Y_{i,z} \end{bmatrix}, \tag{A.9}$$

avec $Y_{i,x}$, $Y_{i,y}$ et $Y_{i,z}$ les vecteurs colonnes composant Y_i. On remarque que :

$$F \; Y_i = \begin{bmatrix} F \; Y_{i,x} & F \; Y_{i,y} & F \; Y_{i,z} \end{bmatrix}. \tag{A.10}$$

Et donc :

$$\|F \; Y_i\|^2 = \left\| \begin{bmatrix} F \; Y_{i,x} & F \; Y_{i,y} & F \; Y_{i,z} \end{bmatrix} \right\|^2 = \|F \; Y_{i,x}\|^2 + \|F \; Y_{i,y}\|^2 + \|F \; Y_{i,z}\|^2 . \tag{A.11}$$

On note alors :

$$Q_i = F^\top \; F = \left(\widetilde{X}_i \widetilde{X}_i^{\;+} - I \right)^\top \left(\widetilde{X}_i \widetilde{X}_i^{\;+} - I \right), \tag{A.12}$$

et on réécrit :

$$\|F \; Y_i\|^2 = Y_{i,x}^\top \; Q_i \; Y_{i,x} + Y_{i,y}^\top \; Q_i \; Y_{i,y} + Y_{i,z}^\top \; Q_i \; Y_{i,z}. \tag{A.13}$$

On remarque ici que Q_i est symétrique. On note ensuite Y, matrice contenant l'ensemble de tous les points déformés du maillage, pouvant être décomposées en trois vecteurs Y_x, Y_y et Y_z, c'est-à-dire :

$$Y = \begin{bmatrix} \widetilde{x}_0 & \widetilde{y}_0 & \widetilde{z}_0 \\ \vdots & \vdots & \vdots \\ \widetilde{x}_m & \widetilde{y}_m & \widetilde{z}_m \end{bmatrix} = \begin{bmatrix} Y_x & Y_y & Y_z \end{bmatrix}. \tag{A.14}$$

À partir de l'ensemble des Q_i, on construit alors Q tel que :

$$\begin{cases} Y_x^\top Q Y_x = \sum_{i=1}^m {Y_{i,x}}^\top Q_i Y_{i,x} \\ Y_y^\top Q Y_y = \sum_{i=1}^m {Y_{i,y}}^\top Q_i Y_{i,y} \\ Y_z^\top Q Y_z = \sum_{i=1}^m {Y_{i,z}}^\top Q_i Y_{i,z} \end{cases} \tag{A.15}$$

Pour cela, on initialise tout d'abord Q à une matrice nulle de taille $m \times m$. Ensuite pour chaque Q_i, on regarde les indices des points contenus dans les X_i (et donc dans les Y_i). Chaque ligne et colonne de Q_i se voit alors attribuer l'indice correspondant à l'ordre des points dans le maillage. Les lignes et colonnes de Q_i sont alors ajoutées aux lignes et colonnes de Q correspondant à ces indices. Les Q_i étant symétriques, Q est alors symétrique elle aussi. De plus, Q est semi-définie positive. En effet, pour tout vecteur Y_v avec $v \in \{x, y, z\}$ on a :

$$Y_v^\top Q Y_v = \sum_{i=1}^m {Y_{i,v}}^\top Q_i Y_{i,v} = \sum_{i=1}^m \left\| \left(\widetilde{X}_i \widetilde{X}_i^{\,+} - I \right) Y_{i,v} \right\|^2 \geq 0. \tag{A.16}$$

Après calcul des valeurs propres de Q en utilisant une décomposition en valeurs singulières, on classe ces dernières par ordre croissant : $0 \leq \lambda_0 < \cdots < \lambda_r$ et on dénote $E_0, \cdots, E_r$ les sous-espaces propres associés à celles-ci. Chaque sous-espace E_k s'écrit alors $E_k = \mathrm{vec}(v_{k,0}, \cdots, v_{k,\alpha})$ avec $\alpha = \dim(E_k)$. Toute configuration déformée $Y = \begin{bmatrix} Y_x & Y_y & Y_z \end{bmatrix}$ peut alors être décomposée en :

$$Y = \begin{bmatrix} \sum_{i=0}^{r} \sum_{j=0}^{\dim(E_i)} \lambda_{i,j,x} v_{i,j} & \sum_{i=0}^{r} \sum_{j=0}^{\dim(E_i)} \lambda_{i,j,y} v_{i,j} & \sum_{i=0}^{r} \sum_{j=0}^{\dim(E_i)} \lambda_{i,j,z} v_{i,j} \end{bmatrix}. \tag{A.17}$$

Les déformations "faibles" (au sens de proche de transformations localement affines)

appartiennent alors à l'espace $\widetilde{E}_l = \bigcup_{i=0}^{l} E_i$ avec l d'autant plus faible que les déformations autorisées seront "faibles" elles aussi. L'espace de recherche à utiliser est donc $\left(\widetilde{E}_l\right)^3$ avec l faible. On ne cherche ainsi qu'a considérer les déformations du type :

$$Y = \left[\sum_{i=0}^{l}\sum_{j=0}^{\dim(E_i)} \lambda_{i,j,x} v_{i,j} \quad \sum_{i=0}^{l}\sum_{j=0}^{\dim(E_i)} \lambda_{i,j,y} v_{i,j} \quad \sum_{i=0}^{l}\sum_{j=0}^{\dim(E_i)} \lambda_{i,j,z} v_{i,j}\right]. \tag{A.18}$$

A.2 Centrage des données

En utilisant la méthode décrite précédemment, toutes les translations du modèle correspondent à des éléments du noyau de Q. Une solution alternative pour calculer un autre espace de déformations "faibles" est alors de ne considérer que des données centrées et de ne plus considérer des transformations localement affines pour calculer Q, mais uniquement des transformations localement linéaires après centrage. De ce fait, les translations du modèles ne sont plus associées à des éléments du noyau de Q. Au final, $\left(\widetilde{E}_l\right)^3$ sera alors plus représentatif de déformations "faibles" car les translations locales après centrage seront réparties dans les autres espaces propres. On considère ainsi cette fois-ci :

$$M_i \stackrel{\text{def}}{=} \underset{M\in\mathbb{R}^{3\times 3}}{\operatorname{argmin}} \sum_{j=1}^{n} \left\| M X_i(j,:)^\top B - Y_i(j,:)^\top B \right\|^2, \tag{A.19}$$

avec B matrice de centrage s'écrivant :

$$B = \begin{bmatrix} 1-\frac{1}{m} & & & & \\ & \ddots & & (-\frac{1}{m}) & \\ & & \ddots & & \\ & (-\frac{1}{m}) & & \ddots & \\ & & & & 1-\frac{1}{m} \end{bmatrix}. \tag{A.20}$$

Cette expression se réécrit cette fois-ci comme :

$$\sum_{j=1}^{n} \left\| M_i X_i(j,:)^\top B - Y_i(j,:)^\top B \right\|^2 = \left\| B^\top X_i M_i^\top - B^\top Y_i \right\|^2 = \left\| B X_i M_i^\top - B Y_i \right\|^2. \tag{A.21}$$

De la même façon que précédemment, $M_i^\top$ est solution aux moindres carrés du système :

$$\min_{\Omega \in \mathbb{R}^{3\times 3}} \left\| B X_i \ \Omega - B Y_i \right\|^2. \tag{A.22}$$

Or, la solution aux moindres carrés de ce système est :

$$\Omega = (B X_i)^+ B Y_i, \tag{A.23}$$

On peut alors réécrire :

$$\left\| B X_i M_i^\top - B Y_i \right\|^2 = \left\| (B X_i)(B X_i)^+ B Y_i - B Y_i \right\|^2 = \left\| ((B X_i)(B X_i)^+ - I) B Y_i \right\|^2. \tag{A.24}$$

On note cette fois-ci :

$$F = ((B X_i)(B X_i)^+ - I) B. \tag{A.25}$$

Et on obtient de la même façon que précédemment

$$\left\| \big((B X_i)(B X_i)^+ - I\big) B Y_i \right\|^2 = \| F \ Y_i \|^2 = \| F \ Y_{i,x} \|^2 + \| F \ Y_{i,y} \|^2 + \| F \ Y_{i,z} \|^2 \tag{A.26}$$

On construit alors Q_i :

$$Q_i = F^\top \ F = (((B X_i)(B X_i)^+ - I) B)^\top ((B X_i)(B X_i)^+ - I) B, \tag{A.27}$$

et Q de la même façon que précédemment (en assemblant les différents Q_i). L'espace des faibles déformations se trouve de la même façon que dans la méthode sans centrage : en étudiant les espaces propres de Q.

A.3 Optimisation numérique par Gauss-Newton

Dans cette section, nous présentons plus en détail le principe général de la résolution par des méthodes énergétiques du problème de recalage de données. Cela correspond ainsi aux méthodes de résolution utilisées dans les chapitres 3 et 4 de ce manuscrit. Nous discuterons ici le principe de l'algorithme de Gauss-Newton. Les autres algorithmes dérivés de la méthode de Newton tel que Levenberg-Marquardt [17] étant relativement similaires sur leur principe, nous ne nous y attarderons pas, le but de cette section étant de préparer la section suivante traitant de la façon dont on peut adapter ces algorithmes pour prendre en compte des espaces réduits pour les déformations. Le recalage est ainsi associé à une fonction d'objectif F_o, représentant l'état du recalage. Cette dernière est construite à partir des contraintes imposées sur le modèle biomécanique. Elle prend ainsi en entrée un état du modèle biomécanique (c'est-à-dire la position de tous les points du modèle $X = (x_0, y_0, z_0, \cdots, x_n, y_n, z_n))$ et donne en sortie une valeur dans $\mathbb{R}^+$ représentant le respect de ces contraintes pour la configuration considérée. Plus la valeur renvoyée $F_o(X)$ sera faible, plus les contraintes seront respectées. Résoudre le recalage revient alors à minimiser F_o sur X. F_o est cependant généralement non convexe, et il n'existe ainsi pas dans la majorité des cas de méthode permettant de résoudre cette minimisation de façon globale dû à la non-convexité du problème. Cette minimisation doit ainsi être réalisée par optimisation locale, après avoir initialisé la solution. Si la fonction est relativement lisse, cette optimisation peut être réalisée à l'aide de méthodes basées sur des descentes de gradient. De plus, cette fonction peut souvent être écrite comme une somme de moindres carrés. Sous cette forme, elle peut ainsi être minimisée à l'aide de méthodes de résolution de moindres carrés non linéaires telles que Gauss-Newton ou Levenberg-Marquardt [17]. L'intérêt de tels algorithmes est qu'ils convergent bien plus rapidement que de simples descentes de gradient. Soit ainsi F_o notre fonction à minimiser pour notre recalage, construite à partir de p résidus et associés à des fonctions $f_1, \ldots, f_p$ prenant elles aussi X en entrée et ayant des valeurs

dans $\mathbb{R}^+$. On définit alors F_o à partir d'une fonction F tel que $F_o(X) = \|F(X)\|^2$ avec :

$$F(X) = F(x_0, y_0, z_0, \ldots, x_n, y_n, z_n) = \begin{bmatrix} f_1(x_0, y_0, z_0 \cdots, x_n, y_n, z_n) \\ f_2(x_0, y_0, z_0 \cdots, x_n, y_n, z_n) \\ \vdots \\ f_p(x_0, y_0, z_0 \cdots, x_n, y_n, z_n) \end{bmatrix}. \tag{A.28}$$

La Jacobienne de cette fonction est alors

$$J_F = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial y_1} & \frac{\partial f_1}{\partial z_1} & \frac{\partial f_1}{\partial x_2} & \frac{\partial f_1}{\partial y_2} & \frac{\partial f_1}{\partial z_2} & \cdots & \frac{\partial f_1}{\partial x_n} & \frac{\partial f_1}{\partial y_n} & \frac{\partial f_1}{\partial z_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial y_1} & \frac{\partial f_2}{\partial z_1} & \frac{\partial f_2}{\partial x_2} & \frac{\partial f_2}{\partial y_2} & \frac{\partial f_2}{\partial z_2} & \cdots & \frac{\partial f_2}{\partial x_n} & \frac{\partial f_2}{\partial y_n} & \frac{\partial f_2}{\partial z_n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial y_1} & \frac{\partial f_n}{\partial z_1} & \frac{\partial f_n}{\partial x_2} & \frac{\partial f_n}{\partial y_2} & \frac{\partial f_n}{\partial z_2} & \cdots & \frac{\partial f_n}{\partial x_n} & \frac{\partial f_n}{\partial y_n} & \frac{\partial f_n}{\partial z_n} \end{bmatrix}. \tag{A.29}$$

Pour résoudre ce problème, on procède alors par itération. Soit β_k la position, à l'itération k des n points du modèle à recaler :

$$\beta_k = \begin{bmatrix} x_{1,k} \\ y_{1,k} \\ z_{1,k} \\ \vdots \\ x_{n,k} \\ y_{n,k} \\ z_{n,k} \end{bmatrix}. \tag{A.30}$$

À l'itération suivante, la position des points du modèle est alors définie par :

$$\beta_{k+1} = \beta_k + \delta_\beta, \tag{A.31}$$

avec :

$$\delta_\beta = -(J_F^\top J_F)^{-1} J_F F, \tag{A.32}$$

avec F contenant la valeur des résidus à minimiser à l'itération k, et J_F la Jacobienne associée. On arrête les itérations lorsque l'algorithme a convergé (c'est-à-dire lorsque $\|\delta_\beta\|^2$ ou $F(X_k)$ sont inférieur à un certain seuil) ou lorsqu'un certain nombre d'itéra-

tions a été atteint. La dernière position atteinte pour les points du modèle correspond alors à la solution du recalage.

A.4 Utilisation des espaces réduits dans le problème de minimisation

Une fois le sous-espace correspondant aux faibles déformations calculé à l'aide de l'une des deux méthodes décrites précédemment, il faut adapter la minimisation pour se limiter à ce sous-espace uniquement. On souhaite ainsi modifier légèrement l'algorithme décrit précédemment afin de prendre en compte l'espace réduit des déformations. On ne considère ainsi pas directement la Jacobienne associée au problème complet, mais une Jacobienne modifiée, de plus faible dimension. L'espace réduit sur lequel on souhaite restreindre correspond à un espace vectoriel de dimension q que l'on écrit

$$V = \mathrm{vec}(X_1, \cdots, X_q), \tag{A.33}$$

avec $\forall k \in [\![1, q]\!], X_k \in \mathbb{R}^{3n}$. On se propose alors de considérer l'application :

$$\begin{array}{rccc} \widetilde{F} & : & R^q & \rightarrow & R^p \\ & & \alpha_1, \ldots, \alpha_q & \mapsto & F\left(\sum\limits_{i=0}^{q} \alpha_i X_i\right) \end{array}. \tag{A.34}$$

Si l'on écrit :

$$\begin{array}{rccc} H & : & R^q & \rightarrow & R^{3n} \\ & & \alpha_1, \ldots, \alpha_q & \mapsto & \sum\limits_{i=0}^{q} \alpha_i X_i \end{array}, \tag{A.35}$$

on a alors $\widetilde{F} = F \circ H$. La Jacobienne de de $\widetilde{F}$ s'écrit alors :

$$J_{\widetilde{F}} = J_F J_H, \tag{A.36}$$

avec :

$$J_H = \left[\frac{\partial\left(\sum\limits_{i=0}^{q} \alpha_i X_i\right)}{\partial \alpha_1} \quad \frac{\partial\left(\sum\limits_{i=0}^{q} \alpha_i X_i\right)}{\partial \alpha_2} \quad \cdots \quad \frac{\partial\left(\sum\limits_{i=0}^{q} \alpha_i X_i\right)}{\partial \alpha_q} \right] = \begin{bmatrix} X_1 & X_2 & \cdots & X_q \end{bmatrix}. \tag{A.37}$$

La minimisation considérée pour le recalage est alors une minimisation de $\widetilde{F}$ sur les différents α_i :

$$\delta_\beta = \sum_{i=0}^{q} \alpha_i X_i, \tag{A.38}$$

et :

$$\gamma = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_q \end{bmatrix}. \tag{A.39}$$

On considère alors les variations de γ. À l'itération k, on cherche alors à définir δ_γ tel que :

$$\gamma_{k+1} = \gamma_k + \delta_\gamma, \tag{A.40}$$

avec cette fois-ci :

$$\delta_\gamma = -(J_{\widetilde{F}}^\top J_{\widetilde{F}})^{-1} J_{\widetilde{F}} \widetilde{F}. \tag{A.41}$$

Les variations de β peuvent alors directement être associées aux variations de γ. En effet, on a alors :

$$\delta_\beta = \delta_\gamma \begin{bmatrix} X_1 & X_2 & \cdots & X_q \end{bmatrix}. \tag{A.42}$$

Cela nous permet alors d'obtenir notre minimisation finale. Les dimensions de γ étant bien inférieures à celles de β, la minimisation utilisant cette méthode est ainsi beaucoup plus rapide. Elle permet, en outre, d'éviter d'obtenir des configurations non réalistes.

Annexe B

Détection d'erreurs de reconstruction par validation croisée stéréo

Cette annexe décrit une méthode de détection d'erreur dans les reconstructions de surface utilisant ce que l'on définit comme étant la validation croisée stéréo. Une tâche importante en chirurgie assistée par ordinateur est, comme présentée dans ce manuscrit, de reconstruire de façon automatique et le plus densément possible, la scène chirurgicale. Les approches les plus utilisées sont à ce jour les techniques basées sur des algorithmes de type SLAM ou SfM. Néanmoins, chacune de ces méthodes peut générer des erreurs dans les reconstructions. La capacité à automatiquement et rapidement détecter de telles erreurs est ainsi cruciale pour des applications robustes de chirurgie assistée par ordinateur. Nous décrivons ainsi, dans cette annexe, une solution permettant de détecter ces erreurs à l'aide d'une validation croisée stéréo. Le principe est adapté pour les reconstructions SfM ou SLAM, fonctionne avec un cœlioscope stéréo, et ne nécessite pas de capteurs additionnels ni d'imageries interventionnelles. Cette annexe est basée sur une contribution publiée à Surgetica 2017 [129].

B.1 Méthode

Les récentes implémentations de SfM comme Theia [183] prennent en entrée un ensemble d'images appelées images clés et retournent un modèle 3D de l'environnement ainsi qu'un ensemble de poses de caméra associées aux images clés. La reconstruction SfM échoue alors en présence d'erreurs significatives dans les poses de caméra et/ou dans le modèle 3D. On propose ainsi de détecter ces erreurs en s'inspirant de la *validation croisée* qui est une technique pour évaluer si les propriétés d'un modèle ayant été validées à l'aide d'une première méthode restent les mêmes lorsque l'on utilise une méthode différente pour générer à nouveau le même modèle. Dans notre cas, le modèle est une reconstruction SfM. Nous proposons ainsi de ne pas utiliser toutes les données pour réaliser la reconstruction mais d'en retenir une partie afin de réaliser cette validation croisée. Pour que les résultats de notre méthode soient fiables, il faut alors minimiser les corrélations entre données utilisées afin de générer la reconstruction et données retenues. La stratégie est ici d'utiliser un cœlioscope stéréo et de réaliser la reconstruction en utilisant uniquement les données provenant de l'une des deux images (gauche ou droite), et de valider ensuite la reconstruction en utilisant les images non utilisées. L'idée est ainsi de calculer pour chaque point de la reconstruction le ratio entre sa profondeur, une fois projeté dans chaque plan image correspondant aux images clés, et la profondeur calculée à l'aide d'un algorithme de reconstruction stéréo semi-dense [212]. Nous calculons ensuite la distribution $\mathcal{R}$ de ces ratios sur toutes les différentes images clés. Une reconstruction sans erreur générera alors une distribution unimodale de ces ratios avec une faible variance tandis qu'un reconstruction comportant des erreurs générera une distribution multimodale, ou une distribution uni-modale mais associée à une forte variance. Pour la méthode stéréo utilisée, un algorithme produisant une carte de profondeur éparse mais fiable est préférable afin d'éviter les faux positifs. À partir de cette distribution de ratio, nous pouvons 1) rejeter ou accepter la reconstruction, 2) calculer un score de confiance pour la reconstruction, 3) rejeter certaines régions comportant des erreurs et correspondant à des modes non principaux de la reconstruction et 4) indiquer les endroits où plus d'information est nécessaire afin de corriger la reconstruction (correspondant aux zones où des images clés ont étés rejetées). Nous nous concentrons ici sur l'option 1). On appelle $\mathcal{P}$ le nuage de points correspondant à la reconstruction, et on décrit alors l'algorithme suivant :

Algorithm 1 Validation SfM

function ESTCORRECT(*Images clés* $\mathcal{K}$, *Nuage de points* $\mathcal{P}$)
  $\mathcal{R} \leftarrow \{\}$
  for $t \in [1, T]$ **do**
    $\mathbf{D} \leftarrow$ carte de profondeur à partir de $\mathbf{K}_t \in \mathcal{K}$
    for $\boldsymbol{p}_j \in \mathcal{P}$ **do**
      Projeter $\boldsymbol{p}_j$ dans $\mathbf{D}$ et mesure la profondeur
      Calculer d en utilisant une interpolation bilinéaire
      if (d est définie) **then**
        Ajouter $\frac{\boldsymbol{p}_j}{d}$ à $\mathcal{R}$
  n $\leftarrow$ nombre de modes de $\mathcal{R}$
  $\sigma \leftarrow$ variance du mode principal
  return (n $==$ 1) $\wedge$ ($\sigma < \tau$)

Une valeur seuil τ est nécessaire et est fixée à 0.07 de façon empirique. Dans le futur, il serait préférable que τ soit appris à partir de données récoltées par la caméra utilisée.

B.2 Résultats expérimentaux

Afin de tester notre méthode, nous avons utilisé des données *ex vivo* porcines. Celles-ci correspondent à une vidéo d'exploration d'environ 2 minutes dans laquelle la surface de l'organe est observée par un cœlioscope stéréo. Nous avons ensuite utilisé l'algorithme de SfM proposé par Theia [183] sur ces données, afin d'obtenir une reconstruction, en ne considérant qu'un certain nombre des images clés, réparties dans la vidéo.

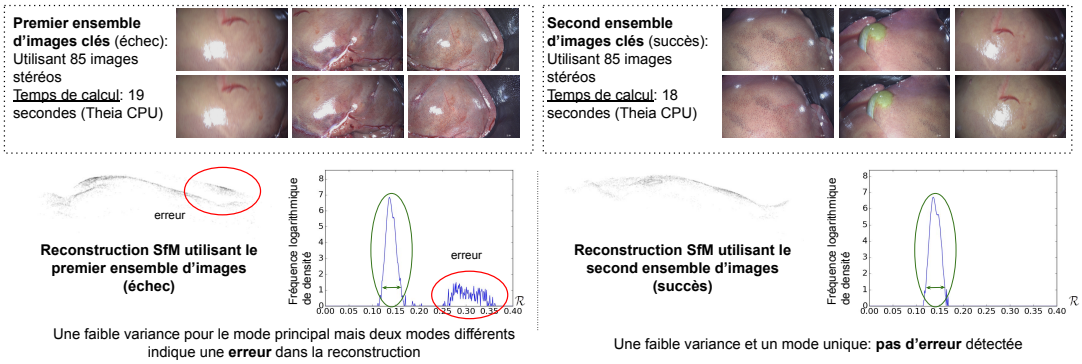

FIGURE B.1 – Méthode de validation croisée appliquée à un foie porcin *ex vivo* utilisant deux ensembles d'images différents.

Dans la Figure B.1, deux exemples, provenant de deux différentes reconstructions sont exposés. Dans chacun de ces exemples, 85 images sont utilisées. Pour la première reconstruction, il existe un mauvais alignement au niveau de l'un des lobes, ce qui peut être du à des erreurs dans les poses estimées du cœlioscope à cause de la texture fortement homogène du foie. Si non détecté, ce type d'erreurs peut causer des erreurs significatives pour l'application de chirurgie assistée considérée. Pour la seconde reconstruction, aucune erreur n'apparaît. Associé à ces deux reconstructions, nous affichons l'histogramme montrant les distributions des fréquences logarithmiques de densité de $\mathcal{R}$. Dans le cas où aucune erreur n'est présente, un seul mode de faible variance apparaît. Dans le cas, contraire, si des erreurs existent, un second mode apparaît. Cela indique que la reconstruction a mal estimé l'échelle du foie en certaines régions. L'étude de ces histogrammes permet ainsi de détecter la présence ou non d'erreurs dans les reconstructions de surface lorsque des données stéréo sont disponibles.

Bibliographie

[1] Yinoussa Adagolodjo, Raffaella Trivisonne, Nazim Haouchine, Stéphane Cotin, and Hadrien Courtecuisse. Silhouette-based pose estimation for deformable organs application to surgical augmented reality. In *Intelligent Robots and Systems (IROS)*, pages 539–544. IEEE, 2017.

[2] Sameer Agarwal, Keir Mierle, and Others. Ceres solver. http://ceres-solver.org.

[3] Pablo Fernández Alcantarilla, Adrien Bartoli, and Andrew J Davison. Kaze features. In *European Conference on Computer Vision*, pages 214–227. Springer, 2012.

[4] Max Allan, Ankur Kapoor, Philip Mewes, and Peter Mountney. Non rigid registration of 3d images to laparoscopic video for image guided surgery. In *Computer-Assisted and Robotic Endoscopy*, pages 109–116. Springer, 2015.

[5] Jérémie Allard, Stéphane Cotin, François Faure, Pierre-Jean Bensoussan, François Poyer, Christian Duriez, Hervé Delingette, and Laurent Grisoni. Sofa-an open source framework for medical simulation. In *MMVR 15-Medicine Meets Virtual Reality*, volume 125, pages 13–18. IOP Press, 2007.

[6] Pierre Alliez, Giuliana Ucelli, Craig Gotsman, and Marco Attene. Recent advances in remeshing of surfaces. In *Shape analysis and structuring*, pages 53–82. Springer, 2008.

[7] Brian Amberg, Sami Romdhani, and Thomas Vetter. Optimal step nonrigid icp algorithms for surface registration. In *CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[8] Danial Pour Arab, Sandrine Voros, and Caroline Essert. Dynamic path planning for percutaneous procedures in the abdomen during free breathing. *International Journal of Computer Assisted Radiology and Surgery*, 2020.

[9] Jordan Bano, Alexandre Hostettler, SA Nicolau, Stephane Cotin, Christophe Doignon, HS Wu, MH Huang, Luc Soler, and Jacques Marescaux. Simulation of pneumoperitoneum for laparoscopic surgery planning. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 91–98. Springer, 2012.

[10] Jordan Bano, Stéphane A Nicolau, Alexandre Hostettler, Christophe Doignon, Jacques Marescaux, and Luc Soler. Registration of preoperative liver model for laparoscopic surgery from intraoperative 3d acquisition. In *Augmented Reality Environments for Medical Imaging and Computer-Assisted Interventions*, pages 201–210. Springer, 2013.

[11] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf : Speeded up robust features. In *European conference on computer vision*, pages 404–417. Springer, 2006.

[12] Peter N Belhumeur, David J Kriegman, and Alan L Yuille. The bas-relief ambiguity. *International journal of computer vision*, 35(1) :33–44, 1999.

[13] Avi Ben-Cohen, Idit Diamant, Eyal Klang, Michal Amitai, and Hayit Greenspan. Fully convolutional network for liver segmentation and lesions detection. In *Deep learning and data labeling for medical applications*, pages 77–85. Springer, 2016.

[14] Sylvain Bernhardt, Stéphane A Nicolau, Adrien Bartoli, Vincent Agnus, Luc Soler, and Christophe Doignon. Using shading to register an intraoperative ct scan to a laparoscopic image. In *CARE*, pages 59–68. Springer, 2015.

[15] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV : control paradigms and data structures*, volume 1611, pages 586–606. International Society for Optics and Photonics, 1992.

[16] Stan Birchfield and Carlo Tomasi. A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(4) :401–406, 1998.

[17] Åke Björck. *Numerical methods for least squares problems*. SIAM, 1996.

[18] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999.

[19] Jean-Daniel Boissonnat and Bernhard Geiger. Three-dimensional reconstruction of complex shapes based on the delaunay triangulation. In *Biomedical Image Processing and Biomedical Visualization*, volume 1905, pages 964–975. International Society for Optics and Photonics, 1993.

[20] Nicolas Bourdel, Toby Collins, Daniel Pizarro, Clement Debize, Anne-sophie Grémeau, Adrien Bartoli, and Michel Canis. Use of augmented reality in laparoscopic gynecology to visualize myomas. *Fertility and sterility*, 107(3) :737–739, 2017.

[21] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[22] Wendy MH Busser, Yvonne L Hoogeveen, Rene PH Veth, HW Bart Schreuder, Angelique Balguid, W KlaasJan Renema, and Leo J SchultzeKool. Percutaneous radiofrequency ablation of osteoid osteomas with use of real-time needle guidance for accurate needle placement : a pilot study. *Cardiovascular and interventional radiology*, 34(1) :180–183, 2011.

[23] Michael Calonder, Vincent Lepetit, Mustafa Ozuysal, Tomasz Trzcinski, Christoph Strecha, and Pascal Fua. Brief : Computing a local binary descriptor very fast. *IEEE transactions on pattern analysis and machine intelligence*, 34(7) :1281–1298, 2011.

[24] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. Openpose : Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[25] Grégory Chagnon, Marie Rebouah, and Denis Favier. Hyperelastic energy densities for soft biological tissues : a review. *Journal of Elasticity*, 120(2) :129–160, 2015.

[26] Jia-Ren Chang and Yong-Sheng Chen. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5418, 2018.

[27] Pauline Chauvet, Toby Collins, Clement Debize, Lorraine Novais-Gameiro, Bruno Pereira, Adrien Bartoli, Michel Canis, and Nicolas Bourdel. Augmented reality in a tumor resection model. *Surgical endoscopy*, 32(3) :1192–1201, 2018.

[28] X Chen, L Xu, Y Wang, H Wang, F Wang, X Zeng, Q Wang, and J Egger. Development of a surgical navigation system based on augmented reality using an optical see-through head-mounted display. *J Biomed Inform*, 2015.

[29] Cheekong Chui, E Kobayashi, X Chen, T Hisada, and I Sakuma. Combined compression and elongation experiments and non-linear modelling of liver tissue for surgical simulation. *Medical and Biological Engineering and Computing*, 42(6) :787–798, 2004.

[30] JH Chung, V Rajagopal, PMF Nielsen, and MP Nash. A biomechanical model of mammographic compressions. *Biomechanics and modeling in mechanobiology*, 7(1) :43–52, 2008.

[31] Logan W Clements, William C Chapman, Benoit M Dawant, Robert L Galloway Jr, and Michael I Miga. Robust surface registration using salient anatomical features for image-guided liver surgery : Algorithm and validation. *Medical physics*, 35(6Part1) :2528–2540, 2008.

[32] Jarrod A Collins, Jared A Weis, Jon S Heiselman, Logan W Clements, Amber L Simpson, William R Jarnagin, and Michael I Miga. Improving registration robustness for image-guided liver surgery in a novel human-to-phantom data framework. *IEEE transactions on medical imaging*, 36(7) :1502–1510, 2017.

[33] Toby Collins and Adrien Bartoli. 3d reconstruction in laparoscopy with close-range photometric stereo. In *MICCAI*, pages 634–642. Springer, 2012.

[34] Toby Collins and Adrien Bartoli. Towards live monocular 3d laparoscopy using shading and specularity information. In *IPCAI*, 2012.

[35] Toby Collins and Adrien Bartoli. [poster] realtime shape-from-template : System and applications. In *ISMAR, 2015 IEEE International Symposium on*, pages 116–119. IEEE, 2015.

[36] Toby Collins, Adrien Bartoli, Nicolas Bourdel, and Michel Canis. Robust, real-time, dense and deformable 3d organ tracking in laparoscopic videos. In *MICCAI*, pages 404–412. Springer, 2016.

[37] Toby Collins, Pauline Chauvet, Clément Debize, Daniel Pizarro, Adrien Bartoli, Michel Canis, and Nicolas Bourdel. A system for augmented reality guided laparoscopic tumour resection with quantitative ex-vivo user evaluation. In *Interna-*

tional Workshop on Computer-Assisted and Robotic Endoscopy, pages 114–126. Springer, 2016.

[38] Toby Collins, Daniel Pizarro, Adrien Bartoli, Michel Canis, and Nicolas Bourdel. Realtime wide-baseline registration of the uterus in laparoscopic videos using multiple texture maps. In *Augmented reality environments for medical imaging and computer-assisted interventions*, pages 162–171. Springer, 2013.

[39] Timothee Cour, Praveen Srinivasan, and Jianbo Shi. Balanced graph matching. In *Advances in Neural Information Processing Systems*, pages 313–320, 2007.

[40] Yan Cui, Li-Yan Zhou, Man-Ku Dong, Ping Wang, Min Ji, Xiao-Ou Li, Chang-Wei Chen, Zi-Pei Liu, Yong-Jie Xu, and Hong-Wen Zhang. Ultrasonography guided percutaneous radiofrequency ablation for hepatic cavernous hemangioma. *World Journal of Gastroenterology*, 9(9) :2132, 2003.

[41] Zhaopeng Cui and Ping Tan. Global structure-from-motion by similarity averaging. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 864–872, 2015.

[42] Andrew J Davison. Real-time simultaneous localisation and mapping with a single camera. In *null*, page 1403. IEEE, 2003.

[43] Martin de La Gorce, Nikos Paragios, and David J Fleet. Model-based hand tracking with texture, shading and self-occlusions. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[44] Joseph DeGol, Timothy Bretl, and Derek Hoiem. Improved structure from motion using fiducial marker matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 273–288, 2018.

[45] Christophe Dehais, Géraldine Morin, and Vincent Charvillat. From rendering to tracking point-based 3d models. *Image and Vision Computing*, 28(9) :1386–1395, 2010.

[46] Amaël Delaunoy and Emmanuel Prados. Gradient flows for optimizing triangular mesh-based surfaces : Applications to 3d reconstruction problems dealing with visibility. *International journal of computer vision*, 95(2) :100–123, 2011.

[47] Yue Deng, Yebin Liu, Qionghai Dai, Zengke Zhang, and Yao Wang. Noisy depth maps fusion for multiview stereo via matrix completion. *IEEE Journal of Selected Topics in Signal Processing*, 6(5) :566–582, 2012.

[48] J Denis Durou. Shape from shading–eclairages, réflexions et perspectives. *Rapport technique*, 2007.

[49] David Eberly. Intersection of convex objects : The method of separating axes. *Geometric Tools, LLC http ://www. geometrictools. com,(1998- 2008)*, 2001.

[50] Jakob Engel, Thomas Schöps, and Daniel Cremers. Lsd-slam : Large-scale direct monocular slam. In *European conference on computer vision*, pages 834–849. Springer, 2014.

[51] Carlos Hernández Esteban and Francis Schmitt. Silhouette and stereo fusion for 3d object modeling. *Computer Vision and Image Understanding*, 96(3) :367–392, 2004.

[52] Sylwester Fabian and Dominik Spinczyk. Target registration error minimization for minimally invasive interventions involving deformable organs. *Computerized Medical Imaging and Graphics*, 65 :4–10, 2018.

[53] J Michael Fitzpatrick. Fiducial registration error and target registration error are uncorrelated. In *Medical Imaging 2009 : Visualization, Image-Guided Procedures, and Modeling*, volume 7261, page 726102. International Society for Optics and Photonics, 2009.

[54] Simon Fraser University. Centre for Systems Science and Mark S Drew. *Shape from color*. Simon Fraser University, Centre for Systems Science, 1992.

[55] CH Quartucci Forster and Clesio L Tozzi. Towards 3d reconstruction of endoscope images using shape from shading. In *Proceedings 13th Brazilian Symposium on Computer Graphics and Image Processing (Cat. No. PR00878)*, pages 90–96. IEEE, 2000.

[56] Tom François, Lilian Calvet, S Madad, Damien Saboul, Simone Gasparini, Prasad Samarakoon, Nicolas Bourdel, and Adrien Bartoli. Detecting the occluding contours of the uterus to automatise augmented laparoscopy : score, loss, dataset, evaluation and user study. *International journal of computer assisted radiology and surgery*, 2020.

[57] Pascal Fua and Yvan G Leclerc. Object-centered surface reconstruction : Combining multi-image stereo and shading. *International Journal of Computer Vision*, 16(1) :35–56, 1995.

[58] Mathias Gallardo, Toby Collins, and Adrien Bartoli. Using shading and a 3d template to reconstruct complex surface deformations. In *BMVC*, 2016.

[59] Mathias Gallardo, Toby Collins, and Adrien Bartoli. Dense non-rigid structure-from-motion and shading with unknown albedos. In *ICCV*, 2017.

[60] Dorian Gálvez-López and Juan D Tardos. Bags of binary words for fast place recognition in image sequences. *IEEE Transactions on Robotics*, 28(5) :1188–1197, 2012.

[61] P Ganesan, V Rajini, and R Immanuvel Rajkumar. Segmentation and edge detection of color images using cielab color space and edge detectors. In *INTERACT-2010*, pages 393–397. IEEE, 2010.

[62] Pau Gargallo and Peter Sturm. Bayesian 3d modeling from images using multiple depth maps. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 885–891. IEEE, 2005.

[63] Sergio Garrido-Jurado, Rafael Muñoz-Salinas, Francisco José Madrid-Cuevas, and Manuel Jesús Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6) :2280–2292, 2014.

[64] Michael Goesele, Brian Curless, and Steven M Seitz. Multi-view stereo revisited. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 2402–2409. IEEE, 2006.

[65] Stefan Gottschalk, Ming C Lin, and Dinesh Manocha. Obbtree : A hierarchical structure for rapid interference detection. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 171–180, 1996.

[66] Sanjay Gupta and David C Madoff. Image-guided percutaneous needle biopsy in cancer diagnosis and staging. *Techniques in vascular and interventional radiology*, 10(2) :88–101, 2007.

[67] Charlie A Hamm, Clinton J Wang, Lynn J Savic, Marc Ferrante, Isabel Schobert, Todd Schlachter, MingDe Lin, James S Duncan, Jeffrey C Weinreb, Julius Chapiro, et al. Deep learning for liver tumor diagnosis part i : development of a convolutional neural network classifier for multi-phasic mri. *European radiology*, 29(7) :3338–3347, 2019.

[68] Lianghao Han, J Alison Noble, and Michael Burcher. A novel ultrasound indentation system for measuring biomechanical properties of in vivo soft tissue. *Ultrasound in medicine & biology*, 29(6) :813–823, 2003.

[69] Nazim Haouchine, Jeremie Dequidt, Igor Peterlik, Erwan Kerrien, Marie-Odile Berger, and Stéphane Cotin. Image-guided simulation of heterogeneous tissue deformation for augmented reality during hepatic surgery. In *2013 IEEE international symposium on mixed and augmented reality (ISMAR)*, pages 199–208. IEEE, 2013.

[70] Nazim Haouchine, Frederick Roy, Lionel Untereiner, and Stéphane Cotin. Using contours as boundary conditions for elastic registration during minimally invasive hepatic surgery. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016.

[71] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 International Conference on Computer Vision*, pages 991–998. IEEE, 2011.

[72] Richard I Hartley. Theory and practice of projective rectification. *International Journal of Computer Vision*, 35(2) :115–127, 1999.

[73] Jon S Heiselman, Jarrod A Collins, Logan W Clements, Jared A Weis, Amber L Simpson, Sunil K Geevarghese, T Peter Kingham, William R Jarnagin, and Michael I Miga. Nonrigid registration for laparoscopic liver surgery using sparse intraoperative data. In *Medical Imaging 2018 : Image-Guided Procedures, Robotic Interventions, and Modeling*, volume 10576, page 105760D. International Society for Optics and Photonics, 2018.

[74] Christoph Hennersperger, Bernhard Fuerst, Salvatore Virga, Oliver Zettinig, Benjamin Frisch, Thomas Neff, and Nassir Navab. Towards mri-based autonomous robotic us acquisitions : a first feasibility study. *IEEE transactions on medical imaging*, 36(2) :538–548, 2016.

[75] Vu Hoang Hiep, Renaud Keriven, Patrick Labatut, and Jean-Philippe Pons. Towards high-resolution large-scale multi-view stereo. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1430–1437. IEEE, 2009.

[76] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2) :328–341, 2007.

[77] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on pattern analysis and machine intelligence*, 30(2) :328–341, 2008.

[78] Berthold KP Horn. Shape from shading : A method for obtaining the shape of a smooth opaque object from one view. 1970.

[79] Berthold KP Horn. Closed-form solution of absolute orientation using unit quaternions. *JOSA A*, 4(4) :629–642, 1987.

[80] Alexandre Hostettler, SA Nicolau, Y Rémond, Jacques Marescaux, and Luc Soler. A real-time predictive simulation of abdominal viscera positions during quiet free breathing. *Progress in biophysics and molecular biology*, 103(2-3) :169–184, 2010.

[81] Peijun Hu, Fa Wu, Jialin Peng, Ping Liang, and Dexing Kong. Automatic 3d liver segmentation based on deep learning and globally optimized surface evolution. *Physics in Medicine & Biology*, 61(24) :8676, 2016.

[82] Peter J Huber et al. Robust regression : asymptotics, conjectures and monte carlo. *The annals of statistics*, 1(5) :799–821, 1973.

[83] John Isidoro and Stan Sclaroff. Stochastic refinement of the visual hull to satisfy photometric and silhouette consistency constraints. In *null*, page 1335. IEEE, 2003.

[84] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al. Kinectfusion : real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568. ACM, 2011.

[85] Pablo Jiménez, Federico Thomas, and Carme Torras. 3d collision detection : a survey. *Computers & Graphics*, 25(2) :269–285, 2001.

[86] Hailin Jin, Stefano Soatto, and Anthony J Yezzi. Multi-view stereo beyond lambert. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 1, pages I–I. IEEE, 2003.

[87] Hailin Jin, Stefano Soatto, and Anthony J Yezzi. Multi-view stereo reconstruction of dense shape and complex appearance. *International Journal of Computer Vision*, 63(3) :175–189, 2005.

[88] Perry B. Johnson, David Borrego, Stephen Balter, Kevin Johnson, Daniel Siragusa, and Wesley E. Bolch. Skin dose mapping for fluoroscopically guided interventions. *Medical Physics*, 2011.

[89] Sing Bing Kang and Richard Szeliski. Extracting view-dependent depth maps from a collection of images. *International Journal of Computer Vision*, 58(2) :139–163, 2004.

[90] Niklas Karlsson, Enrico Di Bernardo, Jim Ostrowski, Luis Goncalves, Paolo Pirjanian, and Mario E Munich. The vslam algorithm for robust localization and mapping. In *Proceedings of the 2005 IEEE international conference on robotics and automation*, pages 24–29. IEEE, 2005.

[91] Hannes G Kenngott, Martin Wagner, Matthias Gondan, Felix Nickel, Marco Nolden, Andreas Fetzer, Jürgen Weitz, Lars Fischer, Stefanie Speidel, Hans-Peter Meinzer, et al. Real-time image guidance in laparoscopic liver surgery : first clinical experience with a guidance system based on intraoperative ct imaging. *Surgical endoscopy*, 28(3) :933–940, 2014.

[92] Amy E Kerdok, Stephane M Cotin, Mark P Ottensmeyer, Anna M Galea, Robert D Howe, and Steven L Dawson. Truth cube : Establishing physical standards for soft tissue simulation. *Medical Image Analysis*, 7(3) :283–291, 2003.

[93] Joachim Kettenbach and Gernot Kronreif. Robotic systems for percutaneous needle-guided interventions. *Minimally Invasive Therapy & Allied Technologies*, 24(1) :45–53, 2015.

[94] Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *2007 6th IEEE and ACM international symposium on mixed and augmented reality*, pages 225–234. IEEE, 2007.

[95] James T Klosowski, Martin Held, Joseph SB Mitchell, Henry Sowizral, and Karel Zikan. Efficient collision detection using bounding volume hierarchies of k-dops. *IEEE TVCG*, pages 21–36, 1998.

[96] Timor Knudsen. Calibration system and method for multi-camera vision system, June 27 2017. US Patent 9,688,200.

[97] Vladimir Kolmogorov and Ramin Zabih. Multi-camera scene reconstruction via graph cuts. In *European conference on computer vision*, pages 82–96. Springer, 2002.

[98] Bongjin Koo, Erol Özgür, Bertrand Le Roy, Emmanuel Buc, and Adrien Bartoli. Deformable registration of a preoperative 3d liver volume to a laparoscopy image using contour and shading cues. In *MICCAI*, pages 326–334. Springer, 2017.

[99] Dennis Krys and Homayoun Najjaran. Development of visual simultaneous localization and mapping (vslam) for a pipe inspection robot. In *2007 International Symposium on Computational Intelligence in Robotics and Automation*, pages 344–349. IEEE, 2007.

[100] Zuzana Kukelova, Jan Heller, and Andrew Fitzgibbon. Efficient intersection of three quadrics and applications in computer vision. In *CVPR*, 2016.

[101] Kiriakos N Kutulakos and Steven M Seitz. A theory of shape by space carving. *International journal of computer vision*, 38(3) :199–218, 2000.

[102] Edwin H Land and John J McCann. Lightness and retinex theory. *Josa*, 61(1) :1–11, 1971.

[103] Thomas Lange, Nils Papenberg, Stefan Heldmann, Jan Modersitzki, Bernd Fischer, Hans Lamecker, and Peter M Schlag. 3d ultrasound-ct registration of the liver using combined landmark-intensity information. *International journal of computer assisted radiology and surgery*, 4(1) :79–88, 2009.

[104] Fabian Langguth, Kalyan Sunkavalli, Sunil Hadap, and Michael Goesele. Shading-aware multi-view stereo. In *European Conference on Computer Vision*, pages 469–485. Springer, 2016.

[105] Duhgoon Lee, Woo Hyun Nam, Jae Young Lee, and Jong Beom Ra. Non-rigid registration between 3d ultrasound and ct images of the liver based on intensity and gradient information. *Physics in Medicine & Biology*, 56(1) :117, 2010.

[106] Jiann Der Lee, Chung Hsien Huang, Tsing Che Huang, Hui Yuan Hsieh, and Shin Tseng Lee. Medical augmented reality using a markerless registration framework. *Expert Systems with Applications*, 2012.

[107] Xiaomeng Li, Hao Chen, Xiaojuan Qi, Qi Dou, Chi-Wing Fu, and Pheng-Ann Heng. H-denseunet : hybrid densely connected unet for liver and tumor segmen-

tation from ct volumes. *IEEE transactions on medical imaging*, 37(12) :2663–2674, 2018.

[108] Miao Liao, Liang Wang, Ruigang Yang, and Minglun Gong. Light fall-off stereo. In *CVPR*, 2007.

[109] Qi Liu-Yin, Rui Yu, Lourdes Agapito, Andrew Fitzgibbon, and Chris Russell. Better together : Joint reasoning for non-rigid 3d reconstruction with specularities and shading. *arXiv preprint arXiv :1708.01654*, 2017.

[110] H Christopher Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(5828) :133–135, 1981.

[111] Charles T Loop and Tony D DeRose. A multisided generalization of bézier surfaces. *ACM Transactions on Graphics (TOG)*, 8(3) :204–234, 1989.

[112] William E Lorensen and Harvey E Cline. Marching cubes : A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4) :163–169, 1987.

[113] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.

[114] Wenjie Luo, Alexander G Schwing, and Raquel Urtasun. Efficient deep learning for stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5695–5703, 2016.

[115] Quang-Tuan Luong. *Matrice fondamentale et autocalibration en vision par ordinateur*. PhD thesis, Paris 11, 1992.

[116] Márcio C F Macedo, Antônio L Apolinário Jr, Antonio C S Souza, and Gilson A Giraldi. High-Quality On-Patient Medical Data Visualization in a Markerless Augmented Reality Environment. *Sbc*, 2014.

[117] Nader Mahmoud, Iñigo Cirauqui, Alexandre Hostettler, Christophe Doignon, Luc Soler, Jacques Marescaux, and JMM Montiel. Orbslam-based endoscope tracking and 3d reconstruction. In *International workshop on computer-assisted and robotic endoscopy*, pages 72–83. Springer, 2016.

[118] Nader Mahmoud, Toby Collins, Alexandre Hostettler, Luc Soler, Christophe Doignon, and Jose Maria Martinez Montiel. Live tracking and dense reconstruc-

tion for handheld monocular endoscopy. *IEEE transactions on medical imaging*, 38(1) :79–89, 2018.

[119] Nader Mahmoud, Óscar G Grasa, Stéphane A Nicolau, Christophe Doignon, Luc Soler, Jacques Marescaux, and J. M.M. Montiel. On-patient see-through augmented reality based on visual SLAM. *International Journal of Computer Assisted Radiology and Surgery*, 12(1) :1–11, 2017.

[120] Lena Maier-Hein, Peter Mountney, Adrien Bartoli, Haytham Elhawary, D Elson, Anja Groch, Andreas Kolb, Marcos Rodrigues, J Sorger, Stefanie Speidel, et al. Optical techniques for 3d surface reconstruction in computer-assisted laparoscopic surgery. *Medical image analysis*, 17(8) :974–996, 2013.

[121] Jitendra Malik, Serge Belongie, Thomas Leung, and Jianbo Shi. Contour and texture analysis for image segmentation. *International journal of computer vision*, 43(1) :7–27, 2001.

[122] Ezio Malis and Manuel Vargas. Deeper understanding of the homography decomposition for vision-based control. 2007.

[123] Abed Malti and Adrien Bartoli. Combining conformal deformation and cook–torrance shading for 3-d reconstruction in laparoscopy. *IEEE Transactions on Biomedical Engineering*, 61(6) :1684–1692, 2014.

[124] Stéphanie Marchesseau, Simon Chatelin, and Hervé Delingette. Nonlinear biomechanical model of the liver. In *Biomechanics of living organs*, pages 243–265. Elsevier, 2017.

[125] J Marescaux, M Diana, and L Soler. Augmented reality and minimally invasive surgery. *Journal of Gastroenterology and Hepatology Research*, 2(5) :555–560, 2013.

[126] Damien Mariyanayagam, Pierre Gurdjos, Sylvie Chambon, Florent Brunet, and Vincent Charvillat. Pose estimation of a single circle using default intrinsic calibration. In *ACCV*, 2018.

[127] David R Martin, Charless C Fowlkes, and Jitendra Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE transactions on pattern analysis and machine intelligence*, 26(5) :530–549, 2004.

[128] Richard Modrzejewski, Toby Collins, Adrien Bartoli, Alexandre Hostettler, and Jacques Marescaux. Soft-body registration of pre-operative 3d models to intra-operative rgbd partial body scans. In *MICCAI*, pages 39–46. Springer, 2018.

[129] Richard MODRZEJEWSKI, Toby COLLINS, Adrien BARTOLI, Alexandre HOSTETTLER, Luc SOLER, and Jacques MARESCAUX. Automatic verification of laparoscopic 3d reconstructions with stereo cross-validation. *Surgetica*, 2017.

[130] Richard Modrzejewski, Toby Collins, Alexandre Hostettler, Jacques Marescaux, and Adrien Bartoli. Light modelling and calibration in laparoscopy. *International journal of computer assisted radiology and surgery*, 2020.

[131] Richard Modrzejewski, Toby Collins, Barbara Seeliger, Adrien Bartoli, Alexandre Hostettler, and Jacques Marescaux. An in vivo porcine dataset and evaluation methodology to measure soft-body laparoscopic liver registration accuracy with an extended algorithm that handles collisions. *International journal of computer assisted radiology and surgery*, pages 1–9, 2019.

[132] Wouter Mollemans, Filip Schutyser, Johan Van Cleynenbreugel, and Paul Suetens. Tetrahedral mass spring model for fast soft tissue deformation. In *International Symposium on Surgery Simulation and Soft Tissue Modeling*, pages 145–154. Springer, 2003.

[133] Peter Mountney and Guang-Zhong Yang. Motion compensated slam for image guided surgery. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 496–504. Springer, 2010.

[134] Andrews Munjiza and KRF Andrews. Nbs contact detection algorithm for bodies of similar size. *International Journal for Numerical Methods in Engineering*, 43(1) :131–149, 1998.

[135] Raul Mur-Artal and Juan D Tardós. Orb-slam2 : An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33(5) :1255–1262, 2017.

[136] Matthieu Nesme, Yohan Payan, and François Faure. Efficient, physically plausible finite elements. 2005.

[137] Fabrice Neyret, Raphael Heiss, and Franck Sénégas. Realistic rendering of an organ surface in real-time for laparoscopic surgery simulation. *The Visual Computer*, 18(3) :135–149, 2002.

[138] Stéphane Nicolau, Luc Soler, Didier Mutter, and Jacques Marescaux. Augmented reality in laparoscopic surgical oncology. *Surgical oncology*, 20(3) :189–201, 2011.

[139] Raymond W Ogden. *Non-linear elastic deformations*. Courier Corporation, 1997.

[140] Takayuki Okatani and Koichiro Deguchi. Shape reconstruction from an endoscope image by shape from shading technique for a point light source at the projection center. *Computer vision and image understanding*, 66(2) :119–131, 1997.

[141] Michael Oren and Shree K Nayar. Generalization of lambert's reflectance model. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 239–246, 1994.

[142] Jonas Östlund, Aydin Varol, Dat Tien Ngo, and Pascal Fua. Laplacian meshes for monocular 3d shape recovery. In *European conference on computer vision*, pages 412–425. Springer, 2012.

[143] Vicente Parot, Daryl Lim, Germán González, Giovanni Traverso, Norman S Nishioka, Benjamin J Vakoc, and Nicholas J Durr. Photometric stereo endoscopy. *Journal of biomedical optics*, 18(7) :076017, 2013.

[144] Yohan Payan. Soft tissue finite element modeling and calibration of the material properties in the context of computer-assisted medical interventions. In *Material Parameter Identification and Inverse Problems in Soft Tissue Biomechanics*, pages 133–144. Springer, 2017.

[145] Alex Pentland. Shape information from shading : a theory about human perception. *Spatial vision*, 4(2-3) :165–182, 1989.

[146] Alex Pentland and Stan Sclaroff. Closed-form solutions for physically based shape modeling and recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (7) :715–729, 1991.

[147] Patrick Pessaux, Michele Diana, Luc Soler, Tullio Piardi, Didier Mutter, and Jacques Marescaux. Towards cybernetic surgery : robotic and augmented reality-assisted liver segmentectomy. *Langenbeck's archives of surgery*, 400(3) :381–385, 2015.

[148] Bui Tuong Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6) :311–317, 1975.

[149] Steve Pieper, Michael Halle, and Ron Kikinis. 3d slicer. In *2004 2nd IEEE international symposium on biomedical imaging : nano to macro (IEEE Cat No. 04EX821)*, pages 632–635. IEEE, 2004.

[150] Rosalie Plantefeve, Igor Peterlik, Nazim Haouchine, and Stéphane Cotin. Patient-specific biomechanical modeling for guidance during minimally-invasive hepatic surgery. *Annals of biomedical engineering*, 44(1) :139–153, 2016.

[151] J-P Pons, Renaud Keriven, and Olivier Faugeras. Modelling dynamic scenes by registering multi-view image sequences. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 822–827. IEEE, 2005.

[152] Roi Poranne, Craig Gotsman, and Daniel Keren. 3d surface reconstruction using a generalized distance function. In *Computer Graphics Forum*, volume 29, pages 2479–2491. Wiley Online Library, 2010.

[153] Kristina Prokopetc, Toby Collins, and Adrien Bartoli. Automatic detection of the uterus and fallopian tube junctions in laparoscopic images. In *International conference on information processing in medical imaging*, pages 552–563. Springer, 2015.

[154] Kun Qian, Junxuan Bai, Xiaosong Yang, Junjun Pan, and Jianjun Zhang. Virtual reality based laparoscopic surgery simulation. In *Proceedings of the 21st ACM Symposium on Virtual Reality Software and Technology*, pages 69–78, 2015.

[155] Yvain Quéau, Tao Wu, and Daniel Cremers. Semi-calibrated near-light photometric stereo. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 656–668. Springer, 2017.

[156] Douglas A Reynolds. Gaussian mixture models. *Encyclopedia of biometrics*, 741, 2009.

[157] Nicolas Loy Rodas, Julien Bert, Dimitris Visvikis, Michel De Mathelin, and Nicolas Padoy. Pose optimization of a C-arm imaging device to reduce intraoperative radiation exposure of staff and patient during interventional procedures. In *Proceedings - IEEE International Conference on Robotics and Automation*, 2017.

[158] Torsten Rohlfing, Calvin R Maurer, David A Bluemke, and Michael A Jacobs. Volume-preserving nonrigid registration of mr breast images using free-form deformation with an incompressibility constraint. *IEEE transactions on medical imaging*, 22(6) :730–741, 2003.

[159] Antoine Rosset, Luca Spadola, and Osman Ratib. Osirix : an open-source software for navigating in multidimensional dicom images. *Journal of digital imaging*, 17(3) :205–216, 2004.

[160] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *European conference on computer vision*, pages 430–443. Springer, 2006.

[161] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. " grabcut" interactive foreground extraction using iterated graph cuts. *ACM transactions on graphics (TOG)*, 23(3) :309–314, 2004.

[162] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb : An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee, 2011.

[163] Nadine Abu Rumman, Marco Schaerf, and Dominique Bechmann. Collision detection for articulated deformable characters. In *Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games*, pages 215–220. ACM, 2015.

[164] Ichiro Sakuma, Yosuke Nishimura, Chee Kong Chui, Etsuko Kobayashi, Hiroshi Inada, Xian Chen, and Toshiaki Hisada. In vitro measurement of mechanical properties of liver tissue under compression and elongation using a new test piece holding method with surgical glue. In *International Symposium on Surgery Simulation and Soft Tissue Modeling*, pages 284–292. Springer, 2003.

[165] Mathieu Salzmann and Pascal Fua. Linear local models for monocular reconstruction of deformable surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5) :931–944, 2010.

[166] Erika Sandford, Yi Chen, Ian Hunter, Greg Hillebrand, and Lynette Jones. Capturing skin properties from dynamic mechanical analyses. *Skin Research and Technology*, 19(1) :e339–e348, 2013.

[167] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4104–4113, 2016.

[168] Stephen Se, David Lowe, and Jim Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *The international Journal of robotics Research*, 21(8) :735–758, 2002.

[169] Alexander Seitel, Markus Engel, Christof M Sommer, Boris A Radeleff, Caroline Essert-Villard, Claire Baegert, Markus Fangerau, Klaus H Fritzsche, Kwong Yung, Hans-Peter Meinzer, et al. Computer-assisted trajectory planning for percutaneous needle insertions. *Medical physics*, 38(6Part1) :3246–3259, 2011.

[170] Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, volume 1, pages 519–528. IEEE, 2006.

[171] Steven M Seitz and Charles R Dyer. Photorealistic scene reconstruction by voxel coloring. *International Journal of Computer Vision*, 35(2) :151–173, 1999.

[172] Vera Skorkovská, Ivana Kolingerová, and Bedrich Benes. A simple and robust approach to computation of meshes intersection. In *VISIGRAPP (1 : GRAPP)*, pages 175–182, 2018.

[173] Gregory G Slabaugh, W Bruce Culbertson, Thomas Malzbender, Mark R Stevens, and Ronald W Schafer. Methods for volumetric reconstruction of visual scenes. *International Journal of Computer Vision*, 57(3) :179–199, 2004.

[174] Randall Smith, Matthew Self, and Peter Cheeseman. Estimating uncertain spatial relationships in robotics. In *Autonomous robot vehicles*, pages 167–193. Springer, 1990.

[175] Liviu Solomon. *Elasticité linéaire*. Masson et Cie Editeurs, 1968.

[176] Danail Stoyanov, George P Mylonas, Fani Deligianni, Ara Darzi, and Guang Zhong Yang. Soft-tissue motion tracking and structure estimation for robotic assisted mis procedures. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 139–146. Springer, 2005.

[177] Peter Sturm and Bill Triggs. A factorization based algorithm for multi-image projective structure and motion. In *European conference on computer vision*, pages 709–720. Springer, 1996.

[178] Deyu Sun, Jiquan Liu, Cristian A Linte, Huilong Duan, and Richard A Robb. Surface reconstruction from tracked endoscopic video using the structure from motion approach. In *Augmented Reality Environments for Medical Imaging and Computer-Assisted Interventions*, pages 127–135. Springer, 2013.

[179] Wei Sun and Michael S Sacks. Finite element implementation of a generalized fung-elastic constitutive model for planar soft tissues. *Biomechanics and modeling in mechanobiology*, 4(2-3) :190–199, 2005.

[180] Niko Sünderhauf and Peter Protzel. Towards a robust back-end for pose graph slam. In *2012 IEEE international conference on robotics and automation*, pages 1254–1261. IEEE, 2012.

[181] Satoshi Suzuki et al. Topological structural analysis of digitized binary images by border following. *Computer vision, graphics, and image processing*, 30(1) :32–46, 1985.

[182] Chris Sweeney. Theia multiview geometry library : Tutorial & reference. http://theia-sfm.org.

[183] Christopher Sweeney, Tobias Hollerer, and Matthew Turk. Theia : A fast and scalable structure-from-motion library. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 693–696. ACM, 2015.

[184] Richard Szeliski. A multi-view approach to motion and stereo. In *Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, volume 1, pages 157–163. IEEE, 1999.

[185] Gabriel Taubin. A signal processing approach to fair surface design. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 351–358, 1995.

[186] C-H Teh and Roland T. Chin. On the detection of dominant points on digital curves. *IEEE Transactions on pattern analysis and machine intelligence*, 11(8) :859–872, 1989.

[187] J Nasehi Tehrani and J Wang. Mooney-rivlin biomechanical modeling of lung with inhomogeneous material. In *2015 37th Annual International Conference of the*

IEEE Engineering in Medicine and Biology Society (EMBC), pages 7897–7900. IEEE, 2015.

[188] Demetri Terzopoulos and Keith Waters. Analysis and synthesis of facial image sequences using physical and anatomical models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(6) :569–579, 1993.

[189] Philippe Thévenaz, Thierry Blu, and Michael Unser. Interpolation revisited [medical images application]. *IEEE Transactions on medical imaging*, 19(7) :739–758, 2000.

[190] Stephen Thompson, Crispin Schneider, Michele Bosi, Kurinchi Gurusamy, Sébastien Ourselin, Brian Davidson, David Hawkes, and Matthew J Clarkson. In vivo estimation of target registration errors during augmented reality laparoscopic surgery. *CARS*, 13(6) :865–874, 2018.

[191] Stephen Thompson, Johannes Totz, Yi Song, Stian Johnsen, Danail Stoyanov, Sébastien Ourselin, Kurinchi Gurusamy, Crispin Schneider, Brian Davidson, and David Hawkes. Accuracy validation of an image guided laparoscopy system for liver resection. In *Medical Imaging 2015 : Image-Guided Procedures, Robotic Interventions, and Modeling*, volume 9415, page 941509. SPIE, 2015.

[192] Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography : a factorization method. *International journal of computer vision*, 9(2) :137–154, 1992.

[193] Masahiro Tomono. Robust 3d slam with a stereo camera based on an edge-point icp algorithm. In *2009 IEEE International Conference on Robotics and Automation*, pages 4306–4311. IEEE, 2009.

[194] Kenneth E Torrance and Ephraim M Sparrow. Theory for off-specular reflection from roughened surfaces. *Josa*, 57(9) :1105–1114, 1967.

[195] Johannes Totz, Stephen Thompson, Danail Stoyanov, Kurinchi Gurusamy, Brian R Davidson, David J Hawkes, and Matthew J Clarkson. Fast semi-dense surface reconstruction from stereoscopic video in laparoscopic surgery. In *International Conference on Information Processing in Computer-Assisted Interventions*, pages 206–215. Springer, 2014.

[196] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. In *International workshop on vision algorithms*, pages 298–372. Springer, 1999.

[197] Stephen T Tully. BodySLAM : Localization and Mapping for Surgical Guidance. 2012.

[198] Radim Tylecek and Radim Sara. Depth map fusion with camera position refinement. In *Computer Vision Winter Workshop*, volume 70, pages 79–83, 2009.

[199] Andrés F Mármol Vélez, Jan Marek Marcinczak, and Rolf-Rainer Grigat. Structure from motion based approaches to 3d reconstruction in minimal invasive laparoscopy. In *International Conference Image Analysis and Recognition*, pages 296–303. Springer, 2012.

[200] S Vijayan, V Rana, Z Xiong, S Setlur Nagesh, S Rudin, and D Bednarek. Th-ab-201-01 : A real-time skin-dose mapping system for region-of-interest (roi) fluoroscopy. *Medical physics*, 42(6Part42) :3717–3717, 2015.

[201] George Vogiatzis, Philip HS Torr, and Roberto Cipolla. Multi-view stereo via volumetric graph-cuts. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 391–398. IEEE, 2005.

[202] Francesco Volonté, François Pugin, Pascal Bucher, Maki Sugimoto, Osman Ratib, and Philippe Morel. Augmented reality and image overlay navigation with osirix in laparoscopic and robotic surgery : not only a matter of fashion. *Journal of Hepato-Biliary-Pancreatic Sciences*, 18(4) :506–509, 2011.

[203] Lawrence T Wang, Timothy D Solberg, Paul M Medin, and Rex Boone. Infrared patient positioning for stereotactic radiosurgery of extracranial tumors. *Computers in biology and medicine*, 31(2) :101–111, 2001.

[204] Lichun Wang, Ran Chen, and Dehui Kong. An improved patch based multi-view stereo (pmvs) algorithm. In *3rd International Conference on Computer Science and Service System*. Atlantis Press, 2014.

[205] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment : from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4) :600–612, 2004.

[206] Jian Wei, Benjamin Resch, and Hendrik PA Lensch. Multi-view depth map estimation with cross-view consistency. In *BMVC*, 2014.

[207] Thomas Whelan, Stefan Leutenegger, R Salas-Moreno, Ben Glocker, and Andrew Davison. Elasticfusion : Dense slam without a pose graph. Robotics : Science and Systems, 2015.

[208] John R Williams, Eric Perkins, and Ben Cook. A contact algorithm for partitioning n arbitrary sized objects. *Engineering Computations*, 2004.

[209] Ivo Wolf, Marcus Vetter, Ingmar Wegner, Marco Nolden, Thomas Bottger, Mark Hastenteufel, Max Schobinger, Tobias Kunert, and Hans-Peter Meinzer. The medical imaging interaction toolkit (mitk) : a toolkit facilitating the creation of interactive software by extending vtk and itk. In *Medical Imaging 2004 : Visualization, Image-Guided Procedures, and Display*, volume 5367, pages 16–27. International Society for Optics and Photonics, 2004.

[210] Robert J Woodham. Photometric method for determining surface orientation from multiple images. *Optical engineering*, 19(1) :191139, 1980.

[211] JZ Wu, W Herzog, and M Epstein. Evaluation of the finite element software abaqus for biomechanical modelling of biphasic tissues. *Journal of biomechanics*, 31(2) :165–169, 1997.

[212] Koichiro Yamaguchi, David McAllester, and Raquel Urtasun. Efficient joint segmentation, occlusion labeling, stereo and flow estimation. In *European Conference on Computer Vision*, pages 756–771. Springer, 2014.

[213] Soji Yamakawa and Kenji Shimada. Removing self intersections of a triangular mesh by edge swapping, edge hammering, and face lifting. In *Proceedings of the 18th International Meshing Roundtable*, pages 13–29. Springer, 2009.

[214] Tomonori Yamamoto, Niki Abolhassani, Sung Jung, Allison M Okamura, and Timothy N Judkins. Augmented reality and haptic interfaces for robot-assisted surgery. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 8(1) :45–56, 2012.

[215] Dong Yang, Daguang Xu, S Kevin Zhou, Bogdan Georgescu, Mingqing Chen, Sasa Grbic, Dimitris Metaxas, and Dorin Comaniciu. Automatic liver segmentation using an adversarial image-to-image network. In *International Conference on*

Medical Image Computing and Computer-Assisted Intervention, pages 507–515. Springer, 2017.

[216] Jimei Yang, Brian Price, Scott Cohen, Honglak Lee, and Ming-Hsuan Yang. Object contour detection with a fully convolutional encoder-decoder network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 193–202, 2016.

[217] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet : Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 767–783, 2018.

[218] Koichiro Yasaka, Hiroyuki Akai, Osamu Abe, and Shigeru Kiryu. Deep learning with convolutional neural network for differentiation of liver masses at dynamic contrast-enhanced ct : a preliminary study. *Radiology*, 286(3) :887–896, 2018.

[219] Jae-Chern Yoo and Tae Hee Han. Fast normalized cross-correlation. *Circuits, systems and signal processing*, 28(6) :819, 2009.

[220] Paul A Yushkevich, Yang Gao, and Guido Gerig. Itk-snap : An interactive tool for semi-automatic segmentation of multi-modality biomedical images. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3342–3345. IEEE, 2016.

[221] Ruo Zhang, Pin-Sing Tsai, et al. Analysis of shape from shading techniques. In *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 377–384. IEEE, 1994.

[222] Yong-Ping Zheng and Arthur FT Mak. An ultrasound indentation system for biomechanical properties assessment of soft tissues in-vivo. *IEEE transactions on biomedical engineering*, 43(9) :912–918, 1996.

[223] Chao Zhou, Hong Zhang, Xiaoyong Shen, and Jiaya Jia. Unsupervised learning of stereo matching. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1567–1575, 2017.

[224] Olgierd Cecil Zienkiewicz, Robert Leroy Taylor, Perumal Nithiarasu, and JZ Zhu. *The finite element method*, volume 3. McGraw-hill London, 1977.

[225] Mariken Zijlmans, Thomas Langø, Erlend Fagertun Hofstad, Christiaan FP Van Swol, and Anna Rethy. Navigated laparoscopy–liver shift and deformation

due to pneumoperitoneum in an animal model. *Minimally Invasive Therapy & Allied Technologies*, 21(3) :241–248, 2012.