Contributions to the Automation of Preoperative 3D Model to 2D Image Registration in Mini-Invasive Liver Surgery : Primitive Detection, Pose Estimation, Registration and Anatomical Modelling

Mathieu Labrunie

Université Clermont Auvergne
EnCoV, Institut Pascal

Ecole doctorale SPI
SCIENCES POUR L'INGENIEUR

Thèse

pour obtenir le titre de

Docteur en Sciences

Présentée et soutenue par
Mathieu LABRUNIE

Contributions to the Automation of Preoperative 3D Model to 2D Image Registration in Mini-Invasive Liver Surgery

–

Primitive Detection, Pose Estimation,
Registration and Anatomical Modelling

–

Direction de thèse :	Adrien BARTOLI	-	EnCoV-UCA, IP; DRCI, CHU Clermont-Ferrand
Co-encadrement :	Daniel PIZARRO	-	Université d'Alcalá
	Christophe TILMANT	-	EnCoV-UCA, IP

soutenue le 2 Décembre 2024

Jury :

Président :	Pierre JANNIN	-	MediCIS-INSERM Rennes
Rapporteurs :	Stefanie SPEIDEL	-	NCT/UCC Dresden
	Timothy COOTES	-	Université de Manchester
Directeur :	Adrien BARTOLI	-	EnCoV-UCA, IP; DRCI, CHU Clermont-Ferrand
Co-encadrants :	Daniel PIZARRO	-	Université d'Alcalá
	Christophe TILMANT	-	EnCoV-UCA, IP
Invité :	Emmanuel BUC	-	Service de chirurgie digestive, CHU Clermont-Ferrand; EnCoV-UCA, IP

Abstract

Minimally invasive liver resection consists in removing liver parts enclosing tumours using surgical tools, while visualising the abdominal cavity through an endoscope, both inserted through small incisions in the abdominal wall. It offers significant advantages over open liver resection, including fewer postoperative complications and shorter hospital stays. However, the localisation of liver inner structures, such as tumours and blood vessels, remains challenging.

This information can be extracted from preoperative imaging and used for building a 3D model of the liver with its inner structures. However, this model must be moved and deformed for its projection to be aligned with the 2D image of the intra-abdominal surgical scene; this is the 3D/2D registration problem. Augmented reality enhances mini-invasive images with information from the registered preoperative model. Accurate augmented information could alleviate the limitations of mini-invasive surgery.

To this end, previous computer-based approaches have established a patient-specific registration pipeline, mainly relying on 3D/2D liver surface landmark correspondences to estimate pose (rigid movement) and then deformation. However, these methods still contain manual steps. In clinical practice, this is not convenient for the surgeon, whose focus should not be disturbed and gloves should be kept sterile. The main objective of this thesis is to improve upon this baseline by automating the 3D-2D intraoperative registration process using 3D/2D correspondence information. We propose two approaches: automating the manual intraoperative steps of the registration pipeline, or using a learning-based framework.

We first review the baseline pipeline and redefine the landmarks. This facilitates the identification of relevant 3D/2D correspondences. Additionally, we compare different deformation models and select one based on biomechanical simulations followed by dimension reduction.

Next, we automate the manual intraoperative steps from the baseline pipeline, comprising landmark annotation on minimally invasive images and pose estimation. We formulate the former as an image segmentation task and compare segmentation neural networks based on encoder-decoder architectures. The best results for image independent inputs are achieved with a fully attention-based network, but these are further improved when incorporating additional information from other images and masks. Pose estimation is tackled using an iterative visibility-aware algorithm, refining 3D/2D landmark point correspondences to estimate pose according to the visible 3D surface landmark parts from the previous iteration. This method obtains competitive results compared to manual pose estimation, while executing in a few seconds.

Regarding the learning-based framework, we draw connections to human body shape reconstruction to adapt an encoder-regressor architecture network to the 3D/2D liver registration problem. Distance maps of automatically or manually annotated landmarks are input to the encoder, while pose and deformation parameters are iteratively regressed. Preoperative training involves simulating corresponding inputs and outputs. This patient-

specific approach obtains registration results on par with previous state-of-the-art methods, while ensuring real-time network inference.

Instead of a patient-specific deformation model, the second learning-based approach uses a generic liver shape model, which is built using anatomical priors. This leads to very low surface registration and reconstruction errors. This patient-generic approach also includes a preoperative block for processing patient-specific data. Although the 3D-2D registration accuracy is slightly lower than that of patient-specific methods, it does not require per-patient retraining and can be applied without patient-specific data, facilitating both patient-generic and patient-specific image augmentation.

Keywords: 3D/2D Registration, Mini-Invasive Surgery, Liver, Generic Model, Segmentation, Augmented Reality

Résumé

La résection hépatique consiste à retirer des parties du foie qui englobent des tumeurs. Elle peut être réalisée de manière mini-invasive, par le biais d'instruments chirurgicaux et d'un endoscope insérés au travers de la paroi abdominale par de petites incisions. La chirurgie mini-invasive a des avantages importants par rapport à la chirurgie ouverte, comme des complications postopératoires et une durée d'hospitalisation réduites. Cependant, la localisation des structures internes du foie telles que les tumeurs et les vaisseaux sanguins est difficile.

Ces positions peuvent être extraites d'une imagerie préopératoire du patient, et utilisées pour construire un modèle 3D du foie avec ses structures internes d'intérêt. Cependant, le modèle 3D doit être déplacé et déformé pour que sa projection corresponde à l'image 2D de la scène mini-invasive intra-abdominale : c'est le problème du recalage 3D/2D. Les informations du modèle préopératoire recalé peuvent être augmentées sur l'image pour être visualisées par le chirurgien, lui apportant la réalité augmentée. Des informations augmentées précises pourraient réduire les problèmes de localisation en chirurgie mini-invasive.

Des travaux précédents ont mis en place une méthodologie informatisée de base pour le recalage 3D/2D spécifique au patient. Elle consiste à estimer le mouvement rigide (la pose) puis la déformation du modèle préopératoire, en se basant principalement sur des correspondances 3D/2D de repères de surface de foie. Cependant, elle contient des étapes manuelles, ce qui n'est pas adapté à la pratique clinique, notamment dû aux besoins de ne pas ajouter de la charge mentale au chirurgien et de généralement préserver son habillage stérile. L'objectif principal de cette thèse est de passer à l'étape suivante, en automatisant la procédure de recalage 3D/2D peropératoire. Deux voies sont envisagées pour le réaliser : automatiser chaque étape peropératoire manuelle de la méthodologie de base, et exploiter l'apprentissage profond.

Pour la première voie, nous formulons d'abord l'étape d'annotation des repères 2D comme un problème de segmentation d'image et comparons différentes architectures encodeur-décodeur. Un réseau de neurones entièrement basé sur le mécanisme d'attention obtient les meilleurs résultats, pour des entrées d'images indépendantes. Cependant, il est surpassé par un réseau prenant en compte des informations supplémentaires provenant d'autres images et masques. L'estimation de pose est ensuite automatisée via un algorithme itératif qui prend en compte la visibilité des repères 3D de l'itération précédente pour affiner la pose estimée. Il s'exécute en quelques secondes et obtient des résultats très compétitifs par rapport à l'estimation manuelle.

Pour la seconde voie basée sur l'apprentissage, nous établissons un lien avec la reconstruction de forme de corps humain afin d'adapter une architecture encodeur-régresseur au problème de recalage 3D/2D du foie. Des cartes de distance de repères annotés alimentent l'encodeur, alors que des paramètres de pose et déformation sont régressés itérativement. L'apprentissage préopératoire est basé sur des simulations. Cette première version obtient des résultats de recalage équivalents aux précédentes

méthodes de l'état de l'art, mais son inférence est en temps-réel. Une seconde version remplace le modèle de déformation spécifique au patient par un modèle de forme de foie générique, construit en utilisant des informations anatomiques. Cela résulte en de très faibles erreurs de recalage et de reconstruction de surface. Cette version générique contient aussi un bloc préopératoire pour traiter des données spécifiques au patient. Bien que sa performance soit légèrement inférieure aux autres méthodes, elle ne requiert pas de ré-entraînement pour chaque patient et peut s'appliquer sans données spécifiques, facilitant l'augmentation générique en plus de spécifique.

Mots-clés : recalage 3D-2D, chirurgie mini-invasive, foie, modèle générique, segmentation, réalité augmentée

Résumé long

Contexte. La résection hépatique consiste à retirer des parties anatomiques ou non anatomiques du foie qui englobent des tumeurs. Elle peut être réalisée de manière mini-invasive, par le biais d'instruments chirurgicaux insérés au travers de la paroi abdominale par de petites incisions équipées de trocarts. Le chirurgien peut visualiser la scène peropératoire intra-abdominale à l'aide de la source de lumière et de la caméra d'un endoscope, également introduit par un trocart. La chirurgie mini-invasive a des avantages importants par rapport à la chirurgie ouverte, comme des complications post-opératoires réduites ainsi qu'une durée d'hospitalisation réduite. Cependant, elle est limitée par l'absence de palpation du foie par le chirurgien qui rend difficile la localisation des structures internes du foie telles que les tumeurs et les vaisseaux sanguins.

Ces informations peuvent être extraites d'une imagerie préopératoire du patient, réalisée par exemple par tomodensitométrie, et utilisées pour construire un modèle 3D du foie avec ses structures internes d'intérêt. Cependant, le modèle 3D doit être déplacé, orienté et déformé pour que sa projection corresponde à l'image 2D de la scène mini-invasive intra-abdominale : c'est le problème du recalage 3D/2D. Les informations du modèle préopératoire recalé peuvent être augmentées sur l'image pour être visualisées par le chirurgien, lui apportant la réalité augmentée. Si les informations augmentées étaient obtenues automatiquement et rapidement tout en étant suffisamment précises, cela réduirait la limite de la chirurgie mini-invasive et contribuerait à son amélioration et son expansion.

Dans ce but, le problème du recalage 3D/2D en chirurgie mini-invasive du foie a commencé à être abordé en utilisant la vision par ordinateur. Des travaux précédents ont mis en place une méthodologie de base pour le recalage 3D/2D spécifique au patient. Elle consiste à estimer le mouvement rigide (la pose) puis la déformation du modèle préopératoire, en se basant principalement sur des correspondances 3D/2D de repères de surface de foie. Cependant, elle contient des étapes manuelles, ce qui n'est pas adapté à la pratique clinique, notamment dû aux besoins de ne pas ajouter de la charge mentale au chirurgien et au fait qu'il revêtit généralement des gants stériles, l'empêchant d'interagir via ses mains avec du matériel informatique. L'objectif principal de cette thèse est de passer à l'étape suivante, en automatisant la procédure de recalage 3D/2D peropératoire, tout en utilisant des informations de correspondance similaires aux travaux précédents.

Deux voies sont envisagées pour le réaliser : soit en automatisant chaque étape peropératoire de la méthodologie de base, soit en suivant une nouvelle approche automatique, exploitant l'apprentissage profond par réseaux de neurones artificiels.

Méthodologie de base. La méthodologie de base comprend de nombreuses étapes, qui peuvent être divisées en deux phases : préopératoire et peropératoire. La phase préopératoire commence par segmenter le volume obtenu par l'imagerie préopératoire du patient, afin de reconstruire les maillages ou modèles de surface du foie du patient et de ses structures internes. Ces maillages sont lissés, ré-échantillonnés et nettoyés. Ensuite,

le maillage volumique du foie du patient est reconstruit avec la contrainte de partager les sommets du maillage de surface, tout en ajoutant des sommets à l'intérieur. Enfin, les sommets des maillages de surface des structures internes sont reliés au maillage volumique du foie à l'aide de coordonnées et repères barycentriques.

Une deuxième étape consiste à modéliser la déformation du foie et de ses structures internes en conditions mini-invasives, afin de réduire l'espace des déformations possibles et faciliter ainsi son estimation. Nous avons comparé plusieurs modèles de déformation et en avons sélectionné un basé sur des simulations biomécaniques de forces nodales appliquées aléatoirement sur la surface du foie et atteignant de grandes amplitudes, suivies par une réduction de dimension des données simulées. La dernière étape préopératoire est l'annotation des repères anatomiques sur le modèle 3D.

La phase peropératoire débute par l'étalonnage de la caméra de l'endoscope afin d'obtenir ses paramètres intrinsèques ainsi que ceux de distorsion. Le chirurgien doit initialement choisir les paramètres de caméra qui lui permettent de visualiser le foie et ses repères de façon globale. Ensuite, il peut procéder à l'étalonnage en filmant une mire contenant des motifs plans de dimensions connues, permettant de retrouver les paramètres de caméra recherchés par le biais d'algorithmes de vision par ordinateur dédiés. Ensuite, l'annotation des repères anatomiques sur l'image mini-invasive est réalisée manuellement. Ainsi, des repères 3D/2D correspondants sont obtenus. Ces étapes préparatoires permettent d'obtenir les données nécessaires pour recaler le modèle préopératoire sur l'image mini-invasive considérée. Dans un premier temps, le recalage 3D/2D rigide est effectué. La pose des modèles dans l'espace de la caméra est d'abord estimée manuellement. Pour cela, le chirurgien essaie de déplacer et tourner le modèle préopératoire virtuellement pour que ses repères s'ajustent à ceux de l'image. Ensuite, le recalage déformable est réalisé automatiquement, utilisant la pose estimée pour initialiser la position du foie dans l'espace de la caméra et déterminer la déformation qui permette l'ajustement des repères du modèle à ceux de l'image. Des contraintes de régularisation de la déformation peuvent également être ajoutées, à l'aide par exemple du modèle de déformation.

Réflexion sur les repères. Préalablement à l'automatisation des étapes peropératoires manuelles, nous proposons une réflexion sur les repères. Il s'agit de la crête antérieure du foie, qui délimite ses surfaces inférieures et supérieures antérieures, de la jonction du foie avec le ligament falciforme, ainsi que de sa silhouette ou frontière occultante supérieure. La jonction du foie avec le ligament falciforme n'est généralement pas visible sur le volume préopératoire du patient et donc généralement tracée grossièrement. Ainsi, sa prise en compte dans les correspondances 3D/2D peut potentiellement nuire à la précision du recalage, sur lequel elle exerce une contrainte. Des expérimentations avec et sans ce repère ont confirmé cette hypothèse. Ensuite, la crête antérieure du foie est considérée comme un seul morceau. Cependant, en fonction de la vue du foie, la partie centrale peut correspondre à différentes positions dans le modèle, et l'unicité de la crête peut rendre la détermination précise de correspondances difficile. En la divisant en plusieurs parties latérales (gauche, droite) et centrales (supérieures et inférieures gauche et droite), une amélioration conséquente des résultats de recalage basés sur des correspondances de points 3D/2D est observée, suggérant que cette division facilite la détermination de correspondances 3D/2D adéquates.

Ces repères sont moins visibles quand le foie est manipulé pour accéder aux structures internes postérieures et donc l'utilisation de ces repères pour le recalage est plus adaptée à la localisation de structures internes du foie antérieures et supérieures. De plus, elle est également plus adaptée à des vues globales exploratoires du foie, avant le début de la résection, où la plupart des repères sont visibles et non occultés par du sang, des instruments ou de la gaze.

Annotation automatique des repères. Nous abordons ensuite l'automatisation de l'annotation des repères sur les images mini-invasives. Nous formulons cette tâche comme un problème de segmentation d'image. Cela nous permet d'étudier et d'implémenter de nombreux réseaux de neurones artificiels dédiés à cette tâche, utilisant donc des méthodes d'apprentissage profond. Avec nos partenaires de centres hospitaliers universitaires, nous avons d'abord collecté et annoté de nombreuses images afin de constituer une base de données à partir de laquelle entraîner et valider les réseaux. Nous avons ensuite implémenté de nombreux réseaux de neurones convolutifs basés sur une structure encodeur pour progressivement extraire les caractéristiques d'image à différentes résolutions, suivie d'une structure décodeur pour progressivement reconstruire des masques de segmentation à partir de ces caractéristiques. Parmi ceux-ci, les réseaux en U (UNet), avec encodeur résiduel (ResUNet), le réseau CASENet avec un décodeur normalement plus adapté à des contours ont été évalués. Parmi ceux-ci, le réseau ResUNet obtient les meilleurs résultats. Il est cependant dépassé par un réseau complètement basé sur le mécanisme d'attention (opération non locale), le Mask2Former. Il a aussi une structure encodeur-décodeur, mais contient deux chemins de décodage combinés, suivant une formulation supplémentaire de classification de masque parmi un nombre élevé de propositions de masques de segmentation, en plus d'une classification par pixel. Il utilise un mécanisme d'attention masquée dans le décodeur supplémentaire, se concentrant sur les positions des objets cibles, afin d'éviter d'être perturbé par du bruit de fond. Une autre manière d'améliorer les résultats obtenus par le ResUNet consiste à insérer entre l'encodeur et le décodeur un bloc de calcul de co-attention entre des caractéristiques provenant d'autres images proches et celles de l'image courante (COSNet). La co-attention peut être également calculée à partir d'un autre encodeur pour lequel les entrées sont à la fois une autre image et son masque de segmentation associé, ce qui permet de surpasser les résultats obtenus par les autres réseaux. Ce réseau (STM) bénéficie d'un entraînement et d'une inférence utilisant des échantillons de contenus proches, comme des images (et certains masques associés) séquentiellement proches. A part le STM, tous les réseaux appris sur une base de données mini-invasives typiques ne généralisent pas bien à l'ensemble d'images utilisé pour la validation des méthodes de recalage, probablement dû à la présence d'une sonde avec des marqueurs noirs et blancs attachés, qui modifie le domaine d'images. Cela nous empêche d'obtenir une validation représentative de la combinaison de la segmentation automatique suivie par les différentes méthodes de recalage.

Automatisation du recalage rigide. Nous automatisons ensuite le recalage rigide, à partir d'un processus itératif qui affine progressivement l'estimation de pose, basée sur des points correspondants provenant des repères 3D/2D correspondants. L'estimation détermine une solution au problème de Perspective-n-Point (PnP) ainsi formé, basée sur

l'algorithme RANSAC. Pour un même ensemble de points correspondants, nous proposons d'utiliser plusieurs seuils de tolérance d'erreur de reprojection pour RANSAC, ce qui permet d'obtenir différents sous-ensembles de points correspondants pour consensus et d'estimer ainsi plusieurs poses. La meilleure pose est sélectionnée selon un critère de distance symétrique entre les ensembles de points correspondants projetés et cibles. Cette utilisation de multiples seuils de tolérance a pour but de s'adapter à différents cas d'approximation d'un champ de déformation plus ou moins important par un modèle rigide, en plus de correspondances plus ou moins précises. La première étape du processus itératif estime grossièrement la pose, en considérant que l'ensemble des repères anatomiques 3D est visible, et que les points correspondants à ceux de l'image sont uniformément répartis sur chaque repère. La deuxième étape considère seulement des points de repères anatomiques visibles à partir de la pose estimée à l'étape précédente. Enfin, la dernière étape ajoute les points de la silhouette à ceux des repères anatomiques visibles pour affiner encore le résultat. A noter, les deux dernières étapes sont répétées. Cette méthode obtient des résultats extrêmement compétitifs avec l'estimation de pose manuelle tout en s'exécutant en seulement quelques secondes. Ses résultats peuvent être encore légèrement améliorés en étant suivie du recalage déformable par optimisation des paramètres du modèle de déformation.

Automatisation du recalage par apprentissage patient-spécifique. Les premiers travaux concernaient l'automatisation de chaque étape manuelle de la méthodologie de base. Les suivants traitent d'une autre approche basée sur de l'apprentissage profond, une spécifique au patient, et une générique. Les deux se basent sur l'architecture d'un réseau de neurones, le HMR (Human Mesh Recovery), initialement conçu pour retrouver le maillage générique d'un corps humain sur des images, à l'aide d'un modèle articulé et de forme du corps humain. Le principe consiste à utiliser un encodeur pour obtenir des caractéristiques d'image qui sont transmises à un régresseur itératif de paramètres de caméra, de pose et du modèle articulé et de forme afin de mettre ces paramètres à jour progressivement en fonction d'un retour d'erreur. L'apprentissage requiert initialement de calculer une fonction de coût entre repères (points d'articulation) cibles et projetés, et donc de nombreuses images annotées.

Nous avons adapté ce réseau au problème de recalage 3D/2D du foie spécifique au patient en utilisant le modèle de déformation du foie du patient à la place du modèle articulatoire et de forme du corps humain. Nous avons utilisé les repères 3D/2D correspondants de surface du foie pour guider l'apprentissage. Ce contexte requiert également une autre adaptation, due au fait que les données réelles du patient deviennent seulement disponibles au moment de la chirurgie, et que le nombre de données annotées est très réduit. Elle consiste à alimenter le réseau par des cartes de distances de repères au lieu d'images mini-invasives, permettant ainsi de réaliser de nombreuses simulations préopératoires de configurations du foie du patient déformé avec les masques de repères associés, afin d'entraîner le réseau. Ces simulations et l'apprentissage prennent à peu près un jour par patient, une durée conséquente mais inférieure au délai classique entre l'imagerie préopératoire et l'opération. Pour l'inférence peropératoire, le réseau utilise les repères détectés automatiquement ou manuellement. Cette adaptation a pour conséquence de ne plus avoir un réseau bout à bout, mais deux réseaux successifs, un de segmentation et

un de recalage. Le temps d'exécution du réseau de recalage en inférence est extrêmement rapide, près de 3 ms, ce qui permet un recalage peropératoire en quasi-temps-réel quand il est combiné à la segmentation automatique. Avec segmentation manuelle, l'erreur de recalage obtenue avec cette approche est comparable à celle des méthodes précédentes de l'état de l'art, mais l'erreur de reprojection est plus élevée.

Automatisation du recalage par apprentissage patient-générique. Nous avons ensuite considéré le problème du recalage 3D/2D générique au patient, et avons remplacé le modèle de déformation du foie du patient par un modèle de forme générique du foie. Nous avons guidé sa construction à l'aide d'informations anatomiques : des points caractéristiques de surface du foie. Ces points caractéristiques sont d'abord annotés sur un ensemble de formes (maillages de surface) de foie. Une forme est sélectionnée comme la référence, et elle est recalée à toutes les autres formes à l'aide des points caractéristiques. Le recalage de surface est mené par une méthode de recalage d'ensembles de points : une méthode itérative non-rigide de points les plus proches (NR-ICP), initialisée par une fonction de base radiale (RBF). Une fois que l'ensemble des recalages de surface est réalisé, les formes ont toutes le même nombre de sommets avec des correspondances une-à-une. Elles sont alignées en utilisant une analyse procustéenne généralisée (GPA) et la forme moyenne résultante est conservée. Le maillage volumique associé à cette forme moyenne est reconstruit en respectant la contrainte de partager ses sommets de surface. A partir de cette forme volumique moyenne, nous utilisons des combinaisons de processus Gaussiens à plusieurs échelles comme composantes de forme du modèle générique (GPMM), qui peuvent être également transférées à la surface. Ces méthodes permettent d'obtenir des erreurs de recalage et de reconstruction de surface beaucoup plus faibles que sans guidage anatomique, moins de 6 mm sur plusieurs jeux de données. Cependant les erreurs de recalage et de reconstruction de points de bifurcation de branches majeures de vaisseaux sanguins du foie sont plus élevées, suggérant des différences entre variations anatomiques (inter-sujets) de surface et internes.

Lorsque ce modèle est intégré à l'approche basée sur de l'apprentissage, devenant générique, le réseau n'a pas besoin d'être ré-entraîné à chaque nouveau patient, ce qui facilite grandement sa déployabilité, et s'applique sans données spécifiques au patient. Cela permet donc l'augmentation d'images mini-invasives par des informations anatomiques définies sur le modèle volumique générique. Cependant, cela permet aussi le recalage spécifique au patient, en ajoutant une étape préopératoire pour transférer les structures internes du patient au modèle générique. L'erreur moyenne de recalage est légèrement supérieure aux méthodes spécifiques au patient, malgré une faible erreur de reprojection.

Actuellement, notre approche générique basée sur de l'apprentissage utilise des cartes de distances de repères en entrée du réseau. Quand un nombre suffisant de données annotées disponibles nous le permettra, nous envisageons de remplacer ces entrées par des images mini-invasives, bien que le réseau continue d'utiliser au moins une fonction de coût d'erreur de reprojection des repères pour être entraîné. Ainsi, nous obtiendrions un réseau bout-à-bout, qui prédit le foie déformé dans l'espace de la caméra à partir d'une image. Le lien que nous avons établi entre le problème du recalage 3D/2D du foie en chirurgie mini-invasive et celui de la reconstruction de forme du corps humain à partir d'une image nous permettra également de bénéficier des travaux les plus récents de ce

domaine, en particulier pour améliorer l'architecture et le mécanisme d'apprentissage des réseaux. De plus, l'utilisation d'une architecture contenant un encodeur permettra également de bénéficier des avancées les plus récentes de ces réseaux imbriqués dans des architectures encodeur-décodeur pour des tâches de segmentation, détection ou encore classification d'images.

Acknowledgments

It takes a village to raise a child. The same goes for this thesis work. I am extremely grateful to my supervisor Adrien Bartoli. First, for giving me the opportunity to explore this fascinating subject, full of different thematics and perspectives. Second, thank you for your reactiveness to answer my questions, your guidance and your support, in particular in difficult redaction times and deadlines. Third, thank you and the SurgAR direction for the ideal working conditions that I have benefited from. I am fully aware of my luck.

I am very thankful to Daniel Pizarro too. You provided me complementary assistance when I needed, in the code, paper figures, and even poster presentations! I have also appreciated a lot your way of supporting and motivating me with many encouragements.

I am also grateful to Christophe Tilmant for his support, in particular at the start of the PhD project, when I had some difficulties to enter in the process, and also for sharing hidden tools.

Obviously, I wish to thank the members of the jury, for accepting either to review my work in a short duration, or to examine it in their busy schedule. Thank you for your interesting questions and remarks too, which can guide future works.

As a guide before this PhD project, I am extremely grateful to Pierre Badin. You have spent a lot of time initiating me to the research universe, developing my critical thinking, providing me necessary basic knowledge and skills. Also, thank you for the extra discussions, which have participated in my personal development.

In addition, I thank all SurgAR or lab colleagues who I have worked with, shared funny moments, or intense discussions.

Many people that I don't know personally have also contributed to this PhD project, making and sharing awesome libraries, tutorials, insights. Thank you all for letting others learn and work more efficiently! Among them, I wish to cite [Commowick 2007] for his great manuscript template.

For others, thank you, my friends, for growing up together while staying kids. My family, for providing me a soft cocoon where I can recharge my batteries. My daily partner, for letting me be busy, travelling, angry, stressful and cope with all of that admirably. And my sunny son, but you are a bit too young to understand yet!

Contents

List of Acronyms

Chapter 1

Introduction

Contents

1.1 Context

This thesis falls within the partnership between the EnCoV[1] (Endoscopy and Computer Vision) research group and the SURGAR[2] (Surgical Augmented Reality) company through a CIFRE PhD fellowship (N° 2021/0184) from ANRT[3] (french National Agency of Research and Technology). This partnership aims to develop solutions for providing augmented reality in various surgical contexts involving different human organs, such as the uterus, the liver and the kidneys in mini-invasive surgery. It initially relies on CHU (french University Hospitals) partners for collecting data, in particular the first ones from Clermont-Ferrand and Saint-Etienne for the considered organ in this thesis, i.e. the liver. The study received ethical approval (IRB00008526-2019-CE58) issued by CPP Sud-Est

[1] Université Clermont Auvergne, Clermont Auvergne INP, CHU Clermont-Ferrand, CNRS, Institut Pascal, F-63000, 63000 Clermont-Ferrand, France. URL: https://encov.ip.uca.fr

[2] SURGAR, 22 Allée Alan Turing, 63000 Clermont-Ferrand, France. URL: https://surgar-surgery.com

[3] https://www.anrt.asso.fr/fr/le-dispositif-cifre-7844

VI in Clermont-Ferrand, France. The medical images of this manuscript come from these centres unless otherwise specified. While augmented reality is a scientific challenge, it is associated to a general clinical context and we tackle it depending on this local initial context.

In the following sections, the surgical context of this thesis, i.e. Minimally Invasive Liver Surgery (MILS), is first described by introducing the liver, its considered diseases and the surgical techniques for treating them. Then, assistance of the surgery through navigation techniques, with a focus on augmented reality, is outlined. Eventually, the motivation and the main contributions to automate augmented reality in MILS are presented.

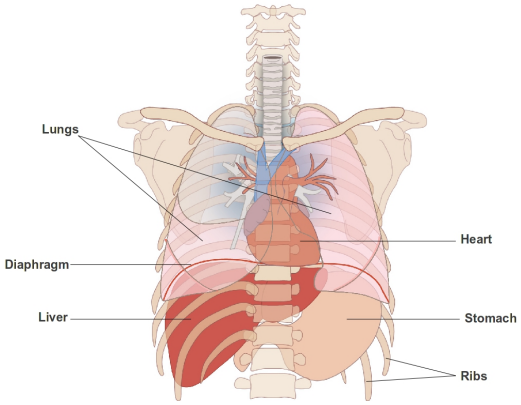
1.2 Minimally Invasive Liver Surgery

In this thesis, the human liver is the organ of interest. First, its characteristics, i.e. location, functions, anatomy and morphology are described. Second, the tumours that can occur and develop in the liver are overviewed. Third, the techniques to remove or ablate them by means of surgery, in particular the minimally invasive ones, are presented, along with their limitations. Eventually, existing navigation techniques for assisting the latter are presented.

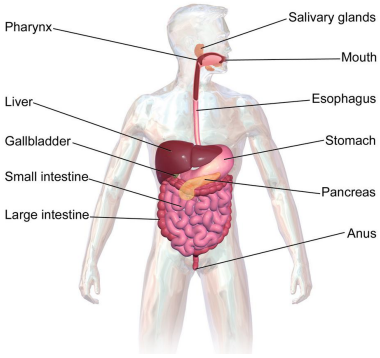
1.2.1 Liver Characteristics

The liver is located in the upper area of the abdominal cavity, beneath the diaphragm and on top of the stomach and the gallbladder, see Figures 1.1a and 1.1b. It belongs to the digestive system, illustrated in Figure 1.1b, and continually secretes bile for assisting the internal digestion of food after its passage into the oesophagus and then the stomach. The bile from the liver is driven to the small intestine (duodenum) through a bile duct tree (biliary tree or tract). It ends with the common bile duct, which joins the cystic duct from the gallbladder and the common hepatic duct, which itself joins increasingly smaller hepatic bile ducts, as shown in Figure 1.1c. The bile facilitates fat absorption and its transformation into energy (adenosine triphosphate) [Ozougwu 2017]. In addition, it excretes some products for regulating their level, e.g. cholesterol, or eliminating waste or toxic ones, such as bilirubin (whose excess causes jaundice) and drugs.

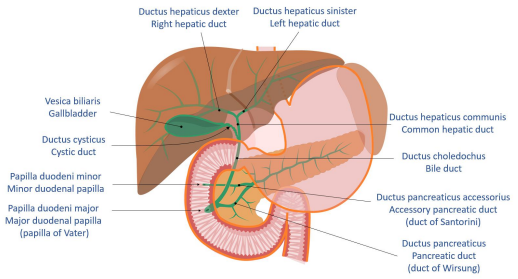
The liver is also an important part of the blood and lymph circulatory (vascular) system, illustrated in Figure 1.1d. It receives its arterial oxygenated blood supply from the heart through the hepatic artery. It also receives blood enriched in nutrient and toxic materials from the stomach and other digestive tract organs through the hepatic portal vein, before to process or filter it [Nagy *et al.* 2020]. Both the hepatic artery and the portal vein split in a tree fashion for providing the blood to the whole liver, as illustrated by Figure 1.1e. The liver sinusoids are the hepatic capillary structures for mixing the blood received from both ways. The processings includes the regulation of its glucose level, as well as the metabolic detoxification of substances such as drugs [Ozougwu 2017]. It also synthesises and regulates plasma proteins, such as clotting factors implied in coagulation. In addition, due to its large vascular network, the liver can help to store a large volume of blood (up to 27% of the total blood volume [Greenway 1983]) and assist the regulation



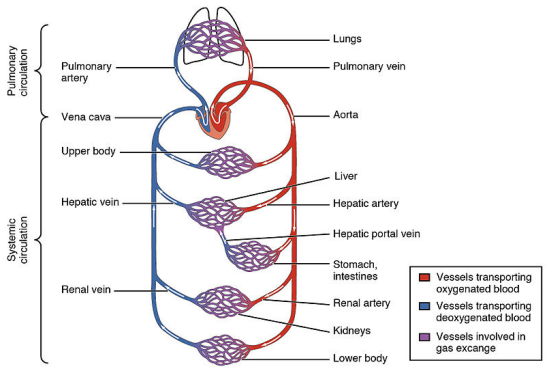
(a) Human liver location, adapted from [Slagter n.d.]



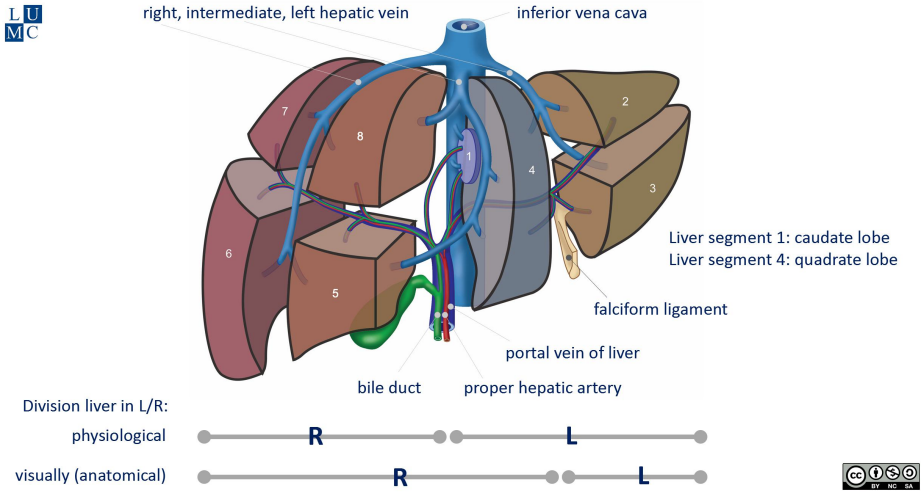
(b) Human digestive system, cropped from [Blausen.com staff 2014]



(c) Human biliary tree, from [Marchn *et al.* n.d.]



(d) Human circulatory system, from [OpenStax College n.d.]



(e) 1) Functional/physiological and 2) morphological anatomy of human liver. 1) Couinaud segments 1-8 and major vessels. 2) Right and left lobes separated by the falciform ligament. From [Blankevoort *et al.* n.d.].

Figure 1.1: Human liver in human body and systems.

of the systemic circulatory volume, e.g. in case of haemorrhages. The liver has also immunologic functions, such as the production of most of the circulating innate immunity proteins [Gao 2016]. Its tissue is mainly made of parenchymal cells (hepatocytes) which represent about 80% of its mass [Werner *et al.* 2015]. They form the liver parenchyma and are responsible for most of the liver functions [Damm *et al.* 2013]. This tissue is very soft and therefore the liver is highly deformable.

The functional liver anatomy is mainly described according to the Couinaud classification [Couinaud 1957], which splits the liver into eight independent macrovascular parenchymal segments, the Couinaud segments, centred on large portal vein branches and separated by large hepatic vein branches, see Figure 1.1e. They were originally numbered by Roman numerals, but the Arabic numerals are now encouraged [Strasberg *et al.* 2000]. The traditional morphological anatomy is based on the external appearance of the liver and divides the liver into two major lobes, separated by the remnants of the embryonic umbilical vein, i.e. the falciform ligament, which does not coincide with the previous functional anatomic division [Nagy *et al.* 2020], as shown in Figure 1.1e. The external surface of a healthy liver is smooth with a colour in the reddish brown palette.

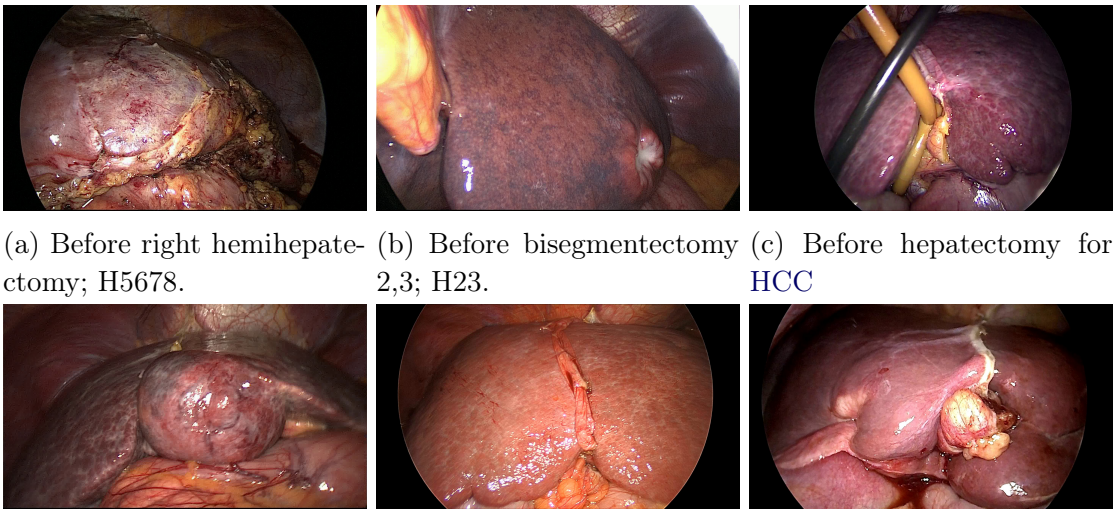
1.2.2 Liver Tumours subject to Surgery

A tumour is an abnormal growth or mass of tissue due to abnormal cell growth, division or death process. It can be benign (not spreading) or malignant, i.e. causing cancer. A cancer is a disease characterised by an uncontrolled (deregulated) growth and spread of abnormal cells. It can be metastatic, i.e. spreading to other parts of the body through the blood or lymph system.

The liver cancer can be primary, i.e. beginning in the liver tissue, or secondary, i.e. spreading from another part of the body to the liver. Primary liver cancer is the sixth most common cancer in the world and the third leading cause of cancer deaths (the second one in men) [Bray *et al.* 2024]. The prognosis of liver cancer is poor. For example, in France, the 5-year survival rate was 18% in 2018 [De Brauer *et al.* 2024]. Primary liver cancer mainly comprises Hepatocellular Carcinoma (HCC) (75% to 85% of cases) and Intrahepatic CholangioCarcinoma (ICC) (10% to 15% of cases). Risk factors include chronic infection of hepatitis B or hepatitis C viruses, aflatoxin exposure, heavy alcohol consumption, excess body weight, type 2 diabetes, and smoking. The main type of secondary liver cancer is due to colorectal liver metastases. Primary colorectal cancer is the third most common cancer in the world and the second leading cause of cancer deaths. Risk factors include alcohol consumption, smoking, consumption of red or processed meat, and body fatness [Bray *et al.* 2024]. 25% to 50% of colorectal cancer patients develop colorectal liver metastases during the course of their illness [Martin *et al.* 2020].

HCC is a progressive process. In response to liver injury, parenchymal cells regenerate and replace the necrotic or apoptotic cells to form a scar tissue. When this wound healing process is deregulated due to the cancer, it leads to liver fibrosis which is mainly characterised by the excessive accumulation of the non-functional scar tissue in the liver parenchyma, replacing the functional hepatic tissue [Rajapaksha 2022]. Hepatic fibrosis can reach several progressive stages as the cancer evolves until to reach cirrhosis, a stage of permanent scarring which interferes with the liver functioning. This modifies the appearance of the liver surface which becomes bumpier, as illustrated in Figure 1.2c. In general,

regarding HCC, liver resection should be considered for patients with non-metastatic disease and normal underlying liver function [Orcutt & Anaya 2018]. Other curative intent alternatives include tumour ablation [Knavel & Brace 2013] and orthotopic liver transplantation, see section 1.2.4.



(a) Before right hemihepatectomy; H5678.

(b) Before bisegmentectomy 2,3; H23.

(c) Before hepatectomy for HCC

(d) Before segmentectomy 3; H3.

(e) Before metastasectomy

(f) Before wedge resection of segment 4b; H4'.

Figure 1.2: Liver surface appearance on mini-invasive images for different patients and tumours before resection. Resections are named according to Brisbane [Strasberg *et al.* 2000] and New World conventions [Nagino *et al.* 2021].

ICC is characterised by the deregulation of cells in the liver bile ducts [Seo *et al.* 2017]. It can result in a mass in the parenchyma (the most common type), a dilatation of the peripheral biliary ducts, a growth inside the duct or a mix of both. Surgical resection is the only potential cure for ICC cancer patients. Among resectable patients, roughly 75% of patients require a hemihepatectomy or an extended one, described in section 1.2.4.3, for removing the tumour [Orcutt & Anaya 2018].

ColoRectal Liver Metastasis (CRLM) mainly occurs due to the direct connection between the colon and rectum and the liver through the portal vein circulation. First, circulating cancer cells reach liver sinusoids and capillaries, then they transit and proliferate in the liver parenchyma. This leads to the creation of a microenvironment that favours tumour growth and disrupts the normal function of the liver [Tsilimigras *et al.* 2023]. As for ICC, surgical resection is the only treatment modality for curative intent in colorectal liver metastases [Chow & Chok 2019].

Benign liver tumours can also be treated by surgery and can be categorised into two main groups: cystic lesions (cysts are fluid-filled sacs) and solid lesions [Gigot *et al.* 2004]. Only symptomatic tumours in which malignancy cannot be excluded are indicated for surgery [Fodor *et al.* 2018]. When a tumour is present in the liver, its surface appearance is generally modified and can show various patterns and colours, see Figure 1.2.

1.2.3 Preoperative Liver Tumour Localisation and Diagnosis

Liver tumour localisation and diagnosis are mainly performed by an imaging technique among Computed Tomography (CT), Magnetic Resonance Imaging (MRI) and abdominal UltraSound (US).

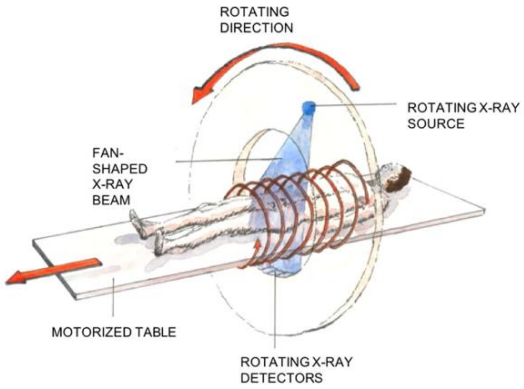
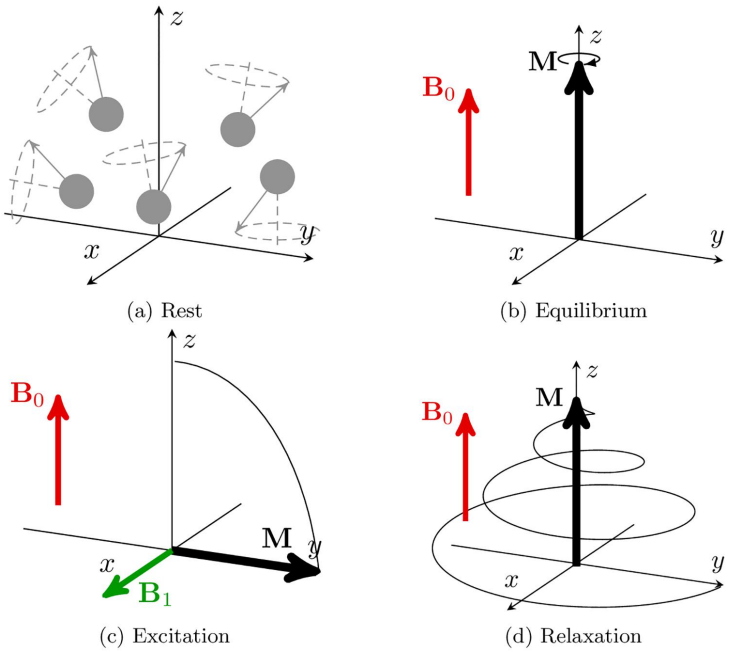
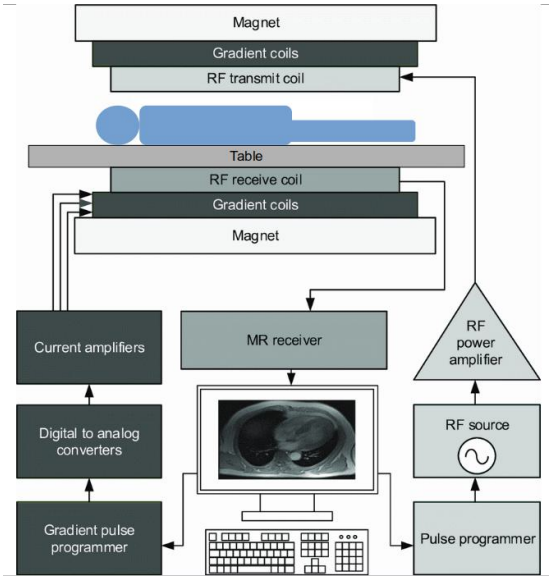


Figure 1.3: Fan beam CT-scanner principle, adapted from [Labriet *et al.* 2018]. The X-ray source and detector rotate around the patient while the table is moved for scanning the whole body. Not shown: a computer system controls the scanning process and translates the detected signals into images.

Computed tomography. It uses X-rays, electromagnetic ionising radiation, which are absorbed at a different extent according to the tissue (bones, parenchyma, tumour ...). CT scanners have a tubular ring-shaped structure with a motorised bed (table) across. CT scanners use a motorised X-ray source that shoots narrow fan beams at a patient lying on a bed in a supine position and rotates around the patient [Jung 2021], see Figure 1.3. The X-rays are attenuated according to the tissue that they pass through before reaching opposite X-ray detectors, which transmit the signal to a computer. After a full rotation, a slice can be computed, illustrated in Figure 1.6a. This is a cross-sectional image of the patient, with a thickness usually ranging from 1 to 10 mm [Abdulkareem *et al.* 2023], where the elements are called voxels. The grey level of a voxel depends on the X-ray attenuation of the tissue corresponding to the voxel. The bed is moved by incremental steps in order to image the next slices and scan the whole patient body, creating a preoperative volumetric (3D) image of the patient. This is the acquisition technique of the third-generation CT scanners. Nuances are present in others, such as spiral and continuous acquisition, while exploiting the same physical principle [Cunningham & Judy 2000]. In order to enhance the contrast between a lesion and the normal surrounding structures, a contrast agent can be injected, requiring to time the acquisition. CT is frequently used as it is quick, widely available, and accurate. However, it is limited by radiation exposure. Indeed, exposure to high intensities can be hazardous to health, causing DNA damage, cancer, burns and radiation sickness. In addition, it has limited characterisation of subcentimer hepatic lesions. Liver lesions with a size of 10 mm or less may appear indeterminate on CT because the attenuation and contrast-enhancement pattern of small lesions may remain nonspecific due to limited spatial resolution [Berger 2002].



(a) Nuclear magnetic resonance principle, from [Puiseux *et al.* 2021]. (a) When no static magnetic field (B_0) is applied, the spins are randomly oriented. When a B_0 field is applied along the z-axis, all the spins precess around the z-axis and (b) an equilibrium magnetisation (M) arises, oriented along the same axis. M is shifted towards the transverse xy-plane by the effects of a Radio Frequency (RF) pulse (B_1) applied at the resonant frequency (c). When the RF excitation is released, the magnetisation relaxes towards its equilibrium value (d).



(b) Block diagram of a typical MRI scanner, from [Jouda 2016]. The magnet generates the static magnetic field B_0. The gradient coils spatially encode the MR signals, controlling gradients to B0 in all axes. The RF transmit coil generates excitation signals which resonate at the desired frequency. The RF receive coil collects the released energy when the RF excitation pulse is switched off and produces an electrical signal representing the magnetic resonance signal. Coils are connected to a computer system which either controls the process or translates the signal into images of the selected slices, allowing the scan of the whole patient body.

Figure 1.4: MRI

Magnetic resonance imaging. It exploits magnetic properties of the hydrogen proton (nucleus) because of its presence in water and fat which are abundant in the human body tissues. The hydrogen nucleus spins about an axis, and this moving electric charge behaves similarly to a current in a loop of wire, i.e. producing a magnetic field. This is nuclear magnetism [Jensen 2014]. The MRI scanners are tubular and long. They use a powerful magnet and an RF system for transmitting and receiving waves through coils, while the patient lies on a bed inside the scanner in a supine position, see Figure 1.4b. The MRI magnet produces a strong magnetic field (of order of 1 Tesla) that forces hydrogen protons in the body to align with the field direction. The hydrogen absorbs energy if this energy is at the resonant frequency, and this energy will subsequently be re-emitted. An RF pulse is emitted at the resonant frequency allowing nuclear magnetic resonance. When turned off, the magnetic proton relaxes, i.e. returns to its resting and aligned state, see Figure 1.4a, and this causes a radio wave to be re-emitted. Different tissues relax at different rates when the transmitted RF pulse is switched off, and the time taken for the protons to fully relax is measured and used to produce a greyscale image of the selected slice [Berger 2002], illustrated in Figure 1.6b. Slices of the body are selected (around 5 mm for the liver tumour diagnosis) and performed incrementally in order to create a preoperative volumetric (3D) image of the patient. MRI is accurate and free of radiation exposure. It was reported to be of high specificity for liver nodules of small size, between 5 to 20 mm, resulting from optimal lesion-to-liver contrast. However, patient factors such as claustrophobia, implanted devices, discomfort, cost and availability may hinder its use for diagnostic imaging of liver disease [Parra *et al.* 2023].

Ultrasound. It is also called sonography and uses acoustic (ultrasound) waves. Abdominal US uses a small ultrasound transducer (probe) converting electrical energy into sound (mechanical) one and vice versa, based on the piezoelectric effect. The probe is pressed firmly against the skin of the abdomen, see Figure 1.5, and high-frequency (of several MHz order) sound waves travel from the probe into the body tissues. The probe collects the sound waves that bounce back (echoes) and they are translated into greyscale images (sonograms), illustrated in Figure 1.6c, visible from a connected mobile cart, taking into account ultrasonic beam direction and pulse round-trip transit time.

In addition to this pulse-echo technique, the Doppler technique can be used. It analyses the returning echoes in terms of Doppler shift rather than amplitude. The Doppler shift is the difference between the frequency of the incident ultrasound beam and that of the received echoes. A series of pulses is transmitted, and echoes from stationary tissue are unchanged from pulse to pulse while echoes from moving elements show differences in the frequency of the signal returned to the receiver, enabling one to detect the movement of blood. A processing allows a colour flow display [Uppal & Mogra 2010], according to the blood flow direction and velocity through arteries and veins in the body, enabling their localisation, see Figure 1.6d.

3D abdominal US can also be performed [Sackmann *et al.* 1994], based on the acquisition of a set of 2D images and computer-based 3D integration [Kim & Choi 2007]. Abdominal ultrasound is inexpensive with respect to CT and MRI and is thus widespread. A major limitation is that ultrasonic waves are transmitted neither through bone nor air, which is why a gel is applied on the skin of the patient in order to reduce the air between

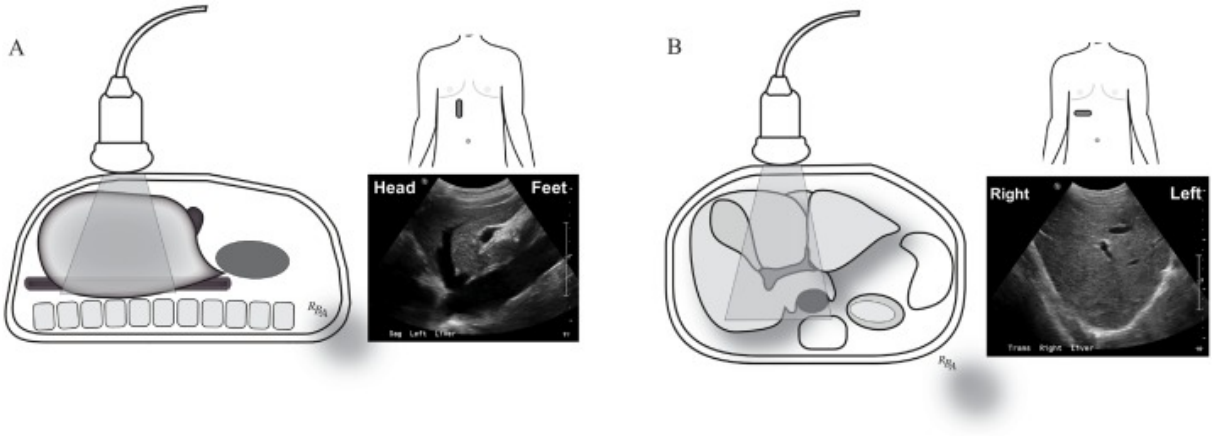
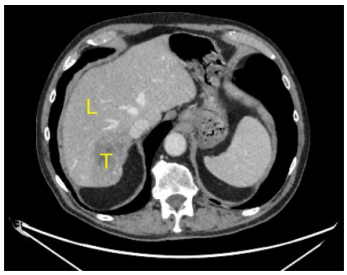
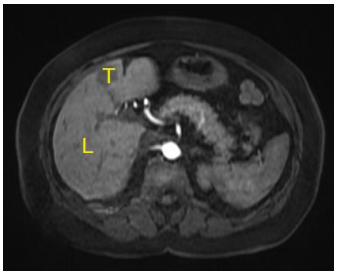


Figure 1.5: Greyscale abdominal sonograms with a limited field of view of the liver can be obtained from different positions and orientations of the probe pressed against the skin, from [Adams 2022]. Not shown: the probe emits sound waves into the body tissues and collects the reflected ones before to translate them into a greyscale image, with the assistance of a computer system.

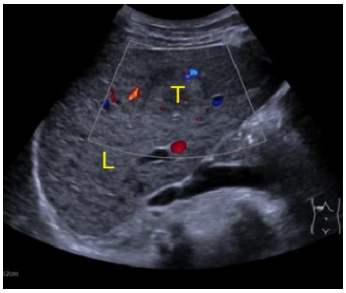


(a) CT-scan slice example.



(b) MRI slice example.

(c) Abdominal pulse-echo US sonogram example, adapted from [Jiang *et al.* 2023].



(d) Abdominal Doppler US sonogram example, adapted from [Jiang *et al.* 2023].

Figure 1.6: Preoperative images from different modalities and patients in similar planes. T and L respectively stand for tumour and liver parenchyma.

the skin and the transducer. There may be issues with imaging of the liver via ultrasound, as echogenicity (brightness) of the liver may be confounded by fibrosis, inflammation, and other features of chronic liver disease. For diagnosing HCC, abdominal ultrasound has low sensitivity and specificity. The liver may also not be entirely visualised due to shadowing from the ribs, gas, and other patient factors, such as patient habitus [Parra *et al.* 2023].

Moreover, ultrasound is operator-dependent. Alternatively, imaging techniques using the previous ones are explored, such as Magnetic Resonance Elastography, which computes the elasticity or stiffness of tissues through the measure of their motion caused by an external tissue exciter, using a specific MRI option, e.g. a phase-contrast technique. It could better characterise malignant tumours, so that the best treatment and surgical methods could be identified and applied. However, this is still in an exploratory phase and has numerous limits to overcome, such as a very low spatial resolution [Yang & Qiu 2021].

Apart from these imaging techniques, hepatic tumour diagnosis could be perfomed with biopsy and serum alpha-foetoprotein (AFP) biomarker measurement (for HCC). However, either MRI or CT-scan is performed in any surgery case, in order to locate the tumour and surrounding critical structures while accessing them during surgery.

1.2.4 Liver Surgery with a Focus on Resection

For the liver, resection (e.g. of parts enclosing tumours), also named hepatectomy, is the most common surgery and the focus of this section. However, other types of surgery exist, such as orthotopic liver transplantation. It is performed through open surgery, described in section 1.2.4.1. Orthotopic liver transplantation consists in replacing the diseased liver from a recipient patient with a healthy liver from a recently deceased donor. The procedure includes donor and recipient hepatectomies, the vascular and bile duct reconstruction and haemostasis [Makowka *et al.* 1988, Lladó & Figueras 2004].

For small tumours, thermal ablation, a non-surgical percutaneous treatment using radiofrequencies or microwaves, can also be performed. It consists in destroying the tumour cells and the surrounding ones (at least 1 cm of margin) by virtue of heat [Ryan *et al.* 2016]. A specific percutaneous needle is guided by imaging to deliver the used energy in the tumour location. Even though this is a non-surgical treatment, it usually follows the mini-invasive surgery process (section 1.2.4.2), for a better guidance and tumour localisation by means of intraoperative navigation techniques [Montalti *et al.* 2024], described in section 1.2.5. The only difference is that resection is replaced with thermal ablation.

In this section, the different surgery modes to access and operate the liver, including the minimally invasive ones, are outlined. All these surgery modes are performed under general anaesthesia and involve a surgical staff comprising surgeon, assisting surgeons, nurses and anaesthetists. Then, the different approaches to resect the liver parts where tumours are present are described, as well as the surgical landmarks which can guide them. Eventually, intraoperative navigation techniques to provide additional guidance such as tumour localisation are presented.

1.2.4.1 Open Surgery

The first Open Liver Resection (OLR) was reported in 1888 [Langenbuch 1888]. Open Liver Surgery (OLS) starts with a laparotomy, i.e. a surgical incision across the superior abdomen, below the rib cage, located and extended according to the surgery requirements [Gaujoux & Goéré 2011]. Retractors are used in order to maintain the opening along the surgery, see Figure 1.7a. The surgeon can then directly manipulate and palpate

the liver and the surrounding structures and use surgical instruments with few spatial constraints.

1.2.4.2 Minimally Invasive Surgery

Unlike laparotomy in open surgery, MILS starts with small surgical incisions (from 5 to 12 mm) through the abdominal wall performed by trocars and used to insert cannulas (small tubes) enabling the insertion of surgical tools and a laparoscopic camera. CO_2 gas is first insufflated in the abdominal cavity (pneumoperitoneum) through a trocar port in order to create surgical space and allow visualisation [Ikoma *et al.* 2015]. Overall, the patient benefits from this reduced invasiveness with a shorter postoperative hospital stay than in OLS, fewer post-operative complications, lower intraoperative blood loss, and faster postoperative functional recovery [Haney *et al.* 2021].

Minimally Invasive Liver Resection (MILR) is composed of Laparoscopic Liver Resection (LLR) and Robot-assisted Liver Resection (RLR). The first LLR was reported by Reich et al. in 1991 [Reich *et al.* 1991]. The first RLR (segmentectomy) was reported in 2003 [Giulianotti *et al.* 2003]. With the continuous development of laparoscopic devices and surgical techniques, the indications for laparoscopic and robot-assisted hepatectomy have expanded rapidly and are now very similar to those of open surgery [Sun *et al.* 2023].

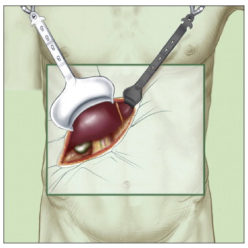
Laparoscopic liver resection. LLR can be performed with various patient positionings according to the surgeon preferences, the resection target location and size as well as the patient morphology. They include the supine position with split legs (French position), the reverse Trendelenburg position with legs apart, where the body is inclined from 10-30 degrees with respect to the previous one so that the feet are lower than the head, or left lateral (decubitus) position where the patient lies on the side. Some positions are more indicated for antero-lateral resections while others for posterior and superior resections such as the left lateral one [Thiruchelvam *et al.* 2021].

Port positioning follows the same principle. Generally, 4 to 5 working trocar ports [Kaneko *et al.* 2008] are placed strategically to optimise manipulation and mobilisation of the liver [Koffron *et al.* 2006].

Pneumoperitoneum is established through a 12-mm port, leading to between 10 and 15 mmHg of pressure in the abdominal cavity. This pressure is higher than the normal portal blood pressure of 6–10 mmHg and is therefore capable of reducing portal blood flow and alterations in hepatic function [Jin *et al.* 2021].

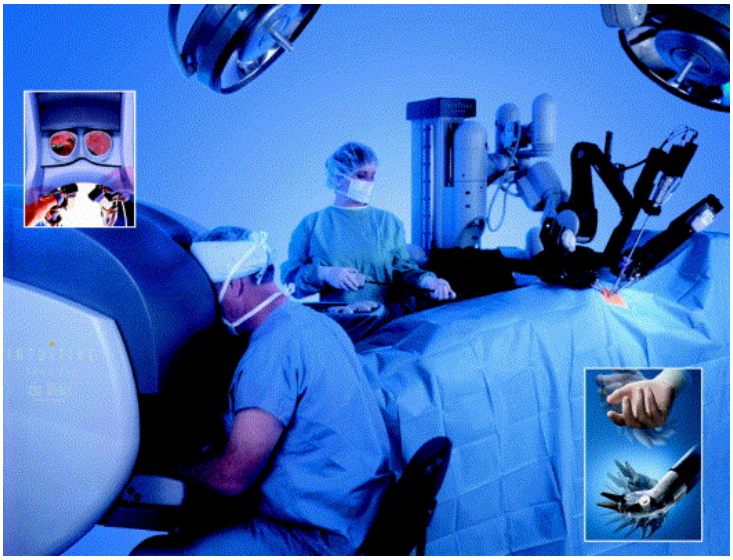
A laparoscopic/endoscopic optical system is inserted through another 12-mm port. Flexible 0-degree laparoscopes (long, flexible tubes with attached monocular camera and light) are mainly used as they allow the 2D visualisation of the different structures of interest in LLR [Yoh *et al.* 2019]. Basic components of the imaging systems include a laparoscope connected to a light source and a controller unit. The images are then transmitted through a monitor that allows the surgical team to visualise the operative field, as illustrated in Figure 1.7b. Examples of laparoscopic images are shown in Figure 1.2.

The other ports are used to introduce the surgical tools. There can be scissors, shears, staplers, sealers, with different energies, such as diathermy, microwave, and ultrasound [Kaneko *et al.* 2008]. They allow one to perform different surgical steps including



(a) Laparotomy in OLS, from [Gaujoux & Goéré 2011]. From the large incision, retractors maintain the large opening through which the surgeon palpates and operates the liver.

(b) Laparoscopic Liver Surgery (LLS), adapted from [Dogeas *et al.* 2021]. Small incisions allow the insertion of surgical tools, through trocar ports, from which the surgeon operates the liver. A monocular laparoscope/endoscope brings light and a camera inside the abdomen in order to allow the surgeon to visualise the projected intra-abdominal surgical scene through a monitor.



(c) Robot-assisted Liver Surgery (RLS), from [Desgranges *et al.* 2004]. The surgical system comprises a surgeon console and a surgical cart where surgical tools and the stereoscopic endoscope are inserted on robotic arms. Left inset, 3D display of the operative field and control handles of the robotic arms from the surgeon console; right inset, tool flexibility mimics human wrist one.

Figure 1.7: Surgery modes for liver resections

marking of the surface to transect, punctures, transection, vessel and bile duct coagulation, vessel clipping, removal of the resected part.

Main surgical complications comprise uncontrolled haemorrhage, biliary fistula (bile in an exteriorised fluid), incisional hernia (protrusion of internal tissue through the abdominal wall, at the incision sites) and gas embolism (gas into vascular structures).

Robot-assisted liver resection. RLR is similar to LLR except that the surgery is performed through robotic arms controlled by the surgeon from a remote console in the operating theatre, see Figure 1.7c. Surgeon tremors are filtered while robotic arms are highly flexible. The surgeon can also remotely control the endoscopic stereoscopic camera, which enables the visualisation of the surgical field through left and right images and thus in 3D. The assisting surgeon remains at the patient's side to change robotic instruments and perform assistive tasks such as stapling through dedicated ports [Bhogal *et al.* 2019].

Compared to LLR, these RLR characteristics increase the comfort of the surgeon and provide additional features such as 3D vision (depth perception) and increased dexterity. The major disadvantages are its high cost and the congestion of the operating theatre caused by all the required equipment. In complex hepatectomies, e.g. with large tumours or proximity of tumour to vital vascular structures, RLR should be performed by highly experienced surgeons [Liu *et al.* 2023].

1.2.4.3 Anatomical and Non-Anatomical Resection Approaches

The Couinaud segments are important in hepatic surgery as they allow a viable anatomical liver resection due to segment functional independence, see section 1.2.1. Indeed, anatomical resections consist in removing the hepatic segments where the tumours are present, resecting both the appropriate hepatic venous drainage, the associated portal venous blood supply and the hepatic arterial one [Nevarez & Yopp 2021], together with the biliary drainage. Nomenclature for anatomical liver resections, based on Couinaud segments, is standardised [Strasberg *et al.* 2000]. The name of the resection is based on which segment or combination of segments is resected, e.g. Figure 1.8. For instance, hemihepatectomy is the resection of the four left or right segments, extended hemihepatectomy removes additional segments, while segmentectomy, bisegmentectomy and trisegmentectomy are respectively the resection of only one, two, and three segments.

Recently, 'New World' terminology was introduced in order to cope with the absence of terms for other types of resection, e.g. non-anatomical resections [Nagino *et al.* 2021]. This non-anatomical alternative approach involves a reduced parenchymal resection, illustrated in Figure 1.8, and therefore also named parenchymal sparing resection [Botea *et al.* 2022]. It is not based on the drainage and blood supply of the anatomical location of the tumour but aims to obtain coarse negative surgical margins [Nevarez & Yopp 2021]. Note that a negative surgical margin is obtained when no cancer cells are present at the edge of the resected tissue, suggesting that all of the cancer has been removed. In this new terminology, the hepatectomy H is followed with a number for designing which segments are resected, such as H46 for a bisegmentectomy 4,6, while a number followed with the prime character is associated to a non-anatomical resection, e.g. H4'6' for a wedge resection in the segment 4 and 6.

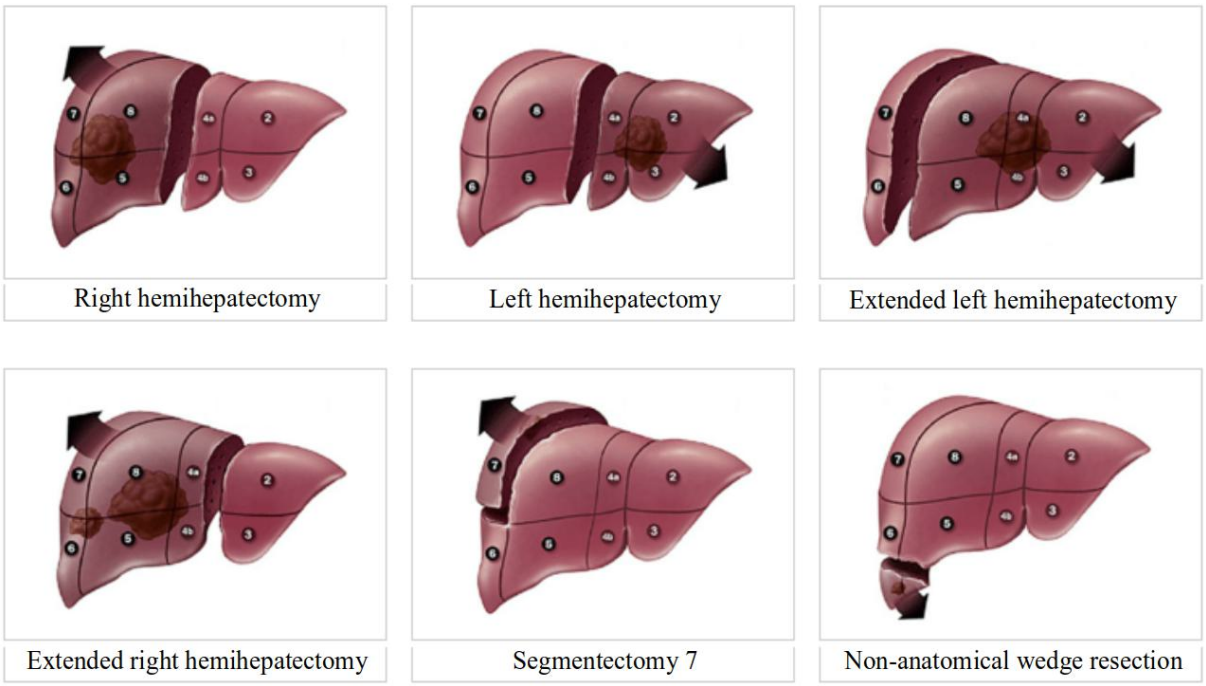


Figure 1.8: Examples of anatomical and non-anatomical resections, adapted from [Bhogal *et al.* 2019].

A consensus on which approach should be undertaken in early-stage tumours is not yet formed [Nevarez & Yopp 2021], so both resection approaches are used. Minimally Invasive Anatomical Liver Resection (MIALR) and Minimally Invasive Non-Anatomical Liver Resection (MINALR) stand for the two alternatives.

When a part of the liver is resected, the remnant can maintain liver functions and regenerate to a great extent under certain conditions. Liver resection should be considered when negative surgical margins and an adequate future liver remnant, with preserved arterial, portal venous, and hepatic venous flows as well as a preserved biliary drainage, can be achieved [Margonis *et al.* 2018].

1.2.4.4 Surgical Landmarks

The fundamental surgical landmarks during MIALR described by expert consensus guidelines [Gotohda *et al.* 2022] include a demarcation line of the region of interest on the liver surface, the root of major hepatic veins and intersegmental veins. The demarcation line can directly be visualised on the liver surface performing either the occlusion of the vascular supply of the region of interest, causing the modification of its colour, or the injection of a staining dye into the portal branch of the tumour-bearing segment [Felli *et al.* 2020]. The veins can be exposed using specific surgical and dissection techniques to avoid their bleeding [Ban *et al.* 2021]. These landmarks thus usually require a manipulation of the liver and some surgical gestures while in some cases product injections are needed. MINALR need alternative means to identify and visualise tumours due to no clear landmarks [Gotohda *et al.* 2022].

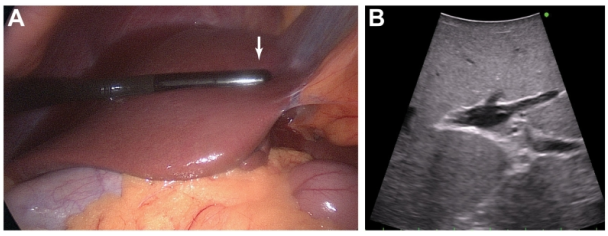
1.2.5 Intraoperative Navigation Techniques for Tumour Localisation

These alternative means are navigation techniques. Navigation in surgery can refer to different aims such as determining the anatomical or non-anatomical target position or determining a safe surgical route to reach the target [Mezger *et al.* 2013]. In MILS, it may refer to intraoperatively locate some inner structures such as the tumour, the blood vessels and the bile ducts involved in the resection.

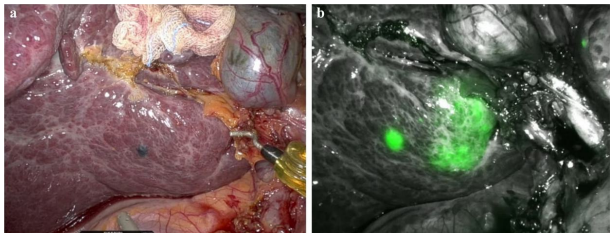
Like abdominal US, IntraOperative UltraSound (IOUS) uses a computer system, connected to a probe, on a mobile cart with a monitor for visualising US images. However, IOUS uses a mini-invasive probe which can directly contact the liver surface, as shown in Figure 1.9a. It can thus benefit from the same US techniques, such as the fundamental (pulse-echo) and Doppler ones [Chu *et al.* 2023] described in section 1.2.3, allowing one to locate the inner structures of interest. In addition, elastography, see section 1.2.3, can also be performed through IOUS, by measuring tissue displacement via probe palpation (strain elastography) or displacement induced by shear waves, being complementary to fundamental IOUS [Chu *et al.* 2023]. The use of IOUS at operative time and its direct contact to the liver surface also allow one to identify new tumours previously unseen on preoperative imaging [Rodrigues *et al.* 2017]. It is therefore essential to MILS and now prescribed in surgical guidelines [Hilal *et al.* 2018]. Its overall diagnostic benefits for adequate intraoperative surgical strategies, besides CT and MRI, are confirmed in some studies [van der Steen *et al.* 2021]. They are mainly due to its higher spatial resolution and its intraoperative real-time imaging features. However it comes with several limitations, including a limited field of view, a low signal-to-noise ratio, shadowing, reflection artifacts, and a variable contrast. In addition, determining the ultrasound image orientation with respect to the mini-invasive camera is challenging [Langø *et al.* 2012]. Moreover, a complete overview of the liver using IOUS is not feasible due to the trocar positioning and the limited IOUS movements.

Another developing navigation system is based on IndoCyanine Green (ICG) Fluorescence Imaging System (FIS). ICG is a water-soluble fluorescent dye (nontoxic at low doses) whose molecules absorb near-infrared light, and emit near-infrared light when relaxing. After intravenous injection, ICG remains fixed in tumoural hepatocytes (HCC) and underactive hepatocytes, particularly present around non-hepatocellular tumours (CRLM), while disappearing from healthy hepatocytes through bile excretion within a few hours [Branch 1982]. FIS also comprises a mobile cart with a monitor for visualising fluorescence images (illustrated in Figure 1.9b), connected to an endoscopic camera constituted of a laser and a sensor. It can excite ICG fluorescence through a laser emission (infrared radiance) over the operative field, and capture it by a real-time camera sensor, which filters the near infrared wavelengths, in the non-visible spectrum. The features of the camera allow the real-time detection of hepatocellular (tumour fluorescence) and non-hepatocellular tumours (peri-tumoural fluorescence), as well as bile ducts [Giorgio *et al.* 2018]. This enables the detection of new and small tumours not diagnosed by preoperative imaging. However, the near-infrared light can only penetrate 5 to 10 mm of tissue, only allowing one to detect tumours close to the liver surface ($<$8 mm) [Kudo *et al.* 2014]. Other disadvantages include a need of repeated injections or a temporal clamp of the hepatic artery in order to reduce washout of the dye and contin-

uously visualise the elements of interest. Meanwhile, a small amount of ICG circulates through the body after injection, which eventually stains the entire liver without specific manoeuvres [Giorgio *et al.* 2018]. Another use of ICG-FIS is for the identification of hepatic segments after injection of ICG in the portal vein of the target segment, located using IOUS. However, this is technically difficult in MILS [Ishizawa *et al.* 2016].



(a) From [Adams 2022]. IOUS probe is pressed against the liver surface (A) and captures the sonogram B.



(b) From [Rompianesi *et al.* 2023]. Left. Mini-invasive image captured by a laparoscope camera. Right: Similar image captured by an endoscope camera from an ICG-FIS. In this case, it allows one to identify HCC and a biliary cyst.

Figure 1.9: Images from the main intraoperative navigation techniques.

Hybrid operating theatres are also envisaged to provide surgical navigation. They are equipped for both laparoscopic surgery and intraoperative radiologic imaging, e.g. angiography machine in order to visualise blood vessels. In particular, C-arm cone-beam CT is used. The X-ray source and detector (see section 1.2.3) are opposite in a C-shaped arm (C-arm), while the patient lies on a table between both. They allow volumetric data acquisition in a single rotation of the C-arm [Orth *et al.* 2009]. However, it uses cone-shaped beam projection, in contrast to a fan-shaped beam in conventional CT (see section 1.2.3), which degrades image quality (lower signal to noise ratio, increased artifacts and inaccuracies in CT calculations) [Raj *et al.* 2013]. Blood vessels can be highlighted in CT angiographic images thanks to the injection of a contrast agent. However, tumours and other intrahepatic structures are not directly discernible. Radiopaque markers can be implanted as fiducials related to the tumour in order to locate it after CT image acquisition [Falkenberg *et al.* 2022]. They need to rely on another navigation system such as IOUS for guiding the implantation and an angiographic microcatheter for performing it. These hybrid rooms allow one to display simultaneously angiographic and laparoscopic images. However, works related to the hybrid rooms have main limitations caused by the radiologic imaging material which requires the surgeon to be equipped with specific radiation protections and the patient to be exposed to supplementary radiations. In addition, for estimating the depth of the tumour, the C-arm must reach positions which disturb the laparoscopic workflow [Falkenberg *et al.* 2022].

1.3 Augmented Reality for Minimally Invasive Surgery Navigation

In this thesis, we deal with MILS, due its numerous advantages over OLS, see section 1.2.4.2. In particular, we focus on a specific navigation technique for assisting MILS, i.e. Augmented Reality (AR). Its principle is first introduced, then the motivation for using and automating it in this surgical context is described. Eventually, the manual AR pipeline from which this thesis project starts is overviewed.

1.3.1 Principle

AR is part of extended reality, which integrates or replicates the real world with a corresponding digital version [Mendoza-Ramírez *et al.* 2023]. Virtual reality completely immerses a user inside a synthetic environment, hiding the real world. In contrast, AR supplements reality with virtual objects superimposed upon the real world [Azuma 1997]. They are usually not initially perceptible. This can be valuable in many fields, including gaming, education, industry, healthcare, medicine. Virtual objects can indeed be of any type, such as imaginary animals, past ones, assembling parts, human bodies and organs, in different imaging contexts [Mendoza-Ramírez *et al.* 2023]. Several application fields can benefit from the AR of a single virtual object, e.g. healthcare, fitness technology and virtual retail ones for the human body.

In surgical context, AR could be used as a visualisation and training aid for surgery [Azuma 1997]. Its application is explored in several fields such as neurosurgery, orthopaedics, spinal surgery, and oncology. In the MILS context, AR consists in augmenting 2D mini-invasive liver images by means of superimposed 3D digital guiding information and can also be considered as a navigation technique. It comprises three general steps:

- The acquisition of the requested guiding information, and its 3D reconstruction if applicable. This step can require another imaging technique, preoperative or intraoperative, in order to extract the requested information through image segmentation, which then requires to be reconstructed in 3D. Alternatively, the guiding information can be directly extracted from the mini-invasive images. This step determines the intrinsic quality of the requested information.

- The 3D/2D registration, i.e. the computation of the position of the 3D requested guiding information in the 2D mini-invasive image. This requires one to retrieve the endoscopic camera projection parameters and the deformation or displacement field of the requested information from the acquisition imaging to the mini-invasive one. This is the most complex and critical step.

- The rendering and visualisation, i.e. the way of displaying the projected 3D guiding information on the mini-invasive image. This is important for perceptual guidance and includes representation choices such as colour, transparency, shading and depth ones.

The AR digital information superimposed on mini-invasive liver images for assisting the surgery can be various, and obtained from preoperative or intraoperative imaging.

(a) Augmentation of hepatic and portal vein branches of a porcine liver, from [Teatini *et al.* 2019].

(b) Augmentation of tumour and resection path with surgical margin, from [Espinel *et al.* 2024].

(c) Augmentation of US intraoperative image position and orientation [Rabbani *et al.* 2022].

Figure 1.10: AR examples in MILS. a) and b) display registered 3D preoperative information while c) displays registered intraoperative information.

It can be 3D intrahepatic elements of interest, such as tumours and blood vessels (see Figure 1.10a) and even Couinaud segments. A 3D resection path with surgical margins can also be suggested, when trocar port position is retrieved along with the tumour one [Espinel *et al.* 2024], see Figure 1.10b. In addition, the 3D position and orientation of an image from another navigation technique, such as IOUS (Figure 1.10c), can be valuable [Kalantari *et al.* 2024].

1.3.2 Motivation

Although combining several digital information would provide a greater assistance in MILS, each one is subject to specific challenges, which require dedicated works. In this thesis, we focus on intrahepatic structure information obtained from a preoperative imaging (CT-scan or MRI) acquisition. Intrahepatic structures, such as tumours and blood vessels, are of high importance for both anatomical and non-anatomical surgeries, see section 1.2.4.3. Information brought by preoperative images (volume) has many advantages over one obtained from IOUS or other navigation imaging techniques. The first is related to the image quality, with a better contrast, signal-to-noise ratio, and reduced artifacts [Langø *et al.* 2012], which eases the localisation and the segmentation of the structures of interest. The second is its quasi-completeness. Indeed, tumours, veins and even the liver surface can be segmented from the preoperative volume and reconstructed in relation to each other, unlike IOUS which can only display a reduced field of view and therefore very partial views of the structures. The falciform ligament as well as small and new tumours are exceptions, due to the CT-scan or MRI resolution limit and the preoperative acquisition. Nonetheless, this preoperative feature is an advantage with regard to the processing time and load in computer-based AR assistance. Indeed, some tasks of the AR process, e.g. segmentation and reconstruction, could be performed prior to surgery, therefore releasing intraoperative processing time and load. Moreover, unlike hybrid rooms (see section 1.2.5), it does not provide additional radiations to the subject and the surgeon. However, there exist some disadvantages, mainly related to the deformation of the preoperative hepatic structures between acquisition and surgery times, due to the pneumoperitoneum, respiration and manipulation through tools, which require one to retrieve it.

Without a navigation system, the surgeon should transfer the position of the requested elements, obtained from the preoperative 3D imaging, to the intraoperative 2D image, i.e. performing 3D-2D registration, which is a very complex task. Indeed, the surgeon should cope with the difference of representation between imaging methods, the scene projection with partial views and occlusions, and the deformation between both stages. Even with the assistance of an image-based navigation system such as IOUS, it can be difficult for the surgeon to locate the actual intrahepatic targets defined preoperatively while comprehending their surrounding critical structures. Assisting the surgeon in this 3D/2D registration task, illustrated in Figure 1.11, is the main motivation of this thesis. Acquisition and reconstruction as well as rendering and visualisation tasks are beyond its scope. More precisely, the objective of the thesis is to automate the 3D-2D registration of preoperative hepatic structures to mini-invasive liver images, in order to facilitate the assistance of MILS by these augmented structures.

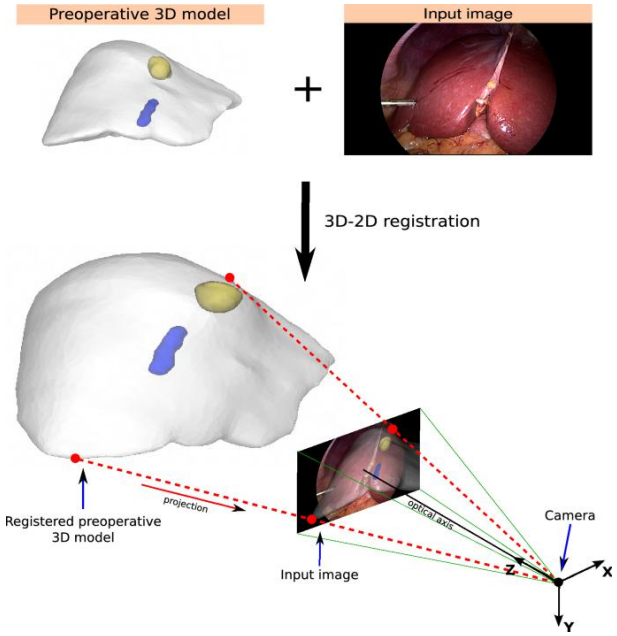
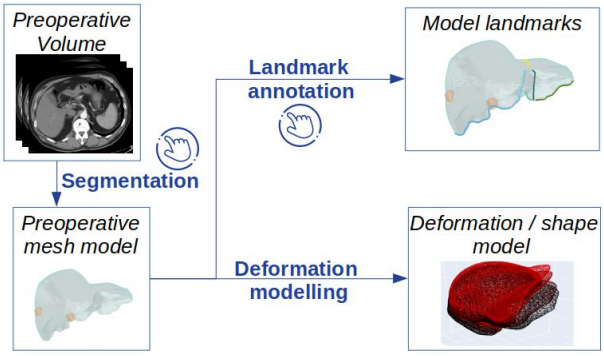


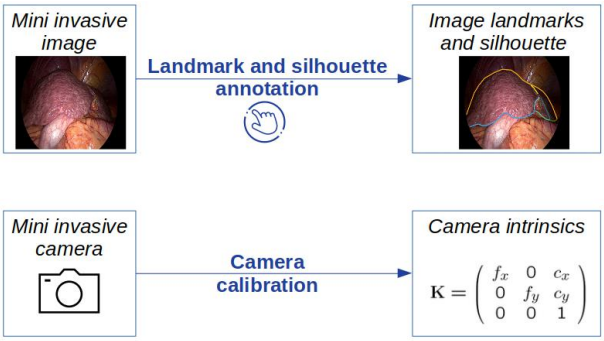
Figure 1.11: 3D/2D registration of a preoperative model to a mini-invasive image, from [Espinel *et al.* 2020]. The preoperative model should be displaced and deformed so that its projection to the image plane is consistent with image information while complying with its deformation parameters.

1.3.3 Registration Baseline

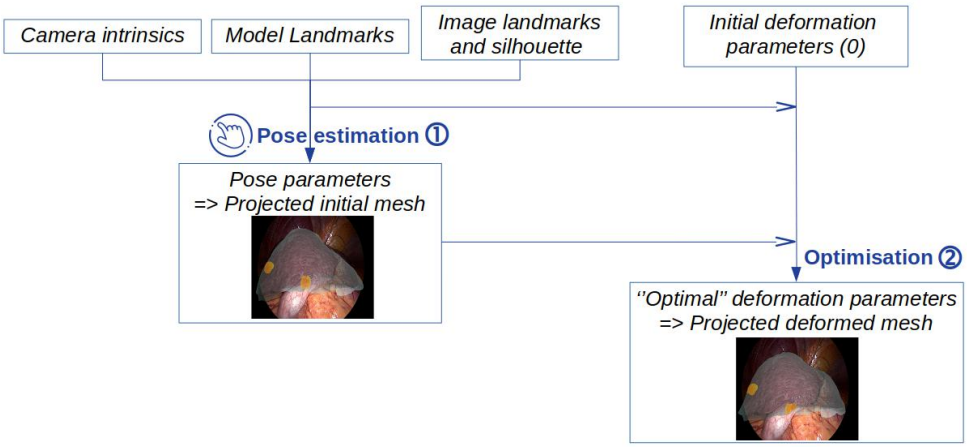
In order to automate the 3D/2D registration, we start from an existing Patient-Specific (PS) pipeline incorporating manual steps. It was built in successive stages from several works, such as [Plantefeve *et al.* 2014, Koo *et al.* 2017b, Özgür *et al.* 2018]. It makes use of 3D-2D corresponding liver surface landmarks for guiding the registration. It assumes that mini-invasive global views of the liver, meaning views where most of the landmarks are visible, are captured. This assumption makes it more adapted to navigation before the start of the liver resection. Indeed, during resection, the surgeon may need to zoom on the area to remove while landmark occlusion by blood, tools, gauze is more likely. This



(a) Preoperative steps. First, the preoperative volume obtained from CT or MRI is segmented in order to reconstruct 3D meshes of structures of interest including the liver and tumours. Then their deformation is jointly modelled and the liver mesh is annotated with surface landmarks.



(b) Intraoperative preparation steps. The liver surface landmarks corresponding to the visible 3D ones in the mini-invasive image are annotated. Calibration of the endoscopic camera is also performed in order to get the camera projection parameters (intrinsics).



(c) Intraoperative registration steps. The translation and rotation (pose) of the preoperative model allowing one to superimpose corresponding 3D/2D landmarks according to the camera intrinsics is first estimated. From this pose and with the same aim, an optimisation of the deformation model parameters is performed.

Figure 1.12: 3D/2D baseline registration pipeline based on 3D/2D corresponding liver surface landmarks.

pipeline does not use sequential information and only processes a still image each time, enabling its application to every imaging devices, e.g. monocular and stereoscopic cameras. However, this does not prevent one to use and combine information from multiple views and registrations.

It is composed of both preoperative and intraoperative stages, see Figure 1.12. In the preoperative stage illustrated in Figure 1.12a, the 3D preoperative hepatic elements of interest are reconstructed from the manually segmented volume of the patient, obtained from CT-scan or MRI. They can include the liver, tumours and veins. Both are represented as surface meshes while the liver also involves a volumetric version. 3D liver surface landmarks are manually annotated. In addition, liver deformation or shape is automatically modelled with respect to the preoperative mesh. Inner structure deformation is modelled conjointly.

In the intraoperative stage, preparation steps are required, see Figure 1.12b. First, the surgeon explores the surgical field and selects adequate camera parameters in order to visualise a global view of the liver in a clean way. The endoscopic camera is calibrated in order to obtain these projection parameters (intrinsics). A mini-invasive global view of the liver is then captured with these parameters and the 2D liver surface landmarks corresponding to the 3D ones and visible on the image are manually annotated.

The 3D/2D registration of the liver can then be performed, see Figure 1.12c. The deformation field is decomposed into pose parameters (i.e. translation and rotation from the preoperative to the intraoperative space and time) and deformation parameters from the deformation model. A rigid 3D/2D registration of the preoperative surface model, i.e. estimation of the pose parameters, is first performed manually in order to attempt to superimpose corresponding landmarks on the augmented image, using the retrieved camera intrinsics. Eventually, from the estimated pose, an automatic algorithm performs deformable 3D/2D registration, through an optimisation of the deformation parameters whose initial values are zeros (no deformation). The requested inner structures are directly retrieved from the registered liver thanks to the joint deformation modelling.

1.4 Thesis Overview, Organisation and Contributions

The baseline registration pipeline of a preoperative 3D model to a 2D mini-invasive image, described in section 1.3.3, comprises several steps. Each step not explored in the other chapters is first described in detail in chapter 2. Its current limits are also highlighted and the future works required to overcome them are suggested. This thesis focuses on automating the registration pipeline in two ways:

- Chapters 3 and 4. The first way maintains the same pipeline and automates the manual intraoperative steps [Labrunie *et al.* 2022], i.e. the 2D liver landmark annotation and the rigid 3D/2D registration (pose estimation):
  - Automatic annotation of the 2D liver landmarks is obtained through an encoder-decoder Neural Network (NN) dedicated to image segmentation, which inputs mini-invasive images and outputs the position of the landmarks through segmentation masks. Several network architectures are explored in chapter 3.

A fully attention-based network obtains the best performance in three different mini-invasive image datasets. We also propose to use information from other images and even their corresponding segmentation masks, which highly facilitates the segmentation of a single one through a specific architecture and attention mechanism between associated data, and substantially improves segmentation performance.

  - Automatic pose estimation follows an iterative process which refines the pose according to the landmark visibility of the translated and rotated preoperative liver of the previous iteration. The pose is initialised with the assumption that the whole 3D landmarks corresponding to the 2D ones are visible. In each iteration, the problem is formulated as a Perspective-n-Point (PnP) problem. It consists of estimating the pose of the liver given the camera intrinsics and a set of n 3D points in the world and their corresponding 2D projections in the image, obtained from the visible landmarks. This process is detailed in section 4.3 and can take only a few seconds. This method performs better than manual and state-of-the-art ones on data acquired with tumour ground truth.

- Chapters 4 and 5. The second way automates both rigid and deformable 3D/2D registration through an alternative pipeline based on deep learning [Labrunie *et al.* 2023]. This alternative pipeline is also extended for allowing Patient-Generic (PG) registration [Labrunie *et al.* n.d.]:

  - PS registration employs an encoder-regressor NN initially developed for human pose and shape parameter recovery from 2D natural human-centred images, using 3D-2D correspondences of joint points. The adaptation consists in replacing 2D natural human-centred images with liver landmark distance maps, while using 3D-2D landmark correspondences and the liver deformation model. The regressor inputs the encoder outputs and iteratively regresses the pose and deformation parameters. Training is achieved through the simulations of pose and deformation parameters as well as the associated landmark distance maps resulting from the projection of the moved and deformed preoperative liver. Section 4.4 describes this method named Liver Mesh Recovery (LMR). It performs on par with the previous state-of-the-art methods on validation tumour ground truth data, while processing in real-time. The computation load due to simulations and training is transferred to preoperative time and can take about a day.

  - PG registration reuses the same encoder-regressor NN architecture as in PS, but replaces the PS liver deformation model with a generic one. It is built as a generic kernel-based model, derived from a mean shape obtained through the shape registration and alignment of numerous patient meshes, incorporating anatomical surface point correspondences. The details of the generic liver modelling and the PG-LMR are given in chapter 5. A specific block also allows PS registration of the inner structures of interest. Despite an accuracy slightly lower than that of the state-of-the-art methods, this method facilitates the deployment of the LMR, requiring a single training of the network for all

patients while maintaining the intraoperative real-time processing. In addition, it can augment generic anatomical features from the generic liver model, and thus paves the way to PG anatomical AR.

Chapter 2

Background

Contents

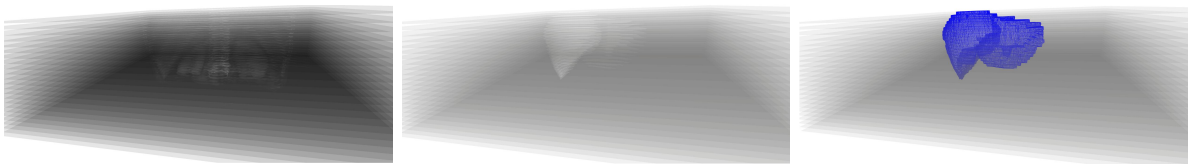
We describe the steps which are not the core of the selected baseline registration pipeline based on corresponding 3D/2D liver surface landmarks, but are still essential. They are introduced in section 1.3.3 and illustrated in Figure 1.12. They consist of both preoperative and intraoperative steps, namely volume segmentation and 3D mesh reconstruction of the liver and its inner structures of interest (sections 2.1 and 2.2), deformation modelling (section 2.3), mini-invasive camera calibration (section 2.4), as well as image selection and corresponding 3D/2D liver surface landmark annotation (section 2.5). We additionally introduce the principles of the deep neural networks which are the core tools of the proposed learning-based automation approach, in section 2.6.

2.1 Liver and Inner Structure Volume Segmentation

The preoperative volume segmentation of the patient/subject, see figure 1.12a, consists in annotating the structures of interest, including the liver and tumours. It can be performed manually by a surgeon or a person experienced in radiology, using open-source solutions such as 3D Slicer or MITK. These tools allow the user to skim through the slices of the patient along the 3 orthogonal axes of the *world* (preoperative imaging) coordinate system. They also allow the manipulation of the brightness and the contrast of the images for easing the structure detection and localisation, while providing an annotation tool of the slice voxels. $V \in \mathbb{R}^{c \times h \times w}$ represents the preoperative volume with the number of slices c, the height h, and the width w. A binary segmentation mask volume of the same dimension as the preoperative volume $M \in \{0, 1\}^{c \times h \times w}$ can be obtained for each structure of interest, as illustrated in figure 2.1b. Nonzero voxels indicate that the structure of interest occupies these voxels.

The segmentation can also be achieved automatically through deep learning, which has risen in the last decade. It consists in training multiple layers of neurons which relate inputs to outputs in order to find the neural parameters which minimise a loss function, see section 2.6. At inference, the network predicts outputs using the optimised parameters. For this task, the inputs are usually 2D or 3D preoperative images, the outputs are the corresponding 2D or 3D segmentation masks, and the NN relies on an encoder-decoder architecture [Sengun *et al.* 2021, Affane 2022, Song *et al.* 2024]. The encoder downsamples and transforms the inputs into encoding features and the decoder inputs the encoding features and possibly additional information from the encoder then decodes and upsamples them to form the segmentation mask. The general encoder-decoder NN architecture as well as the most standard ones for medical images such as the UNet are reviewed in chapter 3. In addition, relevant loss functions for training the NN are described.

A benchmark [Bilic *et al.* 2023] allows a comparison of numerous networks for the segmentation of liver and tumour tasks on the same datasets. Deep learning is also used for attempting to segment the preoperative volume semi-automatically, i.e. interactively. For instance, an encoder-decoder architecture can be embedded in a interaction loop with a user feedback memory, from which the features can be learned [Zhou *et al.* 2023, Mikhailov *et al.* 2024].



(a) Slices enclosing the liver, from preoperative imaging

(b) Liver segmentation on the slices

(c) Reconstructed liver surface (blue) from the segmentation, using marching cubes

Figure 2.1: Segmentation and reconstruction principles

2.2 3D Liver and Inner Structure Mesh Reconstruction

This preoperative step aims to obtain a relevant representation of the structures of interest (liver and inner structures) for enabling or easing other steps of the pipeline, see figure 1.12a. A large geometric domain can be represented by small discrete elements, combined to form a mesh. Elements in a surface mesh are planar polygons called faces. They are enclosed by edges (lines) connecting vertices (points). Even complicated smooth surfaces can be approximated as a collection of planar polygons. 3D surface meshes are the most common representation for rendering, in computer graphics [Pajarola 2000]. In particular, 2D simplices (triangles), i.e. the simplest possible polygons in 2D, are used as faces. Non-degenerate triangles are guaranteed to be planar and each rendering attribute of a triangle, such as depth and shading, can take a single value. Graphics hardware is optimised for fast processing of triangle meshes due to their simplicity, compactness and rendering efficiency [Pajarola 2000].

Volumetric meshes are commonly used to compute solutions of partial differential equations, e.g. for volumetric deformation induced by specific loading and initial conditions. It partitions space into 3D cells over which the equations can be solved, which then approximates the solution over the larger domain. 3D simplices (tetrahedra), i.e. the simplest possible polyhedra in 3D, are also commonly used as cells, even though hexahedral elements are another alternative.

In order to obtain adequate liver surface and volumetric meshes, the surface mesh is first reconstructed from the segmented preoperative volume (section 2.2.1), smoothed and resampled (section 2.2.2), and then cleaned (section 2.2.3). Eventually, the volumetric mesh can be reconstructed from the surface one (section 2.2.4). Inner structure surface meshes are reconstructed similarly to the liver one. However, they are expressed relatively to the liver volumetric mesh (section 2.2.5).

2.2.1 Surface Mesh Reconstruction through Isosurface Extraction

Once the annotation is performed on all the slices containing the structure of interest, its triangular surface mesh can be reconstructed from the occupancy field M, see figure 2.1c, using marching cubes [Lorensen & Cline 1998] (this option is also directly present in the open-source solutions). This method consists in representing the volume through cubes (e.g. groups of $2 \times 2 \times 2$ neighbouring voxels) containing 8 binary vertices: either they

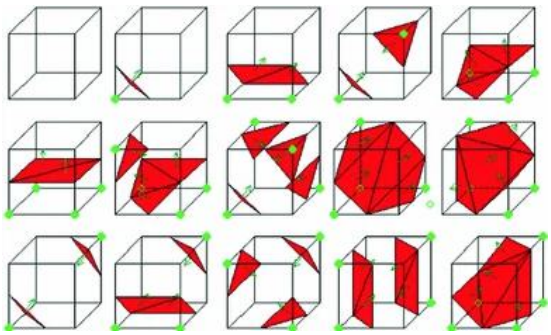
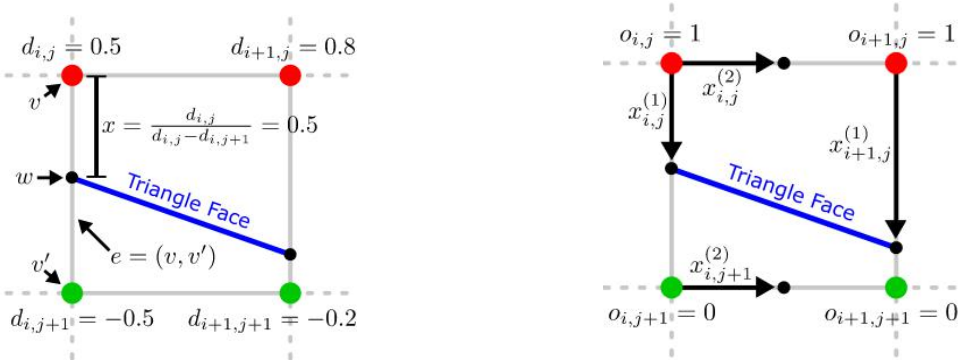


Figure 2.2: The 15 initial patterns of the marching cubes, from [Cirne & Pedrini 2013]. The vertices classified inside the surface are green. When the 8 vertices of the cube are all inside or outside the surface, this corresponds to the first (top left) pattern.



(a) Initial marching cubes

(b) Differentiable marching cubes

Figure 2.3: From [Liao *et al.* 2018]. Representation used by Marching Cubes (a) and the Differentiable Marching Cubes proposed in [Liao *et al.* 2018] (b). The former uses an implicit surface representation based on signed distances D while the latter exploits an explicit surface representation which is parameterised in terms of occupancy probabilities O and vertex displacements X.

are inside (or on) the surface of the structure of interest (e.g. ones) or outside (e.g. zeros). The aim of this method is to extract the isosurface of the structure of interest, which is the level set $f(x, y, z) = c$ such that $(x, y, z) \in \mathbb{R}$, while representing it explicitly as a mesh. c can equal 0.5 when f represents the binary occupancy function, and 0 when f represents the distance from the surface. A cube has $2^8 = 256$ possible configurations of vertex values, where many are symmetric and can be grouped in 15 patterns. Cube vertices are related through 12 edges. Intersection of the surface of the structure of interest is performed along the cube edges and is interpolated linearly, which results in surface vertices. A piecewise linear surface, through triangular faces, is used to join these vertices. There is no intersection when the 8 vertices of the cube are all inside or all outside the surface. Otherwise, between 1 and 4 triangles are built for a pattern, see Figure 2.2. For the patterns where the surface reconstruction is ambiguous, i.e. multiple triangular faces are possible, the simplest reconstruction is selected. A look-up table can thus be created and indicates how to build the triangular surface inside a specific cube configuration.

Marching cubes is the standard procedure for mesh reconstruction through isosurface extraction but alternatives exist. They mainly include extensions of marching cubes.

They can extend vertex-based look-up table to prevent cracks at the boundaries between neighbouring chunks that differ in level-of-detail, such as transvoxels [Lengyel 2010]. In contrast, flying edges [Schroeder *et al.* 2015] use an edge-based look-up table and optimises the parallelism of the computation in order to improve its speed. The surface net method [Gibson 1998] takes the same cube vertex representation as marching cubes, but initialises a net by placing a single node at the centre of each surface cube (whose pattern is different from the first one in marching cubes) and by connecting nodes from adjacent cubes through edges. Once the surface net has been defined, nodes are displaced to iteratively reduce a constrained energy measure related to the edges, in order to smooth the surface while constraining each node to remain inside its original surface cube.

More recently, deep learning has been attempted to tackle this isosurface extraction problem. However, the marching cubes algorithm is not differentiable with respect to topological changes. Deep Marching Cubes [Liao *et al.* 2018] proposes an alternative differentiable formulation where a marching cube (topology) pattern is represented by a binary tensor $T \in \{0,1\}^{2\times2\times2}$ and its probability is the product of 8 occupancy probabilities at its corners. In this aim, an encoder-decoder structure is used and the decoder outputs by two heads both the occupancy probability field $O \in [0,1]^{N\times N\times N}$ and the vertex displacements $X \in [0,1]^{N\times N\times N\times3}$, representing the displacements of the triangle vertices along their associated cube edges, see figure 2.3.

2.2.2 Surface Mesh Smoothing, Resampling or Coarsening

The number of vertices m and triangular faces can be very high, depending on the preoperative imaging resolution, which is usually quite high. Mesh smoothing, decimation, coarsening or resampling can be performed in order to reduce this number. Mesh smoothing consists in filtering the high-frequency surface noise. Laplacian mesh smoothing can be used, for instance from VTK [Schroeder *et al.* 2006]. Considering the vertex coordinates $V \in \mathbb{R}^{m\times3}$, for each vertex i, the n neighbours $V^i \in \mathbb{R}^{n\times3}$ in the graph formed by the mesh are found. Then, its coordinates are modified iteratively according to a weighted average of the connected vertices, with associated weights w and a relaxation factor α. At each iteration, the new vertex coordinates $\vec{\hat{v}}_i$ can be obtained from the current coordinates of $\vec{v}_i$ and its neighbours:

$$\vec{\hat{v}}_i = \vec{v}_i + \alpha \left(\frac{1}{\sum_{j=1}^{n} w_{ij}} \sum_{j=1}^{n} w_{ij} \left(\vec{v}_j^{\,i} - \vec{v}_i \right) \right) \quad \forall\ i \in \{1, ..., m\}$$

The simplest smoothing case uses the uniform Laplacian with $w_{ij} = 1$. One of the disadvantages is the mesh shrinkage after few iterations. Other smoothing methods are variants of the Laplacian smoothing with different ways of computing the weights, such as mean curvature flow using the cotangent Laplacian [Desbrun *et al.* 1999], or different ways to iterate, such as Taubin smoothing [Taubin 1995] which alternates positive and negative relaxation factors. These variants can reduce the mesh shrinkage. An example of mesh smoothing is illustrated in figure 2.4a.

Resampling to a given number of vertices can then be performed with Approximated Centroidal Voronoi Diagrams (ACVD) [Valette & Chassery 2004], e.g. using pyacvd. Given the surface Ω, an open set of $\mathbb{R}^3$, any surface seed or site $\vec{s}$ from $S \in \Omega^{n\times3}$ and any

surface point $\vec{p} \in \Omega^3$, the Voronoi Diagram can be defined as n distinct regions R_i such that:

$$R_i = \{\vec{p} \in \Omega^3 \mid \|\vec{p} - \vec{s}_i\|_2 \leq \|\vec{p} - \vec{s}_l\|_2, \ \forall\, l \neq i\}$$

Each surface region R_i is approximated as the union of several mesh faces, and a triangle F_j is a part of only one region R_i. Each triangle F_j with vertex coordinates $V_j \in \mathbb{R}^{3\times3}$ is approximated by its centroid of coordinates $\vec{\gamma}_j = \frac{1}{3}\sum_{k=1}^{3} \vec{v}_j^k$, with a weight equal to its area ρ_j. A centroidal Voronoi diagram is a Voronoi diagram where each Voronoi site $\vec{s}_i$ is also the mass centroid of its Voronoi region R_i. Therefore, it is approximated as:

$$\vec{s}_i = \frac{\sum_{F_j \in R_i} \rho_j \vec{\gamma}_j}{\sum_{F_j \in R_i} \rho_j}$$

Building the ACVD can be achieved by minimising the energy E:

$$E = \sum_{i=1}^{n} \left(\sum_{F_j \in R_i} \rho_j \|\vec{\gamma}_j - \vec{s}_j\|_2 \right) = \sum_{i=1}^{n} \left(\sum_{F_j \in R_i} \rho_j \left\| \vec{\gamma}_j - \frac{\sum_{F_j \in R_i} \rho_j \vec{\gamma}_j}{\sum_{F_j \in R_i} \rho_j} \right\|_2 \right)$$

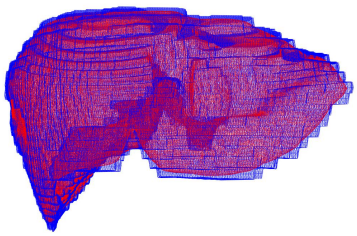
A specific convergent iterative algorithm updating the clusters according to tests on boundary edges is used for minimising the energy term, starting from an adapted initialisation. For each cluster determined by the algorithm, a mesh vertex is then set as the closest surface point from the cluster centroid. The Delaunay triangulation of the mesh vertices is then performed. It is the dual graph of the Voronoi diagram. This means that a vertex is created for each Voronoi region, and then triangles are formed by edges connecting the vertices of all adjacent Voronoi regions, except in specific cases. Figure 2.4b illustrates mesh resampling with ACVD.

Other mesh coarsening or decimation methods can use edge collapsing to coarsen a mesh. It consists in selectively eliminating triangle edges (or arbitrary vertex pairs) from the mesh in order to simplify it, until specified criteria are met, such as a desired face count or a maximum tolerable error [Garland & Heckbert 1997].

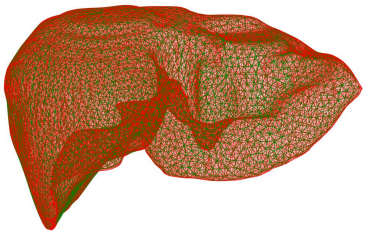
2.2.3 Surface Mesh Cleaning

The surface processed from marching cubes might include singular simplices, holes, intersecting or degenerate triangles. Volumetric mesh reconstruction from surface meshes requires manifold and watertight meshes without degenerate and intersecting elements. A watertight manifold mesh contains no holes or missing faces that would cause leaks into the interior of the shape's volume. Every edge in the mesh is manifold, i.e. part of exactly two faces. A cleaning [Attene 2010] from PyMeshFix can be performed in order to make it manifold, without degenerate or intersecting elements, see figure 2.4e. The algorithm attempts to modify the input mesh only locally within the neighbourhood of undesired configurations. First, topology is reconstructed:

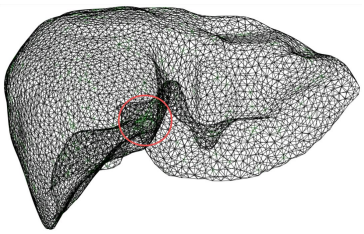
- Singular edges, i.e. adjacent to more than two faces, are identified. The singular vertices are duplicated and the cut is performed through singular edges for creating disconnected manifold surfaces. Then stitching maintains the surface as a manifold while joining boundary edges [Guéziec *et al.* 2001].



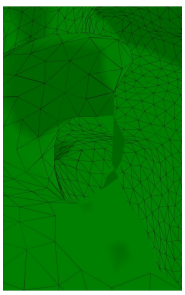
(a) Laplacian smoothed surface (red) from the reconstructed surface (blue)



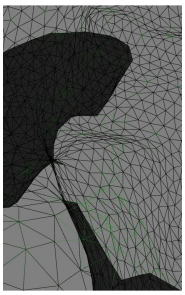
(b) Green surface obtained using ACVD from the Laplacian smoothed one (red)



(c) Cleaned surface (black) from the previous (green) one. The ellipse indicates the major cleaned part.



(d) Zoom in on the major cleaned part, the green mesh is the previous one.



(e) Zoom in on the major cleaned part, the black mesh is the cleaned one.

Figure 2.4: Mesh reconstruction processing

- Hole boundaries are identified and each hole is triangulated, refined and smoothed [Liepa 2003].

Then the geometry is iteratively corrected by removing growing neighbourhoods of undesired elements and by patching the resulting surface gaps until all the defects are removed:

- Degenerate triangles having a nearly flat angle are treated by swapping the edge opposite to such angle, while ones having a nearly null angle are removed by collapsing the edge opposite to such angle to its midpoint.
- Pairs of intersecting triangles are identified and removed.
- The remaining gaps are filled using a partial curve matching technique (geometric hashing) for matching parts of the defects and an optimal triangulation of the 3D polygons is performed for resolving the unmatched parts [Barequet & Sharir 1995].

2.2.4 Volumetric Mesh Reconstruction

The preoperative volumetric liver mesh is reconstructed using constrained Delaunay tetrahedralisation [Shewchuk 2002] from TetGen, illustrated in figure 2.5. It comprises 2 steps:

1. Constrained adaptive mesh generation. Tetrahedralisation is initialised from the input surface mesh triangles, which are connected by additional triangles to form

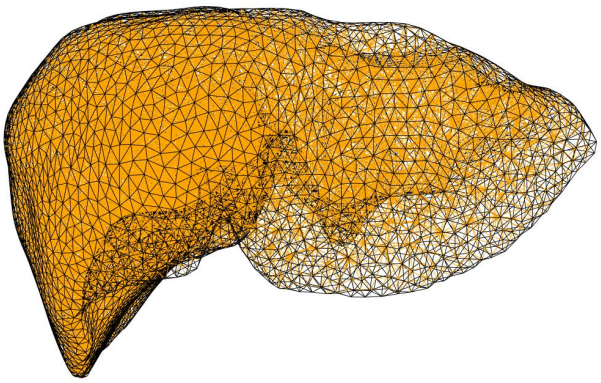


Figure 2.5: Reconstructed volumetric mesh (orange) from the cleaned surface (black). It maintains the simplices of the surface mesh.

tetrahedra. Inner (Steiner) points can be added in order to achieve a valid tetrahedralisation and an initial volumetric mesh quality with a small number of tetrahedra. This initial mesh quality is obtained thanks to an algorithm which provides guarantees on simplex shapes (bounded aspect ratio) and size (number) [Ruppert 1995].

2. Mesh refinement. At first, the tetrahedral mesh constructed in step 1 is refined by inserting new vertices in tetrahedra of poor quality (according to criteria), for splitting these simplices while maintaining some spatial properties [Shewchuk 1998]. Second, the mesh quality is further improved by using a local mesh optimisation scheme, combining vertex smoothing, edge/face swapping, edge contraction, and vertex insertion [Si 2013].

2.2.5 Inner Structure Representation

A barycentric coordinate system is a coordinate system in which the location of a point is specified by reference to a simplex, e.g. a triangle for points in a plane, a tetrahedron for 3D points. Coordinates of a point $\vec{x} \in \mathbb{R}^d$ belonging to a simplex whose vertex coordinates are $V \in \mathbb{R}^{n \times d}$ can be represented through barycentric coordinates λ as:

$$\vec{x} = \sum_{i=1}^{n} \lambda_i \vec{v}_i \text{ such that } \sum_{i=1}^{n} \lambda_i = 1 \tag{2.1}$$

Every surface vertex of the liver inner structures can be represented through the barycentric coordinates of the liver tetrahedron ($n = 4$) from the preoperative volumetric model in which it belongs. This tetrahedron is first determined. Then the barycentric coordinates λ are found by solving the system of linear equations formed by formula 2.1. Note that points on the liver surface can also be represented with this system for $n = 3$. First the triangular face on which the point belongs should be determined, then the barycentric coordinates with respect to the triangle vertices can be determined similarly.

2.3 Deformation Modelling for 3D/2D Deformable Registration

Liver is a highly deformable organ, see section 1.2.1. The liver is deformed between preoperative and intraoperative times, because of some loads such as pneumoperitoneum, see section 1.2.4.2, as well as breathing and manipulation by surgical tools. Loads and associated deformations can be numerically simulated under specific assumptions about the biomechanical properties of the liver and its environmental constraints, through the small finite elements of the preoperative volumetric liver mesh, i.e. the tetrahedra. However, before surgery, the intraoperative loads which will be exerted on the liver cannot be known with precision, and will vary during the surgery. Thus, the liver deformation should be modelled (figure 1.12a) in order to adapt to several scenarios. This can be performed in numerically simulating multiple load cases to obtain a matrix of vertex deformation (section 2.3.1), and then performing dimension reduction (section 2.3.3). This is Proper Orthogonal Decomposition (POD) [Sifakis & Barbic 2012], part of Model Order Reduction (MOR) methods, reducing the computational complexity of mathematical models in numerical simulations to obtain Reduced Order Models (ROM). Simulations can also be performed without biomechanical assumptions and instead be based on free-form deforming geometric models (section 2.3.2). The initial local geometry of the preoperative volumetric liver mesh can also be directly used in order to model deformation in another way, without simulations (section 2.3.4).

2.3.1 The Finite Element Method

The content of this section largely follows from [Sifakis & Barbic 2012]. A deformable body accumulates potential energy when deforming, referred to as strain energy E. Elastic restoring forces tend to bring back the body to its undeformed state, $f_{\vec{x}} = -\frac{\partial E}{\partial \vec{x}}$ where $f_{\vec{x}}$ is the elastic force at any position $\vec{x}$. Stress is a physical quantity which describes the magnitude of forces (per unit area) that cause deformation, while strain describes the proportion of deformation of the material [William Moebs 2016].

The liver can be modelled as a hyperelastic material, i.e. its has a non-linear strain-stress relationship, see Figure 2.6. When a body material is hyperelastic, elastic forces are conservative: the total work done by the internal elastic forces in a deformation path depends solely on the initial and final configurations, not the path itself. This is why E can be related to ϕ, a deformation map of a given configuration. The deformation function $\vec{\phi} : \mathbb{R}^3 \to \mathbb{R}^3$, $\vec{\phi}(\vec{X}) = (\phi_X(\vec{X}), \phi_Y(\vec{X}), \phi_Z(\vec{X}))$ maps every undeformed material point $\vec{X} = (X, Y, Z)^T$ to its respective deformed location $\vec{x} = \vec{\phi}(\vec{X})$. $\vec{X} \in \Omega$, the volumetric domain occupied by the undeformed elastic object in a 3D coordinate system, i.e. the reference (or undeformed) configuration. The relation between deformation and the strain energy is better defined on a local scale, as local parts deform in a different way. An energy density function $\Psi(\phi(\vec{X}))$ measures the strain energy per undeformed volume unit on an infinitesimal domain $\mathrm{d}\vec{X}$ around the material point $\vec{X}$.

$$E[\phi] = \int_{\Omega} \Psi(\phi(\vec{X}))\, \mathrm{d}\vec{X}$$

In the small region $\mathrm{d}\vec{X}$ around a specific material position X_i, the deformation map can be approximated using a first-order Taylor expansion:

$$\phi(\vec{X}) \approx \phi(\vec{X}_i) + \frac{\partial \vec{\phi}_{X_i}}{\partial X_i}(\vec{X} - \vec{X}_i) = \vec{x}_i + \frac{\partial \vec{\phi}_{X_i}}{\partial X_i}(\vec{X} - \vec{X}_i)$$

Upon defining the deformation gradient tensor $F \in \mathbb{R}^{3\times3}$, equivalent to the Jacobian matrix of the deformation map:

$$F = \frac{\partial(\phi_X, \phi_Y, \phi_Z)}{\partial(X, Y, Z)} = \begin{bmatrix} \frac{\partial \phi_X}{\partial X} & \frac{\partial \phi_X}{\partial Y} & \frac{\partial \phi_X}{\partial Z} \\ \frac{\partial \phi_Y}{\partial X} & \frac{\partial \phi_Y}{\partial Y} & \frac{\partial \phi_Y}{\partial Z} \\ \frac{\partial \phi_Z}{\partial X} & \frac{\partial \phi_Z}{\partial Y} & \frac{\partial \phi_Z}{\partial Z} \end{bmatrix}$$

The approximation becomes:

$$\phi(\vec{X}) \approx F_i\vec{X} + \vec{b}$$

where $\vec{b} = \vec{x}_i - F_i\vec{X}_i$, and is thus a constant.

Thus, the energy density function initially expressed with respect to the deformation map as $\Psi(\phi(\vec{X}))$ can be expressed with respect to the deformation gradient F as $\Psi(F)$. Hence:

$$E[\phi] = \int_\Omega \Psi(F) d\vec{X}$$

Ultimately, the precise mathematical expression for $\Psi(F)$ is the property which models the material and can be adapted to multiple ones.

A relation between the internal force density and a deformation can be obtained through the first Piola-Kirchhoff stress tensor $P \in \mathbb{R}^{3\times3}$:

$$\vec{f}(\vec{X}) = \nabla \cdot P(\vec{X}) = \frac{\partial P_X}{\partial X} + \frac{\partial P_Y}{\partial Y} + \frac{\partial P_Z}{\partial Z}$$

For hyperelastic materials, P is purely a function of the deformation gradient, and is related to the strain energy via the simple formula:

$$P(F) = \frac{\partial \Psi(F)}{\partial F}$$

2.3.1.1 Linear Tetrahedra

[Sifakis & Barbic 2012] also outlined a way of dealing with the computation of strain energy and elastic forces for volumetric meshes composed of linear tetrahedra. For these meshes, the reconstructed deformation map $\hat{\phi}$ can be defined to be a piecewise linear function over each tetrahedron of index i:

$$\hat{\phi}(\vec{X}) = A_i\vec{X} + \vec{b}_i$$

with $A_i \in \mathbb{R}^{3\times3}$ and $b_i \in \mathbb{R}^3$ specific to each tetrahedron. In fact, A_i is the deformation gradient $F_i = \frac{\partial \hat{\phi}}{\partial \vec{X}} = A_i$ and is constant on each tetrahedron i. When $\vec{X_1}, ..., \vec{X_4}$ the undeformed (reference) locations of the tetrahedron vertices and $\vec{x_1}, ..., \vec{x_4}$ the corresponding deformed ones:

$$\hat{\phi}(\vec{X}) = F_i\vec{X} + \vec{b} \Longrightarrow \begin{cases} \vec{x}_1 = F_i\vec{X}_1 + \vec{b} \\ \vec{x}_2 = F_i\vec{X}_2 + \vec{b} \\ \vec{x}_3 = F_i\vec{X}_3 + \vec{b} \\ \vec{x}_4 = F_i\vec{X}_4 + \vec{b} \end{cases}$$

When the last equation is subtracted from the three others, this gives:

$$\begin{cases} \vec{x}_1 - \vec{x}_4 = F_i(\vec{X}_1 - \vec{X}_4) \\ \vec{x}_2 - \vec{x}_4 = F_i(\vec{X}_2 - \vec{X}_4) \\ \vec{x}_3 - \vec{x}_4 = F_i(\vec{X}_3 - \vec{X}_4) \end{cases}$$

When converting to a matrix equation:

$$\begin{bmatrix} \vec{x}_1 - \vec{x}_4 & \vec{x}_2 - \vec{x}_4 & \vec{x}_3 - \vec{x}_4 \end{bmatrix} = F_i \begin{bmatrix} \vec{X}_1 - \vec{X}_4 & \vec{X}_2 - \vec{X}_4 & \vec{X}_3 - \vec{X}_4 \end{bmatrix}$$
$$D_s = F_i D_m$$

with D_s the deformed shape matrix for the current tetrahedron i and D_m the reference shape matrix. D_m only depends on the vertex coordinates in the reference (undeformed) configuration and is therefore a constant matrix. It is also not singular, assuming that the reference shape of the tetrahedron is non-degenerate (nonzero volume $V_i = \frac{1}{6}|D_m|$). Therefore, the constant D_m^{-1} can be precomputed, stored and used to directly determine F_i from the locations of the tetrahedron vertices:

$$F_i = D_s D_m^{-1}$$

As F is constant over the linear tetrahedron, the strain energy of this element reduces to:

$$E_i = V_i \cdot \Psi(F_i)$$

Note that when a tetrahedron is inverted and the strain energy is not defined for this case, such as for the Neo-Hookean model (section 2.3.1.2), it can be inverted back using specific operations [Irving *et al.* 2006]. The elastic forces applied on the tetrahedron vertices can be computed from the strain energy:

$$f_k^i = -\frac{\partial E_i}{\partial \vec{x}_k}$$

$$\vec{f} = \begin{bmatrix} \vec{f}_1 & \vec{f}_2 & \vec{f}_3 \end{bmatrix} = -V_i P(F_i) D_m^{-T}$$

As a consequence of conservation of momentum, the sum of all four internal nodal forces equals zero and thus $\vec{f}_4 = -(\vec{f}_1 + \vec{f}_2 + \vec{f}_3)$.

2.3.1.2 Material's Constitutive Models

In order to build a constitutive model through $\Psi(F)$ defining specific expected features of the material deformation, e.g. volume conservation or independence to rotation, certain intermediate quantities, such as invariants and strain measures, are built from

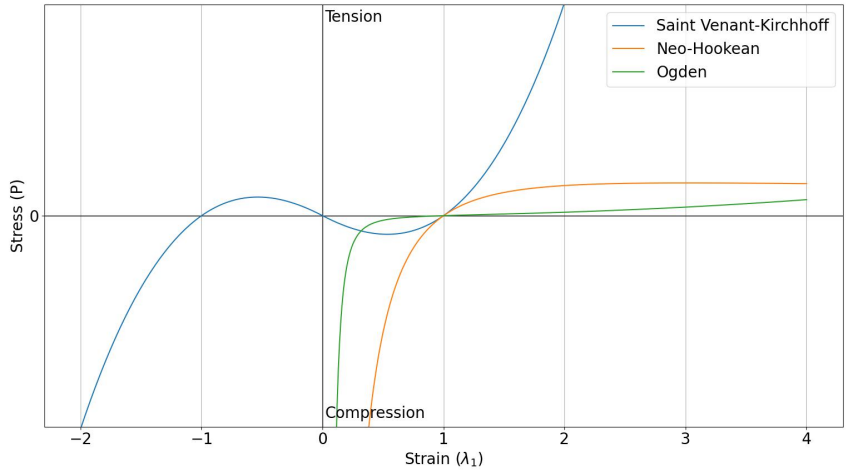


Figure 2.6: Stress-strain curves for uniaxial loading and nearly incompressible ($\lambda_2 = \lambda_3 = |\lambda_1^{\frac{1}{2}}|$ constitutive models with parameters $k = 6$kPa and $\nu = 0.49$ for Saint-Venant Kirchoff and Neo-Hookean models, while $\alpha_1 = \sqrt{10.06}$ and $\mu_1 = 4.1$kPa for the Ogden model, without the volumetric energy component.

F [Sifakis & Barbic 2012]. For instance, $J = \det F = \lambda_1\lambda_2\lambda_3$. $\lambda_1, \lambda_2, \lambda_3$ are the eigenvalues of F and thus the principal stretches oriented along the eigenvector directions and can be obtained by Singular Value Decomposition (SVD), see section 2.3.3.1. The right Cauchy–Green deformation tensor C expresses the square of local change in distances due to deformation

$$C = F^T F$$

Its invariants are often used in the expressions for strain energy density functions and are defined by:

$$I_1^C = \mathrm{Tr}(C) = \lambda_1^2 + \lambda_2^2 + \lambda_3^2$$
$$I_2^C = \frac{1}{2}((\mathrm{Tr}(C))^2 - \mathrm{Tr}(C^T C)) = \lambda_1^2\lambda_2^2 + \lambda_2^2\lambda_3^2 + \lambda_3^2\lambda_1^2$$
$$I_3^C = |C| = J^2 = \lambda_1^2\lambda_2^2\lambda_3^2$$

where $\lambda_1^2, \lambda_2^2, \lambda_3^2$ are the eigenvalues of C.
From C and the identity matrix I, the Green strain tensor E_G, a nonlinear (quadratic) function of deformation, can be built:

$$E_G = \frac{1}{2}(C - I)$$

When the body is in its undeformed configuration, i.e. $\phi(\vec{X}) = \vec{X}$, and when the body is rigidly displaced and thus $\phi(\vec{X}) = R\vec{X} + t$ (rotation matrix and translation vector), $C = I$ and thus $E_G = 0$, which are expected features of a strain measure.

These invariants are used in the following models:

Saint Venant-Kirchhoff model [Truesdell & Toupin 1960]. It uses the Green strain tensor E_G and the Lamé coefficients μ and λ, which are related to the material properties of Young modulus k (measure of stretch resistance) and Poisson's ratio ν (measure of incompressibility).

$$\mu = \frac{k}{2(1+\nu)}$$

$$\lambda = \frac{k\nu}{(1+\nu)(1-2\nu)}$$

$$\Psi(F) = \mu \operatorname{Tr}(E_G^T E_G) + \frac{\lambda}{2}(\operatorname{Tr}(E_G))^2$$

This is a model invariant to rigid body transformations which is a modification of the linear elastic material model for handling nonlinear deformations. Its scope is limited to a certain degree due to its poor resistance to forceful compression: starting from its undeformed configuration (strain = 1 in Figure 2.6), it stiffens when the compression is low but then its resistance decreases as the compression grows (strain ≈ 0.58), even allowing the material to compress to zero volume and inverting [Sifakis & Barbic 2012].

Neo-Hookean model [Mooney 1940].

$$\Psi(I_1^C, I_3^C) = \frac{\mu}{2}(I_1^C - \ln(I_3^C) - 3) + \frac{\lambda}{8}\ln^2(I_3^C)$$

It is also an isotropic constitutive model (for materials whose properties are not direction dependent). Unlike Saint Venant-Kirchhoff materials, it constructs a powerful energy barrier that strongly resists extreme compression [Sifakis & Barbic 2012], see Figure 2.6. The stability and highly non-linear behaviour under compression is consistent with the behaviour of cross-linked polymer chains and has made neo-Hookean materials a popular choice. The relationship between applied stress and strain is initially linear (around the undeformed configuration, strain = 1), but at a certain point the stress-strain curve will plateau at large strains. In contrast, in many soft tissues, the stiffness increases upon increased deformation (strain-stiffening) due to their fibrous, semi-flexible, biopolymeric microstructure [Motte & Kaufman 2013].

Ogden [Ogden 1972]. It uses material coefficients α_i, μ_i, respectively 'non-linearity' and shear modulus parameters, where $i = 1, ..., N$ and N can range from 1 to 6, and a bulk-like modulus parameter β involved in an added volumetric energy component $U(J)$ which facilitates 'near' incompressibility [Bonet & Wood 1997]:

$$\Psi(\lambda_1, \lambda_2, \lambda_3) = \sum_{i=1}^{N} \frac{\mu_i}{\alpha_i^2}(\lambda_1^{\alpha_i} + \lambda_2^{\alpha_i} + \lambda_3^{\alpha_i} - 3 - \alpha_i \ln J) + U(J)$$

$$U(j) = \frac{1}{2}\beta(J-1)^2$$

The Ogden model also represents isotropic materials. For $\alpha \geq 1$, the Ogden model predicts strain-stiffening behaviour, see Figure 2.6. It is this latter property that makes the Ogden model suitable to soft tissues. Under compression, the Ogden model is always stiffening when the compression grows [Lohr *et al.* 2022].

2.3.1.3 Numerical Integration of Newton's Equation of Motion and Quasi-Static Simulations

The motion of each element whose domain is subject to boundary conditions and loads can be dealt with using Newton's ordinary differential equation of motion: $\frac{dx}{dt} = v(t)$ and $\frac{dv}{dt} = a(t)$. The goal of the finite difference methods is to determine the values of x_{n+1} and v_{n+1} at time $t_{n+1} = t_n + \Delta t$. The nature of many of the integration algorithms can be understood by expanding $v_{n+1} = v(t_n + \Delta t)$ and $x_{n+1} = x(t_n + \Delta t)$ in a Taylor series :

$$\begin{cases} v_{n+1} & = v_n + a_n \Delta t + O((\Delta t)^2) \\ x_{n+1} & = x_n + v_n \Delta t + \frac{1}{2} a_n (\Delta t)^2 + O((\Delta t)^3) \end{cases}$$

Most of the numerical integration schemes retain only the $O(\Delta t)$ (first-order) terms. Δt must be chosen so that the integration method generates a stable solution. If the system is conservative, Δt must be sufficiently small so that the total energy is conserved to the desired accuracy [Gould *et al.* 2007]. The implicit (backward) Euler method is stable for any time step Δt for which the nonlinear equations are solved to satisfactory accuracy and uses v_{n+1} in v_{n+1} (implicit function of itself)

$$\begin{cases} x_{n+1} = x_n + v_{n+1} \Delta t \\ v_{n+1} = v_n + a_{n+1} \Delta t \end{cases}$$

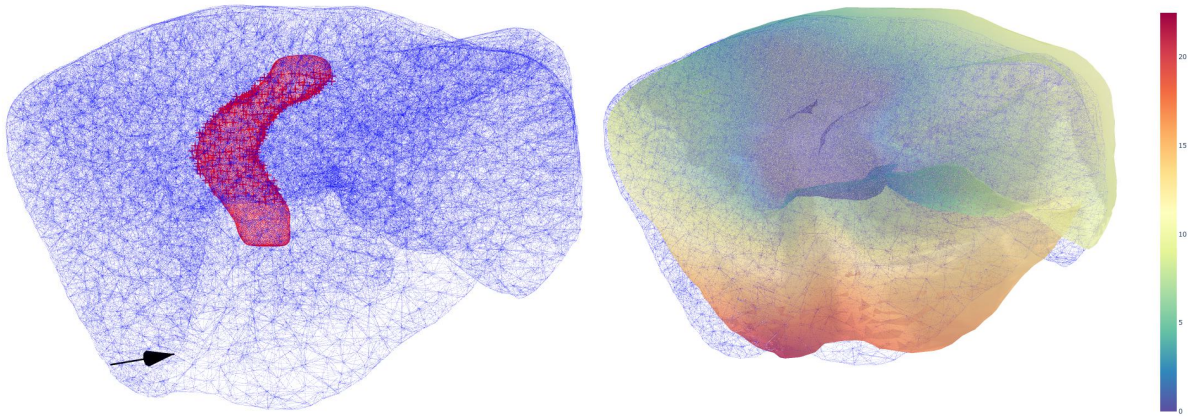
Under the general force case, with f the forces and M the mass matrix, according to Newton's second law of motion:

$$\begin{cases} x_{n+1} = x_n + v_{n+1} \Delta t \\ v_{n+1} = v_n + M^{-1} f(x_{n+1}, v_{n+1}) \Delta t \end{cases} \tag{2.2}$$

f combine internal and external forces. Quasi-static simulations are generally performed for static solids in Finite Element Method (FEM). They comprise a sequence of static simulations over a given duration and assumes that the forces must almost balance for each time step, i.e. their sum should be 0. This assumption is equivalent to defining every configuration over time as the result of a rest configuration subject to the imposed kinematic constraints and boundary conditions [Sifakis & Barbic 2012]. From the nonlinear finite element method, the internal elastic forces can be solved and the time varying positions can be recovered.

Simulation details. We automatically implement the simulations with a Python script which generates 500 text files using an adapted pyFEBio library and input them to the FEBio command line application [Maas *et al.* 2012]. The file should contain all simulations parameters:

- Mesh tetrahedra with volumetric vertex coordinates.
- Material parameters. There is still no clear consensus on which of the polynomial forms models the nonlinear behaviour of the hepatic tissue the most accurately. The variety of experimental conditions (ex-vivo vs in-vivo, strain rate, considered



(a) FEM simulation conditions. A nodal force of 0.5N represented by the arrow is applied on the volumetric liver mesh (blue) whose red vertices (crosses) close to the vena cava (red) are fixed.

(b) FEM simulation results. Surface face colours represent the displacement (colourmap in mm) from the initial blue configuration. The region around the vena cava does not deform much unlike the one where the force is applied.

Figure 2.7: FEM simulation

species ...) makes it very difficult to provide a precise quantitative characterisation of the liver mechanics [Marchesseau *et al.* 2017]. For the simulations, we use the Ogden one-term ($n = 1$) constitutive model with generic liver parameters [Pellicer-Valero *et al.* 2020]: $\alpha_1 = \sqrt{10.06}, \mu_1 = 4.1$ kPa and $\beta = 100\mu_1 = 410$.

- Boundary conditions. The main structure maintaining the liver is the vena cava [Flament *et al.* 1982]. It is segmented and reconstructed together with the liver. Liver vertices inside the vena cava and less than 2 mm away from its surface are fixed and used as fixed boundary condition, see Figure 2.7.

- Loads. We only simulate nodal forces applied on random surface vertices with a random orientation and a load curve of 10 linear steps to reach the magnitude of 0.5N or 5N. The liver deformation due to artificial pneumoperitoneal pressure depends on the other intra-abdominal structures and the abdominal wall. Thus, it requires segmentation of all these structures and the determination of realistic biomaterial modelling and parameters, thus being very challenging. It was attempted on pigs [Bano *et al.* 2012], with very strong assumptions on these modelling and parameters. The same issue occurs for breathing.

- Solving. It is performed with a quasi-Newton optimisation method which uses another approximation of the Hessian matrix with respect to the Newton method compared to the Gauss-Newton one, see section 2.4.5, the Broyden–Fletcher–Goldfarb–Shanno (BFGS) technique.

2.3.2 Free Form Deformation

Free Form Deformation (FFD) [Sederberg & Parry 1986] uses Bernstein polynomials. A polynomial of degree d in a canonical base is defined as:

$$f(x) = \sum_{i=0}^{d} c_i x^i$$

with c_i the coefficients. In the Bernstein base, it can be expressed as:

$$f(x) = \sum_{i=0}^{d} b_i B_i^d(x)$$

where:

$$B_i^d(x) = \binom{d}{i} x^i (1-x)^{d-i}$$

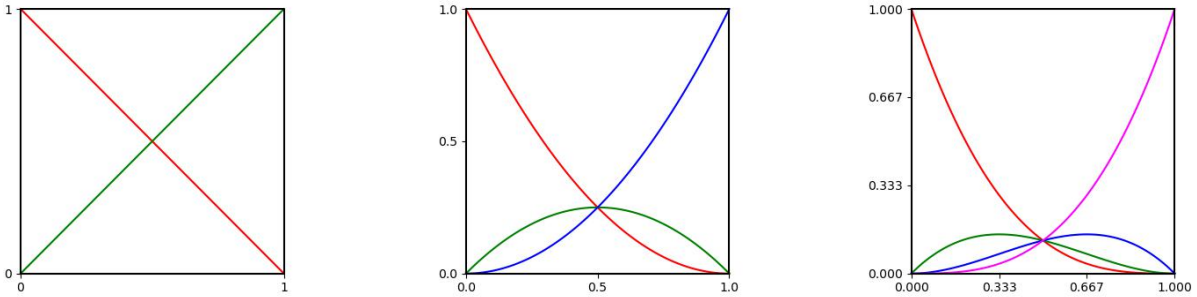


Figure 2.8: The Bernstein polynomials of degree 1,2,3 in the unit square. The extremum value of B_i^d occurs at $x = i/d$. For each degree, they form a partition of unity.

In the interval $x \in [0, 1]$, all Bernstein polynomials lie between 0 and 1 and they form a partition of unity, i.e. their sum equals 1, see figure 2.8. In this interesting interval, the Bernstein base polynomial is a linear convex combination of the points $(i/d,\ b_i)$. Thus the points (x, y) lies inside the convex hull of the control points $(i/d, b_i)$.

FFD exploits these properties in 3D and consists in 3 different steps:

1. Physical domain mapping to a local coordinate system on a rectangular parallelepiped (box) region englobing the object (volumetric liver in our case, see figure 2.9). In this system, for any point $\vec{x}$ in this box of height H, width W and length L, its coordinates (s, t, u) are:

   $$\vec{x} = \vec{x_0} + s\vec{H} + t\vec{W} + u\vec{L} \ \ni \ 0 < s < 1, 0 < t < 1, 0 < u < 1$$

   where $\vec{x_0}$ are the coordinates of the origin of the box.

2. Building of a grid of control points $\vec{p}_{ijk}$ on the box forming a lattice. These form $l+1$ planes in the $\vec{H}$ direction, $m+1$ planes in the $\vec{W}$ one, and $n+1$ planes in the $\vec{L}$ one.

   $$\vec{p}_{ijk} = \vec{x_0} + \frac{i}{l}\vec{H} + \frac{j}{m}\vec{W} + \frac{k}{n}\vec{L} \ni \ i \in \{0, ..., l\}, j \in \{0, ..., m\}, k \in \{0, ..., n\}$$

   .

3. The deformations are specified by moving the control points to positions $\vec{p'}_{ijk}$. The deformed position $\vec{\hat{x}}$ of an arbitrary point is found by evaluating the vector valued trivariate Bernstein polynomial. The control points $\vec{p'}_{ijk}$ are actually the coefficients of the Bernstein polynomial [Sederberg & Parry 1986]:

$$\vec{\hat{x}} = \sum_{i=0}^{l} \binom{l}{i} s^i (1-s)^{l-i} \left(\sum_{j=0}^{m} \binom{m}{j} t^j (1-t)^{m-j} \left(\sum_{k=0}^{n} \binom{n}{k} u^k (1-u)^{n-k} \vec{p'}_{ijk} \right) \right)$$

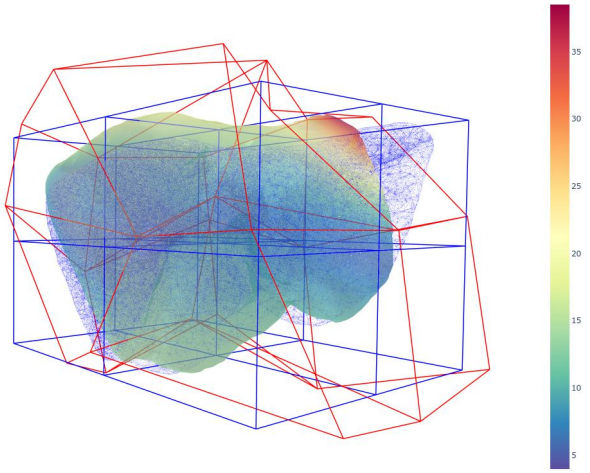


Figure 2.9: FFD simulation with $3 \times 3 \times 3$ control points. The preoperative volumetric liver mesh and lattice are in blue, while the deformed lattice is in red and the deformed surface faces represent the displacement (colourmap in mm) from the initial blue configuration.

Simulation details. Using PyGeM, we first create a box englobing the volumetric liver and define 3 control points per dimension (27 control points on the lattice). Each control point is randomly displaced (with a Gaussian distribution of mean 0 and variance of 20 mm normalised in box units), illustrated in Figure 2.9.

2.3.2.1 As-Rigid-As-Possible Penalty

We optionally constrain the new shape $x \in \mathbb{R}^{n \times 3}$ to be As-Rigid-As-Possible [Sorkine & Alexa 2007] with respect to the initial shape $x^0 \in \mathbb{R}^{n \times 3}$ while being close to the deformed shape of vertices $x^1 \in \mathbb{R}^{n \times 3}$ obtained from FFD. The local rigidity energy can be defined as the squared difference between the new edge lengths and the initial edge lengths, which are characterised for each x_i^0 by the distance to its m adjacent vertices x_{ij}^0 with $j \in \{0, ..., m\}$. Constraint terms can be weighted using α:

$$\arg\min_{x} \sum_{i=1}^{n} \left(\alpha \|\vec{x}_i - \vec{x}_i^1\|_2^2 + \sum_{j=1}^{m} w_{ij} \left(\|\vec{x}_i - \vec{x}_{ij}\|_2 - \|\vec{x}_i^0 - \vec{x}_{ij}^0\|_2 \right)^2 \right)$$

Simulation details. We use uniform weights $w_{ij} = 1$ and a weight for the first term of $\alpha = 0.3$.

2.3.3 Dimension Reduction

This step occurs after the simulations of n deformations (snapshots), from finite element models (section 2.3.1) or free-form geometric models (section 2.3.2). Each snapshot contains the 3 coordinates of the displacement of the p vertices from the preoperative volumetric liver mesh ones and is represented as a vector of dimension $d = 3p$, forming a snapshot matrix $X \in \mathbb{R}^{n \times d}$. No deformation is represented by the zero value.

2.3.3.1 Truncated SVD (PCA)

The rank of a matrix corresponds to the maximal number of linearly independent columns or rows of the matrix. The SVD of a rectangular or square matrix $X \in \mathbb{R}^{n \times d}$ of rank r, where n is the number of variables and d the associated number of dimensions, is the factorisation of X into the product of three matrices:

$$X = U\Sigma V^T \tag{2.3}$$

with $U \in \mathbb{R}^{n \times n}, V \in \mathbb{R}^{d \times d}, \Sigma \in \mathbb{R}^{n \times d}$, where the columns of U and V are orthonormal and respectively called left and right singular vectors and Σ is diagonal with positive real entries. $\vec{\sigma}$, the diagonal values of Σ, are the corresponding singular values and are equal to the root of the eigenvalues of $X^T X$. An eigenvector v of a linear transformation A is a vector which has its directions unchanged by a given linear transformation and is scaled by a constant eigenvalue factor λ, i.e. $A\vec{v} = \lambda\vec{v}$. On the covariance matrix $A = \frac{1}{n-1}X^T X$ where X is centred, eigenvectors of interest point into the successive orthonormal directions of the largest variance of the data, whose magnitude λ equals the variance in these directions [Shlens 2014]. Principal Component Analysis (PCA) is a linear dimension reduction technique which identifies these directions that capture the most variance in the data and project the data onto those directions, which are called principal components. It can be performed using SVD. SVD finds the m-dimensional subspace which minimises the sum of the squares of the perpendicular distances of the observations to the subspace, approximating X as:

$$X \approx \sum_{i=1}^{m} \sigma_i \vec{u}_i \vec{v}_i^T$$

For a given $m \leq d$ and $m \leq n$ dimension (optimally close to the rank r), $\{\vec{u}_1, ..., \vec{u}_m\}$ is an orthonormal basis for the column space and $\{\vec{v}_1, ..., \vec{v}_m\}$ is an orthonormal basis (subspace) for the row space. Truncated SVD consists in performing SVD on the X matrix directly, without centring its columns. In our case, data are already centred as zero values represents no coordinate displacement. $\vec{\mu} \in \mathbb{R}^d$ is the vector of the undeformed preoperative volumetric liver vertex coordinates. A deformed shape of the snapshot matrix X represented by its vertex coordinates $\vec{\hat{x}}_j$ with $j \in \{1, ..., n\}$ can thus be reconstructed from its m principal components or subspace $\phi \in \mathbb{R}^{m \times d}, \vec{\phi}_i = \vec{v}_i^T$ and the scores $\beta \in \mathbb{R}^{n \times m}, \beta_{j,i} = \sigma_i u_{j,i}$ as:

$$\vec{\hat{x}}_j = \vec{\mu} + \sum_{i=1}^{m} \beta_{j,i} \vec{\phi}_i \tag{2.4}$$

This formula is used as a deformation model where ϕ is the subspace and $\hat{\beta}$ are coefficients which are input to obtain deformations following this linear model.

Simulation details. $n \approx 5000$ simulated shapes are generated from FEM or FFD. $\hat{\beta}_i$ coefficients are limited to the range $[-2\sigma_i, 2\sigma_i]$ in order to keep 95% of coefficients resulting from a normal distribution of standard deviation σ. The liver shape model components for a patient obtained from truncated SVD of snapshot matrices from FEM and FFD simulations are illustrated in appendix figures A.1, A.2 and A.3.

2.3.3.2 Local Truncated SVD (Local PCA)

Unlike global PCA, local PCA [Kambhatla & Leen 1997] attempts to obtain locally linear models of reduced dimensions. The algorithm first partitions the data space into disjoint regions and then performs 'local' PCA for each disjoint regions. We choose to partition the preoperative liver volume using k-means clustering, which creates k clusters in order to minimise the within-cluster variance where c_l is the cluster centroid and S_l is a cluster set among the k ones:

$$\arg\min_{S} \sum_{l=1}^{k} \sum_{x \in S_l} \|\vec{x} - c_l\|^2$$

Each vertex index is related to a cluster l, which is used to create specific snapshots matrix for each cluster l: X^l. We then perform truncated SVD on each region cluster X^l:

$$\vec{\tilde{x}}_j^l = \vec{\mu}^l + \sum_{i=1}^{m} \beta_{j,i}^l \vec{\phi}_i^l$$

Simulation and parameter details. From the same $n \approx 5000$ simulated shapes, we use $C = 30$ clusters and keep $m = 7$ components per cluster, which results in 210 shape components. $\hat{\beta}_i^l$ coefficients are also limited to the range $[-2\sigma_i^l, 2\sigma_i^l]$ in order to keep 95% of coefficients resulting from a normal distribution of standard deviation σ^l. It is illustrated for FFD in figure A.4.

2.3.4 Locally Linear Embedding

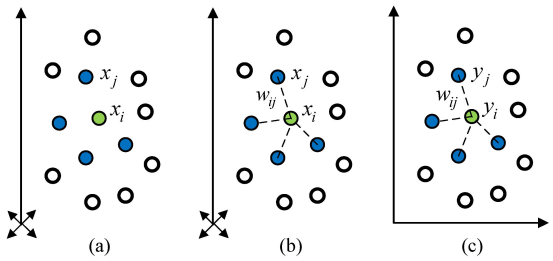


Figure 2.10: Locally Linear Embedding (LLE) steps, from [Wang *et al.* 2015]. (a) Neighbourhood selection, (b) Linear weight computation for data reconstruction from neighbours, (c) Mapping to embedded coordinates to form a subspace.

LLE [Roweis & Saul 2000] does not need simulations and uses the geometry and neighbourhood of a data matrix. We use the preoperative volumetric liver $\mu \in \mathbb{R}^{p \times 3}$. Neighbours can be selected according to Euclidean or geodesic distance from the mesh. For a data matrix $X \in \mathbb{R}^{p \times D}$ with p the number of points and D the number of dimensions,

LLE assumes that each data point $\vec{X}_i$ and its n neighbours $X^i \in \mathbb{R}^{n \times D}$ lie on or close to a locally linear patch of the manifold. Unlike local PCA, see section 2.3.3.2, the patches may overlap. Linear coefficients reconstruct each data point from its neighbours, illustrated in figure 2.10b. These coefficients $W \in \mathbb{R}^{p \times p}$, are obtained by solving the constrained minimisation:

$$\arg\min_{W} \sum_{i=1}^{p} \left\| \vec{X}_i - \sum_{j=1}^{p} W_{ij} \vec{X}_j^i \right\|_2^2$$

$$\text{such that } W_{ij} = 0 \; \forall \; X_j \notin X^i \text{ and } \sum_{j=1}^{p} W_{ij} = 1$$

The weights W_{ij} are obtained by solving a least-squares problem. The local covariance $C^i = X^{iT} X^i$ from centred X^i is obtained and the linear system $C^i W_i = 1$ is solved. The weights summarise the contribution of the j^{th} data point to the i^{th} reconstruction, while being invariant to translations, rotations, and rescalings of each data point and its neighbours. The idea of LLE is that same weights W_{ij} that reconstruct the i^{th} data point in D dimensions should also reconstruct its embedded manifold coordinates $\vec{Y}_i$ in m dimensions, see figure 2.10c. Thus the embedding manifold $Y \in \mathbb{R}^{p \times m}$ is obtained by solving the minimisation [Roweis & Saul 2000]:

$$\arg\min_{Y} \sum_{i=1}^{m} \left\| \vec{Y}_i - \sum_{j=1}^{m} W_{ij} \vec{Y}_j^i \right\|_2^2$$

It can be solved by computing the SVD of $M = I - W$, with I the identity matrix. The right singular vectors corresponding to the m smallest singular values are selected as the embedding manifold, i.e. the subspace $\phi \in \mathbb{R}^{m \times p}$. Therefore, it can be used to formulate a model as equation 2.4, with the difference to be applicable on each point $\hat{\vec{x}}_q$ of a deformed shape with $q \in \{1, ..., p\}$ instead of the coordinates as $d = 3p$:

$$\hat{\vec{x}}_q = \vec{\mu}_q + \sum_{i=1}^{m} \hat{\beta}_i \vec{\phi}_{i,q} \tag{2.5}$$

Parameter details. In our case, p is the number of preoperative liver volumetric vertices and D is 3. We select $n = 10$ nearest neighbours using Euclidean distance, and we 'reduce' the dimension space to $m = 200$ dimensions to form the subspace $\phi = Y^T \in \mathbb{R}^{m \times p}$. This model is assumed to represent linear deformations close to locally affine transformations of the patches [Modrzejewski 2020]. However, as the patches overlap, each component combine multiple patches of the whole liver, see appendix figure A.5.

2.4 Minimally Invasive Camera Calibration

2.4.1 Camera Modelling

A camera maps a lit 3D scene to a 2D image by means of an optical system. An endoscopic camera is usually assumed to perform central perspective projection and is modelled as

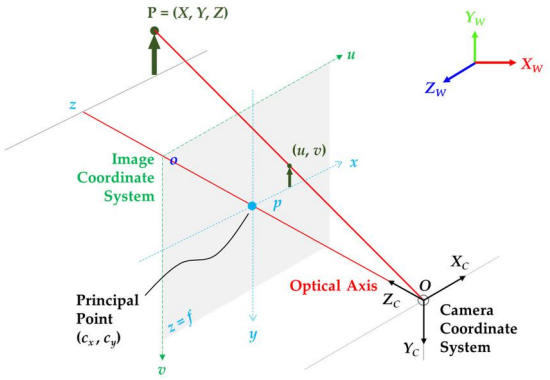


Figure 2.11: The pinhole camera model, from [Yeong *et al.* 2021]. The optical axis (also referred to as principal axis) aligns with the Z-axis of the camera coordinate system (Z_C), and the intersections between the image plane and the optical axis is referred to as the principal point (cx, cy). The pinhole opening serves as the origin (O) of the camera coordinate system and the distance between the pinhole and the image plane is referred to as the focal length (f). Computer vision convention uses right-handed system with the z-axis pointing toward the target from the direction of the pinhole opening, while y-axis pointing downward, and x-axis rightward. Conventionally, from a viewer's perspective, the origin (o) of the 2D image coordinate system (x, y) is at the top-left corner of the image plane with x-axis pointing rightward, and y-axis downward. The (u, v) coordinates on the image plane refers to the projection of points in pixels.

a pinhole camera. The principle consists in capturing through a planar image sensor the light rays reflected by the 3D scene through the pinhole aperture. The pinhole camera aperture is modelled as a point $\vec{O} = (0, 0, 0)^T$ through which all projection lines (reflected light rays) must pass, being at the origin of the *camera* space (a Euclidean coordinate system), i.e. the centre of projection also referred to as optical or camera centre, see figure 2.11. The Z-axis is the optical axis (passing through the geometrical centre of the optical system) or principal axis, and the visible scene has a positive depth Z. While the planar image sensor is physically behind the camera centre, the image or focal plane is modelled by central symmetry in front of the camera centre and is defined as $z = f$, with f the focal length, a camera projection parameter, see figure 2.11. A 3D scene point $\vec{P} = (X_c, Y_c, Z_c)^T$ is mapped to the intersection $\vec{p} = (u, v, f)^T$ between the image plane and a line joining the point P to the centre of projection O. The intersection between the image plane and the optical axis is the principal point $\vec{c} = (c_x, c_y, f)^T$ [Hartley & Zisserman 2003, Yeong *et al.* 2021].

From the intercept theorem, this central projection model maps a 3D point $\vec{P}$ to $\vec{p} = (\frac{f}{Z_c}X_c + c_x, \frac{f}{Z_c}Y_c + c_y)^T$ when the constant last coordinate $z = f$ is discarded. This assumes the image coordinates are orthogonal (no skew) and even orthonormal, but the latter is not always applicable. In the orthogonal case, $\vec{p} = (\frac{f_x}{Z_c}X_c + c_x, \frac{f_y}{Z_c}Y_c + c_y)^T$. However, the same coordinates $(\frac{f_x}{Z_c}X_c + c_x, \frac{f_y}{Z_c}Y_c + c_y)^T$ can be obtained from different 3D points belonging to the same projection line. In the modelled Euclidean coordinate system, these points are the scaled versions of $\vec{P}$ with a factor $\alpha \in]0, +\infty]$ and are considered homogeneous. The 2D image Cartesian coordinates $(u, v)^T$ can be transformed into homogeneous coordinates $\vec{p} = (u', v', w')^T$ where $u = \frac{u'}{w'}, v = \frac{v'}{w'}$. Homogeneous

coordinates are thus defined up to scale. Cartesian 3D coordinates can also be converted to homogeneous coordinates as $\vec{P} = (X_c, Y_c, Z_c, 1)^T$. Using homogeneous coordinates, the central projection model can be formulated linearly as:

$$\underbrace{\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}}_{\substack{\text{image homogeneous coordinates} \\ \vec{p}}} = \underbrace{\alpha}_{\text{scale factor}} \underbrace{\begin{bmatrix} f_x & 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\substack{\text{camera intrinsics} \\ K}} \underbrace{\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}}_{\substack{\text{camera homogeneous coordinates} \\ \vec{P}}} \tag{2.6}$$

However, optical distortion, i.e. bending of scene straight lines in an image, usually occur in endoscopic lenses. This is why lens distortion should also be taken into account in the model. The most commonly encountered distortions are approximately radially symmetric or tangential, and can be of different types, such as barrel, pincushion and moustache distortion, see Figure 2.12.

Radial distortion, including barrel and pincushion ones, can be represented by the k_1, k_2 and k_3 distortion coefficients as $m^{distorted} = m(1 + k_1 r^2 + k_2 r^4 + k_3 r^5)$ where m stands for centred image coordinates with $m_u = u - c_x$ or $m_v = v - c_y$ and r for the distance from the centre. Barrel distortion is typically modelled with negative k values, whereas pincushion distortion has positive k values. Tangential distortion can occur when the lens is not perfectly aligned with the imaging plane and can be represented by the p_1 and p_2 coefficients through:

$$m_u^{distorted} = m_u + 2p_1 m_u m_v + p_2(r^2 + 2m_u^2)$$
$$m_v^{distorted} = m_v + 2p_2 m_u m_v + p_1(r^2 + 2m_v^2)$$

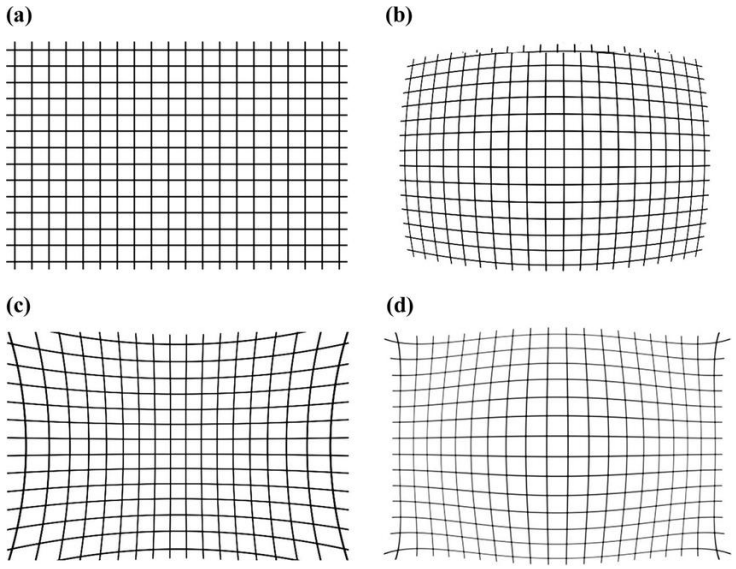
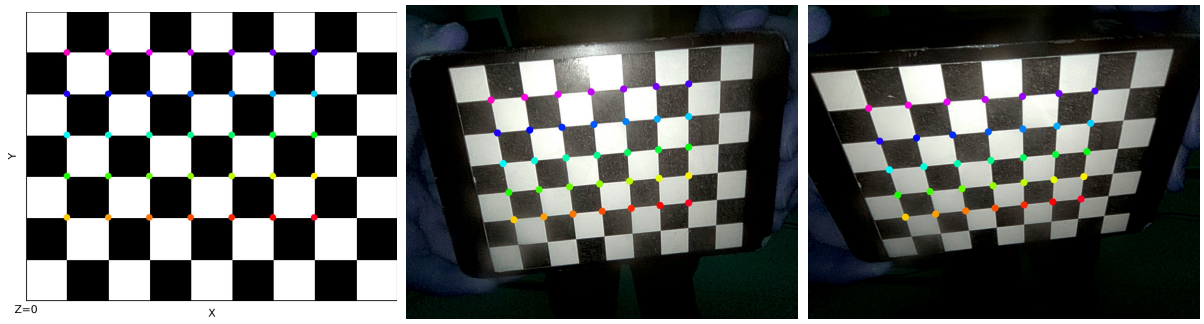


Figure 2.12: Types of lens distortion: (a) Non-distortion (b) Barrel distortion (c) Pincushion distortion (d) Moustache distortion, from [Ramírez-Hernández *et al.* 2020].



(a) Chessboard with feature points in 3D *world* space.

(b) Corresponding 2D feature points detected into different views of the imaged board.

Figure 2.13: Calibration patterns with 3D points and detected 2D correspondences.

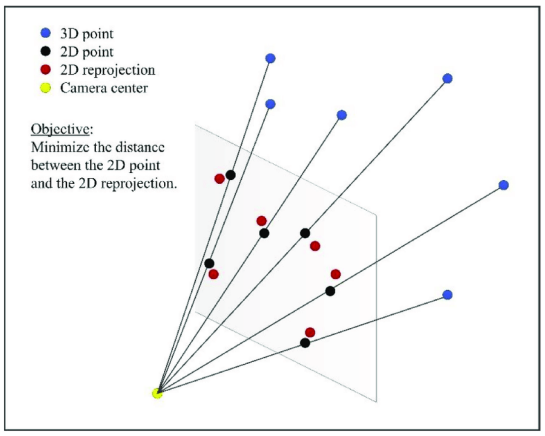


Figure 2.14: PnP principle and objective from [Guo *et al.* 2022a]

2.4.2 Camera Calibration Principle

It consists to retrieve the camera parameters (intrinsics and distortion coefficients) associated to the current imaging, that are: $f_x, f_y, c_x, c_y, k_1, k_2, k_3, p_1, p_2$. It is part of the registration pipeline in the MILS context, illustrated in figure 1.12b. As explained in section 1.3.3, the surgeon first explores the surgical field and adjust the camera in order to image a global view of the liver in a clean way, before calibrating the camera.

Several calibration methods exist in this aim [Cui *et al.* 2023], such as object-based calibration which uses a calibration object whose geometric information is known and self-calibration which moves a camera around an unknown static scene and uses point/line detected correspondences between images [Peng & Li 2010]. The most common procedure [Zhang 2000], implemented in OpenCV, consists in relying on 3D/2D feature point correspondences from multiple views in order to calibrate the camera. They are obtained thanks to the imaging of a rigid board containing well defined patterns with feature points such as corners, whose actual size and relative positions are known. The most commonly used patterns are the chessboard and the ChArUco ones, whose pattern corners are detectable using image processing techniques, illustrated in figure 2.13. In a 3D Euclidean coordinate system, called the *world* space, the board plane is modelled fixed at the X_wY_w plane, with $Z_w = 0$. As the size of the patterns is known, their feature points can be positioned in this 3D space with their actual dimension in the required unit, such as mm.

2D correspondences should be detected into multiple views of the board, see figure 2.13. In this aim, the endoscopic camera is moved either by the surgeon or an automatic system [Dowrick *et al.* 2023] to sample images of the board from different angles of view in a distance range similar to the intra-abdominal one during surgery.

As the board is rigid and thus cannot deform, the projection of the 3D feature points should fit the corresponding 2D points in the image plane. The current camera parameters are thus those which minimise the reprojection error, i.e. the total sum of squared distances between the detected feature points and the projected ones, see Figure 2.14. However, additional parameters should be retrieved. Indeed, the board plane position in the *camera* space is not initially known and is instead initially positioned in its own *world* space. A rigid transformation, i.e. rotation and translation $[R|t]$, is required in order to position it in the camera space, whose camera centre is at the origin. This rigid transformation is referred to as pose. Projective transformations generalise affine transformations. A projective transformation of the projective space $\mathbb{P}^n$ is represented by a non-singular linear transformation of homogeneous coordinates $X' \propto HX$ with H an $(n+1) \times (n+1)$ matrix. Projective transformations in 3D are linear transformations on homogeneous 4-vectors represented by a non-singular 4×4 matrix H [Hartley & Zisserman 2003]. The pose can thus be expressed in a matrix form:

$$M = \left[\begin{array}{c|c} \mathrm{R} & \mathrm{t} \\ \hline 0^T & 1 \end{array}\right] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and the problem can be formulated using homogeneous coordinates:

$$\underbrace{\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}}_{\substack{\text{camera} \\ \text{homogeneous coordinates}}} = \underbrace{\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{\substack{\text{pose transformation matrix} \\ M}} \underbrace{\begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}}_{\substack{\text{world} \\ \text{homogeneous coordinates}}}$$

Combining it with equation 2.6, the images coordinates of the feature points are related to the world coordinates such that:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \alpha KM \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{2.7}$$

2.4.3 Homography Estimation

A planar projective transformation is a linear transformation on homogeneous 3-vectors represented by a non-singular 3×3 matrix H [Hartley & Zisserman 2003]. When the Z_w coordinates of the points are assumed to have a value of 0, for planar board feature points,

equation 2.7 can be simplified:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \alpha \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ r_{31} & r_{32} & t_z \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ 1 \end{bmatrix}$$

This can be written as:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \alpha H \begin{bmatrix} X_w \\ Y_w \\ 1 \end{bmatrix}$$

H is the homography matrix (whose vectorisation is h), a matrix which contains the unknowns to be solved, mixing intrinsic and extrinsic parameters:

$$H = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix}$$

The system of equations can be simplified:

$$\begin{cases} u = \alpha(h_1 X_w + h_2 Y_w + h_3) \\ v = \alpha(h_4 X_w + h_5 Y_w + h_6) \\ 1 = \alpha(h_7 X_w + h_8 Y_w + h_9) \end{cases}$$

$$\begin{cases} u(h_7 X_w + h_8 Y_w + h_9) = h_1 X_w + h_2 Y_w + h_3 \\ v(h_7 X_w + h_8 Y_w + h_9) = h_4 X_w + h_5 Y_w + h_6 \end{cases}$$

When rewriting it into a homogeneous system and thus a matrix equation of the form:

$$Ah = 0$$

This gives:

$$A = \begin{bmatrix} -X_w & -Y_w & -1 & 0 & 0 & 0 & X_w u & Y_w u & u \\ 0 & 0 & 0 & -X_w & -Y_w & -1 & X_w v & Y_w v & v \end{bmatrix}$$

A can thus be filled using the coordinates from all the m corresponding feature points, giving a $2m \times 9$ matrix. Then, a method such as SVD, presented in section 2.3.3.1, can be used for solving the system and retrieves h as the right singular vector associated to the smallest singular values.

2.4.4 Retrieving Parameters from a set of Homographies

In the previous step, the homographies were calculated independently for each of the n views. The homographies encode both the common camera intrinsics as well as the extrinsic transformation parameters that are generally different for each view. The two first columns vectors of the transformation matrix must be orthonormal for allowing a valid rotation matrix. This property yields two fundamental constraints on the intrinsic parameters for a given homography and can be reformulated as a pair of linear equations for a vector defined with respect to intrinsics after specific mathematical operations [Burger 2016]. Stacking the associated $2n$ equations from all n views, this leads to

an overdetermined system of homogeneous linear equations, which can also be solved by SVD to obtain the camera intrinsics of K, i.e. f_x, f_y, c_x, c_y.

Once the camera intrinsics are known, the extrinsic parameters $[R|t]$ can be retrieved for each view i from the corresponding homography h_i. This is part of the Direct Linear Transformation (DLT) method, i.e. rewriting the perspective projection equation as a homogeneous linear equation and solving it by standard methods before recovering intrinsics and extrinsics. Alternatively, the pose can be estimated from the initial intrinsic parameters and n 3D/2D corresponding points, in a PnP problem formulation, see figure 2.14. Multiple methods exist for solving the PnP problem, such as P3P for $n = 3$ corresponding points which use geometric relations between the points and the centre of projection to obtain a system of 3 equations from which multiple solutions can be obtained [Gao *et al.* 2003], and Efficient PnP for $n \geq 4$ which express each of the n points as a weighted sum of four virtual control points whose coordinates become the unknowns of the problem [Lepetit *et al.* 2009]. Each PnP solving method can be used upon certain conditions and assumptions, such as a number of corresponding points, coplanarity (or not) of the points, direct or iterative estimation. Additional PnP details are given in section 4.3.

Then, the lens distortion parameters can be initialised with zeros or estimated by linear least-squares fitting, minimising the reprojection error. They are then refined simultaneously with all other parameters in a final, overall optimisation step.

2.4.5 Parameter Refinement, Optimisation

Parameters θ (intrinsics, extrinsics, distortion coefficients) can be refined using an optimiser dedicated to solve a cost function f of non-linear least squares, a sum of m squared residuals r with respect to the parameters θ, i.e. the reprojection error between the detected feature points $p \in \mathbb{R}^{m\times 2}$ and the projected ones $\hat{p} \in \mathbb{R}^{m\times 2}$ depending on θ:

$$f(\theta) = \sum_{i}^{m} r_i^2(\theta) = \sum_{i}^{m} \left\| \vec{p_i} - \vec{\hat{p}_i}(\theta) \right\|_2^2$$

Optimisation methods allow the determination of parameters which results in a local or global minimum of a multivariate function. They are iterative methods starting from initial parameters θ^0 assumed to obtain a value of f close to a minimum. They create a sequence of parameter iterates $\theta^0, \theta^1, \theta^2, ...$ such that $f(\theta^0) > f(\theta^1) > f(\theta^2)...$ in order to converge to the desired minimum.

Optimisation methods can rely on different approximation schemes of the function f at a current iterate θ^t. For instance, the order of the Taylor polynomial approximation of f near θ^t can differ. The first-order polynomial (i.e. linear) and the second-order polynomial (i.e. quadratic) approximations can be used, respectively:

$$f(\theta) \approx f(\theta^t) + \nabla f(\theta^t)^T(\theta - \theta^t) \tag{2.8}$$

$$f(\theta) \approx f(\theta^t) + \nabla f(\theta^t)^T(\theta - \theta^t) + \frac{1}{2}(\theta - \theta^t)^T H(\theta - \theta^t) \tag{2.9}$$

where the gradient (i.e. the first-order partial derivatives) of f at θ^t is $\nabla f(\theta^t)$ and the Hessian matrix (i.e. the second-order partial derivatives) is H.

Gradient descent. Gradient descent is an optimisation method for a differentiable multivariate function relying on the first-order Taylor approximation of f around the iterates θ^t, see equation 2.8. It consists in moving in the opposite direction of the gradient of f at θ^t, with a step size α^t, i.e. $-\alpha^t \nabla f(\theta^t)$ in order to decrease the error:

$$\theta^{t+1} = \theta^t - \alpha^t \nabla f(\theta^t) \tag{2.10}$$

Moving along this gradient direction is performed through a method called inexact line search which efficiently determines a step length α^t that results in a 'sufficient' decrease in the objective function value for each iteration t along the search direction $-\nabla f(\theta^t)$ [Nocedal & Wright 1999]. This 'sufficient' decrease can be characterised by the satisfaction of a criterion such as the Armijo condition [Armijo 1966]. Let the line search function be $g(\alpha^t) = f(\theta^t + \alpha^t d^t)$ with d^t the search direction (here $d^t = -\nabla f(\theta^t)$). The Armijo condition states that:

$$g(\alpha^t) \leq g(0) + c_1 \alpha^t g'(0)$$
$$f(\theta^t + \alpha^t d^t) \leq f(\theta^t) + c_1 \alpha^t \nabla f(\theta^t)^T d^t$$

The step size α^t should thus allow a reduction in f of at least the c_1-proportion (constant $c_1 \in (0, 1)$) of both the step length α^t and the directional derivative $\nabla f(\theta^t)^T d^t$) [Nocedal & Wright 1999].

This condition ensures that the step size α^t is not excessively large, but does not prevent α^t from being inadequately small. The Armijo backtracking line search strategy thus starts with a relatively large step size, and repeatedly decreases it by a factor $\gamma \in (0, 1)$ until the Armijo condition is fulfilled. An alternative consists in using an additional condition for ensuring that α^t is not unacceptably small, such as the curvature condition. This ensures that the derivative of the line search function at α^t is greater than c_2 times the initial derivative at $\alpha^t = 0$:

$$g'(\alpha^t) \geq c_2 g'(0)$$
$$\nabla f(\theta^t + \alpha^t d^t)^T d^t \geq c_2 \nabla f(\theta^t)^T d^t$$

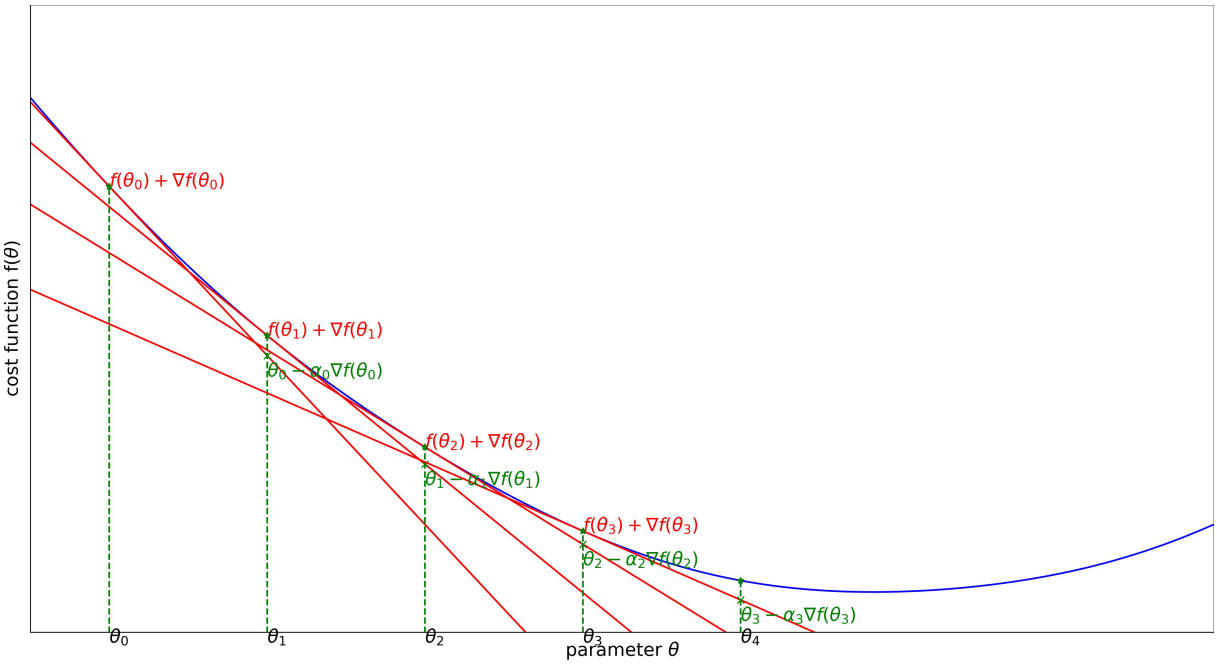
with $c_1 < c_2 < 1$. These two conditions are referred to as the Wolfe conditions [Wolfe 1969].

Newton. A sequence of second-order Taylor approximations of f around the iterates θ^t is used, see equation 2.9:

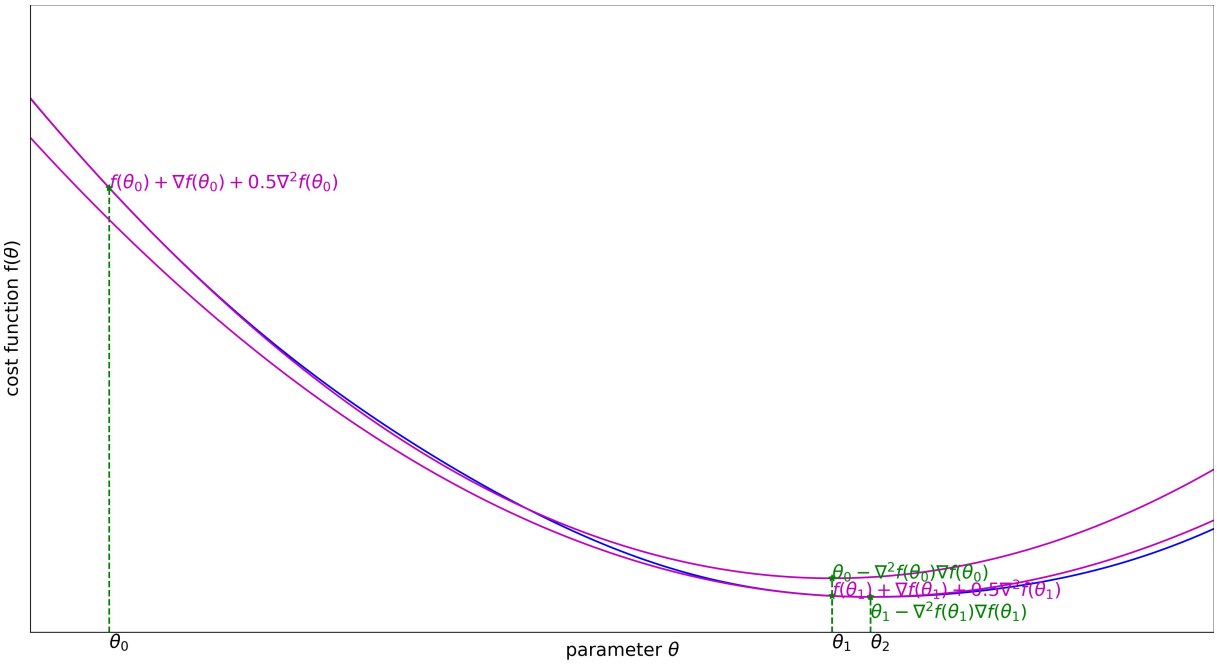
$$\theta^{t+1} = \theta^t - H^{-1} \nabla f(\theta^t) \tag{2.11}$$

For non-linear least squares, J is the Jacobian of the residuals with respect to the parameters:

$$J = \frac{\partial r}{\partial \theta} = \begin{bmatrix} \frac{\partial r_1}{\partial \theta_1} & \cdots & \frac{\partial r_1}{\partial \theta_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial r_m}{\partial \theta_1} & \cdots & \frac{\partial r_m}{\partial \theta_n} \end{bmatrix}$$



(a) Gradient descent



(b) Newton

Figure 2.15: Optimisation method principles (univariate case). Starting from an initial parameter θ_0, first-order and second-order polynomial approximations of the cost function are respectively performed for gradient descent and Newton methods. At each iteration (subscript index), a new iterate is estimated using respective formulas 2.10 and 2.11.

The gradient vector ∇f is defined as:

$$\begin{aligned}
[\nabla f]_j &= \frac{\partial f}{\partial \theta_j} \\
&= 2 \sum_{i=1}^{m} r_i \frac{\partial r_i}{\partial \theta_j} \\
\Rightarrow \nabla f &= 2 J^T r
\end{aligned}$$

The Hessian matrix H is thus:

$$\begin{aligned}
[H]_{kl} &= \frac{\partial^2 f}{\partial \theta_k \theta_l} \\
&= 2 \left(\sum_{i=1}^{m} \frac{\partial r_i}{\partial \theta_k} \frac{\partial r_i}{\partial \theta_l} + \sum_{i=1}^{m} r_i \frac{\partial r_i^2}{\partial \theta_k \partial \theta_l} \right) \\
\Rightarrow H &= 2 J^T J + 2 \sum_{i=1}^{m} r_i \nabla^2 r_i
\end{aligned}$$

Newton requires one to compute the second-order derivatives and does not always produce a descent direction (H is not necessarily positive definite). When applicable, it results in a fast convergence near a local minimum. Newton and gradient-descent optimisation principles are illustrated in figure 2.15.

Gauss-Newton. Gauss-Newton is dedicated to the optimisation of non-linear least square functions and consists in approximating H as $H \approx 2J^T J$ by neglecting the second-order derivative terms. This approximation holds for fast local convergence when the problem is only mildly nonlinear and the residual at the solution is small. From formula 2.11, this reduces to:

$$\theta^{t+1} = \theta^t - (J^T J)^{-1} J^T r$$

Note that this results in the normal equations:

$$J^T J \underbrace{(\theta^{t+1} - \theta^t)}_{\Delta\theta} = -J^T r$$

The linear least-squares problem:

$$\min_{\Delta\theta} \frac{1}{2} \|J\Delta\theta + r\|_2^2 \tag{2.12}$$

is solved for each iteration t. A line search in the direction of the solution is usually performed [Nocedal & Wright 1999, Sun & Yuan 2006] in order to improve the convergence of the method and is called the damped Gauss-Newton method:

$$\theta^{t+1} = \theta^t - \alpha^t (J^T J)^{-1} J^T r$$

Levenberg-Marquardt. Another alternative for non-linear least squares is the Levenberg-Marquardt algorithm which replaces the line search strategy of the Gauss-Newton method with a trust-region one. The linear least-squares problem, see equation 2.12, is constrained:

$$\min_{\Delta\theta} \frac{1}{2} \|J\Delta\theta + r\|_2^2 \text{ s.t. } \|\Delta\theta\|_2 \leq \nu^t$$

It defines a region with radius $\nu^t > 0$ around the current iterate within which it trusts the model to be an adequate representation of the objective function. The solution is characterised by solving the problem:

$$(J^T J + \lambda^t I)\Delta\theta = -J^T r \tag{2.13}$$

where I is the identity matrix and $\lambda \geq 0$ the Marquardt parameter, which simultaneously changes the search direction and the step length. When $\lambda^t = 0$, it reduces to the Gauss-Newton direction. Otherwise, there exists $\lambda^t > 0$ such that the solution satisfies equation 2.13 and $\|(J^T J + \lambda^t I)^{-1} J^T r\| = \nu^t$ [Sun & Yuan 2006].

This method is more robust than Gauss-Newton, in case of an ill-conditioned (rank-deficient) Jacobian and a singular $J^T J$, while local convergence properties of the two methods are similar. This is why Levenberg-Marquadt is mainly used for refining the intrinsics, extrinsics and distortion coefficients. Several implementations can be used [Moré 2006, Wright & Holt 1985].

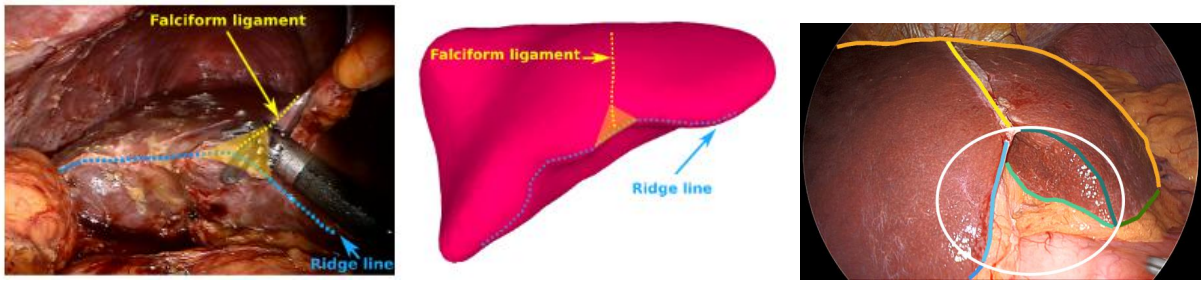
2.5 Corresponding Landmark Annotation

Registration is based on 3D/2D liver surface landmark correspondences, see figure 1.12c. These landmarks should thus be present both in 3D, i.e. annotated on the preoperative mesh, see figure 1.12a, and in 2D, i.e. annotated in the intraoperative mini-invasive images, see figure 1.12b.

2.5.1 Anatomical Surface Landmarks

[Plantefeve *et al.* 2014] introduced the anterior ridge, which delimits the frontal superior and inferior liver surfaces, and the liver junction with the falciform ligament. They are curvilinear liver surface landmarks. Both landmarks are visible for global frontal or fronto-lateral views. They are rarely entirely visible, owing to occlusions and self-occlusions. Previous works [Plantefeve *et al.* 2014, Özgür *et al.* 2018, Koo *et al.* 2017b, Espinel *et al.* 2020] model the ridge as a single curve, called the single ridge model. However, depending on the camera position relative to the liver, the central part of the ridge next to the ligament may correspond to different positions in the model.

This issue is illustrated in figure 2.16. Several annotations can be performed for the left central part of the ridge. This central part corresponds to the part surrounding the round ligament, close to the Rex recessus. This is the outer surface formed by the segment 4b and 3 closer to the intersection delimited by the frontal extremity of the ligament. This part varies substantially across patients. We propose to label both the upper and lower



(a) Single ridge, from [Plantefeve *et al.* 2014]. (b) Split ridge.

Figure 2.16: Ridge models. The central part of the ridge (white circle in (b)) may correspond to different positions in the model according to the camera viewpoint and was not taken into account in the previous single ridge model (a). We thus split it (b).

limits of this surface, for each side, right and left. We thus obtain the following landmarks for the central part:

- The upper-left central limit (may be visible from frontal left views and guessed from right fronto-lateral and frontal views)
- The upper-right central limit (may be visible from frontal right views and guessed from frontal left and frontal views)
- The lower-left central limit (may be visible from opposite lateral views)
- The lower-right central limit (may be visible from opposite lateral views)

We call this the split ridge model. Its most reliable landmarks are the right ridge and left ridge from each side of the central part. While annotated as a curve in 3D, see figure 2.18, the visible ridge edges depend on the viewpoint and can actually correspond to their surrounding surface. In both 3D and 2D, it can be difficult to accurately locate each part extremities. We illustrate the 2D delimitation in figure 2.17. Occluded landmark parts by fat, tools, blood or other elements are not annotated and only parts of the whole landmarks are usually visible in an image. Hence, 3D-2D correspondences are only partial and specific processes should be designed to allow one-to-one correspondences. They can be obtained through different ways and are explained in chapter 4.

Intraoperatively, the falciform ligament can be cut, which happens in the LLR procedures of our university hospital partners, or maintained. Thus, both cases should be taken into account and annotated accordingly in 2D. We choose to annotate the attached ligament junction from the side view, see figure 2.21. However, the falciform ligament is not visible in CT-scans due to the limited resolution, and its orientation from the Rex recessus cannot be determined and annotated with precision from the surface mesh. It should therefore be used with caution in the registration process, or discarded.

2.5.2 Silhouette

In computer graphics, in a 3D scene comprising a surface mesh made up of oriented polygons, silhouette edges are defined as the visible boundaries between adjacent front-facing and back-facing polygons, which are not occluded by the interior of any front-facing

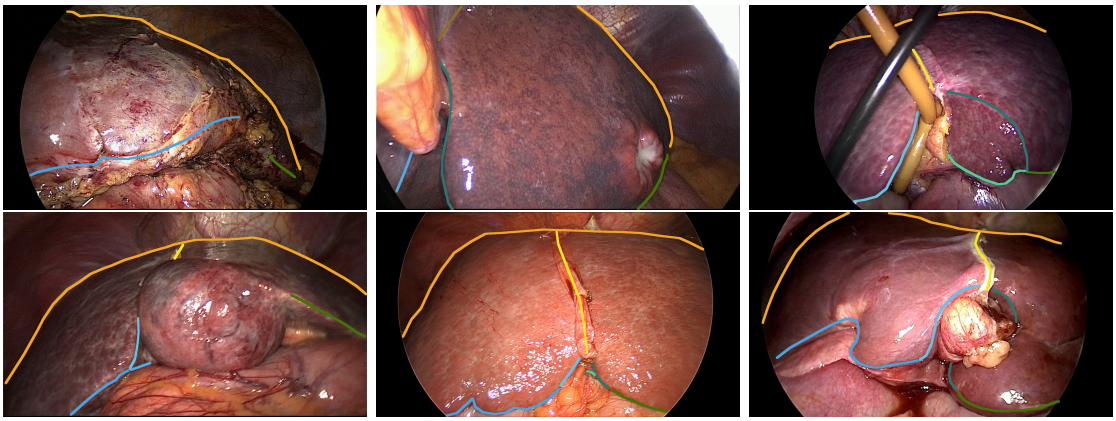


Figure 2.17: Mini-invasive image landmark annotations for different patients. The landmarks with their associated colours are: the silhouette, the junction with cut falciform ligament, the junction with attached falciform ligament, the left ridge, the right ridge, the upper-left central limit, the lower-left central limit, the upper-right central limit, the lower-right central limit.

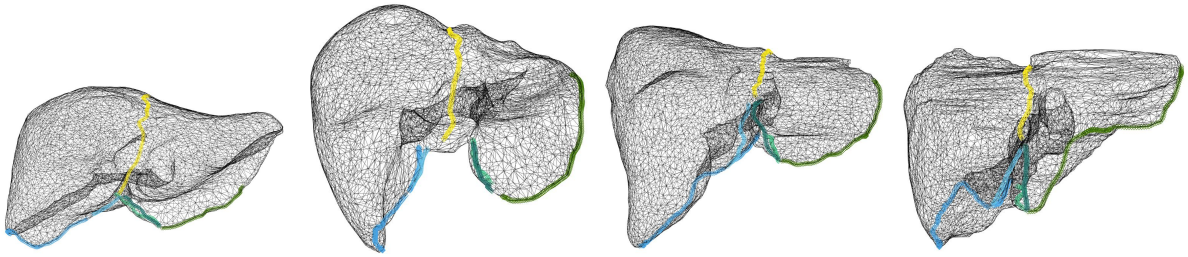


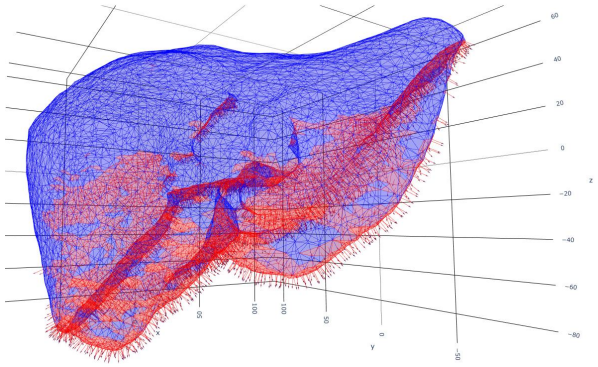
Figure 2.18: Landmark annotations on 3D liver models of different patients

polygons [Raskar & Cohen 1999]. Therefore, they depend on a given viewpoint, as faces can be front-facing or back-facing depending on the viewpoint. They can also be thought as occlusion boundaries, where parts enclosed by the boundaries occludes further parts from the observer viewpoint.

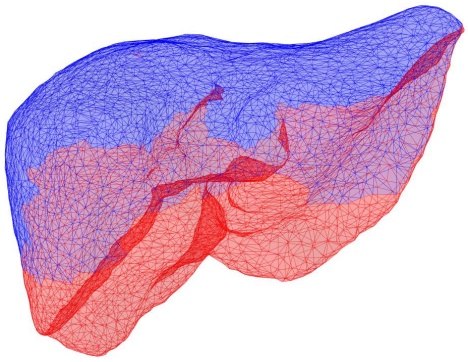
Silhouette rendering consists in rendering the visible parts enclosed by the silhouette edges, or occlusion edges, of an object mesh and producing a silhouette image, illustrated in figure 2.20a. We use pytorch3d with Graphics Processing Unit (GPU) in order to perform an efficient and differentiable silhouette rendering. It depends on the mesh, as well as the camera intrinsic and extrinsic parameters input at the considered registration step. It outputs for each pixel of the rendered image where the liver is present the nearest visible face index of the pixel centre, with its associated barycentric coordinates.

In practice, the upper liver is the visible part in most of the mini-invasive camera port positioning and imaging, when no tools are manipulating the liver. The frontal silhouette part may be confounded with the ridge parts, see figures 2.17 and 2.21. We thus choose to only select the upper liver silhouette, and discard the lower one in addition to the one close to the ridge parts. This selection is performed in several steps:

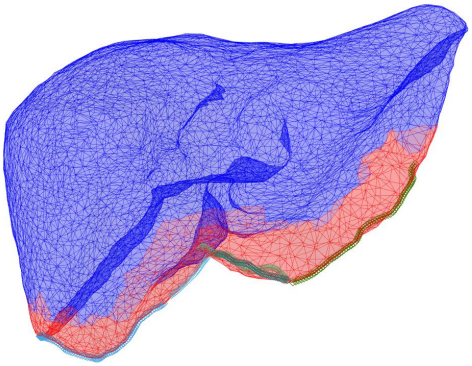
1. From the surface mesh in preoperative (*world*) space, we determine the normalised face normal vectors. A first threshold on the normals $Z_w < -0.4$ selects the faces whose normals mainly points downwards, see figure 2.19a for an example.



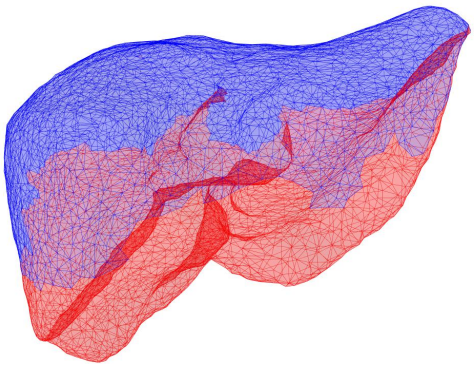
(a) Face normals are z-thresholded. The discarded ones are displayed with magnification.



(b) Morphology closing, which extends the discarded area from a).



(c) Face neighbours of the split ridge (opened diamonds) are discarded.



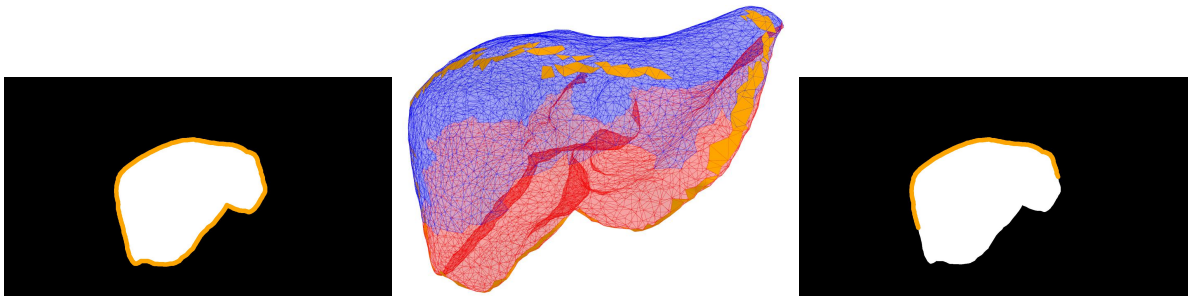
(d) Final selection of potential silhouette faces: the union of b) and c) is discarded.

Figure 2.19: Steps of the selection process of the potential silhouette faces and vertices. In blue: the potential silhouette faces at the considered step; in red: the discarded faces.

2. The surface formed by these faces is extended using graph morphology. The surface mesh is first converted into a graph where face indices are nodes and are connected to adjacent edges by an edge of weight 1. A binary attribute of 1 is given for selected faces from step 1 and 0 for other faces. Morphological closing (dilation followed by erosion) or order 5 is then performed, using thatNode. Thus the faces which were forming 'holes' in the first lower selected surface are gathered. All these faces are discarded from the potential faces of the upper liver silhouette (figure 2.19b).

3. Using networkX, the faces which are less than 3 edges distant from the faces englobing the ridge from the graph are retrieved. They are also discarded from the potential silhouette faces of the upper liver silhouette, illustrated in figure 2.19c.

For extracting the 3D silhouette points, we extract the 3D points (using barycentric coordinates and vertex indices) corresponding to the contour of the rendered surface and only keep the points belonging to faces previously selected as the potential upper liver silhouette ones, see figure 2.20. We determine among them the ones which are the nearest neighbour of each annotated silhouette curve pixel in the image in order to obtain one-to-one correspondences between 3D and 2D silhouette points.



(a) Initial silhouette from rendering and contour extraction.

(b) Silhouette in 3D and its filtering (Figure 2.19).

(c) Silhouette in 2D after filtering.

Figure 2.20: Silhouette rendering, extraction and filtering

2.5.3 View Selection

Mini-invasive image selection should retain the views where most of the landmark parts are visible, i.e. frontal or fronto-lateral global views. It is manually performed but could be automatically performed from automatic and real-time annotation (chapter 3). It would require one to define a selection criterion from the annotated landmarks, such as the number of visible landmarks and the proportion of the image that they represent, see Figure 2.21.

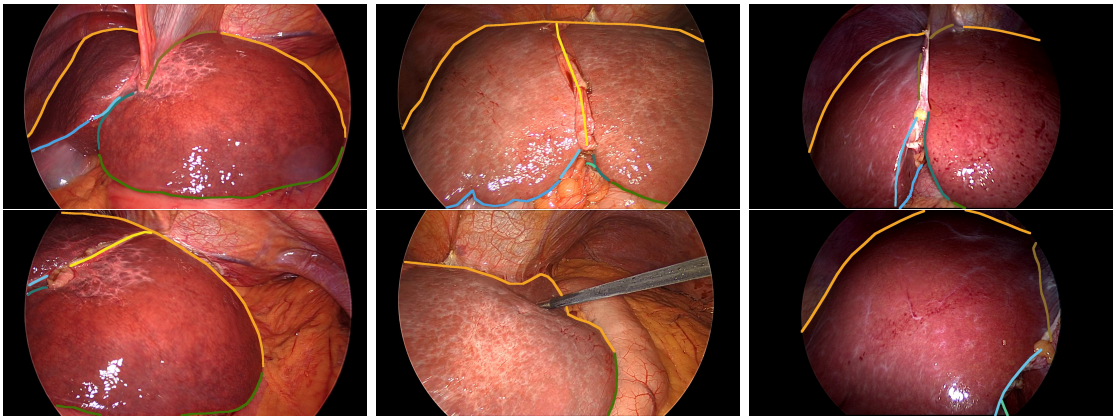


Figure 2.21: View selection. The top row shows views which should be preferred to the bottom row ones, as more landmarks are visible, and the views are more 'global', allowing more registration guidance.

2.5.4 Annotation Tools

For 3D landmarks, interactive annotation tools such as MeshLab or custom ones based on Plotly or OpenGL can be used. Vertex indices can be selected and form a curve which is resampled to get 4 points a mm, smoothed using a Savitzky-Golay filter of kernel size 11 in each dimension. Then the landmark points are obtained by taking the closest surface points which are described by barycentric coordinates and their associated face vertices, see section 2.2.5.

Manual 2D annotation of the images is performed through a dedicated tool such as Supervisely. For each curve part, the user must place points which are connected to form

a polyline. From these curves, we can obtain segmentation masks where all pixels have a label, from background to silhouette.

2.6 Introduction to Neural Networks

With the perspective of automating the 3D/2D registration of the liver using deep neural networks, we introduce them in this section. An artificial NN is a model that attempts to simulate the structure and functionalities (such as learning) of biological neural networks [Krenker *et al.* 2011]. It is a combination and an interconnection of artificial neurons, organised into layers, as illustrated in figure 3.4a. A single neuron is a non-linear function summing weighted inputs (inputs $x \in \mathbb{R}^n$ and weights $w \in \mathbb{R}^n$) and a bias b before passing the sum through an activation function F:

$$y(x; w, b) = F\left(\sum_{i=0}^{n} w_i x_i + b\right) \tag{2.14}$$

The activation function F is a non-linear function such as the sigmoid function $F(x) = \frac{1}{1+e^{-x}}$ and the Rectified Linear Unit (ReLU) function $F(x) = \max(0, x)$. y is a single output.

A NN uses the fact that complexity can grow out of merely few basic and simple rules, modelled by each neuron [Krenker *et al.* 2011]. The weights and biases from all the neurons form the parameters of the model for approximating the actual complex underlying function which relates inputs to the outputs [Basheer & Hajmeer 2000]. These parameters should be determined from related input–output data. This is referred to as NN training, i.e. learning the relevant model parameters for the requested task relating inputs to outputs. Training an artificial neural network architecture of numerous layers and neurons is referred to as deep learning.

Biological NNs are progressively trained on multiple experiments and situations in order to produce the desired response from a specific input, considering the differences between the produced response and the target one. For instance, learning to grasp an object is a progressive process which include many trials and errors in numerous environmental conditions and configurations before actual repetitive success in most of not experienced cases. In the same way, artificial or even deep NNs need numerous related input-output data in order to learn performing or generalising the task successively in many unseen cases.

Parameters of a single artificial neuron can be optimised using gradient descent, see section 2.4.5 and equation 2.10. This requires an initialisation of the parameters and a cost or loss function computed between the target outputs and the outputs predicted from the parameter iterates. Optimising all parameters from all neurons in feedforward NNs, where information propagates, without loops, from the inputs to the outputs through any network layers, can be done with backpropagation. At each iteration, a forward activation first predicts outputs from the whole network and all parameter iterates. Then, it consists in propagating the error or loss in a backward manner to modify the parameters, from the last layer of neurons connected to the output, to the first layer whose inputs are the actual data inputs. First, the loss or cost is obtained between predicted and target outputs. Then the partial derivatives of the loss with respect to the parameters of the

neurons of the last layer are computed. Using the chain rule, the partial derivatives of the loss with respect to the backward connected neuron parameters can be computed and so on up to the first layer neurons. Gradient descent is performed for updating all the parameters.

There are three variants of gradient descent, which differ in how much data are used to compute the gradient of the loss function. The most used is the mini-batch gradient descent, which gives a trade-off between the accuracy of the parameter update and the time it takes to perform an update [Ruder 2016]. Instead of computing the gradient of the cost function with respect to the parameters θ for the entire training dataset (equation 2.10), it performs an update for every mini-batch $b \in \{1, ..., n\}$ of n among N training examples:

$$\theta^{t+1,b} = \theta^{t,b} - \alpha^t \nabla f(\theta^{t,b}; x^b, y^b)$$

This way, it allows one to process datasets that could not fit in memory as a whole, it reduces the variance of the parameter updates, which can lead to more stable convergence, and makes use of highly optimised matrix optimisations common to state-of-the-art deep learning libraries that make computing the gradient with respect to a mini-batch very efficient [Ruder 2016]. Another major alternative is the Adam optimiser, derived from adaptive moment estimation. This method computes individual adaptive learning rates for different parameters from estimates of first and second moments of the gradients [Kingma & Ba 2014].

Neurons can be organised into several layers (section 3.2.1), combined into different network architectures (section 3.2.3), for a same set of inputs-outputs. A neural network training will succeed in generalising the task to unseen data according to many factors. For instance, these ones should be taken into account:

- The way of representing and normalising data. For instance, normalising or standardising pixel values in the range $[0, 1], [-1, 1], [-0.1, 0.1]$, in conjunction with the choice of the neural activation functions, will result in different outputs. Preprocessing data, e.g. resizing, smoothing, ..., also has an influence on the training results.
- The number and partitioning of the data. Training on a medium dataset with various cases which span the target domain adequately should better generalise to unseen cases than a large dataset representing close cases. However, collecting a huge amount of data of various cases is the most adequate, when feasible. Augmenting, balancing, and enriching data could help training when augmentations represent realistic variations of the data and the initial dataset is small. Geometric transformations (translation, rotation, elastic deformations, ...) or appearance transformation (modifying colour scale, blurring, ...) can be used for instance.
- The learning rate (step size α^t), the number of epochs (iterations) and other optimisation parameters. These are essential elements of the training performance.
- The network parameter initialisation. Optimisation is sensitive to initial parameters.

For evaluating the performance of a neural network, the whole dataset should be split into training, validation and test sets. The training set is used for updating the

parameters, the validation set determines which iteration (epoch) the training should be stopped, i.e when the generalisation error on this set is the lowest. The test set ensures that the network generalises well to unseen data without biases.

2.7 Conclusion

Except neural networks implied in another learning-based registration pipeline, the presented steps are essential blocks of the registration baseline pipeline. They can be performed with different methods. For each step, characteristics and future works are:

- Liver and inner structure segmentation of the preoperative volume. This is a preoperative step which is essential in order to extract the preoperative position of the liver and of the inner structures, with an accuracy depending on the imaging resolution. Instead of manual segmentation which was performed in the last decades, automatic or semi-automatic segmentation using deep learning, currently in progress, would allow a surgeon or radiologist to obtain and correct results in an efficient manner.

- Surface mesh reconstruction and processing. This is a preoperative step allowing an efficient representation of the liver and inner structure surfaces as meshes, easing rendering and registration computation. It relies on several steps, comprising isosurface extraction from the segmented volume, surface mesh smoothing and resampling as well as cleaning. Each step can be performed with several methods. While deep learning methods are envisaged for this problem, the current existing methods seem sufficiently convenient in the 3D-2D registration framework for MILS.

- Volumetric mesh reconstruction and inner structure representation. These are automatic preoperative steps for representing the liver volume as a mesh and associating the inner structures to it, enabling deformation and registration computation. The current existing methods also seem sufficiently convenient in the 3D-2D registration pipeline for MILS.

- Deformation modelling. This a preoperative step which can rely on multiple assumptions, e.g. geometrical or biomechanical ones. The modelling can be based on a matrix of multiple simulations whose dimension is reduced in order to form a linear model. Accurate biomechanical simulations are very challenging because of:

  – The current impossibility to obtain in-vivo biomechanical parameters. Development and progress of preoperative elastography, see section 1.2.3, would ease the determination of biomechanical parameters and of the adequate material model. Currently, we use generic parameters obtained from ex-vivo liver tissue.

  – The requirement of segmenting many intra-abdominal structures for pneumoperitoneum simulations and lungs and diaphragm for breathing, where each one should have defined biomechanical parameters.

  Simulations can also be performed using free form deformation of geometric models. The previous method modelled the liver directly from models based on local linearity assumptions.

The evaluation of the deformation model is very challenging due to the lack of data of 3D deformation inside the liver in mini-invasive conditions. Without surgical tool loads, pneumoperitoneum and breathing would require hybrid rooms, with both a CT scanner, and the equipment for insufflating gas through a trocar port after a mini-invasive incision of the patient. A recent technique named 'artificial pneumoperitoneum CT' [Wang *et al.* 2021] uses this environment and seems promising for providing such data. This would also allow one to model the deformation directly from data, and not relying neither on strong assumptions on liver biomechanics nor segmentations of many intra-abdominal structures. However, this technique cannot be used in all operating theatres due to the requirement of specific hybrid ones.

- Mini-invasive camera calibration. This is an intraoperative step for obtaining the camera projection parameters (intrinsics) which relates the image to the camera space. We use Zhang's method [Zhang 2000]. It is based on the optimisation of the parameters for minimising the distances between corresponding 3D pattern points (from a planar board) and the corresponding detected 2D ones, from multiple views of the planar board. Rigid transformation parameters of the board to place it in the camera space are dissociated from the intrinsics thanks to the multiple views. Calibration is performed under the assumption that camera parameters are not going to change. Thus, the surgeon should not change the parameters, e.g. zoom, after calibration and should determine the optimal parameters in order to image global views of the liver before calibration. This assumption may not hold for robotic endoscopes where they are automatically adapted or when a surgeon accidentally modify parameters. Thus, continuous calibration would be required in this case. However, this is very challenging. It could require patterns present intraoperatively in the abdominal cavity [Cui *et al.* 2023]. It should be investigated in future works as this issue seems critical.

- Annotation of the corresponding 3D/2D registration landmarks. Annotation of the 3D landmarks is a preoperative step while corresponding 2D landmarks are annotated intraoperatively. Anatomical landmarks include the split ridge and falciform ligament. While the falciform ligament is visible in the intraoperative images, it is not visible in the preoperative imaging and cannot be determined accurately on the liver 3D mesh. In addition, another non-anatomical landmark is the liver upper silhouette (occlusion boundary), depending on the viewpoint. The silhouette can be determined automatically according from the viewpoint, the position of the mesh in the camera space and the camera intrinsics.

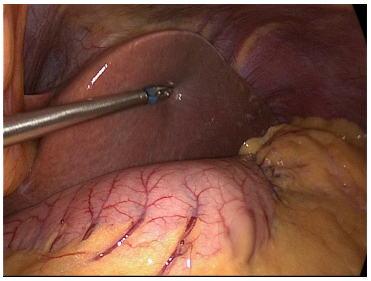


Figure 2.22: Case where our registration landmarks are not adapted.

Due to these registration landmarks, our registration process would be more adapted for the localisation of anterior inner structures or ones close to the upper liver surface. Indeed, for operating posterior inner structures close to the lower liver surface, the surgeon needs to raise the liver, and only the ridge elements would be visible and would occlude the further parts of the liver, while fat or lower organs such as stomach would occlude most of the rest of the silhouette, as illustrated in figure 2.22.

Annotation of the landmarks on mini-invasive images can be performed manually. However, we attempt to automate it, and propose solutions in the next chapter (3).

Chapter 3

Automatic Laparoscopic Image Landmark Prediction

Contents

3.1 Introduction

Detecting the landmarks (including the upper silhouette) in a mini-invasive image can be approached in different ways, mainly as:

- A semantic segmentation task: classifying each pixel in an image according to predefined classes. In the problem at hand, classes are the landmarks and the upper silhouette, see section 2.5, as well as the background (e.g. other elements). Nowadays, semantic segmentation is quasi-systematically performed by deep learning methods, which take whole image pixel values as inputs and outputs concatenated binary segmentation masks, one for each class, where 1-value pixels are given for the class elements, unlike 0-value ones.

- Iterative fitting of deformable models to image information (appearance) models. Building a 2D deformable (shape) model of each landmark or one combining all the landmarks is first required. A Point Distribution Model can be used [Cootes *et al.* 1992]. From landmark instances annotated on multiple training images, where all landmark curve instances are represented by the same number of points, a PCA is performed to obtain a statistical shape model. The appearance models should be built from the training images and related to the shape model elements. Multiple methods exist, focusing on different image support with respect to the landmarks. For instance, for each point, image intensities along the normal of the landmark curve could be modelled at several scales [Cootes *et al.* 1995]. Instead of only considering the normal, statistics of local patches centred on landmark curve points could be modelled [Cristinacce & Cootes 2006]. Alternatively, for contour landmarks forming a closed surface, one could focus on the inside of the surface. In this aim, each training image can be warped so that the landmark points match those of the mean shape, obtaining a shape-free patch whose statistics can be modelled [Cootes *et al.* 1998]. Appearance models could also built using deep learning [Lombardi *et al.* 2018].

Reliably guiding the 3D-2D registration task is the aim of this detection. Thus, only visible landmark parts should be detected. Managing the occlusion of deformable models is challenging, as a model forms a connected structure, in contrast to semantic segmentation, where pixels are classified individually. In addition, combined shape and appearance modelling is difficult due to the high variation of the liver texture among patients as well as the partial views of the landmarks. These are the main reasons for which we initially choose to tackle the detection problem as a semantic segmentation task, using deep learning.

Deep segmentation NNs are introduced in section 3.2, with their usual layers, including attention mechanisms, and the architectures that we implement, together with the networks used in related works. We train and evaluate the networks on datasets described in section 3.3.

3.2 Image Segmentation using Deep Learning

3.2.1 Segmentation Network Layers

Fully-connected (or densely connected) layers connect every neuron of a layer to every neuron of adjacent ones, see figure 3.4a. Therefore, every input of a layer can influence every input of the next fully-connected layer. Then, if the input size is large, the computation can be very costly and limits its applicability. This is the case for image pixel values, which form a matrix of size $H \times W \times C$, with H, W the height and width of the image and C the number of channels, also referred to as depth. In addition, they can be considered as 'structure agnostic' as they do not take into account the spatial structure of the data. Thus, it is generally not used in the first layers of the segmentation networks.

In contrast, the neurons within a convolutional layer only connect to a small region of the previous layer (referred to as the receptive field). Convolutional layers use multiple 2D spatial filters referred to as kernels which are usually small, but spread along the

depth of the input. This means that one filter, of size $m \times n \times C$, with small m, n, is itself composed of multiple 2D filters specific to each input channel. A layer convolves each filter across the first 2 dimensions of the input. For each filter, this results in activation maps for each channel, which are summed along the channel dimension to produce a unique 2D activation map. This allows pattern or feature recognition and thus activation maps are also referred to as feature maps. The feature maps from every kernel are stacked along the channel dimension to form the full output volume from the convolutional layer [O'shea & Nash 2015].

Image convolution with a predefined kernel or filter K is used in various other image processing tasks such as smoothing (e.g. Gaussian filter), edge detection (e.g. Sobel filter) and texture analysis (e.g. Gabor filters). Convolutional layers in deep learning aims to automatically learn the adequate filters for the required task. The learnt filters can be relevant for multiple tasks such as object detection, image classification and segmentation and for different kinds of data. This is why initialising network parameters from parameters learnt on other datasets or tasks with a common architecture can be very valuable for networks based on convolutions, i.e. Convolutional NNs.

For a 2D matrix I, the convolution operation is defined by:

$$S(i,j) = \sum_{k=-(m-1)/2}^{(m-1)/2} \left(\sum_{l=-(n-1)/2}^{(n-1)/2} I(i-k, j-l) K(k,l) \right)$$

where the 2D kernel K is centred (pixel coordinates $(0,0)$ is at the centre of the image) and of odd dimensions m, n. In practice, many implementations use the cross-correlation operation instead, due to its computational efficiency. In cross-correlation, the kernel does not need to be flipped with respect to the input and this thus corresponds to the dot product of the local receptive field and the centred kernel (filter matrix) K, as:

$$S(i,j) = \sum_{k=-(m-1)/2}^{(m-1)/2} \left(\sum_{l=-(n-1)/2}^{(n-1)/2} I(i+k, j+l) K(k,l) \right) \tag{3.1}$$

Figure 3.1 illustrates the convolution operation. In convolutions, strides are the number of shifts of the kernel along the input rows and columns for each step. Zero-padding at the border of the input may be performed in order to maintain the shape of the input layers. Convolutional layers usually use the ReLU activation function.

Convolutional NNs are mainly a succession of convolutional and pooling layers. The latter reduce the spatial dimension of the representation gradually and thus further reduce the number of parameters and the computational complexity of the model [O'shea & Nash 2015]. Pooling also enables the convolution of the same kernel size at different resolutions, allowing the network to extract features (of other features) at multiple scales. Pooling is usually performed with strides $s_x = 2, s_y = 2$ in order to downsample the height and width of the feature maps by 2, and thus its spatial dimension (HW) by 4, while maintaining the channel dimension. It can be achieved through maximum pooling:

$$S(i,j,c) = \max_{k,l} I(is_x + k, js_y + l, c) \; \forall \; k \in \{0, ..., s_x - 1\}, l \in \{0, ..., s_y - 1\}$$

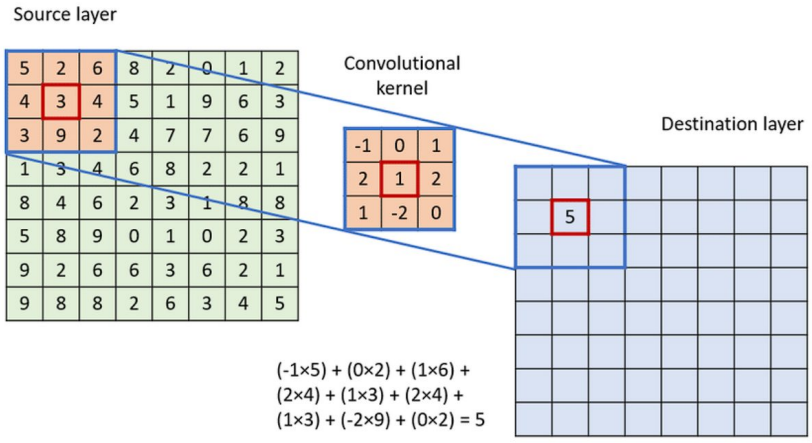


Figure 3.1: Convolution principle in deep learning (equivalent to cross-correlation), from [Podareanu *et al.* 2019].

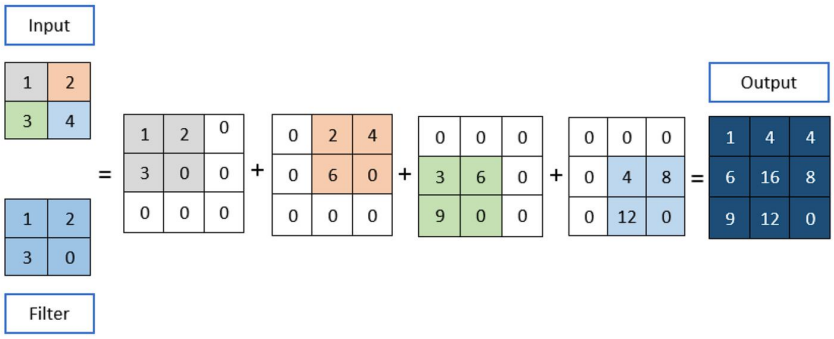
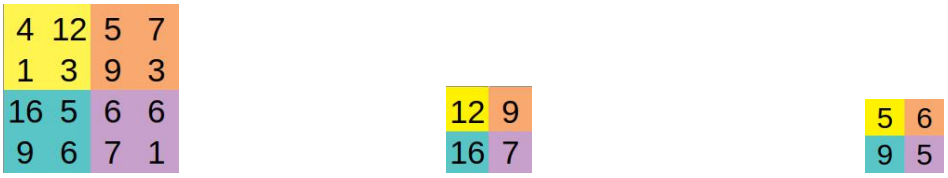


Figure 3.2: Transposed convolution principle for upsampling, from [Al Mamun & Kadir 2020].



(a) 2D input (b) Maximum pooling output (c) Average pooling output

Figure 3.3: Illustration of pooling with strides $2, 2$ for downsampling.

or average pooling:

$$S(i,j,c) = \frac{1}{kl} \sum_{k=0}^{s_x-1} \sum_{l=0}^{s_y-1} I(is_x + k, js_y + l, c)$$

where I is an image or feature map, and i, j, c are respective indices of height, width and channel dimensions. They are illustrated in figure 3.3.

From reduced dimension feature maps, progressive upsampling can be required in order to produce a segmentation mask at the same resolution as the input image. Upsampling can be performed using different layers:

- The upsampling layer, without trained parameters, which consists in resizing the image with a user-defined interpolation mode, such as the bilinear or bicubic ones.

- The transposed convolutional layer, based on the transposed convolution illustrated in figure 3.2. It uses the convolution principle with kernel size and number parameters as well as stride and padding ones but attempts to perform the process backwards, i.e. from S to obtain I in equation 3.1:

$$I(i,j) = \sum_{k=-(m-1)/2}^{(m-1)/2} \left(\sum_{l=-(n-1)/2}^{(n-1)/2} S(i-k, j-l) K(k,l) \right)$$

  In 2D, for a given upsampling size, this is achieved by summing p intermediate results of the upsampled output I where p is the number of elements in the input S. Unlike convolution where padding is applied to the input, padding is applied to the output for transposed convolution. Each intermediate matrix result I_t is initialised as zeros and the corresponding element $S(i,j)$ is multiplied by the kernel so that the resulting matrix replaces the corresponding receptive field in the intermediate one, as illustrated in figure 3.2.

Other layers can be used for easing the training:

- The dropout layer [Hinton *et al.* 2012]. It consists in randomly dropping some neurons from the neural network during training, along with their connections, see figure 3.4, in order to prevent units from co-adapting too much and thus reduce overfitting. At inference, no dropout is performed and all neurons are used.

(a) A feedforward network with fully connected layers (used at inference).

(b) The network after dropout (used in training). Neurons are randomly dropped at each epoch.

Figure 3.4: Dropout illustration, adapted from [Srivastava *et al.* 2014].

- Batch normalisation [Ioffe & Szegedy 2015]. During training, for a layer, the parameters of the previous layers change and this changes the distribution of the layer inputs, referred to as the internal covariate shift. This makes the training harder, requiring for instance lower learning rates and adequate parameter initialisation. The problem can be alleviated by normalising the layer inputs for each training mini-batch, as a part of the model architecture.

Batch normalisation with μ_b, σ_b the batch mean and variance and ε an arbitrarily small constant for numerical stability, can be expressed as:

$$\hat{x}_i^t = \frac{x_i^t - \mu_b^t}{\sqrt{(\sigma_b^t)^2 + \varepsilon}}$$

However, at inference, batch normalisation uses the running statistics (mean and variance) computed during training.

3.2.2 Attention Mechanisms

One of the limits of the convolution is its limited receptive field. This is not the case of attention [Vaswani *et al.* 2017]. In computer vision, it consists in diverting attention to the most important regions of an image and disregarding irrelevant parts. The idea is to use non local operations in order to capture long-range dependencies, which compute the response at a position as a weighted sum of the features at all positions [Wang *et al.* 2018]. This relates to non-local means in computer vision for image denoising. For each pixel, it consists in taking a weighted mean of all pixels in the image, weighted by how similar these pixels are to the target one, according to a predefined affinity function. An attention mechanism is a dynamic selection process that is achieved by adaptively weighting features according to the importance of the input. This non-local operation can be described by this general equation:

$$y_i = \frac{1}{c(x)} \sum_j f(x_i, x_j) g(x_j) \tag{3.2}$$

where x is the input. It is generally a feature map of size $NCHW$ or $NTCHW$, where N is the number of batch elements, T the number of temporal elements, C the number of channels, H the height and W the width. y is the output of same size as x, f is a pairwise function for computing affinity, g is an unary function for representing the input and c a normalisation factor. i are 'query' indices and j are 'key' indices.

g can be a convolutional layer or a linear embedding with a weight matrix to be learnt and its output can be referred to as 'values' (related to keys):

$$g(x_j) = W_g x_j$$

For computing affinity between the query and keys, f can take many forms. For instance some use specific embedding for x_i and x_j:

$$\theta(x_i) = W_\theta x_i$$
$$\phi(x_j) = W_\phi x_j$$

- Dot product:

$$f(x_i, x_j) = \theta(x_i)^T \phi(x_j)$$
$$c(x) = N$$

- Embedded Gaussian for one head attention in multi-head self-attention [Vaswani *et al.* 2017]:

$$f(x_i, x_j) = e^{\theta(x_i)^T \phi(x_j)}$$
$$c(x) = \sum_j f(x_i, x_j)$$
$$\implies f(x_i, x_j) = \text{softmax}(x_i^T W_\theta^T W_\phi x_j)$$

  Using multiple heads, see figure 3.5, follows the same idea as using multiple kernels in convolutional layers, allowing the model to learn various affinity rules.

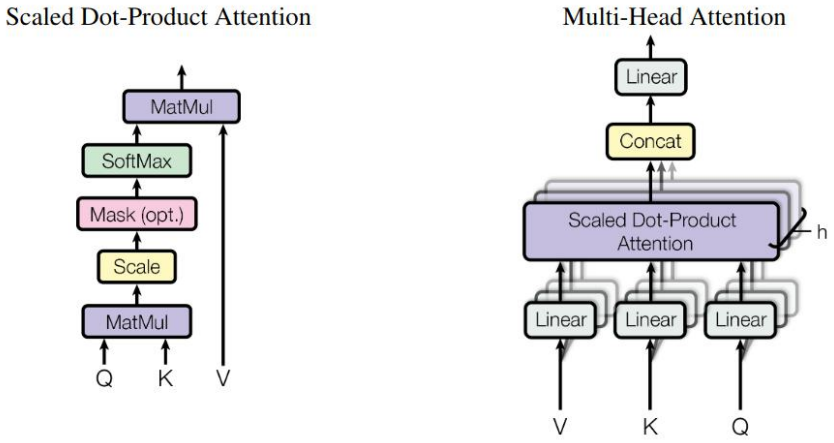


Figure 3.5: Self-attention mechanism from [Vaswani *et al.* 2017]. Q, K, V respectively states for queries, keys and values, and MatMul for matrix multiplication.

A non-local operation, see equation 3.2, can be inserted into any pretrained NNs as a residual connection or a non-local block [Wang *et al.* 2018]:

$$z_i = W_z y_i + x_i$$

This ensures that its initial behaviour is maintained (if W_z is initialised as zeros). The non-local block is lightweight when it is used in downsampled feature maps of low dimensions but is costly otherwise.

The domain of attention depends on the set of key indices j. It can be among :

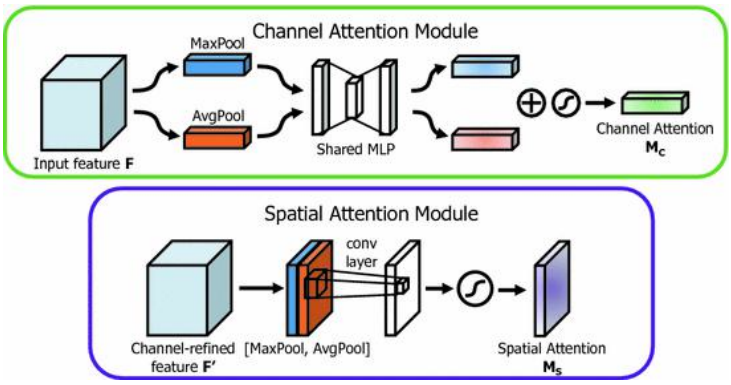
- all channels (C): channel attention (what to pay attention to)
- all spatial elements (HW): spatial attention (where to pay attention)
- all temporal elements (T): temporal attention (when to pay attention)
- a combination of previous elements such as both channel and spatial elements (CHW) or all spatio-temporal elements (THW) [Guo *et al.* 2022b]

In the problem at hand, attention mechanisms in both channel and spatial domains are of first interest. We focus on two mechanisms highlighted in a survey [Guo *et al.* 2022b]:

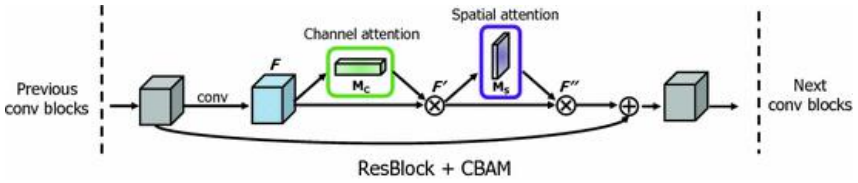
- The convolutional block attention module (CBAM) [Woo *et al.* 2018] stacks channel attention and spatial attention in series, see figure 3.6.

  The channel attention module uses a Squeeze-And-Excitation block [Hu *et al.* 2018], with 2 parallel poolings: average and max ones. The squeeze module initially applies the pooling in order to get a mean or maximum value per channel. The excitation module captures channel-wise relationships and outputs an attention vector by using two successive fully-connected layers with activation functions (ReLU and sigmoid) in order to learn non-linear interaction between channels. Both parallel outputs are summed, then input to the sigmoid in order to generate the attention map.

  The spatial attention module stacks max and average poolings in order to get a mean value per spatial element. It then applies a convolutional layer with a large kernel followed with a sigmoid to generate the attention map. Because of the convolution, the spatial module may suffer from a limited receptive field [Guo *et al.* 2022b].



(a) Channel and spatial attention modules, from [Woo *et al.* 2018].



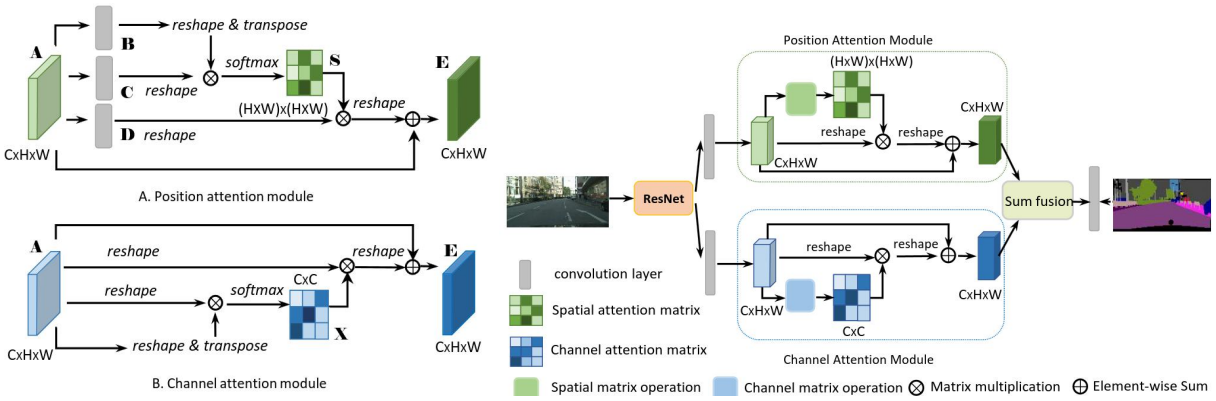
(b) Attention module combination, from [Woo *et al.* 2018].

Figure 3.6: Convolutional Block Attention Module.

- Instead of being stacked in series, for Dual Attention [Fu *et al.* 2019], the spatial (position) attention and the channel attention blocks are performed in parallel and then the results are fused with a sum, see figure 3.7. It adopts a self-attention mechanism to compute both, switching the set of indices i, j between channel and spatial dimensions. However, for the channel attention module, the features are directly used as inputs to model cross-channel relations, while for the spatial attention module, it deals with outputs from intermediary convolutional layers. Dual Attention can be computationally costly, especially for large input feature maps.

3.2.3 Segmentation Network Architectures

Segmentation inputs are usually images (e.g. N RGB images of size C_0HW) and outputs are segmentation masks (e.g. N masks of size C_nHW). A segmentation network is usually



(a) Position and channel attention modules, from [Fu *et al.* 2019]. (b) Attention module combination, from [Fu *et al.* 2019].

Figure 3.7: Dual attention

an encoder network followed with a decoder network. The task of the encoder is feature or latent space extraction of reduced dimensions, while the task of the decoder is to semantically project the discriminative features (lower resolution) learnt by the encoder onto the pixel space (higher resolution) to get a dense classification. Thus, the encoder represents a downsampling path while the decoder represents an upsampling path, see figure 3.8a.

The encoder can be common to networks dedicated to other image-related tasks, such as classification or object detection. Thus, a pretrained network from another task or another dataset can be used. In contrast, the decoder network is more specific to the segmented object type and problem, and its parameters cannot be initialised from pretrained networks dedicated to other tasks.

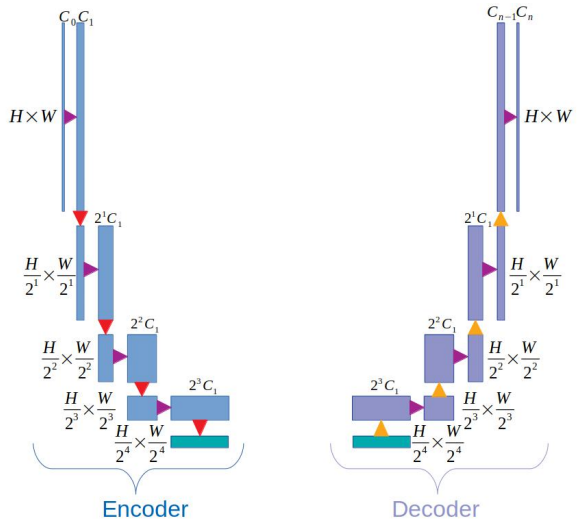
We implement several encoder-decoder networks with different characteristics and which make use of different information. They are presented in the following sections.

3.2.3.1 UNet

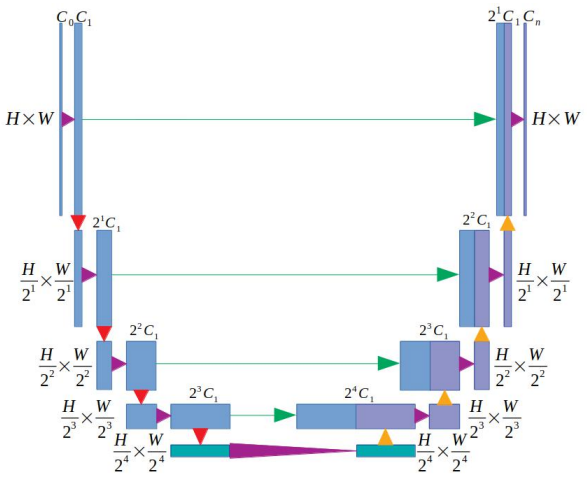
Since upsampling is a sparse operation, a good prior from earlier stages can be needed in order to improve the localisation representation, and this can be provided by higher resolution feature maps from the encoder network. This is the idea of UNet [Ronneberger *et al.* 2015] where the decoder combines upsampling and concatenation from intermediate features of similar resolution from the encoder. The direct connections between the encoder and decoder layers are named skip connections, see figure 3.8b.

This improves the representation learning with following convolutions and the performance of deep NNs even using small datasets. They are particularly used in medical image segmentation tasks.

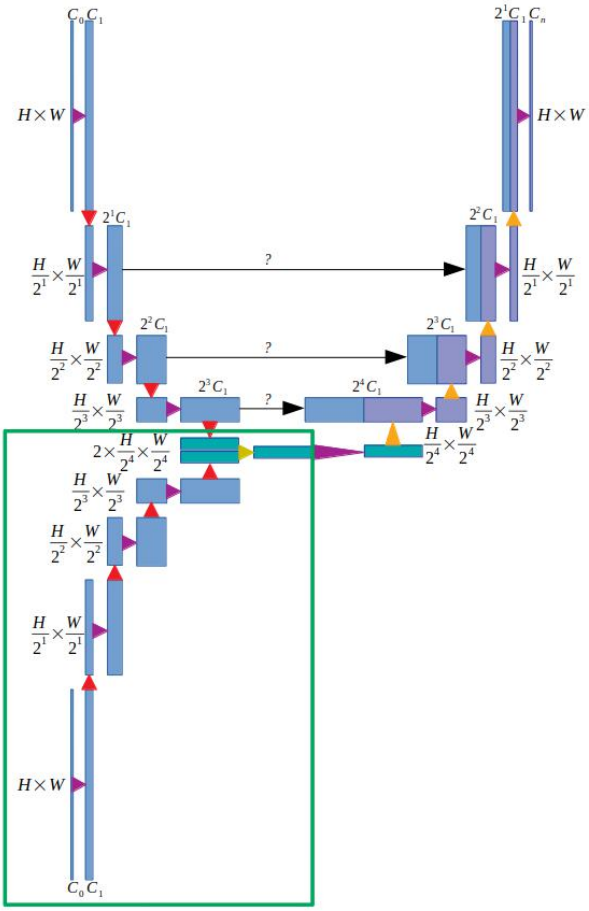
In details, the encoder comprises 4 blocks of successive convolutional layers followed with batch normalisation and max pooling layers which downsample the feature maps with strides $2, 2$, and thus each block reduces the spatial dimension by 4. This create feature map outputs of different resolutions. The feature maps of the i^{th} resolution between 1 and 5 are of successive sizes : $2^{i-1}C_1 \frac{H}{2^{i-1}} \frac{W}{2^{i-1}}$. The decoder is constituted of 4 blocks



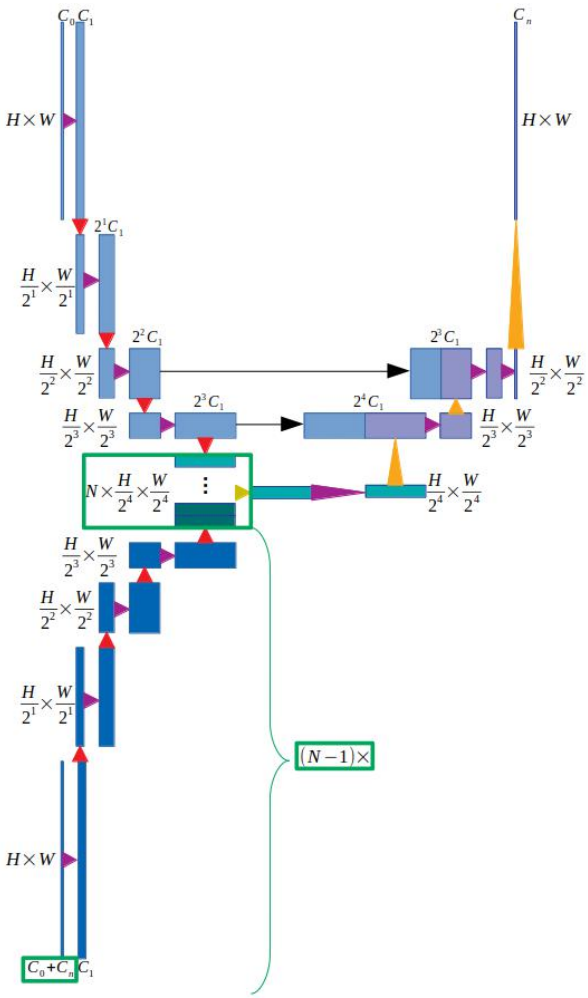
(a) Encoder - Decoder network architecture



(b) UNet architecture



(c) CO-Segmentation (COSNet) architecture



(d) Space-Time memory network (STM) architecture

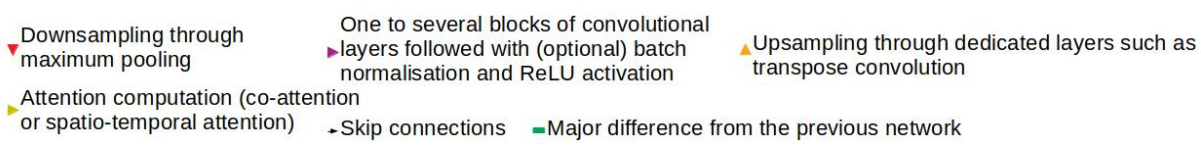


Figure 3.8: Network architectures. From inputs (left) to outputs (right).

of upsampling layers concatenated with the corresponding encoder block output of same resolution, see figure 3.8b.

3.2.3.2 ResUNet

Many variants of the UNet exist and share its general architecture, illustrated in figure 3.8b. One is the ResUNet, which consists in using a ResNet as encoder [He *et al.* 2016] and creating a decoder network accordingly for propagating feature maps from the encoder through skip connections.

The ResNet has alleviated a fundamental problem of deep NNs. They are universal function approximators, for which accuracy should improve with the number of layers. However, they suffer in practice from the vanishing gradient problem. Backpropagation computes gradients by the chain rule. One side effect is that multiplying several small gradient values can result in vanishingly small gradients, preventing weights from being updated. From a basis network, if the number of layers is increased, the accuracy saturates at one point and eventually degrades. Before the ResNet, shallower networks seemed to learn better than their deeper counterparts. While traditional NNs try to learn the true distribution of outputs $F(x)$, residual NNs try to learn the residuals $R(x)$:

$$R(x) = F(x) - x$$
$$F(x) = R(x) + x$$

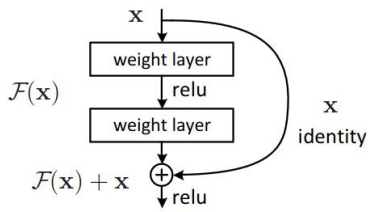
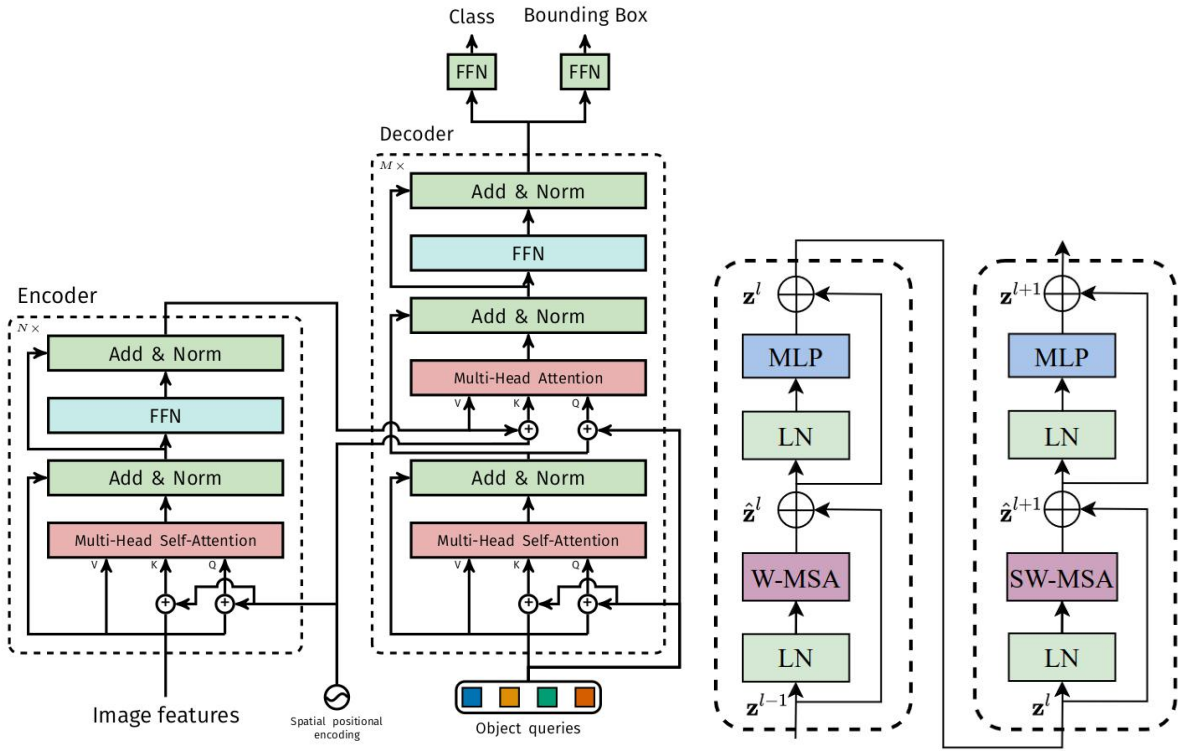


Figure 3.9: Residual connection block, from [He *et al.* 2016]. ©2016 IEEE.

In this aim, they use residual blocks, see figure 3.9. Skip connections provide alternative shortcuts for the gradient to pass through and thus larger gradients can be propagated to initial layers. Hence, residuals are easier to learn than outputs. Enabling the model to use identity functions ensures that the higher layers of the model do not perform any worse than the lower layers. Residual blocks allow information to flow from initial to last layers, or conversely.

3.2.3.3 Attention-based Segmentation Networks

Unlike convolution, attention has not a limited receptive field, see section 3.2.2. However, the drawback is that the required computation memory for attention could be costly for high dimensional inputs. Therefore, images should be processed in order to be treated in dimensions for which the computation remains feasible. Transformers using the self-attention mechanism and adapted to computer vision tasks are alternatives to convolution NNs and perform such processing. A transformer block is an encoder-decoder block and uses sub-blocks of stacked multi-head attention followed with fully-connected layers



(a) The DETR transformer block, from [Carion *et al.* 2020].

(b) The Swin transformer block, from [Liu *et al.* 2021].

Figure 3.10: Different transformer blocks with different inputs and outputs. Encoder inputs in a) are image features with spatial positional encoding while decoder inputs are object queries. The decoder outputs are input to fully-connected layers in order to output object classes and bounding boxes for an object detection task. In contrast, encoder and decoder inputs and outputs are only patch features in b), even though the inputs of the decoder are shifted and its first multi-head self-attention block is discarded. MLP and FFN represent fully-connected layers. LN states for Layer Normalisation. W-MSA and SW-MSA are multi-head self attention modules with regular and shifted windowing.

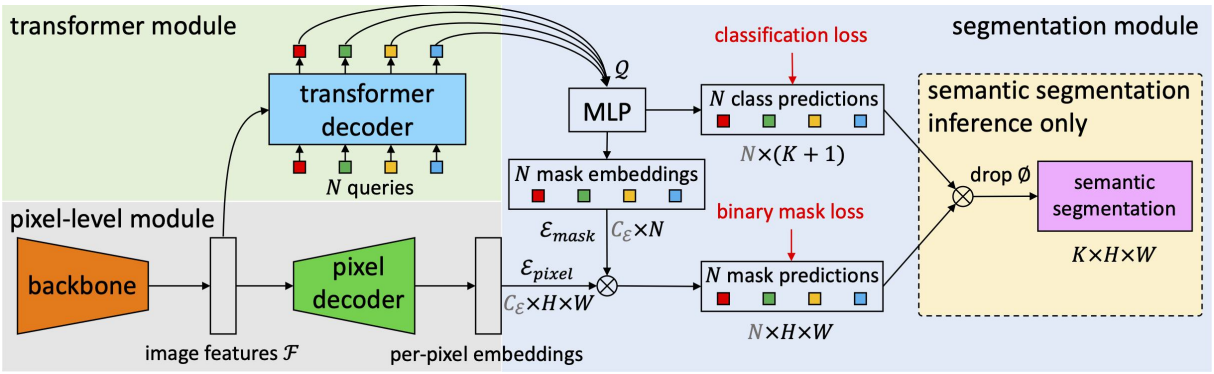
with residual connections and layer normalisation, for both the encoder and decoder, see figure 3.10. The encoder only has one of these sub-blocks with self-attention while the decoder combines two sub-blocks. The first one uses self-attention where both the query, keys as well as the values are from the decoder inputs. The second one uses cross-attention, where the query elements (x_i in equation 3.2) are from the decoder inputs while the keys (x_j in equation 3.2) and the associated values are from the encoder outputs. The encoder and decoder inputs and outputs can be chosen differently according to the problem, as shown in figure 3.10.

Swin transformer encoder. For instance, the Swin transformer is a common alternative to the ResNet for encoding image features. Images or feature maps of a layer are first partitioned into distinct small patches or rectangular windows (e.g. groups of 4×4 pixels with their channel values). Multi-head self-attention is computed within each window in a transformer encoder module. In the next layer, the partitioning is shifted, resulting

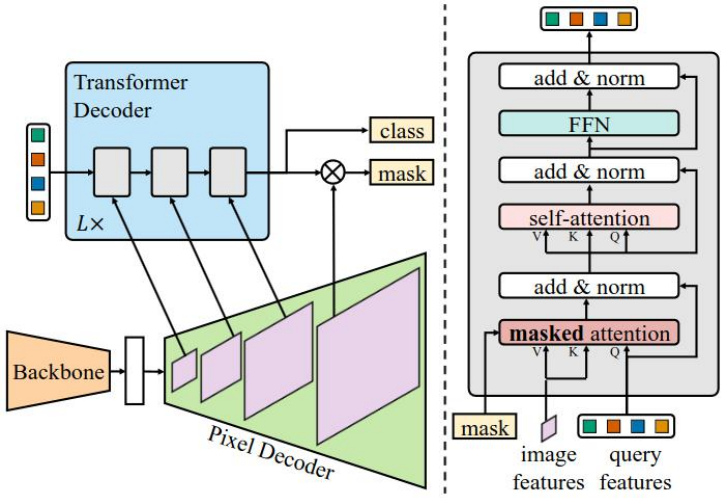
in new windows where multi-head attention is computed again in a transformer decoder module where the first sub-block is discarded and the query values also comes from the encoder outputs which are shifted, see figure 3.10b. This provides connections among the previous windows in the previous layer [Liu *et al.* 2021]. For downsampling, the features of each group of 2×2 neighbouring patches are concatenated (merged), before applying a linear layer to output the double of the previous number of channels, which mimics the downsampling factors of a standard convolutional NN encoder producing hierarchical feature maps (figure 3.8a).

Detection transformer decoder. In contrast, for decoding image features produced by an image encoder and predicting objects (position, size, class), the detection transformer (DETR) [Carion *et al.* 2020] first inputs image features with associated fixed spatial positional encodings into its encoder module. A convolution layer first reduces the channel dimension of the image features, allowing the transformer encoder to output feature maps of size $d \times HW$ where HW is the spatial dimension of the image encoding features. The spatial positional encodings are present for alleviating the issue caused by the property of self-attention mechanism to be equivariant to reordering, e.g. of the spatial elements. The decoder module inputs learnable object embeddings of size $N \times d$ referred to as object queries. The cross-attention is computed using keys and values coming from the transformer encoder outputs with associated spatial positional encodings and object queries processed by the first multi-head self-attention sub-block, see figure 3.10a. The transformer decoder outputs are fed into two independent regression network, composed of 3 successive fully-connected layers with ReLU activation which respectively predict bounding boxes of objects (centre coordinates, height and width) with their associated class labels. As the number of target objects in each image is not constant, the number N of predicted objects (including 'no objects or background' ones) is chosen to be large (e.g. $N = 100$) and a bipartite matching loss between the two sets is necessary for unique predictions. The general performance of this end-to-end object detection network is on par with state-of-the-art heavily tuned convolution NNs [Carion *et al.* 2020], better on large objects but worse on small objects.

MaskFormer segmentation network. State-of-the-art transformer-based segmentation NNs (MaskFormer [Cheng *et al.* 2021a] and Mask2Former [Cheng *et al.* 2022]) use an encoder-decoder architecture such as the convolution ones. They can use a Swin transformer or even a ResNet as encoder. However, they combine two decoders, see figure 3.11. A first one, referred to as the pixel decoder, progressively upsamples the features to generate per-pixel embeddings, as standard convolution-based decoders. However, at final layers, it does not reduce the d channel dimension to the number of target segments. Instead, the dot product is computed between its output and the output of size $N \times d$, referred to as mask embeddings, of a parallel object decoder made of successive detection transformer decoders with final fully-connected layers, see figure 3.11a. This results in N mask predictions of size HW. The output of the object decoder also contains $N \times (K+1)$ class predictions, the additional one being the background or 'no object' class. The dot product between both results in segmentation masks of size $K \times H \times W$. In order to train the network, a sum of a classification loss term for class predictions and a mask



(a) MaskFormer architecture. A backbone encoder extracts image features. A pixel decoder gradually upsamples image features to extract per-pixel embeddings ε_{pixel}. A transformer decoder attends to image features and produces N per-segment embeddings Q. The embeddings independently generate N class predictions with N corresponding mask embeddings ε_{mask}. Then, the model predicts N possibly overlapping binary mask predictions via a dot product between pixel embeddings ε_{pixel} and mask embeddings ε_{mask} followed by a sigmoid activation. For semantic segmentation task the final prediction is obtained by combining N binary masks with their class predictions using a simple matrix multiplication. From [Cheng *et al.* 2021a].



(b) Mask2Former architecture. In contrast to MaskFormer, in order to deal with small objects, it feeds high-resolution features from a pixel decoder layer scale to the corresponding transformer decoder layer. In addition, it uses masked attention which only attends within the foreground region of the resized mask prediction of the previous transformer decoder layer for each query, instead of the standard cross-attention. The order of self and cross-attention are switched. From [Cheng *et al.* 2022].

Figure 3.11: Attention-based MaskFormer and Mask2Former segmentation networks.

loss term for mask predictions is performed. The mask loss uses bipartite matching in order to associate the set of N predictions with the set of K classes. Ablation studies in [Cheng *et al.* 2021a] have shown that this formulation leads to improvements of the segmentation results over the per-pixel classification formulation without class predictions and only K mask embeddings, larger when the number N of classes is larger.

Mask2Former segmentation network. The pixel decoder used in Mask2Former is not based on convolutional and upsampling layers as in MaskFormer but instead uses the multi-scale deformable detection transformer encoder [Zhu *et al.* 2020] with a last upsampling layer. It is a variant of the detection transformer encoder, which takes as inputs multi-scale features, from the 3 last resolution levels of the encoder blocks. Each transformer encoder block inputs image encoder features of a resolution level (as skip connections of a UNet), from low to high, together with fixed spatial positional encodings and learnable scale-level embedding. Each deformable transformer encoder block outputs features of corresponding resolution.

MaskFormer uses the detection transformer decoder as object or segment decoder and thus does not deal well with small objects or segments. Mask2Former attempts to alleviate this issue in computing the cross-attention in 3 successive transformer decoder layers between object queries and features from low to higher resolution from the 3 corresponding block outputs of the pixel decoder (i.e. the deformable transformer encoder), together with fixed spatial positional encodings and learnable scale-level embedding, see figure 3.11b. However, the computational complexity related to high-resolution feature maps is alleviated in using masked cross-attention. It constrains the cross-attention to use key elements within the foreground region of the predicted mask, hence localised features, for each query, instead of attending to the full feature map. The predicted mask is the binarised output of the resized mask prediction of the previous transformer decoder layer. In addition, the order of the (masked) cross-attention and of the self-attention sub-blocks are reversed in detection transformer decoder layers, see figure 3.11b. Ablation studies in [Cheng *et al.* 2022] have shown that these modifications lead to important improvements of performance with respect to MaskFormer.

3.2.3.4 Co-Segmentation Networks

Instead of only using the information contained in an image for segmenting it, providing additional information from other images could help the segmentation of the image. The COSNet [Lu *et al.* 2019] uses paired images/views/frames in order to perform image segmentation based on multiple views. A pair of images is input to the encoder, which produces feature encodings. A co-attention module computes the attention between the two encodings. For each image, the feature encodings of the image and the attention output are concatenated and input to the decoder in order to produce segmentation outputs, as shown in figure 3.8c.

At inference, multiple pairs with the same first query image can be formed in order to use information from multiple views for the segmentation of the query image. It consists in computing the mean co-attention from all the pairs and inputing it to the decoder, together with the feature encodings of the image.

3.2.3.5 Spatio-Temporal Memory Networks

Additional information that could be beneficial for the segmentation of an image are the information of segmentation masks corresponding to other images. The Spatio-Temporal Memory Network (STM) [Oh *et al.* 2019] inputs both these corresponding image and segmentation masks to a specific 'memory' encoder. Another 'query' encoder inputs the

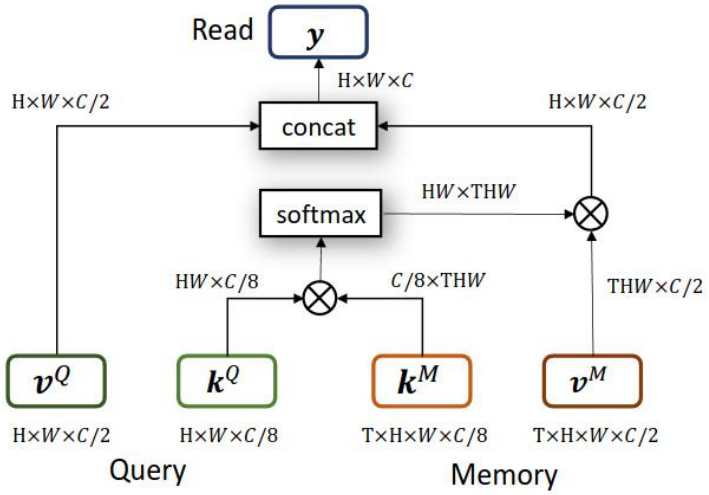


Figure 3.12: Space-time memory attention block with the circle-cross symbol denoting matrix inner-product, from [Oh *et al.* 2019]. Every spatio-temporal locations in the memory key map is compared with every spatial location in the query key map.

query image. Both these encodings are concatenated and a spatio-temporal 'memory' attention module outputs the attention. It is input to the decoder, with skip connections from the query encoder, which outputs a segmentation mask, see figure 3.8d. The attention module uses a residual block. Convolutional layers first splits the query and memory encodings into keys and values and then the Gaussian affinity function for f in equation 3.2 is computed, as illustrated in figure 3.12.

3.2.4 Related Work

A CASENet (Category-Aware Semantic Edge detection Network) [Yu *et al.* 2017] is used in [Koo *et al.* 2022]. Classes only comprise a single ridge and the silhouette, in addition to the background. CASENet has a specific decoder structure. Unlike the UNet where features are hierarchically and progressively upsampled, feature maps from the encoder are directly upsampled (using bilinear interpolation) to the input image resolution after a unique additional convolutional layer and no decoder-specific additional features, see figure 3.13. The convolutional layer has a different role according to the feature map resolution. The one with the lowest resolution performs classification, i.e. its output number of channels is the number of segmentation classes. In contrast, the convolutional layer higher resolution feature maps produces a single-channel output. Then, a shared concatenation separately concatenates these outputs with each of the activation outputs from the lowest-resolution features. One of the assumptions is that the receptive field of the high resolution features is limited. Thus performing semantic classification at an early stage should be avoided given that context information plays an important role in semantic classification. In contrast, high resolution features can be helpful in order to provide detailed edge localisation and structure information and augmenting the classifications from the low resolution features maps [Yu *et al.* 2017]. Their network is pretrained on ImageNet dataset [Deng *et al.* 2009]. They first train it on a synthetic dataset from [Pfeiffer *et al.* 2019], which uses image-to-image translation and style transfer in or-

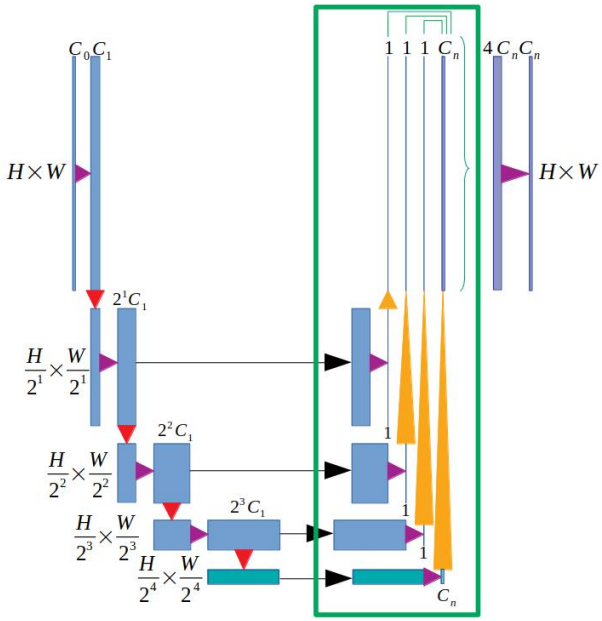


Figure 3.13: CASENet principle. Considering a defined encoder, the decoder performs convolution only from the encoder features, and interpolation-based upsampling in order to obtain the same resolution as the initial image. For the lowest resolution features, it performs convolution in order to get a number of channels equivalent to the segmentation classes (classification). For the higher resolution features, it performs convolution in order to get only one channel. These upsampled features are concatenated with the upsampled features from the classification level before to perform a final convolution on all the concatenated pairs.

der to produce mini-invasive appearance styles from content images. These content images are simulated and projected 3D scenes with a liver, a gallbladder, fat, ligaments, surgical tools and the inflated abdominal wall from 10 liver meshes extracted from CT scans associated to other extracted or artificial meshes. Each structure has a specific default texture with small random details. 2000 simulations from random perspectives are performed for each configuration, leading to 20000 synthetic images. 5 styles obtained with the image-to-image translation method are obtained from the Cholec80 dataset [Twinanda *et al.* 2016] and are applied to each image, leading to 100 000 synthetic images. Then, the actual CASENet training and evaluation is only performed on 133 images from from two laparoscopic interventions in [Koo *et al.* 2022], which does not allow one to evaluate its generalisation abilities.

In order to improve the image landmark annotation from a ResUNet, a very recent specific work [Pei *et al.* 2024] adds depth-driven geometric information obtained by a pretrained depth estimation network from a single monocular image [Yin *et al.* 2022] followed with a pretrained foundation Segment Anything Model [Kirillov *et al.* 2023] encoder whose weights are fixed. Another work also dealing with thin contours, but for the uterus, uses the UNet [François *et al.* 2020]. In their case, it performs better than CASENet.

3.3 Datasets, Training and Evaluation

Exploratory mini-invasive liver videos from 68 patients were collected with our university hospital partners. Between 10 and 30 1080p frames were extracted from these patient videos, leading to 1415 annotated images. The frames were selected to represent various views and configurations of each liver. Tools can be present in the views. Manual annotations of the landmarks were performed on each image. Examples of images and annotations can be found in figure 2.17, which illustrate the high diversity of liver texture and some annotation choices. This dataset is referred to as LaparoLiver.

Another clinical dataset was built in [Rabbani *et al.* 2022]. A sticker representing a chessboard pattern is attached on an ultrasound probe. After camera calibration, see section 2.4, the pose of the ultrasound probe in the mini-invasive camera space can be retrieved. If the ultrasound probe is also calibrated, this enables one to retrieve the Ground Truth (GT) tumour contour from the ultrasound slice in the mini-invasive camera space. This publicly shared dataset contains tens of images for each of 4 laparascopic procedures, which have been annotated. This dataset is referred to as RT-GT.

We also use another test set from the L3D dataset [Pei *et al.* 2024]. It contains annotations where lower central limits are absent and the left and right ridges are merged with the upper central limits and labelled as the same landmark. As the annotation rules can be different, extremities may not correspond.

For training, the 1415 samples from the LaparoLiver dataset are split into training (62 patients, representing 1303 samples) and validation (6 patients, representing 112 samples) datasets. The test sets correspond to the RT-GT and the L3D datasets (4 patients each). However, their characteristics are different from the training dataset, respectively due to the presence of the ultrasound probe and the chessboard, and the annotation rule differences. Evaluation on both validation and test sets allows then to obtain a range of errors from easy to hard configurations.

3.3.1 Implementation and Pretraining

The segmentation networks presented in this chapter are trained and evaluated. All images are first resized to $H \times W$ with $H = 256$ and $W = 256$ using bilinear interpolation, while masks are resized to the same size using nearest interpolation from PyTorch. All convolutional networks use $C_0 = 3, C_1 = 64, C_n = 10$. We provide details of the chosen implementations and pretraining:

- The UNet, see section 3.2.3.1, uses the Pytorch-UNet implementation. Its parameters are not pretrained.
- The ResUNet, see section 3.2.3.2, uses the query encoder from the STM implementation which takes up to the 4th residual block (and not the fifth one) from the standard ResNet50 implementation. Each block contains successive convolutional layers, batch normalisation and max pooling layers which perform 2×2 downsampling. Indeed, the first block uses strides of 2 and outputs feature maps of resolution $\frac{H}{2^1} \times \frac{W}{2^1}$ with C_1 channels, while the second, third and fourth blocks respectively output feature maps of size $2^2C_1 \times \frac{H}{2^2} \times \frac{W}{2^2}$, $2^3C_1 \times \frac{H}{2^3} \times \frac{W}{2^3}$ and $2^4C_1 \times \frac{H}{2^4} \times \frac{W}{2^4}$. The skip connections relate these feature maps to the decoder layers. Its architecture is

thus slightly different from the UNet one. The encoder parameters are pretrained on the ImageNet dataset [Deng *et al.* 2009] (for a classification task). The decoder is automatically built from the DynamicUNet from fastai implementation and is not pretrained.

- The CASENet, see section 3.2.4, uses the same encoder as the ResUNet, whose parameters are also pretrained on ImageNet, with the decoder parts of the CASENet implementation.

- The Mask2Former, see section 3.2.3.3, uses the MMsegmentation implementation. Both Resnet-50 and Swin-S transformer are tested as the backbone encoder, also pretrained on ImageNet, and produce features maps of size $2^5C_1 \times \frac{H}{2^5} \times \frac{W}{2^5}$. Pixel decoder features used in transformer decoder layers are of respective resolutions $\frac{H}{2^5} \times \frac{W}{2^5}$, $\frac{H}{2^4} \times \frac{W}{2^4}$ and $\frac{H}{2^3} \times \frac{W}{2^3}$. For the object decoder, $L = 3$ successive transformer decoders and $N = 100$ object queries are employed.

- The COSNet, see section 3.2.3.4, uses the same encoder and decoder architecture as the ResUNet. We maintain skip connections as the UNet unlike the original COSNet. An additional co-attention module and a convolutional layer are inserted between the encoder output and the decoder. The co-segmentation module is tested with different attention mechanisms and either at the single (S) lowest resolution level or at multiple (M) resolution levels, the three lowest ones. The resolution levels of additional modules with respect to the single co-segmentation one are represented by question marks in figure 3.8c. The tested attention mechanisms are the channel attention from the original COSNet [Lu *et al.* 2019], as well as the CBAM and the dual attention ones combining channel and spatial attention in different ways, see section 3.2.2. They are adapted from this Attention Module implementation.

- The STM, see section 3.2.3.5, uses the same ResNet-50 for both the query and memory encoders, from the STM implementation. The difference resides in the number of channels in the memory encoder input (the sum of the number of channels of an image and of a segmentation mask: $C_0 + C_n$), see figure 3.8d. For the spatio-temporal attention module, instead of the dot product $k_i^M \cdot k_j^Q$, we use the simplified L2 similarity $-\|k_i^M - k_j^Q\|_2^2$ from [Cheng *et al.* 2021b]. Only a subset of memory points has a chance to contribute the most for any query with the dot product while every memory point can contribute with the L2 similarity. The other blocks come from the STCN implementation.

3.3.2 Training mode, Losses and Parameters

During training, at each epoch, batch training samples (from LaparoLiver) are randomly selected (shuffled), using the same random seed. The network parameters are validated on the LaparoLiver validation dataset. For the COSNet, 100 image pairs are randomly selected per patient and form the training samples. For the STM network, each training sample consists of an image and a corresponding mask, which are augmented twice in the same way for the STM, using random translation, rotation, scale and resized crops of respective ranges [-0.1,0.1], [-15°,15°], [0.8,1.2] and [0.5,1.0], using PyTorch. Each time,

the order of the 3 images and masks is randomly shuffled. The second image is segmented with keys and values from the first image and mask. The third image is segmented with keys and values from the two first images and masks. No augmentation is performed for the other networks.

The standard loss function for image segmentation network training is the cross-entropy, as it corresponds to a measure of the difference between two probability distributions, the target class distribution y of the pixels and the predicted one $\hat{y}$:

$$f = -\sum_{i}^{C_n} y_i \log(\hat{y}_i)$$

Outputs of a network (logits) z pass through a softmax function to get a probability distribution over the C_n segmentation classes:

$$\hat{y}_i = \frac{\exp(z_i)}{\sum_{j=1}^{C_n} \exp(z_j)}$$

However, this loss function is not adapted to thin objects as the presence of the classes in the data is highly unbalanced. In order to alleviate this issue, we choose to use a weighted sum of cross-entropy and Tversky [Salehi *et al.* 2017] loss terms rather than using a weighted cross-entropy. The Tversky index is an equivalent of a classification metric with True Positives (TP), the number of predicted pixels which are target pixels, False Positives (FP), the number of predicted pixels which are not target pixels and False Negatives (FN), the number of target pixels which are not predicted. However, it uses probabilities of pixels to be or not to be in a specific class i (from the softmax outputs) in order to compute the equivalent of TP_i, FP_i and FN_i [Salehi *et al.* 2017]. The Tversky index uses weighted FP_i and FN_i:

$$g_i = \frac{\text{TP}_i}{\text{TP}_i + \gamma \text{FP}_i + \delta \text{FN}_i}$$

With N pixels, p_j^i the probability of a pixel to be a specific class i and $\bar{p}_j^i$ is the probability not being from this class. t_j^i is 1 if the pixel is of the target class or 0 otherwise and conversely for $\bar{t}_j^i$. This gives this formula for the mean Tversky index:

$$g = \frac{1}{C_n} \sum_{i=1}^{C_n} \left(\frac{\sum_{j=1}^{N} p_j^i t_j^i}{\sum_{j=1}^{N} p_j^i t_j^i + \gamma \sum_{j=1}^{N} p_j^i \bar{t}_j^i + \delta \sum_{j=1}^{N} \bar{p}_j^i t_j^i} \right)$$

The combined loss is:

$$h = \alpha f + \beta(1 - g)$$

The weights are set to: $\alpha = 5, \beta = 1, \gamma = 0.05, \delta = 0.95$, in order to limit FN. Other training parameters include:

- A batch size of 2
- Less than 50 epochs are used and factor reduction of 0.1 of the learning rate is applied every 15 steps.

- For all networks but the UNet and the CASENet, the Adam optimiser is used with a learning rate of 10^{-5}, a weight decay of 0.0005 and β of 0.9 and 0.99.
- For the UNet and the CASENet, the stochastic gradient descent is used with a learning rate of 10^{-2}, a weight decay of 0.0005 and a momentum of 0.9.
- The loss is only different for the Mask2Former. As it is combined with a cross-entropy classification loss term e, see section 3.2.3.3, the combined loss is $h = \varepsilon e + \alpha f + \beta(1-g)$. We choose equivalent $\alpha = 5, \beta = 5$ and $\varepsilon = 2$ parameters as [Cheng *et al.* 2022], which uses the mean Dice index for g term instead of the Tversky one, that we maintain with $\gamma = 0.05, \delta = 0.95$.

3.3.3 Evaluation Criteria

We use several criteria in order to evaluate the segmentation networks. First, we use the Mean Sum of Distances (MSD) [Li *et al.* 2005] between the predicted landmark point set $P = (\vec{p}_1, ..., \vec{p}_m)$ and the annotated (target) ones $T = (\vec{t}_1, ..., \vec{t}_n)$:

$$\text{MSD}(P,T) = \frac{1}{m+n}\left(\sum_{i=1}^{m}\min_j \|\vec{p}_i - \vec{t}_j\|_2 + \sum_{i=1}^{n}\min_j \|\vec{t}_i - \vec{p}_j\|_2\right)$$

The MSD can be considered as the average of the symmetric closest distances between predictions and targets and is also named Average Symmetric Distances (ASD) [Bilic *et al.* 2023]. We also evaluate the asymmetric mean Closest Distances to Targets (CD2T):

$$\text{CD2T}(P,T) = \frac{1}{n}\left(\sum_{i=1}^{n}\min_j \|\vec{t}_i - \vec{p}_j\|_2\right)$$

The previous evaluation criteria can provide a notion of average distance between predictions and targets. However, the same average distance can correspond to multiple configurations, for instance when there are holes or shifts in the predicted landmarks, with a different distribution of errors. In addition, when there are unpredicted landmarks, these criteria cannot provide a relevant measure. Thus, we also use standard average classification metrics, discarding the background class of label C_n:

$$\begin{aligned}
\text{Dice} &= \frac{1}{C_n-1}\sum_{i=1}^{C_n-1}\left(\frac{2\text{TP}_i}{2\text{TP}_i + \text{FN}_i + \text{FP}_i}\right)\\
\text{Precision} &= \frac{1}{C_n-1}\sum_{i=1}^{C_n-1}\left(\frac{\text{TP}_i}{\text{TP}_i + \text{FP}_i}\right)\\
\text{Recall} &= \frac{1}{C_n-1}\sum_{i=1}^{C_n-1}\left(\frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i}\right)
\end{aligned}$$

We add a specific condition when there is not ground truth for a landmark i. Both classification metrics are equal to 1 when there is no prediction, and 0 otherwise. In addition, we adapt these classification metrics to thin landmark curves. Indeed, if there is 1-pixel shift, the score of the classification metrics would be decreased. This is not

a desirable feature. Instead, this score should only decrease when the distance between thinned predictions and targets is above a distance tolerance threshold. We choose two thresholds of 1% and 2% of the image diagonal length, see figure 3.14. We compute the minimal distances and nearest neighbours between both point sets. Target (GT) points for which the distance from their nearest predicted neighbour is below the threshold are considered as TP, while the other points are FN. FP are all predicted points for which their target nearest neighbour are above the distance threshold. The corresponding classification metrics are named with the suffix 1% and 2%.

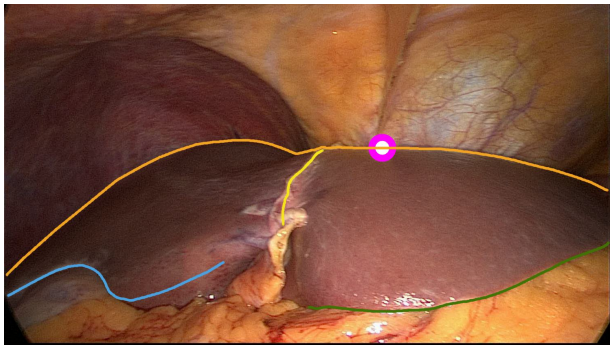


Figure 3.14: Illustration of the distance thresholds of 1% (white) and 2% (white + magenta) of the image diagonal.

3.3.4 Evaluation Results

Table 3.1 provides the evaluation results for all segmentation networks in LaparoLiver validation set, as well as RT-GT and L3D test datasets.

Networks based on independent image inputs. UNet is the only network whose encoder is not pretrained, and it obtains the lowest Dice scores on both validation and test datasets. Between CASENet and ResUNet, both using the pretrained ResNet encoder, ResUNet obtains a much better performance: around 6, 3, and 5% of Dice1% score increase and 6, 6, and 10% of recall1% improvement for respective LaparoLiver, RT-GT and L3D sets. However, it is surpassed by the Mask2Former, and in particular the version using the Swin transformer encoder. Indeed, it obtains the highest classification scores and the lowest distance metrics in all datasets. Compared to the ResUNet, it obtains 6.8, 4.2 and 5% Dice1% score increase for reaching 72.9%, 45.9% and 69.1% in respective datasets, but also improves precision and recall by a margin, except for the precision in RT-GT. The MSD and CD2T are fairly low for both datasets, below 2% of the image diagonal. Qualitative results in figures 3.15, 3.17 and 3.16 confirm the overall best and fairly good performance of the Mask2Former using the Swin transformer encoder, even if some false detections sometimes occur in the test set examples. They also confirm that the UNet and the CASENet overally detect fewer parts (more FN) than the other methods, in particular illustrated on the test set examples.

When comparing the results on the different datasets, similar results are obtained for the LaparoLiver validation set and the L3D test dataset. Even though landmarks are not annotated in the exact same manner, the image domains seem to be similar. In contrast,

Method	MSD	CD2T	Dice		Precision		Recall	
			1%	2%	1%	2%	1%	2%
LaparoLiver - validation set								
UNet	$2.4^{\pm 1.3}$	$2.7^{\pm 1.4}$	$58.8^{\pm 12.5}$	$66.7^{\pm 13.0}$	$68.1^{\pm 11.9}$	$74.9^{\pm 12.6}$	$56.5^{\pm 11.9}$	$64.6^{\pm 12.4}$
CASENet	$2.2^{\pm 1.1}$	$2.4^{\pm 1.0}$	$60.2^{\pm 11.6}$	$67.3^{\pm 12.0}$	$67.0^{\pm 12.3}$	$73.5^{\pm 12.7}$	$58.7^{\pm 10.8}$	$65.9^{\pm 11.0}$
ResUNet	$1.7^{\pm 0.8}$	$1.6^{\pm 0.7}$	$66.1^{\pm 9.8}$	$73.2^{\pm 9.8}$	$71.8^{\pm 9.1}$	$77.7^{\pm 9.4}$	$65.3^{\pm 9.3}$	$73.1^{\pm 9.3}$
Mask2Former-R	$1.0^{\pm 0.6}$	$1.0^{\pm 0.6}$	$66.7^{\pm 10.2}$	$71.6^{\pm 9.2}$	$70.6^{\pm 10.4}$	$74.4^{\pm 9.8}$	$65.8^{\pm 9.8}$	$70.9^{\pm 8.5}$
Mask2Former-S	$\mathbf{0.9^{\pm 0.4}}$	$\mathbf{0.9^{\pm 0.4}}$	$\mathbf{72.9^{\pm 7.9}}$	$\mathbf{78.5^{\pm 7.4}}$	$\mathbf{76.0^{\pm 9.1}}$	$\mathbf{80.6^{\pm 8.6}}$	$\mathbf{72.5^{\pm 6.6}}$	$\mathbf{78.5^{\pm 6.1}}$
COSNet	$1.6^{\pm 0.7}$	$1.7^{\pm 0.7}$	$66.8^{\pm 9.3}$	$73.1^{\pm 8.2}$	$72.3^{\pm 9.9}$	$77.6^{\pm 9.2}$	$65.5^{\pm 8.5}$	$71.9^{\pm 7.3}$
STM	$1.3^{\pm 0.8}$	$1.5^{\pm 0.8}$	$\mathbf{74.8^{\pm 10.3}}$	$\mathbf{81.2^{\pm 9.9}}$	$\mathbf{79.8^{\pm 8.0}}$	$\mathbf{85.6^{\pm 8.0}}$	$\mathbf{73.3^{\pm 11.3}}$	$\mathbf{79.8^{\pm 10.6}}$
RT-GT - test set								
UNet	$2.8^{\pm 1.4}$	$3.0^{\pm 1.7}$	$38.5^{\pm 7.3}$	$49.5^{\pm 10.1}$	$44.6^{\pm 10.9}$	$57.1^{\pm 16.7}$	$38.9^{\pm 9.4}$	$48.3^{\pm 10.1}$
CASENet	$2.2^{\pm 1.1}$	$2.8^{\pm 1.8}$	$39.6^{\pm 6.0}$	$49.2^{\pm 4.3}$	$47.2^{\pm 10.1}$	$57.1^{\pm 9.1}$	$38.5^{\pm 6.8}$	$47.6^{\pm 4.5}$
ResUNet	$3.1^{\pm 2.8}$	$2.0^{\pm 1.8}$	$41.7^{\pm 10.1}$	$48.6^{\pm 10.1}$	$45.6^{\pm 10.3}$	$52.7^{\pm 11.3}$	$44.6^{\pm 9.3}$	$51.5^{\pm 8.7}$
Mask2Former-R	$1.6^{\pm 0.4}$	$1.6^{\pm 0.7}$	$42.4^{\pm 7.0}$	$48.1^{\pm 7.7}$	$44.0^{\pm 8.8}$	$49.4^{\pm 9.4}$	$43.7^{\pm 7.0}$	$49.1^{\pm 7.2}$
Mask2Former-S	$\mathbf{1.6^{\pm 0.5}}$	$\mathbf{1.1^{\pm 0.7}}$	$\mathbf{45.9^{\pm 7.6}}$	$\mathbf{51.5^{\pm 10.5}}$	$\mathbf{46.1^{\pm 8.6}}$	$\mathbf{51.8^{\pm 11.7}}$	$\mathbf{50.2^{\pm 6.1}}$	$\mathbf{55.4^{\pm 9.1}}$
COSNet	$1.8^{\pm 0.7}$	$2.3^{\pm 1.1}$	$44.1^{\pm 11.5}$	$51.5^{\pm 11.9}$	$53.0^{\pm 18.2}$	$59.6^{\pm 17.1}$	$42.4^{\pm 7.2}$	$49.1^{\pm 8.6}$
STM	$\mathbf{0.9^{\pm 0.3}}$	$1.3^{\pm 0.4}$	$\mathbf{69.2^{\pm 4.0}}$	$\mathbf{78.1^{\pm 3.9}}$	$\mathbf{76.7^{\pm 5.3}}$	$\mathbf{83.5^{\pm 6.2}}$	$\mathbf{65.8^{\pm 2.5}}$	$\mathbf{75.0^{\pm 2.4}}$
L3D - test set								
UNet	$2.7^{\pm 0.5}$	$2.7^{\pm 1.0}$	$58.4^{\pm 11.1}$	$68.0^{\pm 10.5}$	$63.2^{\pm 15.6}$	$73.8^{\pm 13.5}$	$56.7^{\pm 9.1}$	$66.0^{\pm 8.3}$
CASENet	$2.6^{\pm 1.5}$	$3.0^{\pm 2.0}$	$59.2^{\pm 10.7}$	$68.0^{\pm 9.7}$	$68.3^{\pm 16.3}$	$76.9^{\pm 14.9}$	$56.3^{\pm 11.7}$	$64.5^{\pm 10.4}$
ResUNet	$3.0^{\pm 0.7}$	$1.7^{\pm 0.7}$	$64.1^{\pm 9.3}$	$72.4^{\pm 8.8}$	$66.4^{\pm 9.0}$	$73.4^{\pm 8.1}$	$66.3^{\pm 9.7}$	$\mathbf{75.1^{\pm 8.9}}$
Mask2Former-R	$\mathbf{1.8^{\pm 0.3}}$	$1.8^{\pm 0.5}$	$68.8^{\pm 12.7}$	$75.1^{\pm 12.9}$	$71.6^{\pm 13.3}$	$77.4^{\pm 13.0}$	$67.5^{\pm 12.4}$	$74.1^{\pm 12.6}$
Mask2Former-S	$1.9^{\pm 0.5}$	$\mathbf{1.6^{\pm 0.6}}$	$\mathbf{69.1^{\pm 14.6}}$	$\mathbf{75.4^{\pm 14.2}}$	$\mathbf{72.6^{\pm 16.1}}$	$\mathbf{78.9^{\pm 15.1}}$	$\mathbf{68.8^{\pm 12.5}}$	$74.9^{\pm 12.4}$
COSNet	$2.4^{\pm 0.3}$	$1.9^{\pm 0.5}$	$61.2^{\pm 12.1}$	$69.9^{\pm 11.6}$	$66.1^{\pm 15.1}$	$74.2^{\pm 14.1}$	$60.3^{\pm 10.8}$	$69.2^{\pm 10.0}$
STM	$\mathbf{1.5^{\pm 1.0}}$	$1.7^{\pm 1.1}$	$\mathbf{76.7^{\pm 11.0}}$	$\mathbf{84.3^{\pm 10.8}}$	$\mathbf{82.4^{\pm 9.4}}$	$\mathbf{89.5^{\pm 8.8}}$	$\mathbf{74.8^{\pm 12.2}}$	$\mathbf{82.0^{\pm 12.2}}$

Table 3.1: Results on both validation and test datasets for all segmentation networks based on independent images, paired images, or paired images and masks, split by double horizontal lines. Classification metrics are in % and both MSD and CD2T evaluation criteria are in % of the image diagonal length.

much worse results are obtained on the RT-GT one. We assume that this is due to the difference of image domain caused by the ultrasound probe with the white and black chessboard sticker, which results in different lightning conditions and occlusion of some landmark parts, see figure 3.17.

Network based on paired images (COSNET). We employ the dual attention mechanism in multiple resolution scales. Pairing is performed with all other images of the patient in each set. While obtaining equivalent results to the ResUNet in the LaparoLiver validation set, it improves the segmentation performance in the RT-GT test set and degrades it in the L3D test set, with respective Dice1% score increase of 2.4% and drop of 2.9%. This denotes the importance of paired image selection. In particular, the RT-GT set contains images whose content is close and thus suggests to mainly pair such images, which could be performed in taking close frames in a video sequence.

Network based on paired images and masks (STM). Pairing is performed with all other images and masks of the patient in each set. Utilising these information is very beneficial, as shown in the results. Indeed, it outperforms networks based on independent

Pretraining	MSD	CD2T	Dice		Precision		Recall	
			1%	2%	1%	2%	1%	2%
None	$4.4^{\pm 2.3}$	$4.8^{\pm 2.7}$	$40.7^{\pm 16.3}$	$48.2^{\pm 17.3}$	$51.8^{\pm 16.5}$	$58.1^{\pm 16.6}$	$37.6^{\pm 15.8}$	$45.0^{\pm 17.1}$
ImageNet	$\mathbf{1.7^{\pm 0.8}}$	$\mathbf{1.6^{\pm 0.7}}$	$\mathbf{66.1^{\pm 9.8}}$	$\mathbf{73.2^{\pm 9.8}}$	$\mathbf{71.8^{\pm 9.1}}$	$\mathbf{77.7^{\pm 9.4}}$	$\mathbf{65.3^{\pm 9.3}}$	$\mathbf{73.1^{\pm 9.3}}$

Table 3.2: Effect of pretraining on the ResUNet results on the LaparoLiver dataset. Classification metrics are in % and both MSD and CD2T evaluation criteria are in % of the image diagonal length.

image inputs by a margin for test datasets, as denoted in the Dice1% score respective increase of 23.3% and 7.6%. In particular, there are less FP as precision scores are higher, above 76% for precision1% in all sets. Note that this evaluation is for validating the relevance of using these information, but the network could be made more efficient in tuning augmentations to be closer to the expected image domain, using elastic image deformations, blurring ... Qualitative results in both datasets (figures 3.15, 3.17 and 3.16) confirm that very few wrong detections occur (few FP) and all present landmarks are generally detected, even though some parts are sometimes missed.

3.3.5 Ablation Studies

Ablation studies for selecting the encoder pretraining and the loss parameters are performed for the ResUNet, as the CASENet, COSNet and STM are based on the same encoder: from the ResUNet. Then, we compare different attention mechanisms for the co-attention module of the COSNet. Finally, we access the influence of the samples used in STM training, with close or away content samples.

Pretraining. In table 3.2, we compare the results obtained without and with ResNet encoder pretraining, on ImageNet dataset, for the ResUNet. The training uses all the same parameters, and the samples are processed in the same pseudo-random order in a deterministic way. It can be seen that using pretraining to initialise the encoder parameters allows a large improvement of the results. In this case, it is preferable to use pretrained encoder parameters from a large number of training samples (more than one million for ImageNet), even though it was pretrained for a classification task and not a segmentation one.

Losses. In table 3.3, an ablation study of the loss parameters is performed. We first compare the results using the combined cross-entropy + Tverksy loss with respect to the only cross-entropy one ($\alpha = 1, \beta = 0$). The Dice1% score increases of more than 10% (66.1 versus 53.4) on the validation LaparoLiver set. This confirms that using the combination is very relevant. Then changing the different parameters allows the best results on a (ResUNet) network for $\alpha = 5, \beta = 1, \gamma = 0.05$ and $\delta = 0.95$. It confirms that attempting to reduce FN is relevant in case of very thin landmarks.

Co-attention module. Tables 3.4 describe results for different co-attention modules applied to either single or multiple scaled feature maps, see section 3.2.3.4. When using both spatial and channel attention instead of the only channel one, both the distance

Loss params	MSD	CD2T	Dice		Precision		Recall	
			1%	2%	1%	2%	1%	2%
$\alpha = 1, \beta = 1,$ $\gamma = 0.5, \delta = 0.5$	$1.9^{\pm 0.9}$	$2.1^{\pm 0.8}$	$61.3^{\pm 11.8}$	$67.7^{\pm 11.6}$	$70.3^{\pm 10.8}$	$75.4^{\pm 10.7}$	$58.0^{\pm 11.3}$	$65.3^{\pm 11.2}$
$\alpha = 1, \beta = 1,$ $\gamma = 0.3, \delta = 0.7$	$1.8^{\pm 0.8}$	$2.0^{\pm 0.9}$	$62.4^{\pm 12.6}$	$68.7^{\pm 12.0}$	$70.3^{\pm 11.5}$	$75.3^{\pm 11.1}$	$60.1^{\pm 11.9}$	$67.0^{\pm 11.6}$
$\alpha = 1, \beta = 1,$ $\gamma = 0.1, \delta = 0.9$	$1.8^{\pm 1.0}$	$2.0^{\pm 1.0}$	$65.3^{\pm 9.3}$	$72.3^{\pm 8.8}$	$\mathbf{72.5^{\pm 8.5}}$	$\mathbf{78.2^{\pm 8.8}}$	$63.3^{\pm 9.0}$	$71.1^{\pm 8.2}$
$\alpha = 1, \beta = 1,$ $\gamma = 0.05, \delta = 0.95$	$1.7^{\pm 0.7}$	$1.8^{\pm 0.7}$	$65.8^{\pm 9.8}$	$73.1^{\pm 8.9}$	$71.8^{\pm 8.9}$	$78.1^{\pm 8.7}$	$64.6^{\pm 9.7}$	$72.5^{\pm 8.6}$
$\alpha = 1, \beta = 1,$ $\gamma = 0.01, \delta = 0.99$	$1.9^{\pm 0.7}$	$1.8^{\pm 0.6}$	$65.7^{\pm 10.1}$	$73.1^{\pm 8.7}$	$70.8^{\pm 10.4}$	$77.3^{\pm 9.3}$	$\mathbf{65.3^{\pm 8.6}}$	$\mathbf{73.5^{\pm 7.1}}$
$\alpha = 1, \beta = 0$	$2.8^{\pm 1.3}$	$2.9^{\pm 1.2}$	$53.4^{\pm 13.4}$	$61.3^{\pm 13.4}$	$64.9^{\pm 11.8}$	$71.0^{\pm 12.2}$	$49.3^{\pm 12.9}$	$58.1^{\pm 13.0}$
$\alpha = 1, \beta = 5,$ $\gamma = 0.05, \delta = 0.95$	$\mathbf{1.5^{\pm 0.5}}$	$\mathbf{1.6^{\pm 0.6}}$	$61.4^{\pm 12.0}$	$67.4^{\pm 11.2}$	$65.6^{\pm 13.2}$	$70.8^{\pm 12.4}$	$60.7^{\pm 11.5}$	$66.9^{\pm 10.5}$
$\alpha = 5, \beta = 1,$ $\gamma = 0.05, \delta = 0.95$	$1.7^{\pm 0.8}$	$\mathbf{1.6^{\pm 0.7}}$	$\mathbf{66.1^{\pm 9.8}}$	$\mathbf{73.2^{\pm 9.8}}$	$71.8^{\pm 9.1}$	$77.7^{\pm 9.4}$	$\mathbf{65.3^{\pm 9.3}}$	$73.1^{\pm 9.3}$

Table 3.3: ResUNet results on the LaparoLiver validation dataset for different combinations of loss parameters. ResUNet was pretrained on ImageNet. Classification metrics are in % and both MSD and CD2T evaluation criteria are in % of the image diagonal length.

COSNet	MSD	CD2T	Dice		Precision		Recall	
			1%	2%	1%	2%	1%	2%
channel single	$2.1^{\pm 1.0}$	$2.1^{\pm 0.8}$	$61.8^{\pm 10.2}$	$68.6^{\pm 9.5}$	$70.2^{\pm 9.2}$	$75.5^{\pm 9.4}$	$59.2^{\pm 10.3}$	$66.8^{\pm 9.4}$
channel multiple	$2.0^{\pm 1.1}$	$2.2^{\pm 1.0}$	$62.3^{\pm 10.9}$	$69.1^{\pm 9.9}$	$72.3^{\pm 9.9}$	$77.2^{\pm 9.1}$	$58.5^{\pm 10.8}$	$66.0^{\pm 9.8}$
CBAM single	$1.7^{\pm 0.8}$	$1.9^{\pm 0.9}$	$63.4^{\pm 9.5}$	$70.4^{\pm 8.8}$	$71.7^{\pm 9.8}$	$76.9^{\pm 9.5}$	$60.9^{\pm 9.0}$	$68.9^{\pm 8.1}$
CBAM multiple	$2.0^{\pm 0.7}$	$\mathbf{1.7^{\pm 0.5}}$	$65.7^{\pm 7.9}$	$71.9^{\pm 7.1}$	$70.6^{\pm 7.6}$	$75.9^{\pm 7.2}$	$\mathbf{66.0^{\pm 7.5}}$	$\mathbf{72.5^{\pm 6.5}}$
Dual single	$\mathbf{1.6^{\pm 0.9}}$	$\mathbf{1.7^{\pm 0.9}}$	$65.9^{\pm 10.6}$	$72.7^{\pm 9.9}$	$\mathbf{73.1^{\pm 9.7}}$	$\mathbf{78.4^{\pm 9.5}}$	$63.5^{\pm 10.5}$	$70.9^{\pm 9.7}$
Dual multiple	$\mathbf{1.6^{\pm 0.7}}$	$\mathbf{1.7^{\pm 0.7}}$	$\mathbf{66.8^{\pm 9.3}}$	$\mathbf{73.1^{\pm 8.2}}$	$72.3^{\pm 9.9}$	$77.6^{\pm 9.2}$	$65.5^{\pm 8.5}$	$71.9^{\pm 7.3}$

Table 3.4: Results on the LaparoLiver dataset validation for the COSNet using different attention mechanisms on one or multiple resolution levels. Classification metrics are in % and both MSD and CD2T evaluation criteria are in % of the image diagonal length.

and classification metrics improve, in particular for the recall. Dual attention gets better results than CBAM one. Using co-attention on multiple resolution levels also slightly improves the segmentation results. However, segmentation performance with co-attention is lower than or on par with the ResUNet without co-attention, when inferred on paired images from the same patient, but away from each other. This may suggest that information from different away images is difficult to deal with. Results on the RT-GT dataset in table 3.1 indeed suggests that co-attention inferred with close content images (e.g. sequentially-close frames) is more beneficial.

STM	MSD	CD2T	Dice		Precision		Recall	
			1%	2%	1%	2%	1%	2%
LaparoLiver - validation set								
away	$2.1^{\pm 1.1}$	$2.3^{\pm 1.0}$	$62.8^{\pm 10.3}$	$70.7^{\pm 10.4}$	$70.8^{\pm 9.3}$	$77.7^{\pm 9.7}$	$60.6^{\pm 10.1}$	$68.4^{\pm 10.1}$
close	$\mathbf{1.3^{\pm 0.8}}$	$\mathbf{1.5^{\pm 0.8}}$	$\mathbf{74.8^{\pm 10.3}}$	$\mathbf{81.2^{\pm 9.9}}$	$\mathbf{79.8^{\pm 8.0}}$	$\mathbf{85.6^{\pm 8.0}}$	$\mathbf{73.3^{\pm 11.3}}$	$\mathbf{79.8^{\pm 10.6}}$
RT-GT - test set								
away	$2.4^{\pm 0.8}$	$2.6^{\pm 1.0}$	$35.9^{\pm 6.2}$	$45.2^{\pm 9.1}$	$40.4^{\pm 8.0}$	$50.3^{\pm 9.9}$	$36.9^{\pm 4.2}$	$45.4^{\pm 7.7}$
close	$\mathbf{0.9^{\pm 0.3}}$	$\mathbf{1.3^{\pm 0.4}}$	$\mathbf{69.2^{\pm 4.0}}$	$\mathbf{78.1^{\pm 3.9}}$	$\mathbf{76.7^{\pm 5.3}}$	$\mathbf{83.5^{\pm 6.2}}$	$\mathbf{65.8^{\pm 2.5}}$	$\mathbf{75.0^{\pm 2.4}}$
L3D - test set								
away	$2.2^{\pm 0.3}$	$2.3^{\pm 0.9}$	$56.3^{\pm 9.8}$	$66.5^{\pm 9.6}$	$63.8^{\pm 13.5}$	$75.0^{\pm 15.5}$	$53.8^{\pm 11.3}$	$63.5^{\pm 10.1}$
close	$\mathbf{1.5^{\pm 1.0}}$	$\mathbf{1.7^{\pm 1.1}}$	$\mathbf{76.7^{\pm 11.0}}$	$\mathbf{84.3^{\pm 10.8}}$	$\mathbf{82.4^{\pm 9.4}}$	$\mathbf{89.5^{\pm 8.8}}$	$\mathbf{74.8^{\pm 12.2}}$	$\mathbf{82.0^{\pm 12.2}}$

Table 3.5: Results on both validation and test datasets for STM networks based on close or away training samples. Classification metrics are in % and both MSD and CD2T evaluation criteria are in % of the image diagonal length.

STM training samples We compare the STM segmentation performance for different training samples in table 3.5. First, the synthetic close content triplet samples described in section 3.3.2. Second, 100 triplet images and masks randomly selected from the same patient, but whose content is potentially away. This has a major importance as the training from away content samples performs much worse in all sets, even performing worse than the other methods. The performance boost compared to independent image segmentation networks should also be superior for close content inference samples. It is suggested in table 3.1 where the performance boost on the RT-GT test set is superior to the ones on the other sets.

3.4 Conclusion

We have implemented and trained multiple segmentation networks on the same dataset. Evaluation on validation and test datasets as well as ablation studies first reveals that:

- Pretraining the encoder on a large dataset, even for classification tasks, allows a network to obtain high performance improvements, i.e. learning better. This highlights the importance of the network parameter initialisation.
- The training loss benefits from adding a term to the cross-entropy one, dedicated to reduce the FN, through the Tversky loss.
- Among tested convolutional-based NNs dedicated to independent image inference, the ResUNet obtains the best results.
- However, it is surpassed by state-of-the-art attention-based NNs, the Mask2Former one, in particular on the test sets. It additionally benefits from replacing a convolutional-based encoder, the ResNet-50, by an attention-based one, the Swin-B transformer.
- Among NN options dedicated to paired image inference, using a co-attention module with an attention mechanism on both spatial and channel domains, in particular

the dual one, allows the segmentation network to obtain a higher performance. Using the attention mechanism on multiple resolution scales also slightly improves it. However, paired image selection is fundamental. In particular, using close content paired images, such as sequentially close video frames, is desirable.

- Using additional mask information from other images boosts the segmentation performance with a large margin. It highly benefits from training using close images, and also benefits from inferring close content paired samples, such as sequentially close video frames.

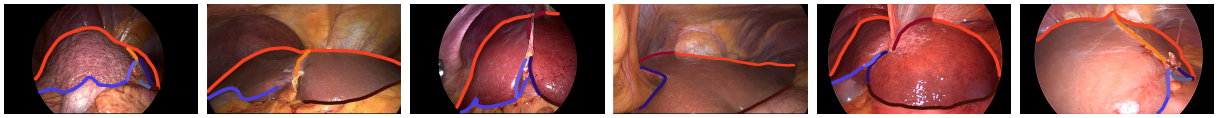
- The segmentation networks trained on LaparoLiver does not generalise well to the RT-GT test set. We assume that this is due to the presence of the ultrasound probe with chessboard patterns, not present in the training set. However, they generalise quite well to the L3D test dataset, as the image domains should relatively match.

Hence, segmenting mini-invasive images independently leads to results which can be improved when combining information cautiously. In future work, we envisage to transform the Mask2Former in order to combine informations from different images and/or masks, as the COSNet and the STM. In addition, we will evaluate the relevance of adding estimated depth information, from pretrained and frozen depth estimation networks, such as the very recent work from [Pei *et al.* 2024].

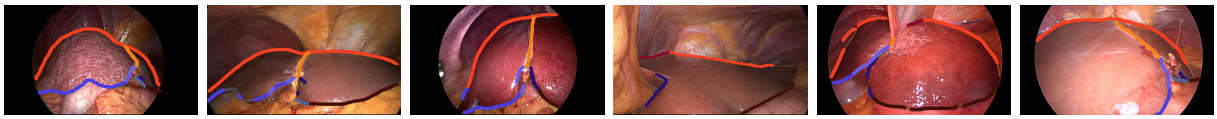
In clinical practice, combining multiple segmentation networks could also be a solution. First, a segmentation network only using image information from multiple sequentially close views could first detect the landmarks. Then, from the first detections, the segmentation network using mask information from previous video frames could be used.

As this landmark annotation task is critical for guiding the 3D-2D registration, a semi-automatic correction tool could be beneficial in case of errors. For instance, an encoder-decoder architecture embedded in a interaction loop with a user feedback memory [Zhou *et al.* 2023, Mikhailov *et al.* 2024] could be explored, as for preoperative volume segmentation, see section 2.1.

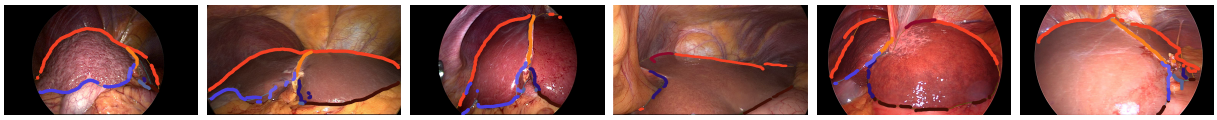
We also envisage the exploration of an alternative formulation of the problem, such as the regression of landmark curve parameters, based on an encoder-regressor architecture, see section 4.4. This would be similar to the approach of fitting deformable models to appearance models, presented in section 3.1, using deep learning.



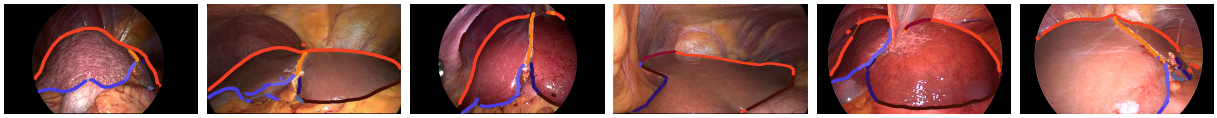
(a) GT targets



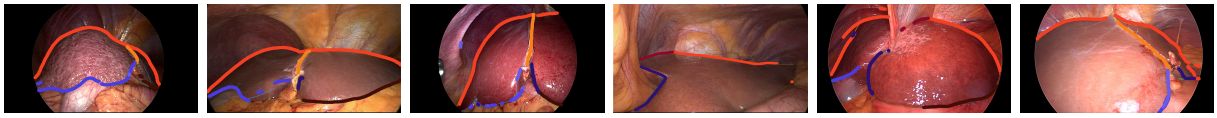
(b) UNet



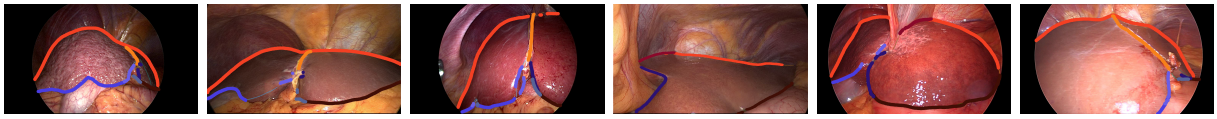
(c) CASENet



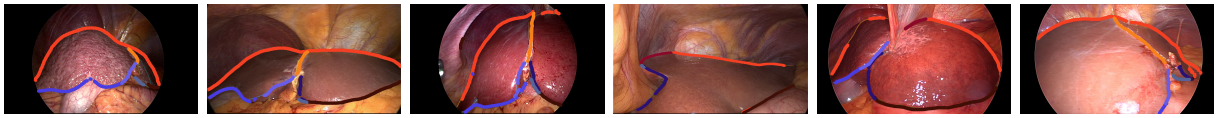
(d) ResUNet



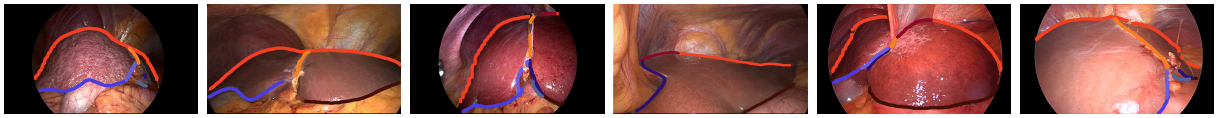
(e) Mask2Former - ResNet encoder



(f) Mask2Former - Swin transformer

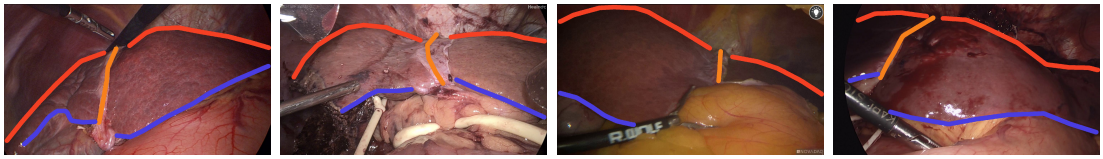


(g) COSNet

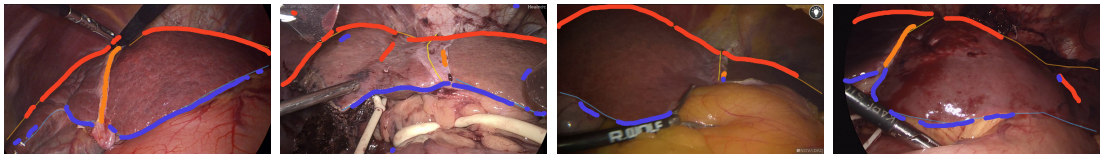


(h) STM

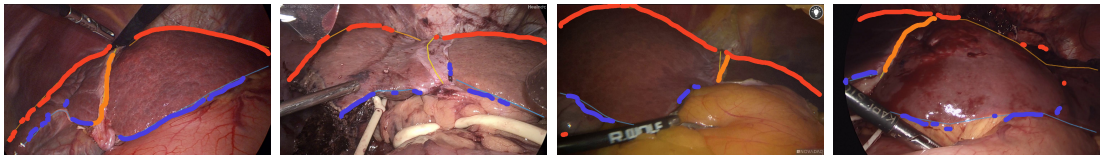
Figure 3.15: Illustration of segmentation results on one image of each patient of the LaparoLiver validation set, for each of the tested segmentation networks. Predicted landmarks (enlarged for visualisation purposes) are: **silhouette**, **junction with cut ligament falciform**, **junction with attached ligament falciform**, **left ridge**, **right ridge**, **upper-left central limit**, **lower-left central limit**, **upper-right central limit**, **lower-right central limit** while the target ones have the properties of the figure 2.17, except for the first row where they share the predicted landmark properties.



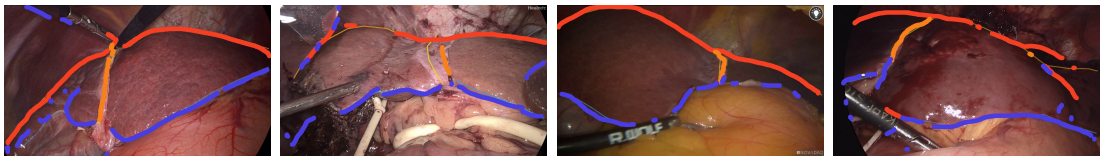
(a) GT targets



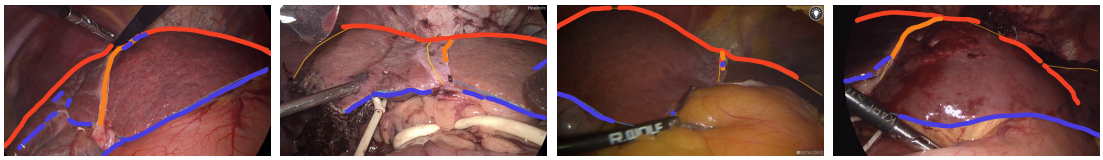
(b) UNet



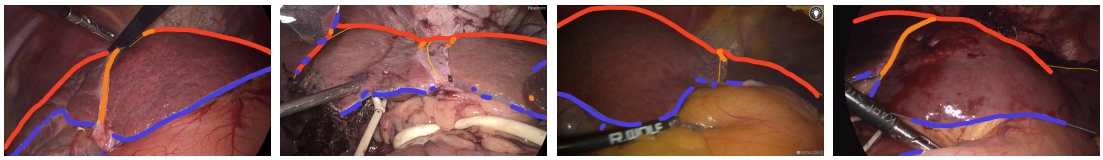
(c) CASENet



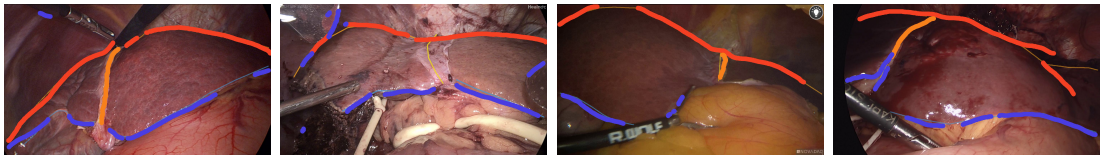
(d) ResUNet



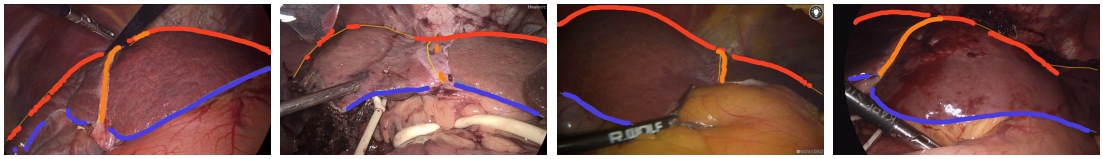
(e) Mask2Former - ResNet encoder



(f) Mask2Former - Swin transformer encoder

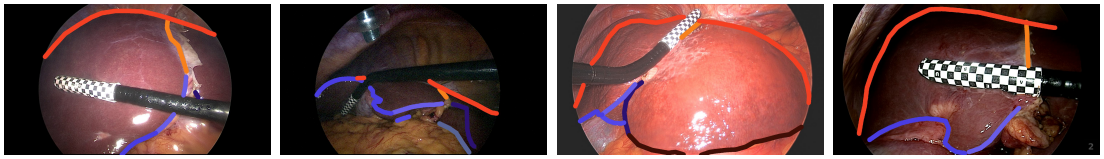


(g) COSNet

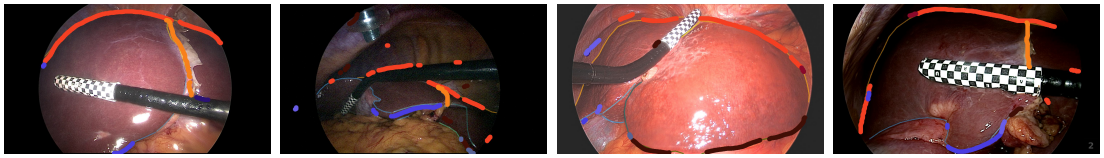


(h) STM

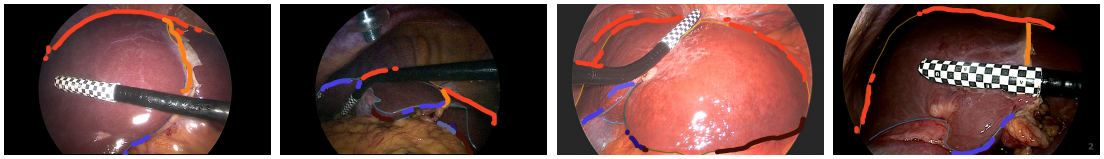
Figure 3.16: Illustration of segmentation results on one image of each patient of the L3D test set, for each of the tested segmentation networks. The colour and properties of each landmark follow the rules of figure 3.15.



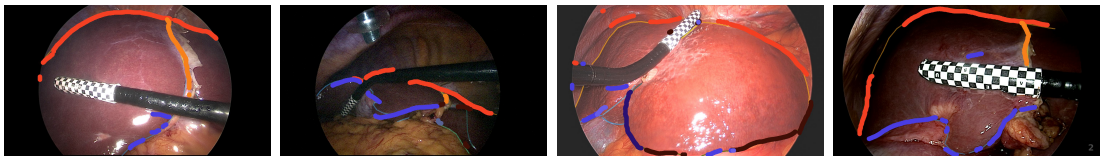
(a) GT targets



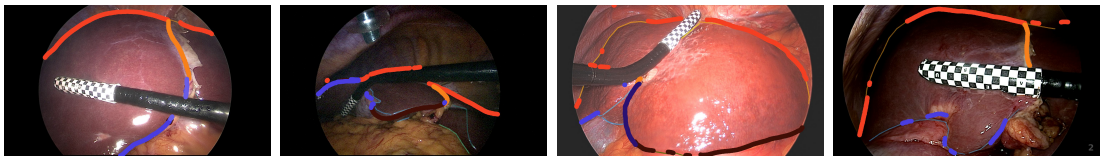
(b) UNet



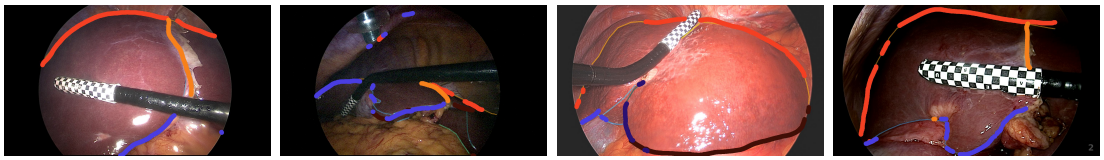
(c) CASENet



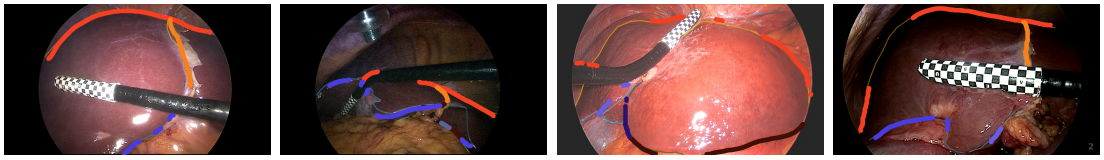
(d) ResUNet



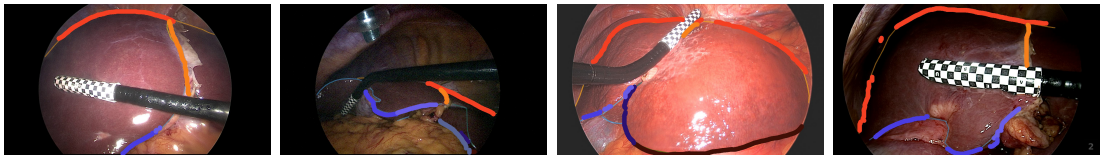
(e) Mask2Former - ResNet encoder



(f) Mask2Former - Swin transformer encoder



(g) COSNet



(h) STM

Figure 3.17: Illustration of segmentation results on one image of each patient of the RT-GT test set, for each of the tested segmentation networks. The colour and properties of each landmark follow the rules of figure 3.15.

Chapter 4

Automatic Patient-Specific 3D/2D Registration

Contents

4.1 Introduction

In the baseline pipeline, once all the preoperative and intraoperative preparation steps have been performed, intraoperative registration based on corresponding 3D/2D landmarks can be dealt with, see figure 1.12. We split registration into two parts, rigid and deformable. Both require the camera intrinsics as well as corresponding image and model

landmarks. The deformable registration may also require a deformation model in order to constrain deformations in a plausible range while making the computation efficient enough for being compatible with clinical usage. In the absence of prior knowledge of the deformation applied to the preoperative model, it is first manually rigidly registered so that its landmarks coarsely fit the image ones. Then, from this pose in the camera scene, the preoperative model is deformed so that its landmarks finely fit the image ones while satisfying realistic deformation constraints. We explore two ways of automating the registration: either automating each registration step, i.e. the rigid one (section 4.3), as the deformable one is already automated, or automating both rigid and deformable steps simultaneously using a pipeline based on deep learning (section 4.4). We first overview previous works designed for automating each of these registration tasks, in section 4.2.

4.2 Related Work

4.2.1 Classical Rigid Registration

Rigid registration consists in estimating the camera pose with respect to the preoperative liver model. It is typically done manually [Adagolodjo *et al.* 2017, Koo *et al.* 2017b, Özgür *et al.* 2018, Espinel *et al.* 2020]. Indeed, PnP pose estimation, see section 2.4.4, cannot be directly used because 3D-2D point correspondences are not precisely known between the preoperative 3D model and the 2D image landmarks. If coarse 3D-2D point correspondences were given, PnP would be prone to errors. Indeed, this occurs when there are mismatches, which do not fit the model, in the set of point correspondences. PnP solving can be made more robust to mismatches when combined to methods such as RANdom SAmple Consensus (RANSAC), for estimating the parameters of a mathematical model from a set of observed data that contains outliers [Fischler & Bolles 1981]. RANSAC performs repeated random sub-sampling, keeping the minimal number of samples necessary for determining the model parameters. It eventually selects the subset for which the fitted model results in the largest 'consensus' set of inliers. In RANSAC-based PnP, the consensus set can be defined as the set of points which obtain a reprojection error below a predefined distance threshold. The reprojection error can be defined as the (squared) distance between corresponding target and projected points. The PnP can then be solved only using the largest consensus set of correspondences.

A brute-force approach [Koo *et al.* 2022] indirectly obtains 3D-2D correspondences from the anterior ridge and the silhouette of the liver and estimates the pose from these correspondences using RANSAC-based PnP. It consists in sampling a large set of camera poses from limited rotation ranges around a camera keyhole pose assumption and estimating the visible liver surface and landmarks on the 3D model. 3D-2D point correspondences between the projected and target image landmarks are computed by searching for the closest neighbour satisfying a constraint of limited angle difference between projected and target 2D contour normals. RANSAC-based PnP can then be applied to estimate the pose, using P3P [Gao *et al.* 2003], see section 2.4.4. Among all the estimated poses, the one resulting in a minimal distance between the 2D contours is selected. The distance measure is the modified Hausdorff distance, the maximum of the mean average closest distance between each set taken alternatively as the target one. The main limits of this

method is its reliance on pose simulations and a unique RANSAC threshold. If the simulation domain is not appropriate or not enough simulations are performed, the pose can be far from optimal. For the issue of using a unique RANSAC threshold, we refer the reader to section 4.3.

Alternatively, rigid registration can be performed using stereoscopic images. In [Robu *et al.* 2018], it uses the ridge and a reconstructed intraoperative 3D point cloud from stereo by means of a shape matching technique. Other 3D registration methods such as [Min *et al.* 2019] could also be used.

4.2.2 Classical Deformable Registration

4.2.2.1 Position-Based Dynamics

Previous works iteratively deform the preoperative model from the estimated pose. [Adagolodjo *et al.* 2017] only uses the silhouette information in order to perform it while [Koo *et al.* 2017b] uses anatomical landmarks in addition to the silhouette. The latter utilises position-based dynamics [Müller *et al.* 2007] of continuous materials [Bender *et al.* 2014], which does not employ a deformation model. In contrast, it attempts to displace each volumetric mesh vertex position according to the explicit numerical integration of Newton's equation of motion, as well as constraints. Instead of the implicit integration formulation in equation 2.2 and described in section 2.3.1.3, the explicit formulation is:

$$\begin{cases} v^{n+1} = v^n + M^{-1} f(x^n, v^n)\Delta t \\ x^{n+1} = x^n + v^{n+1}\Delta t \end{cases} \tag{4.1}$$

The total force is assumed to be a sum of external and constraint forces, respectively F_{ext} and F_{con}, i.e. $f(x^n, v^n) = F_{ext} + F_{con}$. The predicted $\tilde{x}$ related to the external forces can be computed as:

$$\begin{cases} \tilde{v} = v^n + \Delta t M^{-1} F_{ext} \\ \tilde{x} = x^n + \Delta t \tilde{v} \end{cases} \tag{4.2}$$

The constraints can be positional targets, used as attachment constraints, as well as elastic and shading ones [Koo *et al.* 2017b]. For positional targets, the position of a concerned vertex $\tilde{x}^k$ is simply updated at every time step to coincide with the target y^k [Müller *et al.* 2007]. When only considering elastic constraints: $f(x^n, v^n) = F_{ext} - \Delta E$ where E is the strain energy related to a given constitutive model, see section 2.3.1.2. Replacing related terms in equation 4.1, this leads to the following equation:

$$M(x^{n+1} - \tilde{x}) + \Delta^2 t \nabla E = 0 \tag{4.3}$$

Let $\Delta\tilde{x} = x^{n+1} - \tilde{x}$. E is linearly approximated as $E = E(\tilde{x} + \Delta\tilde{x}) \approx E(\tilde{x}) + \nabla E(\tilde{x})\Delta\tilde{x} = 0$, and therefore $\nabla E(\tilde{x})\Delta\tilde{x} = -E(\tilde{x})$. A Lagrange multiplier λ replacing $-\Delta^2 t$ is introduced. This gives the system of equation:

$$\begin{cases} M\Delta\tilde{x} - \lambda \nabla E(\tilde{x})^T = 0 \\ \nabla E(\tilde{x})\Delta\tilde{x} = -E(\tilde{x}) \end{cases}$$

Algorithm 1 An iteration of position-based dynamics used in [Koo *et al.* 2017b]

Time integration step for model vertices using explicit Euler scheme for external forces. *In the problem at hand, only damping is used.*
$\tilde{v} \leftarrow v^n + \Delta t M^{-1} F_{ext}$
$\tilde{x} \leftarrow x^n + \Delta t \tilde{v}$
Alternating projection of constraints for modifying previous predicted positions
Solving positional target constraints
for each correspondence k **do**
    $\tilde{x}_k \leftarrow y_k$
Solving elastic constraints
$N \leftarrow 0$
for each tetrahedron j **do**
    Compute E_j and ΔE_j (see section 2.3.1)
    $\lambda_j \leftarrow -\frac{E_j(\tilde{x})}{\nabla E_j(\tilde{x}) M^{-1} \nabla E_j(\tilde{x})^T}$ (equation 4.4)

    for each vertex i in tetrahedron j **do**
        $\Delta \tilde{x}_{i,j} \leftarrow M^{-1} \nabla E_{i,j}(\tilde{x})^T \lambda_j$ (equation 4.5)
        $N_i \leftarrow N_i + 1$
Solving shading constraints (facultative)
Constraint averaging for Jacobi iterative solver (different for Gauss-Seidel)
for each vertex i **do**
    $\Delta \tilde{x}_i \leftarrow \frac{1}{N_i} \sum_j \Delta \tilde{x}_{i,j}$
Update of every positions and then velocities
$x^{n+1} \leftarrow \tilde{x} + \Delta \tilde{x}$
$v^{n+1} \leftarrow \frac{1}{\Delta t}(x^{n+1} - x^n)$

The system can be expressed with respect to the two unknowns $\Delta\tilde{x}$ and λ. Taking the Schur complement with respect to M [Macklin *et al.* 2016], this reduces to:

$$\Rightarrow \lambda = -\frac{E(\tilde{x})}{\nabla E(\tilde{x}) M^{-1} \nabla E(\tilde{x})^T} \tag{4.4}$$

$$\Delta\tilde{x} = M^{-1} \nabla E(\tilde{x})^T \lambda \tag{4.5}$$

Hence, the positions x_{n+1} can be retrieved in several steps. The pseudo code in Algorithm 1 describes the iterative position-based dynamics. First, a time integration step taking into account the external forces and using an Explicit Euler scheme is performed for preoperative model vertices in order to obtain new locations of the vertices. In the problem at hand, external forces are unknown, only damping is applied. However, positional targets are known. At each iteration, the silhouette is updated using the process described in section 2.5.2. Then, unlike the pose estimation method which uses 2D correspondences, each 3D landmark point $\tilde{x}^k$ is associated to a target point y^k in the camera space, for 3D correspondences. For each 3D landmark point, the closest points on the

projection lines passing through the 2D target landmarks are first retrieved. Among all, the closest from the 3D landmark point is selected as the correspondence.

For the landmark vertices, the predicted positions $\hat{x}$ from external forces are modified in order to first satisfy these positional target constraints, which are the landmark correspondences. Then all vertex positions are modified in order to satisfy elastic constraints. Thus, for each tetrahedron, the strain energy as well as the elastic forces applied on the tetrahedron vertices are computed, following the process described in section 2.3.1. The Lagrange multiplier related to the tetrahedron is then computed using equation 4.4, and the position step $\Delta\tilde{x}$ for each tetrahedron vertex can be deduced using equation 4.5. A shading constraint can also be taken into account. The positions are updated according to the iterative scheme, e.g. Gauss-Seidel or Jacobi ones. For instance, the Gauss-Seidel scheme updates the vertex position at every vertex position step computation, while the Jacobi scheme updates it only once, after every position steps related to this vertex are computed. It takes the average position step. In the final step, the corrected positions are used to update the velocities of the volumetric vertices.

The main issues with position-based dynamics is the dependency of the results on the amount of solver iterations [Bartels 2015]. The amount of solver iterations, either using Gauss-Seidel or Jacobi, should be very high in order to respect all the elastic constraints. Then, in practice, these constraints are not fully respected and the algorithm does not converge.

4.2.2.2 Optimisation

Optimisation can also be performed in order to deform the preoperative model initialised in the camera space from pose estimation. Previous works use it on other organs, such as the uterus [Collins *et al.* 2020]. The cost function is a weighted sum on non-linear square residuals, combining m distance residuals r^d as well as n strain energy residuals r^s depending on deformation model parameters θ:

$$f(\theta) = \alpha \sum_{i=1}^{m} r_i^d(\theta)^2 + \beta \sum_{j=1}^{n} r_i^s(\theta)^2 \tag{4.6}$$

The deformation model can be built in several ways, see section 2.3. At each iterate, the silhouette is updated using the process described in section 2.5.2. Then, it uses the same process as the position-based dynamics for estimating correspondences. Each 3D landmark point $\hat{p}_i$ is associated to a target point p_i in the camera space, for 3D correspondences. The closest points on the projection lines passing through the 2D target landmarks from each 3D landmark point are first retrieved. Among them, the closest one from the 3D landmark point is selected as the correspondence for computing distance residuals. For computing the n strain energy residuals E associated to the tetrahedra of the deformed shape iterate, as well as their gradients in order to fill the Jacobian, the procedure described in section 2.3.1 can be used.

4.2.3 Learning-based Rigid and Deformable Registration

To our knowledge, before our work publication [Labrunie *et al.* 2023], there were no previous works on learning-based 3D/2D registration in MILS. We have thus explored this

novel approach. Human body shape reconstruction is the problem of, from an image showing a human, estimating a physically plausible human body mesh aligned with the image. Therefore, it shares many similarities with the problem of registering a preoperative liver mesh onto a mini-invasive image. It also requires a camera model and thus the estimation of camera parameters. In addition, the general pose of the camera with respect to the mesh should also be determined. For human body mesh modelling, most methods combine a reference shape model and joint pose [Loper *et al.* 2015]. The shape model include variations of height and body proportions among human bodies, while the joint pose deal with the 3D deformations with respect to joint parts, i.e. relative rotations of joint parts with respect to parents in a human body kinematic tree.

Both optimisation-based and learning-based approaches were studied for solving this problem [Tian *et al.* 2022]. Optimisation-based methods first compute an initial pose and shape, followed by an optimisation of the shape parameters for multiple image cues, including the silhouette [Guan *et al.* 2009]. This is similar to the method applied to the liver and described in section 4.2.2.2. Learning-based methods are recent and stem from the end-to-end Human Mesh Recovery (HMR) framework [Kanazawa *et al.* 2018], see figure 4.2. HMR starts with a ResNet-50 encoder, see section 3.2.3.2, to extract image features. It then concatenates the image features with initial pose, shape and camera parameters, which are all fed to a regression network with two fully-connected layers with ReLU activation layers, see section 3.2.1. The regression network uses Iterative Error Feedback (IEF) [Carreira *et al.* 2016] in order to update all the parameters iteratively. At training, the image joint locations are known. The projection of the estimated 3D joints allows one to compute a 2D reprojection loss. When 3D GT is known, a 3D loss is computed directly between the predicted and target 3D joints and parameters. In addition, a discriminator is used to improve the shape and pose plausibility, which is especially important in the absence of 3D GT.

4.3 Visibility-Aware Pose Estimation

In contrast to previous works, see section 4.2.1, we propose a rigid registration method that searches for the pose directly from the landmarks, without requiring stereo, and without assuming a known keyhole position and simulated camera poses.

We propose to solve this challenging pose problem iteratively, in three coarse to fine steps, as given by Algorithm 2. The refinement makes use of the visibility of the anatomical landmarks in the previous iterates and eventually takes into account the silhouette information. Therefore, the main differences between these steps are the preoperative 3D vertices (with their image correspondences) selected for pose estimation in a RANSAC-based PnP solution. The coarse step ① uses all anatomical landmark vertices. The following step ② refines pose iteratively using the visible anatomical landmark vertices only, determined from the current pose estimate (line 34, figure 4.1c). The fine step ③ adds the correspondences between the silhouette vertices related to the image silhouette (line 32), determined using the process described in section 2.5.2. The correspondences are $\hat{V}_S$ and $\hat{I}_S$ in line 31.

The procedure for estimating the pose is given in lines 14-22. First, the landmark image curves are sampled uniformly to create one-to-one correspondences with the model

Algorithm 2 Pose Estimation Pseudo-code

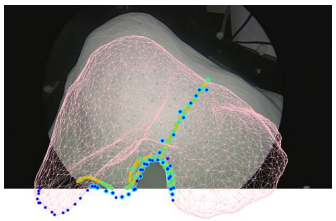
main variables:
V_L: 3D landmark vertex coordinates in preoperative space
I_L, I_S: 2D landmark and silhouette coordinates in image space
T: threshold set for inlier selection in extended RANSAC-based PnP

MinError $\leftarrow +\infty$, BestPose $\leftarrow \varnothing$
① Coarse estimation from all landmark vertices
BestPose, MinError $\leftarrow$ EstimatePose(V_L,I_L,T,MinError,BestPose)
② Refinement from visible anatomical landmark vertices only
BestPose, MinError $\leftarrow$ RefinePose(False,V_L,I_L,I_S,T,MinError,BestPose)
③ Refinement from visible anatomical landmark vertices and the silhouette
BestPose, MinError $\leftarrow$ RefinePose(True,V_L,I_L,I_S,T,MinError,BestPose)

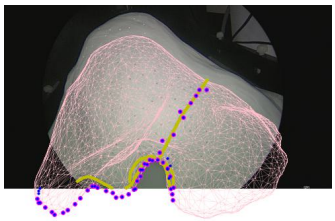
procedure EstimatePose(V_t,I_t,T,MinError,InitialPose)
    BestPose $\leftarrow$ InitialPose
    $\hat{I}_t \leftarrow$ SampleImageLandmarksAsModelVertexOnes(I_t,V_t)
    for each τ **in** T **do**
        EstimatedPose $\leftarrow$ RANSACPnP(V_t,$\hat{I}_t$,τ,InitialPose)
        ProjectedV_t $\leftarrow$ ProjectToImagePlane(V_t,EstimatedPose)
        MSD $\leftarrow$ ComputeMSD(GetVisibleProjections(ProjectedV_t),I_t)
        if MSD<MinError **then** MinError$\leftarrow$MSD, BestPose $\leftarrow$ EstimatedPose
    return BestPose, MinError

procedure RefinePose(WithSilhouette,V_L,I_L,I_S,T,MinError,BestPose)
    PreviousMinError $\leftarrow +\infty$
    while MinError<PreviousMinError **do**
        PreviousMinError $\leftarrow$ MinError
        VisibleV_L $\leftarrow$ DetermineVisibleVerticesInImage(V_L,BestPose)
        if WithSilhouette **then**
            $V_S \leftarrow$ DetermineSilhouetteVertices(BestPose)
            $\hat{V}_S, \hat{I}_S \leftarrow$ GetCorrespondences(V_S,I_S,BestPose)
            $V_t \leftarrow$ [VisibleV_L, $\hat{V}_S$], $I_t \leftarrow [I_L, \hat{I}_S]$
        else
            $V_t \leftarrow$ VisibleV_L, $I_t \leftarrow I_L$
        BestPose,MinError $\leftarrow$ EstimatePose(V_t,I_t,T,MinError,BestPose)
    return BestPose, MinError

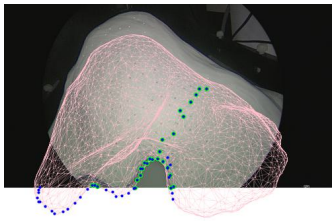
vertices (line 16). From these 3D-2D correspondences, we solve a RANSAC-based PnP (line 18) using OpenCV. PnP uses a non-linear Levenberg-Marquardt minimisation to refine the initial pose, see section 2.4.5. For ①, this is initialised with Direct Linear Transformation (section 2.4.4) while for ② and ③ it uses the best pose from the previous steps.



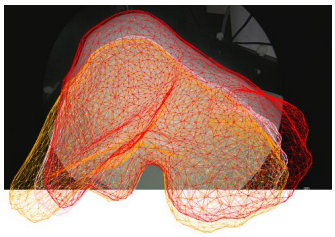
(a) Original image and model landmarks (yellow and blue) used in ① (coarse pose estimation) for solving PnP. In pink, best pose result for ① w.r.t. MSD after inlier selection (cyan) by RANSAC.



(b) Example of image and model landmarks (respectively magenta and yellow) used for MSD computation in one of the first two steps (① and ②).



(c) Example of visible landmarks (lime) used as inputs in ②.



(d) Example of refined pose estimate after step ② and final one ③ (respectively orange and red).

Figure 4.1: Illustration of the proposed method details for automatic pose estimation, on simulated mini-invasive data using a 3D printed liver phantom.

RANSAC highly depends on the reprojection error threshold defining the inlier set. An unadapted threshold leads to an inlier set with too many or too few points and then to an incorrect pose. In the problem at hand, the threshold cannot be chosen a priori. This is because the error not only depends on observation noise, as in classical vision problems, but also on modelling error. The latter stems from several factors, the use of a rigid registration to approximate the real deformation field being the strongest one. We propose an extension of RANSAC to determine an optimal threshold at runtime. This works by repeating RANSAC for several thresholds from a set $T \subset \mathbb{R}$. Each threshold $\tau \in T$ gives an inlier set (figure 4.1a) and a pose estimate. We eventually select the best solution using the MSD (ASD) criterion (section 3.3.3) between the image landmarks I_t and the reprojected visible model vertices (line 20, figure 4.1b). Unlike line 28, only self-occluded landmark vertices are considered invisible. They are determined by a ray-triangle intersection method from trimesh or in another fashion by mesh rasterising with silhouette shading using GPU, from pytorch3d. The latter determines the non-occluded liver mesh faces associated to the image pixels and thus enables the determination of visible landmark vertices (which are associated to mesh faces). A problem could occur if the landmark vertex or face is occluded but very close to a visible face, the landmark vertex is discarded. In contrast, we would prefer to maintain the landmark as visible in these conditions. Indeed, our landmarks are not highly precise points, but rather surface areas around the defined curves. In this aim, we first preoperatively determine the neighbour faces of each mesh face associated to a landmark vertex, and if one among them is visible, we also consider the landmark vertex as visible. Projected landmark vertices in the black

border and outside the image are kept in the MSD computation. Figure 4.1d shows results for each step of our pipeline.

4.4 Liver Mesh Recovery

Following the HMR framework, see section 4.2.3, we propose a learning-based registration pipeline. However, we adapt it to the liver problem and call it LMR. The LMR network is displayed in figure 4.3 while the global registration pipeline is shown in figure 4.4. The first adaptations consists in using 1) a PS liver deformation model instead of the person-generic human joint pose and shape model and 2) corresponding liver landmarks for guiding the registration training in place of corresponding human joint points. This context leads to another modification, required by the limited amount of annotated data in MILR and the fact that real PS data only becomes available at the time of surgery, in contrast to a person-generic model in HMR trained from massive datasets. It consists in replacing images by PS landmark (primitive) masks as inputs. This enables us to perform numerous preoperative simulations of liver mesh configurations and associated landmark masks for training the network. This supervised training makes the adversarial discriminator from HMR unnecessary. At inference, the annotated masks are fed to the network. In addition, LMR uses the pinhole camera model, which reduces depth ambiguities, in contrast to the scaled-orthographic model in HMR.

Hence, LMR uses a ResNet-50 encoder which delivers features from the image primitive masks, represented as distance maps, see figure 4.3. The features are concatenated with the current pose and deformation parameters. They are fed to the regression network, based on fully connected layers, which iteratively updates the pose R, T and deformation parameters β through IEF. We use the same number of layers and neurons in the encoder and regressor networks as in HMR [Kanazawa *et al.* 2018], except for the input and output

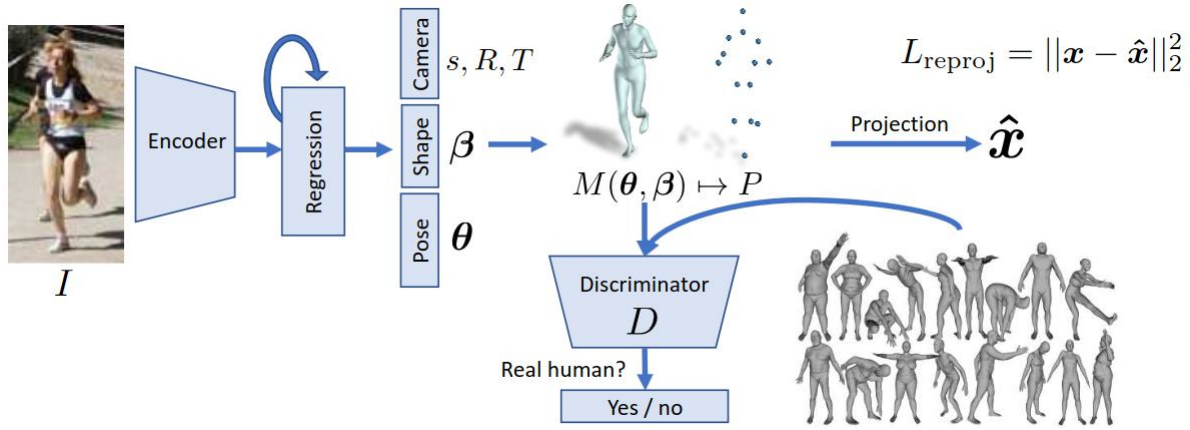


Figure 4.2: Overview of the Human Mesh Recovery framework. An image is passed through an image encoder. The encoder features are sent to an iterative 3D regression module that infers the latent 3D representation of the human that minimises the joint reprojection error. The 3D parameters are also sent to a discriminator D, whose goal is to tell if these parameters come from a real human shape and pose. From [Kanazawa *et al.* 2018].

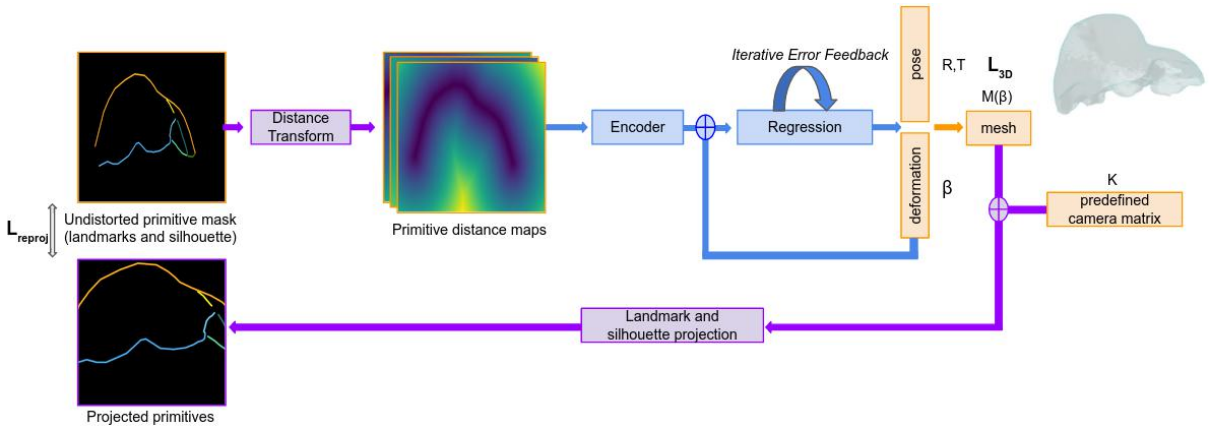


Figure 4.3: Proposed Liver Mesh Recovery (LMR) framework. The framework inputs the primitive mask for the surgical image (top left) and outputs the corresponding 3D model pose and deformation parameters (top right). It computes a distance transform, encodes it, and iteratively regresses the parameters to minimise a sum of 3D losses and primitive (landmark) reprojection loss.

layers of the regressor, adapted to the number of deformation parameters. LMR outputs the registered liver mesh, in other words, a mesh properly deformed and expressed in camera coordinates, whose reprojection matches the liver observed in the mini-invasive image. LMR is inside a pipeline comprising both preoperative and intraoperative stages.

4.4.1 Preoperative Stage

The preoperative steps reconstruct the preoperative 3D model, which is used to synthesise images and train the LMR network.

Step 1: Preoperative 3D model reconstruction and landmark annotation. The preoperative volume is segmented (section 2.1), producing a surface mesh, which is upgraded to a volume by constrained Delaunay tetrahedralisation [Shewchuk 2002], presented in section 2.2. This yields a volumetric mesh retaining the surface vertices, whose n vertex coordinates, of order 10000, are stacked into the column vector $\mu^\top = [\mu_1^\top, \cdots, \mu_n^\top] \in \mathbb{R}^{3n}$. The tumours and vena cava are marked as inner regions of the volumetric mesh (section 2.2.5) and the landmarks are annotated on the surface, following the process described in 2.5.

Step 2: Deformable liver shape modelling Multiple ways of modelling the liver deformation are presented in section 2.3. They can come from (FEM or FFD) $k = 5000$ simulations whose dimensions are reduced using global(-G) truncated SVD. FEM-based ones with Ogden constitutive model using low and high nodal force amplitudes are respectively referred to as FEM-OLA-G and FEM-OHA-G, while the other is referred to as FFD-G. A deformed shape is represented by its vertex coordinates $\hat{x}_i$, obtained from its m deformation coefficients $\beta_i = [\beta_{i,1}, \cdots, \beta_{i,m}]^\top$ as $\hat{x}_i = \mu + \sum_{j=1}^{m} \beta_{i,j}\phi_j$.

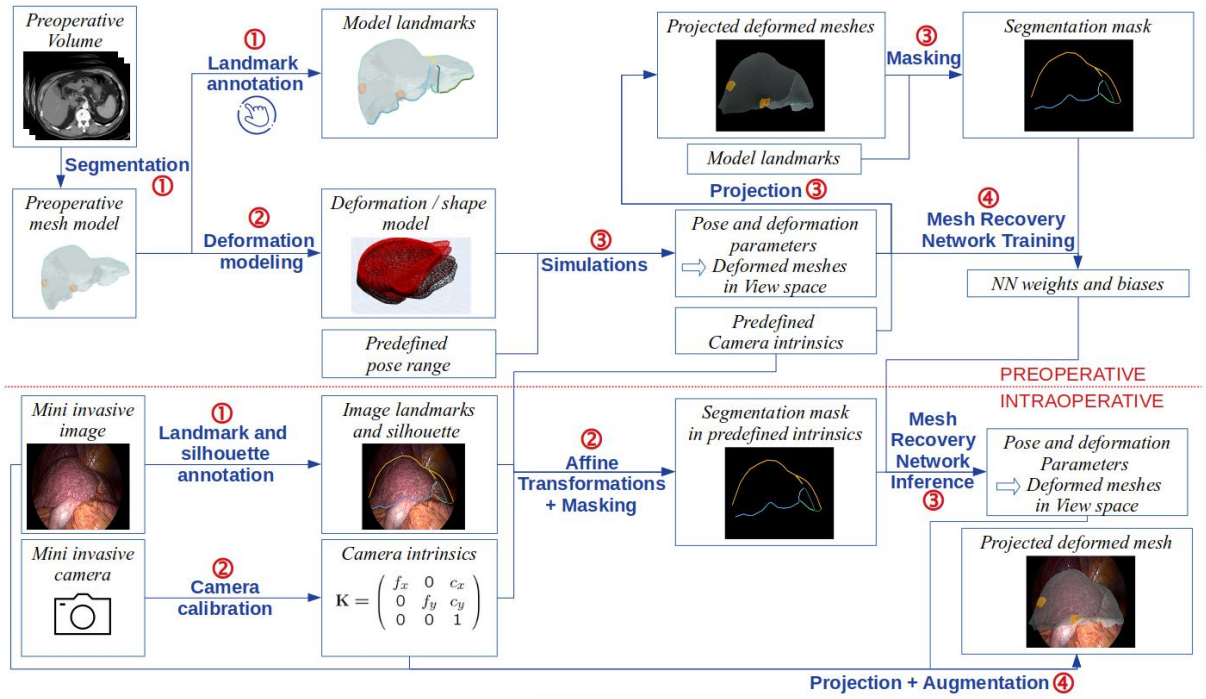
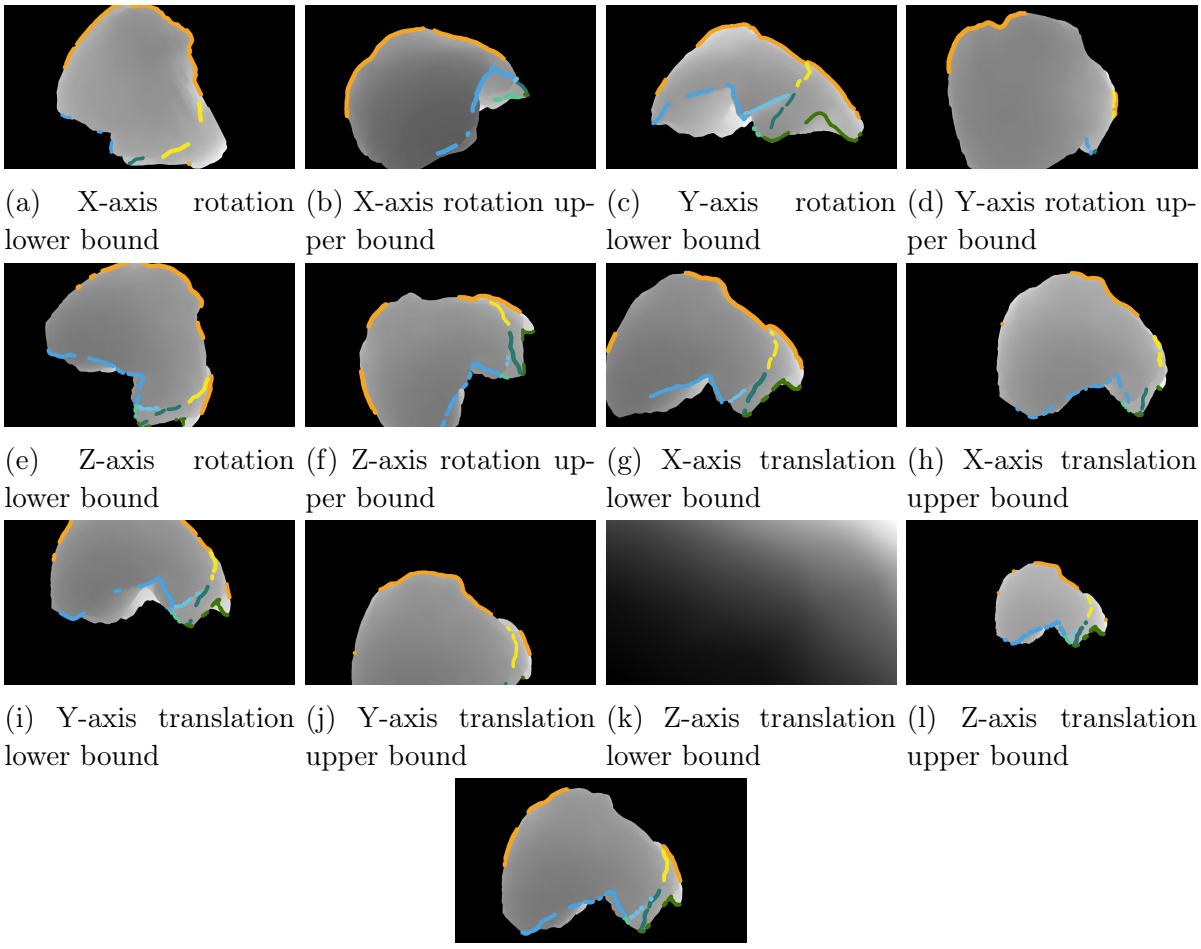


Figure 4.4: Proposed learning-based pipeline in MILR. (top) Preoperative steps: (1) reconstruct the 3D liver model from the CT-scan with anatomical landmarks, (2,3) synthesise pose and deformations and the corresponding anatomical landmark masks, and (4) train LMR. (bottom) Intraoperative steps: (1) segment the surgical image to recover the landmark masks, (2) compensate the intrinsic camera parameters, (3) infer the registration with LMR, and (4) augment the image.

Step 3: Deformation sampling and mask generation. We complete the k simulated shapes to form $l \gg k$, with $l = 40000$ shapes to serve as training dataset. We fix the camera intrinsics to a default value K_{default}, estimated from the laparoscope used for the first patient in our experiments. We generate l camera poses by composing a default typical pose with a random pose perturbation sampled in $SE(3)$. The default typical pose puts the liver in a frontal view where its anterior ridge, ligament and silhouette are mostly visible by the camera. In this aim, it uses X, Y, Z Euler rotation angles of $-65, 45, -45$ degrees and X, Y, Z translations of $0, 0, 175$ mm. The random pose perturbation uses a uniform distribution on the X, Y, Z rotation angles within $50, 30, 40$ degrees and on the X, Y, Z translations of $40, 40, 100$ mm. We use PyTorch3D with its default coordinate system. Figure 4.5 illustrates the above ranges for one of the patients from our experiments. We project the l shapes using a z-buffer to handle visibility and obtain the image primitives as contours. The dataset has up to $l = 40000$ pairs of shapes and image primitives, from which samples for which fewer than two primitives are visible are removed. We split the dataset in 80% training, 10% validation and 10% test. We convert the primitive contours into segmentation masks, which we randomly perturb to emulate the typical error of automatic detection. We finally transform the masks to distance maps, using the Distance Transform [Rosenfeld & Pfaltz 1966] which we min-max normalise to $[-1, 1]$, with 1 associated to the image diagonal length. Note that the extra $(l-k)$ shapes are sampled from a normal distribution using as standard deviation $\frac{2}{3}$ of the one from the



(a) X-axis rotation lower bound (b) X-axis rotation upper bound (c) Y-axis rotation lower bound (d) Y-axis rotation upper bound

(e) Z-axis rotation lower bound (f) Z-axis rotation upper bound (g) X-axis translation lower bound (h) X-axis translation upper bound

(i) Y-axis translation lower bound (j) Y-axis translation upper bound (k) Z-axis translation lower bound (l) Z-axis translation upper bound

(m) Initial position

Figure 4.5: Undeformed liver configurations obtained within the bounds of the pose simulation range, where only one parameter is modified compared with the reference configuration (m), for Patient 4. Examples of advanced simulated primitives are superimposed to the depth map and enlarged for visualisation purposes. Landmarks share the same colour code as figure 2.17.

initial k coefficients. This guarantees that 99% of the complete set of shapes is within twice the initial standard deviation.

Step 4: LMR training. We use a loss with three terms. First, the reprojection loss term, which is the MSD or ASD between the predicted and input primitives, see section 3.3.3. Second, the 3D mesh loss term, which is the Mean Absolute Error (MAE) of the euclidean distances between the predicted and GT mesh vertices. Third, the pose and deformation coefficient loss term, which is the MAE between the normalised predicted and GT pose and deformation coefficients. The last two terms are referred to as the 3D loss terms. We use weights of 60, 0.1 and 1 respectively for the three terms. We use the Stochastic Gradient Descent (SGD) optimiser with 8 samples in the batch, 3 IEF iterations and 50 training epochs. We use the same learning rate for the encoder and regressor, fixed to 1e-3 for the first 45 epochs and then set to 1e-4 with a weight decay of 1e-4.

4.4.2 Intraoperative Stage

We use four intraoperative steps. Step 1 extracts the primitives automatically by means of a segmentation network, among the best performing ones described in chapter 3. Step 2 calibrates the camera when surgery starts, giving K_{actual}. The parameters were unknown preoperatively; recall that default intrinsics K_{default} were used instead. LMR was thus trained to handle images with different intrinsics than the ones of the actual surgical image. Step 2 copes with this difference by adapting the primitive segmentation masks prior to their use by LMR, by applying a 2D affine warp $A = K_{\text{default}} K_{\text{actual}}^{-1}$ [Fuentes-Jimenez *et al.* 2022]. Step 3 performs registration using LMR and step 4 uses the registration to augment the image with the hidden anatomical structures transferred from the preoperative 3D model.

4.5 Dataset and Evaluation

In order to evaluate the registration, we use three datasets:

- The open-source RT-GT dataset, defined in section 3.3, with GT tumour contour from ultrasound slice in the mini-invasive camera space.
- The phantom dataset from [Espinel *et al.* 2021]. A preoperative model subject to 10 non-rigid synthetic deformations was 3D printed for each deformation. 10 views were captured for each, along with the camera parameters, and the optimal poses were computed. The surface GT mesh is thus known for each image. The landmarks and silhouette are manually annotated for each image and the model.
- Synthetic test datasets built from step 3 of the LMR preoperative stage, which associates masks to surface meshes, see section 4.4.1.

4.5.1 Evaluation Criteria

For the phantom and synthetic datasets, we can evaluate the MAE in mm between predicted and target liver or tumour vertices. For the pose estimation, we also evaluate the orientation error angle $\theta = \arccos \frac{\text{Tr}\,(R\hat{R}^T)-1}{2}$ in degrees (°) which is computed from the optimal and estimated pose rotation matrices $R, \hat{R} \in SO(3)$.

We also use the CD2T landmark reprojection error defined in section 3.3.3, in % of the image diagonal length. In addition, we compute the Target Registration Error (TRE) [Rabbani *et al.* 2022] in mm between each GT tumour profile and the predicted tumour volume, for the RT-GT dataset.

4.5.2 Registration Method Details

The thresholds T used in iterative pose estimation algorithm (section 4.3) comprise 6 values between 0.05% and 25% of the image diagonal (1-500 px in 1080p images). The number of iterations for steps ② and ③ are respectively set to 6 and 12.

Optimisation (section 4.2.2.2) is used for the deformable registration, following the pose estimation. The optimisation employs $\alpha = 0.5$ and $\beta = 1$ in equation 4.6. By

	GT vs optimal pose	GT vs predicted pose a) split ridge model b) single ridge model	Optimal vs predicted pose a)
Error	MAE(mm)	MAE (mm) Steps ①/②/③	$\theta(^\circ)$ ①/②/③
Mean	$7.7^{\pm 0.7}$	$a)68.8/36.8/15.9^{\pm 2.9}$ $b)178.5/109.8/95^{\pm 28.6}$	29/20/11
Median	7.4	$a)51.0/24.0/11.6$ $b)108.3/66.9/19.7$	23/14/8
$\eta_{.90}$	13.5	$a)112.2/69.5/27.1$ $b)359.5/291.1/338.1$	45/39/17

Table 4.1: Pose estimation accuracy on a phantom dataset. MAE statistics are estimated across images between predicted pose from a) split and b) single ridge models and GT for each algorithm step. Optimal pose is also evaluated. The rotation angle between predicted and optimal poses is computed.

default, we use the Neo-Hookean constitutive model with Young modulus of 0.006 MPa and Poisson coefficient of 0.49 for computing the strain energy cost term in equation 4.6, see section 2.3.1.2. We use an equivalent method of the Levenberg-Marquardt, see section 2.4.5, for solving the non-linear least square problem described by equation 4.6. It is dedicated to sparse matrices through trust region reflective algorithm with a specific solver [Fong & Saunders 2011], from scipy.

We compare several deformation models referred to as FEM-OHA-G, FEM-OHA-L, FEM-OLA-G, FEM-OLA-L, FFD-G and FFD-L, LLE. -G and -L respectively state for global and local truncated SVD whose parameters are described in sections 2.3.3.1, 2.3.3.2. FEM, FFD and LLE parameters are respectively described in sections 2.3.1, 2.3.2, 2.3.4. Ogden High Amplitude (OHA) (5 N) and Ogden Low Amplitude (OLA) (0.5 N), related to FEM, are described as the simulation details of FEM in section 2.3.1.

4.5.3 Pose Estimation Evaluation

The proposed pose estimation method, referred to as Visibility-Aware Pose Estimation (VAPE), is first evaluated on the phantom dataset, using the MAE and the orientation error angle θ criteria, see table 4.1. It either uses the proposed split ridge model or the single ridge model, see section 2.5.1. We evaluate the results after each algorithm step (①, ②, ③), see section 4.3.

We first compute MAE between GT and both optimal and predicted poses. The predicted pose using the proposed split ridge model obtains mean and median errors of respectively 15.9 mm and 11.6 mm in comparison to the 7.7 mm and 7.4 mm ones of the optimal pose, due to the simulated deformations. Using the single ridge model obtains substantially worse results due to the wrong correspondences in fronto-lateral views.

We can see that each refinement step improves the previous MAE result. This is also the case for the orientation which gets progressively closer to the optimal one. 90% of the views obtain a MAE lower than 2.7 cm and a orientation error lower than 17°. The largest

	TRE (mm)		CD2T reprojection error (% of the image diagonal length)	
Method	Average	Average without P2	Average	Average without P2
Manual pose	$24.4^{\pm 10.2}$	$20.6^{\pm 8.6}$	$4.5^{\pm 3.0}$	$3.1^{\pm 1.3}$
VAPE	$24.7^{\pm 24.5}$	$\mathbf{12.6^{\pm 4.3}}$	$2.4^{\pm 0.9}$	$\underline{2.2^{\pm 1.0}}$
Optimisation	$28.4^{\pm 26.9}$	$15.0^{\pm 3.9}$	$3.0^{\pm 2.0}$	$\underline{2.2^{\pm 1.3}}$
LMR pose	$29.8^{\pm 14.3}$	$22.7^{\pm 2.1}$	$3.3^{\pm 1.1}$	$3.0^{\pm 1.1}$
VAPE+optimisation	$26.1^{\pm 26.8}$	$\underline{12.8^{\pm 4.0}}$	$2.0^{\pm 0.8}$	$\mathbf{1.8^{\pm 0.8}}$

Table 4.2: Comparison of pose estimation methods on the RT-GT set. In blue, the results for manual pose estimation.

	Synthetic Pose	
Average	MAE (mm)	CD2T reprojection error (% of the image diagonal length)
VAPE	$\underline{6.4^{\pm 4.5}}$	$\underline{0.6^{\pm 0.3}}$
Optimisation	$13.6^{\pm 5.4}$	$0.8^{\pm 0.3}$
LMR pose	$\mathbf{3.1^{\pm 0.4}}$	$\mathbf{0.5^{\pm 0.1}}$

Table 4.3: Comparison of automatic pose estimation methods on the RT-GT set on the synthetic pose test dataset.

errors are obtained for side views where very few landmarks are visible, and should not be considered in actual liver laparoscopy.

VAPE pose estimation performance is confirmed in the clinical RT-GT dataset, when Patient 2 (P2) is discarded from the evaluation. Indeed, the PnP problem is ill-posed in this procedure owing to a strong non-rigid deformation induced by the ultrasound probe and the narrowness of the views. As a consequence, all automatic initialisation method fail for this procedure (e.g. figure 4.7), as denoted by the large difference between average tumour TRE with and without P2 as well as their standard deviation difference. This configuration should be avoided in practice, as this does not enter into the general problem conditions. Hence, while also displaying the average results with P2 for future reference, we focus on the average results without P2, for the rest of the evaluations.

VAPE leads to an average tumour TRE 8 mm lower than the manual initialisation one, with a lower CD2T reprojection error. We also compare VAPE to optimisation of pose parameters, replacing deformation parameters in equation 4.6. They are initialised to the default typical pose used in LMR (section 4.4.1). In addition, for each patient, LMR only using pose parameters is trained on synthetic data based on random pose sampling, and is compared to the previous methods. VAPE obtains the best results, while optimisation and LMR only using pose respectively obtain an average error of about 3 and 9 mm more. When using VAPE estimated pose as initialisation of optimisation, the CD2T reprojection error decreases but the tumour TRE does not.

CD2T reprojection errors also indicate that LMR-pose does not fit the image landmarks as well as the two other methods. This is expected as it is trained on a simulation domain without deformations. However, when compared on the synthetic test dataset

from the same patient meshes, see table 4.3, LMR obtains the best results for both tumour MAE (about 3 mm) and CD2T reprojection error. Note that optimisation gets the worse results, with a MAE about 1 cm more in average, even though it starts from the same initial pose. VAPE obtains fairly good results with a MAE about 6 mm. Both methods obtain CD2T reprojection errors below 1% of the image diagonal.

4.5.4 Evaluation of the Complete Registration

We compare both the influence of the registration methods and the deformation models, whose acronyms are given in section 4.5.2, on the tumour TRE and CD2T results for the RT-GT dataset, see table 4.4. We use three baselines and comparing blocks:

1. The manual pose, given by an expert and used as initialisation of related works of [Adagolodjo *et al.* 2017] and [Koo *et al.* 2017a] also initialises the optimisation of deformation parameters from different models. We compare the performance between all these methods and options.

2. The VAPE estimated pose initialises the optimisation of deformation parameters from different models, whose performance is compared.

3. The LMR-pose is used as a reference baseline result. LMR using deformation models built from global truncated SVD (and trained on simulated data from these models) are compared.

4.5.4.1 Deformation Models

From the poses estimated manually (baseline 1) and automatically (baseline 2), we first compare the optimisation results between the same deformation models only differing by the dimension reduction. It can be seen that the tumour TRE is generally reduced for global truncated SVD, for both FEM and FFD-based deformation models. We refer to such reduced dimension models in the rest of the evaluation. Between simulation-based (FEM and FFD) and geometrically-based (LLE) deformation models, the latter obtains lower CD2T reprojection errors but higher tumour TRE ones. This suggests that this model can highly deform for fitting the image landmarks well, but its deformations can be unrealistic. We also use baseline 3 for comparing simulation-based deformation models. FEM obtains better results in most cases. In particular, FEM-OHA-G is in all cases the best or second-best performing method, in terms of tumour TRE. This suggests that it is more realistic than other deformation models, as we would expect of a model based on biomechanical simulations. It generalises better than OLA, probably due to the higher simulated nodal force amplitudes. However, they do not reproduce all the constraints in mini-invasive conditions and these results should thus be observed cautiously, as they would need extensive experiments on numerous patients for confirming this trend.

4.5.4.2 Registration Methods

Another comparison, between optimisation and other related works such as [Koo *et al.* 2017a] which uses position-based dynamics, see section 4.2.2.1, can be per-

		TRE (mm)		CD2T reprojection error (% of the image diagonal length)	
Method	Deformation model	Average	Average without P2	Average	Average without P2
1. Manual pose	None	$24.4^{\pm 10.2}$	$20.6^{\pm 8.6}$	$4.5^{\pm 3.0}$	$3.1^{\pm 1.3}$
Manual pose + [Adagolodjo *et al.* 2017]	NA	$22.4^{\pm 12.9}$	$\underline{17.5^{\pm 10.2}}$	NA	NA
Manual pose+ [Koo *et al.* 2017a]	NA	$23.0^{\pm 12.4}$	$17.6^{\pm 7.8}$	NA	NA
Manual pose+ optimisation	FEM-OHA-G	$23.1^{\pm 16.5}$	$\mathbf{15.3^{\pm 7.1}}$	$3.9^{\pm 4.5}$	$\underline{1.6^{\pm 0.7}}$
	FEM-OHA-L	$23.4^{\pm 10.9}$	$19.4^{\pm 9.0}$	$3.7^{\pm 3.6}$	$1.9^{\pm 1.4}$
	FEM-OLA-G	$22.9^{\pm 8.8}$	$19.7^{\pm 7.5}$	$4.6^{\pm 3.9}$	$2.7^{\pm 0.8}$
	FEM-OLA-L	$24.3^{\pm 10.2}$	$20.6^{\pm 8.6}$	$4.4^{\pm 3.1}$	$2.9^{\pm 1.4}$
	FFD-G	$25.2^{\pm 13.6}$	$19.9^{\pm 10.3}$	$5.0^{\pm 5.4}$	$2.3^{\pm 1.1}$
	FFD-L	$24.1^{\pm 10.4}$	$20.2^{\pm 8.5}$	$3.6^{\pm 3.6}$	$1.9^{\pm 1.2}$
	LLE	$24.1^{\pm 10.7}$	$20.2^{\pm 8.9}$	$3.0^{\pm 3.7}$	$\mathbf{1.2^{\pm 1.3}}$
2. VAPE	None	$24.7^{\pm 24.5}$	$12.6^{\pm 4.3}$	$2.4^{\pm 0.9}$	$2.2^{\pm 1.0}$
VAPE+ optimisation	FEM-OHA-G	$23.6^{\pm 25.4}$	$\underline{11.0^{\pm 4.1}}$	$2.1^{\pm 1.2}$	$1.5^{\pm 0.4}$
	FEM-OHA-L	$24.3^{\pm 24.6}$	$12.1^{\pm 3.2}$	$2.0^{\pm 1.1}$	$1.6^{\pm 1.0}$
	FEM-OLA-G	$25.5^{\pm 23.9}$	$13.6^{\pm 3.5}$	$2.9^{\pm 1.6}$	$2.4^{\pm 1.3}$
	FEM-OLA-L	$24.5^{\pm 24.5}$	$12.3^{\pm 3.9}$	$2.0^{\pm 1.1}$	$1.6^{\pm 1.0}$
	FFD-G	$22.9^{\pm 25.3}$	$\mathbf{10.3^{\pm 1.6}}$	$1.8^{\pm 1.1}$	$\underline{1.3^{\pm 0.4}}$
	FFD-L	$24.4^{\pm 24.5}$	$12.2^{\pm 3.6}$	$2.1^{\pm 1.2}$	$1.7^{\pm 1.1}$
	LLE	$24.2^{\pm 24.7}$	$11.9^{\pm 2.9}$	$1.6^{\pm 1.4}$	$\mathbf{1.2^{\pm 1.3}}$
3. LMR-Pose	None	$29.8^{\pm 14.3}$	$22.7^{\pm 2.1}$	$3.3^{\pm 1.1}$	$3.0^{\pm 1.1}$
LMR	FEM-OHA-G	$29.3^{\pm 23.0}$	$\underline{17.9^{\pm 2.8}}$	$3.0^{\pm 1.0}$	$\mathbf{2.5^{\pm 0.4}}$
	FEM-OLA-G	$26.5^{\pm 18.2}$	$\mathbf{17.3^{\pm 0.3}}$	$3.8^{\pm 1.2}$	$3.6^{\pm 1.4}$
	FFD-G	$27.1^{\pm 13.9}$	$20.5^{\pm 5.1}$	$3.7^{\pm 1.5}$	$3.0^{\pm 1.0}$

Table 4.4: Comparison of the results between deformable registration methods and models on the RT-GT set for 3 blocks. In blue, the associated pose results for each block baseline method without deformation. NA states for Not Available.

formed from block 1. The latter obtain tumour TRE errors 2 mm higher than the optimisation based on FEM-OHA-G (which obtains an average error of around 15 mm) and lower than the optimisation based on other deformation models. However, both obtain higher final TRE errors than the one from VAPE pose estimation only (block 2), which obtains a tumour TRE of 12.6 mm. This highlights the critical importance of pose estimation on the results of methods performing pose then deformation.

The LMR (block 3), combining pose and deformation parameters, performs on par with the manual pose followed by related work methods, while obtaining higher CD2T reprojection errors than the other methods. Figure 4.7 shows the detected and reprojected primitives for OLA for both VAPE + optimisation as well as LMR-FEM-OLA-G, and confirm that the reprojection errors of the latter are higher. We also evaluate the LMR using the test dataset generated along with the LMR training data for each patient, see section 4.4.1. For both OLA and OHA, we measure the MAE of the euclidean distances

MAE (mm)	Average	Average without P2
LMR-FEM-OHA-G	$16.4^{\pm 3.5}$	$15.1^{\pm 2.7}$
LMR-FEM-OLA-G	$8.5^{\pm 2.3}$	$9.3^{\pm 2.0}$

Table 4.5: MAE results for LMR based on FEM for OHA and OLA on their own synthetic dataset.

Loss terms	Both	Reprojection only	3D only
MAE (mm)	$\mathbf{8.5^{\pm 2.3}}$	$34.5^{\pm 15.7}$	$\underline{9.7^{\pm 2.2}}$

Table 4.6: Ablation study on the loss terms, using either both reprojection and 3D, reprojection only or 3D only terms, for LMR with FEM-OLA-G deformation model on the FEM-OLA-G simulation test dataset.

between the estimated and GT liver mesh vertices, see table 4.5. The average MAE is of 8.5 mm for OLA, while OHA obtains a larger MAE, about twice as large as the first. We thus assume that the discrepancy between results on synthetic and real datasets for OLA is due to the low amplitude of the deformation simulations which do not fully span the real domain. For OHA, there is less difference, however, the error is quite high. This could indicate either some ambiguities of pose and deformation parameter combination (several combinations leading to similar projections), or detailed position information not enough conveyed in the current pipeline. Indeed, only low-resolution information from the last encoder layer is transferred to the regressor.

Table 4.6 performs the ablation of the loss terms for OLA on the synthetic test sets. This confirms that using both reprojection and 3D terms in the loss function for LMR training results in lower MAE results. We can also notice that only training with the reprojection loss leads to much higher average MAE of 34.5 mm, which could confirm the ambiguities between combinations of pose and deformation parameters. Figure 4.6 shows results of each intermediate IEF iteration for Patient 1 on its synthetic test dataset. It can be seen that the landmarks are usually reprojected closer to the target at each iteration, and the MAE also usually decreases.

4.5.4.3 Automatic versus Manual Segmentation

We also evaluate the influence of the segmentation method on VAPE and LMR-FEM-OHA-G results, see table 4.7. While the Mask2Former-S obtains the best segmentation scores, it results in higher tumour TRE than other best performing segmentation networks, for both VAPE and LMR. We assume this is due to wrong class parts that it sometimes predicts, see figure 3.17, which can lead to 2D correspondences apart for a same landmark and thus a difficult registration problem. Mask2Former-R and the COSNet gets slightly lower tumour TRE results. The latter, combining information from multiple images, helps to reduce the CD2T reprojection but tumour TRE results are still much higher than the ones obtained with manual segmentation. In contrast, combining information from multiple masks in addition to images allows the attainment of equivalent tumour TRE and CD2T reprojection errors (even better for the VAPE). This reinforces the idea to transform Mask2Former-S into COSNet and STM equivalents in order to benefit from the

	TRE (mm)		CD2T reprojection error (% of the image diagonal length)	
Segmentation	Average	Average without P2	Average	Average without P2
VAPE				
Manual	$24.7^{\pm 24.5}$	$12.6^{\pm 4.3}$	$2.4^{\pm 0.9}$	$2.2^{\pm 1.0}$
Mask2Former-R	$43.3^{\pm 32.3}$	$27.8^{\pm 10.6}$	$3.9^{\pm 1.4}$	$3.3^{\pm 0.7}$
Mask2Former-S	$46.9^{\pm 30.8}$	$32.5^{\pm 13.6}$	$4.3^{\pm 2.4}$	$3.3^{\pm 1.7}$
COSNet	$47.6^{\pm 40.9}$	$27.7^{\pm 11.1}$	$3.4^{\pm 2.2}$	$2.3^{\pm 0.6}$
STM	$56.4^{\pm 86.2}$	$\mathbf{13.3^{\pm 3.2}}$	$2.2^{\pm 1.2}$	$\mathbf{1.6^{\pm 0.6}}$
LMR-FEM-OHA-G				
Manual	$29.3^{\pm 23.0}$	$17.9^{\pm 2.8}$	$3.0^{\pm 1.0}$	$2.5^{\pm 0.4}$
Mask2Former-R	$34.4^{\pm 14.6}$	$30.3^{\pm 14.8}$	$6.2^{\pm 3.3}$	$5.7^{\pm 3.9}$
Mask2Former-S	$41.8^{\pm 12.6}$	$38.4^{\pm 12.8}$	$6.4^{\pm 3.2}$	$5.6^{\pm 3.5}$
COSNet	$39.5^{\pm 21.1}$	$33.0^{\pm 20.5}$	$6.1^{\pm 4.0}$	$5.3^{\pm 4.4}$
STM	$28.5^{\pm 22.6}$	$\mathbf{17.4^{\pm 5.0}}$	$3.2^{\pm 1.2}$	$\mathbf{2.6^{\pm 0.8}}$

Table 4.7: Comparison of the influence of the segmentation method on VAPE pose estimation results. In blue, the results obtained using manual segmentation. Reprojection error is computed on the automatic detected landmarks.

effect of these additional information. However, all these segmentation results should be considered cautiously due to the difference of image domain between the RT-GT dataset and the LaparoLiver training dataset caused by the presence of the ultrasound probe, which thus does not generalise as well as it would on standard mini-invasive conditions, see section 3.3.4.

		TRE (mm)		CD2T reprojection error (% of the image diagonal length)	
Method	Ligament	Average	Average without P2	Average	Average without P2
VAPE pose-only	Yes	$24.7^{\pm 24.5}$	$12.6^{\pm 4.3}$	$2.4^{\pm 0.9}$	$\mathbf{2.2^{\pm 1.0}}$
	No	$25.1^{\pm 28.4}$	$\mathbf{11.0^{\pm 2.3}}$	$2.9^{\pm 1.8}$	$\mathbf{2.2^{\pm 1.2}}$
VAPE+optimisation FFD-G	Yes	$22.9^{\pm 25.3}$	$10.3^{\pm 1.6}$	$1.8^{\pm 1.1}$	$1.3^{\pm 0.4}$
	No	$24.2^{\pm 29.1}$	$\mathbf{9.6^{\pm 2.2}}$	$1.9^{\pm 1.9}$	$\mathbf{0.9^{\pm 0.6}}$

Table 4.8: Influence of the presence of the ligament on the registration results for the RT-GT set.

4.5.4.4 Influence of the Falciform Ligament

We also evaluate if taking into account the falciform ligament is fruitful for 3D-2D registration methods. Recall that it is not seen in preoperative imaging and can only be

annotated on the preoperative mesh in a coarse manner. Results tend to show that the supposed ligament position can harm the accuracy of the registration, as shown in table 4.8. Indeed, without ligament, the tumour TRE is slightly lower on the RT-GT dataset for both VAPE only and VAPE then optimisation with FFD-G.

4.5.4.5 Runtimes

All experiments were run on a medium-end Nvidia GeForce RTX 2080 GPU. The inference time was 5 ± 3 s for VAPE, which is competitive to manual pose estimation by an expert. For optimisation and LMR inference, the average runtimes are respectively 43 ± 47 s and 3.4 ± 0.2 ms. FEM simulation and mask generation took 4 hours to run and LMR 16 hours to train per patient. This can be reduced by using a high-end GPU, but is already compatible with the typical delay between preoperative scanning and surgery. Primitive detection with Mask2Former-R and Mask2Former-S respectively takes around 33 ms and 45 ms, while COSNet and STM respectively take around 17 ms and 9 ms for one pair. This means that our automatic segmentation with LMR pipeline allows quasi-real-time intraoperative registration when using independent image segmentation networks or few paired information for the other networks.

4.6 Conclusion

We have proposed two ways of automating the registration of a preoperative liver model on a mini-invasive image.

- The first consists in automating each registration step of the background pipeline, consisting of manual pose estimation followed with deformable registration. Our pose estimation method (VAPE) uses RANSAC-based PnP for estimating the pose from 3D-2D correspondences. However, it processes in an iterative manner in 3 coarse-to-fine steps, refining the 3D-2D correspondences based on the visibility of the model landmarks from previous estimates. This approach is validated on a phantom dataset and a real one (RT-GT), even obtaining a lower tumour TRE than manual pose estimation by an expert, on the latter, while taking on average a few seconds, being thus very competitive to manual initialisation. The deformable registration method was already automatic, and was chosen as the optimisation of deformation parameters from a model. Registration results from multiple deformation models, described in section 2.3 have been compared. The FEM-OHA-G based on high amplitude biomechanical simulations, obtain the overall best performance. It compares favourably to related works, starting from the same pose. However, the deformable registration has few influence on the accuracy of the estimated tumour position. Indeed, it mainly depends on the pose estimation result and can only slightly improve it.

- The second consists in proposing a new automatic registration pipeline based on a mesh recovery network with an encoder-regressor architecture, building a bridge between human body shape reconstruction field and ours. The regressor contains an iterative error feedback (IEF) loop. The network inputs a segmentation mask

and outputs both pose and patient-specific deformation parameters. Training is performed with associated simulated inputs and outputs. It succeeds in obtaining results on par with manual pose followed with deformable registration methods from related work. In addition, it transfers most of the computational load in preoperative steps, for training, while its inference takes a few ms, allowing real-time intraoperative registration. However, it obtains higher reprojection errors than other methods. This could be due to the fact that only low-resolution information from the last encoder layer is transferred to the regressor. A solution could be the replacement of the IEF loop in the regression network with Pyramidal Mesh Alignment Feedback (PyMAF) [Zhang *et al.* 2021]. IEF reuses the same global low resolution features in its feedback loop, so the mesh-image misalignment in the inference phase is difficult to perceive by the regressor. In contrast, PyMAF uses an upsampling path (decoder) in order to produce features of finer resolution. The predicted parameters from low resolution features enable the obtention of mesh-aligned information in finer resolution features, which can give a more direct feedback for parameter correction. Another solution could consist in representing the landmarks in another format with higher resolution information, such as sampled landmark point coordinates [Mhiri *et al.* 2024], or curve parameters reconstructed from the segmentation masks [Gao *et al.* 2019].

Both approaches heavily rely on a correct annotation of the image landmarks for relevant 3D/2D correspondences. When providing landmarks from best performing segmentation methods described in chapter 3, but not generalising well to the non-standard RT-GT dataset, the accuracy of the estimated tumour highly degrades, except for the STM, which takes information from other images and masks. Similarly, coarsely annotating the falciform ligament on the 3D model can also harm the registration accuracy.

Otherwise, without prior knowledge of the deformation applied to the preoperative model, first initialising the pose is almost mandatory. However, if the pneumoperitoneum could be modelled, using for example data from artificial pneumoperitoneum [Wang *et al.* 2021], see section 2.7, the logical order would be the application of deformation then pose. This is why studies should be pursued in the deformation modelling, in particular related to the artificial pneumoperitoneum, which is the main loading applied to the liver in exploratory mini-invasive conditions.

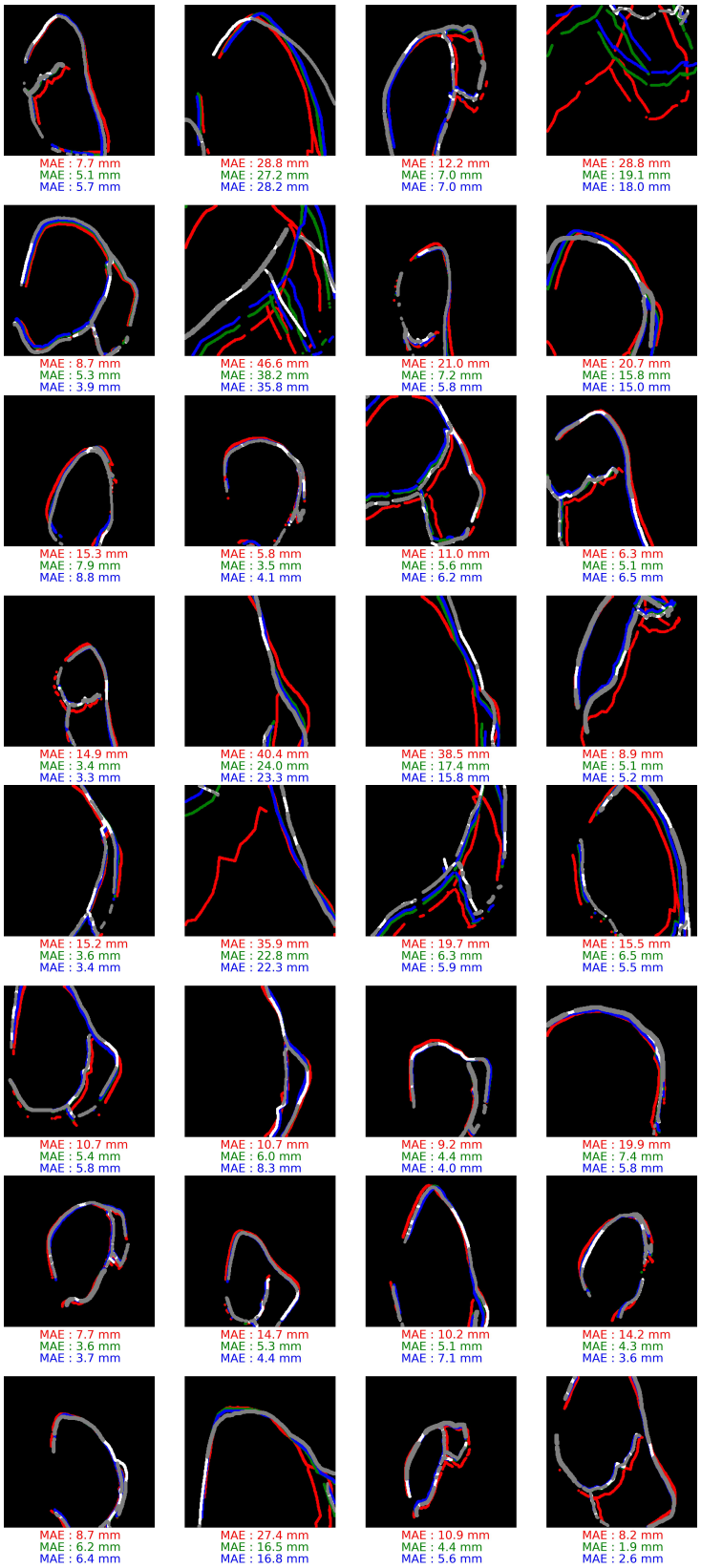
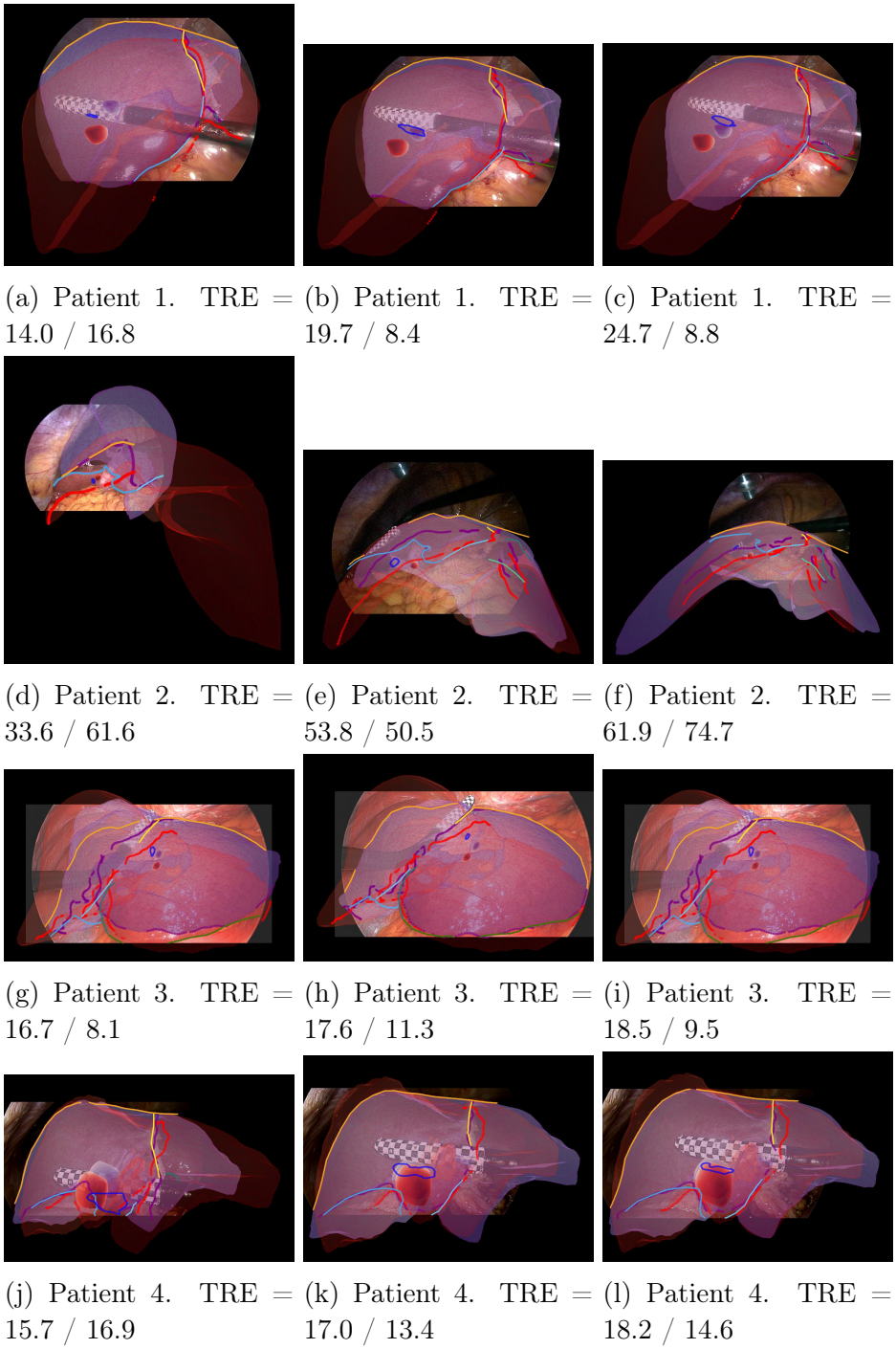


Figure 4.6: Results on 32 samples of the test dataset for Patient 1, using OLA. In gray, the initial segmentation mask, in white the projected GT primitives and in red, green and blue the respective projected results of each IEF iteration with associated MAE values.



(a) Patient 1. TRE = 14.0 / 16.8
(b) Patient 1. TRE = 19.7 / 8.4
(c) Patient 1. TRE = 24.7 / 8.8
(d) Patient 2. TRE = 33.6 / 61.6
(e) Patient 2. TRE = 53.8 / 50.5
(f) Patient 2. TRE = 61.9 / 74.7
(g) Patient 3. TRE = 16.7 / 8.1
(h) Patient 3. TRE = 17.6 / 11.3
(i) Patient 3. TRE = 18.5 / 9.5
(j) Patient 4. TRE = 15.7 / 16.9
(k) Patient 4. TRE = 17.0 / 13.4
(l) Patient 4. TRE = 18.2 / 14.6

Figure 4.7: Comparative results between the VAPE+optimisation and the proposed LMR-FEM-OLA-G method, for manual annotation of the primitives. Three images of each patient, selected as the minimal (left) and maximal (right) TRE, and the closest TRE to the mean (middle) of the LMR-FEM-OLA method. The TRE given in the captions are respectively for LMR-FEM-OLA / VAPE+optimisation. VAPE+optimisation results are in purple and LMR results are in red. Visible model landmarks are additionally shown using the same colour code. The projected tumour outline registered from ultrasound is in blue.

Chapter 5

Automatic Patient-Generic 3D/2D Registration

Contents

5.1 Introduction

The LMR method developed in section 4.4, hereinafter referred to as PS-LMR, is trained with data simulated from the patient's 3D model. This is a strong limitation: PS-LMR performs preoperative simulations and trains the network specifically. This requires about a day of computation and computational resources for each patient under the supervision and expertise of an engineer, hindering usage in clinical practice. In addition, it only works if the preoperative images of the patient are available.

Instead, we propose to employ a generic liver shape model, see section 5.2, and then to incorporate it into a PG-LMR pipeline, described in section 5.3. As a consequence, the same neural model can be used for all patients, without requiring additional training, hence facilitating usage in clinical practice. This also opens the way to a novel type of AR that we call *anatomical AR*, which consists in overlaying generic anatomical features from the PG liver model only, on the surgical images. Anatomical AR does not require

preoperative patient data and may reduce navigation time and contribute to surgeon training to MILS.

5.2 Patient-Generic Liver Shape Modelling

The generic modelling of the liver shape is highly challenging because of its substantial inter-subject morphological and localised variations [Netter 2014, Singh & Rabi 2019]. In the problem at hand, a patient liver shape is represented by mesh surface and possibly with inner vertex coordinates. It is usually reconstructed from preoperative imaging, see sections 2.1 and 2.2.

5.2.1 Related Work

Shape modelling has been attempted with two main approaches: statistical and learning-based ones. The statistical approach uses Statistical Shape Modelling (SSM) with a 3D Point Distribution Model (PDM) [Cootes *et al.* 1992]. Its construction requires a set of shapes in one-to-one correspondences. Then, the shape samples are aligned, using for instance Generalised Procrustes Analysis (GPA). Finally, the statistics of the set of shapes are captured using PCA, see section 2.3.3.1. The main difficulty resides in obtaining the one-to-one correspondences. They are usually obtained using a non-rigid point set registration method, where a template shape is registered to every other shapes. The probabilistic Coherent Point Drift is used in [Pellicer-Valero *et al.* 2020]. It simultaneously finds both the non-rigid transformation and the correspondence between two point sets without making any prior assumption of the transformation model except that of motion coherence [Myronenko *et al.* 2006]. Other non-rigid point set registration methods could be used, such as the Non-Rigid Iterative Closest Point (NR-ICP) [Amberg *et al.* 2007]. It is an iterative algorithm with two loops. From an initial high local stiffness weight, an outer loop successively lowers the weight in order to start fitting with global deformations and progressively refining it with more localised deformations. The inner loop determines the optimal deformation of a template for a given stiffness. It iteratively determines correspondences as nearest neighbours in the given configuration and then the optimal deformation according to a cost function, giving rise to new correspondences. The cost function is designed as a weighted sum on multiple distance terms, such as vertex-to-surface and surface landmark ones, as well as deformation regularisation terms such as a local stiffness one.

The learning-based approach uses the Neural Diffeomorphic Flow (NDF) model [Sun *et al.* 2022]. It utilises another representation of a surface, by a continuous volumetric field, the Surface Distance Field (SDF). The amplitude associated to a point in the field is the closest distance to the surface. The sign indicates if it is inside or outside the shape. The surface is implicitly represented as the zero-level set of the learnt function. DeepSDF [Park *et al.* 2019] attempts to regress the continuous SDF from point coordinates and a shape latent code vector using a NN. The zero-level set surface associated to a latent vector can then be retrieved evaluating numerous spatial samples and the mesh may be extracted using marching cubes, see section 2.2. A network can thus model a whole class of shapes. Deep Implicit Templates [Zheng *et al.* 2021] try to formulate

SDF as conditional deformations of an implicit template. Hence, the shape variance can be reflected by the SDF differences relative to a SDF template which captures their common structure. This formulation introduces correspondences between the shape instances. The NDF attempts to additionally preserve the topology in representing the deformation function as a conditional diffeomorphic flow, making the deformation field invertible.

Both previous methods try to match liver shapes without any prior knowledge of its anatomy. Though effective in fitting the shapes globally, corresponding parts with high morphological differences are likely to mismatch.

5.2.2 Generic Liver Shape Modelling with Anatomical Priors

In order to tackle this local mismatching issue, we propose to guide the matching through 14 anatomical surface feature point correspondences. We describe PG liver shape modelling[1]. Our first contribution is the preparation of a dataset of $K = 71$ registered 3D liver mesh models with $n = 4978$ vertices in one-to-one correspondence. We first reconstruct 3D meshes, define manual correspondences, register them densely, and find global correspondences. Our second contribution is two shape models constructed from the prepared data.

5.2.2.1 Data Source and Preprocessing

We use the first K preoperative CT scans of the v2 AMOS training database, which includes automatic liver segmentation results. We preprocess the data in three steps, described in sections 2.1 and 2.2. First, we manually correct the automatic segmentations, using 3D Slicer. Second, we perform triangular surface mesh reconstruction using marching cubes. Third, we perform Laplacian mesh smoothing, resampling to 5000 vertices with ACVD, and cleaning in order to make it manifold.

5.2.2.2 Definition of Sparse Anatomical Correspondences

Netter's classification [Netter 2014] shows that the liver has substantial shape variability. The reconstructed liver shapes thus exhibit important local variations, e.g. in the left lobe (figure 5.1). Non-rigid registration with local correspondence priors is therefore required, which we manually select as sparse point correspondences. Specifically, we propose the 14 anatomical feature points shown in figure 5.1.

5.2.2.3 Surface Registration

We register surface meshes from a reference one in a pair-wise manner. We select among the K ones, the one which exhibits the most neutral shape. This model (n°2) has $n = 4978$ vertices. Each other model is successively taken as the registration target, resulting in $K - 1$ deformations of the reference model. We first min-max normalise both the reference and target model coordinates to $[0, 1]^3$. We then proceed in three steps. First, we initialise the deformations with the interpolation of a polyharmonic Radial Basis Functions (RBF) [Buhmann 2000] of degree 2 and radius 0.5, from pygem.

[1]Generic modelling data are available in https://encov.ip.uca.fr/ab/code_and_datasets/

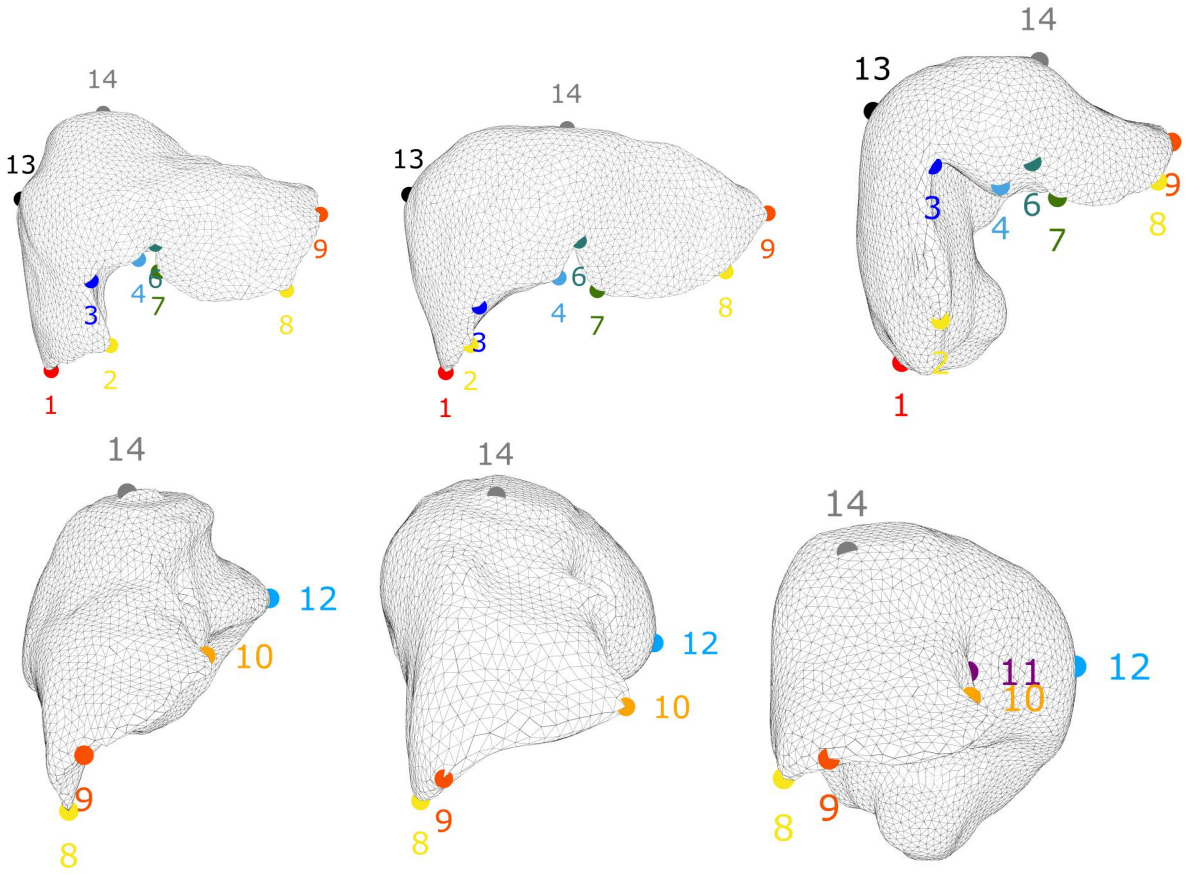
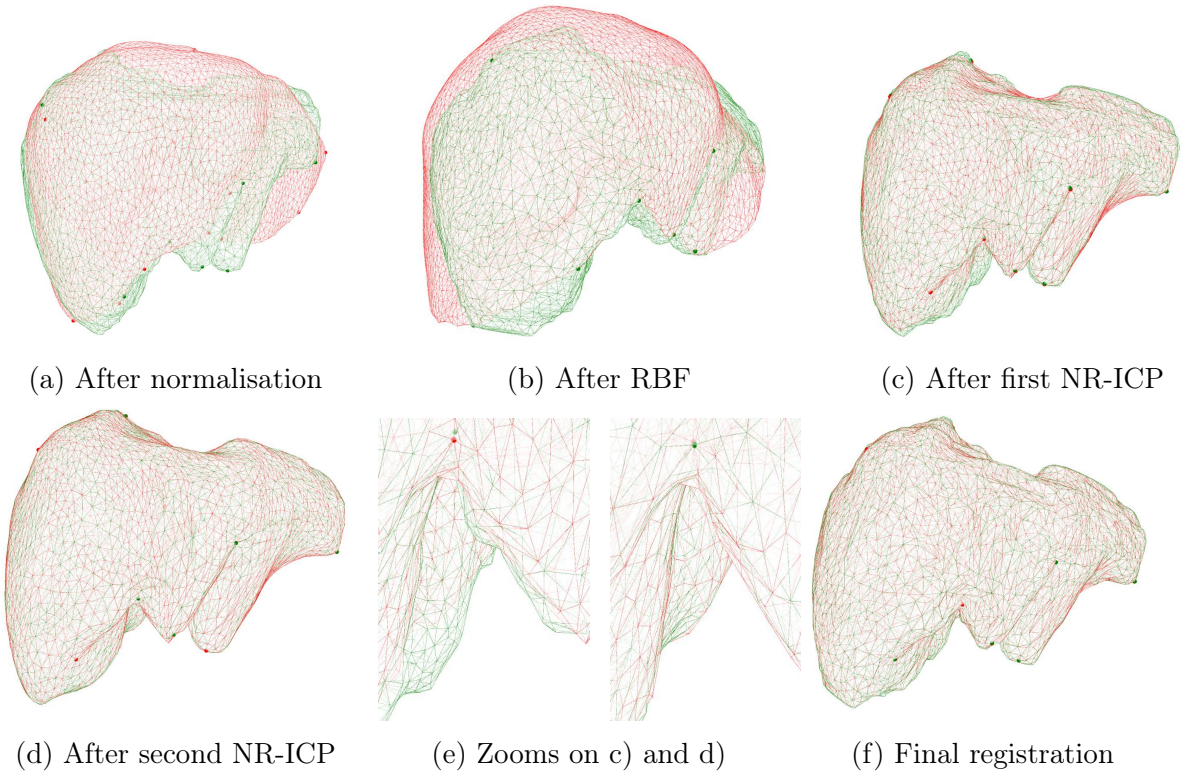


Figure 5.1: Liver shape examples, with frontal (top) and left (bottom) views. The proposed correspondences are shown with colour and number coding, 1) the extreme lower right point, 2) the lower anterior right point, 3) the gallbladder anterior right point, 4) the start of the anterior right ridge, 5) the start of the right rex recessus, 6) the start of the left rex recessus (usually same as point 5), 7) the start of the anterior left ridge, 8) the extreme anterior lower left point, 9) the extreme anterior upper left point, 10) the extreme posterior upper left point, 11) the centre of vena cava, 12) the extreme posterior right point, 13) the extreme upper right point, and 14) the extreme upper point.

The RBF control points are deformed from the reference feature points to the target ones. Second, we refine the deformations using NR-ICP [Amberg *et al.* 2007] of pytorch-nicp. We customise the cost function to use four terms: vertex-to-surface distances, feature point distances and Laplacian smoothing, with respective constant weights of 1, 5, and 100, and stiffness with decreasing weight from 50 to 0.2. We use 50 and 150 iterations respectively in the inner and outer loops.

Closest points are known to be asymmetrical between the two sets; we mitigate this issue in performing two runs of NR-ICP. The first run takes the reference model as the moving shape. In the second run, the registered reference model is taken as the target of the NR-ICP, while the initial one is the moving shape. Third, we eventually compute the correspondences of the n vertices of the reference model using the closest surface points from the final registered shapes.



(a) After normalisation (b) After RBF (c) After first NR-ICP

(d) After second NR-ICP (e) Zooms on c) and d) (f) Final registration

Figure 5.2: Registration steps. The reference and target meshes are respectively in red and green. The inputs are the normalised meshes a). After RBF interpolation b), the sparse correspondences coincide. After the first NR-ICP c), the deformed reference is closer to the target. After the second one d), the deformed target and reference are almost superimposed. The benefits of the second step is shown on a local part e). The closest deformed target surface points from the deformed reference vertices in d) are used as correspondences to produce the target mesh f) with the same connectivity as the reference.

5.2.2.4 Construction of the Models

We construct two models for the liver shape:

- The first one learns the liver shape from the data using SSM with a PDM. We first run GPA to rigidly align the K shapes and compute the average shape $\mu^{s\top} = [\mu_1^{s\top}, \cdots, \mu_n^{s\top}] \in \mathbb{R}^{3n}$. The two constructed models are linear combinations of m shape components $\phi_j^s = [\phi_{j,1}^s, \cdots, \phi_{j,n}^s]^\top$ for $j = 1, \ldots, m$, so that a shape $\hat{x}_i^s$ can be generated from configuration weights $\beta_i = [\beta_{i,1}, \cdots, \beta_{i,m}]^\top$ as $\hat{x}_i^s = \mu^s + \sum_{j=1}^m \beta_{i,j}\phi_j^s$. The first model then uses PCA on the K shape data to perform reduced order modelling and using the 'subspace loadings' as shape components. SSM makes the assumption that the shape data are representative of the population of human livers.

- The second model we construct is more generic, using a Gaussian Process Morphable Model (GPMM) [Lüthi *et al.* 2017] with locally-scaled or multi-scale Gaussian kernels, from Statismo. Concretely, this boils down to using combinations of Gaussian processes as shape components, from the mean (template) shape μ^s.

5.3 Patient-Generic Liver Mesh Recovery Framework

This PG model is applied to the LMR framework instead of the PS preoperative one and enables the first PG neural registration framework for MILS. However, the PS model can be registered to the PG one preoperatively and thus the PG-LMR can also allow PS 3D/2D registration. This is important to enable the augmentation of PS features such as tumour locations, on top of anatomical features such as the Couinaud segments, available from the PG model.

Figure 5.3 shows the proposed general pipeline for 3D-2D registration. The preoperative steps are split into two categories: the PG steps include generic shape modelling and 3D-2D registration network training; the PS steps include inner liver structure registration. These steps are explained in the following sections.

5.3.1 Liver Mesh Recovery Network

We use the LMR network, see section 4.4, for performing 3D/2D registration. However, we replace the original PS deformation parameters by the PG shape parameters of the selected model. The training, validation and test set simulations described in section 4.4.1 are performed accordingly, from 120000 initial simulations. All rotation angle (°) and translation (mm) ranges are respectively set to [−70,70] and [−50,50] for pose simulations,

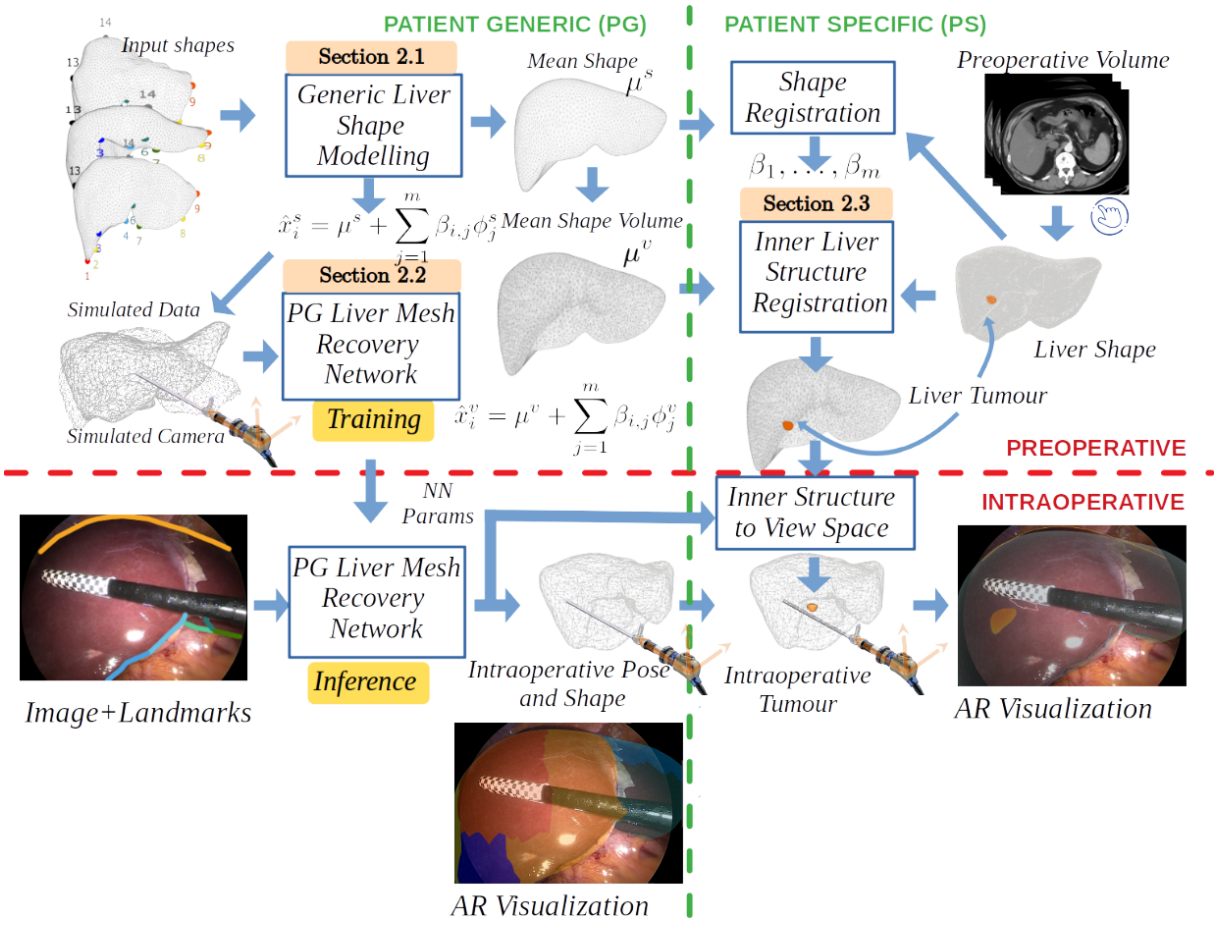


Figure 5.3: Proposed PG 3D-2D registration framework in MILS.

while keeping the default typical pose. Instead of the preoperative liver mesh used in PS-LMR, the initial one in the proposed PG-LMR uses the mean PG shape μ^s. At inference, distance maps from the annotated landmarks and silhouette of the input image are fed to the network which predicts pose and shape parameters and thus the liver mesh in view space.

5.3.2 Inner Liver Structure Registration

For PG AR guidance, generic anatomical features, such as the Couinaud segments, are directly found in the PG model and do not need an extra registration. For PS AR guidance, the query inner structures of the actual patient, such as tumours, should be registered to the predicted generic shape volume. This could be achieved in two ways: either using intraoperative volumetric registration for each predicted deformed liver; or preoperative registration for the template shape, which then requires one to transfer the inner structures to the predicted shape. We follow the latter, as its computation load is mostly preoperative. We propose to build a volumetric model $\hat{x}_i^v = \mu^v + \sum_{j=1}^{m} \beta_{i,j}\phi_j^v$, including the shape model such that $\mu^{v\top} = [\mu^{s\top}, \mu^{t\top}]$ where $\mu^{t\top} \in \mathbb{R}^{3p}$ with p the number of inner vertices, and $\phi^v = [\phi^s, \phi^t]$. Importantly, the scores β are common to the volumetric and shape models. First, we compute the volumetric template mesh μ^v using constrained Delaunay tetrahedrisation, see section 2.2.4, where $p = 1106$. We use it for the volumetric GPMM model. Alternatively, we can use a coarser result using classical Delaunay tetrahedrisation, in which case $p = 0$. We employ it for the 'volumetric' SSM, as our input registered shape samples are only surfacic. In order to find the $\hat{\beta}$ scores associated to the aligned preoperative surface vertices, we then solve the Ordinary Least-Squares problem applied on the linear shape model. We input them to the volumetric model for predicting the volumetric deformed vertices. Then, we compute the barycentric coordinates and associated tetraedra of the patient inner structure vertices with respect to the volumetric template. At inference, the inner structure vertices can be interpolated from the liver vertices predicted by the LMR network, using the volumetric model.

5.4 Dataset and Evaluation

The AMOS dataset, from which the generic shape model is built, is also used for evaluating registration and reconstruction errors. The main evaluation dataset for 3D-2D registration is the RT-GT, see section 3.3, and consists of 4 annotated livers where 3D tumour GT is retrieved in real annotated laparoscopic images from ultrasound images and appropriate calibrations.

5.4.1 Surface Registration

We first evaluate the surface registration method. The Mean Mean of closest Distances (MMD) between the registered and annotated landmarks on RT-GT is 4.8 ± 1.0mm. The largest discrepancies are in the landmark endpoints, whose annotation may be subjective. We also compare results between a) NDF, see section 5.2.1, which does not use prior knowledge of the anatomy, b) our method without corresponding feature points, i.e. with-

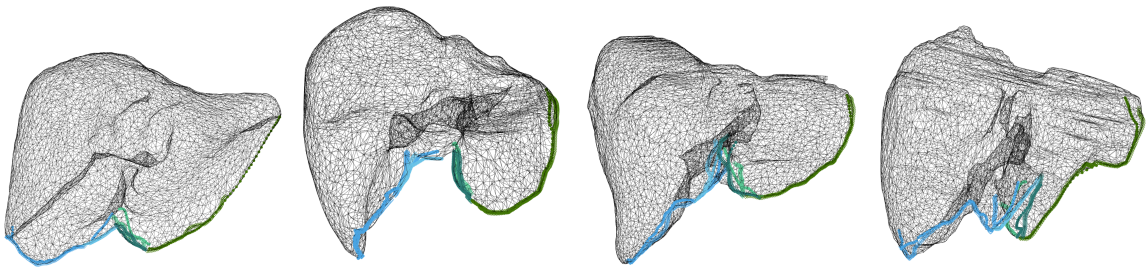
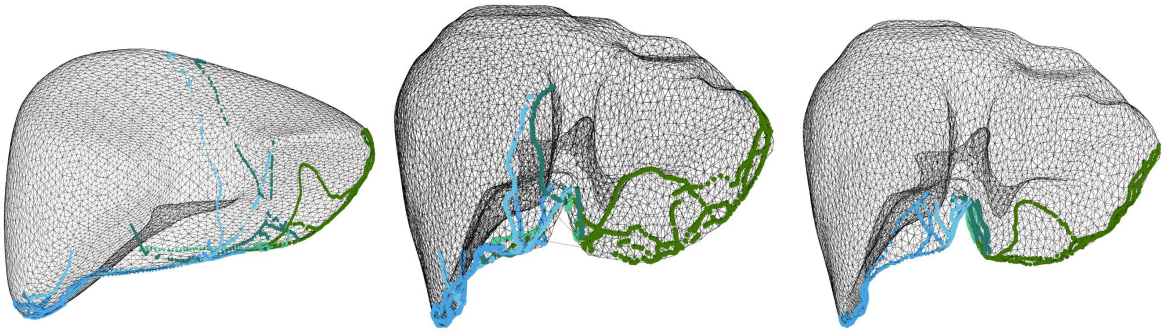


Figure 5.4: Comparison of registered (connected dots) and annotated (connected open diamonds) landmarks for the liver shapes of the RT-GT dataset. Corresponding landmarks share the same colour, defined in figure 2.17.



(a) NDF (b) 2 runs of NR-ICP without the feature point term (c) Proposed registration method

Figure 5.5: Annotated landmark correspondences on the template after surface registration of RT-GT meshes, using methods a) to c). Landmark colours are the same as figure 2.17.

out RBF and the feature point term in the two runs of NR-ICP, and c) our method. NDF is trained using the publicly available code and data. For each RT-GT mesh, we determine the correspondences of the annotated landmarks on the template, i.e the closest registered surface points. The template coordinates are min-max normalised. We evaluate the average MMD between all pair combinations of each landmark in the normalised space. The errors are respectively 0.145, 0.132 and 0.033 in normalised units for a), b) and our method c), showing an improvement factor of 4.4 and 4.0. The same landmarks from different RT-GT patients may have distant shifts on the template for the methods without prior knowledge of the anatomy, see figure 5.5.

5.4.2 Generic Shape Modelling

The liver has an overall length of about 200 mm. We define a first GPMM with a unique Gaussian kernel of standard deviation 20 mm (GPMM-L), to cope with large local variations. We define a second GPMM combining multiple Gaussian kernels of standard deviations 20, 40, 80 and 160 mm (GPMM-M), to deal with several spatial ranges of morphological variations. We keep 200 components in both GPMM and 25 in SSM. This allows us to obtain equivalent global vertex reconstruction errors of about 2 mm on AMOS, as shown in table 5.1. Reconstruction errors are obtained using MAE. On RT-

Dataset	AMOS after 5.2.2		RT-GT			
Object	Global	Features	Global	Features	Landmarks	
Criterion	MAE	MAE	MAE	MAE	MAE	MMD
SSM	2.5 ± 0.3	$\mathbf{3.9 \pm 0.7}$	6.3 ± 0.9	9.0 ± 0.9	7.5 ± 0.9	9.8 ± 2.1
GPMM-L	2.6 ± 0.5	5.6 ± 1.4	3.8 ± 0.3	7.5 ± 0.9	4.1 ± 0.8	7.2 ± 1.3
GPMM-M	$\mathbf{2.0 \pm 0.3}$	4.3 ± 1.0	$\mathbf{3.0 \pm 0.2}$	$\mathbf{5.7 \pm 0.7}$	$\mathbf{3.1 \pm 0.8}$	$\mathbf{6.5 \pm 1.3}$

Table 5.1: Average and standard deviation ($\pm$) of the reconstruction errors (mm) of global vertices, feature points and landmarks for both datasets and models. The last column compares the reconstructed landmarks with the annotated ones.

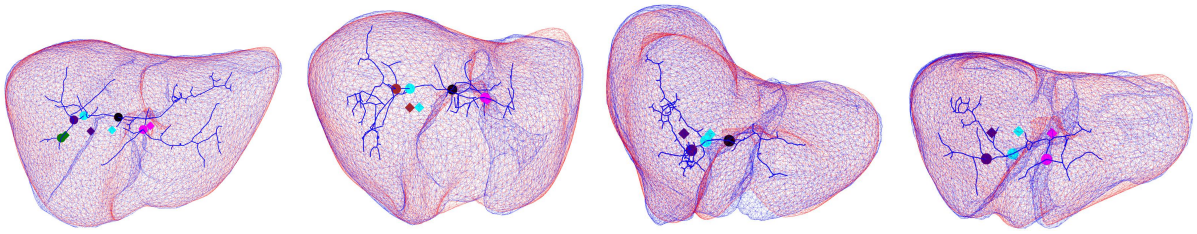


Figure 5.6: Inner portal vein bifurcation point registration results (diamonds) and targets (disks) on RT-GT, with a specific colour for each bifurcation. Registered and target livers are respectively in red and blue. The portal vein centreline is also in blue.

GT, the errors fall between 3 and 4 mm for GPMM models and reach 6.3 mm for SSM. SSM thus does not generalise as well as GPMM, indicating that the training liver shapes may underrepresent the population. Table 5.1 also indicates that the 14 correspondences have a higher MAE than both the general vertices and the landmarks, reaching at least 5.7 mm for the GPMM-M. This may owe to the uneven sampling density of the feature points around the liver, causing some local parts such as the posterior ones to be less well constrained and reconstructed. The last column of table 5.1 also indicates the registration errors after model reconstruction. It only reaches 6.5 mm for GPMM-M and almost 10 mm for SSM. In summary, GPMM-M obtains the best surface reconstruction and registration results.

We also compute internal reconstruction and registration errors, using corresponding bifurcation points of the main portal vein branches. AMOS only contains 15 scans where vessels are contrast-enhanced. We partially segment their portal vein and we extract their centreline, using VMTK. Between 2 and 5 corresponding bifurcation points are manually selected, depending on the amount of segmentation.

We use the transformation estimates from GPA on the corresponding points in order to obtain the mean internal points in template coordinates. We then compute the barycentric coordinates with respect to the PG volumetric model. This allows us to compute the reconstruction MAE of these internal points on AMOS, respectively $11.3 \pm 4.3, 10.2 \pm 3.9$ and 10.9 ± 4.2 mm for SSM, GPMM-L and GPMM-M models. They are all about 11 mm and higher than surface reconstruction errors, with a higher average standard deviation. Internal inter-subject anatomical differences are not explicitly registered and could thus be partially unrelated to the surface ones. Inner registration (section 5.3.2) errors of the template vessel bifurcation points are then computed on RT-GT. Table 5.2 shows

Patient	1	2	3	4	Average
SSM	25.3 ± 6.8	20.5 ± 11.5	$\mathbf{9.3 \pm 1.2}$	21.0 ± 1.5	19.0
GPMM-L	20.4 ± 6.0	22.2 ± 10.7	13.0 ± 2.7	$\mathbf{18.9 \pm 3.0}$	18.6
GPMM-M	$\mathbf{14.7 \pm 9.4}$	$\mathbf{12.3 \pm 6.6}$	11.9 ± 4.2	22.3 ± 1.4	**15.3**

Table 5.2: Average inner portal vein bifurcation registration MAE errors (mm) and standard deviation (±) for the RT-GT dataset for the different PG models.

Dataset	SSM		GPMM-L		GPMM-M	
Criteria	Reproj.	MAE	Reproj.	MAE	Reproj.	MAE
PG-LMR	3.8 ± 3.1	19.3 ± 10.1	8.0 ± 5.0	30.9 ± 9.9	3.9 ± 2.9	16.8 ± 8.9

Table 5.3: Average MAE (mm) and reprojection errors (% of image diagonal) and their standard deviations (±) for the simulated test datasets for the PG-LMR.

that GPMM-M obtains the best average results of 15.3 mm, between 11.9 and 22.3 mm for all patients, shown in figure 5.6. This is slightly larger than the reconstruction ones obtained for AMOS, probably due to limited representativeness of the 15 samples. SSM obtains the worse average results of 19 mm, which could be due to its coarse Delaunay tetrahedrisation. GPMM-M is thus selected as the PG model.

5.4.3 3D-2D Registration

LMR was trained for the PG model in 31 epochs. The registration results for all the simulated test datasets (section 5.3.1) are shown in table 5.3, namely the reprojection error, which is the mean of the mean closest distances of the projected landmark and silhouette vertices from corresponding targets, and the MAE between the predicted and simulated surface liver vertices. Average errors are higher for the GPMM-L set which, owing to very high synthetic local variations, is more difficult to fit. PG-LMR obtains an MAE between 16.8 and 19.3 mm and reprojection errors around 4% of the image diagonal on both SSM and GPMM-M sets.

On RT-GT, reprojection errors, evaluated using the CD2T criterion, are lower, see table 5.4, and also look rather low, as illustrated in figure 5.7. Regarding tumour position evaluation, TRE is computed. In table 5.4, we compare the results to state-of-the-art patient-specific methods [Adagolodjo *et al.* 2017, Koo *et al.* 2017a] including ours, presented in chapter 4. TREs for our method are marginally larger than for the PS-LMR, except for Patient 3 where it is 3 cm higher, while reprojection errors are lower on average. This discrepancy between TRE and reprojection errors suggests that there are registration ambiguities, i.e. several shapes could correspond to the 2D landmarks. Initialising the shape from the predicted patient scores $\hat{\beta}$ at inference could alleviate this issue. In a broader perspective (table 5.5), the PG-LMR shares the architecture of the PS-LMR and thus has the same advantages of very fast runtime in the operating theatre, unlike optimisation-based methods. However, it outperforms in deployability due to the single training for all patients. It also facilitates anatomical AR, illustrated in figure 5.7, as it only requires generic anatomical data related to the PG model. These assets would also ease surgeon education.

Patient	1		2		3		4		Average without P2	
Ann.	TRE	CD2T	TRE	CD2T	TRE	CD2T	TRE	CD2T	TRE	CD2T
Manual pose + [Adagolodjo *et al.* 2017]	8.3	NA	37.3	NA	28.4	NA	15.8	NA	$17.5^{\pm 10.2}$	NA
Manual pose+ [Koo *et al.* 2017a]	9.5	NA	39.0	NA	25.0	NA	18.4	NA	$17.6^{\pm 7.8}$	NA
VAPE+optimisation (FEM-OHA-G)	15.6	1.2	61.4	3.9	7.8	2.0	9.5	1.4	$\mathbf{11.0}^{\pm \mathbf{4.1}}$	$\mathbf{1.5}^{\pm \mathbf{0.4}}$
PS-LMR (FEM-OHA-G)	19.7	2.4	63.7	4.4	19.3	2.9	14.6	2.1	$17.9^{\pm 2.8}$	$2.5^{\pm 0.4}$
PG-LMR (GPMM-M)	22.8	1.2	60.1	2.6	49.4	1.9	14.6	2.4	$28.9^{\pm 18.2}$	$1.8^{\pm 0.6}$

Table 5.4: Average of the TRE (mm) and CD2T errors (% of image diagonal) for state-of-the-art methods and proposed PG-LMR for RT-GT, from manual annotations.

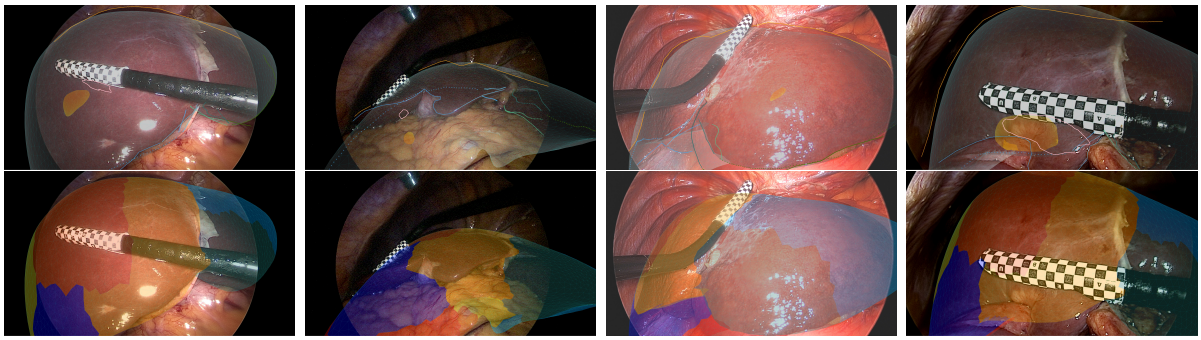


Figure 5.7: PS (top) and PG (bottom) AR results from PG-LMR on RT-GT samples, respectively showing the predicted (orange) and GT (pink) tumours and coarse Couinaud segments, manually defined on the PG template.

Automatic versus manual segmentation. We also evaluate the influence of the segmentation method on the PG-LMR results, see table 5.6. This is consistent with the results on the previous methods, see section 4.5.4.3. The highest tumour TRE is obtained for the best performing segmentation network Mask2Former-S, probably due to wrong class parts that it sometimes predicts. Other segmentation networks obtained higher tumour TRE with respect to manual segmentation, which could be due to the fact they do not generalise well to the non-standard RT-GT dataset, except for the STM which makes use of manually annotated masks associated to other images of the given procedure, see section 3.3.4.

5.5 Conclusion

We have proposed a novel generic liver shape registration and modelling method using prior anatomical knowledge, through surface feature points. It highly reduces global and local surface registration and reconstruction errors, reaching below 6 mm on both the AMOS and RT-GT datasets. It obtains higher internal vessel point reconstruction errors of about 11 mm on AMOS and registration ones on RT-GT reach about 15 mm. This suggests that internal inter-subject anatomical differences, which are not explicitly regis-

	Optimisation	Patient-specific LMR	Patient-generic LMR
Accuracy	++	++	+
Deployability	+	-	++
Runtime in operating theatre	-	++	++
Anatomical AR	-	-	++
Surgeon education	+	+	++

Table 5.5: General feature comparison between the different frameworks.

	TRE (mm)		CD2T reprojection error (% of the image diagonal length)	
Segmentation	Average	Average without P2	Average	Average without P2
Manual	$36.7^{\pm 21.5}$	$28.9^{\pm 18.2}$	$2.0^{\pm 0.6}$	$1.8^{\pm 0.6}$
Mask2Former-R	$41.5^{\pm 5.6}$	$\underline{39.9}^{\pm 5.7}$	$4.1^{\pm 2.0}$	$3.6^{\pm 2.2}$
Mask2Former-S	$46.1^{\pm 8.9}$	$47.4^{\pm 10.4}$	$3.9^{\pm 1.8}$	$\underline{3.1}^{\pm 1.1}$
COSNet	$45.8^{\pm 3.5}$	$45.9^{\pm 4.3}$	$4.1^{\pm 1.8}$	$3.3^{\pm 1.2}$
STM	$36.4^{\pm 16.4}$	$\mathbf{30.4}^{\pm \mathbf{13.8}}$	$2.3^{\pm 0.7}$	$\mathbf{2.0}^{\pm \mathbf{0.4}}$

Table 5.6: Comparison of segmentation method influence on PG-LMR results. In blue, the results obtained using manual segmentation. Reprojection error is computed on the automatic detected landmarks.

tered, could be partially unrelated to the surface ones. Therefore, the gap between surface and internal inter-subject anatomical differences should be thoroughly characterised. We plan to extend the number and area (e.g. using the hepatic vein) of internal correspondences. This would allow us to build generic Couinaud segments accordingly. We also plan to evaluate feature point annotation inter-operator variability.

We have also proposed the first PG neural 3D-2D registration framework for MILS, making use of the generic liver shape model. It facilitates both PS and PG image augmentation. Indeed, unlike previous work, our framework does not require per-patient retraining and is applicable without PS data. Its registration accuracy is slightly worse than PS methods. Possible directions for future work include the initialisation of the registration closer to the patient shape. In addition, transferring higher resolution feature maps to the regressor, as suggested in section 4.6, could also be attempted. It would also be important to study the potential of anatomical augmentation in surgical navigation and surgeon training. When numerous annotated mini-invasive data will be available, this framework will also enable one to feed the LMR directly with image data instead of segmentation masks, as the HMR.

Chapter 6

Conclusion

Contents

6.1 Synthesis

6.1.1 General Points

We have dealt with the problem of registering a 3D preoperative liver model with its inner structures onto a mini-invasive image, using 3D/2D anatomical anterior ridge and upper silhouette landmark correspondences, as well as optional falciform ligament ones. Due to these landmarks, less visible when the liver is manipulated for accessing posterior inner structures, this framework is more adapted to contexts requiring the localisation of anterior inner structures close to the upper liver surface. In addition, it is more adapted to exploratory mini-invasive global views of the liver, before the start of resection, where most of the landmarks are visible and not occluded by blood, tools, or gauze. This 3D/2D registration serves a purpose of assisting surgeons with AR, providing them the position of augmented inner liver structures of interest on the mini-invasive image, for guiding the resection. Two main resection approaches exist: anatomical, which is removing hepatic Couinaud segments defined by major hepatic blood vessels, and non-anatomical, which is only removing the liver parts enclosing tumours, with a margin. Hence, the inner structures of main interest for resection guidance consist of both tumours and major hepatic blood vessels.

We have based our work on a baseline registration pipeline, which processes a still image each time, enabling its application to every endoscopic surgical camera, i.e. both monocular and stereoscopic cameras. It is composed of several blocks and divided into two stages: preoperative and intraoperative ones. The preoperative stage comprises the

segmentation of the preoperative volume obtained from the CT-scan or MRI, in order to extract and process the meshes of the liver and its inner structures of interest, such as tumours and blood vessels. It also includes their deformation modelling. In addition, the landmarks are annotated on the model. The intraoperative stage requires a semi-automatic camera calibration, in order to obtain the intrinsic camera parameters, in addition to the annotation of the landmarks on the mini-invasive image. Once these preparation steps are completed, registration can be performed in two steps, rigid then deformable, respectively estimating pose and deformation parameters.

The objective of this thesis was the automation of the 3D/2D intraoperative registration process, in order to ease the AR guidance deployment in clinical practice. We have explored two ways. The first automates each intraoperative manual step of the baseline pipeline, i.e. the annotation of the landmarks on mini-invasive images and the pose estimation. The second one employs an alternative pipeline adapted to deep learning methods which transfers most of the computation cost in the intraoperative stage, i.e. for deep neural network training, while also making use of image landmark annotation.

6.1.2 Baseline Pipeline Automation

We have first redefined the anterior ridge landmark, which was previously modelled as a single curve, because its central part could correspond to different positions in the model according to the view, and its singleness could hinder the determination of 3D/2D correspondences. We have split it into several side parts (left, right) and central ones (upper and lower left and right). As a consequence, it allows a substantial performance improvement of an automatic pose estimation method relying on 3D/2D correspondences, implying that this splitting facilitates the determination of relevant 3D/2D correspondences.

Then, we have collected and annotated mini-invasive image datasets with our university hospital partners. This has allowed the training of several segmentation networks for automatically detecting and annotating landmarks from mini-invasive images. Convolution-based segmentation networks for independent image inputs were surpassed on three mini-invasive datasets by a fully attention-based network, the Mask2Former-S. It employs a transformer-based encoder, and two transformer-based decoders, following a mask classification formulation instead of an only pixel-based one, and uses a specific masked attention mechanism. However, segmentation performance from convolutional-based neural networks can be slightly improved when combining information from other image inputs or boosted with a large margin when using additional mask information associated to other images. The latter (STM) highly benefits from training using close images, and also benefits from inferring close content paired samples, such as sequentially close video frames. The segmentation networks generalise well to other typical datasets, but not to a real dataset dedicated to 3D/2D registration evaluation, due to the presence of non-standard black and white patterns on an ultrasound probe, except for the STM, thanks to information from other masks.

In order to automate 3D/2D rigid registration, we have proposed an automatic pose estimation method (VAPE) which makes use of RANSAC-based PnP for estimating the pose from 3D-2D correspondences. It proceeds in an iterative manner in 3 coarse-to-fine steps, refining the 3D-2D correspondences based on the visibility of the model landmarks from previous estimates. In addition, it uses multiple reprojection error thresholds in

RANSAC for estimating multiple poses, and selects the best one at each step according to a distance-based criterion. This approach is very competitive to manual pose initialisation by an expert as it obtains lower reprojection and registration errors on the real registration evaluation dataset, while taking on average a few seconds.

While the deformable registration was already automatic, we have accessed the influence of several deformation models on deformation parameter optimisation results. This first comparison suggests the utilisation of a deformation model based on finite element biomechanical simulations of nodal forces reaching high amplitudes, followed with a dimension reduction using global truncated SVD. However, the deformable registration has limited influence on the accuracy of the estimated inner structures position, compared to the pose estimation, and can only slightly improve it.

Combining automatic pose and deformation parameter estimation from manual landmark annotation leads to the lowest registration errors (about 11 mm) on the real evaluation dataset. Combining all automatic steps, including image segmentation, results in much larger errors, except for the STM. However, this result should be considered cautiously because of this non-typical image domain dataset for which the trained segmentation networks do not generalise well.

6.1.3 Automatic Learning-based Pipeline

We have built a bridge between the problems of 3D/2D registration in MILS and human body shape reconstruction from a natural human-centred image. This has allowed us to adapt a mesh recovery network with an encoder-regressor architecture to the liver framework, the LMR. The network inputs distance transforms of landmark segmentation masks and iteratively regresses both pose and deformation (or shape) parameters, respectively initialised to default typical parameters and zeros. The regressor also inputs features from the encoder. Training is performed on a large dataset from the simulation of associated inputs and outputs, restricted to plausible ranges. We have explored two approaches:

- Patient-specific. This approach uses a patient-specific deformation model and the LMR is thus trained from preoperative specific simulations, which takes about a day for each patient. We also choose to employ the deformation model based on finite element biomechanical simulations of nodal forces reaching high amplitudes, followed with a dimension reduction using global truncated SVD. On the real registration evaluation dataset, it obtains the lowest, although still high, reprojection error among the other tested deformation models, and a registration error of about 18 mm on par with a manually estimated pose followed with deformable methods from previous works. Its main asset is its very fast intraoperative runtime (about 3 ms), allowing quasi-real-time intraoperative registration when combined to automatic segmentation.

- Patient-generic. This approach uses a patient-generic liver shape model. We propose generic liver shape registration and modelling methods using prior anatomical knowledge, through surface feature points. Registration between a reference shape and others, using RBF and two rounds of NR-ICP, is guided by these surface feature point correspondences. This results in a set of shapes in one-to-one correspondences.

They are then aligned using GPA, and the mean surface shape is extracted. The mean volumetric shape is built from the surface one and shares its vertices. The generic modelling uses combinations of multi-scaled Gaussian processes as shape components from the mean volumetric shape (GPMM), and can also be applied to the surface. Using prior anatomical knowledge highly reduces global and local surface registration and reconstruction errors, reaching both below 6 mm on several datasets. However, internal vessel point reconstruction and registration errors are not as low.

Unlike the patient-specific LMR, the patient-generic one does not require per-patient retraining and is applicable without patient-specific data. It facilitates image augmentation of anatomical information defined on the patient-generic volumetric model. It also facilitates patient-specific registration, using a specific preoperative block for transferring inner structures of the patient liver to the generic template. When applied on the registration validation real dataset, its registration accuracy is slightly worse than patient-specific methods, while its reprojection error is low.

Both approaches obtain degraded registration results on the real evaluation dataset when using automatically segmented landmarks, except for the STM, as for the method combining pose estimation and deformation optimisation. The same caution should apply due this non-typical image domain dataset.

6.2 Discussion and Future Work

As this work deals with many subjects and fields with their own challenges, we suggest some points which seem either critical or promising to us to be studied and processed further.

6.2.1 Deformation Modelling

Very realistically modelling the deformation of a liver between the preoperative and intraoperative times is critical; in particular, for the artificial pneumoperitoneum as it is the principal loading constraint in exploratory mini-invasive conditions. Up to now, its modelling has been attempted using biomechanical simulations of the whole intra-abdominal space, requiring numerous structure segmentations and biomechanical parameter assumptions, which limit its feasibility. Instead of simulations, modelling could be performed from 3D corresponding data, before and after artificial pneumoperitoneum in mini-invasive conditions. They could be obtained from the artificial pneumoperitoneum CT technique [Wang *et al.* 2021], which uses a hybrid room with both a CT scanner and the equipment for insufflating gas through a trocar port, after a mini-invasive incision of the patient. However, this requires access to such very specific equipment or data. Note that such data with contrast-enhanced vessels would also facilitate generic liver surface and volumetric shape modelling as well as the determination of generic Couinaud segments.

6.2.2 Clinical 3D/2D Registration Evaluation Datasets

The limited number of available clinical evaluation data is an obstacle for fully validating the 3D-2D registration methods. The method from [Rabbani *et al.* 2022], calibrating the ultrasound probe in addition to the camera, and using a sticker with patterns for retrieving the ultrasound probe in the camera space, allows such an evaluation. However, it suffers from the need of calibrating the probe again at each procedure, as the sticker detaches during the sterilisation process, and thus cannot be performed at a large scale. Marking a pattern directly on an ultrasound probe would be a solution in order to perform the probe calibration only once, but would need the assistance of manufacturers or other partners while satisfying ethics, regulatory and quality checks. Although, the evaluation would still be restricted to local inner areas, and not the whole liver deformation. In addition, the problem of validating a neural network method trained on a typical mini-invasive image domain onto a non-typical one due to the evaluation technique would subsist. Hence, other evaluation modes, using other specific equipments, such as stereoscopic endoscopes, ICG FIS or hybrid rooms, should be explored in parallel.

6.2.3 Patient-Generic Liver Mesh Recovery

Currently, due to the limited number of available annotated mini-invasive data, the patient-generic LMR pipeline is not end-to-end and is split into image segmentation and then mesh recovery networks. When a sufficient amount of annotated data would allow it, the pipeline could be transformed to a end-to-end one directly fed by images, although trained with a reprojection loss, as HMR. It could combine multiple tasks, such as image segmentation as well as shape and pose regression, and benefit from advances in both domains. For instance, it could be built as a fully attention-based network, for both encoder and decoders, while using information from multiple resolution encoder layers in order to give a more direct feedback to the regressor with more localised information for parameter correction, such as PyMAF [Zhang *et al.* 2021].

6.2.4 3D/2D Corresponding Landmarks

As our 3D-2D registration methodology is based on 3D-2D correspondences, the relevance and quality of annotations of both 3D and 2D landmarks is fundamental:

- The falciform ligament, when not visible in the preoperative images, could thus be disregarded as it can bring unreliable correspondences and degrade registration accuracy.
- The work on automatic annotation of the landmarks on mini-invasive images should be pursued. While network accuracy would improve with more annotated and training data from additional procedures, studying the effect of combining information, from other images and masks, for example, with a fully attention-based network, would be relevant. In addition, investigating post-processing and semi-automatic correction tools [Zhou *et al.* 2023, Mikhailov *et al.* 2024] could be fruitful for AR guidance solution deployment.

- Due to the accuracy of the generic shape registration method, it could be used in order to transfer annotated 3D anatomical landmarks from the template shape to any new liver shape, after manual annotation of the corresponding surface feature points, in an atlas-based annotation fashion. In the future, attempting to fully automatically annotate these surface feature points on the 3D liver models could further simplify the pipeline.

- Anatomical surface feature points which are visible on mini-invasive images, e.g. ones related to the anterior ridge, could also serve as additional 3D/2D corresponding point landmarks, for both patient-specific and patient-generic approaches. They would further constrain the problem, and reduce shifts due to correspondence ambiguities between 3D and 2D landmark curves.

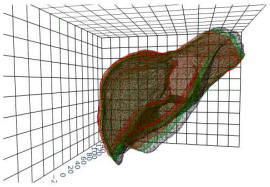
6.2.5 Combining Information

In the problem at hand, there can be ambiguities between pose and deformation due to limited data constraints. For mitigating them, using other information from other images and even segmentation masks could be envisaged. Combining the mini-invasive image information to other information registered from other imaging system would also be relevant. In particular, from IOUS, which constitutes the standard intraoperative navigation procedure and is thus systematically performed. For instance, our registration method could additionally take into account inner liver cues registered from laparoscopic ultrasound [Kalantari *et al.* 2024].

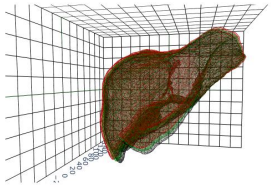
Appendices

Appendix A

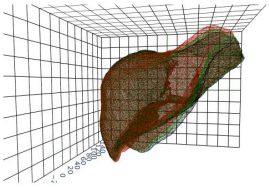
Nomograms of the Deformation Models



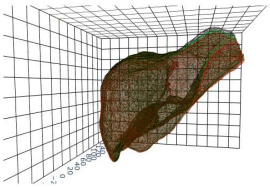
(a) 1st, max=9.6



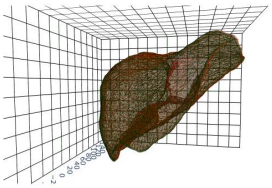
(b) 2nd, max=6.9



(c) 3rd, max=4.6

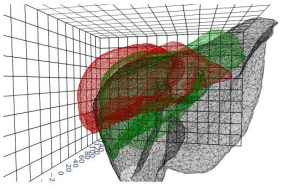


(d) 4th, max=3.6

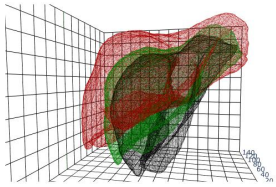


(e) 5th, max=2.0

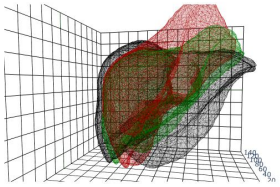
Figure A.1: Nomograms of the deformation components for OLA with global truncated SVD in Patient 1. In green, the initial mesh, and in red and black the deformed meshes within the bounds of the deformation ranges. Captions include the maximal vertex displacement (mm) from the rest configuration.



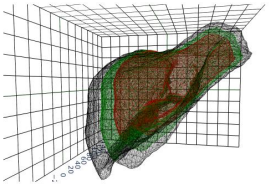
(a) 1st, max=75.4



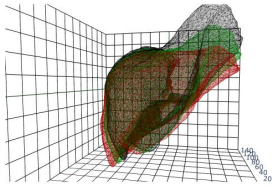
(b) 2nd, max=54.5



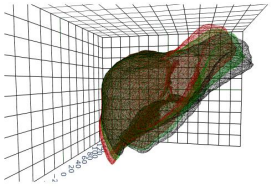
(c) 3rd, max=35.3



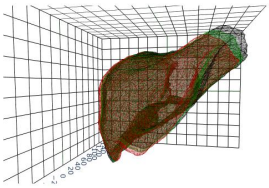
(d) 4th, max=26.0



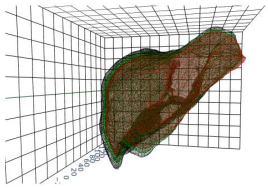
(e) 5th, max=20.9



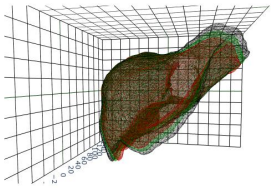
(f) 6th, max=13.0



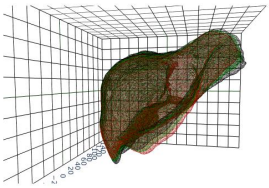
(g) 7th, max=10.6



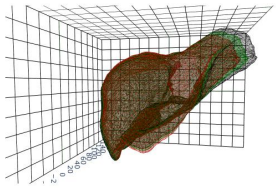
(h) 8th, max=10.0



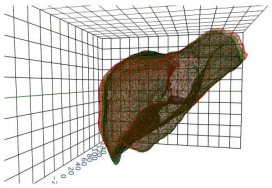
(i) 9th, max=9.3



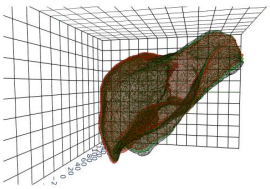
(j) 10th, max=7.5



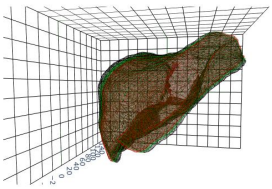
(k) 11th, max=8.8



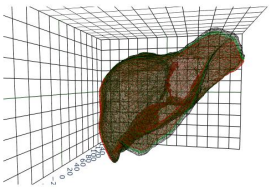
(l) 12th, max=6.0



(m) 13th, max=5.7

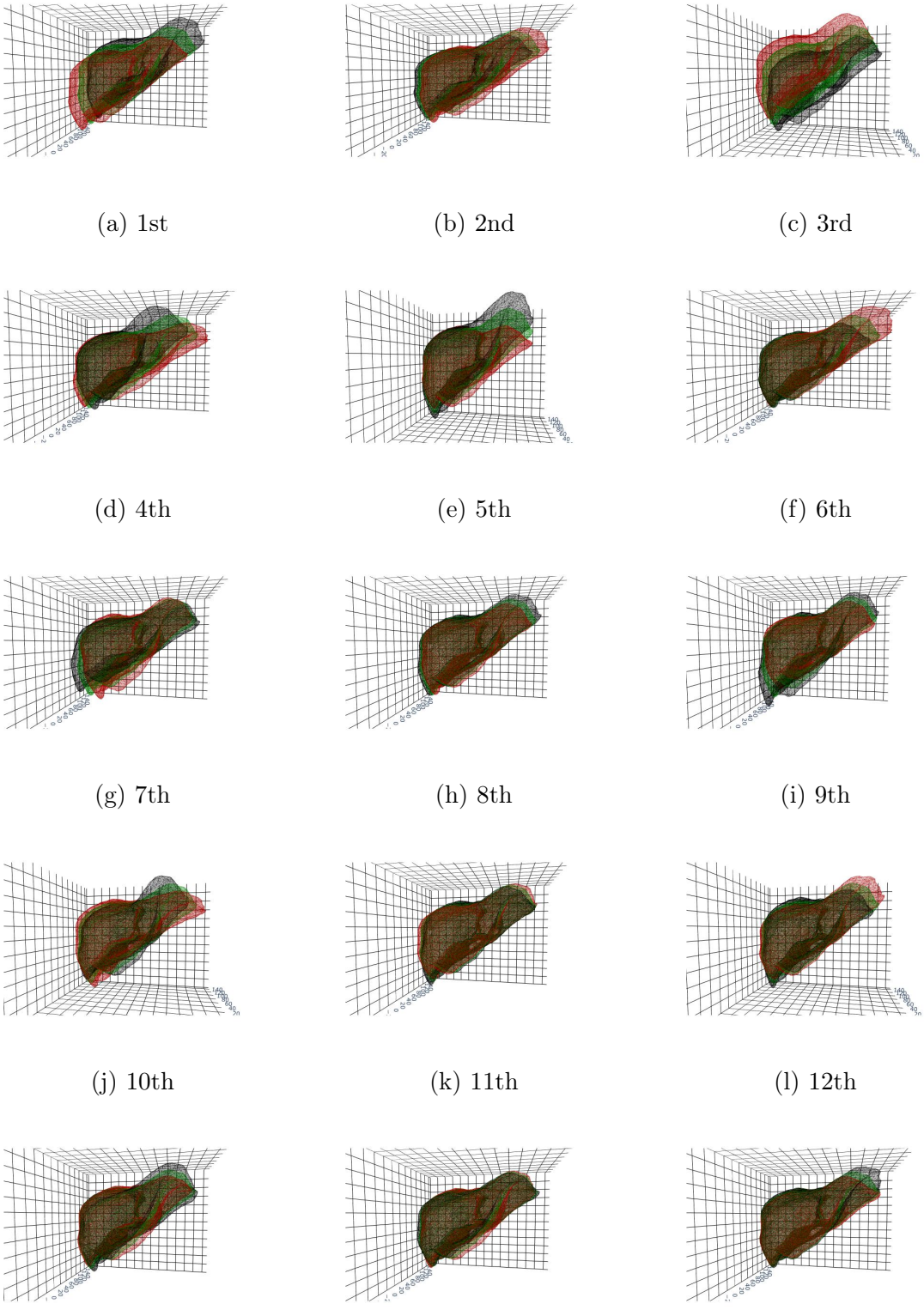


(n) 14th, max=4.7



(o) 15th, max=4.3

Figure A.2: Nomograms of the first 15 deformation components (out of 20) for OHA with global truncated SVD in Patient 1. In green, the initial mesh, and in red and black the deformed meshes within bounds of the deformation ranges. The captions include the maximal vertex displacement (mm) from the rest configuration.



(a) 1st (b) 2nd (c) 3rd

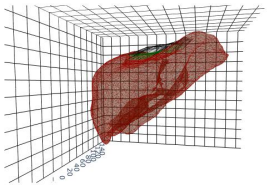
(d) 4th (e) 5th (f) 6th

(g) 7th (h) 8th (i) 9th

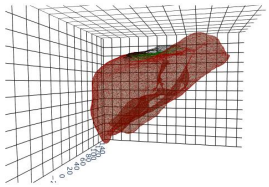
(j) 10th (k) 11th (l) 12th

(m) 13th (n) 14th (o) 15th

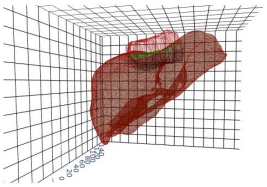
Figure A.3: Nomograms of the first 15 deformation components (out of 29) for FFD with global truncated SVD in Patient 1. In green, the initial mesh, and in red and black the deformed meshes within bounds of the deformation ranges.



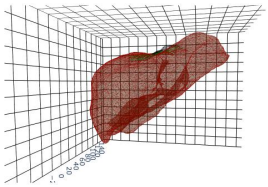
(a) 1st - One cluster



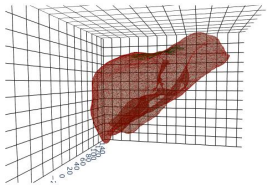
(b) 2nd - One cluster



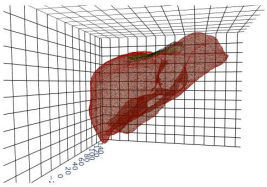
(c) 3rd - One cluster



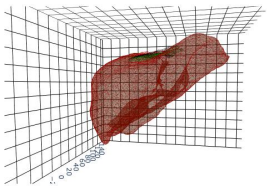
(d) 4th - One cluster



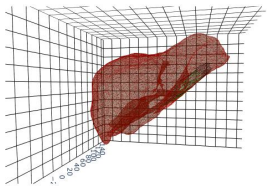
(e) 5th - One cluster



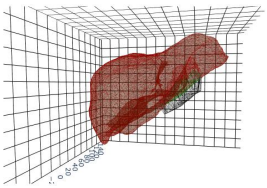
(f) 6th - One cluster



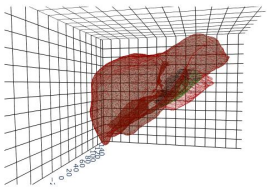
(g) 7th - One cluster



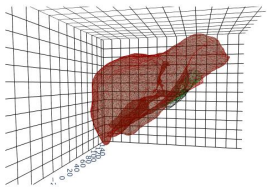
(h) 1st - Another cluster



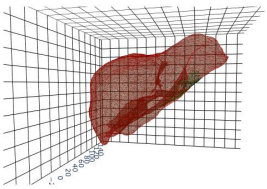
(i) 2nd - Another cluster



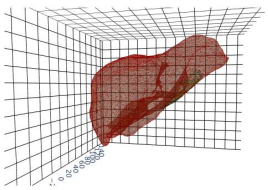
(j) 3rd - Another cluster



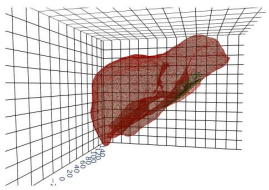
(k) 4th - Another cluster



(l) 5th - Another cluster

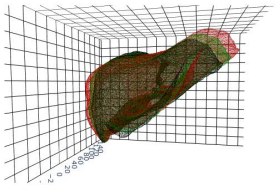


(m) 6th - Another cluster

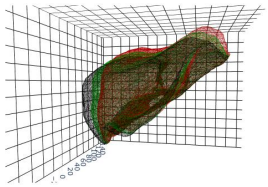


(n) 7th - Another cluster

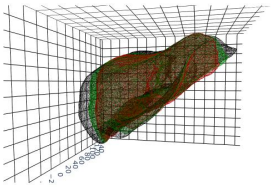
Figure A.4: Nomograms of some deformation components (out of 210) for FFD with local truncated SVD in Patient 1. In green, the initial mesh, and in red and black the deformed meshes within bounds of the deformation ranges.



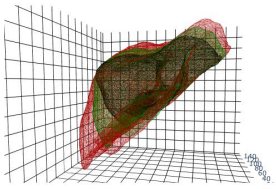
(a) 1st



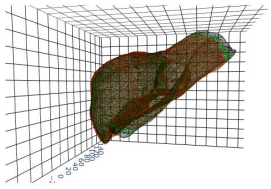
(b) 2nd



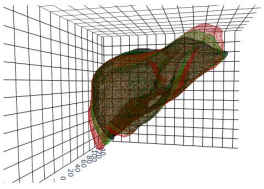
(c) 3rd



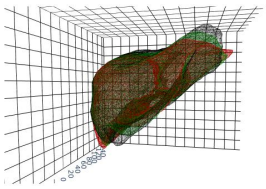
(d) 4th



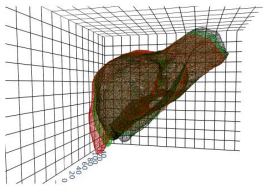
(e) 5th



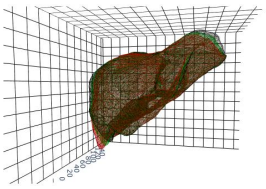
(f) 6th



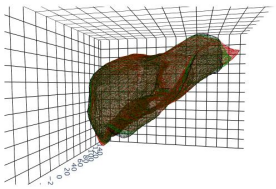
(g) 7th



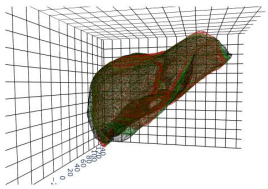
(h) 8th



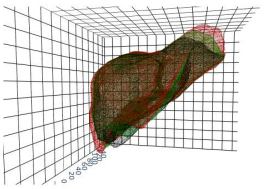
(i) 9th



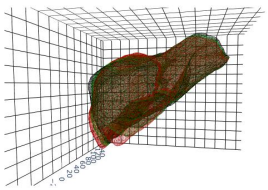
(j) 10th



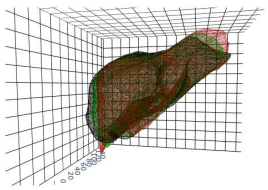
(k) 11th



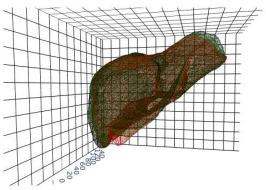
(l) 12th



(m) 13th



(n) 14th



(o) 15th

Figure A.5: Nomograms of the first 15 deformation components (out of 200) for LLE in Patient 1. In green, the initial mesh, and in red and black the deformed meshes within bounds of the deformation ranges.

Bibliography

[Abdulkareem *et al.* 2023] Nashwan Karkhi Abdulkareem, Shereen Ismail Hajee, Fatiheea Fatihalla Hassan, Ilham Khalid Ibrahim, Ruaa Emad Hussein Al-Khalidi and Noor Abubaker Abdulqader. *Investigating the slice thickness effect on noise and diagnostic content of single-source multi-slice computerized axial tomography.* Journal of Medicine and Life, vol. 16, no. 6, page 862, 2023. (Cited on page 24.)

[Adagolodjo *et al.* 2017] Yinoussa Adagolodjo, Raffaella Trivisonne, Nazim Haouchine, Stéphane Cotin and Hadrien Courtecuisse. *Silhouette-based pose estimation for deformable organs application to surgical augmented reality.* In International Conference on Intelligent Robots and Systems, 2017. (Cited on pages 112, 113, 126, 127, 143 and 144.)

[Adams 2022] Reid B Adams. *Ultrasound scanning techniques.* Surgery Open Science, vol. 10, pages 182–207, 2022. License: CC BY-NC-ND. (Cited on pages 27 and 34.)

[Affane 2022] Abir Affane. *Robust liver vessel segmentation in medical images using 3-D deep learning approaches.* PhD thesis, Université Clermont Auvergne, 2022. (Cited on page 43.)

[Al Mamun & Kadir 2020] Md Afif Al Mamun and Imamul Kadir. *An-Eye: safe navigation in footpath for visually impaired using computer vision techniques.* PhD thesis, Daffodil International University, 06 2020. (Cited on page 84.)

[Amberg *et al.* 2007] Brian Amberg, Sami Romdhani and Thomas Vetter. *Optimal step nonrigid ICP algorithms for surface registration.* In IEEE conference on Computer Vision and Pattern Recognition, pages 1–8. IEEE, 2007. (Cited on pages 135 and 137.)

[Armijo 1966] Larry Armijo. *Minimization of functions having Lipschitz continuous first partial derivatives.* Pacific Journal of mathematics, vol. 16, no. 1, pages 1–3, 1966. (Cited on page 68.)

[Attene 2010] Marco Attene. *A lightweight approach to repairing digitized polygon meshes.* The visual computer, vol. 26, pages 1393–1406, 2010. (Cited on page 47.)

[Azuma 1997] Ronald T Azuma. *A survey of augmented reality.* Presence: teleoperators & virtual environments, vol. 6, no. 4, pages 355–385, 1997. (Cited on page 35.)

[Ban *et al.* 2021] Daisuke Ban, Satoshi Nara, Takeshi Takamoto, Takahiro Mizui, Jun Yoshino, Minoru Esaki and Kazuaki Shimada. *Revisiting the role of the hepatic vein in laparoscopic liver resection.* Hepatoma Res, vol. 7, page 13, 2021. (Cited on page 32.)

[Bano *et al.* 2012] Jordan Bano, Alexandre Hostettler, SA Nicolau, Christophe Doignon, HS Wu, MH Huang, Luc Soler and Jacques Marescaux. *Simulation of the abdominal wall and its arteries after pneumoperitoneum for guidance of port positioning in laparoscopic surgery.* In Advances in Visual Computing, pages 1–11. Springer, 2012. (Cited on page 56.)

[Barequet & Sharir 1995] Gill Barequet and Micha Sharir. *Filling gaps in the boundary of a polyhedron.* Computer Aided Geometric Design, vol. 12, no. 2, pages 207–229, 1995. (Cited on page 48.)

[Bartels 2015] Pieterjan Bartels. *Position based dynamics.* PhD thesis, MS thesis, Dept. Des. Eng., Bournemouth Univ., Poole, England, 2015. (Cited on page 115.)

[Basheer & Hajmeer 2000] Imad A Basheer and Maha Hajmeer. *Artificial neural networks: fundamentals, computing, design, and application.* Journal of microbiological methods, vol. 43, no. 1, pages 3–31, 2000. (Cited on page 76.)

[Bender *et al.* 2014] Jan Bender, Dan Koschier, Patrick Charrier and Daniel Weber. *Position-based simulation of continuous materials.* Computers & Graphics, vol. 44, pages 1–10, 2014. (Cited on page 113.)

[Berger 2002] Abi Berger. *How does it work?: Magnetic resonance imaging.* BMJ: British Medical Journal, vol. 324, no. 7328, page 35, 2002. (Cited on pages 24 and 26.)

[Bhogal *et al.* 2019] Ricky Harminder Bhogal, Stephanos Pericleous, Aamir Z Khan, G Tsoulfas and L Rodrigo. *Robotic liver surgery.* In Liver Disease and Surgery. IntechOpen, 2019. License: CC BY. (Cited on pages 31 and 32.)

[Bilic *et al.* 2023] Patrick Bilic, Patrick Christ, Hongwei Bran Li, Eugene Vorontsov, Avi Ben-Cohen, Georgios Kaissis, Adi Szeskin, Colin Jacobs, Gabriel Efrain Humpire Mamani, Gabriel Chartrand*et al. The liver tumor segmentation benchmark (lits).* Medical Image Analysis, vol. 84, page 102680, 2023. (Cited on pages 43 and 101.)

[Blankevoort *et al.* n.d.] Bas Blankevoort, LUMC and O. Paul Gobée. *Leiden - Drawing Liver segments and vascularisation - English labels*, n.d., https://anatomytool.org/content/leiden-drawing-liver-segments-and-vascularisation-english-labels. Accessed 01 May 2024. License: CC BY-NC-SA. (Cited on page 21.)

[Blausen.com staff 2014] Blausen.com staff. *Medical gallery of Blausen Medical 2014.* WikiJournal of Medicine 1 (2), 2014, doi: 10.15347/wjm/2014.010, issn: 2002-4436. (Cited on page 21.)

[Bonet & Wood 1997] Javier Bonet and Richard D Wood. Nonlinear continuum mechanics for finite element analysis. Cambridge university press, 1997. (Cited on page 54.)

[Botea *et al.* 2022] Florin Botea, Alexandru Bârcu, Alin Kraft, Irinel Popescu and Michael Linecker. *Parenchyma-sparing liver resection or regenerative liver surgery: which way to go?* Medicina, vol. 58, no. 10, page 1422, 2022. (Cited on page 31.)

[Branch 1982] Robert A Branch. *Drugs as indicators of hepatic function.* Hepatology, vol. 2, no. 1, pages 97S–105S, 1982. (Cited on page 33.)

[Bray *et al.* 2024] Freddie Bray, Mathieu Laversanne, Hyuna Sung, Jacques Ferlay, Rebecca L. Siegel, Isabelle Soerjomataram and Ahmedin Jemal. *Global cancer statistics 2022: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries.* CA: A Cancer Journal for Clinicians, 2024, doi: https://doi.org/10.3322/caac.21834. (Cited on page 22.)

[Buhmann 2000] Martin Dietrich Buhmann. *Radial basis functions.* Acta numerica, vol. 9, pages 1–38, 2000. (Cited on page 136.)

[Burger 2016] Wilhelm Burger. *Zhang's camera calibration algorithm: in-depth tutorial and implementation.* HGB16-05, pages 1–6, 2016. (Cited on page 66.)

[Carion *et al.* 2020] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov and Sergey Zagoruyko. *End-to-end object detection with transformers.* In European Conference on Computer Vision, pages 213–229. Springer, 2020. (Cited on pages 92 and 93.)

[Carreira *et al.* 2016] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki and Jitendra Malik. *Human pose estimation with iterative error feedback.* In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 4733–4742, 2016. (Cited on page 116.)

[Cheng *et al.* 2021a] Bowen Cheng, Alex Schwing and Alexander Kirillov. *Per-pixel classification is not all you need for semantic segmentation.* Advances in Neural Information Processing Systems, vol. 34, pages 17864–17875, 2021. (Cited on pages 93 and 94.)

[Cheng *et al.* 2021b] Ho Kei Cheng, Yu-Wing Tai and Chi-Keung Tang. *Rethinking space-time networks with improved memory coverage for efficient video object segmentation.* Advances in Neural Information Processing Systems, vol. 34, pages 11781–11794, 2021. (Cited on page 99.)

[Cheng *et al.* 2022] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov and Rohit Girdhar. *Masked-attention mask transformer for universal image segmentation.* In Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, pages 1290–1299, 2022. (Cited on pages 93, 94, 95 and 101.)

[Chow & Chok 2019] Felix Che-Lok Chow and Kenneth Siu-Ho Chok. *Colorectal liver metastases: An update on multidisciplinary approach.* World journal of hepatology, vol. 11, no. 2, page 150, 2019. (Cited on page 23.)

[Chu *et al.* 2023] Kai-Jian Chu, Yoshikuni Kawaguchi and Kiyoshi Hasegawa. *Current use of intraoperative ultrasound in modern liver surgery.* Oncology and Translational Medicine, vol. 9, no. 4, pages 168–175, 2023. (Cited on page 33.)

[Cirne & Pedrini 2013] Marcos Vinicius Mussel Cirne and Hélio Pedrini. *Marching cubes technique for volumetric visualization accelerated with graphics processing units*. Journal of the Brazilian Computer Society, vol. 19, pages 223–233, 2013. (Cited on page 45.)

[Collins *et al.* 2020] Toby Collins, Daniel Pizarro, Simone Gasparini, Nicolas Bourdel, Pauline Chauvet, Michel Canis, Lilian Calvet and Adrien Bartoli. *Augmented reality guided laparoscopic surgery of the uterus*. IEEE Transactions on Medical Imaging, vol. 40, no. 1, pages 371–380, 2020. (Cited on page 115.)

[Commowick 2007] Olivier Commowick. *Création et utilisation d'atlas anatomiques numériques pour la radiothérapie*. PhD thesis, Université Nice Sophia Antipolis, 2007. (Cited on page 11.)

[Cootes *et al.* 1992] Timothy F Cootes, Christopher J Taylor, David H Cooper and Jim Graham. *Training models of shape from sets of examples*. In Proceedings of the British Machine Vision Conference, pages 9–18. Springer, 1992. (Cited on pages 82 and 135.)

[Cootes *et al.* 1995] Timothy F Cootes, Christopher J Taylor, David H Cooper and Jim Graham. *Active shape models-their training and application*. Computer vision and image understanding, vol. 61, no. 1, pages 38–59, 1995. (Cited on page 82.)

[Cootes *et al.* 1998] Timothy F Cootes, Gareth J Edwards and Christopher J Taylor. *Active appearance models*. In European Conference on Computer Vision, pages 484–498. Springer, 1998. (Cited on page 82.)

[Couinaud 1957] Claude Couinaud. Le foie: études anatomiques et chirurgicales. Masson, 1957. (Cited on page 22.)

[Cristinacce & Cootes 2006] David Cristinacce and Timothy F Cootes. *Feature detection and tracking with constrained local models*. In Proceedings of the British Machine Vision Conference, pages 929–938. The Bristish Machine Vision Association Press, 2006. (Cited on page 82.)

[Cui *et al.* 2023] Zejian Cui, João Cartucho, Stamatia Giannarou and Ferdinando Rodriguez y Baena. *Caveats on the first-generation da Vinci research kit: Latent technical constraints and essential calibrations*. IEEE Robotics & Automation Magazine, 2023. (Cited on pages 64 and 79.)

[Cunningham & Judy 2000] Ian A Cunningham and Philip F Judy. *Computed tomography*. The biomedical engineering handbook, vol. 1, pages 62–61, 2000. (Cited on page 24.)

[Damm *et al.* 2013] Georg Damm, Elisa Pfeiffer, Britta Burkhardt, Jan Vermehren, Andreas K Nüssler and Thomas S Weiss. *Human parenchymal and non-parenchymal liver cell isolation, culture and characterization*. Hepatology international, vol. 7, pages 951–958, 2013. (Cited on page 22.)

[De Brauer *et al.* 2024] Camille De Brauer, Philippe-Jean Bousquet and Lionel Lafay. *Cancers: incidence and survival in metropolitan France*. La Revue du Praticien, vol. 74, no. 1, pages 30–35, 2024. (Cited on page 22.)

[Deng *et al.* 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li and Li Fei-Fei. *Imagenet: A large-scale hierarchical image database*. In IEEE conference on Computer Vision and Pattern Recognition, pages 248–255. IEEE, 2009. (Cited on pages 96 and 99.)

[Desbrun *et al.* 1999] Mathieu Desbrun, Mark Meyer, Peter Schroder and Alan H Barr. *Implicit fairing of irregular meshes using diffusion and curvature flow*. In Proceedings of the Conference on Computer Graphics and Interactive Techniques, page 317–324. 1999. (Cited on page 46.)

[Desgranges *et al.* 2004] P Desgranges, A Bourriez, I Javerliat, O Van Laere, F Losy, A Lobontiu, D Melliere and JP Becquemin. *Robotically assisted aorto-femoral bypass grafting: lessons learned from our initial experience*. European journal of vascular and endovascular surgery, vol. 27, no. 5, pages 507–511, 2004. (Cited on page 30.)

[Dogeas *et al.* 2021] Epameinondas Dogeas, Samer Tohme and David A Geller. *Laparoscopic liver resection: global diffusion and learning curve*. Annals of the Academy of Medicine, Singapore, vol. 50, no. 10, pages 736–738, 2021. License: CC BY-NC-SA. (Cited on page 30.)

[Dowrick *et al.* 2023] Thomas Dowrick, Guofang Xiao, Daniil Nikitichev, Eren Dursun, Niels van Berkel, Moustafa Allam, Bongjin Koo, Joao Ramalhinho, Stephen Thompson, Kurinchi Gurusamy, Ann Blandford, Danail Stoyanov, Brian R Davidson and Matthew J Clarkson. *Evaluation of a calibration rig for stereo laparoscopes*. Medical Physics, vol. 50, no. 5, pages 2695–2704, 2023. (Cited on page 65.)

[Espinel *et al.* 2020] Yamid Espinel, Erol Özgür, Lilian Calvet, Bertrand Le Roy, Emmanuel Buc and Adrien Bartoli. *Combining visual cues with interactions for 3D–2D registration in liver laparoscopy*. Annals of Biomedical Engineering, vol. 48, no. 6, pages 1712–1727, 2020. (Cited on pages 37, 71 and 112.)

[Espinel *et al.* 2021] Yamid Espinel, Lilian Calvet, Karim Botros, Emmanuel Buc, Christophe Tilmant and Adrien Bartoli. *Using multiple images and contours for deformable 3D-2D registration of a preoperative ct in laparoscopic liver surgery*. In Medical Image Computing and Computer Assisted Intervention, pages 657–666. Springer, 2021. (Cited on page 123.)

[Espinel *et al.* 2024] Yamid Espinel, Navid Rabbani, Thien Bao Bui, Mathieu Ribeiro, Emmanuel Buc and Adrien Bartoli. *Keyhole-aware laparoscopic augmented reality*. Medical Image Analysis, vol. 94, page 103161, 2024. (Cited on page 36.)

[Falkenberg *et al.* 2022] Mårten Falkenberg, Magnus Rizell, Malin Sternby Eilard, Alois Regensburger, Roya Razazzian and Niclas Kvarnström. *Radiopaque Fiducials Guiding Laparoscopic Resection of Liver Tumors*. Surgical Laparoscopy Endoscopy

& Percutaneous Techniques, vol. 32, no. 1, pages 140–144, 2022. (Cited on page 34.)

[Felli *et al.* 2020] Eric Felli, Takeshi Urade, Mahdi Al-Taher, Emanuele Felli, Manuel Barberio, Laurent Goffin, Giuseppe M Ettorre, Jacques Marescaux, Patrick Pessaux, Lee Swanstrom and Michele Diana. *Demarcation line assessment in anatomical liver resection: an overview.* Surgical Innovation, vol. 27, no. 5, pages 424–430, 2020. (Cited on page 32.)

[Fischler & Bolles 1981] Martin A Fischler and Robert C Bolles. *Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography.* Communications of the ACM, vol. 24, no. 6, pages 381–395, 1981. (Cited on page 112.)

[Flament *et al.* 1982] JB Flament, JF Delattre and G Hidden. *The mechanisms responsible for stabilising the liver.* Anatomia Clinica, vol. 4, no. 2, pages 125–135, 1982. (Cited on page 56.)

[Fodor *et al.* 2018] Margot Fodor, Florian Primavesi, Eva Braunwarth, Benno Cardini, Thomas Resch, Reto Bale, Daniel Putzer, Benjamin Henninger, Rupert Oberhuber, Manuel Maglione, Christian Margreiter, Stefan Schneeberger, Dietmar Öfner and Stefan Stättner. *Indications for liver surgery in benign tumours.* European Surgery, vol. 50, pages 125–131, 2018. (Cited on page 23.)

[Fong & Saunders 2011] David Chin-Lung Fong and Michael Saunders. *LSMR: An iterative algorithm for sparse least-squares problems.* SIAM Journal on Scientific Computing, vol. 33, no. 5, pages 2950–2971, 2011. (Cited on page 124.)

[François *et al.* 2020] Tom François, Lilian Calvet, Sabrina Madad Zadeh, 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, vol. 15, pages 1177–1186, 2020. (Cited on page 97.)

[Fu *et al.* 2019] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang and Hanqing Lu. *Dual attention network for scene segmentation.* In Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, pages 3146–3154, 2019. Copyright: ©2019, IEEE. (Cited on pages 88 and 89.)

[Fuentes-Jimenez *et al.* 2022] David Fuentes-Jimenez, Daniel Pizarro, David Casillas-Pérez, Toby Collins and Adrien Bartoli. *Deep Shape-from-Template: Single-image quasi-isometric deformable registration and reconstruction.* Image and Vision Computing, vol. 127, page 104531, 2022. (Cited on page 123.)

[Gao *et al.* 2003] Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang and Hang-Fei Cheng. *Complete solution classification for the perspective-three-point problem.* IEEE transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 8, pages 930–943, 2003. (Cited on pages 67 and 112.)

[Gao *et al.* 2019] Jun Gao, Chengcheng Tang, Vignesh Ganapathi-Subramanian, Jiahui Huang, Hao Su and Leonidas J Guibas. *Deepspline: Data-driven reconstruction of parametric curves and surfaces.* arXiv preprint arXiv:1901.03781, 2019. (Cited on page 131.)

[Gao 2016] Bin Gao. *Basic liver immunology.* Cellular & molecular immunology, vol. 13, no. 3, pages 265–266, 2016. (Cited on page 22.)

[Garland & Heckbert 1997] Michael Garland and Paul S Heckbert. *Surface simplification using quadric error metrics.* In Proceedings of the conference on Computer Graphics and Interactive Techniques, pages 209–216, 1997. (Cited on page 47.)

[Gaujoux & Goéré 2011] S Gaujoux and D Goéré. *Surgical approach for hepatectomy.* Journal of visceral surgery, vol. 148, no. 6, pages e422–e426, 2011. (Cited on pages 28 and 30.)

[Gibson 1998] Sarah FF Gibson. *Constrained elastic surface nets: Generating smooth surfaces from binary segmented data.* In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 888–898. Springer, 1998. (Cited on page 46.)

[Gigot *et al.* 2004] Jean-François Gigot, Catherine Hubert, Radu Banice and Michael L Kendrick. *Laparoscopic management of benign liver diseases: where are we?* HPB, vol. 6, no. 4, pages 197–212, 2004. (Cited on page 23.)

[Giorgio *et al.* 2018] Rossi Giorgio, Tarasconi Antonio, Baiocchi Gianluca, Luigi de'Angelis Gian, Gaiani Federica, Catena Fausto and Raffaele Dalla Valle. *Fluorescence guided surgery in liver tumors: applications and advantages.* Acta Bio Medica: Atenei Parmensis, vol. 89, no. Suppl 9, page 135, 2018. (Cited on pages 33 and 34.)

[Giulianotti *et al.* 2003] Pier Cristoforo Giulianotti, Andrea Coratti, Marta Angelini, Fabio Sbrana, Simone Cecconi, Tommaso Balestracci and Giuseppe Caravaglios. *Robotics in general surgery: personal experience in a large community hospital.* Archives of surgery, vol. 138, no. 7, pages 777–784, 2003. (Cited on page 29.)

[Gotohda *et al.* 2022] Naoto Gotohda, Daniel Cherqui, David A Geller, Mohammed Abu Hilal, Giammauro Berardi, Ruben Ciria, Yuta Abe, Takeshi Aoki, Horacio J Asbun, Albert CY Chan*et al. Expert Consensus Guidelines: How to safely perform minimally invasive anatomic liver resection.* Journal of Hepato-Biliary-Pancreatic Sciences, vol. 29, no. 1, pages 16–32, 2022. (Cited on page 32.)

[Gould *et al.* 2007] Harvey Gould, Jan Tobochnik and Wolfgang Christian. An introduction to computer simulation methods third edition (revised). 3rd édition, 2007. (Cited on page 55.)

[Greenway 1983] Clive V Greenway. *Role of splanchnic venous system in overall cardiovascular homeostasis.* In Federation proceedings, volume 42, pages 1678–1684, 1983. (Cited on page 20.)

[Guan *et al.* 2009] Peng Guan, Alexander Weiss, Alexandru O Balan and Michael J Black. *Estimating human shape and pose from a single image.* In IEEE International Conference on Computer Vision, pages 1381–1388. IEEE, 2009. (Cited on page 116.)

[Guéziec *et al.* 2001] André Guéziec, Gabriel Taubin, Francis Lazarus and B Hom. *Cutting and stitching: Converting sets of polygons to manifold surfaces.* IEEE Transactions on Visualization and Computer Graphics, vol. 7, no. 2, pages 136–151, 2001. (Cited on page 47.)

[Guo *et al.* 2022a] Kai Guo, Hu Ye, Xin Gao and Honglin Chen. *An accurate and robust method for absolute pose estimation with UAV using RANSAC.* Sensors, vol. 22, no. 15, page 5925, 2022. (Cited on page 64.)

[Guo *et al.* 2022b] Meng-Hao Guo, Tian-Xing Xu, Jiang-Jiang Liu, Zheng-Ning Liu, Peng-Tao Jiang, Tai-Jiang Mu, Song-Hai Zhang, Ralph R Martin, Ming-Ming Cheng and Shi-Min Hu. *Attention mechanisms in computer vision: A survey.* Computational visual media, vol. 8, no. 3, pages 331–368, 2022. (Cited on pages 87 and 88.)

[Haney *et al.* 2021] Caelán M Haney, Alexander Studier-Fischer, Pascal Probst, Carolyn Fan, Philip C Müller, Mohammad Golriz, Markus K Diener, Thilo Hackert, Beat P Müller-Stich, Arianeb Mehrabi and Felix Nickel. *A systematic review and meta-analysis of randomized controlled trials comparing laparoscopic and open liver resection.* HBP, vol. 23, no. 10, pages 1467–1481, 2021. (Cited on page 29.)

[Hartley & Zisserman 2003] Richard Hartley and Andrew Zisserman. Multiple view geometry in computer vision. Cambridge university press, 2003. (Cited on pages 62 and 65.)

[He *et al.* 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. *Deep residual learning for image recognition.* In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 770–778, 2016. (Cited on page 91.)

[Hilal *et al.* 2018] Mohammad Abu Hilal, Luca Aldrighetti, Ibrahim Dagher, Bjorn Edwin, Roberto Ivan Troisi, Ruslan Alikhanov, Somaiah Aroori, Giulio Belli, Marc Besselink, Javier Briceno*et al. The Southampton consensus guidelines for laparoscopic liver surgery: from indication to implementation.* Annals of surgery, vol. 268, no. 1, pages 11–18, 2018. (Cited on page 33.)

[Hinton *et al.* 2012] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever and Ruslan R Salakhutdinov. *Improving neural networks by preventing co-adaptation of feature detectors.* arXiv preprint arXiv:1207.0580, 2012. (Cited on page 85.)

[Hu *et al.* 2018] Jie Hu, Li Shen and Gang Sun. *Squeeze-and-excitation networks.* In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 7132–7141, 2018. (Cited on page 88.)

[Ikoma *et al.* 2015] Naruhiko Ikoma, Osamu Itano, Go Oshima and Yuko Kitagawa. *Laparoscopic liver mobilization: tricks of the trade to avoid complications*. Surgical Laparoscopy Endoscopy & Percutaneous Techniques, vol. 25, no. 1, pages e21–e23, 2015. (Cited on page 29.)

[Ioffe & Szegedy 2015] Sergey Ioffe and Christian Szegedy. *Batch normalization: Accelerating deep network training by reducing internal covariate shift*. In International Conference on Machine Learning, pages 448–456. pmlr, 2015. (Cited on page 85.)

[Irving *et al.* 2006] Geoffrey Irving, Joseph Teran and Ronald Fedkiw. *Tetrahedral and hexahedral invertible finite elements*. Graphical Models, vol. 68, no. 2, pages 66–89, 2006. (Cited on page 52.)

[Ishizawa *et al.* 2016] Takeaki Ishizawa, Akio Saiura and Norihiro Kokudo. *Clinical application of indocyanine green-fluorescence imaging during hepatectomy*. Hepatobiliary surgery and nutrition, vol. 5, no. 4, page 322, 2016. (Cited on page 34.)

[Jensen 2014] Ellen C Jensen. *Technical Review, Types of Imaging, Part 4—Magnetic Resonance Imaging*. The Anatomical Record, vol. 297, no. 6, pages 973–978, 2014. (Cited on page 26.)

[Jiang *et al.* 2023] Dong Jiang, Yi Qian, Bi-Bo Tan, Xia-Ling Zhu, Hui Dong and Rong Qian. *Preoperative prediction of microvascular invasion in hepatocellular carcinoma using ultrasound features including elasticity*. World Journal of Gastrointestinal Surgery, vol. 15, no. 9, page 2042, 2023. License: CC BY-NC. (Cited on page 27.)

[Jin *et al.* 2021] Danfeng Jin, Mingyue Liu, Jian Huang, Yongfeng Xu, Luping Liu, Changhong Miao and Jing Zhong. *Gas embolism under standard versus low pneumoperitoneum pressure during laparoscopic liver resection (GASES): study protocol for a randomized controlled trial*. Trials, vol. 22, pages 1–12, 2021. (Cited on page 29.)

[Jouda 2016] Mazin Jouda. *Innovative Concepts for the Electronic Interface of Massively Parallel MRI Phased Imaging Arrays*. PhD thesis, Karlsruher Institut für Technologie, 2016. (Cited on page 25.)

[Jung 2021] Haijo Jung. *Basic physical principles and clinical applications of computed tomography*. Progress in Medical Physics, vol. 32, no. 1, pages 1–17, 2021. (Cited on page 24.)

[Kalantari *et al.* 2024] Mohammad Mahdi Kalantari, Erol Ozgur, Mohammad Alkhatib, Emmanuel Buc, Bertrand Le Roy, Richard Modrzejewski, Youcef Mezouar and Adrien Bartoli. *LARLUS: laparoscopic augmented reality from laparoscopic ultrasound*. International Journal of Computer Assisted Radiology and Surgery, pages 1–6, 2024. (Cited on pages 36 and 151.)

[Kambhatla & Leen 1997] Nandakishore Kambhatla and Todd K Leen. *Dimension reduction by local principal component analysis*. Neural computation, vol. 9, no. 7, pages 1493–1516, 1997. (Cited on page 60.)

[Kanazawa *et al.* 2018] Angjoo Kanazawa, Michael J Black, David W Jacobs and Jitendra Malik. *End-to-end recovery of human shape and pose*. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 7122–7131, 2018. (Cited on pages 116 and 119.)

[Kaneko *et al.* 2008] H Kaneko, Y Otsuka, M Tsuchiya, A Tamura, T Katagiri and K Yamazaki. *Application of devices for safe laparoscopic hepatectomy*. HPB, vol. 10, no. 4, pages 219–224, 2008. (Cited on page 29.)

[Kim & Choi 2007] Se Hyung Kim and Byung Ihn Choi. *Three-dimensional and four-dimensional ultrasound: techniques and abdominal applications*. Journal of medical Ultrasound, vol. 15, no. 4, pages 228–242, 2007. (Cited on page 26.)

[Kingma & Ba 2014] Diederik P Kingma and Jimmy Ba. *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980, 2014. (Cited on page 77.)

[Kirillov *et al.* 2023] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, Piotr Dollár and Ross B Girshick. *Segment anything*. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 4015–4026, 2023. (Cited on page 97.)

[Knavel & Brace 2013] Erica M Knavel and Christopher L Brace. *Tumor ablation: common modalities and general practices*. Techniques in vascular and interventional radiology, vol. 16, no. 4, pages 192–200, 2013. (Cited on page 23.)

[Koffron *et al.* 2006] Alan Koffron, David Geller, T Clark Gamblin and Michael Abecassis. *Laparoscopic liver surgery: shifting the management of liver tumors*. Hepatology, vol. 44, no. 6, pages 1694–1700, 2006. (Cited on page 29.)

[Koo *et al.* 2017a] 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 Medical Image Computing and Computer Assisted Intervention, pages 326–334. Springer, 2017. (Cited on pages 126, 127, 143 and 144.)

[Koo *et al.* 2017b] 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 International conference on Medical Image Computing and Computer-Assisted Intervention, pages 326–334. Springer, 2017. (Cited on pages 37, 71, 112, 113 and 114.)

[Koo *et al.* 2022] Bongjin Koo, Maria R Robu, Moustafa Allam, Micha Pfeiffer, Stephen Thompson, Kurinchi Gurusamy, Brian Davidson, Stefanie Speidel, David Hawkes, Danail Stoyanov and Matthew J Clarkson. *Automatic, global registration in laparoscopic liver surgery*. International Journal of Computer Assisted Radiology and Surgery, vol. 17, no. 1, pages 167–176, 2022. (Cited on pages 96, 97 and 112.)

[Krenker *et al.* 2011] Andrej Krenker, Janez Bešter and Andrej Kos. *Introduction to the artificial neural networks.* Artificial Neural Networks: Methodological Advances and Biomedical Applications. InTech, pages 1–18, 2011. (Cited on page 76.)

[Kudo *et al.* 2014] Hiroki Kudo, Takeaki Ishizawa, Keigo Tani, Nobuhiro Harada, Akihiko Ichida, Atsushi Shimizu, Junichi Kaneko, Taku Aoki, Yoshihiro Sakamoto, Yasuhiko Sugawara, Kiyoshi Hasegawa and Norihiro Kokudo. *Visualization of subcapsular hepatic malignancy by indocyanine-green fluorescence imaging during laparoscopic hepatectomy.* Surgical endoscopy, vol. 28, pages 2504–2508, 2014. (Cited on page 33.)

[Labriet *et al.* 2018] Hélène Labriet, Christian Nemoz, Michel Renier, Paul Berkvens, Thierry Brochard, R Cassagne, Hélène Elleaume, François Estève, Camille Verry, Jacques Balosso, Jean François Adam and Emmanuel Brun. *Significant dose reduction using synchrotron radiation computed tomography: first clinical case and application to high resolution CT exams.* Scientific reports, vol. 8, no. 1, page 12491, 2018. License: CC BY. (Cited on page 24.)

[Labrunie *et al.* 2022] M Labrunie, M Ribeiro, F Mourthadhoi, C Tilmant, B Le Roy, E Buc and A Bartoli. *Automatic preoperative 3D model registration in laparoscopic liver resection.* International Journal of Computer Assisted Radiology and Surgery, pages 1–8, 2022. (Cited on page 39.)

[Labrunie *et al.* 2023] Mathieu Labrunie, Daniel Pizarro, Christophe Tilmant and Adrien Bartoli. *Automatic 3D/2D Deformable Registration in Minimally Invasive Liver Resection using a Mesh Recovery Network.* In Medical Imaging with Deep Learning, 2023. (Cited on pages 40 and 115.)

[Labrunie *et al.* n.d.] Mathieu Labrunie, Daniel Pizarro, Christophe Tilmant and Adrien Bartoli. *Generic Liver Modelling with Application to Mini-invasive Surgery Guidance.* International Conference on Medical Imaging and Computer-Aided Diagnosis 2024, n.d. (In press). (Cited on page 40.)

[Langenbuch 1888] Carl Langenbuch. *Ein Fall von Resektion eines linksseitigen Schnurlappens der Leber, Heilung.* Berl Klin Wochenschr, vol. 25, pages 37–38, 1888. (Cited on page 28.)

[Langø *et al.* 2012] Thomas Langø, Toril N Hernes, Ronald Mårvik and A Malik. *Navigated ultrasound in laparoscopic surgery.* Advances in Laparoscopic Surgery, pages 77–98, 2012. (Cited on pages 33 and 36.)

[Lengyel 2010] Eric Stephen Lengyel. Voxel-based terrain for real-time virtual simulations. University of California, Davis, 2010. (Cited on page 46.)

[Lepetit *et al.* 2009] Vincent Lepetit, Francesc Moreno-Noguer and Pascal Fua. *EPnP: An accurate O(n) solution to the PnP problem.* International Journal of Computer Vision, vol. 81, pages 155–166, 2009. (Cited on page 67.)

[Li *et al.* 2005] Min Li, Chandra Kambhamettu and Maureen Stone. *Automatic contour tracking in ultrasound images.* Clinical linguistics & phonetics, vol. 19, no. 6-7, pages 545–554, 2005. (Cited on page 101.)

[Liao *et al.* 2018] Yiyi Liao, Simon Donne and Andreas Geiger. *Deep marching cubes: Learning explicit surface representations.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2916–2925, 2018. Copyright: *© 2018, IEEE.* (Cited on pages 45 and 46.)

[Liepa 2003] Peter Liepa. *Filling holes in meshes.* In Proceedings of the Eurographics/ACM SIGGRAPH symposium on Geometry processing, pages 200–205, 2003. (Cited on page 48.)

[Liu *et al.* 2021] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin and Baining Guo. *Swin transformer: Hierarchical vision transformer using shifted windows.* In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 10012–10022, 2021. (Cited on pages 92 and 93.)

[Liu *et al.* 2023] Rong Liu, Mohammed Abu Hilal, Go Wakabayashi, Ho-Seong Han, Chinnusamy Palanivelu, Ugo Boggi, Thilo Hackert, Hong-Jin Kim, Xiao-Ying Wang, Ming-Gen Hu*et al. International experts consensus guidelines on robotic liver resection in 2023.* World Journal of Gastroenterology, vol. 29, no. 32, page 4815, 2023. (Cited on page 31.)

[Lladó & Figueras 2004] L Lladó and J Figueras. *Techniques of orthotopic liver transplantation.* HBP, vol. 6, no. 2, pages 69–75, 2004. (Cited on page 28.)

[Lohr *et al.* 2022] Matthew J Lohr, Gabriella P Sugerman, Sotirios Kakaletsis, Emma Lejeune and Manuel K Rausch. *An introduction to the Ogden model in biomechanics: benefits, implementation tools and limitations.* Philosophical Transactions of the Royal Society A, vol. 380, no. 2234, page 20210365, 2022. (Cited on page 54.)

[Lombardi *et al.* 2018] Stephen Lombardi, Jason Saragih, Tomas Simon and Yaser Sheikh. *Deep appearance models for face rendering.* ACM Transactions on Graphics, vol. 37, no. 4, pages 1–13, 2018. (Cited on page 82.)

[Loper *et al.* 2015] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll and Michael J Black. *SMPL: A skinned multi-person linear model.* ACM Transactions On Graphics, vol. 34, no. 6, pages 1–16, 2015. (Cited on page 116.)

[Lorensen & Cline 1998] William E Lorensen and Harvey E Cline. *Marching cubes: A high resolution 3D surface construction algorithm.* In Seminal graphics: pioneering efforts that shaped the field, pages 347–353. 1998. (Cited on page 44.)

[Lu *et al.* 2019] Xiankai Lu, Wenguan Wang, Chao Ma, Jianbing Shen, Ling Shao and Fatih Porikli. *See more, know more: Unsupervised video object segmentation with co-attention siamese networks.* In Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, pages 3623–3632, 2019. (Cited on pages 95 and 99.)

[Lüthi *et al.* 2017] Marcel Lüthi, Thomas Gerig, Christoph Jud and Thomas Vetter. *Gaussian process morphable models.* IEEE transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 8, pages 1860–1873, 2017. (Cited on page 138.)

[Maas *et al.* 2012] Steve A Maas, Benjamin J Ellis, Gerard A Ateshian and Jeffrey A Weiss. *FEBio: finite elements for biomechanics.* Journal of biomechanical engineering, vol. 134, no. 1, 2012. (Cited on page 55.)

[Macklin *et al.* 2016] Miles Macklin, Matthias Müller and Nuttapong Chentanez. *XPBD: position-based simulation of compliant constrained dynamics.* In Proceedings of the International Conference on Motion in Games, pages 49–54, 2016. (Cited on page 114.)

[Makowka *et al.* 1988] Leonard Makowka, Andrei C Stieber, Linda Sher, Del Kahn, Louis Mieles, James Bowman, J Wallis Marsh and Thomas E Starzl. *Surgical technique of orthotopic liver transplantation.* Gastroenterology Clinics of North America, vol. 17, no. 1, pages 33–51, 1988. (Cited on page 28.)

[Marchesseau *et al.* 2017] Stéphanie Marchesseau, Simon Chatelin and Hervé Delingette. *Nonlinear biomechanical model of the liver.* In Biomechanics of living organs, pages 243–265. Elsevier, 2017. (Cited on page 56.)

[Marchn *et al.* n.d.] Jordi Marchn, O. Paul Gobée and LUMC. *JMarchn - Biliary system - Latin and English labels*, n.d., https://anatomytool.org/content/jmarchn-biliary-system-latin-and-english-labels. Accessed 01 May 2024. License: CC BY-SA. (Cited on page 21.)

[Margonis *et al.* 2018] Georgios A Margonis, Theodoros N Sergentanis, Ioannis Ntanasis-Stathopoulos, Nikolaos Andreatos, Ioannis-Georgios Tzanninis, Kazunari Sasaki, Theodora Psaltopoulou, Jaeyun Wang, Stefan Buettner, Apostolos E Papalois, Jin He, Christopher L Wolfgang, Timothy M Pawlik and Matthew J Weiss. *Impact of surgical margin width on recurrence and overall survival following R0 hepatic resection of colorectal metastases: a systematic review and meta-analysis.* Annals of surgery, vol. 267, no. 6, pages 1047–1055, 2018. (Cited on page 32.)

[Martin *et al.* 2020] Jack Martin, Angelica Petrillo, Elizabeth C Smyth, Nadeem Shaida, Samir Khwaja, HK Cheow, Adam Duckworth, Paula Heister, Raaj Praseedom, Asif Jah, Anita Balakrishnan, Simon Harper, Siong Liau, Vasilis Kosmoliaptsis and Emmanuel Huguet. *Colorectal liver metastases: Current management and future perspectives.* World journal of clinical oncology, vol. 11, no. 10, page 761, 2020. (Cited on page 22.)

[Mendoza-Ramírez *et al.* 2023] Carlos E Mendoza-Ramírez, Juan C Tudon-Martinez, Luis C Félix-Herrán, Jorge de J Lozoya-Santos and Adriana Vargas-Martínez. *Augmented reality: survey.* Applied Sciences, vol. 13, no. 18, page 10491, 2023. (Cited on page 35.)

[Mezger *et al.* 2013] Uli Mezger, Claudia Jendrewski and Michael Bartels. *Navigation in surgery.* Langenbeck's archives of surgery, vol. 398, pages 501–514, 2013. (Cited on page 33.)

[Mhiri *et al.* 2024] Islem Mhiri, Daniel Pizarro and Adrien Bartoli. *Neural patient-specific 3D–2D registration in laparoscopic liver resection.* International Journal of Computer Assisted Radiology and Surgery, pages 1–8, 2024. (Cited on page 131.)

[Mikhailov *et al.* 2024] Ivan Mikhailov, Benoit Chauveau, Nicolas Bourdel and Adrien Bartoli. *A deep learning-based interactive medical image segmentation framework with sequential memory.* Computer Methods and Programs in Biomedicine, vol. 245, page 108038, 2024. (Cited on pages 43, 107 and 150.)

[Min *et al.* 2019] Zhe Min, Li Liu and Max Q.-H. Meng. *Generalized Non-rigid Point Set Registration with Hybrid Mixture Models Considering Anisotropic Positional Uncertainties.* In Medical Image Computing and Computer Assisted Intervention, pages 547–555. Springer, 2019. (Cited on page 113.)

[Modrzejewski 2020] Richard Modrzejewski. *Recalage déformable, jeux de données et protocoles d'évaluation pour la chirurgie mini-invasive abdominale augmentée.* PhD thesis, Université Clermont Auvergne, August 2020. (Cited on page 61.)

[Montalti *et al.* 2024] Roberto Montalti, Gianluca Cassese, Ahmed Zidan, Gianluca Rompianesi, Mariano Cesare Giglio, Silvia Campanile, Lorenza Arena, Marco Maione and Roberto I Troisi. *Local recurrence risk factors and outcomes in minimally invasive thermal ablation for liver tumors: a single-institution analysis.* HPB, 2024. (Cited on page 28.)

[Mooney 1940] Melvin Mooney. *A theory of large elastic deformation.* Journal of applied physics, vol. 11, no. 9, pages 582–592, 1940. (Cited on page 54.)

[Moré 2006] Jorge J Moré. *The Levenberg-Marquardt algorithm: implementation and theory.* In Numerical analysis: proceedings of the biennial Conference held at Dundee, June 28–July 1, 1977, pages 105–116. Springer, 2006. (Cited on page 71.)

[Motte & Kaufman 2013] Stéphanie Motte and Laura J Kaufman. *Strain stiffening in collagen I networks.* Biopolymers, vol. 99, no. 1, pages 35–46, 2013. (Cited on page 54.)

[Müller *et al.* 2007] Matthias Müller, Bruno Heidelberger, Marcus Hennix and John Ratcliff. *Position based dynamics.* Journal of Visual Communication and Image Representation, vol. 18, no. 2, pages 109–118, 2007. (Cited on page 113.)

[Myronenko *et al.* 2006] Andriy Myronenko, Xubo Song and Miguel Carreira-Perpinan. *Non-rigid point set registration: Coherent point drift.* Advances in Neural Information Processing Systems, vol. 19, 2006. (Cited on page 135.)

[Nagino *et al.* 2021] Masato Nagino, Ronald DeMatteo, Hauke Lang, Daniel Cherqui, Massimo Malago, Shoji Kawakatsu, Michelle L DeOliveira, René Adam, Luca Aldrighetti, Karim Boudjema*et al. Proposal of a new comprehensive notation for hepatectomy: the "New World" terminology.* Annals of surgery, vol. 274, no. 1, pages 1–3, 2021. (Cited on pages 23 and 31.)

[Nagy *et al.* 2020] Peter Nagy, Snorri S Thorgeirsson and Joe W Grisham. *Organizational principles of the liver*. The Liver: Biology and Pathobiology, pages 1–13, 2020. (Cited on pages 20 and 22.)

[Netter 2014] Frank H Netter. Atlas of human anatomy. Elsevier health sciences, 2014. (Cited on pages 135 and 136.)

[Nevarez & Yopp 2021] Nicole M Nevarez and Adam C Yopp. *Anatomic vs. non-anatomic liver resection for hepatocellular carcinoma: standard of care or unfilled promises?* Hepatoma Research, vol. 7, page 66, 2021. (Cited on pages 31 and 32.)

[Nocedal & Wright 1999] Jorge Nocedal and Stephen J Wright. Numerical optimization. Springer, 1999. (Cited on pages 68 and 70.)

[Ogden 1972] Raymond William Ogden. *Large deformation isotropic elasticity–on the correlation of theory and experiment for incompressible rubberlike solids*. Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, vol. 326, no. 1567, pages 565–584, 1972. (Cited on page 54.)

[Oh *et al.* 2019] Seoung Wug Oh, Joon-Young Lee, Ning Xu and Seon Joo Kim. *Video object segmentation using space-time memory networks*. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 9226–9235, 2019. (Cited on pages 95 and 96.)

[OpenStax College n.d.] OpenStax College. *Blood Flow Through the Heart*, n.d., https://openstax.org/books/anatomy-and-physiology-2e/pages/20-1-structure-and-function-of-blood-vessels. Accessed 03 May 2024. License: CC BY. (Cited on page 21.)

[Orcutt & Anaya 2018] Sonia T Orcutt and Daniel A Anaya. *Liver resection and surgical strategies for management of primary liver cancer*. Cancer Control, vol. 25, no. 1, page 1073274817744621, 2018. (Cited on page 23.)

[Orth *et al.* 2009] Robert C Orth, Michael J Wallace and Michael D Kuo. *C-arm cone-beam CT: general principles and technical considerations for use in interventional radiology*. Journal of vascular and interventional radiology, vol. 20, no. 7, pages S538–S544, 2009. (Cited on page 34.)

[O'shea & Nash 2015] Keiron O'shea and Ryan Nash. *An introduction to convolutional neural networks*. arXiv preprint arXiv:1511.08458, 2015. (Cited on page 83.)

[Özgür *et al.* 2018] Erol Özgür, Bongjin Koo, Bertrand Le Roy, Emmanuel Buc and Adrien Bartoli. *Preoperative liver registration for augmented monocular laparoscopy using backward–forward biomechanical simulation*. International Journal of Computer Assisted Radiology and Surgery, vol. 13, no. 10, pages 1629–1640, 2018. (Cited on pages 37, 71 and 112.)

[Ozougwu 2017] Jevas C Ozougwu. *Physiology of the liver*. International Journal of Research in Pharmacy and Biosciences, vol. 4, no. 8, pages 13–24, 2017. (Cited on page 20.)

[Pajarola 2000] Renato Pajarola. *Advanced 3D Computer Graphics*, 2000. (Cited on page 44.)

[Park *et al.* 2019] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe and Steven Lovegrove. *Deepsdf: Learning continuous signed distance functions for shape representation.* In Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, pages 165–174, 2019. (Cited on page 135.)

[Parra *et al.* 2023] Natalia Salinas Parra, Heather M Ross, Adnan Khan, Marisa Wu, Risa Goldberg, Lokesh Shah, Sarah Mukhtar, Jacob Beiriger, Alexis Gerber and Dina Halegoua-DeMarzio. *Advancements in the diagnosis of hepatocellular carcinoma.* International Journal of Translational Medicine, vol. 3, no. 1, pages 51–65, 2023. (Cited on pages 26 and 27.)

[Pei *et al.* 2024] Jialun Pei, Ruize Cui, Yaoqian Li, Weixin Si, Jing Qin and Pheng-Ann Heng. *Depth-Driven Geometric Prompt Learning for Laparoscopic Liver Landmark Detection.* arXiv preprint arXiv:2406.17858, 2024. (Cited on pages 97, 98 and 107.)

[Pellicer-Valero *et al.* 2020] Oscar J Pellicer-Valero, María José Rupérez, Sandra Martínez-Sanchis and José D Martín-Guerrero. *Real-time biomechanical modeling of the liver using machine learning models trained on finite element method simulations.* Expert Systems with Applications, vol. 143, page 113083, 2020. (Cited on pages 56 and 135.)

[Peng & Li 2010] En Peng and Ling Li. *Camera calibration using one-dimensional information and its applications in both controlled and uncontrolled environments.* Pattern Recognition, vol. 43, no. 3, pages 1188–1198, 2010. (Cited on page 64.)

[Pfeiffer *et al.* 2019] Micha Pfeiffer, Isabel Funke, Maria R Robu, Sebastian Bodenstedt, Leon Strenger, Sandy Engelhardt, Tobias Roß, Matthew J Clarkson, Kurinchi Gurusamy, Brian R Davidson, Lena Maier-Hein, Carina Riediger, Thilo Welsch, Jürgen Weitz and Stefanie Speidel. *Generating large labeled data sets for laparoscopic image processing tasks using unpaired image-to-image translation.* In Medical Image Computing and Computer Assisted Intervention, pages 119–127. Springer, 2019. (Cited on page 96.)

[Plantefeve *et al.* 2014] Rosalie Plantefeve, Nazim Haouchine, Jean-Pierre Radoux and Stéphane Cotin. *Automatic alignment of pre and intraoperative data using anatomical landmarks for augmented laparoscopic liver surgery.* In International Symposium on Biomedical Simulation, pages 58–66. Springer, 2014. (Cited on pages 37, 71 and 72.)

[Podareanu *et al.* 2019] Damian Podareanu, Valeriu Codreanu, S Aigner, C Leeuwen and V Weinberg. *Best practice guide - deep learning.* Partnership for Advanced Computing in Europe, vol. 2, 2019. (Cited on page 84.)

[Puiseux *et al.* 2021] Thomas Puiseux, Anou Sewonu, Ramiro Moreno, Simon Mendez and Franck Nicoud. *Numerical simulation of time-resolved 3D phase-contrast magnetic resonance imaging.* PLoS One, vol. 16, no. 3, page e0248816, 2021. License: CC BY. (Cited on page 25.)

[Rabbani *et al.* 2022] Navid Rabbani, Lilian Calvet, Yamid Espinel, Bertrand Le Roy, Mathieu Ribeiro, Emmanuel Buc and Adrien Bartoli. *A methodology and clinical dataset with ground-truth to evaluate registration accuracy quantitatively in computer-assisted laparoscopic liver resection.* Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, vol. 10, no. 4, pages 441–450, 2022. (Cited on pages 36, 98, 123 and 150.)

[Raj *et al.* 2013] Santhosh Raj, Farah Gillan Irani, Kiang Hiong Tay and Bien Soo Tan. *C-arm cone beam computed tomography: a new tool in the interventional suite.* Annals of the Academy of Medicine, Singapore, vol. 42, no. 11, pages 585–92, 2013. (Cited on page 34.)

[Rajapaksha 2022] Indu Rajapaksha. *Liver fibrosis, liver cancer, and advances in therapeutic approaches.* Livers, vol. 2, no. 4, pages 372–386, 2022. (Cited on page 22.)

[Ramírez-Hernández *et al.* 2020] Luis R Ramírez-Hernández, Julio C Rodríguez-Quinoñez, Moises J Castro-Toscano, Daniel Hernández-Balbuena, Wendy Flores-Fuentes, Raúl Rascón-Carmona, Lars Lindner and Oleg Sergiyenko. *Improve three-dimensional point localization accuracy in stereo vision systems using a novel camera calibration method.* International Journal of Advanced Robotic Systems, vol. 17, no. 1, page 1729881419896717, 2020. (Cited on page 63.)

[Raskar & Cohen 1999] Ramesh Raskar and Michael Cohen. *Image precision silhouette edges.* In Proceedings of the symposium on Interactive 3D graphics, pages 135–140, 1999. (Cited on page 73.)

[Reich *et al.* 1991] Harry Reich, Fran McGlynn, John DeCaprio and Robert Budin. *Laparoscopic excision of benign liver lesions.* Obstetrics & Gynecology, vol. 78, no. 5, pages 956–957, 1991. (Cited on page 29.)

[Robu *et al.* 2018] Maria R Robu, João Ramalhinho, Stephen Thompson, Kurinchi Gurusamy, Brian Davidson, David Hawkes, Danail Stoyanov and Matthew J Clarkson. *Global rigid registration of CT to video in laparoscopic liver surgery.* International Journal of Computer Assisted Radiology and Surgery, vol. 13, no. 6, pages 947–956, 2018. (Cited on page 113.)

[Rodrigues *et al.* 2017] Túlio Felício da Cunha Rodrigues, Bianca Silveira, Flávia Pádua Tavares, Gustavo Moreira Madeira, Iara Proença Xavier, Jorge Henrique Costa Ribeiro, Rayanna Mara de Oliveira Santos Pereira and Sávio Lana Siqueira. *Open, laparoscopic, and robotic-assisted hepatectomy in resection of liver tumors: a non-systematic review.* ABCD. Arquivos Brasileiros de Cirurgia Digestiva (São Paulo), vol. 30, pages 155–160, 2017. (Cited on page 33.)

[Rompianesi *et al.* 2023] Gianluca Rompianesi, Francesca Pegoraro, Lorenzo Ramaci, Carlo DL Ceresa, Roberto Montalti and Roberto I Troisi. *Preoperative planning and intraoperative real-time navigation with indocyanine green fluorescence in robotic liver surgery.* Langenbeck's Archives of Surgery, vol. 408, no. 1, page 292, 2023. License: CC BY. (Cited on page 34.)

[Ronneberger *et al.* 2015] Olaf Ronneberger, Philipp Fischer and Thomas Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation.* In Medical Image Computing and Computer-Assisted Intervention, 2015. (Cited on page 89.)

[Rosenfeld & Pfaltz 1966] Azriel Rosenfeld and John L Pfaltz. *Sequential operations in digital picture processing.* Journal of the ACM, vol. 13, no. 4, pages 471–494, 1966. (Cited on page 121.)

[Roweis & Saul 2000] Sam T Roweis and Lawrence K Saul. *Nonlinear dimensionality reduction by locally linear embedding.* Science, vol. 290, no. 5500, pages 2323–2326, 2000. (Cited on pages 60 and 61.)

[Ruder 2016] Sebastian Ruder. *An overview of gradient descent optimization algorithms.* arXiv preprint arXiv:1609.04747, 2016. (Cited on page 77.)

[Ruppert 1995] Jim Ruppert. *A Delaunay refinement algorithm for quality 2-dimensional mesh generation.* Journal of algorithms, vol. 18, no. 3, pages 548–585, 1995. (Cited on page 49.)

[Ryan *et al.* 2016] Michael J Ryan, Jonathon Willatt, Bill S Majdalany, Ania Z Kielar, Suzanne Chong, Julie A Ruma and Amit Pandya. *Ablation techniques for primary and metastatic liver tumors.* World journal of hepatology, vol. 8, no. 3, page 191, 2016. (Cited on page 28.)

[Sackmann *et al.* 1994] M Sackmann, J Pauletzki, FM Zwiebel and J Holl. *Three-dimensional ultrasonography in hepatobiliary and pancreatic diseases.* Bildgebung, vol. 61, no. 2, pages 100–103, 1994. (Cited on page 26.)

[Salehi *et al.* 2017] Seyed Sadegh Mohseni Salehi, Deniz Erdogmus and Ali Gholipour. *Tversky loss function for image segmentation using 3D fully convolutional deep networks.* In International workshop on machine learning in medical imaging, pages 379–387. Springer, 2017. (Cited on page 100.)

[Schroeder *et al.* 2006] Will Schroeder, Ken Martin and Bill Lorensen. The visualization toolkit (4th ed.). Kitware, 2006. (Cited on page 46.)

[Schroeder *et al.* 2015] William Schroeder, Rob Maynard and Berk Geveci. *Flying edges: A high-performance scalable isocontouring algorithm.* In IEEE Symposium on Large Data Analysis and Visualization, pages 33–40. IEEE, 2015. (Cited on page 46.)

[Sederberg & Parry 1986] Thomas W Sederberg and Scott R Parry. *Free-form deformation of solid geometric models.* In Proceedings of the conference on Computer Graphics and Interactive Techniques, pages 151–160, 1986. (Cited on pages 56 and 58.)

[Sengun *et al.* 2021] KE Sengun, YT Cetin, MS Guzel, Serhat Can and Erkan Bostanci. *Automatic liver segmentation from CT images using deep learning algorithms: a comparative study.* arXiv preprint arXiv:2101.09987, 2021. (Cited on page 43.)

[Seo *et al.* 2017] Nieun Seo, Do Young Kim and Jin-Young Choi. *Cross-sectional imaging of intrahepatic cholangiocarcinoma: development, growth, spread, and prognosis.* American Journal of Roentgenology, vol. 209, no. 2, pages W64–W75, 2017. (Cited on page 23.)

[Shewchuk 1998] Jonathan Richard Shewchuk. *Tetrahedral mesh generation by Delaunay refinement.* In Proceedings of the symposium on Computational geometry, pages 86–95, 1998. (Cited on page 49.)

[Shewchuk 2002] Jonathan Richard Shewchuk. *Constrained Delaunay Tetrahedralizations and Provably Good Boundary Recovery.* In IMR, pages 193–204. Citeseer, 2002. (Cited on pages 48 and 120.)

[Shlens 2014] Jonathon Shlens. *A tutorial on principal component analysis.* arXiv preprint arXiv:1404.1100, 2014. (Cited on page 59.)

[Si 2013] Hang Si. *TetGen: A quality tetrahedral mesh generator and a 3D Delaunay triangulator (Version 1.5—User's Manual).* 2013. (Cited on page 49.)

[Sifakis & Barbic 2012] Eftychios Sifakis and Jernej Barbic. *FEM simulation of 3D deformable solids: a practitioner's guide to theory, discretization and model reduction.* In ACM SIGGRAPH 2012 courses, pages 1–50. 2012. (Cited on pages 50, 51, 53, 54 and 55.)

[Singh & Rabi 2019] Haobam Rajajee Singh and Suganthy Rabi. *Study of morphological variations of liver in human.* Translational Research in Anatomy, vol. 14, pages 1–5, 2019. (Cited on page 135.)

[Slagter n.d.] Ron Slagter. *Drawing Position of the heart and other organs in the thorax - no labels*, n.d., https://anatomytool.org/content/slagter-drawing-position-heart-and-other-organs-thorax-no-labels. Modified, labels added. Accessed 01 May 2024. License: CC BY-NC-SA. (Cited on page 21.)

[Song *et al.* 2024] Zhendong Song, Huiming Wu, Wei Chen and Adam Slowik. *Improving automatic segmentation of liver tumor images using a deep learning model.* Heliyon, vol. 10, no. 7, 2024. (Cited on page 43.)

[Sorkine & Alexa 2007] Olga Sorkine and Marc Alexa. *As-rigid-as-possible surface modeling.* In Symposium on Geometry processing, volume 4, pages 109–116. Citeseer, 2007. (Cited on page 58.)

[Srivastava *et al.* 2014] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever and Ruslan Salakhutdinov. *Dropout: a simple way to prevent neural networks from overfitting.* The journal of machine learning research, vol. 15, no. 1, pages 1929–1958, 2014. (Cited on page 85.)

[Strasberg *et al.* 2000] SM Strasberg, J Belghiti, P-A Clavien, E Gadzijev, JO Garden, W-Y Lau, M Makuuchi and RW Strong. *The Brisbane 2000 terminology of liver*

anatomy and resections. HBP, vol. 2, no. 3, pages 333–339, 2000. (Cited on pages 22, 23 and 31.)

[Sun & Yuan 2006] Wenyu Sun and Ya-Xiang Yuan. Optimization theory and methods: nonlinear programming, volume 1. Springer Science & Business Media, 2006. (Cited on pages 70 and 71.)

[Sun *et al.* 2022] Shanlin Sun, Kun Han, Deying Kong, Hao Tang, Xiangyi Yan and Xiaohui Xie. *Topology-preserving shape reconstruction and registration via neural diffeomorphic flow*. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 20845–20855, 2022. (Cited on page 135.)

[Sun *et al.* 2023] Hui-Chuan Sun, Ying-Hao Shen, Cheng Huang, Xiao-Dong Zhu, Chang-Jun Tan, Zhao-You Tang, Jia Fan and Jian Zhou. *The development and prospects of liver surgery*. Clinical Surgical Oncology, vol. 2, no. 1, page 100009, 2023. (Cited on page 29.)

[Taubin 1995] Gabriel Taubin. *Curve and surface smoothing without shrinkage*. In Proceedings of IEEE International Conference on Computer Vision, pages 852–857. IEEE, 1995. (Cited on page 46.)

[Teatini *et al.* 2019] Andrea Teatini, Egidijus Pelanis, Davit L Aghayan, Rahul Prasanna Kumar, Rafael Palomar, Åsmund Avdem Fretland, Bjørn Edwin and Ole Jakob Elle. *The effect of intraoperative imaging on surgical navigation for laparoscopic liver resection surgery*. Scientific Reports, vol. 9, no. 1, page 18687, 2019. License: CC BY. (Cited on page 36.)

[Thiruchelvam *et al.* 2021] Nita Thiruchelvam, Ser Yee Lee and Adrian Kah Heng Chiow. *Patient and port positioning in laparoscopic liver resections*. Hepatoma Res, vol. 7, page 22, 2021. (Cited on page 29.)

[Tian *et al.* 2022] Yating Tian, Hongwen Zhang, Yebin Liu and Limin Wang. *Recovering 3D human mesh from monocular images: A survey*. Eprint arXiv:2203.01923, 2022. (Cited on page 116.)

[Truesdell & Toupin 1960] Clifford Truesdell and Richard Toupin. The classical field theories. Springer, 1960. (Cited on page 53.)

[Tsilimigras *et al.* 2023] Diamantis I Tsilimigras, Ioannis Ntanasis-Stathopoulos and Timothy M Pawlik. *Molecular mechanisms of colorectal liver metastases*. Cells, vol. 12, no. 12, page 1657, 2023. (Cited on page 23.)

[Twinanda *et al.* 2016] Andru P Twinanda, Sherif Shehata, Didier Mutter, Jacques Marescaux, Michel De Mathelin and Nicolas Padoy. *Endonet: a deep architecture for recognition tasks on laparoscopic videos*. IEEE transactions on medical imaging, vol. 36, no. 1, pages 86–97, 2016. (Cited on page 97.)

[Uppal & Mogra 2010] Talat Uppal and Ritu Mogra. *RBC motion and the basis of ultrasound Doppler instrumentation*. Australasian journal of ultrasound in medicine, vol. 13, no. 1, pages 32–34, 2010. (Cited on page 26.)

[Valette & Chassery 2004] Sébastien Valette and Jean-Marc Chassery. *Approximated centroidal voronoi diagrams for uniform polygonal mesh coarsening.* In Computer Graphics Forum, volume 23, pages 381–389. Wiley Online Library, 2004. (Cited on page 46.)

[van der Steen *et al.* 2021] Koen van der Steen, Koop Bosscha and Daan J Lips. *The value of laparoscopic intraoperative ultrasound of the liver by the surgeon.* Annals of Laparoscopic and Endoscopic Surgery, vol. 6, 2021. (Cited on page 33.)

[Vaswani *et al.* 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser and Illia Polosukhin. *Attention is all you need.* Advances in Neural Information Processing Systems, vol. 30, 2017. (Cited on pages 86 and 87.)

[Wang *et al.* 2015] Xiang Wang, Yuan Zheng, Zhenzhou Zhao and Jinping Wang. *Bearing fault diagnosis based on statistical locally linear embedding.* Sensors, vol. 15, no. 7, pages 16225–16247, 2015. License: CC BY. (Cited on page 60.)

[Wang *et al.* 2018] Xiaolong Wang, Ross Girshick, Abhinav Gupta and Kaiming He. *Non-local neural networks.* In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 7794–7803, 2018. (Cited on pages 86 and 87.)

[Wang *et al.* 2021] Gui-sheng Wang, Zhi-yi Zhang, Xue-ting Qi, Jin Liu, Ting Liu, Jing-wei Zhao, Xiao-xia Chen and Yi Chen. *The technology of artificial pneumoperitoneum CT and its application in diagnosis of abdominal adhesion.* Scientific Reports, vol. 11, no. 1, page 20785, 2021. (Cited on pages 79, 131 and 149.)

[Werner *et al.* 2015] Melanie Werner, Sabrina Driftmann, Kathrin Kleinehr, Gernot M Kaiser, Zotlan Mathé, Juergen-Walter Treckmann, Andreas Paul, Kathrin Skibbe, Joerg Timm, Ali Canbay, Guido Gerken, Joerg F Schlaak and Ruth Broering. *All-in-one: advanced preparation of human parenchymal and non-parenchymal liver cells.* PloS one, vol. 10, no. 9, page e0138655, 2015. (Cited on page 22.)

[William Moebs 2016] Jeff Sanny William Moebs Samuel J. Ling. University physics volume 1. OpenStax, 2016. (Cited on page 50.)

[Wolfe 1969] Philip Wolfe. *Convergence conditions for ascent methods.* SIAM review, vol. 11, no. 2, pages 226–235, 1969. (Cited on page 68.)

[Woo *et al.* 2018] Sanghyun Woo, Jongchan Park, Joon-Young Lee and In So Kweon. *Cbam: Convolutional block attention module.* In Proceedings of the European Conference on Computer Vision (ECCV), pages 3–19, 2018. (Cited on page 88.)

[Wright & Holt 1985] SJ Wright and John Norman Holt. *An inexact levenberg-marquardt method for large sparse nonlinear least squares.* The ANZIAM Journal, vol. 26, no. 4, pages 387–403, 1985. (Cited on page 71.)

[Yang & Qiu 2021] Jin-Ying Yang and Ben-Sheng Qiu. *The advance of magnetic resonance elastography in tumor diagnosis.* Frontiers in Oncology, vol. 11, page 722703, 2021. (Cited on page 28.)

[Yeong *et al.* 2021] De Jong Yeong, Gustavo Velasco-Hernandez, John Barry and Joseph Walsh. *Sensor and sensor fusion technology in autonomous vehicles: A review.* Sensors, vol. 21, no. 6, page 2140, 2021. (Cited on page 62.)

[Yin *et al.* 2022] Wei Yin, Jianming Zhang, Oliver Wang, Simon Niklaus, Simon Chen, Yifan Liu and Chunhua Shen. *Towards accurate reconstruction of 3D scene shape from a single monocular image.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 45, no. 5, pages 6480–6494, 2022. (Cited on page 97.)

[Yoh *et al.* 2019] Tomoaki Yoh, François Cauchy and Olivier Soubrane. *Techniques for laparoscopic liver parenchymal transection.* Hepatobiliary surgery and nutrition, vol. 8, no. 6, page 572, 2019. (Cited on page 29.)

[Yu *et al.* 2017] Zhiding Yu, Chen Feng, Ming-Yu Liu and Srikumar Ramalingam. *Casenet: Deep category-aware semantic edge detection.* In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 5964–5973, 2017. (Cited on page 96.)

[Zhang *et al.* 2021] Hongwen Zhang, Yating Tian, Xinchi Zhou, Wanli Ouyang, Yebin Liu, Limin Wang and Zhenan Sun. *Pymaf: 3D human pose and shape regression with pyramidal mesh alignment feedback loop.* In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 11446–11456, 2021. (Cited on pages 131 and 150.)

[Zhang 2000] Zhengyou Zhang. *A flexible new technique for camera calibration.* IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 11, pages 1330–1334, 2000. (Cited on pages 64 and 79.)

[Zheng *et al.* 2021] Zerong Zheng, Tao Yu, Qionghai Dai and Yebin Liu. *Deep implicit templates for 3D shape representation.* In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 1429–1439, 2021. (Cited on page 135.)

[Zhou *et al.* 2023] Tianfei Zhou, Liulei Li, Gustav Bredell, Jianwu Li, Jan Unkelbach and Ender Konukoglu. *Volumetric memory network for interactive medical image segmentation.* Medical Image Analysis, vol. 83, page 102599, 2023. (Cited on pages 43, 107 and 150.)

[Zhu *et al.* 2020] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang and Jifeng Dai. *Deformable detr: Deformable transformers for end-to-end object detection.* arXiv preprint arXiv:2010.04159, 2020. (Cited on page 95.)