High performance image analysis for very high resolution biomedical data acquired with synchrotron imaging

Xiaowen Liang

UNIVERSITÉ DE REIMS CHAMPAGNE-ARDENNE

ÉCOLE DOCTORALE SCIENCES DU NUMERIQUE ET DE L'INGENIEUR n°620

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE REIMS CHAMPAGNE-ARDENNE

Discipline : INFORMATIQUE

Spécialité : Traitement d'images

Présentée et soutenue publiquement par

XIAOWEN LIANG

Le 3 octobre 2023

Analyse d'images haute performance pour les données biomédicales très haute résolution acquises en imagerie synchrotron

---

Thèse dirigée par **NICOLAS PASSAT, SÉBASTIEN ALMAGRO ET LAURENT DEBELLE**

JURY

M. Xavier Descombes,	Directeur de Recherche,	Inria,	**Rapporteur**
M. Nicolas Loménie,	Maître de Conférences HDR,	Université Paris-Cité,	**Rapporteur**
Mme Sophie Desset,	Ingénieure de Recherche	INSERM,	**Examinatrice**
M. Cédric Wemmert,	Professeur,	Université de Strasbourg,	**Examinateur**
M. Laurent Debelle,	Professeur,	Université de Reims Champagne-Ardenne,	**Examinateur**
M. Nicolas Passat,	Professeur,	Université de Reims Champagne-Ardenne,	**Directeur de thèse**
M. Sébastien Almagro,	Maître de Conférences HDR,	Université de Reims Champagne-Ardenne,	**Co-directeur de thèse**

Acknowledgments

This thesis is under the supervision of the research units CReSTIC and MEDyC. I wish to convey my thanks to Rezak Ayad, Bernard Riera, and Laurent Martiny for your trust.

I would like to thank Xavier Descombes and Nicolas Loménie for accepting to review my thesis and for the time they spent on the reviewing process. I would also like to thank Sophie Desset and Cédric Wemmert for being members of my jury.

I sincerely express my gratitude to my directors of thesis, Nicolas Passat, Sébastien Almagro, and Laurent Debelle, for all the invaluable support throughout these three years.

Nicolas, I am so delighted to have had the opportunity to complete a thesis under your guidance. Thank you for your patience, as well as your comprehensive and detailed advice during all this time. Every discussion and every thorough correction contributes to my personal growth journey. Moreover, you've been such a great role model with your diligence, your meticulousness, and how you treat others. I appreciate all of these and they will keep inspiring me in the future.

Sébastien, thank you for all the things you have taught me during the countless communications. When I asked you questions about biology, you always explained patiently in a way that I, as a non-biology major student, could understand. You have also impacted me with your positive attitude and encouragement. Your willingness to help extends beyond research and has made a positive impact on various aspects of my life as a student, including the thoughtful help in so many tasks.

Laurent, thank you for your detailed explanations in response to my questions and your consistent encouragement with calm presence. When I was confused about the relationships between several concepts, you always identified where I was stuck and provided comprehensive answers with patience. Thank you for continually inspiring me when I found it challenging to make progress, your attitude had a profoundly positive influence on me.

I would also like to thank my co-advisors Jean-Charles Boisson and Luiz Angelo

Steffenel, who have followed this work from the beginning. Jean-Charles, thank you for patiently helping me fix bugs and providing advice on coding techniques. Luiz, thank you for offering valuable input during every meeting.

Aïcha, thank you for the work you did before my thesis, including sample preparation and image acquisition, which forms the substantial foundation of my thesis. Also, thank you for your encouragement throughout this time.

The days in Synchrotron SOLEIL in 2022 are unforgettable. I sincerely express my gratitude to my colleagues Zara, Hua, and Guillaume, for your significant and substantial help in the image acquisition process during the three days that were really tough for everyone, where we worked in day and night shifts. Thank you so much for your tremendous help. I would also thank Timm Weitkamp, researcher of ANATOMIX beamline, for all the guidance and support.

Camille, Kévin, Zara, Hayet, Pinar, and Manon, thank you for creating a very friendly environment and for your support. All of this has greatly motivated me every day. I cherish the time we spent together and am grateful for the lasting memories we've made. Your support and encouragement have made this experience truly special.

Jean-Marc and Manuel, I'm so glad to have worked with you, who are nice and also fun. You have offered me help and joy that are ingrained in my memory in this office. Thank you for all your support. I would also thank Stéphanie, Jessica, and Nicolas, thank you for your help and encouragement.

Ramzi and Juliet, I will never forget the time we spent together, thank you for your support and inspiration.

My sincere thanks also go to Ida Lenclume and Marielle Tur, who are always patient and friendly. Thank you for all the help you have provided in the administrative tasks.

I would like to extend my gratitude to Lolita Lebrun, the international mobility coordinator. Thank you for your indispensable help, addressing every minor issue that international students may encounter, and doing so with great patience.

Xia, when we met each other during the internship, I didn't know that this would become a friendship that would last for several stages of our lives. Thank you for all your support from the day we met.

To my mother and my father: I'm so lucky to be your beloved child, receiving your unconditional support every second of my life. The bond between us is one of the most important things that has supported me, not only during these three years but in every phase of my life.

Contents

List of Figures

List of Tables

Acronyms and abbreviations

CVD	Cardiovascular diseases
ECM	Extracellular matrix
ATA	Ascending aortic aorta
AA	Abdominal aorta
TAA	Thoracic aortic aneurysm
AAA	Abdominal aortic aneurysm
VSMC	Vascular smooth muscle cell
IEL	Internal elastin lamina
EEL	External elastin lamina
micro-CT	Computed microtomography
MAb	*Media-Adventitia* border
IVUS	Intravascular ultrasound
μCT	Microtomography
CT	Computerized axial tomography
PCI	Phase contrast imaging
RF	Radio-frequency
PBS	Phosphate-buffered saline
ROI	Region of interest
HE	Histogram equalization

CDF	Cumulative distribution function
AHE	Adaptive histogram equalization
CLAHE	Contrast-limited adaptive histogram equalization
BBHE	Brightness preserving bi-histogram equalization
DSIHE	Dualistic sub-image histogram equalization
MMBEBHE	Minimum mean brightness error bi-histogram equalization
RMSHE	Recursice mean-separate histogram equalization
DHE	Dynamic histogram equalization
BPDHE	Brightness preserving dynamic histogram equalization
ESIHE	Exposure based sub-image histogram equalization
AGCWD	Adaptive gamma correction with weighting distribution
WD	Weighting distribution
AGC	Aaptive gamma correction
MR	Magnetic resonance
SRG	Seeded region growing
SSL	Sequential sorted list
$\mathcal{ADAP}$	Adaptive homogeneity test
CB	Current boundary
IB	Internal boundary
ACB	Average contrast boundary
PCB	Peripheral contrast boundary
GLCM	Gray-level co-occurrence matrix
LBP	Local binary patterns
SIFT	Scale invariant feature transform

DoG	Difference of Gaussian
CNN	Convolutional neural network
ReLU	Rectified linear unit
T-CNN	Texture convolutional neural network
NL-means	Non-local means

Résumé

Introduction

Les maladies cardiovasculaires (MCV) sont la première cause de mortalité dans le monde. Par conséquent, des efforts ont été déployés pour détecter et prévenir ces maladies liées à l'âge, favorisées par le diabète ou l'insuffisance rénale chronique, qui se développent progressivement, affectant en particulier les grandes artères élastiques (athérosclérose, anévrisme) [1]. Le remodelage vasculaire est une conséquence du vieillissement moléculaire des composants de la paroi artérielle. Dans ce contexte, le vieillissement des fibres (collagène et élastine) de la matrice extracellulaire (MEC) est crucial. En effet, ces grandes fibres ont une longue durée de vie et ne connaissent pratiquement pas de renouvellement [2]. On considère donc qu'un individu vieillit avec son capital d'élastine et que toute altération de l'élastine est fondamentalement irréversible [3]. Notre étude porte sur l'imagerie du micro-remodelage de la MEC qui se produit au cours du vieillissement et/ou du développement de maladies vasculaires.

La paroi artérielle est constituée de trois couches, comme le montre la figure 1. La *tunica intima* (également appelée *tunica interna*) est la couche la plus interne. Il s'agit d'un simple épithélium pavimenteux entouré d'une membrane basale de tissu conjonctif avec des fibres élastiques [5]. Cette fine couche est en contact direct avec le sang et assure l'étanchéité de l'artère. Notons que cette couche ne sera pas visible dans notre étude. La *tunica media* est la couche intermédiaire. Elle est essentiellement musculaire et constitue généralement la couche la plus épaisse [5]. Elle est largement composée de couches concentriques de cellules musculaires lisses vasculaires au sein d'un MEC composé de fibres de collagène et d'élastine [6]. Le rôle des cellules musculaires est de réguler le diamètre des artères afin de diminuer/augmenter le débit en fonction des besoins des organes ou de s'adapter à une position (debout, couché, etc.). L'étude de la MEC de cette couche sera un point clé de ce travail. La *tunica externa* (ou *tunica adventitia*) est la couche la

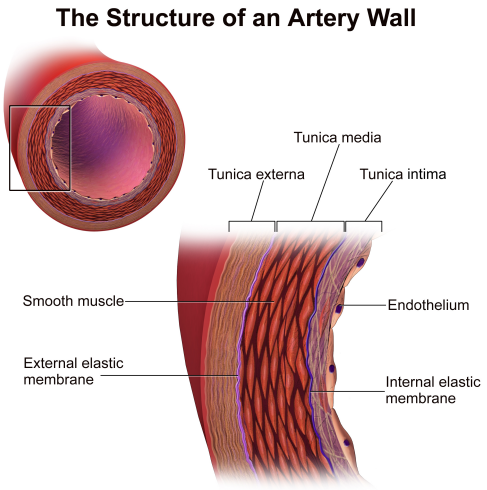


Figure 1: Structure d'une paroi artérielle [4]. Les composants comprenant l'endothélium, la membrane élastique interne, le muscle lisse et la membrane élastique externe forment les trois couches de la paroi artérielle : *tunica intima*, *tunica media* et *tunica externa*.

plus externe, qui attache le vaisseau au tissu environnant [5]. Ces trois couches reposent sur leur matrice extracellulaire ou y sont intégrées [7]. Dans la *tunica media*, les fibres élastiques sont disposées en feuilles fenêtrées, ou lames, et alignées circonférentiellement avec les cellules musculaires lisses vasculaires [7]. Notre travail a été réalisé sur les lames élastiques dans la *tunica media*, comme illustré par la figure 1. Nous avons étudié leur alterations pendant le vieillissement sans et avec l'occurrence des maladies vasculaires.

Un synchrotron est un type d'accélérateur circulaire [9]. Dans un tel système, les électrons se déplacent à une vitesse proche de celle de la lumière et sont maintenus dans un anneau de stockage par des champs électromagnétiques [10]. Les particules ainsi accélérées par le synchrotron ont une énergie élevée. Cette technique nous permet d'acquérir des images de très haute résolution. Les expériences ont lieu au niveau des lignes de faisceaux, qui sont des cabanes/laboratoires où le rayonnement est collecté, après avoir été sélectionné, focalisé et dirigé vers l'échantillon à étudier [11]. L'acquisition des nos images a été réalisée au synchrotron SOLEIL, sur ANATOMIX, une ligne de 200 mètres de long pour les techniques de tomographie plein champ. Les énergies des photons sont comprises entre 5 et 25 keV (environ 11 keV dans notre cas); la taille du faisceau peut atteindre 40×10 mm^2 [12].

Les artères de souris[1] ont été utilisées dans cette étude. Ces souris ont été

[1]Les procédures concernant les souris ont été réalisées en accord avec le Comité des sujets animaux de la Région Champagne-Ardenne (France).

Figure 2: Structures élastiques de la paroi artérielle. Cette image de microscopie montre la paroi artérielle d'une section d'artère de souris avec l'élastine marquée en bleu/noir. Les structures élastiques sont disposées comme des gaines concentriques autour de l'artère [8].

achetées chez Charles River (Lyon, France) et ont été mises en cage dans un environnement à température et humidité contrôlées avec un cycle lumière/obscurité de 12:12 heures. Les échantillons finaux avaient la forme de bâtonnet de paraffine d'une longueur de 40 mm et d'une largeur de 5 mm contenant le cœur et l'aorte. Les souris ont été principalement divisées en deux groupes : le premier groupe était celui des souris ayant subi un vieillissement normal, le second groupe était celui des souris présentant des pathologies afin d'observer les altérations pathologiques de leurs artères. La plupart des souris pathologiques utilisées pour faire des échantillons étaient des souris diabétiques. L'âge des souris varie de 6 mois à 12 mois.

Analyse des données originales et prétraitement

Notre étude a été réalisée sur 55 stacks d'images 3D, numérotées de 0 à 54. Chaque stack contient un volume de 2048^3 voxels (à l'exception du stack numéro 24, qui ne contient que 1354 images). Chaque voxel représente une taille réelle de $(0.65\mu m)^3$. Le traitement de l'image est effectué sur des volumes 3D et des images 2D de 2048^2 pixels. Ces images ont des caractéristiques communes. Par exemple, elles contiennent toutes du bruit, généré pendant l'acquisition de l'image. En outre, il existe des caractéristiques variant d'un stack à l'autre, qui n'apparaissent que dans certains stacks ou qui se situent à des niveaux différents. Par exemple, le nombre de bifurcations n'est pas le même dans chaque stack, et l'existence de matières à fort contraste attachées à la paroi interne dépend de l'échantillon.

Chaque stack de volume contient une partie de l'aorte et de ses tissus environnants, ainsi que des éléments introduits lors de la préparation de l'échantillon. Ces structures peuvent être observées dans des images 2D. La figure 3 est la première image du stack numéro 43 correspondant à une souris normale (contrôle). Sur cette coupe transversale, on peut voir la paroi artérielle, le tissu périvasculaire et l'agarose qui existe à l'intérieur et à l'extérieur de la lumière. Ce dernier a été introduit lors de la préparation des échantillons. Le champ visible, où le bruit existe, est le cercle tangent intérieur à partir duquel on peut voir la bordure circulaire. Tous les pixels ont la même intensité en dehors de ce champ.

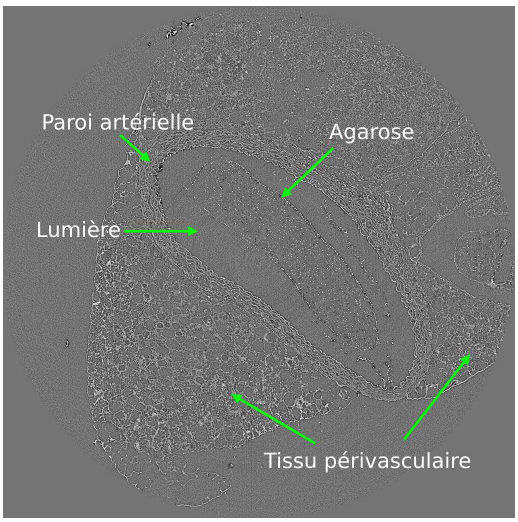


Figure 3: Une coupe du stack numéro 43 (contrôle). La lumière de l'aorte est entourée par la paroi artérielle, le tissu périvasculaire se trouve autour de la paroi artérielle (face extérieure) et l'agarose exogène est présente à la fois à l'intérieur et à l'extérieur de la paroi artérielle.

L'exemple de la figure 3 est une image qui a une structure "régulière" présentant peu de perturbations pour le processus de segmentation, elle montre une paroi artérielle correctement fermée avec un peu d'agarose détectable à l'intérieur de la lumière mais sans que celui-ci n'adhère à la paroi interne. La première couche de lames peut ainsi être vue clairement, ce qui facilite la détection de la limite interne des lames élastiques. Cependant, dans tous les stacks, nous avons une grande partie des images qui ne présentent pas des qualités optimales. En effet, diverses caractéristiques peuvent apparaître dans différents stacks ou dans des images individuelles du même stack. Différentes méthodes sont appliquées pour le prétraitement et les opérations suivantes.

Nous avons appliqué un ajustement du contraste sur les images ayant des contrastes instables. Dans le stack 20, le problème de saturation se pose dans les trames

160-163, où la différence d'intensité entre la région des lamelles et la région du lumen est presque nulle. On a utilisé l'image la plus proche considérée comme normale (pour l'image 160, l'image de référence est l'image 164) pour la comparaison. Une dilatation et une érosion (avec une taille d'élément structurant de 7×7) ont été appliquées à l'image 164 afin de trouver les valeurs voisines les plus lumineuses et les plus sombres pour chaque pixel. On a ensuite comparé ces images opérées avec l'image à ajuster, en mesurant la différence entre chaque paire de pixels de même coordonnée sur l'image dilatée/érodée et l'image à ajuster. Plus cette différence est grande, plus le pixel correspondant doit être ajusté. La valeur ajustée est calculée à partir des valeurs initiales et d'une image de la trame de référence lissée par le filtre gaussien. On a aussi appliqué des opérations spécifiques pour les pixels ayant des valeurs extrêmes. Avec cette image principalement ajustée, on applique une concordance d'histogramme avec la trame de référence. Pour les autres images qui ont un faible contraste, on les a ajustées en faisant correspondre l'histogramme avec un histogramme de référence généré par les 155 premières images. La figure 4 montre deux frames avant et après des ajustements.

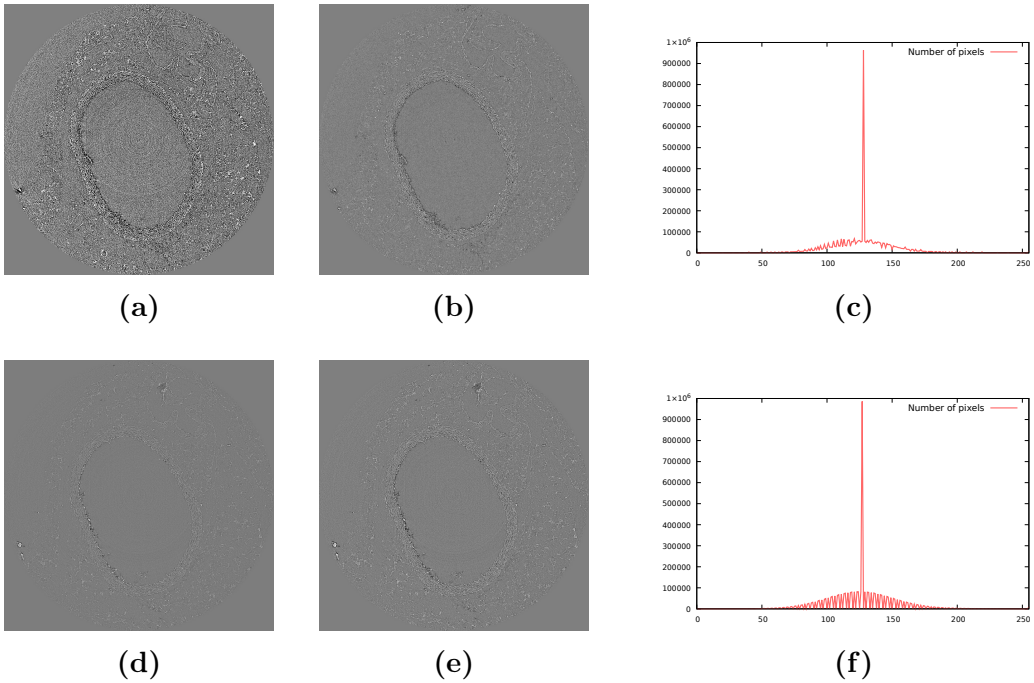


Figure 4: (a) Trame 160 (saturée) et (d) Trame 200 (faible contraste) du stack 20 avant l'ajustement. (b) et (e) sont les résultats correspondants après ajustement. (c) et (f) sont les histogrammes des images ajustées.

Ensuite, nous avons appliqué un ajustement global à toutes les stacks. Un filtre gaussien 3D avec $\sigma = 1$ a été appliqué sur chaque pile avant cet ajustement pour

éliminer une partie du bruit. Le contraste local de chaque pixel p sur une image est calculé par :

$$C(p) = I_{dilate}(p) - I_{erode}(p) \tag{1}$$

qui est la différence entre l'image dilatée et l'image érodée. Cette valeur est normalisée à $[0,\ 1]$ et noté C_{norm}. Un coefficient γ est calculé par :

$$\gamma = \gamma_0 - 0,05 \cdot \frac{std_{adjusted} - std_{stack}}{std_{adjusted}} \tag{2}$$

où $\gamma_0 = 0,1$ et std_{stack} est la moyenne des écarts-types des trames sélectionnées parmi les 128 trames d'un stack. Un indicateur de contraste est ensuite calculé :

$$C'_{norm}(p) = \begin{cases} 0, & si\ C_{norm}(p) < 0,25 \\ C^{\gamma}_{norm}(p), & sinon \end{cases} \tag{3}$$

L'intensité ajustée $I_{adjusted}(p)$ est donnée par :

$$I_{adjusted}(p) = (I(p) - \bar{I}) \cdot \left(\left(\frac{std_{adjust}}{std_{stack}} - 1 \right) \cdot C'_{norm_g}(p) + 1 \right) + \bar{I} \tag{4}$$

où $\bar{I}$ est la moyenne des intensités des images sélectionnées parmi les 128 trames d'un stack.

La binarisation d'images a été réalisée par la méthode d'Otsu [13] et le calcul du contraste. Nous avons d'abord défini un seuil T_{OTSU} par la méthode d'Otsu, puis nous avons calculé un nouveau seuil T'_{OTSU} par :

$$T'_{OTSU} = \frac{T_{OTSU} + m_{OTSU}}{2} \tag{5}$$

La valeur du pixel p sur la nouvelle image est :

$$I_{bin}(p) = \begin{cases} 255, & si\ I_{init}(p) > T_{OTSU}\ et\ I_{init}(p) > T'_{OTSU} - \frac{255 - C(p)}{2} \\ 0, & sinon \end{cases} \tag{6}$$

où $C(p)$ est le contraste calculé par Equation 1, et I_{init} est l'image initiale.

Avec les images binaires, nous avons regroupé les 32 images consécutives en un seul groupe et trouvé toutes les composantes connexes en 3D. Dans chaque groupe, la plus grande composante a été conservée et le reste des voxels a été éliminé. Cette opération nous a permis d'éliminer une grande partie d'impureté dans la lumière.

Segmentation de la lumière

Pour extraire la lumière, un nettoyage a été d'abord effectué sur les images obtenues par la sélection des composantes connexes comme présenté dans la précédente section. Pour les stacks présentant des bulles d'air, nous avons calculé les contrates locaux et éliminé les pixels d'intensités extrêmes et de hauts contrastes. Pour les données où l'agarose n'a pas été bien éliminée par la sélection des composantes connexes et où la forme de l'espace vide à l'intérieur de la paroi vasculaire a été affectée par la présence d'impureté, nous avons appliqué un processus de nettoyage de la lumière. Une sphère glissante d'un rayon R a été utilisée pour sélectionner les espaces 3D ayant suffisamment de voxels à l'intérieur de la sphère. Nous avons d'abord calculé le volume d'une sphère $V_R = \frac{4\pi R^3}{3}$ et préservé le pixel de son centre O si et seulement si :

$$Card(S_{O,R}) > \frac{V_R}{8} \tag{7}$$

où S_{O_R} est l'ensemble des pixels blancs contenus dans la sphère :

$$S_{O,R} = \{p \mid I_{Binary_{3D}}(p) > 0,\ \|(p - O)\|_2 \leq R\} \tag{8}$$

où $I_{Binary_{3D}}$ est l'image binaire 3D obtenue par l'étape de prétraitement. Pour réduire le coût calculatoire, les centres ont été testés tous les R pixels dans les directions X et Y, et tous les $R/2$ pixels dans la direction Z. X et Y sont les axes dans une trame, et Z est l'axe sur laquelle les trames consécutive sont alignées. Pour tirer parti de la continuité dans l'axe Z, pour $S_{O_{z_1},R}$, qui sont des centres conservés à $z = z_1$, les centres candidats pour $z_2 = z_1 + R/2$ à vérifier sont ceux qui sont à une distance de R de $S_{O_{z_1},R}$ dans la direction X ou Y. Ainsi, lorsque $S_{O_{z_1},R}$ est connu, les centres préservés pour $z_2 = z_1 + R/2$ sont ceux qui satisfont :

$$Card(S_{O_{z_2},R}) > \frac{V_R}{8} \tag{9}$$

où :

$$\begin{aligned} S_{O_{z_2},R} = \{p \mid I_{Binary_{3D}}(p) > 0,\ \|(p - O_{z_2})\|_2 \leq R, \exists O_{z_1} \in S_{O_{z_1},R} \\ s.t.\ x_{O_{z_2}} = x_{O_{z_1}} \pm R \ or\ y_{O_{z_2}} = y_{O_{z_1}} \pm R\} \end{aligned} \tag{10}$$

Nous avons noté les centres préservés et généré des volumes sphériques de rayon R centrés par ces voxels. Tous les voxels générés définissent une forme générale autour

de la lumière.

À partir de ce résult de nettoyage (noté $I_{Artery_{3D}}$), nous avons effectué une extraction de la lumière. Chaque fois que le nombre de pixels contenus dans l'espace entre les deux sphères de rayons R_1 et R_2 est vérifié ($R_2 > R_1$), nous avons conservé des centres de sphères qui satisfont :

$$Card(S_{O_{R_1,R_2}}) < \frac{V_{O_{R_1,R_2}}}{16} \tag{11}$$

où O est le centre de la sphère, $V_{O_{R_1,R_2}}$ est le nombre de pixels si l'espace entre les deux rayons est rempli de pixels, et $S_{O_{R_1,R_2}}$ est l'ensemble des pixels blancs contenus dans cet espace :

$$S_{O_{R_1,R_2}} = \{p \mid I_{Artery_{3D}}(p) > 0,\ R_1 < \|(p-O)\|_2 \leq R_2\} \tag{12}$$

Nous avons appliqué cette méthode sur la trame $z = R_2$ pour une initialization, puis nous avons utilisé la continuité entre frames pour générer les centres de sphère du reste du stack. L'ensemble des centres connexes correspondant à la lumière a été manuellement sélectionné et gardé, le reste a été supprimé. Pour une trame au voisinage d'une trame où les centres ont été dèjà identifiés, nous n'avons vérifié que les pixels entre la région dilatée et erodée des centres de la trame déjà calculée. Pour réduire le coût calculatoire, les centres ont été calculés toutes les $R_2/2$ trames consécutives et tous les $R_2/4$ pixels dans les directions X et Y. Les centres de sphères dans chaque stack ont ainsi engendré les sphères 3D, qui ont composé un espace de la lumière extraite.

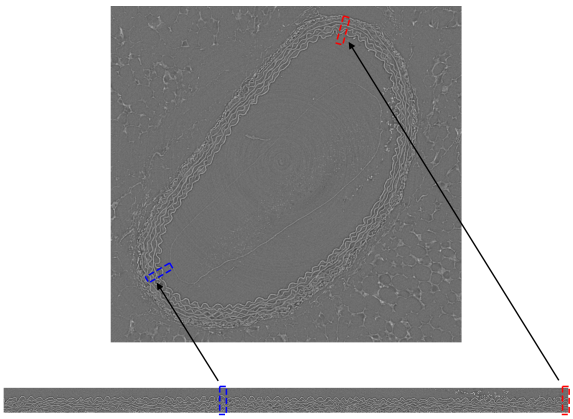


Figure 5: Une image en niveaux de gris lissée par un débruitage gaussien dans le stack numéro 6 et la paroi artérielle déroulée correspondante.

Disposant désormais de la forme générale de la lumière, nous avons extrait le

contour externe de la lumière chaque 32 trames, et déroulé chaque groupe de 32 trames avec le contour de la 16ème trame de ce groupe. La figure 5 illustre une image déroulée. Les pixels ont été orientés par colonne, en utilisant le masque et le contour de la lumière de la trame centrale de son groupe. Une marge a été ajoutée en haut dans chaque image pour ne pas perdre les information de la première lame élastique.

Segmentation des lames élastiques

Dans les images des parois artérielles, les lames élastiques sont positionées dans un ordre, avec celles qui sont plus proches de la lumière étant situées à des hauteurs supérieures. Dans les étapes de cette section, nous avons effectué deux types de segmentations. Il s'agit d'une segmentation de la première lame élastique et d'une segmentation de la *tunica media* où se trouve toutes les lames élastiques.

Segmentation de la première lame élastique

Un débruitage de type NL-means [14] [15] a été d'abord appliqué sur l'image déroulée. Une binarisation basée sur la méthode d'Otsu et le calcul du contraste ont été ensuite effectués. Nous avons aussi déroulé les images binaires pour éliminer les impuretés supprimées dans le processus de nettoyage de la lumière. La figure 6 illustre des images déroulées en niveaux de gris et binarisées à l'aide des images après une élimination de l'agarose.

Sur ces images binaires, nous avons effectué l'extraction de la première lame elastique. Les images ont été segmentées par groupe de 32 trames. Une croissance de région en 3D a été d'abord appliquée. Un voxel v dans un sous-stack peut être localisé par (z_v, y_v, x_v). Nous avons appliqué le processus de croissance de régions aux segments de l'extrême gauche à l'extrême droite de la sous-pile, la taille de chaque segment est de $(32, 128, 100)$, et le pas de décalage est de 25, ce qui signifie que chaque voxel est impliqué dans 4 opérations. La figure 7 illustre le résultat d'une croissance de région par segments sur un sous-stack de 32 trames. Une sélection de la plus grande région connexe a été appliquée après ce processus.

Le problème principal dans cette application de croissance de région est la fuite des voxels, comme il existe des trous sur la première (et les autres) lame(s) élastique(s) qui permettent à la croissance de région d'entrer dans l'espace sous la première lame. Pour gérer cette mauvaise croissance de région, nous avons appliqué une sélection pour choisir les régions à garder et les régions à éliminer. Pour obtenir les

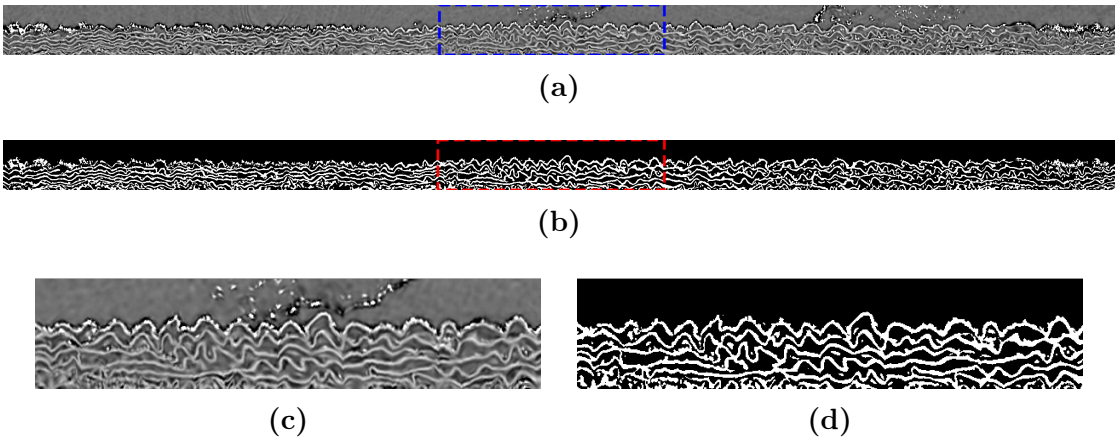


Figure 6: (a) Une image débruitée déroulée après un rehaussement du contraste. (b) Image binarisée de (a). Les impuretés au-dessus de la première lame élastique dans (a) n'apparaissent pas dans le résultat binarisé (b) parce qu'une contrainte de l'image binaire dépliée après le processus de nettoyage de la lumière est ajoutée. (c) et (d) sont des segments agrandis correspondant aux régions marquées en (a) et (b).

régions séparées R_i, nous recherchons les connexions en traçant des lignes verticales à partir du haut des espaces 3D. Dès qu'une ligne touche des pixels blancs de l'image binaire initiale des lamelles élastiques, elle cesse de descendre. Toutes ces lignes forment des espaces directement liés au sommet, et si nous les supprimons des régions développées, le reste de l'image sera constitué de pixels séparés qui se trouvent sous les lamelles. On note la région engendrée par croissance de région (après sélection de la plus grande composante connexe) R_{rg}, l'espace composé de voxels directement connectés au haut R_{top}, l'espace occupé par les lamelles élastiques (et les autres tissus qui sont blancs dans l'image binarisée) R_{lame}, et l'espace restant (régions candidates) après avoir retiré R_{top} de R_{rg} est noté $R_{rg_{can}} = \{v \mid v \in R_{rg}, v \notin R_{top}\}$ et $\bigcup_i R^i_{rg_{can}} = R_{rg_{can}}$, i est l'indice de la région candidate. Les connexions de R^i_{con} sont définies comme les points dans R_{top} et adjacents à $R^i_{rg_{can}}$:

$$R^i_{con} = \{v \mid v \in R_{top},\ \exists v^{'} \in \mathcal{N}(v)\ s.t.\ v^{'} \in R^i_{rg_{can}}\} \tag{13}$$

où $\mathcal{N}(v)$ représente le 26 voisinage du voxel v. Pour une région candidate $R^i_{rg_{can}}$, la taille de connexion $c(R^i_{rg_{can}})$ est le nombre de voxels qui la relient :

$$c(R^i_{rg_{can}}) = \left|\{v \mid v \in R^i_{con}\}\right| \tag{14}$$

Les poids des voxels de cette région sont calculés par propagation à partir de ses connexions. Pour chaque $R^i_{rg_{can}}$, nous avons calculé une matrice 3D $W^i_{rg_{can}}$ qui

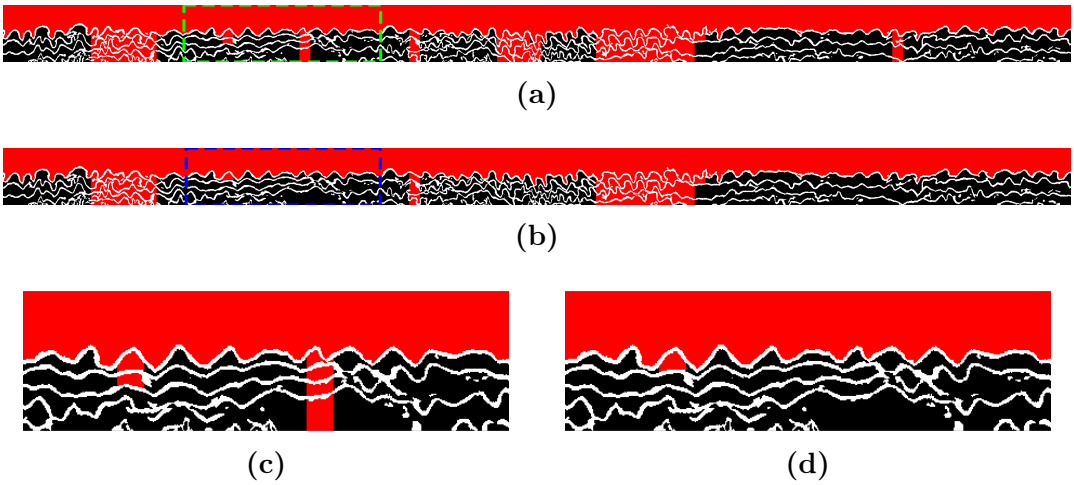


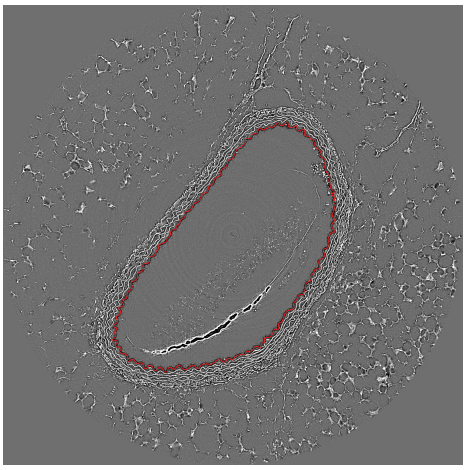
Figure 7: Projection 2D sur une image de la croissance de la région par segment en 3D (a) et les résultats après une sélection de la plus grande composante connexe en 3D (b). Les pixels blancs proviennent de l'image initiale des lamelles élastiques auxquelles l'opération est appliquée. Les pixels rouges sont dans des régions ajoutées par la croissance de région. (c) et (d) sont des segments agrandis correspondant aux régions marquées en (a) et (b).

contient le poids de chaque voxel dans $R^i_{rg_{can}}$. La somme des poids est $w(R^i_{rg_{can}})$. Dans chaque composante, les voxels frontaliers qui sont adjacents aux connexions se voient attribuer la valeur 1. Ensuite, les voxels sont examinés et leurs valeurs attribuées sont mises à jour avec les étapes qui s'incrémentent. Une fois que tous les voxels sont munis d'une valeur, le poids de la composante est la somme de toutes ces étapes. Avec $c(R^i_{rg_{can}})$ et $w(R^i_{rg_{can}})$, un rapport $r_{w/c}$ peut être calculé comme suit :

$$r^i_{w/c} = \frac{w(R^i_{rg_{can}})}{c(R^i_{rg_{can}})} \tag{15}$$

On fixe un seuil $Th_{r^i_{w/c}}$ (20 dans notre implémentation) pour rejeter les régions indésirables. Pour un voxel v dans $R^i_{rg_{can}}$, si $r^i_{w/c} < Th_{r^i_{w/c}}$, v est préservé, sinon il est rejeté. En rejetant les espaces ajoutés par erreur, nous avons obtenu un espace au-dessus de la première lame et un espace en dessous de la première lame. Les voxels entre ces deux espaces et ayant des valeurs positives (pixels blancs) dans l'image binaire initiale sont considérés comme étant situés sur la première lame élastique. Cette lame a été extraite par une sphère glissante, et un post-traitement a été appliqué pour éliminer des voxels des autres lames extraits par erreur. Cette première lame a été ensuite restaurée dans sa forme initiale de manière inverse de son déroulement, suivi d'un post-traitement pour remplir des trous générés pendant

cette restauration basée sur les rotations des pixels. La figure 8 illustre des résultats en 2D et en 3D de la segmentation de la première lame élastique.



(a) Superposition de la lame extraite sur une image initiale (après un ajustement du contraste).

(b) Reconstruction 3D de la première lame extraite avec 512 trames.

Figure 8: Lame élastique restaurée après post-traitement, illustrée en 2D et en 3D.

Classification des textures pour trouver la frontière *media-adventitia*

Par rapport à la partie interne de la *tunica media*, la partie externe de cette structure est plus complexe car elle est en contact avec le *tunica adventitia* qui a des densités et des intensités similaires à celles du *tunica media* et l'épaisseur de la *media* varie dans les stacks et dans les régions d'un même stack. La frontière entre ces deux couches peut être repérée à l'œil nu en distinguant la différence de texture. Cela nous a incités à utiliser des méthodes d'analyse de texture pour séparer ces deux structures. Un réseau siamois [16] [17] est capable de distinguer une paire d'entrées en générant des paires de descripteurs entre lesquels la distance indique la similarité des entrées.

Avec les résultat de l'extraction de la première lame élatisque, nous avons été en mesure de dérouler la paroi arterielle en utilisant ce contour et l'espace entouré par ce contour. Cela nous a permis de déplier l'artère de manière plus précise, en mettant la première lame élastique tout en haut de l'image, comme illustré par la figure 9.

Nous avons utilisé un réseau siamois et des patchs ayant des textures similaires et non similaires pour réaliser un entraînement pour chaque stack d'images. Une

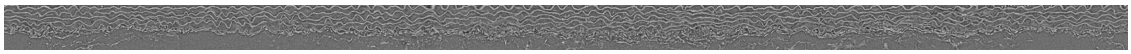


Figure 9: Image déroulée en niveaux de gris. La première lame élastique se trouve en haut de l'image.

image déroulée débruitée a été découpée en patchs de taille 128 par une fenêtre mobile glissant de la gauche vers la droite de l'image avec un pas de 64. Dans ces patchs, la partie supérieure est principalement remplie par des lamelles élastiques et la partie inférieure contient des tissus de la *tunica adventitia*. Nous avons découpé les sous-patchs comme indiqué dans la figure 10 : les régions centrales des parties supérieure et inférieure de chaque patch ont été découpées et combinées en tant que paires similaires et dissemblables pour entrer dans le réseau siamois.

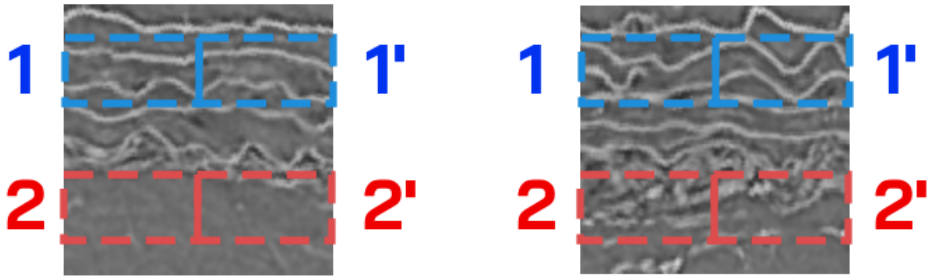


Figure 10: Deux patchs avec les 4 sous-patchs marqués. Les sous-patchs 1 et $1^{'}$ représentent la texture de la *tunica media*; les sous-patchs 2 et $2^{'}$ représentent la texture de ce qui n'est pas la *tunica media* ; il peut s'agir de la *tunica adventitia* ou de la structure qui se trouve à l'extérieur. Les sous-patchs sont découpés à partir du milieu de la moitié supérieure et de la moitié inférieure du patch entier. Le patch de gauche contient moins de texture de la *tunica adventitia* que celui de droite.

La figure 11 montre l'architecture du réseau. L'entrée est un sous-patch de taille $(H, W) = (32, 64)$, les couches $ConvRelu$ dans la figure sont des couches convolutives suivies d'une fonction ReLU, $ConvRelu1$ à $ConvRelu6$ sont des groupes composés de deux structures « couche convolutive-ReLU » avec des noyaux de convolution de taille 3×3; les nombres de canaux sont 64-64-64-64-128-128-128-128-256-256-256-256. La couche $ConvRelu7$ est une couche convolutive avec un noyau de taille 1×1 et 256 canaux suivis d'une fonction ReLU. Chacune des couches $ConvRelu1$ à $ConvRelu6$ est suivie d'une mise en commun maximale, la dernière mise en commun étant appliquée dans la direction X. La dernière couche est également une couche entièrement convolutive avec un noyau de taille 1×1 et elle produit un descripteur de taille 16.

Les 4 sous-patchs P_{ij} obtenus à partir d'un patch P entrent dans le même réseau.

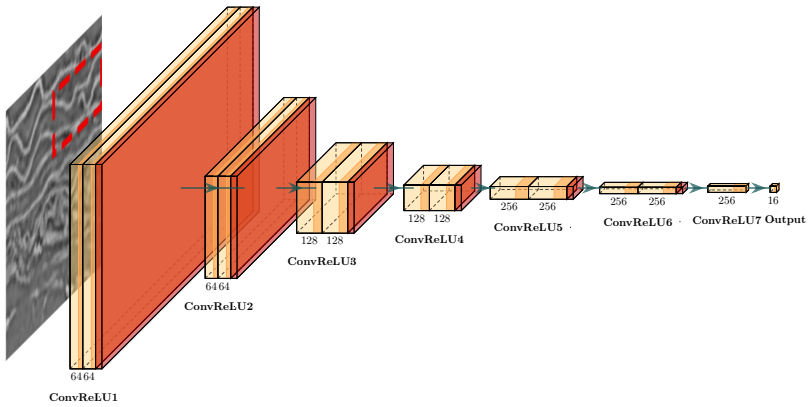


Figure 11: Architecture du réseau. L'entrée est de taille $(H, W) = (32, 64)$; la sortie est un vecteur de taille 16.

Cela conduit à 4 descripteurs de sortie (normalisés) d_{ij} de longueur 16, combinés en tant que 6 paires de descripteurs (distincts). La perte associée à une paire de sous-patchs est définie comme suit :

$$\mathcal{L}(d_{ij}, d_{kl}) = \alpha.\delta_{ik}.\Delta^T.\Delta \ + \beta.(1 - \delta_{ik}).[\max\{0, m_p - (\Delta^T.\Delta)^{1/2}\}]^2 \tag{16}$$

où $\Delta = d_{ij} - d_{kl}$, δ est le symbole de Kronecker et $\alpha = 2$, $\beta = 1$, $m_p = 2$. Il s'agit d'une perte de contraste. Avec le descripteur normalisé à une distance euclidienne de 1, lorsque $\delta = 1$, ce qui signifie que les entrées sont des patchs similaires, le produit $\Delta^T.\Delta$ donne un résultat plus petit de la fonction de perte lorsque les descripteurs sont proches les uns des autres ; lorsque $\delta = 0$ avec des entrées non similaires, $\Delta^T.\Delta$ donne un résultat plus petit lorsque les descripteurs ne sont pas similaires. Dans notre cas, avec $m_p = 2$ et d_{ij} ainsi que d_{kl} normalisés à la taille 1, ce terme est égal à $m_p - (\Delta^T.\Delta)^{1/2}$.

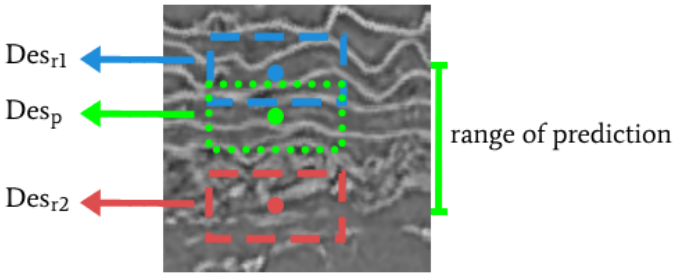


Figure 12: Méthode de prédiction au niveau du pixel. Les Des_p générés par le sous-patch centré sur le pixel vert sont comparés à ceux générés par les deux patchs de référence situés au-dessus et au-dessous. L'intervalle de prédiction est $y \in (\frac{L}{4}, \frac{3L}{4})$.

Un stack de 2048 images génère environ 100000 patchs. 10 époques ont été effectuées sur chaque pile pour générer son propre modèle. À chaque époque, 80% des patchs ont été utilisés pour l'apprentissage et le reste pour la validation. Nous avons évalué la performance d'un modèle directement par prédiction au niveau du pixel. La méthode est illustrée dans la figure 12. Pour une zone découpée dans une image déroulée en niveaux de gris, la frontière *media-adventitia* est supposée se situer entre $y = \frac{L}{4}$ et $y = \frac{3L}{4}$ dans une zone de taille $L \times L$ ($L = 128$ dans notre cas). Pour prédire la classification du pixel p à (y_p, x_p), deux sous-patchs de référence de taille $(H, W) = (\frac{L}{4}, \frac{L}{2})$ (H : hauteur, W : largeur) avec des centres à $(\frac{L}{4}, x_p)$ et $(\frac{3L}{4}, x_p)$ sont découpés afin de générer deux descripteurs représentant les textures *media* et les textures non *media*. Les pixels p avec $y_p \in (\frac{L}{4}, \frac{3L}{4})$ et $x_p \in (\frac{L}{4}, \frac{3L}{4})$ sont prédits en comparant le descripteur Des_p généré par le sous-patch $P_{(y_p,x_p)}$ centré sur (y_p, x_p) avec les deux descripteurs Des_{r1} et Des_{r2} générés par les sous-patchs de référence $P_{sub}(\frac{L}{4}, x_p)$ et $P_{sub}(\frac{3L}{4}, x_p)$ respectivement, selon la règle suivante :

$$T_p = \begin{cases} media & si \ \|Des_p - Des_{r1}\| < \|Des_p - Des_{r2}\| \\ non\ media & sinon \end{cases} \tag{17}$$

où T_p est la classe de texture à laquelle appartient p.

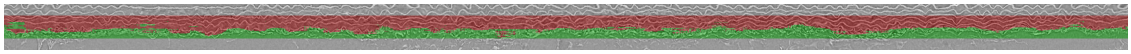


Figure 13: Résultat de la prédiction sur une paroi artérielle déroulée. Les pixels rouges correspondent à la texture *media* et les pixels verts à la texture non *media*.

La figure 13 illustre une prédiction sur une image entière. Nous avons choisi des points équidistants sur le bord (après des post-traitements) entre ces deux textures et projeté ces points sur l'image restaurée de taille 2048×2048. Une courbe fermée a été ensuite ajustée pour générer une frontière continue entre les deux textures. Un résultat de la frontière ajustée est illustré dans la figure 14.

Analyse des lames élastiques

Deux analyses ont été effectuées sur les lames élastiques : une analyse de courbure en 3D sur les premières lames élastiques et une analyse d'ondulation en 2D sur les fragments de toutes les lames élastiques.

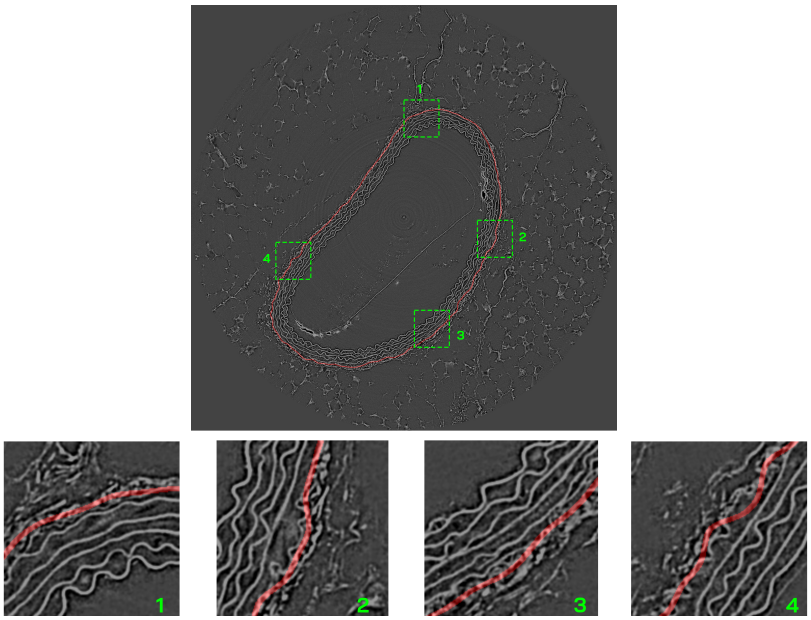


Figure 14: Frontière *media-adventitia* (rouge) ajustée par une courbe à partir des points sélectionnés toutes les 25 colonnes dans l'image déreoulée.

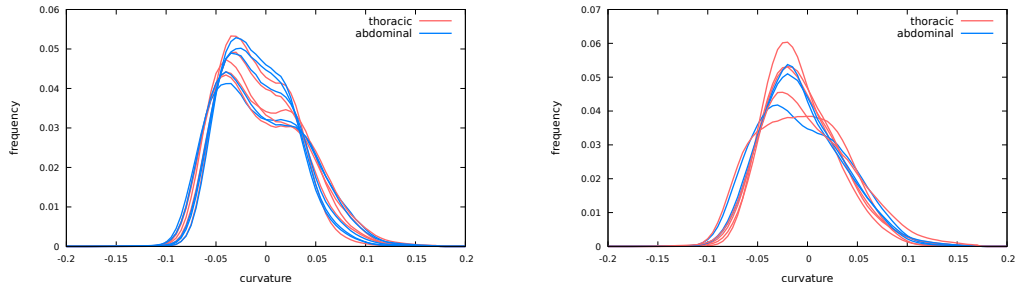
Analyse de courbure de la première lame élastique

Cette analyse a été faite sur la première lame en 3D, en utilisant une estimation de courbure [18] [19] [20] par :

$$H_r(X, x) = \frac{8}{3r} - \frac{4V_r(x)}{\pi r^4} \tag{18}$$

où X est la surface où se trouve x, r est le rayon de la sphère centrée sur x et $V_r(x)$ est le volume à l'intérieur de l'espace fermé par la surface et à l'intérieur de la sphère.

Comme la première lamelle est attachée à des impuretés dans certains stacks, nous n'avons utilisé que celles qui ont une première lamelle relativement propre dont les courbes extraites peuvent montrer précisément la forme de la lamelle sans introduire d'erreur significative. 18 stacks (10 diabétiques et 8 normaux) ont été utilisés pour cette analyse. Nous avons calculé les courbures des points sur les contours du masque qui existaient également sur la première lamelle extraite puisque les masques avaient des contours fermés sur les bifurcations où il n'y a pas de lamelle extraite. Nous avons rejeté les images pour lesquelles les formes des lamelles n'étaient pas applicables dans les images originales ou pour lesquelles les contours du masque n'étaient pas suffisamment précis.



(a) Stacks diabétiques. **(b)** Stacks normaux.

Figure 15: Superpositions des histogrammes de courbure 3D des stacks diabétiques (a) et normaux (b). Les stacks thoraciques et abdominaux se distinguent respectivement par des courbes orange et bleues.

Après plusieurs expériences, nous avons choisi $r = 15$ ($9,75\ \mu m$ en taille réelle) comme rayon de la sphère et avons généré un histogramme de la fréquence de la gamme de courbures pour chaque stack. $V_r(x)$ est le nombre de voxels à l'intérieur de la sphère et de l'espace clos, la taille de l'intervalle étant fixée à 0,005. La figure 15 montre la superposition de l'histogramme de chaque groupe. On peut observer que dans les stacks diabétiques, la première lame élastique a deux distributions de courbure dominantes dans des intervalles autour d'une valeur négative et d'une valeur positive respectivement. Dans les stacks normaux, la distribution autour de la valeur positive est relativement proche de celle autour de la valeur négative où sa hauteur n'est pas aussi importante que dans les stacks diabétiques.

Analyse d'ondulation des fragments

De la deuxième lame élastique à la lame la plus externe, les structures ne sont pas toujours complètes que la première couche qui peut, pour sa part, former un contour fermé. Nous avons appliqué une analyse sur les fragments de l'ensemble des lames élastiques. Les images à analyser sont les lames squelettisées. Un prétraitement a été effectué pour éliminer les bifurcations courtes sur les fragments, puis une mesure a été appliquée pour estimer leur ondulation. La mesure est définie comme :

$$\rho = \frac{L_{Euc}}{L_{geo}} \tag{19}$$

où L_{Euc} est la longueur euclidienne définie par :

$$L_{Euc} = \sum_{0 \leq j < L_{geo}/\sigma,\ (j+1)\sigma \leq L_{geo}} \| V_{(j+1)_\sigma} - V_{j_\sigma} \|_2 \tag{20}$$

et L_{geo} est la longueur géodésique définie par :

$$L_{geo} = \sum_{0 \leq i < k} \|V_{i+1} - V_i\|_2 \tag{21}$$

Dans les deux équations ci-dessus, k est le nombre de pixels et V_i sont les sommets de la courbe.

La figure 16 est une illustration de la mesure. Les pixels blancs représentent la courbe initiale squelettisée à laquelle ont été ajoutés des points rouges. Ces pixels rouges sous-échantillonnent des points dont la signification sera expliquée ultérieurement. Les lignes vertes forment un polygone ouvert ajusté pour calculer la L_{Euc}. Cette longueur est calculée en fonction de L_{geo} puisque la sélection des sommets V_i dépend de L_{geo}.

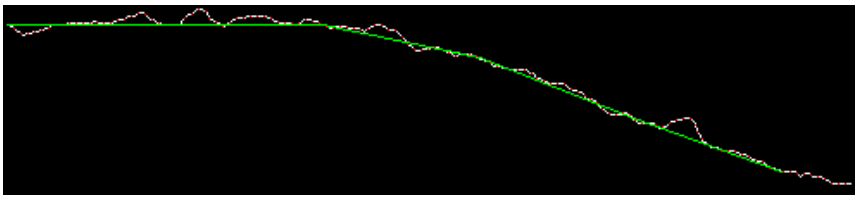


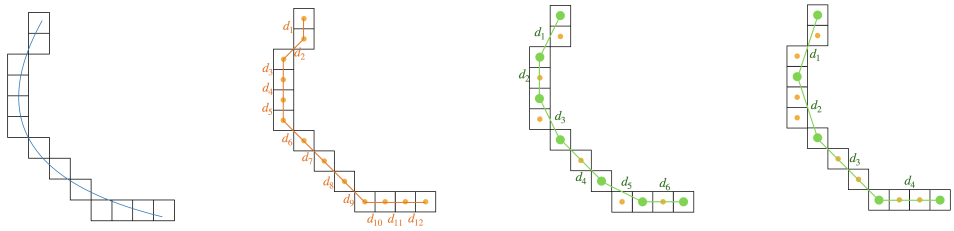
Figure 16: Illustration de la mesure de l'ondulation. Une courbe initiale est représentée en blanc. Les pixels rouges sont des sommets sous-échantillonnés. Un polygone ouvert (en vert) est généré en connectant un sous-ensemble de sommets.

L_{geo} est la distance géodésique représentant la courbe réelle. Comme le montre la figure 17, la longueur d'une courbe peut être calculée en additionnant les distances consécutives entre les pixels :

$$L = \sum_{i=1}^{(N-1)/s} d_i \tag{22}$$

où N est le nombre de pixels sur la courbe, s est le pas de sélection des pixels comme points d'extrémité des lignes consécutives, d_i est la distance euclidienne entre les pixels p_{is} et $p_{(i-1)s}$. Il existe une erreur entre la distance euclidienne ainsi calculée et la longueur réelle de la courbe; cette erreur dépend du choix du paramètre s. Pour choisir une valeur s qui génère la plus petite erreur, nous avons utilisé des segments de cercle de différents rayons pour calculer leurs périmètres théoriques et la longueur estimée avec différents s. Après plusieur expériences, nous avons choisi $s = 4$ comme paramètre d'échantillonnage. Après avoir choisi $s = 4$ pour le calcul de la distance géodésique, nous avons choisi des sommets sous-échantillonnés pour accumuler les distances entre eux afin d'obtenir la distance euclidienne L_{Euc}.

Nous avons suréchantillonné les pixels sélectionnés afin d'obtenir le même nombre de sommets que les pixels d'origine.



(a) **(b)** **(c)** **(d)**

Figure 17: Illustration du comptage de la longueur d'une courbe. (a) illustre la relation entre une courbe et sa visualisation au niveau des pixels ; (b), (c) et (d) sont leurs processus de comptage de la longueur avec différents pas s dans l'équation 22.

Les courbes sont de différentes longueurs. Nous les avons classées en 3 types en fonction de leur longueur géodésique :

$$type\ de\ courbe = \begin{cases} courte, & L_{geo} \leq 100 \\ moyenne, & 100 \leq L < 300 \\ longue, & L > 300 \end{cases} \tag{23}$$

Pour les courbes courtes, nous avons sélectionné ses deux extrémités, comme début et fin du polygone. Pour les courbes de longueur moyenne, nous sélectionnons ses deux extrémités et les sommets de l'échantillon supérieur dont la distance cumulée est la plus proche de $L_{geo}/2$. Pour les courbes longues, nous choisissons un sommet tous les 100 pixels de distance cumulée pour former le polygone.

Nous avons appliqué cette analyse deux fois, sur une base de données composée de 21 stacks (lames segmentées par une méthode basée sur la méthode d'Otsu avant le développement de la méthode de segmentation décrites dans les sections précédentes) et les lames segmentées par la méthode dans ce travail (51 stacks).

Il y a 10 stacks de souris normales et 11 stacks de souris diabétiques dans la première base de données. La p-valeur du test U de Mann-Whitney est de 0,0035 $<$ 0,05. La différence entre les valeurs moyennes $\bar{\rho}_{diabetic} - \bar{\rho}_{control}$ est égale à 0,0662. Ces résultats permettent de conclure que les lamelles élastiques des souris diabétiques sont globalement plus lisses que celles des souris contrôles (normales).

Dans la deuxième base de données, 23 sont des échantillons normaux et 28 sont des échantillons diabétiques. Les moyennes des deux dernières lignes ont été calculées sans les cinq échantillons de 3 mois et 12 mois. Les statistiques ont été effectuées sur les 46 échantillons restants, 18 normaux et 28 diabétiques. L'ondulation

moyenne des échantillons diabétiques est supérieure à celle des échantillons normaux, aussi bien dans la partie thoracique que dans la partie abdominale. La p-valeur des échantillons diabétiques et normaux dans la partie abdominale est de 0,1158; celle de la partie thoracique est de 0,00325, et celle des deux parties est de 0,000976, ce qui montre une différence significative entre ces deux groupes. Cette différence provient principalement des échantillons thoraciques. L'une des raisons est que le nombre d'échantillons abdominaux est inférieur à celui des échantillons thoraciques, et que ce nombre n'est probablement pas suffisant pour révéler la différence entre les deux types d'échantillons. D'autre part, cela peut être dû à la différence d'impact de la pression sur la partie thoracique et la partie abdominale. L'artère thoracique est plus exposée à la pression que l'artère abdominale, ce qui fait que son élasticité est plus affectée par certains facteurs.

Conclusion

Les maladies cardiovasculaires sont l'une des principales causes de décès dans le monde. Il est donc important d'intervenir sur ces maladies avant qu'elles ne deviennent graves et donc développer des méthodes de détection de ces pathologies est d'un intérêt sociétal majeur. Ces maladies sont étroitement liées à des troubles métaboliques tels que le diabète. Dans notre travail, des images artérielles de souris saines et diabétiques ont été obtenues par microtomographie synchrotron à rayons X avec une taille de 2048^3 voxels par stack, ce qui nous a permis de visualiser des structures très fines de la paroi artérielle de la souris. Le travail peut être divisé en deux parties principales : la segmentation des lames élastiques et l'analyse sur ces lames élastiques segmentées. La première analyse a été réalisée sur la première lame élastique segmentée par une méthode basée sur la croissance de région. Cette analyse était en 3D. Les histogrammes des courbures des stack normaux et diabétiques ont été comparés, et nous avons observé un épaulement dans les histogrammes des souris diabétiques. La deuxième analyse a été réalisée sur l'ensemble des lames élastiques extraites par une méthode de deep learning permettant de distinguer la *tunica media* et la *tunica adventitia*. Une mesure d'ondulation a été appliquée sur les fragments de ces lames segmentées et squelettisées. Une différence significative a été observée. Cela implique que les lames élastiques des souris diabétiques sont plus lisses que celles des souris normales. Les résultats de ces deux analyses peuvent contribuer à identifier des biomarqueurs susceptibles de caractériser l'apparition de maladies vasculaires.

Chapter 1

Introduction

Cardiovascular diseases (CVDs) are the world's leading cause of death. As a consequence, efforts have been made to detect and prevent these age-related diseases, favored by diabetes or chronic renal failure, that develop progressively, affecting, in particular, the large elastic arteries (atherosclerosis, aneurysm) [1]. Vascular remodeling is a consequence of molecular aging of the arterial wall components. In this context, extracellular matrix (ECM) fibers (collagen and elastin) aging is critical. Indeed, these large fibers are long lived and experience almost no turn-over [2]. It is therefore considered that an individual ages with his/her elastin capital and any alteration of elastin is fundamentally irreversible [3].

Considerable research has described the impact of age and diseases on arteries morphology and function. Thijssen *et al.* [21] focused on direct effects of advancing age on the structure and function of the vasculature. Their work demonstrates a gradual, age-related decline in vascular function that occurs throughout the vascular tree (that can be assimilated as a network), ranging from coronary arteries to peripheral conduits in humans. Structurally, older arteries have thicker walls, which are stiffer and exhibit a larger diameter. This points out that vascular function can be assessed by the description of the vascular tree structure. This knowledge might then be used for the prediction of future CVDs and may help to distinguish between the biological and chronological age of arteries in humans *in vivo*.

In 2013, Tsamis *et al.* [22] synthesized the results of studies on the interrelation of location, age and disease impact with the microstructure of elastin and collagen in the human aorta. All aortic diseases are associated with microstructural changes, either to the content or architecture of the connective fibres (elastic or collagen fibres). For elastic fibres, the wall of human ascending thoracic aorta (ATA) is impacted by age and diseases. The elastin content decreases with age [23] [24] in

the wall of human abdominal aorta (AA). In fact, the effects of diseases on elastin have been documented in several studies. For collagen, effects of age and disease have both been found on ATA and AA. However, the collagen content and concentration were shown to increase with age in AA [25].

Our study focuses on the imaging study of ECM micro-remodeling that occurs during aging and/or the development of vascular diseases.

1.1 Biological background

1.1.1 Cardiovascular system and aortic diseases

The cardiovascular system is the system responsible for delivering blood to different parts of the body, consisting of the heart and a closed system of blood vessels including arteries, veins and capillaries [26]. These components form two blood circulatory systems. In the pulmonary circulation, blood is moved between the heart to the lungs where it captures oxygen. The systemic circulation is the circulation, controlled by the left heart, which supplies oxygenated blood to the systemic organs (heart muscle, brain, liver, etc.) [27].

Figure 1.1 shows the structures of a heart. The right atrium receives deoxygenated blood from the body via the superior and inferior vena cava. The right ventricle receives blood from the right atrium and then pumps blood into the pulmonary trunk to the lungs. The oxygenated blood is received by the left atrium via the four pulmonary veins. It then goes to the left ventricle which pumps blood in the aorta [28]. Thus, the aorta is the most proximal artery connected directly to the heart and acts both as a conduit and an elastic chamber, by damping the systolic pressure surge, providing a constant pressure in systemic organs [27]. Note that the study of the integrity of the elastic structures of the aorta is in fact the main biological goal of this study.

Cardiovascular diseases are the leading cause of death globally, taking an estimated 17.9 million lives each year [32]. Aortic disease is, with coronary heart disease, stroke and peripheral arterial disease, one of the main of CVD. The most common aortic disease is the aortic aneurysm, where the wall of the aorta becomes weakened and bulges outwards [33]. An aneurysm can be defined as a permanent and irreversible localized dilation of a vessel [34], as is shown in Figure 1.2. Aortic aneurysms can be classified into thoracic aortic aneurysms (TAA) and abdominal aortic aneurysms (AAA) according to their location [30], as shown in Figure 1.3. Patients with AAAs frequently suffer from atherosclerosis [35], which is a narrowing

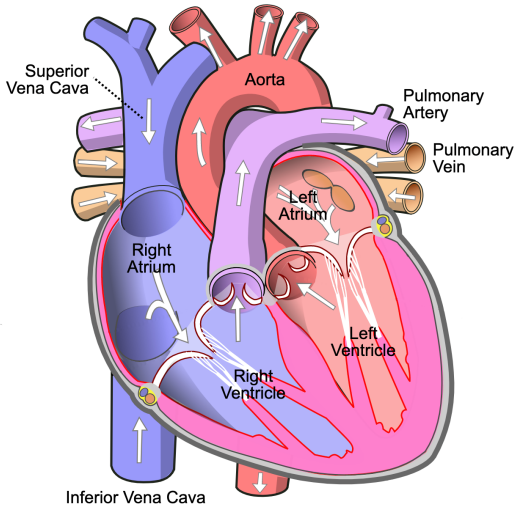


Figure 1.1: Structures of a heart [29] (modified). Blood from the superior and inferior vena cava goes to the right atrium then the right ventricle and is pumped into the pulmonary artery. The blood that carries oxygen enters the left atrium then the left ventricle and is pumped into the aorta. The aorta, where blood carries oxygen, is the part of the artery that is concerned in our study.

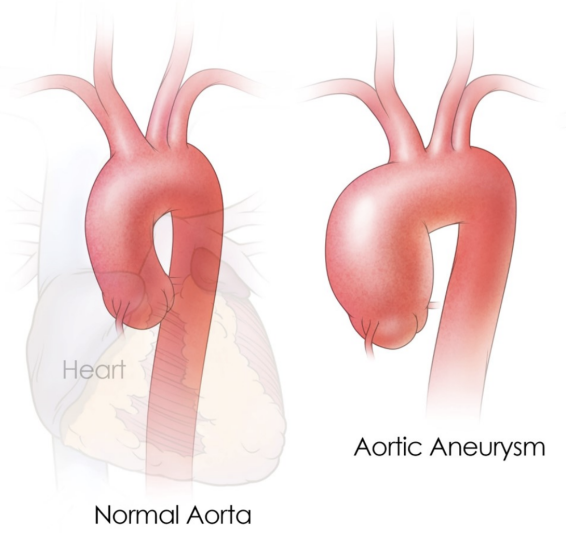


Figure 1.2: Normal aorta and aortic aneurysm [30].

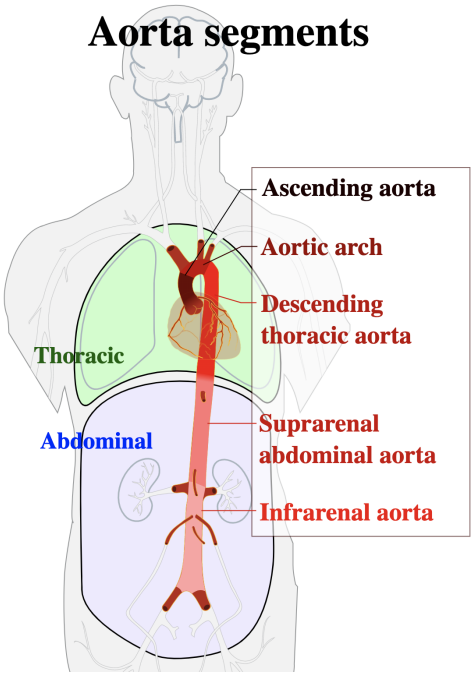


Figure 1.3: Segments of the aorta [31]. Thoracic aorta includes ascending aorta, tortic arch and descending thoracic aorta; abdominal aorta includes suprarenal abdominal aorta and infrarenal abdominal aorta.

of the arterial lumen caused by deposition of an atheromatous plaque, that eventually breaks and causes stroke [36]. Atherosclerosis is asymptomatic for decades because the arteries enlarge at all plaque locations; thus there is no effect on blood flow [37]. Aortic dissection is another type of aortic disease. It is a disruption of the aortic wall that forms an intimal flap and separates a true from a false lumen [38]. It is clear that CVD, especially age-related ones, is the leading cause of death.

1.1.2 Arterial wall, extracellular matrix and elastic lamina

An arterial wall consists of three layers, as shown in Figure 1.4. The *tunica intima* (also called *tunica interna*) is the innermost layer. It is a simple squamous epithelium surrounded by a connective tissue basement membrane with elastic fibers [5]. This thin layer is in direct contact with the blood and ensures the sealing of the artery. Note that this layer will not be visible in our study. The *tunica media* is the middle one. It is primarily muscle and is usually the thickest layer [5]. It is largely composed of concentric layers of vascular smooth muscle cells (VSMCs) within an ECM composed of collagen and elastin fibers [6]. The purpose of the muscle cells is to regulate the diameter of the arteries in order to decrease/increase the flow according to the needs of the organs or to adapt to a position (standing,

lying down, etc.). The study of the ECM of this layer will be a key point of this work. The *tunica externa* (or *tunica adventitia*) is the outermost layer, which attaches the vessel to the surrounding tissue [5]. These three layers lie on or are embedded in their extracellular matrix [7].

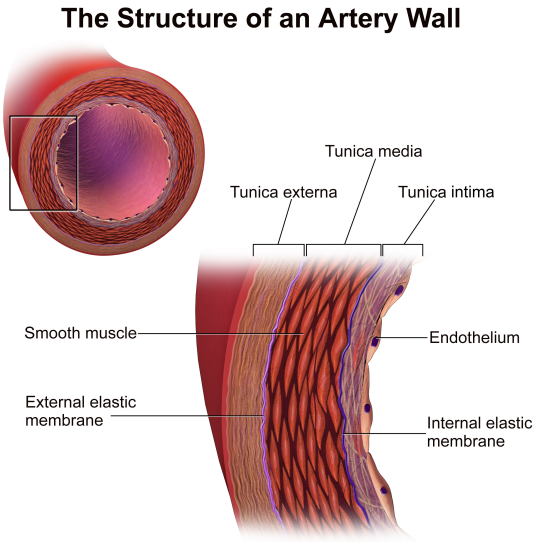


Figure 1.4: Structure of an artery wall [4]. Components including endothelium, internal elastic membrane, smooth muscle and external elastic membrane form three layers of an artery wall: *tunica intima*, *tunica media* and *tunica externa*.

The extracellular matrix is an active and dynamic structure that plays a fundamental role in regulating vascular function in normal and pathological conditions [3]. It serves both as a structural scaffold and as an influencing medium. Indeed, ECM orients cell migration, differentiation, and proliferation by interacting with membrane-associated receptors or by modulating the bioavailability of matrix-bound biochemical signals [8]. The ECM is composed of fibres made of collagens or elastin, of glycoproteins, and of glycosaminoglycans [3]. Collagens and elastin are the main protein constituents of vessels [7]. Elastin contributes up to 50% of the vessel's dry weight [39]. It allows vascular elasticity and recoil [7]. In the *tunica media*, elastic fibres are arranged in fenestra sheets, or lamellae, and are circumferentially aligned with VSMCs [7]. In most arteries, the medial layer is separated from the *intima* and *adventitia* by concentric elastin sheets referred to as the internal elastic lamella (IEL) and external elastic lamina (EEL). The IEL influences the mechanical and mass transport properties of arterial layers, and the inner surface of the *adventitia* is lined with the EEL, which is generally less pronounced than the IEL [6]. In the vascular wall, the lamellar units support and evenly distribute the stress in the

arterial wall. This is extremely important for maintaining tissue homeostasis [40]. Figure 1.5 shows the colored elastin in the elastic mouse aorta.

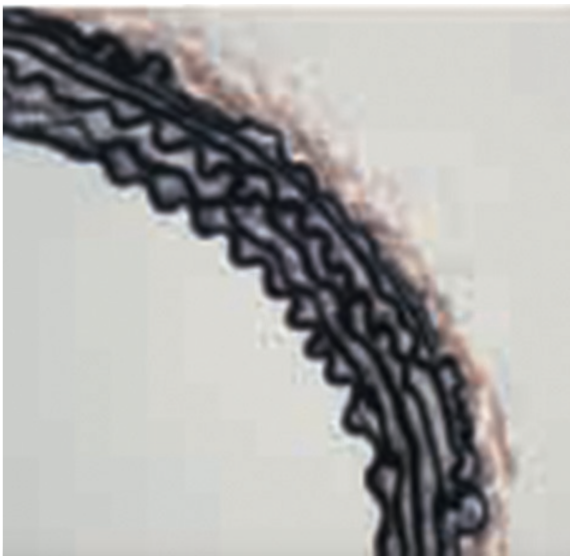


Figure 1.5: Elastic structures of the arterial wall. This microscopy image shows the arterial wall of a section of a mouse artery with elastin marked in blue/black. The elastic structures are arranged like concentric sheaths around the artery [8].

1.1.3 ECM alterations during aging and development of diseases

Many risk factors including pathological lipid levels, diabetes, genetic factors and aging contribute to CVDs [41]. As a highly dynamic structure, the ECM constantly undergoes a remodeling process, during which ECM components are deposited, degraded, or modified [42]. Arterial ECM remodeling can be observed during aging and pathological process. As is described in the previous section, elastin is the main ECM component of the vessel wall [43]. The malfunction or loss of elastic structures are associated to aging and this is devastating for the cardiovascular system [41] [44]. It has been shown that there is an age-associated increase of *intima media* thickness due to both luminal dilation and increased wall stiffness [7]. The relative elastin content decreases during the aging process [45], and due to repeated mechanical loading and oxidative stress, elastic lamellae show clear signs of fragmentation and rupture [46]. Age-related arterial alterations that affect cells and ECM are the main causes of cardiovascular disease initiation and progression [43]. Numerous correlations have been found between cardiovascular diseases and ECM alterations. The ECM remodeling in aneurysmal aortas is characterized by the structural disorgani-

zation and disappearance of the elastic lamellae in association with an inflammatory adventitial fibrosis [7]. Aortic dissections and aneurysms show a decrease in collagen content that could be related to a weakness of the wall underlying the diseases [47]. Elastin concentration was found to decrease in dissections [48]. A study showed that during the development of an arterial luminal stenosis, there are compensatory changes in the *media*. ECM remodeling promotes expansion of the artery intending to normalize lumen diameter and shear stresses [49].

When a pathology is clinically declared, the damages have already reached the arterial structures and this is considered too late. Thus, the detection of alterations indicating an early stage of these age-related pathologies is of significance; the earlier they are detected, the better the development of pathologies can be managed and potentially stopped. As described above, significant alterations of ECM components appear during aging and pathological processes, and the alteration of elastic lamellae may contain a lot of information related to these processes. For this reason, our work focuses on arterial wall ECM alterations during aging and vascular diseases development. We shall specifically consider elastic lamellae structures and extract their geometric deformations to discover biomarkers that may indicate a possible future occurrence of a vascular disease.

1.2 Related works of elastic lamellae image analysis

There are several studies on elastic lamellae from their images, including segmentation and analysis. Most of these biological studies are limited to the visual aspect of these structures but without measuring them. However, some works presented below have proposed to go one step further in the quantification. Yu *et al.* [50] used a multiphoton microscope to image the elastic lamellar layers in a mouse carotid artery to study their microstructural deformation under physiological pressurization and axial stretching. In this study, three elastic lamellae were extracted with a line-tracking by firstly extracting seeds and then selecting the pixel of which the light intensity is the maximum of the 8 adjacent pixels. This line-tracking process is performed in 2D images, and it terminates when all the 8 adjacent pixels reached a value of 0. The analysis of the structure of these extracted lamellae is performed on a 3D stack which is a pile of 2D cross-sectional images. This process is illustrated in Figure 1.6.

Walton *et al.* [51] used X-Ray computed microtomography (micro-CT) to visualize arterial wall structure and analyzed the intra-luminal pressure-induced re-

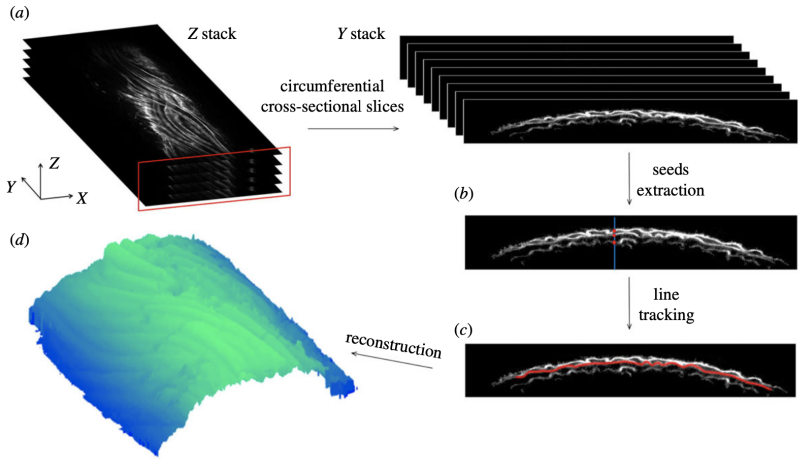


Figure 1.6: Elactic lamellae extraction and 3D reconstruction process in the work of Yu *et al.* [50].

modeling of rat common carotid arteries. In the segmentation process, the vessel wall was firstly unwrapped, then thresholded and applied with a 1D morphological opening operation along the vertical axis which enables to remove most features in the *adventitia*. To improve the result of the boundary between *media* and *adventitia*, an optimization approach based on the fast marching method [52] [53] was applied. This method is related to Dijkstra's method and it aims to find the geodesic path through a weighted image constructed by determining a mean edge position along the horizontal axis with a Gaussian mixture model. Weights were distributed by there distances to this mean edge position and their presence in the thresholded image. The images were finally wrapped to their original shapes. This process is illustrated by Figure 1.7.

Trachet *et al.* [54] used synchrotron imaging to determine the geometry and material properties of the arterial microstructure with carotid arteries from mice. The segmentation part consisted of two main steps. The first step was to separate the arterial wall and the background. It began with a binarisation, then small connected items were removed, a morphological closing was applied and a convex hull was computed. With the help of this convex hull and the morphologically closed image, the inside lumen was found by selecting the largest connected item in the image generated from the subtraction of these two binary images. The part of the arterial wall was then found with the convex hull and this inner lumen mask. This process is shown in Figure 1.8. The second step aimed to segment the three elastic lamellae layers. The outer half of aorta in the gray-scale image was first removed to accelerate the process; then an optimal thresholding was applied to get a binary

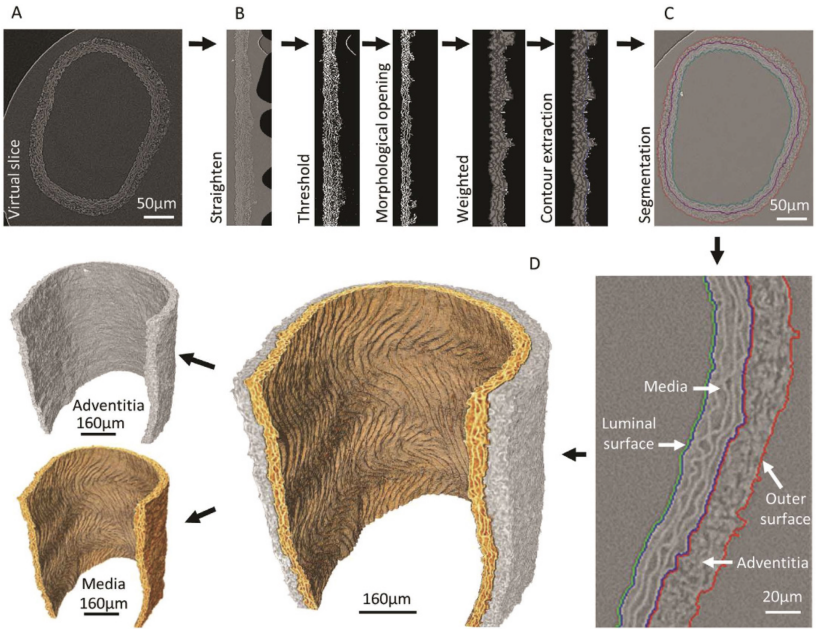


Figure 1.7: Steps of segmentation of *media* and *adventitia* in the work of Walton *et al.* [51].

arterial wall. A skeletonization was performed followed by a gap-filling as well as a branch-removing process. Next, the geometric center of gravity (x_c, y_c) of the convex hull computed in the first step was used. This was for getting a weighted image. Pixels of coordinates $(x = x_c, y > y_c)$ were subsequently set to zero. A graph of junctions and vertices was then computed; a start node and an end node were set on the left and on the right of $x = 0$. The previously computed convex hull was shrunk to distribute weights to each junction. The weight of a junction is the mean distance of those from its two nodes to this shrunk convex hull. With these junctions and corresponding weights, a modified Dijkstra algorithm was applied to find the shortest path. This path was considered to be the centerline of the first lamella and a local thickness corresponding to each pixel was estimated by doubling the distance between the centerline pixel and its inner lining in the binary image before the skeletonization. The centerline was then smoothed with these thicknesses. This centerline and the estimated thickness were used to remove the first lamella from the original image; then steps of centerline construction were repeated to extract the second and the third lamellar layer. In the extraction process of the third layer, the outer half of the arterial wall was not removed to avoid having a wall that is too thin. Finally, the outer lining of *adventitia* was also extracted from the outer convex hull and was smoothed. This second segmentation step is illustrated in Figure 1.9. These

results were then used to quantify how each of the elastic lamellae straightened when they were pressurized.

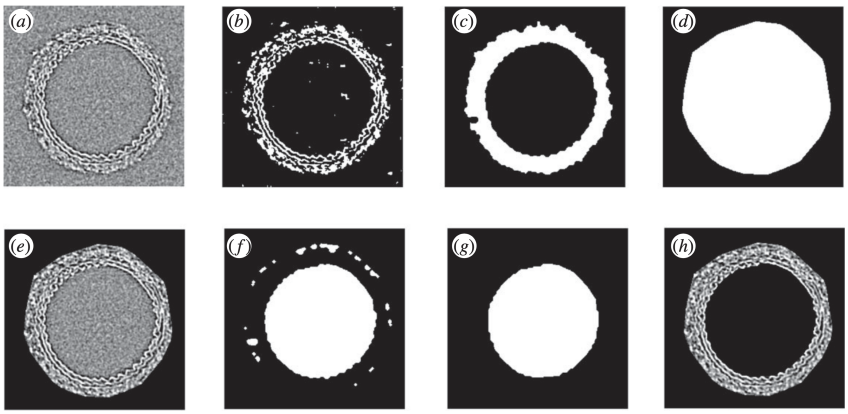


Figure 1.8: Arterial wall extraction process in the work of Trachet *et al.* [54].

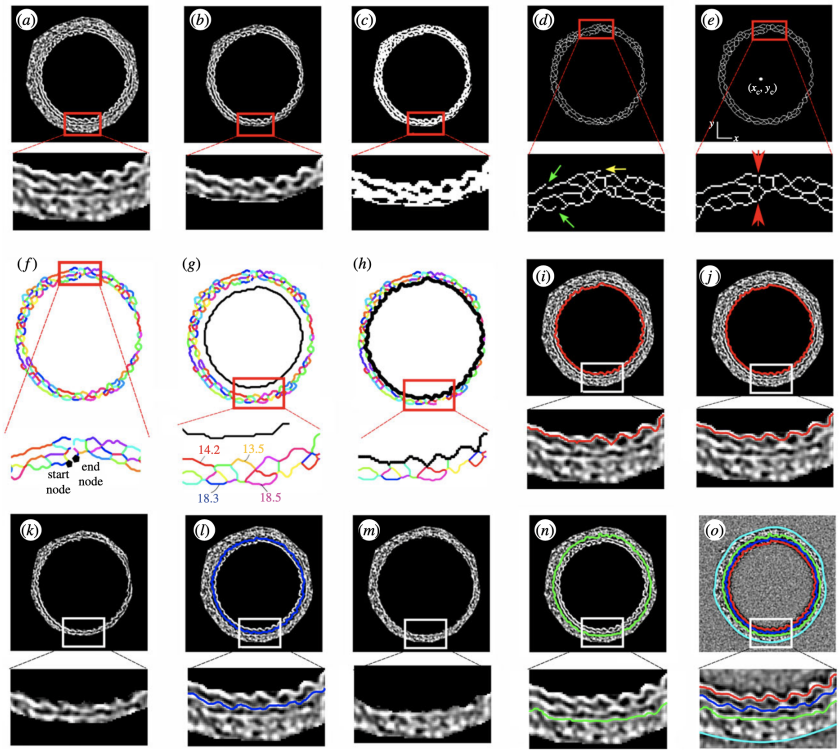


Figure 1.9: Extraction process of three elastic lamellae and *adventitia* outer lining in the work of Trachet *et al.* [54].

López-Guimet *et al.* [55] studied the characterization of Marfan aortic samples by elastic lamellae. In this work, an en-face z-stack of confocal images was acquired using multiphoton microscopy, starting from the *tunica intima* towards the outer side of aorta. The acquired images were on the XY plane and were stacked along

the Z axis. This 3D image volume enabled a visualization from the YZ plane, where the image processing was performed. This image was thresholded by the Niblack [56] algorithm to generate a binary mask. The IEL was then manually selected; images were grouped every 15 consecutive images and were used to recover the XY plane followed by an erosion routine and manual rectification if needed to correct small errors. This mask was applied to the original stack and a 3D segmented lamella as well as a maximal projection image were created. The fenestrae were found by a binarization with Sauvola local threshold [57]. The thickness and the waviness were measured for analysis; the latter was assessed by the height variation in the Z axis. The second elastic lamella was also segmented and analyzed by the same strategy.

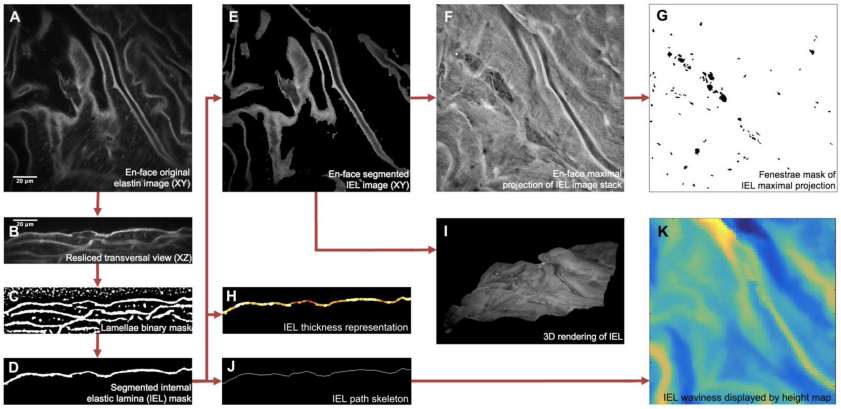


Figure 1.10: Protocol of image processing in study [55].

Ciompi *et al.* [58] proposed an approach to detect the *Media-Adventitia* border (MAb) in intravascular ultrasound (IVUS) images. This border detection consists of two main parts: a morphology-oriented task for a coarse definition of border and an appearance-oriented task for a refinement of the rough detection. This methodology is illustrated in Figure 1.13. In this study, an image was annotated with 5 structures: lumen border, external elastic lamella border, guide-wire artifact effect, calcifications and their corresponding shadows. Textures were extracted and described as feature vectors of length 17 from the image in polar coordinates by Gabor filters [59], Local Binary Pattern [60], Sobel operator, mean and standard deviation as well as their ratio in a sliding window, two author-defined measures related to the cumulative gray level of the polar image and the barycenter vector associated to the first order absolute moment operator [61]. A contextual classification based on multi-scale stacked sequential learning [62] was applied to get a robust tissue labeling with the information surrounding the pixel of interest. A coarse MAb detection was performed by looking for a curve, of which the parameters were esti-

mated by maximizing a quality measure that encodes the relationships between the curve and all the tissues of the image. To refine this rough border obtained by the morphology-oriented approach, a statistical filter representative of the real tissue appearance in correspondence of the *media-adventitia* interface was modeled, which is the matched filter in Figure 1.13. The filter was modeled with the help of the manually annotated borders and the idea is that, when the filter is applied to the image, a high response corresponds to the appearance that is more likely to be the border.

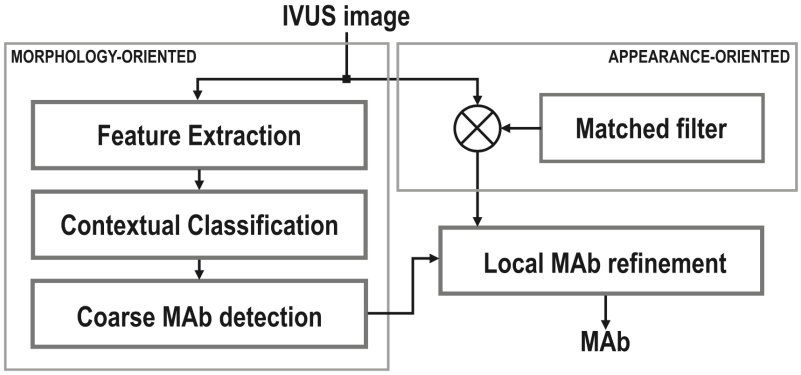


Figure 1.11: Schematic of the methodology of *media-adventitia* border detection in the work of Ciompi *et al.* [58].

1.3 Image acquisition

From a biological point of view, a technical problem arises to study the vascular structures. The first problem is that in order to visualize these structures, they must be cut to use conventional techniques that could reach a sufficient resolution. However, cutting a structure such as an aorta into thin 2D slices alters this structure which is sometimes already damaged by age or diseases. The second problem is that in order to be visualized, these structures must be stained, which requires the use of chemical agents that can also alter the tissue. The work done by Walton *et al.* [51] shows that it is possible to visualize in 3D an aortic segment, without slicing or staining it. This work inspired the MEDyC laboratory to transpose this method on mouse arteries [63].

The images we used in our study were phase contrast images without any contrasting agent. They were acquired by Synchrotron X-ray microtomography (μCT) performed on the ANATOMIX beamline [12] at SOLEIL synchrotron. In this sec-

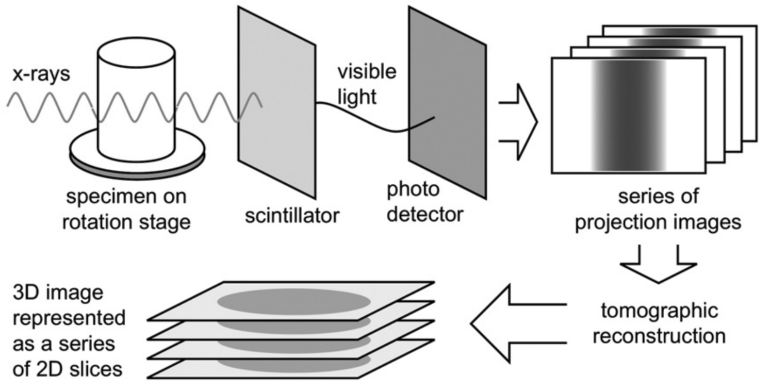


Figure 1.12: Illustration of X-ray CT acquisition and reconstruction process [65]. A 3D map of X-ray absorption is produced by a series of X-ray projection images, which is presented as a series of 2D slice images.

tion, a brief introduction of the related techniques will be presented and the image acquisition process of the samples we used will be described.

1.3.1 X-ray microtomography

X-ray microscopy is a nondestructive technique that generates an image of internal features of the examined materials. The image contrast is determined with X-rays absorbed differently by the materials [64]. X-ray microtomography is a radiographic imaging technique that enables one to obtain 3D images of the inner structure of a material at a spatial resolution less than 1 micrometer [65]. It is based on computerized axial tomography (CT) scans [66] [67]. This technique is complementary to higher resolution 2D microcopy and lower resolution 3D ultrasonic imaging, with the resulting 3D images displayed (after reconstruction) as a series of 2D slices [65]. The acquisition and reconstruction process of images are illustrated in Figure 1.12.

1.3.2 X-ray phase contrast imaging

X-ray imaging is a technique that mainly uses the absorption of light by matter to visualize samples. During the last three decades, innovating imaging techniques such as phase contrast imaging (PCI) have been proposed, which are particularly suitable for visualizing soft matter such as biological tissues with the advantages of increasing contrast for soft tissue and reduction of radiation dose [68]. When imaging a sample with X-rays, there are three main phenomena that can be observed:

attenuation, phase shift and small angle scattering [68], as is shown in Figure 1.13.

The base of X-ray PCI is the visualization of the wave front changes when radiation passes through an object [69], corresponding to the second phenomenon: refraction (or phase shift). The interaction during this process can be described by the refractive index:

$$n = 1 - i\beta - \delta \tag{1.1}$$

where β and δ are related to attenuation and phase shift respectively. The linear absorption coefficient μ_0 and the coefficient η of the change of phase ϕ are given by:

$$\mu_0 = 4\pi\beta/\lambda \tag{1.2}$$

$$\eta = 2\pi\delta/\lambda \tag{1.3}$$

where λ is the X-ray wavelength.

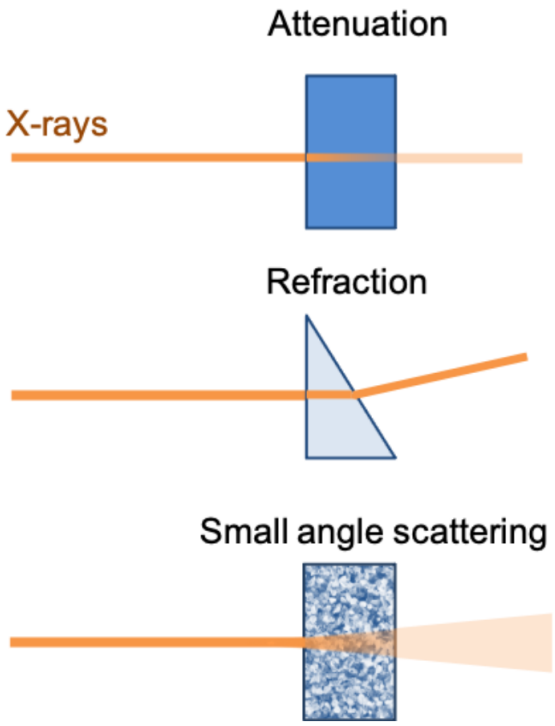


Figure 1.13: Three main phenomena observed when imaging a sample with X-rays [68].

The linear absorption coefficient μ_0 decreases approximately as $1/E^3$ while η, the coefficient of the change of phase ϕ is proportional to $1/E$, where E is the photon

energy. Also, there is a large difference between the numerical values of μ_0 and η, which makes the latter two to three orders greater than the former for light elements of human body tissue, indicating that the variation in soft tissue density induces a much stronger rise of phase contrast than that of absorption contrast [69]. There are different phase retrieval methods including propagation-based imaging, analyzer-based imaging, grating interferometry, mesh-based imaging and modulation-based imaging techniques. Among these techniques, propagation-based imaging is the most used experimental set-up in synchrotrons all over the world [68].

1.3.3 Synchrotron X-ray imaging

A synchrotron is a type of circular accelerator [9]. In such a system, electrons travel near the speed of light and are maintained in a storage ring by electromagnetic fields [10]. A schema of the principle of a synchrotron is illustrated in Figure 1.14. Particles running the synchrotron have high energy. According to the Lorentz force law, the force due to the magnetic field is perpendicular to the direction of the field and that of the motion of the charged particle. This force towards the center of the circle enables the particle to change directions constantly and run along a circle at high speed. Figure 1.15 is a simplified schema corresponding to the structures in Figure 1.14 with each part labeled. Electrons are firstly produced in the electron gun and are accelerated in a linear particle accelerator (linac), then enter the booster to have a second acceleration. In this stage, electrons are curved by bending magnets and are accelerated in the straight section by a radio-frequency (RF) voltage source, and are kept circulating in the storage ring [9]. In this circulating process, particles emit synchrotron radiation. During this emitting process, the lost energy is compensated for by RF cavities [70]. Experiments take place in beamlines, which are laboratories where radiation is collected, selected, focused and directed towards the samples to be studied [11]. A beamline has typically three hutches: control cabin, experimental hutch and optics hutch [9], as is drawn in Figure 1.14.

1.3.4 Image acquisition process

Beamline

Images we used were acquired by synchrotron X-ray phase contrast microtomography, which is suitable for acquiring biomedical images due to the advantages described in the sections above. The acquisition was done at SOLEIL synchrotron, on ANATOMIX, a 200 meters long undulator beamline for full-field tomography

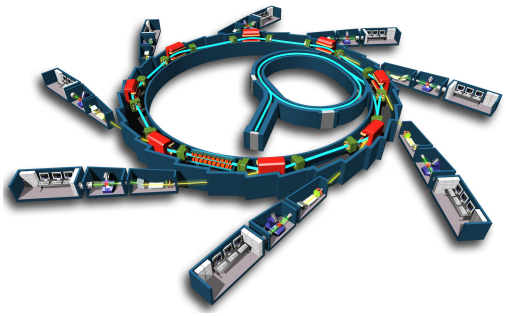


Figure 1.14: Schema of principle of synchrotron [71].

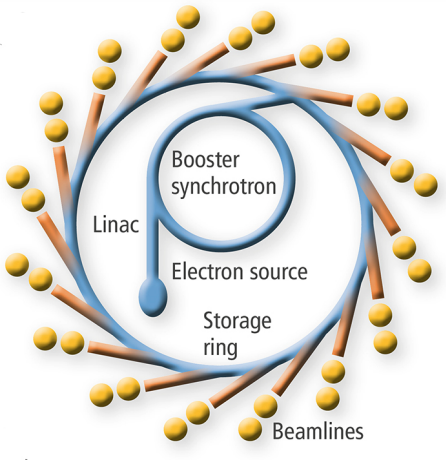


Figure 1.15: Schematic image of synchrotron with the name of each main part [72].

techniques. Photon energies are from 5 to 25 keV (in our case 11 keV), the beam size is up to 40×10 mm^2 [12].

Samples

Mice[1] arteries were used in this study. These mice were purchased from Charles River (Lyon, France) and were caged in temperature and humidity-controlled environment with a 12:12 hour light/dark cycle. After euthanasia, mice aortae were collected and the heart was injected by 2500 UI heparin; residual blood was removed by washing heart and aorta with 10 mL Phosphate-buffered saline (PBS). 5 mL of 4% formalin was injected to prefix the aorta and 6 mL of 1% low melting agarose was injected to keep the aorta open and to prevent collapse. Tissues surrounding heart and aorta were also collected. Samples were fixed in 4% formalin for 24–48 hours, dehydrated and embedded in paraffin. The final samples were paraffin rods with lengths of 40 mm and widths of 5 mm containing the heart and aorta.

The mice were mainly divided into two groups: the first group gathers those undergoing normal aging, the second group gathers those having pathologies so that we could observe the pathologic alterations of their arteries. For the first group, mice of inbred strain C57BL6J (most widely used in preclinical research) were chosen, noted control, with ages ranging from 2 to 24 months (2, 3, 6, 9, 12, 18 and 24 months). There are two types of pathologic mice in the second group: diabetic mice

[1]Mouse procedures were realized in accordance with the Animal Subjects Committee of the Champagne-Ardenne Region (France).

which were highly obese (db/db), and mice that developed atherosclerotic plaques around 1 year ($ApoE^{-/-}$). Because of uncertain availability of $ApoE^{-/-}$ mice, most of the pathologic mice used for making samples were diabetic mice.

Image acquisition

The X-ray source in ANATOMIX beamline is from an undulator. The beam to image to the samples is polychromatic ("white") and the source was set to a gap of 8.5 mm. The beam was filtered by a 0.6-mm-thick diamond plate and a 10-μm-thick layer of gold. The detector was an indirect lens-coupled system with a 20-μm-thick lutetium aluminum garnet single-crystal scintillator coupled to a CMOS-based scientific-grade camera with 2048^2 pixels via microscope optics (10× objective), resulting in an effective pixel size of 0.65 μm on the sample level. The distance between sample and scintillator was 22 mm. The exposure time for the camera was set to 100 ms per projection image. 1500 projections were taken over an angular range of 180° most of the time. The samples were positioned vertically with the heart in the lower part. Immediately after the acquisition, the imaged volume was reconstructed and checked. The reconstructed volume stacks for each scan contained 2048^3 voxels of size $(0.65\mu m)^3$, each represented by a 32-bit single-precision float value, i.e. a total of 32 GB per volume.

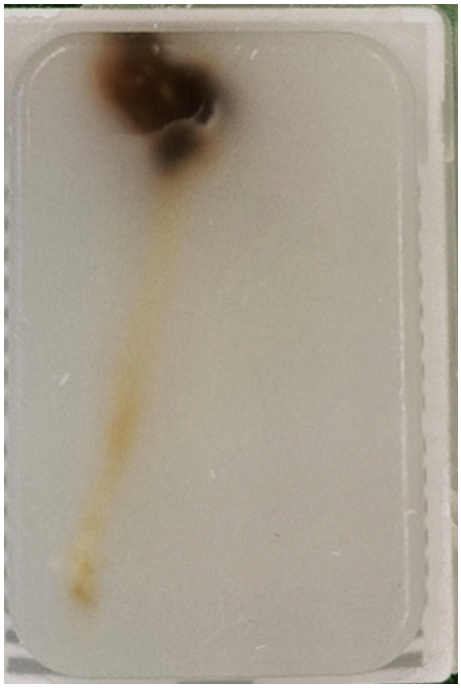


Figure 1.16: A sample of a mouse aorta.

Figure 1.16 shows a sample of a mouse aorta. The process of making the samples and acquiring the images was done by a graduated PhD student, Aïcha Ben Zemzem. We used the images she handed over to us in her third year for this study.

The thesis manuscript is organized into 6 chapters. Chapter 1 introduces the context and biological background as well as the related techniques of our work. Chapter 2 analyzes the raw data and describes the pre-processing methods of this data. Chapter 3 describes the lumen extraction process, followed by Chapter 4, which presents the segmentation methods for the first elastic lamella and the *tunica media* based on the result of Chapter 3. Chapter 5 analyzes the results of the previous chapter from two aspects. Chapter 6 concludes our work and discusses the methods as well as future works.

Chapter 2

Raw data analysis & pre-processing

Our study was done on 55 3D image stacks, numbered from 0 to 54. Each stack contains a volume of 2048^3 voxels (except for those of stack no.24, which has only 1354 frames), each voxel represents actual size of $(0.65\mu m)^3$. The image processing is done on 3D volumes and 2D frames of size 2048^2 pixels. These stacks are composed of 24 stacks of normal mice (control), 28 stacks of diabetic (db/db) and 3 stacks of mice that have atherosclerosis ($ApoE^{-/-}$). Their specific information is given in Table 2.1.

These images have some features in common. For example, they all contain noise, which was generated during the image acquisition. Besides, there exist features varying by stacks, that appear only in certain stacks or are at different levels. For example, the number of bifurcations is not the same in every stack, and the existence of high contrast matters attached to the inner wall depends on the sample. In this chapter, we first summarize the features of each stack and then perform preprocessing according to these features to reduce their disruption to the segmentation process.

Table 2.1: Sample numbers and their information of age and type.

Number	**0**	**1**	**2**	**3**	**4**	**5**	**6**
Age (month)	6.5	6.5	6.5	6	6	6	6
Type	control	control	control	$ApoE^{-/-}$	$ApoE^{-/-}$	$ApoE^{-/-}$	db/db
Number	**7**	**8**	**9**	**10**	**11**	**12**	**13**
Age (month)	6	6	6	6	6	6	6
Type	db/db	db/db	db/db	db/db	db/db	db/db	db/db

Number	**14**	**15**	**16**	**17**	**18**	**19**	**20**
Age (month)	6	6	6	6	6	6	6
Type	db/db	db/db	db/db	db/db	db/db	db/db	db/db
Number	**21**	**22**	**23**	**24**	**25**	**26**	**27**
Age (month)	6	6	6	6	6	6	6
Type	db/db	db/db	db/db	db/db	db/db	db/db	db/db
Number	**28**	**29**	**30**	**31**	**32**	**33**	**34**
Age (month)	6	6	6.5	6.5	6.5	6.5	6.5
Type	db/db	db/db	control	control	control	control	control
Number	**35**	**36**	**37**	**38**	**39**	**40**	**41**
Age (month)	6.5	6.5	3	12	12	12	12
Type	control	control	control	control	control	control	control
Number	**42**	**43**	**44**	**45**	**46**	**47**	**48**
Age (month)	8	8	8	8	8	8	8
Type	control	control	control	control	control	control	control
Number	**49**	**50**	**51**	**52**	**53**	**54**	
Age (month)	8	8	6	6	6	6	
Type	control	control	db/db	db/db	db/db	db/db	

2.1 Raw data analysis

2.1.1 Components in images

Each volume stack contains a part of the aorta and its surrounding tissues as well as some matters that have been introduced during the sample preparation process. These structures can be observed in 2D frames. Figure 2.1 is the first frame of stack no.43. It is a stack of the aorta of a normal (control) mouse. From this cross-section, we can see the arterial wall, the perivascular tissue and the agarose that exists inside and outside the lumen. The latter was introduced during the preparation of samples. The visible field, where the noise exists, is the inner tangent circle from which we can see the round border. All the pixels have the same intensity outside the field.

The example in Figure 2.1 is a frame that has a "regular" structure without much disturbance for the segmentation process. It shows an arterial wall which is well closed with some agarose that is inside the lumen but does not cling the inner wall. The first lamella layer can be seen clearly, which makes it easy to define the inner border of elastic lamellae. However, all over the stacks we have a large portion of images that do not look as ideal as that. Indeed, various features may appear in different stacks or in individual frames of the same stack.

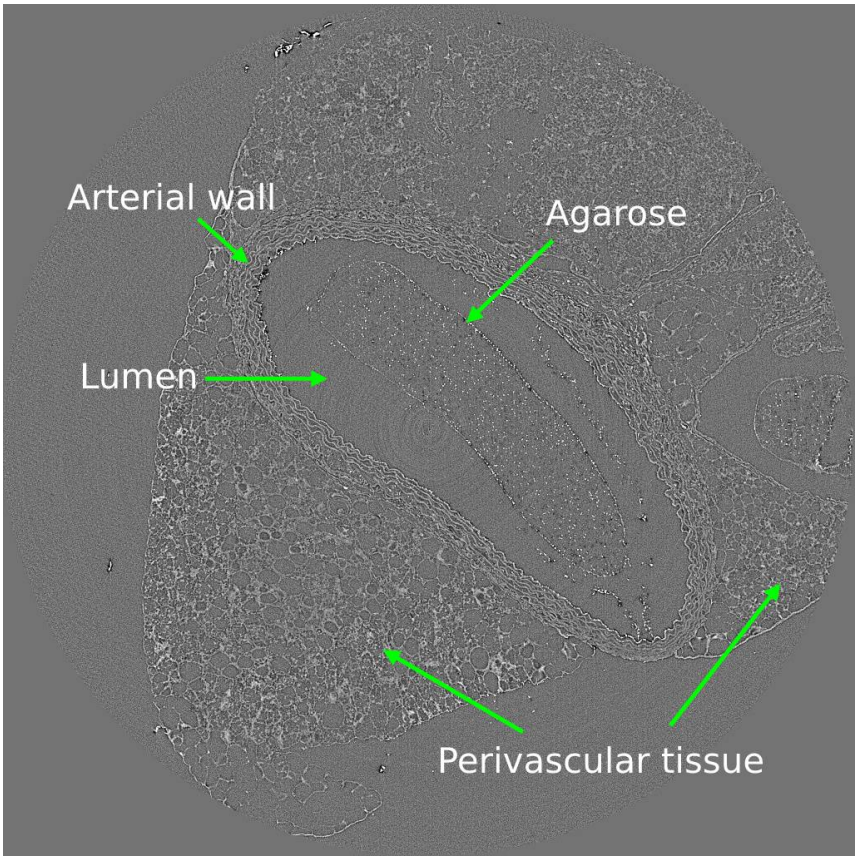


Figure 2.1: A frame of stack number 43 (control). The lumen of the aorta is surrounded by the arterial wall. The perivascular tissue is around the outer side of the arterial wall and the agarose exists both inside and outside the arterial wall.

2.1.2 Feature analysis

Features related to the acquisition of sample/image

Images in different stacks have common as well as specific features related to sample-making and image-acquisition processes. Problems such as low contrast and noise both in the background and our regions of interest (ROIs) exist in all the stacks,

the ROI here refers to the arterial wall. There also exists a typical ring artifact that commonly exists in synchrotron images, which can be observed as concentric circles in the center of a 2D image. Some stacks have their own problems. In stack no.20, sudden changes of contrast exist in several frames. Some stacks have high contrast matters accumulated on the inner side of the arterial wall or all over the image; some stacks have air bubbles inside the lumen or outside the arterial wall. All these features may become disturbances in the segmentation process.

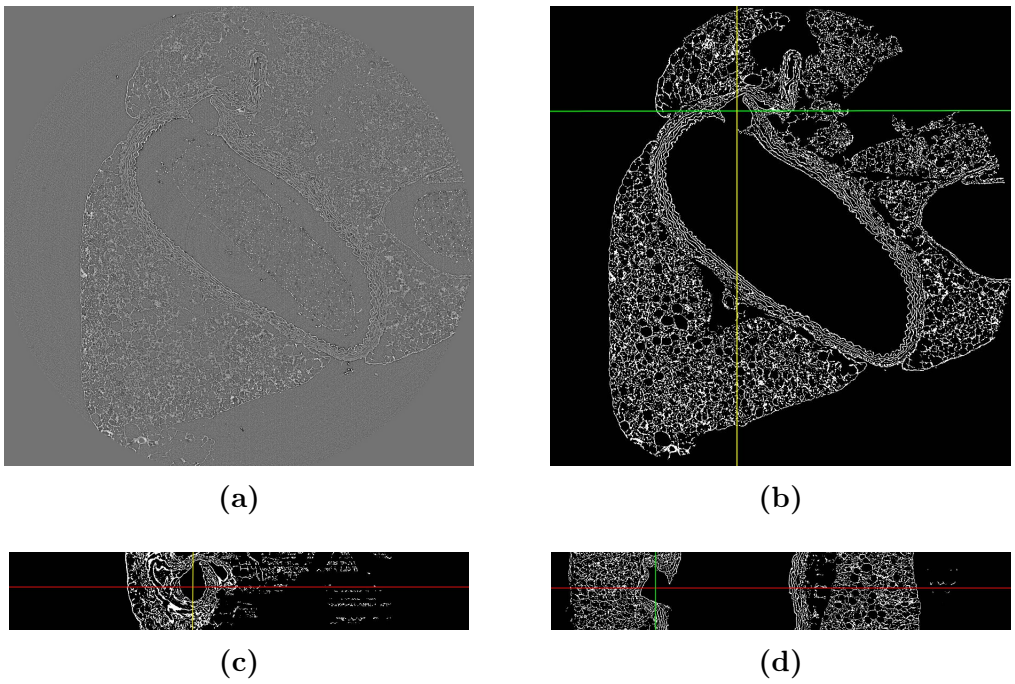


Figure 2.2: Illustration of a bifurcation in stack 43. (a) is a frame of the raw data. (b) is the binary image obtained by performing a series of pre-processing (which will be introduced later in this chapter) on (a). (b), (c) and (d) show an axial, a coronal and a sagittal plane of a part of this pre-processed stack respectively. On each of them, the other two planes are marked by lines with colors (red: axial; green: coronal; yellow: sagittal).

Features of artery samples

The artery samples were placed vertically during the acquisition of image. So in general they have a vertical and curved tubular shape. However, this generality can only describe a part of 3D volume in each stack. A structure that commonly exists in these arterial walls is the bifurcation, where grows a secondary vessel, as is shown in Figure 2.2. Another structure that can not be generalized by "tubular shape" is a lumen where the arterial walls in opposite positions contact or are too close to each

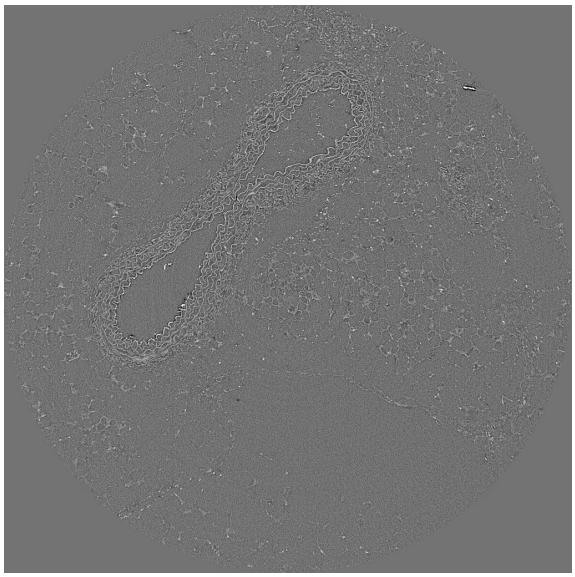


Figure 2.3: A frame of stack 16, opposite parts of arterial wall touch each other.

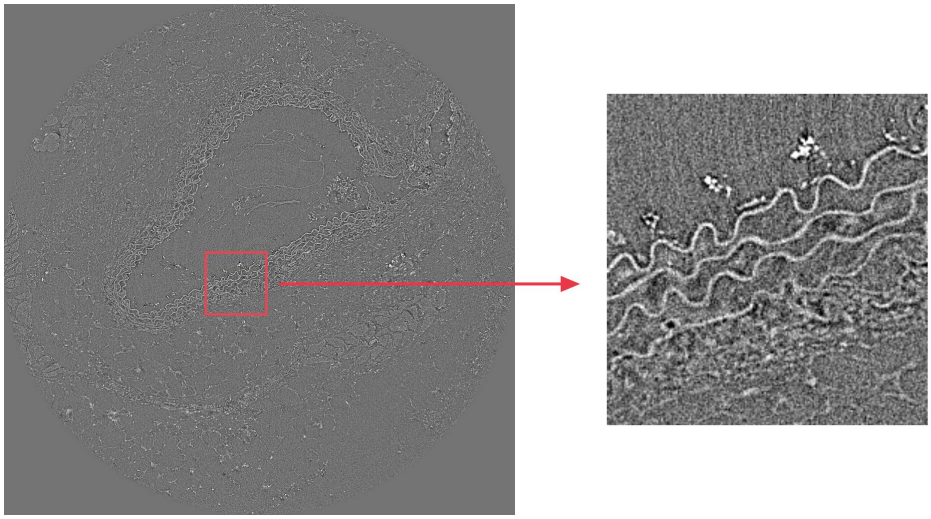


Figure 2.4: A frame of stack 17, the zoomed patch shows a part composed of 4 elastic lamellae.

other, as is shown in Figure 2.3. Besides these overall structural irregular features, if we look at the details of the elastic lamellae, we can observe that the regular "5 concentric layers" structure does not exist everywhere. Some layers have missing parts or are fractured; some are in contact with the layer next to it, and in some parts of the arterial wall, there are more than 5 layers. This feature is shown in Figure 2.4.

Table 2.2 lists important features observed in each stack. 2D image examples corresponding to these features are attached in Appendix A.

Table 2.2: Notable features in each stack. Feature types are numbered for Appendix A. For features 1 and 2, the number of "+" reflects the amount of impurities or the extent to which it affects the image analysis and "/" means there is almost no impurity. The number in the column of feature 3 is the number of bifurcations appearing in a stack. Columns corresponding to features 4 and 5 show the existence of these features in each stack. Notes in column 6 describe in detail the other features in some frames that are not included in the first 5 columns.

	1. Agarose or impurity inside the lumen	**2. Impurity on the first lamella**	**3. Bifurcation**	**4. Artery outside the field**	**5. Air bubble**	**6. Others**
0	+++	+++	2	No	No	/
1	/	+++	2	No	No	/
2	+++	+++	2	No	No	/
3	++	/	4	No	Yes	/
4	++	/	5	No	Yes	/
5	++	+	1	Yes	Yes	Irregular shape
6	+	/	2	No	Yes	/
7	++	+	3	No	No	/
8	+++	+++	3	Yes	No	/
9	+++	+++	2	No	No	/
10	+++	+++	3	No	No	/
11	+++	+++	3	No	No	/
12	+++	++	2	No	No	/
13	++	+	2	No	No	/
14	+	/	0	Yes	No	/
15	++	++	2	No	No	/
16	++	+	2	No	No	Walls close to each other
17	+	++	3	No	No	/
18	/	+	3/4	Yes	No	Irregular torsion

19	/	+ + +	2	Yes	No	Ruptured wall
20	+	+ + +	2	No	No	Abnormal contrast
21	/	+	1	No	No	/
22	+	++	2	No	No	/
23	/	/	3	Yes	No	/
24	+ + +	+ + +	3	No	No	/
25	+ + +	+ + +	2	No	No	/
26	+ + +	+ + +	2	No	No	/
26	+ + +	+ + +	2	No	No	/
27	+ + +	+ + +	4	No	No	/
28	+ + +	+ + +	3	No	Yes	/
29	+ + +	++	1	Yes	Yes	/
30	+ + +	++	2	No	No	/
31	+ + +	++	3	No	No	/
32	+ + +	+ + +	2	No	No	/
33	+ + +	++	3	No	No	/
34	++	++	3	Yes	No	/
35	/	/	3	No	No	Irregular bifurcation
36	++	/	2	No	Yes	/
37	+ + +	++	2	No	Yes	/
38	+ + +	/	0	No	No	Incomplete artery
39	+ + +	++	3	Yes	No	Incomplete artery
40	+ + +	++	2	No	No	/
41	+ + +	++	2	No	No	/
42	++	/	3	No	No	/
43	+ + +	++	2	No	No	/
44	++	+ + +	3	No	No	/
45	+ + +	++	2	No	No	/
46	+ + +	+ + +	2	No	No	/

47	+++	++	3	No	No	/
48	+++	++	3	No	No	/
49	+++	++	2	No	No	/
50	+++	+++	2	Yes	No	/
51	++	+++	2	No	No	Walls close to each other
52	+++	++	2	No	No	/
53	+++	+++	1	No	No	Walls stacked together
54	+++	+++	2	Yes	No	/

2.2 Pre-processing

All these features give these 3D stacks similarities and also make them different from each other. Thus, different methods are applied in both pre-processing and further operations.

The preprocessing step aims to facilitate the following segmentation steps. The first segmentation step is a segmentation of lumen on binary images, which will be described in Chapter 3. So in the preprocessing process, we tackle the problems that are caused by some of the features described in Section 2.1. The aim is to generate images in which the impact on the lumen segmentation process is reduced. In this chapter, we focus on pre-processing methods of contrast adjustment and binarisation, aiming to get binary images for lumen extraction.

We performed the processes in this chapter and the following chapters on 52 stacks (diabetic and normal samples), without stacks 3, 4 and 5, which are $ApoE^{-/-}$ samples.

2.2.1 Related works

Contrast enhancement

The acquired images have low contrast that makes details less obvious. Contrast enhancement techniques aim to improve this situation. There exist various image enhancement methods in the spatial domain. Histogram equalization (HE) [73] is a

common method that modifies the image contrast by flattening the histogram with a transformation function (for discrete values):

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) \quad k = 0, 1, 2, ..., L-1 \tag{2.1}$$

$$p_r(r_k) = \frac{n_k}{MN} \tag{2.2}$$

where L is the number of intensity levels, T is the transformation function, MN is the total number of pixels in the image, p_r is the probability density function (PDF) of the input image, n_k is the number of pixels of intensity r_k, s_k is the mapped intensity in the output image. If there are N_k pixels with intensities lower than or equal to r_k in the input image, then there are N_k pixels with intensities lower than or equal to s_k in the output image, this corresponds to the cumulative distribution function (CDF). However, with the drawback of changing the mean intensity of the image to the middle level, HE can cause artifacts and intensity saturation problems [74]. HE based methods were proposed to improve the performance in terms of adaptation to images. For instance, adaptive histogram equalization (AHE) improved HE by performing the equalization based on regions that surround a pixel; still, it may generate undesired features [75]. Contrast-limited adaptive histogram equalization (CLAHE) [75] improved the performance of HE by limiting the amplification of contrast according to the slope of the function that maps input to output intensities. Pixels that exceed a prefixed clipping limit are redistributed uniformly across the intensity range of the whole image. Another improved method is brightness preserving bi-histogram equalization (BBHE) [76]. In this method, an image is split into two sub-images; one is composed of pixels having lower intensities than the mean intensity of the original image, another is composed of those having (strictly) greater intensities. Histogram equalizations are then performed on these two images independently. Equal area dualistic sub-image histogram equalization (DSIHE) [77] uses an image separation strategy by maximizing the entropy. Minimum mean brightness error bi-histogram equalization (MMBEBHE) [78] takes into consideration the output intensity when searching a threshold separating the sub-images. In this method, the image is split by a threshold that achieves the minimal absolute mean brightness error between the input and the output image, which is the absolute difference between their means. Recursive mean-separate histogram equalization (RMSHE) [79] splits an image r times, which means if $r = 2$, sub-images generated by BBHE are further split into 2 sub-images respectively. This method preserves better the

brightness and gives a more natural enhancement. Dynamic histogram equalization (DHE) [80] aims to divide the image histogram into several sub-histograms until there are no dominating components among them. This method is composed of 3 steps: histogram partition, gray level allocation and histogram equalization on each sub-histogram. In the last step, HE operations are performed according to the ranges of gray level that have been decided in the second step. Brightness preserving dynamic histogram equalization (BPDHE) [81] uses a similar strategy. In this method, the histogram is firstly smoothed; then local maxima are detected from this smoothed histogram. Each partition is mapped into a dynamic range; partitions are equalized independently and at the end the output intensity is normalized so that the means intensity of the output is almost the same as the input. Exposure based sub-image histogram equalization (ESIHE) [74] takes image intensity exposure value into consideration to calculate the boundary that divides the images into two sub-images. A histogram clipping is involved in the method to limit the enhancement rate. Sub-histograms are then divided and equalized; sub-images are combined into one at the end.

Besides histogram transformation, there also exist intensity transformation methods that map directly the intensity of a pixel in the input image to a new intensity value in the output image, for example, *log* transformation and gamma transformation; the latter can be written as:

$$I_{out} = c \cdot I_{in}^{\gamma} \tag{2.3}$$

where c and γ are positive constants. The value of γ controls the effect of the enhancement; when $\gamma = 1$, this becomes the identity transformation. However, this method modifies all the pixels in different images in the same way, with the same c and γ. Different contrast enhancement methods based on gamma correction have been proposed.

Adaptive gamma correction with weighting distribution (AGCWD) [82] is a modified form of gamma correction with γ related to the CDF of the image, which is defined by a weighting distribution (WD) function based on the PDF. With an adjusted parameter α, the WD function is defined as:

$$pdf_w(I) = pdf_{max} \left(\frac{pdf(I) - pdf_{min}}{pdf_{max} - pdf_{min}} \right)^{\alpha} \tag{2.4}$$

where I is the intensity. The CDF is then calculated by:

$$cdf_w(I) = \sum_{l=0}^{I} pdf_w(l) / \sum pdf_w \quad \text{and} \quad \sum pdf_w = \sum_{I=0}^{I_{max}} pdf_w(I) \tag{2.5}$$

The gamma correction is finally performed by:

$$I_{out} = I_{max}(I_{in}/I_{max})^{\gamma} \tag{2.6}$$

with γ defined by:

$$\gamma = 1 - cdf_w(I_{in}) \tag{2.7}$$

The functions of the AGCWD method were designed for avoiding significant decrements of the high intensity while increasing progressively the low intensity. Somasundaram *et al.* [83] proposed a contrast enhancement method based on gamma correction for medical images. In this method, the minimum, the maximum and the median values are found from the range of $[5\%,\ 95\%]$ of the cumulative histogram of the image. A value g is calculated by:

$$g = \log\left(\frac{median - minimum}{maximum - median}\right) \tag{2.8}$$

g is then bounded to $[0.8,\ 1.2]$. The gamma transformation is finally performed by:

$$I_{out}(p) = I_{in}(p)^{\frac{1}{g}} \tag{2.9}$$

Rahman *et al.* [84] proposed an adaptive gamma correction (AGC) by firstly classifying image into 4 types according to their contrast and brightness. Given $1/\tau$ a parameter used for defining the contrast of an image, images that have a range of 4σ centered by μ lower than $1/\tau$ are classified into low contrast ones; otherwise, they are considered as moderate or high contrast ones. The values σ and μ are the standard deviation and the mean of the image intensity. Images in each group are then classified into dark and bright types. For low contrast images, γ and c in Equation 2.3 are calculated by:

$$\gamma = -\log_2(\sigma) \tag{2.10}$$

$$c = \frac{1}{1 + Heaviside(0.5 - \mu) \cdot (k - 1)} \tag{2.11}$$

where k is defined by:

$$k = I_{in}^{\gamma} + (1 - I_{in}^{\gamma}) \cdot \mu^{\gamma} \tag{2.12}$$

with the Heaviside function [85] given by:

$$Heaviside(x) = \begin{cases} 0, & x \leq 0 \\ 1, & x > 0 \end{cases} \tag{2.13}$$

For moderate or high contrast images, the stretching of contrast is lowered; γ is then calculated differently by:

$$\gamma = \exp[(1 - (\mu + \sigma))/2] \tag{2.14}$$

Image thresholding

Image thresholding is a method to obtain binary images according to intensity values. Sezgin *et al.* [86] classified image thresholding methods into 6 categories: histogram shape-based thresholding, clustering-based thresholding, entropy-based thresholding, object attribute-based thresholding, spatial thresholding and locally adaptive thresholding. Otsu's method [13] is one of the most referenced thresholding method [86], which is classified as clustering-based thresholding. In Otsu's method, the between-class variance is maximized to get the corresponding threshold:

$$\sigma_B^2(k^*) = \max_{1 \leq k < L} \sigma_B^2(k) \tag{2.15}$$

with $\sigma_B^2(k)$ calculated by:

$$\sigma_B^2(k) = \frac{[\mu_T \omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]} \tag{2.16}$$

where $\omega(k)$ is the sum of the percentage of pixels that have intensity values in range $[1,\ k]$, regarded as a probability:

$$\omega(k) = \sum_{i=1}^{k} p_i \tag{2.17}$$

and $\mu(k)$ is the mean value of these pixels:

$$\mu(k) = \sum_{i=1}^{k} ip_i \tag{2.18}$$

The value μ_T is the mean value of all the pixels in range of gray levels of the whole image $[1,\ L]$:

$$\mu_T = \mu(L) = \sum_{i=1}^{L} ip_i \tag{2.19}$$

Minimum error thresholding [87] assumes that for the foreground and the background of the image, each of their probability density function $p(g \mid i)$ is a normal distribution with mean μ_i, standard deviation σ_i and *a priori* probability P_i. The mixture of the foreground and the background is:

$$p(g) = \sum_{i=1}^{2} P_i p(g \mid i) \tag{2.20}$$

with each of the distributions expressed by:

$$p(g \mid i) = \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{(g-\mu_i)^2}{2\sigma_i^2}\right) \tag{2.21}$$

where g is the gray level. τ is the Bayes minimum error threshold that binarizes the image:

$$\begin{cases} P_1 p(g \mid 1) > P_2 p(g \mid 2), & g \leq \tau \\ P_1 p(g \mid 1) < P_2 p(g \mid 2), & g > \tau \end{cases} \tag{2.22}$$

In the method proposed in the work [87], when dealing with the binarization task, the threshold T can be found by minimizing a criterion $J(T)$:

$$J(T) = 1 + 2 \cdot \sum_{i=1}^{2} \{P_i(T_i)[\log_{\sigma_i}(T(i)) - \log_{P_i}(T(i)])\} \tag{2.23}$$

Locally adaptive thresholding calculates thresholds according to each pixel instead of using a global threshold for the whole image. This kind of method can deal with images having problems such as uneven illumination and thus needing different thresholds in different parts. The locally adaptive method can be based on local

variance or local contrast. In the method proposed by Niblack *et al.* [56], a local threshold for pixel p is calculated by:

$$T(p) = m(p) + k \cdot \sigma(p) \tag{2.24}$$

where $m(p)$ and $\sigma(p)$ are mean and standard deviation of the neighborhood surrounding pixel p, and k is a constant. In the method proposed by Sauvola *et al.* [57] for document image binarization, the threshold is:

$$T(p) = m(p) \cdot [1 + k \cdot \left(\frac{\sigma(p)}{R} - 1\right)] \tag{2.25}$$

where k and R are constants. Bernsen [88] proposed a contrast-based thresholding method by applying local thresholding:

$$T(p) = 0.5 \cdot (I_{max}(p) + I_{min}(p)) \tag{2.26}$$

where $I_{max}(p)$ and $I_{min}(p)$ are the maximum and minimum in a window centered by pixel p. The local contrast is defined as:

$$C(p) = I_{max}(p) - I_{min}(p) \tag{2.27}$$

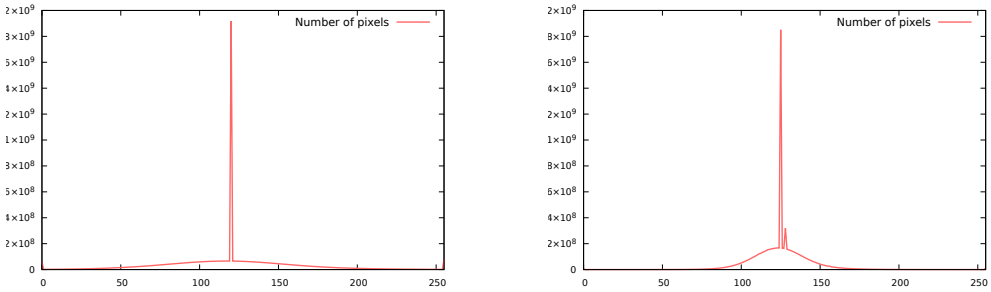
If this local contrast is not high enough, a global threshold will be applied to this pixel.

2.2.2 Contrast adjustment

Abnormal contrast adjustment

For 2D frames in a stack, the contrasts are not always stable. Some of them need to be adjusted with different methods according to their problems. We select the frames that have relatively abnormal contrasts by observing the images themselves and their histograms. For each 3D stack, we calculated a histogram with the pixel intensities of all the frames and observed. The two subfigures in Figure A.21 in Appendix A from stack no.20 both have problems of contrast. The former has saturated pixels and the latter has low contrast. Figure 2.5 shows two types of histograms from two stacks from stack no.0, and stack no.20 respectively. The latter has a more concentrated shape than the former, which reflects that stack 20 has a lower contrast in general. In addition, the latter has two intensity peaks, which

is caused by the different distributions of intensity in different segments of the 3D stack.



(a) **(b)**

Figure 2.5: Intensity histograms of stack 0 (a) and stack 20 (b).

By observing each frame of stack 20, we found that the contrasts are relatively high and normal in the first 155 frames. As shown in Figure 2.7, the histogram of the intensity of these frames is similar to that of stack 0. We use the substack composed of the first 155 frames (from frame 0 to frame 154) as a reference and compare its histogram with each frame from the rest of the stack by calculating their correlation. The result is in Table 2.3, compared with that of stack 0, showing a sudden change of intensity distribution from the 156th frame. The saturation problem exists in frames 160 to 163, and the problem of low contrast exists in frames 155–159 and 165–2047 (frame 164 is relatively well contrasted). Figure 2.7 shows frame 0, which is considered normal, and two other frames with these two problems respectively as well as their corresponding histograms.

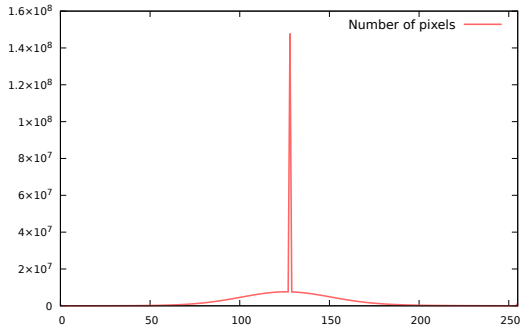


Figure 2.6: Intensity histogram of the first 155 frames of stack 20.

Table 2.3: Correlation between the histogram of the first 155 frames (reference) and other frames.

frame number	**155**	**160**	**164**	**175**	**183**
stack 0	0.999996	0.999993	0.99999	0.999992	0.999996
stack 20	0.982109	0.413614	0.99955	0.154369	0.0626226

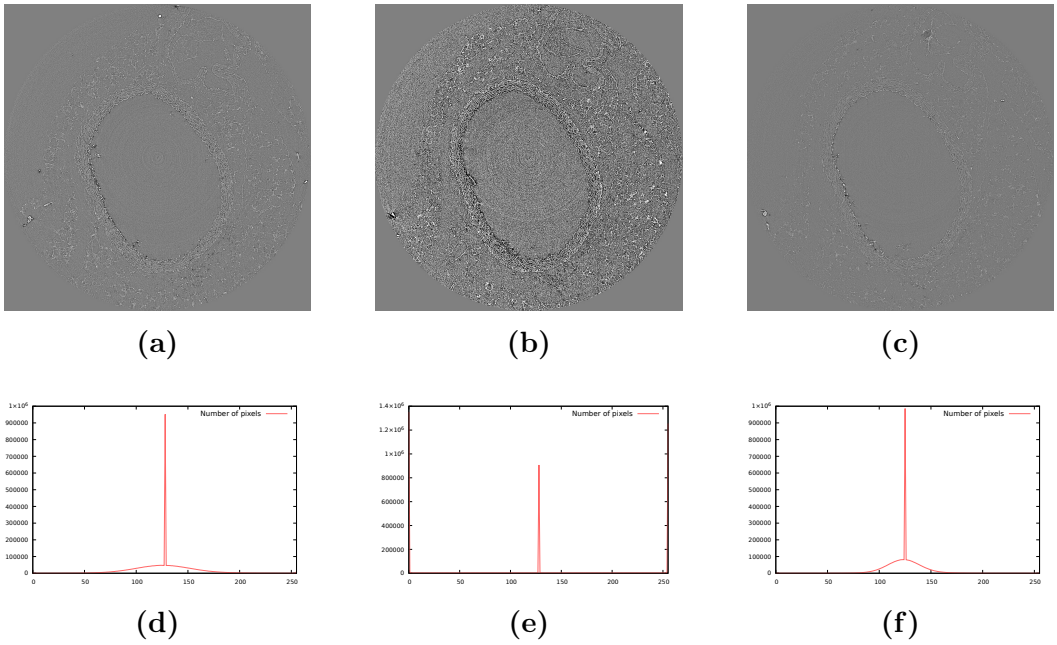
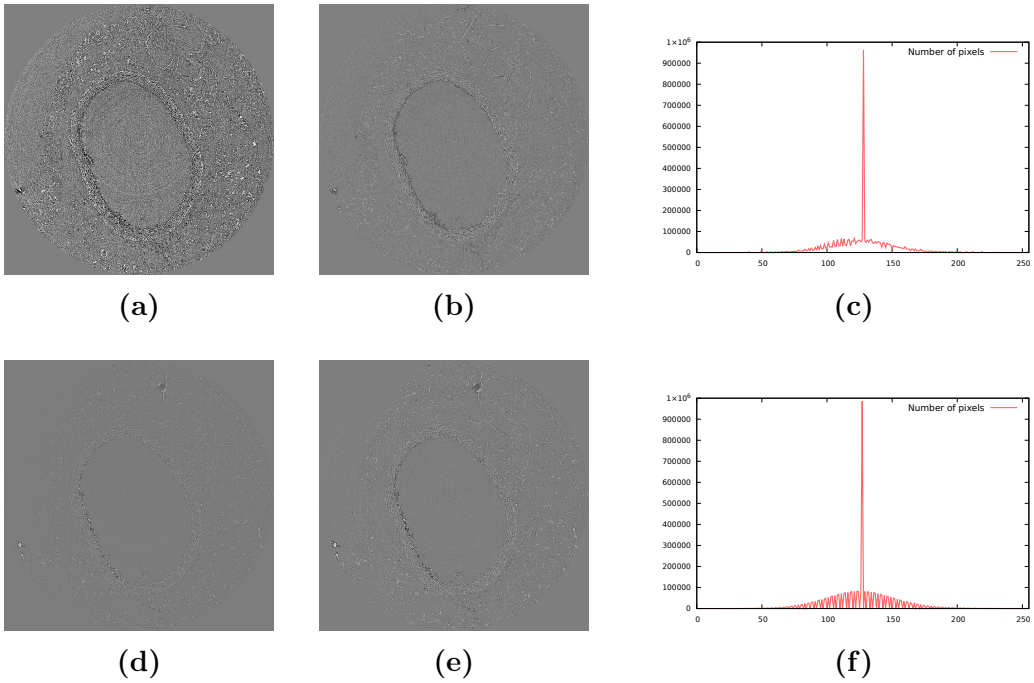


Figure 2.7: Frames 0 (a), 160 (b) and 200 (c) in raw data and their corresponding histograms (d) (e) (f).

The problem of saturation exists in frames 160–163. The intensity difference between the lamellae region and the lumen region is almost 0. We used the nearest frame that was considered normal (for frame 160, the reference frame is frame 164) for comparison. A dilation and an erosion (with a structuring element size of 7×7) were applied on frame 164 in order to find the brightest and the darkest neighbor values for each pixel. We then compared these operated images with the one to adjust, measuring the difference between each pair of pixels of the same coordinate on the dilated/eroded image and the image to adjust. The greater this difference is, the more the corresponding pixel should be adjusted. The adjusted value is calculated with initial values and from a Gaussian-smoothed image of the reference frame. In the histogram on the unadjusted image, we can see two peaks on the very left and the very right, which means the values are not well distributed. Pixels

that should be in the "gray" range are turned to white or black. We then picked these pixels (here we set the ranges out of [10, 245]) to adjust; the more they are far from the reference pixels, the more weight is added to the value of the pixel in the Gaussian smoothed image. For pixels that have values greater or lower than the reference dilated/eroded images, we apply this adjustment. For pixels in the extreme range (pixel value lower than 10 and greater than 245) but are not detected by the comparison with the dilated/eroded image, we simply add 0.3 on the initial value and 0.7 on the Gaussian-smoothed image value. With this primarily adjusted image, we apply a histogram matching with the reference frame. For the other frames which have low contrast, we adjusted them by histogram matching with a reference histogram generated by the first 155 frames. Figure 2.8 shows 2 frames before and after adjustment, which correspond to Figures 2.7b and 2.7c.



(a) **(b)** **(c)**

(d) **(e)** **(f)**

Figure 2.8: (a) Frame 160 (saturated) and (d) frame 200 (low-contrast) of stack 20 before the adjustment. (b) and (e) are the corresponding results after adjustment, (c) and (f) are histograms of the adjusted images.

Overall contrast enhancement

After adjusting images in stack 20 to be consistent, we applied an overall adjustment to all the stacks. A 3D Gaussian filter with $\sigma = 1$ was applied on each stack before this adjustment, to eliminate a part of noise. The following adjustment was

performed on 2D frames. One of the main problems in all of the stacks is low contrast. A contrast enhancement process is necessary for further operations including image binarisation. In a frame where lies the cross-section of an artery, the part we want to enhance the contrast is the arterial wall. If we apply an enhancement on the whole image with the same method, the ring artifact and the noise in the background inside and outside the lumen are also enhanced. In this step we adopt a contrast-based adaptive approach to enhance the contrast of the foreground while avoiding an enhancement of unwanted structures in the background. We first fix a standard deviation as a goal to adjust the 2D frames; here we set $std_{adjust} = 30$. Then we calculate a map of contrast for each image, where each pixel has a contrast indicator $C(p)$. This value $C(p)$ is defined by:

$$C(p) = I_{dilate}(p) - I_{erode}(p) \tag{2.28}$$

which is the difference between the dilated and the eroded image (with round structural elements of size 31×31) generated from the Gaussian-smoothed image. This value is normalized to $[0,\ 1]$, noted by C_{norm}. Instead of adjusting all the pixels intensities according to the desired $std_{adjusted}$, we change the intensity of a pixel only when its contrast indicator is greater than 0.25. For the rest of pixels, the contrast indicator is modified to $C'_{norm}(p)$ by a power function, with the coefficient γ based on a basic value $\gamma_0 = 0.1$. However, if the same is applied on all the stacks, then for those which have a lower overall standard deviation, this contrast stretching process is less effecient. Then, we modify the coefficient γ in the function by:

$$\gamma = \gamma_0 - 0.05 \cdot \frac{std_{adjusted} - std_{stack}}{std_{adjusted}} \tag{2.29}$$

where std_{stack} is the average of the standard deviations of frames selected from every 128 frames in a stack. The contrast indicator is modified by:

$$C'_{norm}(p) = \begin{cases} 0, & if\ C_{norm}(p) < 0.25 \\ C^{\gamma}_{norm}(p), & otherwise \end{cases} \tag{2.30}$$

To reduce the border effect caused by clipping, we applied a Gaussian smoothing on $C'_{norm}(p)$, for pixels of which the contrast indicator was 0 before this smoothing. The values were turned to 0 again after this smoothing. We note these smoothed values as $C'_{norm_g}(p)$.

Then, for each pixel p with intensity $I(p)$, the adjusted intensity $I_{adjusted}(p)$ is

given by:

$$I_{adjusted}(p) = (I(p) - \bar{I}) \cdot \left(\left(\frac{std_{adjust}}{std_{stack}} - 1 \right) \cdot C'_{norm_g}(p) + 1 \right) + \bar{I} \tag{2.31}$$

where $\bar{I}$ is the average of the intensities of frames selected from every 128 frames in a stack.

In some cases, the clipping point of contrast indicator C_{norm} set at 0.25 (Equation 2.30) is too high, which will cause a loss of details in some parts of the arterial wall. In this situation, we adjusted their clipping point to $0.25 - (1.0/std_{stack})$. This adjustment was performed by examining the results with a clipping point of 0.25. In stack 29, the air bubble is too contrastive compared to the arterial wall. We adjusted the clipping point to $0.25 - (2.0/std_{stack})$.

This adaptive contrast enhancement method enables us to adjust an image by enhancing only the contrastive regions where lie the structures that are considered as those needing a contrast augmentation, whereas details in the background are less or not enhanced. Figure 2.9 illustrates the results of different image enhancement methods. Row 6 is the result of RMSHE, which is also capable of keeping a low background contrast while improving the contrast of ROI. However, as is shown in Figure 2.10, in the zoomed-in image, some details of the ROI are erased and this is a modification that is not desirable.

2.2.3 Image binarization

The aim of the pre-processing steps is to generate well-binarized images that will be used in the lumen extraction process that follows. Thus it is important to preserve as much of the structure of the arterial wall while dividing pixels into foreground and background represented by white and black pixels. Our thresholding method is based on Otsu's method. For an adjusted image I_{init}, we first found its threshold T_{OTSU} by Otsu's method. Then for pixels having intensities greater than T_{OTSU}, we calculated their mean intensity noted as m_{OTSU}, and a new threshold T'_{OTSU} is calculated by:

$$T'_{OTSU} = \frac{T_{OTSU} + m_{OTSU}}{2} \tag{2.32}$$

In the binarization process, we also take the contrast in consideration since with the same threshold, the pixels in low contrast region might be lost during the binarization. We calculated the contrast indicator $C(p)$ as in the last section of overall

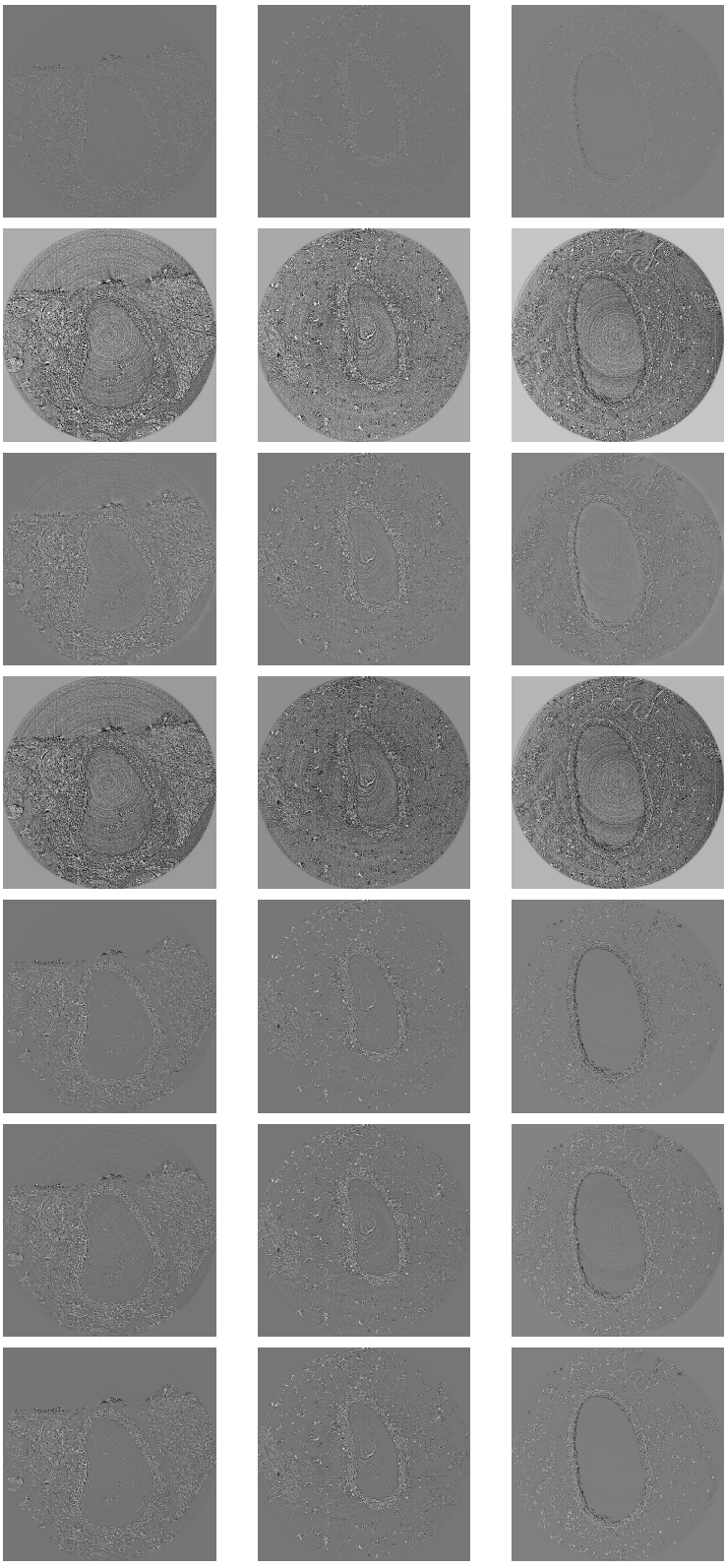
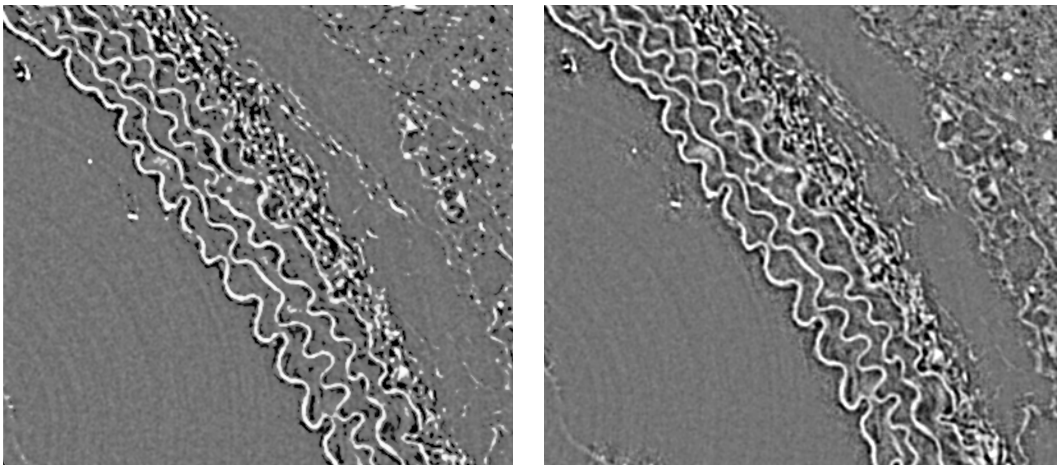


Figure 2.9: Row 1: frames from Gaussian-smoothed stack 4, 14 and 20. Row 2: images adjusted by HE. Row 3: images adjusted by CLAHE. Row 4: images adjusted by MMBEBHE. Row 5: images adjusted by RMSHE. Row 6: images adjusted by stretching the adjusted standard deviation to 30. Row 7: images adjusted by the proposed method; the background details are less contrastive.



(a) **(b)**

Figure 2.10: A part of a frame adjusted by RMSHE (a) and the same part adjusted by the proposed method (b).

contrast enhancement with Equation 2.28; the results are in the range of [0, 255]. The binarization process can be expressed by:

$$I_{bin}(p) = \begin{cases} 255, & if\ I_{init}(p) > T_{OTSU}\ and\ I_{init}(p) > T'_{OTSU} - \frac{255-C(p)}{2} \\ 0, & otherwise \end{cases} \quad (2.33)$$

where I_{bin} is the binary image. As in the contrast enhancement step, we processed stack 29 differently. We divided T_{OTSU} with 1.1 and used this new value as T_{OTSU} in the operations.

2.2.4 Selection of large connected components

With each stack composed of binary frames, in general, we got the foreground objects represented by white pixels and the background in black. However, these foreground pixels are not only those of the arterial wall and perivascular tissues but also contain impurities in the lumen which are mainly composed of agarose. The main operation in this process is a selection of the largest connected component in 3D by grouping consecutive frames to perform a primary elimination of undesired components, this process can eliminate part of the agarose. We grouped each 32 consecutive frames into one group and label all the connected components in 3D. In each group, the largest component was preserved and the rest of the voxels was eliminated. Figure 2.11 illustrates two pairs of frames from two different stacks before and after this operation. These binary images are results of the binarisation step

described in previous section, after a selection of connected components; impurities inside the lumen are partially removed. We also illustrate a 3D reconstructed image of the first 250 frames of the preproccesed image of stack 0 in Figure 2.12 to show a global view of the result. We note that in stack 35, there are 32 consecutive frames of which the largest component is not the arterial wall. This false selection was manually corrected.

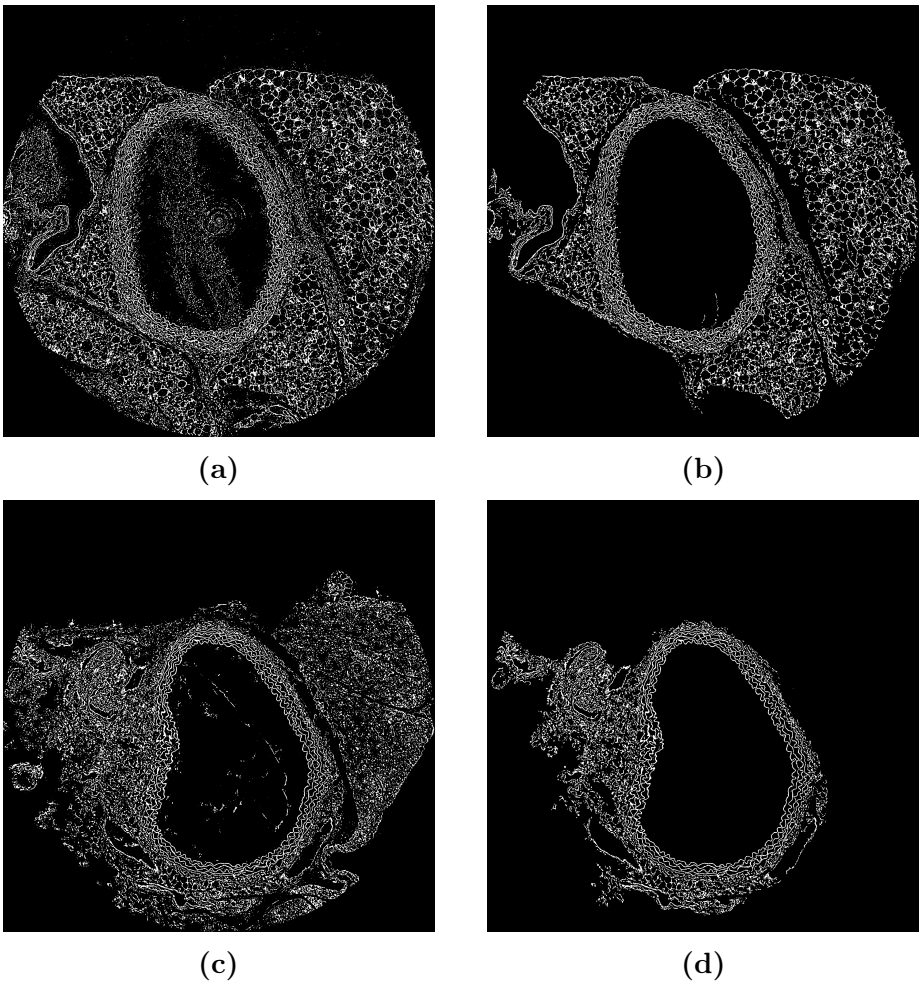


Figure 2.11: (a) and (c): binary images generated by operations described in section 2.2.3. (b) and (d): corresponding frames after a selection of the largest component in 3D. These two frames are from stack 0 and stack 4 respectively.

2.2.5 Conclusion

In this chapter, we first analyzed the features of stacks and described each stack by noting the existence or the level of these features. Some of the features commonly exist in all of the stacks; some of them appear in several stacks. There are also

Figure 2.12: 3D reconstruction of the first 250 frames of pre-processed stack 0.

some features that are not typical and were generated accidentally in certain frames of a certain stack. Features such as contrast are concerned in the pre-processing step; those about the existence of agarose are concerned in the later segmentation steps. Before the general pre-processing of all the stacks, we first processed abnormal frames in a stack to adjust them and make them have similar contrasts as the reference frames (which are considered relatively normal frames in the stack), then we applied a pre-processing composed by denoising, image enhancement, binarization and selection of large connected components. The denoising operation and the connected components selection were performed in 3D, the other operations were performed in 2D. The image enhancement improved the contrast of the high-contrast regions, which are more likely to be ROIs; at the same time, features in the background were not much enhanced. The binarisation was done based on Otsu's method. The goal of this binarization is not a segmentation of elastic lamellae, but to preserve pixels in the foreground and eliminate those in the background. We hope that the preserved pixels can form a shape of the lumen. Finally, a selection of large connected components was applied to eliminate small objects. Since the agarose is a component that is not as dense as the arterial wall, groups of frames were processed in 3D. This operation could remove part of the agarose. The pre-processed binary result enables us to perform a extraction of the shape of lumen that is described in the next chapter.

Chapter 3

Lumen segmentation

By observing all the stacks, we found that the inner border of elastic lamellae is more apparent than the outer border because the outer border of elastic lamellae is the inner border of the *tunica adventitia* which is tightly connected to the *tunica media*, where lie elastic lamellae. For this reason, the segmentation of elastic lamellae starts from the segmentation of the lumen, in other words, the extraction of the inner contour of the elastic lamellae. There exist several works of 3D lumen segmentation. In these studies, the aim of segmentation is to obtain the lumen contour. However, in our work, the lumen is surrounded by the first elastic lamella. The extraction of the lumen contour is the extraction of the inner contour of this lamella, which is much curvier than the lumen contours in most of the related studies. The result of this lumen segmentation step is not the final result, but an intermediate one that can help us focus on the region where lie the *media* and the *adventitia* in order to extract finer structures.

3.1 Related works

Many studies have been conducted on 3D lumen segmentation methods. Lesage *et al.* [89] did a review of common 3D vessel lumen segmentation techniques, summarizing them from 3 aspects: (1) appearance and geometric models; (2) image features; (3) extraction schemes. From the first aspect, luminance properties can be expressed by appearance models and the specific shapes of vessels, which is a key characteristic, can be expressed by geometric models. The second aspect concludes the vessel-dedicated features extracted by detectors or filters such as Canny detector. Extraction schemes include three general classes: region growing algorithms, active contours and centerline-based approaches. Region growing performs the segmenta-

tion from seed points and makes the region grow until some criteria are met; active contour uses an evolving interface to perform the segmentation; centerline-based methods directly extract the centerline instead of finding the contour of the vessel. In our images, for the appearance part of the first aspect, we can conclude that the arterial wall, perivascular tissues and impurities are brighter than the lumen. For the geometry, the arteries are cylinders with a slight curvature in the vertical direction and a changing cross-sectional area. Bifurcations also appear in most of the cases. From the image features aspect, our image samples that have only one artery in each stack with the lumen taking up nearly a quarter of the image space give us 2D cross-sectional features. The shape of the lumen is easier to find from the view inside the vessel other than from the vessel's external side. From the extraction scheme aspect, the method that we applied is closer to region growing algorithms. The aim of this lumen segmentation is not to extract the precise contour but to segment the region composed of connected regions that meet a criterion. Here we introduce several related works that segment the lumen from an intravascular view mostly involving region growing schemes, which is similar to our case where the lumen is segmented from the inside on consecutive frames.

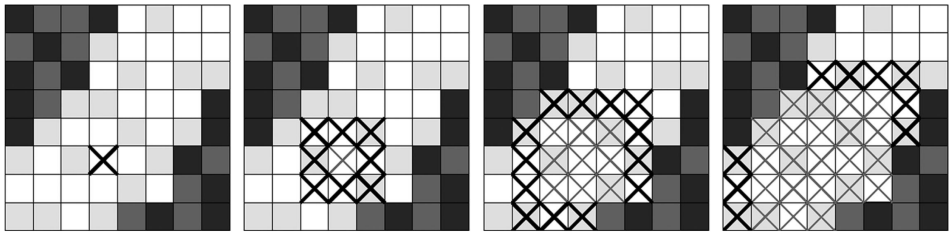


Figure 3.1: Region growing process applied in the work of Boskamp *et al.* [90]. The segmentation starts from a manually selected seed point. Neighboring voxels are added until criteria are met.

Boskamp *et al.* [90] proposed a method to quantify and visualize vessels in CT and MR (magnetic resonance) angiography data. For an extraction of the centerline, a region growing-based vessel segmentation is first performed with seed points manually selected in the vessels of interest; the process is illustrated in Figure 3.1. The segmentation starts from seed points, more and more neighboring voxels are included as long as they satisfy the inclusion criteria. Three criteria are applied: (1) a lower gray value threshold; (2) an upper threshold combined with a size limit in order to avoid including significant bright structures; (3) a gray level difference threshold for neighboring voxels. Brathwaite *et al.* [94] used a region growing-based method including a correction of leakage by component-labeling with dilation-erosion to seg-

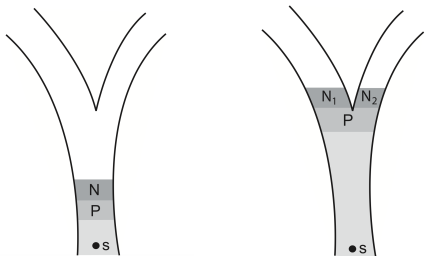


Figure 3.2: Bifurcation detection method in the work of Metz *et al.* [91]. Region growing starts from seed point s. P is the region grown in two previous region grow iterations; N, N_1 and N_2 are regions grown in the last two iterations. In (b), N_1 and N_2 are not connected; thus they are assumed to be part of a bifurcation region.

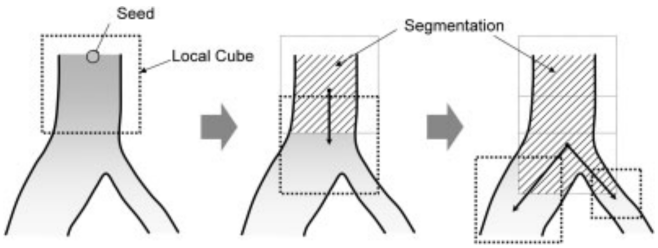


Figure 3.3: Local cube tracking in the work of Yi and Ra [92], depicted in 2D space. Cubes are generated along the vessel with the length of every edge adapted to the diameter of the selected vessel area.

ment the lumen from IVUS images. Metz *et al.* [91] proposed a method that is able to detect bifurcations and leakages during the region growing process for vessel segmentation used for centerline extraction in CT angiography data. As is illustrated in Figure 3.2, a connected component analysis is performed after every two region growing iterations. If the voxels added during these two iterations are not connected, it is considered that there is a bifurcation. Otherwise, no bifurcation is detected. Leaked region detection is realized by setting a threshold of the number of voxels added during every two iterations. A segmented region is considered a leakage into non-luminal regions when the threshold is exceeded. Yi and Ra [92] proposed a vascular segmentation algorithm for 3D CT angiography data by locally applying a competitive region growing method [95] on consecutive cubes. As is shown in Figure 3.3, local cubes are placed along a vessel to track its trajectory, the next local cubes locations are estimated based on connected components labeling. Two examples with three and two connected components are shown in Figure 3.4; the center location of each next cube is determined by the locations of the voxels of each connected component. The vascular part in each cube is segmented by a com-

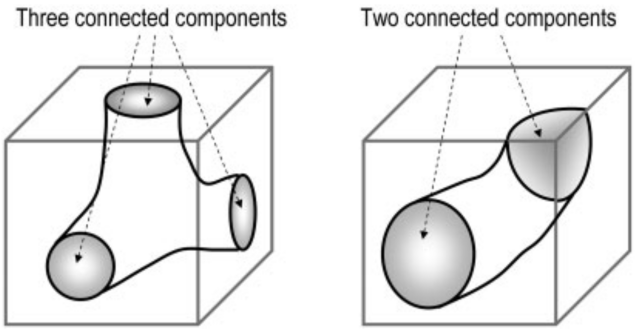


Figure 3.4: Connected component labeling in the work of Yi and Ra [92]. The voxels on these connected components determine the location of the next local cubes.

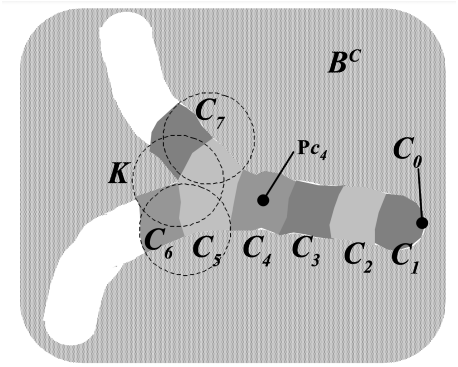
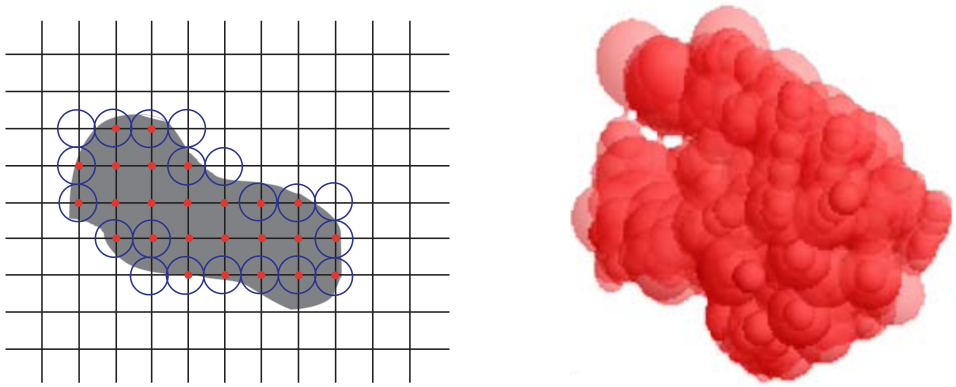


Figure 3.5: Shape-based region growing model proposed by Masutani *et al.* [93]. C_i is the i-th cluster formed by voxels of the i-th growth generation. P_C is the cluster's center of gravity. K is the kernel for a bounded morphological operation. B^C is the complement of a boundary shape.

petitive region growing method that starts from multiple seed points belonging to different regions. The seed area for the vessel object is obtained from the segmented vessel from the previous cube. The nonvessel seed area is extracted based on a minimum-path analysis and a cost histogram analysis, the latter is able to distinguish the seed for unwanted objects by finding peaks in the histogram. Masutani *et al.* [93] proposed a vascular segmentation method with shape-based region growing. Bounded space morphological operations are defined with the help of a boundary shape B and its complement B^C. Space B is obtained by thresholding. With a seed positioned, two processes are iterated: simple growing which is realized by bounded space dilation, and growth-front smoothing which is realized by bounded space closing. This shape-based region growing model is illustrated by Figure 3.5, where the set of voxels belonging to the same growth generation is defined as a cluster. Rivera-Rovelo and Bayro-Corrochano [96] proposed a medical image segmentation and modeling method for tumors in CT images. In the segmentation

step, a region growing process is performed from the seed points extracted with the help of texture information characterizing each pixel with a vector. The criterion of growth is defined with the texture vectors and the intensities of pixels. In the volume representation step, the shape of an object is represented with the help of a marching cubes algorithm. Instead of using cubes, spheres are used to define the cases and to find the surface. A 2D illustration is shown in Figure 3.6a. Figure 3.6b shows an approximation of an extracted tumor.



(a) (b)

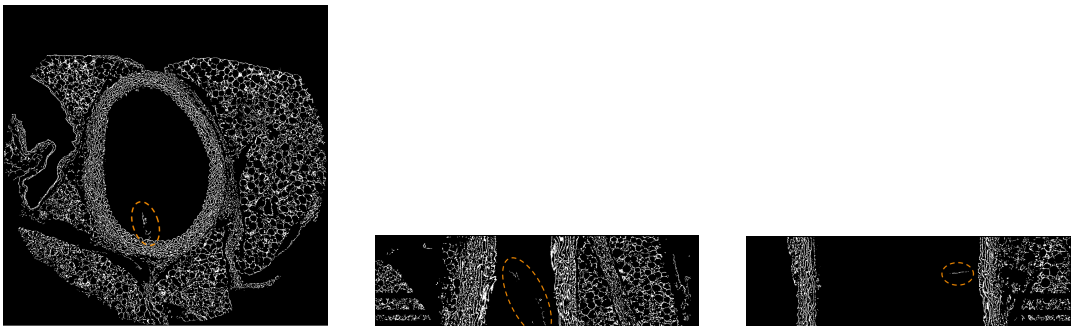
Figure 3.6: Method and result of volume representation in the work of Rivera-Rovelo and Bayro-Corrochano [96]. (a) A 2D representation of the marching cubes approach based on circles (spheres in 3D). (b) A representation of the extracted tumor shape.

3.2 Lumen cleaning

One of the disturbances to the lumen segmentation of our images is the existence of agarose. Although, in the work of the previous chapter, a selection of large 3D connected components already eliminated some of them, those adhering to the inner wall of the aorta may not be fully removed. The disconnection between the agarose and the arterial wall is not guaranteed in each group of consecutive frames. Thus this property was just used in the preprocessing to facilitate the cleaning of the lumen but is not the final operation of the lumen cleaning process.

One of the important properties in each entire stack is the continuity of consecutive frames. This continuity is along the Z axis since the arteries were placed along this direction in the image acquisition process. Compared to the artery, the agarose shows a greater discreteness, but this is particularly noticeable along the Z axis direction. No matter if bifurcations exist in a stack, the arterial wall is always

an entirety, where the evolution of shape is vertically continuous, while it is not the case for agarose. As is shown in Figure 3.7, from the vertical cross-section of a 3D binary image, we can observe the difference between the arterial wall and the agarose inside the lumen. To deal with the existence of agarose, we take advantage of the vertical discontinuity of the agarose to eliminate it.



(a) **(b)** **(c)**

Figure 3.7: Illustration of agarose in the binary image. (a), (b) and (c) are an axial, a coronal and a sagittal plane of a segment of a 3D image respectively.

3.2.1 Pre-processing

The impurities in the lumen are mainly agarose, but air bubbles also exist in several frames of certain stacks. The process in the last chapter can eliminate part of the agarose, but for air bubbles that have continuous, high-contrast borders and that are usually coherent on the inner contour of the arterial wall, this operation can barely affect its existence. Thus, before a general agarose elimination process for all stacks, we first performed an air bubble-eliminating process to facilitate the following processes. Figure 3.8 shows a stack with air bubbles. The air bubbles in our images have high contrasts compared to the other structures. In some frames, these bubbles do not touch the elastic lamellae; in some frames, they touch the first lamella but are not fused with the lamella, which means the contacting part has a greater thickness than that of the lamella; while in the other frames, the bubble and the lamella are fused and the thickness of the contacting part is almost the same as the lamella.

We observed that in most of the frames with air bubbles, the inner side of the high-contrast bubble is darker than the other structures in the image. We first calculated the contrasts on the whole Gaussian-smoothed gray-scale images that we obtained in the first step of pre-processing in Chapter 2, by calculating the difference

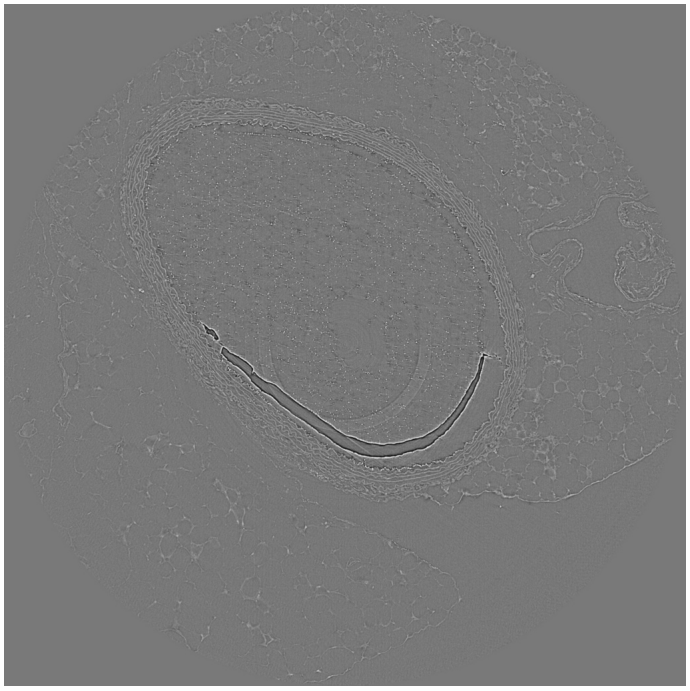


Figure 3.8: A frame with an air bubble with high contrast and a dark inner side.

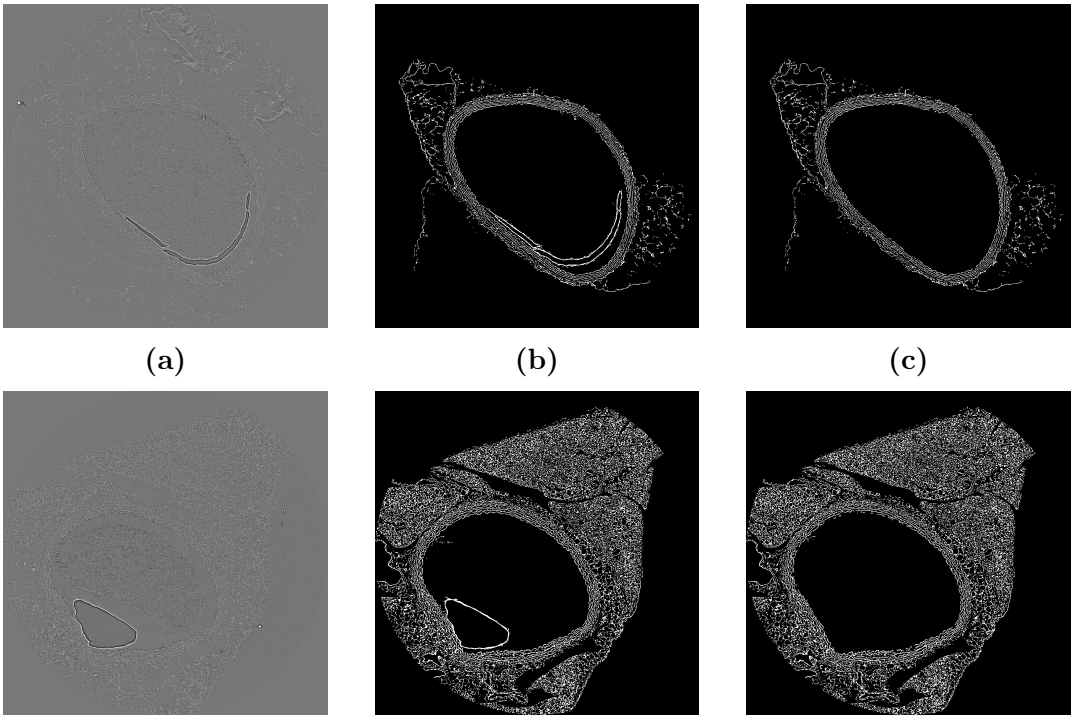


Figure 3.9: Results of the air bubble-removing process in two stacks. Column 1: Gaussian-smoothed images. Column 2: binary images obtained by a selection of 3D connected components. Column 3: binary images after the air bubble-removing process.

between the dilated and the eroded image with a round structural element of radius 10. Then we preserved the pixels that have a difference greater than 100 and an intensity lower than 50. With these low-intensity pixels in high-contrast regions, we performed a dilation with a round structural element of radius 10 on these high-contrast pixels. We eliminated all the pixels in this dilated image from the binary images obtained in the last step in Chapter 2. Finally, we applied again a selection of the largest 3D connected component every 32 frames to these processed images.

Figure 3.9 shows the results of the bubble-removing process of stack 28 and stack 37. The bubbles in these two cases have different shapes. In the second case, the bubble has more contact with the elastic lamella. In both of two situations, the general shape of the lumen can be preserved.

3.2.2 Method

After the preprocessing of removing air bubbles, we performed the lumen cleaning process with the information on the density difference between the arterial wall and the lumen. We used a sphere moving all around the 3D stack and preserved the spaces where there exist enough voxels. Besides this constraint of voxel density, the continuity along the Z axis can be used by preserving the centers that are in a range based on the results of previous frames. This range is composed of the positions generated by a dilation of the previously preserved ones. Thus for a 3D stack, the lumen cleaning operation was performed from the first frame to the last frame.

3.2.3 Implementation

A sliding sphere with a radius R was used to select the 3D spaces that have enough voxels inside the sphere. We first calculated the volume of a sphere $V_R = \frac{4\pi R^3}{3}$ and preserved a center O only when:

$$Card(S_{O,R}) > \frac{V_R}{8} \tag{3.1}$$

where S_{O_R} is the set of white pixels that are contained in the sphere:

$$S_{O,R} = \{p \mid I_{Binary_{3D}}(p) > 0,\ \|(p - O)\|_2 \leq R\} \tag{3.2}$$

where $I_{Binary_{3D}}$ is the 3D binary image obtained by the pre-processing step. For centers O that are at the start and the end of a stack, where the space around the center is not enough for an entire sphere, we replaced V_R by $V_{R_{part}}$ in Equation 3.1.

We used a Cartesian coordinate system to express the location of each pixel. The point (x, y, z) is located in frame z at row y, column x. This partial volume is calculated by:

$$V_{R_{part}} = \begin{cases} \sum_{x\in\mathbb{Z},\ y\in\mathbb{Z},\ z\in\mathbb{Z},\ z\geq 0}^{z\leq z_c+R} 1, & if\ z_c < R \\ \sum_{x\in\mathbb{Z},\ y\in\mathbb{Z},\ z\in\mathbb{Z},\ z\geq z_c-R}^{z\leq z_{max}} 1, & if\ z_c > z_{max} - R \end{cases} \tag{3.3}$$

where z_c is the coordinate of axis Z of the sphere center O and z_{max} is the maximal coordinate of axis Z of the stack.

To reduce the calculation cost, centers were tested for every R pixels in X and Y directions and for every $R/2$ pixels in Z direction. To take advantage of the continuity in Z axis, for $S_{O_{z_1},R}$, which are centers preserved at $z = z_1$, candidate centers for $z_2 = z_1 + R/2$ to check are those which have a distance of R from $S_{O_{z_1},R}$ in X or Y direction. Thus when $S_{O_{z_1},R}$ is known, preserved centers for $z_2 = z_1 + R/2$ are those satisfying:

$$Card(S_{O_{z_2},R}) > \frac{V_R}{8} \tag{3.4}$$

where:

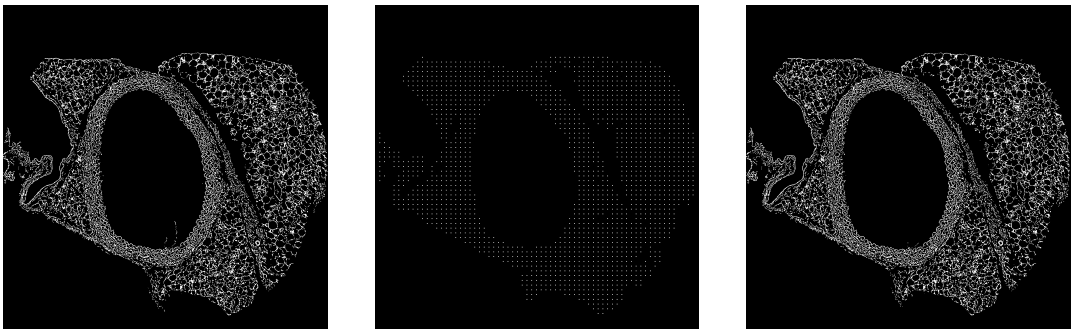
$$\begin{aligned} S_{O_{z_2},R} = \{p \mid I_{Binary_{3D}}(p) > 0,\ \|(p - O_{z_2})\|_2 \leq R, \exists O_{z_1} \in S_{O_{z_1},R} \\ s.t.\ x_{O_{z_2}} = x_{O_{z_1}} \pm R\ or\ y_{O_{z_2}} = y_{O_{z_1}} \pm R\} \end{aligned} \tag{3.5}$$

We noted the preserved centers and generated sphere volumes of radius R centered by these voxels. All the generated voxels form a general shape around the lumen. Figure 3.10 illustrates a binary result of the last step (selection of 3D connected components) in Chapter 2, the sphere centers preserved in this frame and the preserved pixels generated by these centers. Figure 3.11 shows the result of lumen cleaning corresponding to Figure 3.7. The parameter R was set to 30 in our implementation. Fig 3.12 compares the 3D image before and after this lumen cleaning process.

3.3 Lumen segmentation

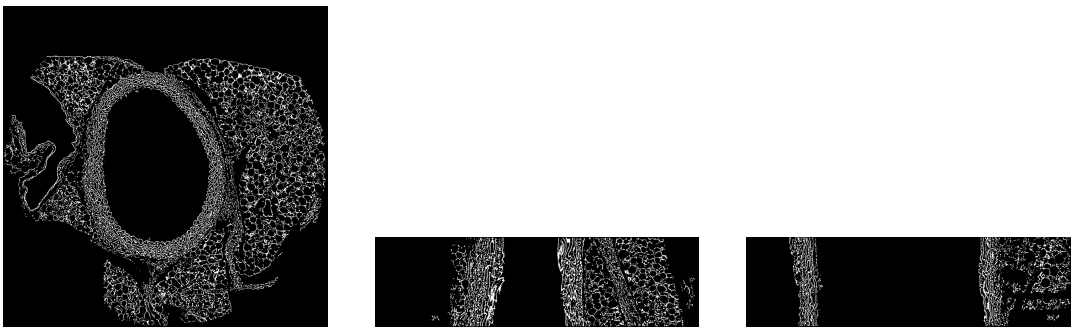
3.3.1 Method

In our work, we have a more inside view of the lumen, which is different from common vessel lumen segmentation tasks. Moreover, the outer border of the artery



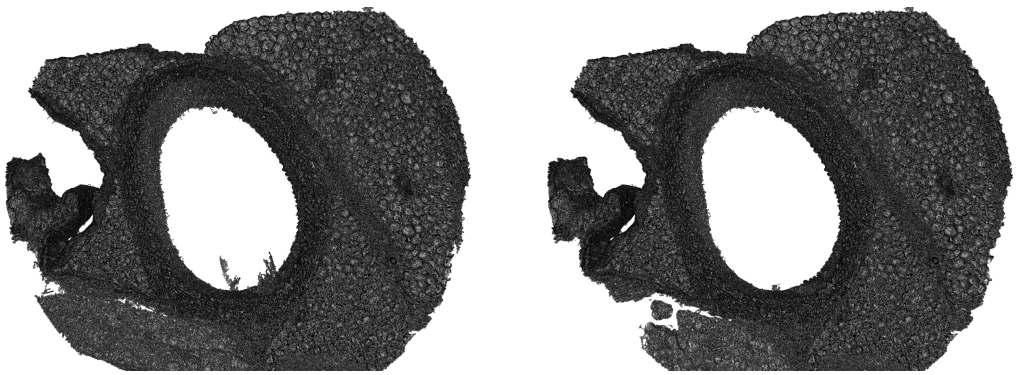
(a) **(b)** **(c)**

Figure 3.10: Lumen cleaning process on a frame. (a) is a frame with agarose on the inner vessel border, which may affect the lumen extraction process in the next section. (b) shows the sphere centers of this frame, which are used for generating spheres and preserving the pixels inside the generated spaces, as is shown in (c).



(a) **(b)** **(c)**

Figure 3.11: Illustration of a part of the lumen after an impurity-cleaning process in the binary image corresponding to Figure 3.7. (a), (b) and (c) are an axial, a coronal and a sagittal plane of a segment of a 3D image respectively.



(a) **(b)**

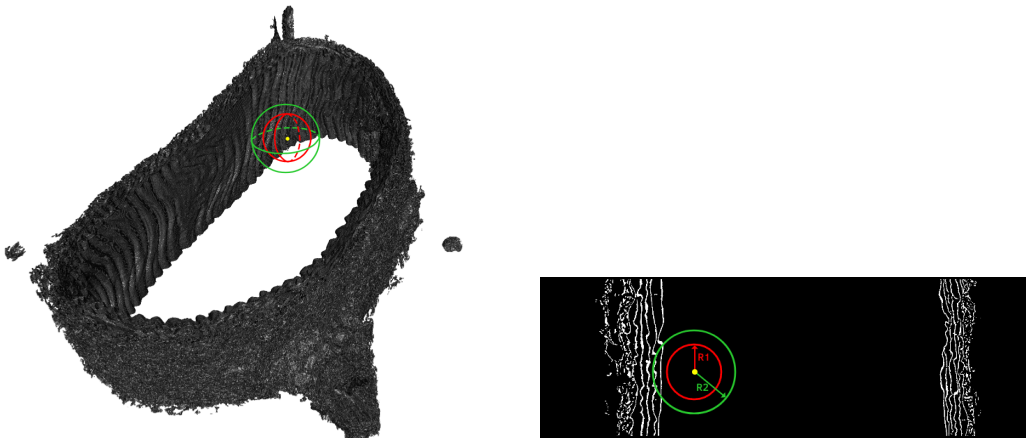
Figure 3.12: 3D illustration of a part of the artery before (a) and after (b) the lumen cleaning.

is not apparent with the surrounding tissue. Thus, instead of an outer view of the vessel shape, what we have is a stack of cross-sections. The area of the cross section is about a quarter of the 2D image size. The aim of the lumen segmentation step is not to extract the precise inner contour of the lumen, in other words, not to extract the innermost layer of elastic lamellae, which is curvy and has a convex or concave shape everywhere. This step aims to extract the general shape of the inner contour that is smoother than the first elastic lamella, which allows us to unfold the vessel wall (as will be described in Section 3.4).

The scheme came from the method of region growing and the extraction of connected components, which use the similarities between neighboring pixels to grab a number of them. The latter can be seen as a special case of region growing, where pixels have only two possible values and the tolerance between the neighboring pixels is set to 0. If we apply this method directly to our 3D binary images, not only it extracts the precise inner contour that we do not need in this step, but it also fails to extract only the lumen region without making the grown region spread to the spaces between lamellae. Since the lamellae are not closed/continuous everywhere, once there is a "hole", the seed can spread throughout it. If we apply region growing on gray-scale images, a similar problem occurs. In addition, in regions where pixel intensities change slowly but continuously, the region growing method can lead to a false pixel spreading. According to the above analysis, the problem comes from the scale of the growing element, in the original region growing method. This element is a pixel, with the assumption that the "wall" or the boundary between the ROI and the rest of the image is compact enough to prevent pixels from spreading to the other side of this boundary. In our case, the boundary is not closed at pixel level, but if we increase the scale and use a larger element to spread the ROI, the "wall" built by the elastic lamellae is compact enough to keep this element inside the lumen. According to this idea, we can imagine a rolling sphere moving inside the lumen. All the regions that the sphere contains while moving constitute the area of the lumen, and the surface that the sphere touches are pixels on the inner surface of the lumen.

We used a hollow and thick sphere that moves inside the lumen to build the 3D ROI. The reason for choosing a hollow sphere is to guarantee the robustness of small elements in the lumen that were not eliminated in the lumen cleaning step. Using a solid sphere will make it avoid touching every tiny impurity in the lumen and this can result in a ROI with unwanted empty spaces. With a hollow sphere, impurities that can be contained in the sphere will not affect the trace of the moving sphere.

The reason for using a sphere with a thickness is to make the method robust to the noise and impurities near the first elastic lamella. The thickness creates a tolerance; without this tolerance, once the surface of the sphere touches a noise or impurities on the inner surface of the lumen, it will stop going further and stop getting closer to the surface. With tolerance, the sphere can roll on the inner surface while ignoring the disturbances that prevent it from getting closer to the surface. Figure 3.13 is an illustration of this strategy.



(a) **(b)**

Figure 3.13: Illustration of the lumen extraction strategy in 3D (a) and in 2D (b). Two concentric hollow spheres move inside the lumen. Voxels inside the smaller sphere are not counted. The number of voxels between the two spheres is limited, which enables tolerance of impurities near the inner border of the arterial wall.

3.3.2 Implementation

This method was realized on 3D stacks. An intervention was needed for each stack for selecting the location of the lumen by choosing a point from a 2D frame. The thick sphere has two radii R_1 and R_2 with $R_2 = R_1 + \epsilon$; The thickness of the sphere is ϵ, the space with distances to the center d less than R_1 is empty.

To reduce the calculation amount and to specify the location of the lumen, we initialize a mask for one frame of each stack. We note the 3D image obtained from the lumen cleaning process as $I_{Artery_{3D}}$. The initialization is done on frame $z = R_2$. In this process, a hollow thick sphere moves with the sphere center located in frame $z = R_2$. Every time the number of pixels contained in the space between the two

spheres with radii R_1 and R_2 is checked, we preserved sphere centers that satisfy:

$$Card(S_{O_{R_1,R_2}}) < \frac{V_{O_{R_1,R_2}}}{16} \tag{3.6}$$

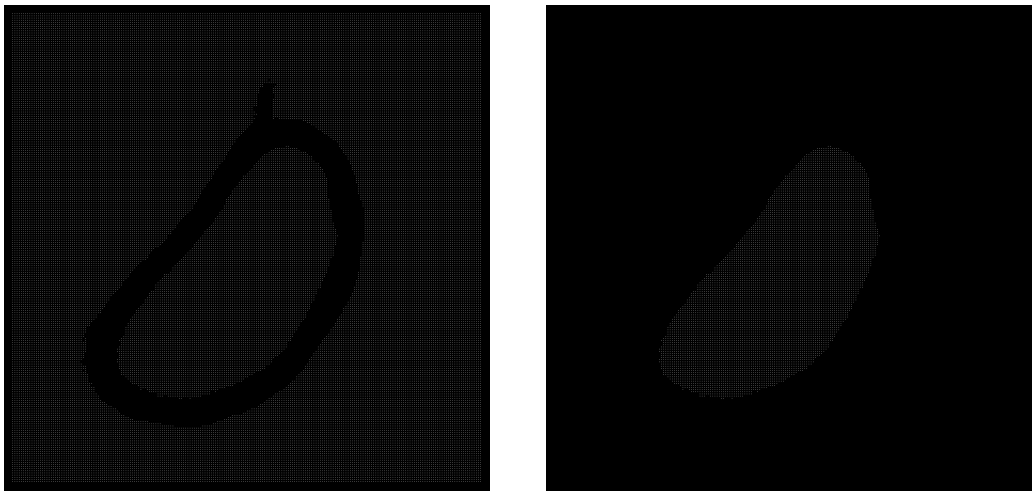
where O is the sphere center, $V_{O_{R_1,R_2}}$ is the number of pixels if the space between the two radii is filled with pixels, and $S_{O_{R_1,R_2}}$ is the set of white pixels that are contained in this space:

$$S_{O_{R_1,R_2}} = \{p \mid I_{Artery_{3D}}(p) > 0,\ R_1 < \|(p-O)\|_2 \leq R_2\} \tag{3.7}$$

All the centers that satisfy Equation 3.6 were preserved. To reduce the calculation amount, the step of the moving sphere was set to $s = R_2/4$ in X and Y directions. Figure 3.14 is an example of these points. With these points, an intervention is then needed to select the group of pixels of the lumen. We manually pointed a point p_c inside the lumen in frame $z = R_2$ and preserved a group of connected pixels that includes p_c. Here "connected pixels" means pixels that have a distance of $s = R_2/4$ in X or Y direction.

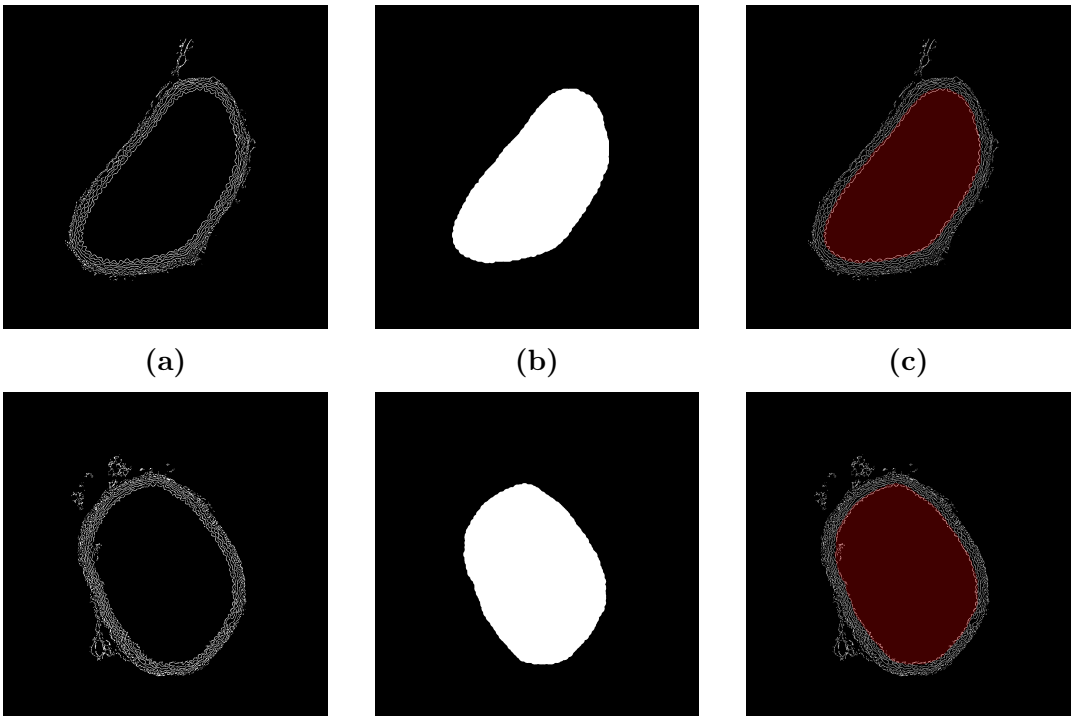
With this initialization, we can find the sphere centers in 3D space. Since the modification of the lumen area is progressive along the Z axis, we do not need to generate center pixels from zero in every frame. With the sphere centers C_i in frame $z = i$, to find the sphere centers in frame $z = i + 1$ or $z = i - 1$, we can apply a dilation and an erosion on C_i to generate $C_{i_{dilate}}$ and $C_{i_{erode}}$, and find the new lumen contour in the space in between. In this way, only the candidate pixels appearing in $C_{i_{dilate}}$ but not appearing in $C_{i_{erode}}$ need to be checked. The sphere centers in frames $z < R_2$ and $z > R_2$ were then generated from the initialized centers in frame $z = R_2$. To reduce the calculation amount, the step of the moving sphere was set to $s = R_2/2$ in Z direction. For images where there exist connected components other than the lumen region, we used vertical continuity to preserve only the component of the lumen.

Figure 3.15 illustrates the result of lumen extraction on two stacks. The first row is a stack with a cleaning lumen, the second row is a stack with impurities in the lumen although a lumen cleaning process has been performed. The use of hollow spheres with a tolerance ensures robustness when there exists some impurity near the inner border of the vessel. Figure 3.16 shows all the sphere centers corresponding to the stack of the first row in Figure 3.15 after performing the process described above as well as the generated lumen space. The small branch in the lower right part of the image was generated by a bifurcation.



(a) **(b)**

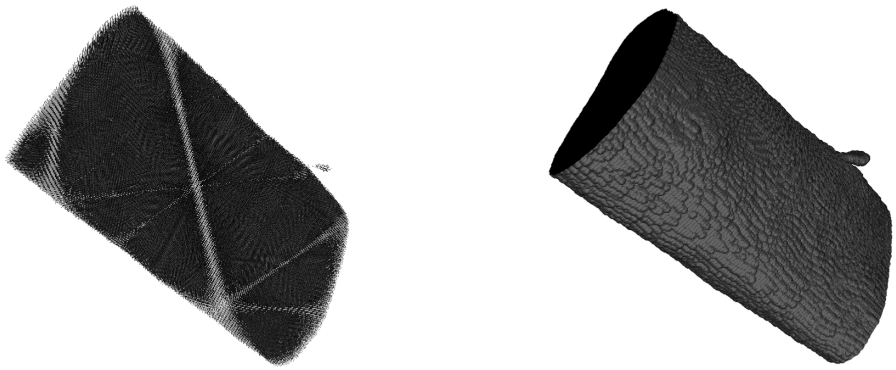
Figure 3.14: Initilization of the sphere centers. (a) shows initialized centers found on frame $z = R_2$, (b) is the group of selected centers among them.



(a) **(b)** **(c)**

(d) **(e)** **(f)**

Figure 3.15: Results of lumen extraction on stack 6 (row 1) and stack 20 (row 2). (a) and (c) are 2 frames of cleaned aorta images. (b) and (e) are lumen spaces generated by the sphere centers of the corresponding frames. (c) and (f) are the superimposed images of the two first columns. In stack 20, the outer contour of the extracted lumen does not bypass the impurity in the upper left part of the lumen.



(a) **(b)**

Figure 3.16: (a) A 3D view of the sphere centers of stack 6 (corresponding to the first row of Figure 3.15), which is sparse because only a part of voxels were checked. (b) The shape of the lumen generated by the sphere centers in (a).

3.4 Arterial wall unfolding

3.4.1 Lumen contour extraction

With the mask of the lumen in each frame, we can extract its outer contour from the binary image. The selection of the contour is for straightening the arterial wall along it for a finer segmentation, thus, this contour should indicate a path with all the pixels on it sequentially noted by numbers. Since the shape and the area of the lumen change along the Z axis, we grouped consecutive frames and assigned each group of frames a shared contour. We chose 32 as the number of frames of one group and used the contour of the $16th$ frame as the shared contour. The contour was obtained by extracting the contour of the lumen in this frame, and the points on this contour were enumerated.

Contours with a width of 1 were extracted from the lumen image of each frame. For a contour $\mathcal{C}$ composed of positive pixels of the contour image on a reference frame, we first select an arbitrary point p_0 on $\mathcal{C}$. Then we remove p_0 from $\mathcal{C}$ and check the neighbors of p_0 to decide the second point p_1. Points are receded sequentially until there is no point on $\mathcal{C}$.

3.4.2 Normal vectors computation

For each point of the contour, a normal vector was calculated with the help of neighboring frames. With a sphere of radius $R = 30$, a normal vector $\overrightarrow{n}(p)$ corresponding to a point $p(x, y, z)$ was calculated by adding all the vectors with a

start point on p and an end point in the sphere and normalizing the sum, which can be formulated as:

$$\overrightarrow{n}(p) = \frac{\sum \overrightarrow{p\ p_s}}{\|\sum \overrightarrow{p\ p_s}\|},\ \|\overrightarrow{p\ p_s}\| \leq R \tag{3.8}$$

The projection of this vector on the frame where $p(x, y, z)$ lies is:

$$\overrightarrow{n}_{XY}(p) = \frac{\sum \overrightarrow{(p\ p_s)}_X + \overrightarrow{(p\ p_s)}_Y}{\left\|\sum \overrightarrow{(p\ p_s)}_X + \overrightarrow{(p\ p_s)}_Y\right\|},\ \|\overrightarrow{p\ p_s}\| \leq R \tag{3.9}$$

where $\overrightarrow{(p\ p_s)}_X$ and $\overrightarrow{(p\ p_s)}_Y$ are the projections of $\overrightarrow{p\ p_s}$ on axis X and Y respectively. Figure 3.17 shows a lumen contour of a frame in stack 6 with a part of the normal vectors labeled.

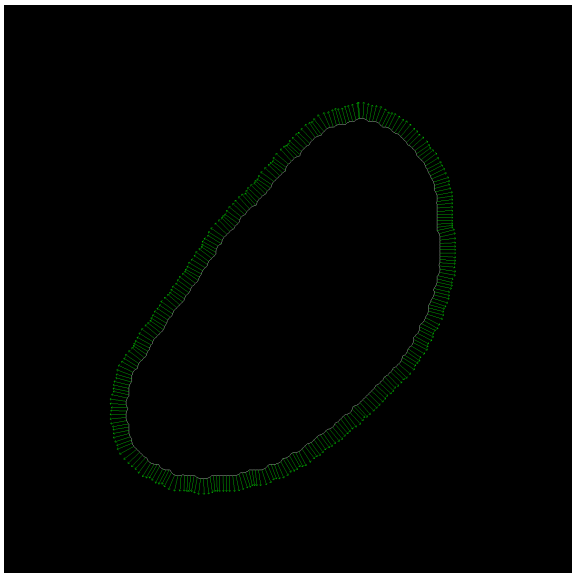


Figure 3.17: Contour of the lumen of a frame in stack 6. White pixels form the extracted contour of the lumen; green arrowed lines show the orientations of the normal vectors.

3.4.3 Image unfolding

We calculated the normal vectors on each point of the contour of the frame in the middle of each group (the 16th frame of a group of 32 frames) and unfolded all the images of this group with this reference contour labeled with a pixel order and the corresponding normal vectors.

For a grayscale image I_o, we got the contour $\mathcal{C}$ of the lumen with every point on it noted with an order and a corresponding normal vector with direction. The artery unfolding process can be considered as rotating slices with a width of 1 and stitching them horizontally. The desired unfolded image I_{uf} is a rectangle with a height H_{uf} covering elastic lamellae and a width that is equal to the length of the lumen contour. For a slice i, the unfolded image is I_{uf_i}. We can note the H_{uf} pixels $p_{uf_i}(x, y)$ from $p_{uf_i}(0, 0)$ to $p_{uf_i}(0, H_{uf}-1)$. These pixels lie vertically on a slice. This slice was rotated with the center $p^c_{uf_i} = p_{uf_i}(0, \frac{H_{uf}}{2})$. This center corresponds to a pixel $p^i_{o_c}(x^i_{o_c}, y^i_{o_c})$ in I_o located on the contour lumen $\mathcal{C}$. With the direction θ_{p_o} of the normal vector corresponding to rotation center $p^c_{uf_i}$, we can calculate the rotation matrix $\mathcal{M}^i_{uf\to o}$ that maps the pixel $p_{uf_i}(x^i_{uf}, y^i_{uf})$ to the initial pixel $p_{o_i}(x^i, y^i)$ on I_o:

$$\begin{bmatrix} x^i_o \\ y^i_o \\ 1 \end{bmatrix} = \mathcal{M}^i_{uf\to o} \begin{bmatrix} x^i_{uf} \\ y^i_{uf} \\ 1 \end{bmatrix} + \begin{bmatrix} x^i_{o_c} \\ y^i_{o_c} - \frac{H_{uf}}{2} \\ 0 \end{bmatrix} \tag{3.10}$$

With this correspondence of pixels positions, we can map the pixel intensities from I_o to I_{uf}. H_{uf} was set to 128 in our case. Figure 3.18 illustrates an unfolded arterial wall as well as some corresponding pixels on the initial image.

3.5 Additional explanation of lumen extraction process

The above lumen cleaning and extraction operations are based on an assumption that a stack, after the last step in Chapter 2 (and the bubble elimination step for certain stacks), contains agarose that may affect the lumen extraction. For some stacks, the lumen is clean enough after the pre-processing in Chapter 2. For such data, we could apply the lumen extraction directly on these results, without the lumen cleaning process. Meanwhile, some stacks contain dense impurities that need an additional agarose cleaning process after the described one with a sliding sphere. We used the continuity between frames to eliminate these impurities by removing the pixels in frame $i + 1$ that do not exist in the dilated image in frame i. The same operation in the inverse direction was also applied. We also found that it is possible to apply this additional agarose-eliminating process directly on some stack after the pre-processing. This kind of stacks have just a few impurities to eliminate.

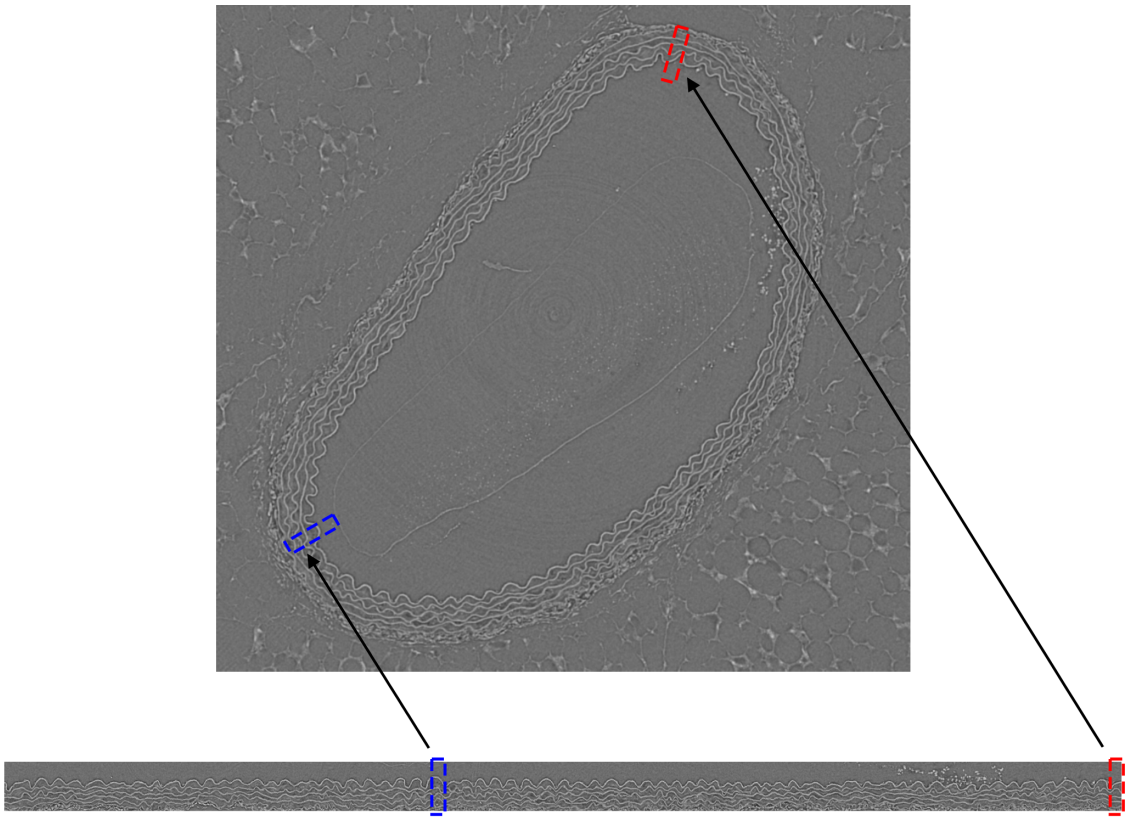


Figure 3.18: A Gaussian-smoothed grayscale image in stack 6 and the corresponding unfolded arterial wall.

Thus, each stack has its own combination of the above operations. In the lumen elimination process, the bifurcations can induce more than one connected object that can be recognized as a lumen in the image unfolding process. In this case, we used the continuity between the frames to eliminate the falsely induced objects.

3.6 Conclusion

In this chapter, we described the 3D lumen extraction process. The idea is based on the observation that in our images, the inner side of the arterial walls is relatively clean and its structure is less complex compared to the outer side that contains the *tunica adventitia* and is in contact with perivascular tissues. This process is composed of two steps: lumen cleaning and lumen extraction. The lumen cleaning process was performed to eliminate impurities inside the lumen to avoid misleading the extraction of the lumen. In both two steps, the vertical continuity of an artery was taken advantage of and spheres were used to implement the methods. With the extracted lumen, we got a general shape of this structure surrounded by the elastic

lamellae and we were capable of calculating the normal vectors starting from the contour to the outside of the lumen, which enables us to unfold the arterial wall. This operation converts the spatial relationship between different elastic lamellae layers as well as that between the *tunica media* and the *tunica adventitia* from "inside-outside" to "top-bottom". This positional information enables us to perform lamella-level segmentation processes in the next Chapter.

Chapter 4

Elastic lamellae segmentation

The aim of our work is to analyze the features of elastic lamellae of arterial walls in different samples. In the previous chapter, we extracted the shape of the lumen and unfolded the images with the help of this shape which described generally the inner contour of an arterial wall. In this chapter, we will perform segmentation processes based on these unfolded images and restore them to their original shapes for further analysis. The segmentation consists of two parts: extraction of the first elastic lamellae and classification of the *tunica media* (where lie elastic lamellae) and the *tunica adventitia*.

4.1 Related works

We used two main approaches in the work of this chapter: region growing for the extraction of the first elastic lamella and texture classification by deep learning for finding the *media-adventitia* border.

4.1.1 Region growing

Region growing is a segmentation method that extracts a region from seed points to the spread area with respect to the pixels connectivity and criteria measuring pixels similarities such as intensity similarity. In the work of Adams and Bischof [97], the seeded region growing (SRG) was proposed. This method uses the similarity between a set A_i including involved pixels after m steps and a pixel x adjacent to A_i, which is defined as:

$$\delta(x) = |g(x) - \underset{y \in A_{i(x)}}{mean}[g(y)]| \tag{4.1}$$

where $g(x)$ is the gray value of x, $i(x) \in \{1, 2, \ldots, n\}$ is the index such that the neighborhood of x and $A_{i(x)}$ has a nonempty intersection. At step $m+1$, $z \in T$ is taken if:

$$\delta(z) = \underset{x \in T}{min}\{\delta(x)\} \tag{4.2}$$

where T is the set of unallocated pixels that are neighbors of any A_i. Pixels are added in an order according to their score δ until no pixel is left. They are then labeled with a sequentially sorted list (SSL). Mehnert and Jackway [98] improved SRG by fixing the order dependencies from the inherent aspect related to Equation 4.2 and from the implementation aspect related to the SSL structure. Chang and Li [99] proposed an adaptive region growing method with an adaptive homogeneity test ($\mathcal{ADAP}$) on region R_i and R_j that is defined as:

$$\mathcal{ADAP}(R_i \cup R_j) = \begin{cases} true & if\ l_{j1} < \overline{X} < l_{j2}\ and\ l_{i1} < \overline{Y} < l_{i2} \\ false & otherwise \end{cases} \tag{4.3}$$

where $\overline{X}$ and $\overline{Y}$ are feature means of R_i and R_j that are composed of primitive regions $\{X_1, ..., X_n\}$ and $\{Y_1, ..., Y_m\}$ respectively; (l_{i1}, l_{i1}) is the adaptive range that satisfies:

$$Pr(l_{i1} < X_j < l_{l2}) = \lambda\ and\ Pr(l_{i2} \leq X_j) = Pr(X_j \leq l_{i1}) = \frac{1-\lambda}{2} \tag{4.4}$$

where λ is a user-defined parameter between 0 and 1. Hojjatoleslami and Kittler [100] proposed a region growing method for finding the boundary of a ROI defined as a grey-level blob. In this method, a region grows by each time joining the pixel with the highest gray level among the neighbors. Two discontinuity measures are used to control this process: average contrast and peripheral contrast. These two measures are calculated based on the concepts of current boundary (CB), which is the set of neighboring pixels of the current region, and internal boundary (IB), which is the set of connected outermost pixels of the current region. The average contrast boundary (ACB) is the segmentation result based on the maximum of the first measure which is the difference between the average gray value of the pixels in the current region and that of the pixels on CB. The peripheral contrast boundary (PCB) based on the second measure is found on the last local maximum of the difference between the average gray value of IB and CB before finding the ACB. Revol and Jourlin [101] proposed a method using spatial neighborhood and histogram neighborhood information with morphological operations. Merged points can be revised and the

resulting regions are not necessarily connected. A dilation is first applied at each step; the homogeneity of the dilated region is examined by comparing the standard deviation of its histogram and a threshold. If the dilated region is homogeneous, then it replaces the previous region. Otherwise, the dilated region is modified according to the range of the histogram of the initial region. In this case, the homogeneity of the modified dilated region is checked again. It will be enlarged or reduced by widening or narrowing the range of its histogram. This method can be applied to 2D and 3D images.

3D segmentation is often required for medical images. Revol-Muller *et al.* [102] proposed an automated 3D region growing algorithm that converts this problem to maximum function detection. In this process, a sequence of segmented regions is found with the homogeneity threshold σ ranging from σ_0 to σ_m, the growing process stops when an optimal threshold σ_{opt} is found by an evaluation function $f_{ev}(\sigma)$ which can be based on a boundary or region criterion. Singh *et al.* [103] proposed a 3D adaptive region growing method with gain calculation for volumetric CT data of airway tree. For voxels in the N_{26} neighborhood of a seed point, those having intensities lower than a prefixed threshold are included, those having higher intensities will be calculated with an entropy-based gain, and voxels that give gains greater than a threshold will be included. Yau *et al.* [104] proposed a method to segment the inferior alveolar nerve canal from CT images. This method applies an adaptive region growing on cross-sectional images. The seed point of the current image can be generated according to the result of the previous image, and the growing criterion is whether the pixel intensity is in the range bounded by a lower and an upper threshold calculated from pixel intensities in a ROI centered on the seed point. Martínez-Mera *et al.* [105] applied region growing method to segment a part of the aorta by using 17-pixel neighborhoods as is shown in Figure 4.7. This is for restricting the growing process in one direction in order to avoid segmentation of the ascending aorta and consequently the heart, which is not the goal of this segmentation step.

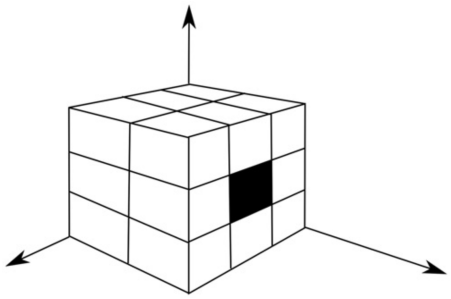


Figure 4.1: The 17 neighbors for volumetric region growing in the work of Martínez-Mera *et al.* [105].

4.1.2 Texture classification

The *tunica media* and the *tunica adventitia* in the arterial wall are adjacent to each other. Their intensities and pixel densities are similar. The main difference between them is the texture, which is the most important feature that we use when distinguishing them by eyes. Image texture can be defined as a function of the spatial variation in pixel intensities [106]. There exist various texture classification methods. A texture can be expressed by a descriptor which is a vector containing information of a piece of the image. Gray-level co-occurrence matrix (GLCM) [107] describes an image in a statistical way. Given an angle θ and a distance D, a GLCM can be generated and it shows the spatial information between pairs of pixels by placing the number of the pixel pairs with gray levels i, j and with an angle θ as well as a distance d between them on the position (i, j) of the matrix. Texture features can be extracted from a GLCM from different aspects, for example, contrast and entropy:

$$Contrast(\theta, d) = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} (i-j) M(i, j, \theta, d) \tag{4.5}$$

$$Entropy(\theta, d) = \sum_{i=0}^{N_g-1} \sum_{j=0}^{N_g-1} M(i, j, \theta, d) \log M(i, j, \theta, d) \tag{4.6}$$

where $M(i, j, \theta, d)$ is the value in the i-th row and the j-th column in a GLCM defined with θ and d, while N_g is the number of distinct gray levels in the quantized image. Local binary patterns (LBP) [108] [60] are another frequently used texture descriptor [109]. LBP operator turns thresholded neighboring pixels around a center pixel to a decimal number, as is shown in Figure 4.2. Pixels around the center are binarized and are calculated to decimal numbers according to their positions. There

are 2^8 combinations that can generate a histogram with 256 bins in this case of 8 neighborhoods. Figure 4.3 illustrates several LBPs with circular neighborhoods [110], where P and R control the quantification of the angular space and the space resolution respectively. A Gabor filter bank is generated from a Gabor function with a set of parameters which is defined as [111]:

$$G(x, y, k_x, k_y) = \exp(-\frac{(x - x_0)^2 + (y - y_0)^2}{2\sigma^2}) \exp(j(k_x x + k_y y)) \tag{4.7}$$

where x, y are spatial coordinates, k_x, k_y are frequency coordinates, x_0 and y_0 are spatial localizations of the Gaussian function that envelopes the shape, thereby modulating the sinusoidal pattern. The spatial frequency and the orientation that can distinguish different filters are given by:

$$\omega = 2\pi\sqrt{k_x^2 + k_y^2} \quad and \quad \theta = arctan\frac{k_x}{k_y} \tag{4.8}$$

Equation 4.7 shows that the Gabor function gives a complex number; the real and imaginary parts can be used to generate filters independently. There exist other types of texture extraction and classification methods. As classified in the work of Armi and Fekri-Ershad [112], GLCM and LBP are statistical methods; Gabor filter bank is a transform-based method; there are also structural methods and model-based methods. Scale invariant feature transform (SIFT) [113] [114] is a structural method that generates feature descriptors with the help of Difference of Gaussian (DoG) in four steps: scale-space extrema detection, key point localization, orientation assignment and generating key point descriptor. With various scales taken into account and an orientation assignment, the descriptors are scale and rotational invariant. Model-based methods, such as autoregressive, are used for texture modeling [112].

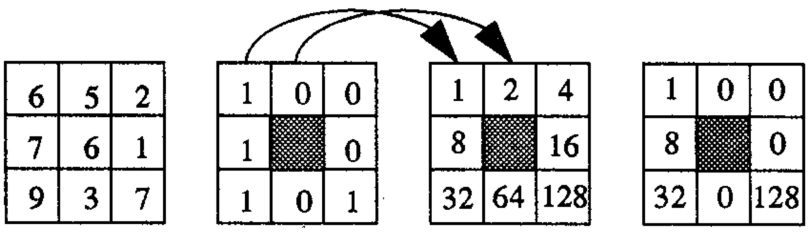


Figure 4.2: A LBP operator [115]. Neighbor pixels are thresholded, then numbers assigned to each position are summed up.

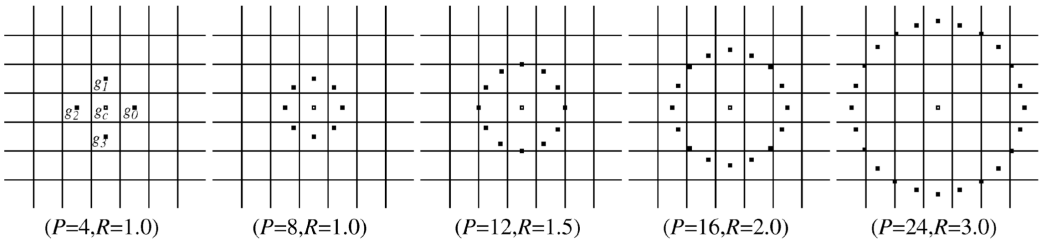


Figure 4.3: LBPs with circular neighborhoods [115].

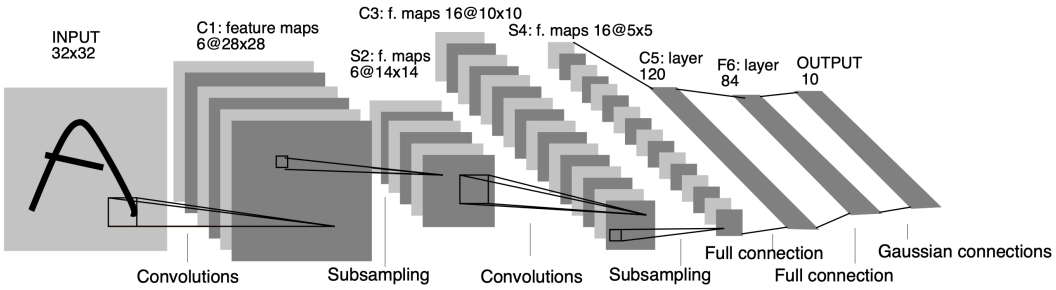


Figure 4.4: LeNet-5 network [116].

Deep learning approaches are a powerful tool for feature extraction and pattern recognition. They have enabled huge performance enhancement in biomedical applications [117]. Convolutional neural network (CNN) is widely used in deep learning tasks. CNNs do not need to extract image features manually, which is different from traditional methods [118]. With multiple layers containing a number of filters that generate various feature maps, this structure is able to extract low-level features in the layers that the input passes earlier and high-level features in the last layers [119]. Figure 4.4 shows the structure of LeNet-5 network [116], which is composed of three types of layers: convolutional layers, pooling layers, and fully-connected layers. In convolutional layers, different filter banks are used to generate different feature maps. Units in the current layer are connected to patches of the previous layer. The features extracted from the filter bank pass through a non-linear activation function such as sigmoid and rectified linear unit (ReLU) [120] [121]. The subsampling process in Figure 4.4 corresponds to pooling layers, which semantically merge similar features into one to gain the shift-invariance [121] [122]. In the fully-connected layer, each unit is connected to all the units in the previous layer. A feature map convolved by a filter bank in the convolutional layers can be expressed as [122]:

$$z^l_{i,j,k} = {\mathbf{w}^l_k}^T \mathbf{x}^l_{i,j} + b^l_k \tag{4.9}$$

$$a^l_{i,j,k} = a(z^l_{i,j,k}) \tag{4.10}$$

where (i, j) is the location of the element in the feature map, l is the index of the layer, k is the index of the feature map, $\mathbf{w}$ and b are the weights for performing the convolution and a bias. $z^l_{i,j,k}$ is the value at location (i, j) after the convolution with the kernel of the k-th channel using value of the l-th layer $\mathbf{x}^l_{i,j}$ as the input. $a^l_{i,j,k}$ is the result of applying an activation function $a(.)$. A pooling layer reduces the dimension of a feature map by a method such as calculating the mean value or selecting the maximal value in each patch while shifting by more than one length unit in a direction. A CNN is trained to fit the desired input-output relation by minimizing a loss function. This function can be written as [122]:

$$\mathcal{L} = \frac{1}{N} \sum_{n=1}^{N} l(\theta; y^{(n)}, o^{(n)}) \tag{4.11}$$

where o is the output of the last layer after being processed by an operation such as softmax, y is the label corresponding to the output, n is the number of the input and the output data, θ is the set of the parameters. Parameters are adjusted during the training process performed by a backpropagation where gradient vectors are computed and the weights are adjusted in the opposite directions of the gradient vectors [121].

According to the work of Cavalin and Oliveira [109], texture classification approaches with CNN can be categorized into two types: those processing a whole image at once by end-to-end learning and the extensions of patch-based methods that combine the outputs extracted from patches of a whole image and perform classification. One of the first types of methods is texture convolutional neural network (T-CNN) [123], which is based on AlexNet [124] with one of the convolutional layers connected to the first fully-connected layer by an energy layer. Figure 4.5 shows the structure of T-CNN-2; the second convolutional layer of the CNN is followed by an energy layer pooling the whole feature map and outputting a value for each feature map. This structure aims to preserve less part of the analysis of the global shape and extract only the information that is of more importance for texture analysis.

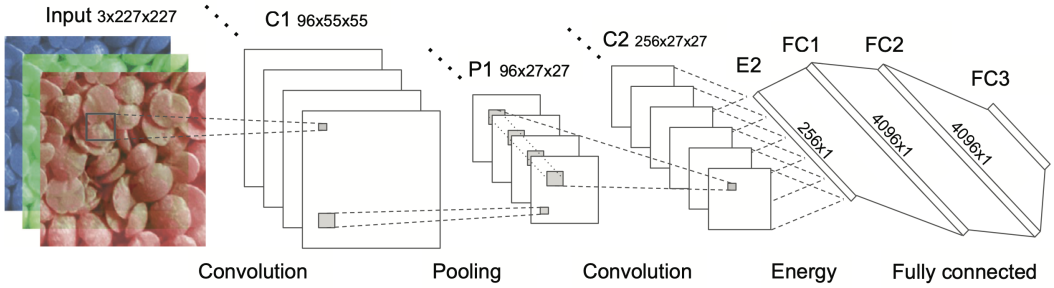


Figure 4.5: Architecture of T-CNN-2 with two convolutional layers [123].

4.1.3 Contrastive learning and Siamese neural network

Deep learning approaches are capable of extracting features from a given dataset by training a neural network with a number of parameters. However, in the lack of a dataset with the ground truth of classifications corresponding to texture images, we cannot apply these approaches by training a network that directly outputs the texture type. A supervised model is not suitable for our needs. Self-supervised learning makes it possible to train a model without manually labeling the training data. Under this category, contrastive learning intends to compare the input samples by classifying inputs into positive (similar) pairs, and negative (dissimilar) pairs. With defined similarity distributions, contrastive learning aims to map close representations from similar inputs and map more distant ones from dissimilar inputs [125].

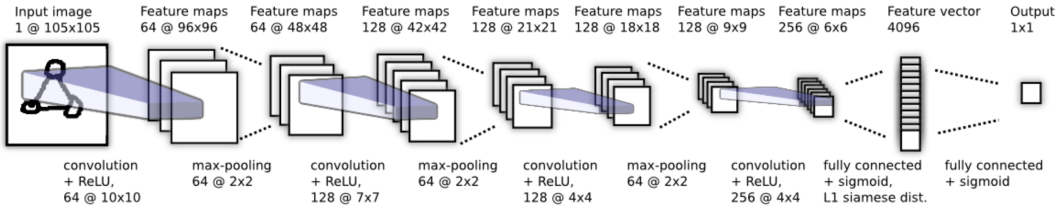


Figure 4.6: CNN architecture proposed by Koch *et al.* [17] for one-shot learning.

Siamese neural network was first proposed by Bromley *et al.* [16] for signature verification. In this work, two inputs enter two identical sub-networks separately and output two feature vectors between which the cosine of the angle is considered the distance value. A typical application of Siamese network is one-shot learning [17], which is for tasks where little supervised information is available and only one example can be observed in each class. Figure 4.6 illustrates the architecture of the network. A pair of inputs enter this network, and the distance $\mathbf{p}$ between the

outputs of the fully-connected layer $\mathbf{h}_{1,L-1}^{(j)}$ and $\mathbf{h}_{2,L-1}^{(j)}$ is measured by:

$$\mathbf{p} = \sigma\left(\sum_j \alpha_j |\mathbf{h}_{1,L-1}^{(j)} - \mathbf{h}_{2,L-1}^{(j)}|\right) \tag{4.12}$$

where σ is the sigmoid function and α_j are parameters to learn during the training process that weight the importance of the $L1$ component-wise distance. Omniglot dataset [126] that contains alphabet images is used with a data augmentation done by affine distortions and a regularized cross-entropy is used as the loss function:

$$\mathcal{L}(x_1^{(i)}, x_2^{(i)}) = \mathbf{y}(x_1^{(i)}, x_2^{(i)}) \log \mathbf{p}(x_1^{(i)}, x_2^{(i)}) + (1 - \mathbf{y}(x_1^{(i)}, x_2^{(i)})) \log(1 - \mathbf{p}(x_1^{(i)}, x_2^{(i)})) + \boldsymbol{\lambda}^T |\mathbf{w}|^2 \tag{4.13}$$

where i is the index of the minibatch, $\mathbf{y}(x_1^{(i)}, x_2^{(i)}) = 1$ if x_1 and x_2 are from the same class, otherwise $\mathbf{y}(x_1^{(i)}, x_2^{(i)}) = 0$, $\boldsymbol{\lambda}$ gathers the L_2 regularization weights, $\mathbf{w}$ contains the weights of the network. Melekhov *et al.* [127] used a Siamese network to learn an image-level feature to perform image matching. The network structure is based on HybridCNN [128] which is used for object and scene classification. The loss function is a margin-based contrastive loss proposed by Hadsell *et al.* [129]:

$$\mathcal{L} = \frac{1}{2} l D^2 + \frac{1}{2}(l-1)(\max(0, m - D))^2 \tag{4.14}$$

where l is the binary label indicating whether the inputs are a positive pair or a negative pair, D is the Euclidean distance between feature vectors generated by the two inputs, m is a parameter that is set to twice the average Euclidean distance between features of training pairs before learning. Yamada *et al.* [130] combined the Siamese network with hierarchical region merging to perform texture segmentation. As is shown in Figure 4.7, the network is used to generate feature vectors and these vectors are grouped with the region adjacency graph (RAG) used in the region merging algorithm proposed by Kurita [131]. DeTone *et al.* [132] proposed a self-supervised framework that can be used for image registration containing a detection of feature points and a generation of feature descriptors. Figure 4.8 illustrates the network architecture and its training process. Two branches share a VGG-style [133] encoder with different decoders for interest point detection and feature description, respectively. Two losses are combined to train two branches simultaneously. The

loss l_d for the descriptor part is:

$$l_d(d, d^{'}; s) = \lambda_d \cdot s \cdot \max(0, m_p - d^T d^{'}) + (1 - s) \cdot \max(0, d^T d^{'} - m_n) \quad (4.15)$$

where d and $d^{'}$ are descriptors generated by two patches cut from the same image but before and after a homographic transformation. The indicator s reflects their similarity. The value of s is decided by whether the position of the first patch after this transformation is close enough to that of the second patch on the transformed image. The value λ is used for balancing the amount of negative and positive pairs; m_p and m_n are margins to control the value of the descriptor. This method is capable of training a Siamese network without pre-labeled image pairs by generating similar and dissimilar patches from the same image and labeling them according to their spatial relationship.

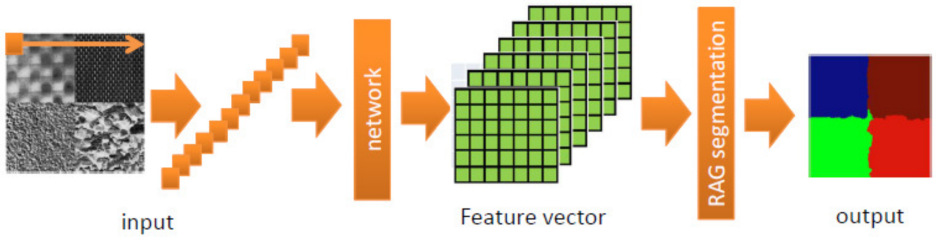


Figure 4.7: Texture segmentation method proposed by Yamada *et al.* [130]. Vectors generated from the neural network are merged to achieve segmentation.

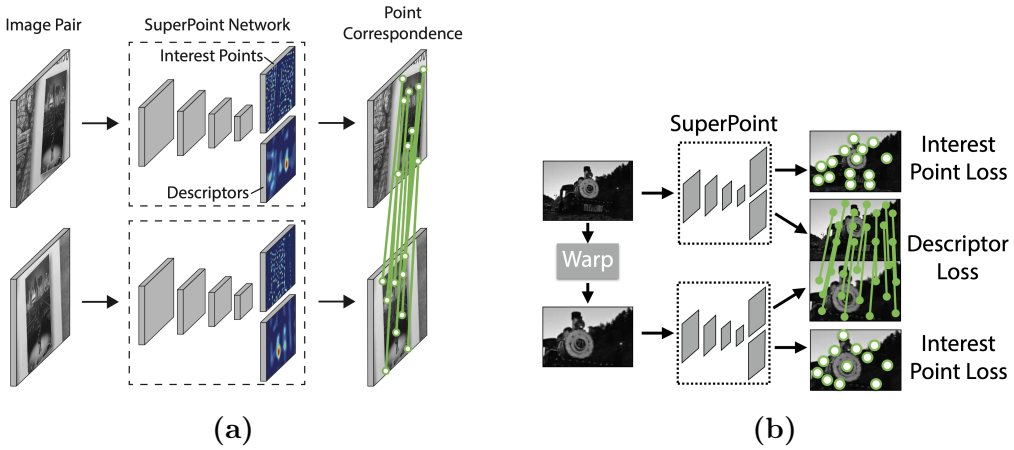


Figure 4.8: Architecture of SuperPoint network (a) and its training process (b) [132].

4.2 Segmentation of the first elastic lamella

In this section, we present the segmentation process of the first elastic lamella. This lamella is the innermost layer of the *tunica media* in the original image and the uppermost layer in the unfolded image. In this process, an unfolded image is first pre-processed to a binary image. Then the first lamella is extracted by a region growing-based method. Finally, a shape restoration operation is applied to fold the layer to its original shape.

4.2.1 Pre-processing

The unfolded images were obtained from the 3D Gaussian-smoothed grayscale images. To perform the following segmentation processes, we need to perform a second denoising process to get grayscale images with less noise and a binarization process that focuses on the extracted arterial region.

Denoising

The straightened images are in gray scales. Before being straightened, they were processed by simple denoising of 3D Gaussian smoothing as described in Chapter 2. However, this first denoising process in the pre-processing was performed to facilitate the binarization to generate images for a lumen extraction. For finer segmentation in elastic lamellae scale, a non-local means (NL-means) denoising [14] [15] was applied. This method filters an image by searching neighborhoods of each pixel and averaging them according to the similarity measured by the Euclidean distances between patches. The denoised value of pixel p with value $u(p)$ is:

$$\hat{u}(p) = \frac{1}{C_{nlm}(p)} \sum_{q \in B(p,r)} w(p,q)u(q) \tag{4.16}$$

with

$$C_{nlm}(p) = \sum_{q \in B(p,r)} w(p,q) \tag{4.17}$$

where $B(p,r)$ is the patch centered at p and of size $(2r+1)(2r+1)$ (searching zone), $w(p,q)$ is the weight added to q that measures the patch similarity. This weight is

calculated by:

$$w(p,q) = \exp\left(-\frac{\max(d^2 - \sigma^2, 0)}{h^2}\right) \tag{4.18}$$

where d is the Euclidean distance between patches:

$$d^2(B(p,f), B(q,f)) = \frac{1}{(2f+1)^2} \sum_{j \in B(0,f)} (u(p+j) - u(q+j))^2 \tag{4.19}$$

and $B(0,f)$ is the neighborhood of size $(2f+1)(2f+1)$ centered at p and q. The parameters to set are h (filtering strength), $(2f+1)$ (patch size) and $(2r+1)$ (searching window size), h and σ are related. We set $(h, 2f+1, 2r+1)$ to (9, 13, 7) and applied this method to unfolded grayscale images. A result is shown in Figure 4.9.

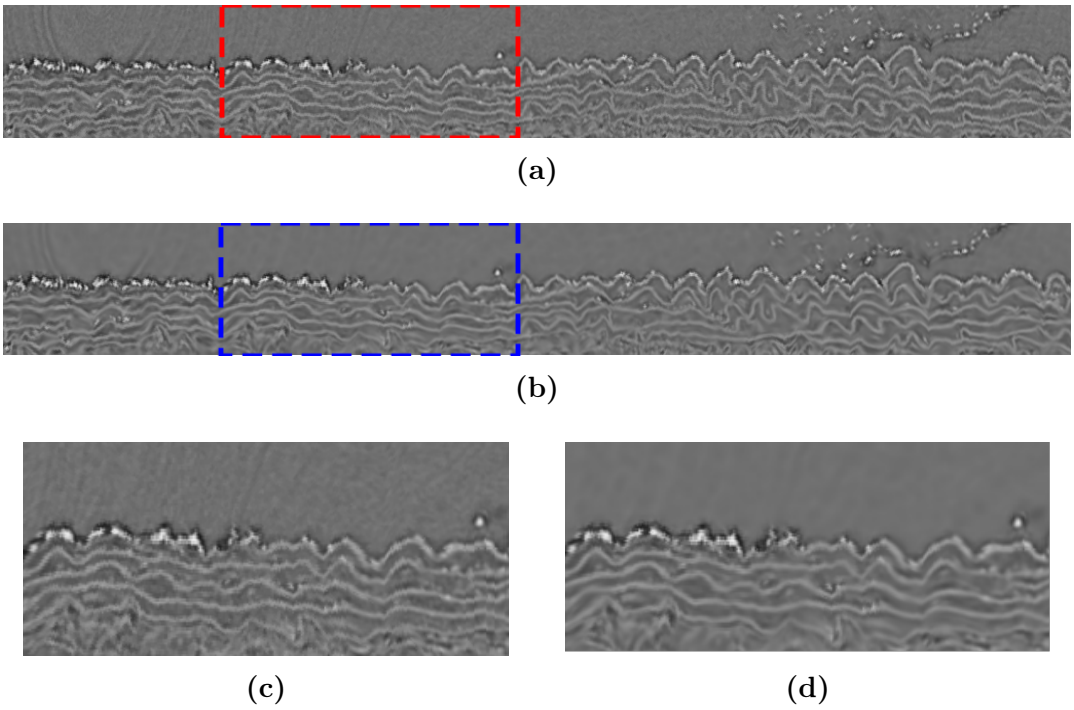


Figure 4.9: Unfolded image before (a) and after (b) NL-means denoising (cropped segment) as well as two magnified segments (c) and (d) corresponding to the marked regions in (a) and (b).

Binarization

The segmentation of elastic lamellae is performed on binary images. Thus the aim is to obtain images where pixels on the elastic lamellae are binarized to 255

(white) and the spaces between them are binarized to 0 (black).

We first performed a contrast enhancement to the NL-means-filtered images by fixing a standard deviation (results denoted as $I_{adjusted_{unfold}}$). Then a binarization with the consideration of local contrast was applied. In the pre-processing steps, we obtained images with a lumen-cleaning process. We unfolded these binary images (denoted as $I_{bin_{unfold}}$) as well and used them to calculate the threshold for this binarization. We calculated the mean value $m_{adjusted}$ of the pixels in $I_{adjusted_{unfold}}$ that are white in $I_{bin_{unfold}}$, and the minimum $min_{adjusted}$ among this values. A basic threshold $Th^0_{unfolded}$ was then calculated by:

$$Th^0_{unfolded} = \frac{(m_{adjusted} + min_{adjusted})}{2} \tag{4.20}$$

We also calculated the local contrast $C(p)$ at each pixel p by the method in Equation 2.28. The threshold for a pixel in $I_{adjusted_{unfold}}$ was then calculated by:

$$Th_{unfolded}(p) = Th^0_{unfolded} - \frac{C(p)}{6} \tag{4.21}$$

This is because sometimes there exist high-contrast particles on the first lamella and in these regions, the intensities of the pixels that should be continuous are relatively low. The unfolded images were binarized from the NL-means-filtered ones followed by an adjustment. These images contain impurities on the inner side of the first lamellae. In the pre-processing steps, we obtained images with a lumen-cleaning process. We unfolded these binary images as well and used them as a constraint of the binarized unfolded images. The binary image after a lumen cleaning, denoted $I_{bin_{init}}$, was unfolded to $I_{bin_{unfold}}$. We first applied a dilation to this image and got $I_{bin^{dil}_{unfold}}$. The $I_{unfold_{bin}}$ was then processed by removing pixels in each column that is higher than the highest one in $I_{bin_{unfold}}$. The largest connected component in every sub stack was then selected. For sub stacks where there are two or more connected components, we selected the two (or more) largest components after verifying the results of selecting the largest component. An example is shown in Figure 4.10.

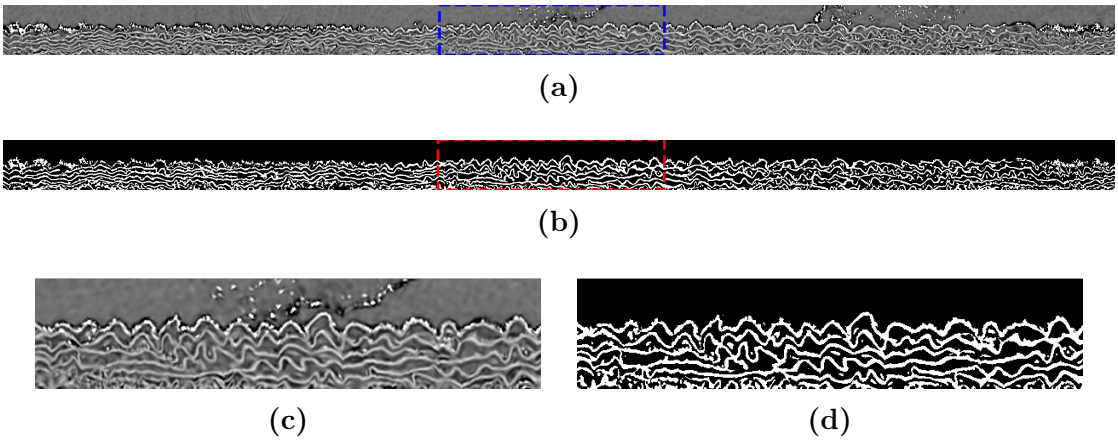


Figure 4.10: (a) An unfolded denoised image after a contrast enhancement. (b) Binarized image of (a). The impurities above the first elastic lamella in (a) do not appear in the binarized result (b) because a constraint of the unfolded binary image after the lumen-cleaning process is added. (c) and (d) are magnified segments corresponding to the marked regions in (a) and (b).

4.2.2 Extraction of the first elastic lamella

In all of the elastic lamellae layers, the first layer is relatively much more complete. We start our segmentation from the extraction of this first elastic lamella.

In a straightened image, the elastic lamella of layer l is above layer $l+1$ globally. However, locally, two pixels $p(x_1, y_1)$ and $p(x_2, y_2)$ with $x_1 = x_2$ and $y_1 < y_2$ (y_1 is above y_2) do not necessarily meet the condition of $L(p(x_1, y_1)) < L(p(x_2, y_2))$ ($L(p)$ is the number of layer of the pixel p). This is because the elastic lamellae can be very curvy. As is shown in Figure 4.11, the same elastic lamella can overlap in the vertical direction.

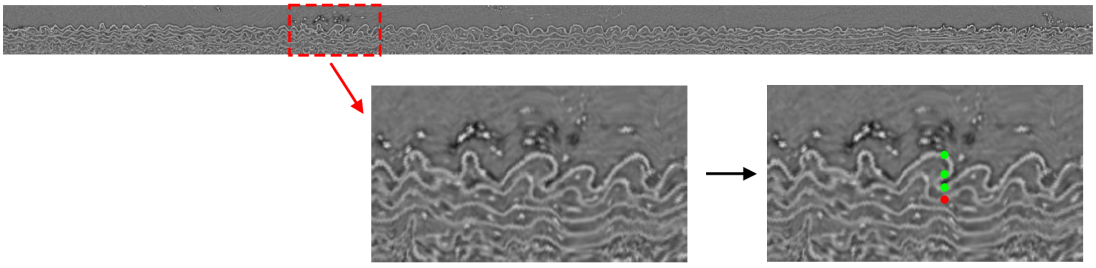


Figure 4.11: An unfolded arterial wall image. In the magnified part, three green points of the first elastic lamella are aligned on the same vertical line. The red point lying on the second elastic lamella is the 4th point on this line.

Since the elastic lamellae are not strictly sequentially placed on the same vertical line, a relatively global method would be more accurate than locally counting the

order of lamellae from top to bottom. As was discussed above, with $x_1 = x_2$ and $y_1 < y_2$, we cannot conclude that $L(p(x_1, y_1)) < L(p(x_2, y_2))$. However, according to the nature of the arrangement of these layers, with $L(p(x_1, y_1)) < L(p(x_2, y_2))$, we can infer that $p(x_1, y_1)$ is on the inner side of $p(x_2, y_2)$. This relationship cannot be expressed by the coordinates (x_1, y_1) and (x_2, y_2), but can be described as "if we start from the top of the unfolded image, we cannot reach layer $L(p(x_2, y_2))$ without passing any point of layer $L(p(x_2, y_2))$". Layer l can be considered as a wall blocking the upper space from layer $l + 1$, which is what "the elastic lamella of layer l is globally above layer $l + 1$" means in the last paragraph. This feature motivated us to use the method of region growing, in which the segmentation starts from seed points and pixels spread to their neighborhoods progressively. The goal is to fill the space enclosed by boundaries.

3D region growing by segments

One of the problems of the region growing method is the leakage of pixels/voxels, which means the region keeps growing as long as there is "access", even if this is caused by the weak intensities of pixels that should stop the growing process. In our images, it is almost impossible to find a first elastic lamella that is completely "closed" of which the binarized image gives no access from the space above it to the space under it. Consequently, if we apply the region growing method on an entire substack of unfolded images (32 frames) by placing a seed point in the upper space of the first lamella, the space under the first lamella will be filled because of a "hole", and this could spread to a huge range of the image. However, the leaking points do not exist all along the lamella. If we apply the region growing process on a short segment of the 3D data, some of the results are without leakage. Each substack of unfolded images is of a different length. The size of each substack is (L_z, L_y, L_x), where $L_z = 32$ is the number of frames, $L_y = 128$ is the height of each unfolded image and L_x varies in each stack, which is between 2000 and 4000 pixels in most of the cases. A voxel v in a substack can be localized by (z_v, y_v, x_v). We applied the region growing process to the segments from leftmost to rightmost of the substack, the size of each segment is $(32, 128, 100)$, and the sliding step is 25, which means each voxel is involved in 4 operations. The left part and the right part of a substack are concatenated to make sure the voxels in these parts are operated the same way. Figure 4.12 gives a 2D illustration of this idea. Region growing is applied in the gray and the blue windows (dashed boxes), the gap of the curve falls in the blue window, thus resulting in leakage during the growing process. Pixels in region 1 are

involved in both of the growing processes while region 2 is only involved in the right one. For a cropped segment of size $(32, 128, 100)$, a seed is searched from the top among the voxels with $y_v = 0$. Once a black voxel is found, it is used as the seed for a 3D region growing in this segment. With the segment window sliding, a voxel can be contained in the grown region for several times. We preserved the voxels that were contained in the segmented region at least 3 times. Figure 4.13 illustrates the result of region growing by segments on a substack of 32 frames. A selection of the largest connected region was then applied after this process. Figure 4.14 is a 3D illustration of the selected regions corresponding to Figure 4.13b.

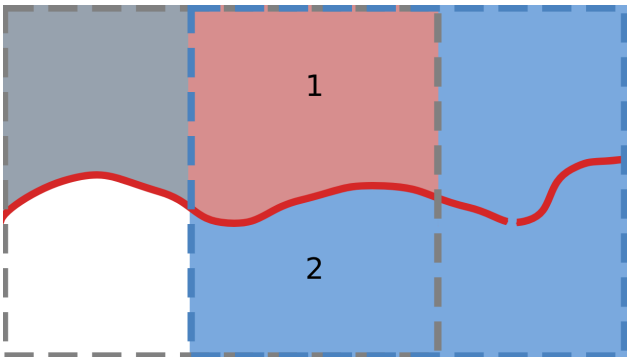


Figure 4.12: Two region growing processes in the sliding window with seeds on the upper side. Gray pixels and blue pixels are those involved in the left and the right growing processes, respectively; pink pixels are involved twice.

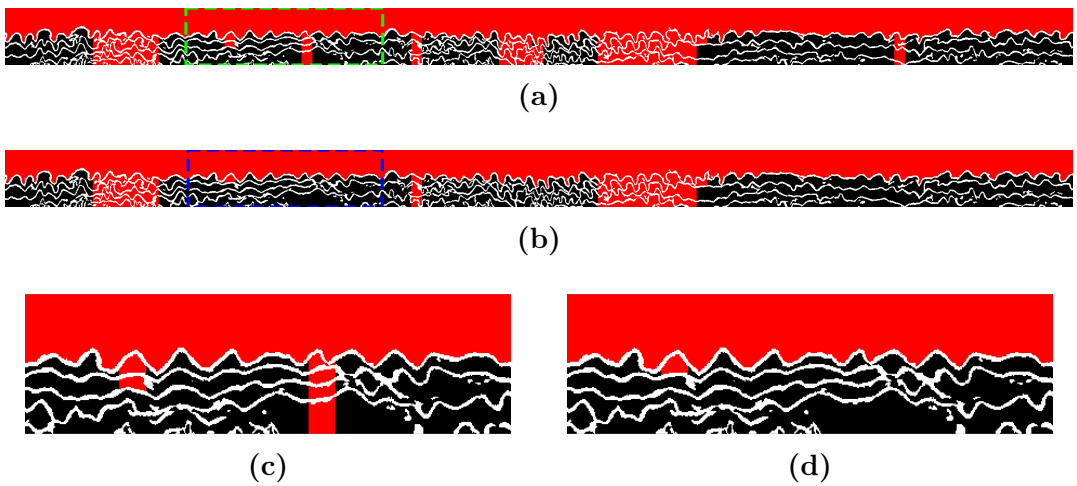


Figure 4.13: 2D projection on one frame of the region growing by segment in 3D (a) and the results after a selection of the largest 3D connected component (b). White pixels are from the initial image of elastic lamellae to which the operation is applied. Red pixels are from the grown regions. (c) and (d) are magnified segments corresponding to the marked regions in (a) and (b).

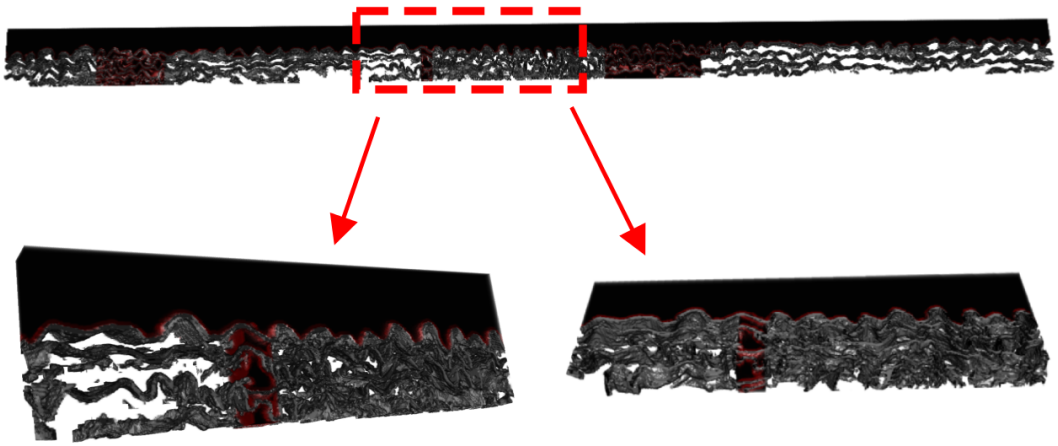


Figure 4.14: 3D illustration of a sub stack to which a region growing by segment followed by a component selection is applied. Gray voxels are from the initial image; dark red voxels are from the grown regions. The two images below are enlarged ones from the same segment and observed from two perspectives.

By region growing, we hope to obtain the upper space which is above and "blocked" by the first elastic lamella. The gaps on the lamella make this space spread to those under the lamella. As is shown in Figure 4.15, pixels involved by the previous operations are colored in red. The majority of them are pixels in the upper space that we need; several regions are spread by mistake. With Figure 4.11 we discussed the fact that the order of the lamellae and the order that a vertical line meets the lamella are not the same. The grown spaces that are met by the line cannot reflect their real order, either. We cropped and enlarged three segments of the space viewed in 2D in Figure 4.15. In the first two cropped images, wrongly spread regions are under the desired upper spaces due to the holes that exist in the 3D spaces nearby. If we start from the top of the patch and extend the straight lines downward, the white points will be met first. Then the line will cross the gap formed by the first elastic lamella and finally meet the upper points (green) of the unwanted region. However, in the third cropped image, the vertical lines meet white points and cross the gap. The first points (green) they meet after the gap are still parts of the upper space, which is due to the curvature of the lamellae. The 3D region growing is capable of filling the small corners folded by the lamellae. Meanwhile, even a small hole can lead to leakage and make the lower spaces filled too. Based on the above results, our next operations aim to eliminate the wrongly spread 3D regions.

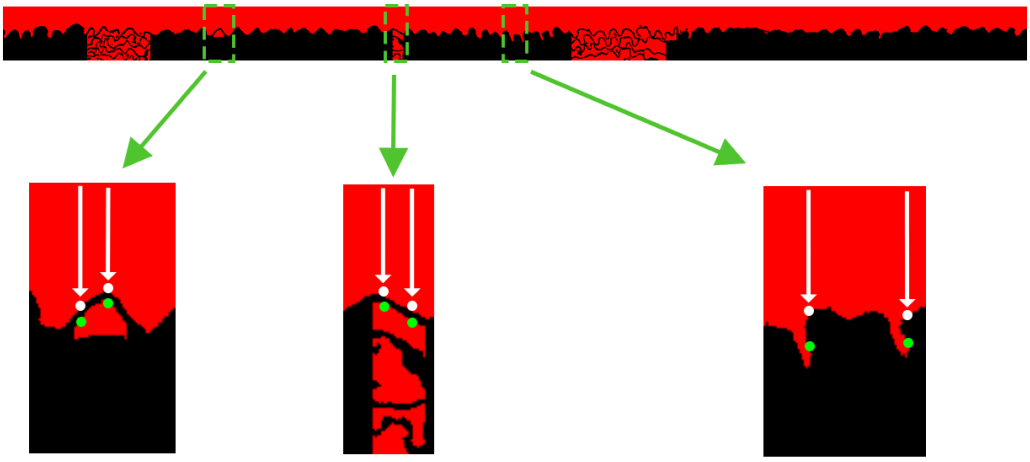


Figure 4.15: 2D illustration of the problem of selecting the correctly spread upper space above the first elastic lamella.

Grown space pruning

To remove unwanted voxels, we applied an operation on entire regions (connected objects) rather than on individual voxels since the latter cannot provide enough information about which type of space they belong to, as analyzed in the previous paragraph. By observing the regions by eyes, it is quite easy to distinguish most of the regions that should be under the first lamella from all the grown ones. One of their characteristics is that these regions present a "hanging" appearance compared to the upper spaces that are firmly attached to the top of the image. Starting from the holes on the lamella, leaked voxels spread further and grow until meeting other voxels under the first lamella that enclose the region or meeting the boundaries of the segment. Intuitively, the unwanted regions can be regarded as "heavy" ones that hang from the points that are not under the lamella. These points are the beginning of the leakages. The further a voxel is from the hanging node, the "heavier" it is. Figure 4.16 is a 2D illustration of the difference between the two types of regions. p_1 and p_2 are both grown pixels (yellow region), the pink and the green curves are the first and the second lamellae. A leaking point lies on the first lamella so that p_2 is also involved although it is under the first lamella. Spatially, p_2 is above p_1 ($y_{p_2} < y_{p_1}$) while p_2 is in the space "lower" than p_1. We draw two vertical lines (blue) down from the top without crossing any lamella, the solid parts together with the lamella enclose regions containing p_1 and p_2, denoted as R_{p_1} and R_{p_2}, respectively. The two regions are attached to the two solid lines (surfaces in 3D), on which the

pixels can be considered as "connecting points". p_1 is not attached directly to the top of the image but is not far from the connecting line while it takes a longer path to arrive at p_2 from its connecting line on the right. Also, the amount of pixels in R_2 is much more than R_1, which also makes R_2 a region weighing more.

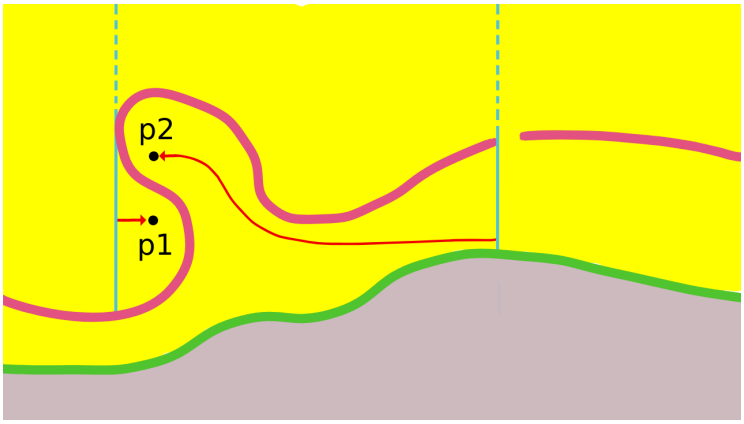


Figure 4.16: 2D illustration of points spread by region growing belonging to different types of spaces. p_1 is a correctly spread pixel, while p_2 is a wrongly spread one due to the leakage. Each of them is connected to the vertical line (in blue) on their side.

This leads us to classify grown spaces according to their weight compared to the size of nodes that connect them to the upper region (which is attached to the top). A measure $r_{w/c}$ that estimates the ratio between the weight of a region and the solidity of the connection is defined as:

$$r^i_{w/c} = \frac{w(R_i)}{c(R_i)} \tag{4.22}$$

where $w(R_i)$ is the weight of the region R_i and $c(R_i)$ is the corresponding connection size.

To get separated regions R_i, we search the connections by drawing vertical lines from the top of the 3D grown spaces of a 3D stack. Once a line touches white pixels of the initial binary image of elastic lamellae, it stops going down. All these lines form spaces directly attached to the top, and if we remove them from the grown regions, the rest of the image would be separated ones that are under the lamellae. We denote the grown region (after a selection of the largest connected component) R_{rg}, the space composed of voxels directly connected to the top R_{top}, the space occupied by elastic lamellae (and other tissues that are white in the binarized image) R_{lame}, and the left space (candidate regions) after removing R_{top} from R_{rg} is denoted $R_{rg_{can}} = \{v \mid v \in R_{rg}, v \notin R_{top}\}$ and $\bigcup_i R^i_{rg_{can}} = R_{rg_{can}}$, i is the index of

the candidate region. The connection of the R^i_{con} are defined as the points in R_{top} and adjacent to $R^i_{rg_{can}}$:

$$R^i_{con} = \{v \mid v \in R_{top},\ \exists v' \in \mathcal{N}(v)\ s.t.\ v' \in R^i_{rg_{can}}\} \tag{4.23}$$

where $\mathcal{N}(v)$ is the 26-neighborhoods of voxel v. For a candidate region $R^i_{rg_{can}}$, the connection size $c(R^i_{rg_{can}})$ is the number of its connecting voxels:

$$c(R^i_{rg_{can}}) = \left|\{v \mid v \in R^i_{con}\}\right| \tag{4.24}$$

The weights of the voxels in this region are calculated by propagation starting from its connections. For each $R^i_{rg_{can}}$, we calculated a 3D matrix $W^i_{rg_{can}}$ that contains the weight of each voxel in $R^i_{rg_{can}}$. The sum of the weights is $w(R^i_{rg_{can}})$. The calculating process is described in Algorithm 1. In each component, the border voxels that are adjacent to the connections are assigned value 1. Then the voxels are involved and labeled with the steps that increment. After all the voxels are assigned a value, the weight of the component is the sum of all these steps.

With $c(R^i_{rg_{can}})$ and $w(R^i_{rg_{can}})$, the ratio $r_{w/c}$ in Equation 4.22 can be calculated by:

$$r^i_{w/c} = \frac{w(R^i_{rg_{can}})}{c(R^i_{rg_{can}})} \tag{4.25}$$

We set a threshold $Th_{r^i_{w/c}}$ to reject undesired regions. For a voxel v in $R^i_{rg_{can}}$, if $r^i_{w/c} < Th_{r^i_{w/c}}$, then v is labeled "preserved", otherwise it is labeled "rejected".

In the implementation, we set the threshold $Th_{r^i_{w/c}}$ to 20. We labeled a component $R^i_{rg_{can}}$ as "rejected" once its weight $w(R^i_{rg_{can}})$ was incremented to a value greater than $20 \cdot Th_{r^i_{w/c}}$. In the binary image, the discrete voxels can generate small isolated components that are not candidate components caused by leakage. Also, the probability of a small $R^i_{rg_{can}}$ being in the lower space is small, thus to reduce the calculation amount, we examined only the components of size larger than 50 voxels, and those who are not larger than 50 were labeled "preserved". The spaces directly connected to the top together with those labeled as "preserved" compose the upper space that we wanted. Thus, we have three types of components: elastic lamellae (R_{lame}), space above the first lamella (R^1_{above}) and space below the first lamella (R^1_{below}).

Figure 4.17 illustrates the result of the classification of candidate components after the upper space pruning. The 2D image is a frame of a substack; white pixels

Algorithm 1: Generation of mask and contour for single elastic lamella.

Data: $R^i_{con}, R^i_{rg_{can}}$
Result: $w(R^i_{rg_{can}})$
$step \leftarrow 1$;
$w(R^i_{rg_{can}}) \leftarrow 0$;
$W^i_{rg_{can}}$: array;
w_{last}: integer;
$continue_{grow} = True$;
for $w \in W^i_{rg_{can}}$ **do**
  $w \leftarrow 0$;
end
for $v \in R^i_{rg_{can}}$ **do**
  if $\exists v^{'} \in \mathcal{N}(v)$ **then**
    $W^i_{rg_{can}}(v) \leftarrow step$; `/* Assigning weight 1 to voxels on the border */`
    $w(R^i_{rg_{can}}) \leftarrow w(R^i_{rg_{can}}) + step$;
  end
end
while $continue_{grow} = True$ **do**
  $step \leftarrow step + 1$;
  $w_{last} \leftarrow w(R^i_{rg_{can}})$;
  for $v \in R^i_{rg_{can}}$ **do**
    if $W^i_{rg_{can}}(v) = step - 1$ **then**
      for $v^{'} \in \mathcal{N}(v)$ ***and*** $v^{'} \in R^i_{rg_{can}}$ **do**
        if $W^i_{rg_{can}}(v^{'}) = 0$ **then**
          $W^i_{rg_{can}}(v^{'}) = step$;
          $w(R^i_{rg_{can}}) \leftarrow step$;
          $w(R^i_{rg_{can}}) \leftarrow w(R^i_{rg_{can}}) + step$;
        end
      end
    end
  end
  if $w_{last} = w(R^i_{rg_{can}})$ **then**
    $continue_{grow} = False$;
  end
end

are those directly connected to the top; red pixels are those classified as the space above the first lamella and green pixels are wrongly spread pixels to eliminate. The red pixels contain not only those hidden by the curvy lamella, but also those caused by the non-smooth surface of the binarized lamella or noise/impurities. The green part in the middle is involved in a bifurcation. Figure 4.18 shows 3D segments of the classified voxels corresponding to this substack. Figure 4.19 shows the result in a stack where the appearance of the first lamella is not as regular as that in Figure 4.17. In the region with high contrast particles, the binarization generated a number of holes that caused leakages (cropped segment on the left). With $Th_{r^i_{w/c}}$ set to 20, these voxels covered by particles were mostly classified as "the first lamella". This is why we constrained the white pixels in $I_{unfold_{bin}}$ with the images unfolded after lumen cleaning. The impurities above the first lamella show a similar appearance of the tore lamella (fused with particles) after binarization. If the amount of the impurities is too high, it is difficult to distinguish the space below them (to preserve) from those below the tore lamellae (to reject). The red parts in this image are mainly caused by impurities and the non-smoothness of the lamella surface.

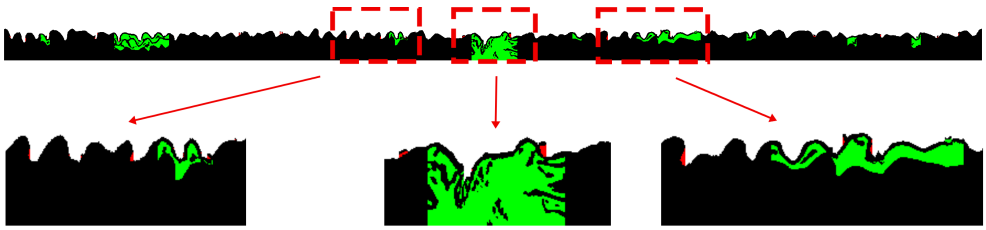


Figure 4.17: 2D illustration of classified candidate components. Upper space pixels under the curvy lamella are preserved as well as candidate pixels caused by impurities and non-smoothness of the lamella surface.

Extraction of elastic lamella

To extract the first lamella, we used a sphere of radius r_{lame} that slides in the 32-frame substack to find the border between R^1_{above} and R^1_{below}. We collected the centers of the spheres that contain voxels of the three spaces at the same time:

$$R^1_{lame_{seg}} = \{v \mid \exists (v_1, v_2, v_3) \in (R_{lame}, R^1_{below}, R^1_{above}) \ s.t. \ v_1, v_2, v_3 \in S(v, r_{lame})\} \tag{4.26}$$

$R^1_{lame_{seg}}$ is the 3D region segmented as the first elastic lamellae, $S(v, r_{lame})$ is the

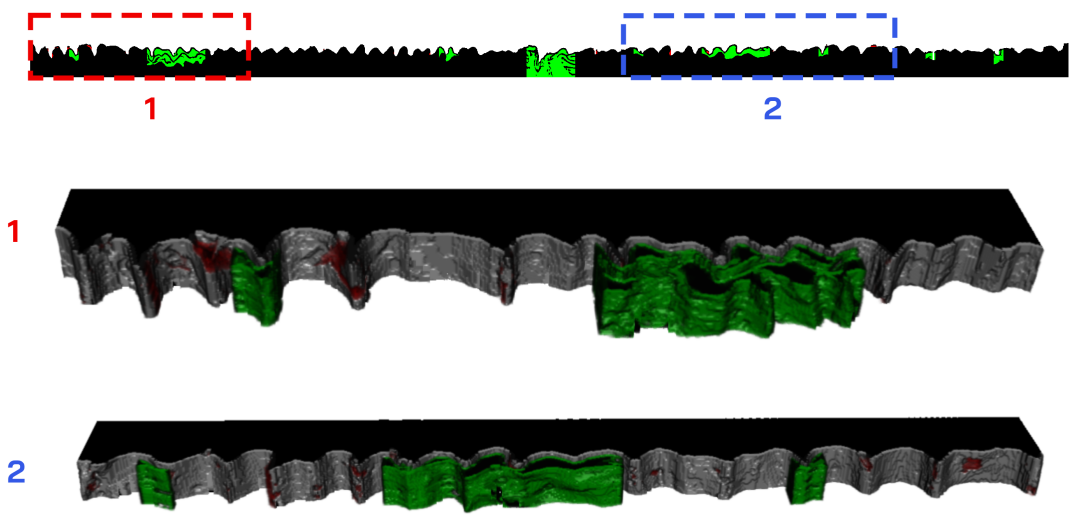


Figure 4.18: 3D segments cropped from the sub stack corresponding to Figure 4.17. Red and green voxels show the classification of candidate components.

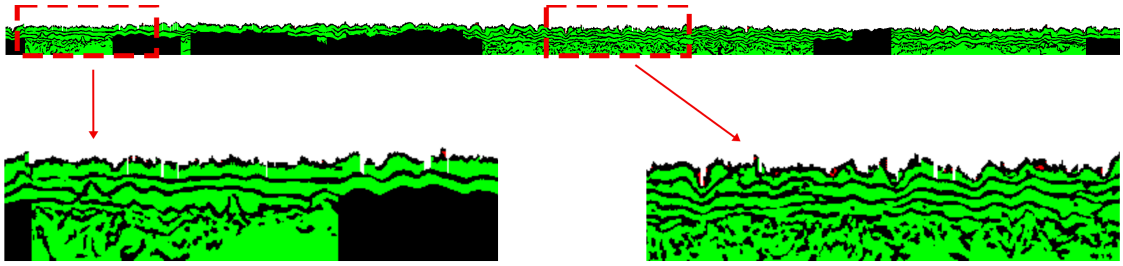


Figure 4.19: 2D illustration of classified candidate components. In this image, red pixels are mainly caused by impurities and non-smoothness of the lamella surface.

sphere centered in v of radius r_{lame}. An extracted region is shown in Figure 4.20, which is mainly composed of voxels of the first lamella, but also contains voxels from the second lamella. Thus, a post-processing of this segmented region is needed.

Figure 4.20: A frame of the first elastic lamella extracted by Equation 4.26.

The wrongly extracted pixels from the second lamella were generated for two main reasons: 1. the contact of different lamellae; 2. the gaps resulted from the holes on the first lamella that are filled with voxels in R^1_{above}. The second is the main cause of most of the errors. In this case, voxels on the second lamella that are in contact with R^1_{above} are considered as a part of the first lamella.

The two types of problems are illustrated by a 2D segment in Figure 4.21. The

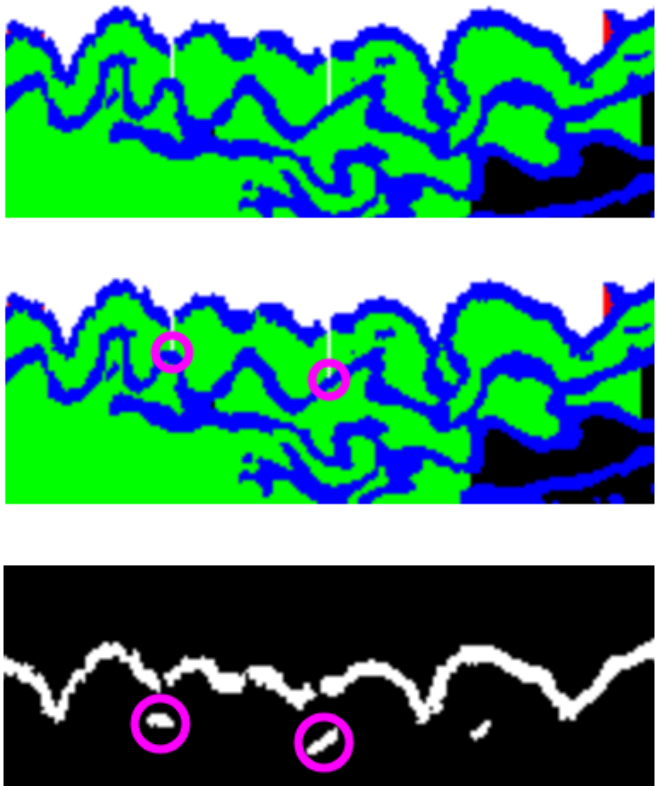


Figure 4.21: A 2D segment showing the main cause of wrongly extracted lamellae. The top one shows different types of pixels, with two white gaps induced by the "hole" on the first lamella, the two circles in the middle image contain upper space, lamellae space and lower space inside, among which the upper space is the gap which is not expected to be touched by the circles. The bottom image is the extracted first lamella in this frame. Two wrongly extracted components corresponding to the circles in the middle image are marked. The unmarked component on the right is caused by other frames in the 3D substack.

image on the top shows different types of pixels. White ones are those directly attached to the top, thus are part of R_{top}. Red ones and green ones are candidate components that are classified as upper and lower space, respectively. Black ones are regions that were not involved by the region growing process (after the selection of the largest connected component), thus is part of R^1_{below}. White and red pixels are in R^1_{above}; black and green pixels are in R^1_{below}. Consequently, regions marked by circles on the middle image contain these two spaces as well as R_{lame}, which causes false extraction of pixels. The bottom image shows the extracted pixels including two wrongly extracted regions that are already marked in the middle image, and a region (on the right) that is induced by the same problem in other frames in the 3D substack.

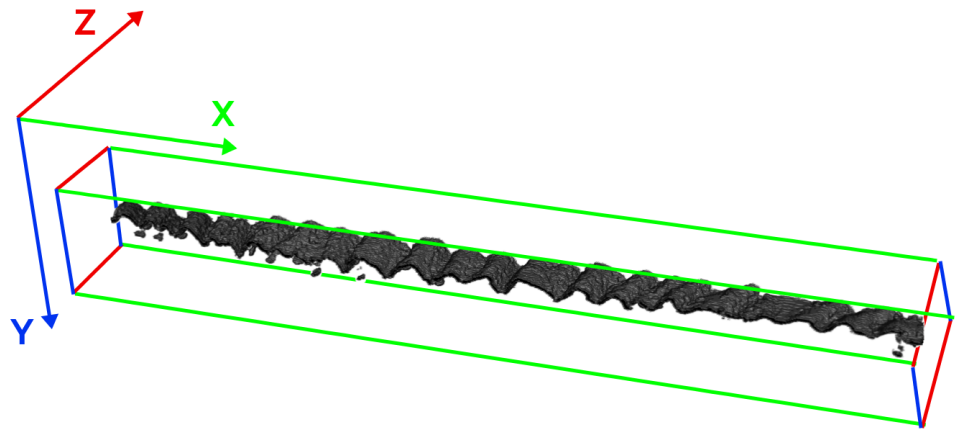
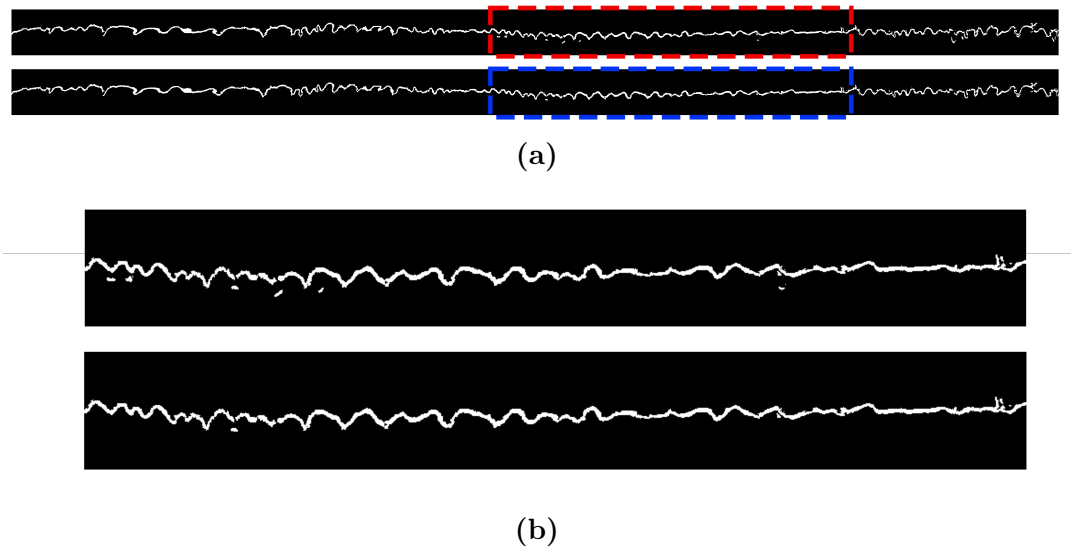


Figure 4.22: Coordinate system used for a substack.



(a)

(b)

Figure 4.23: A 2D view of extracted first elastic lamella before (row 1) and after (row 2) the post-processing. (a) shows the entire lamellae. (b) are enlarged segments marked in (a).

To eliminate them, we first examined the size of the connected components in the 3D image of the extracted voxels. The wrongly involved components are not all isolated; some of them are connected to the first lamella. Thus to avoid false elimination of the voxels of interest, we only operated on components of size smaller than 5000. The components induced by the "gaps" of the first lamella have a part below pixels in the rejected regions, which are colored in green in the illustration images above. The coordinates of voxels are noted as shown in Figure 4.22; in a substack of 32 frames, (z_v, y_v, x_v) is the coordinate of voxel v, where z_p is the number of the frame. The operation process is described in Algorithm 2. $R_{rejected}$

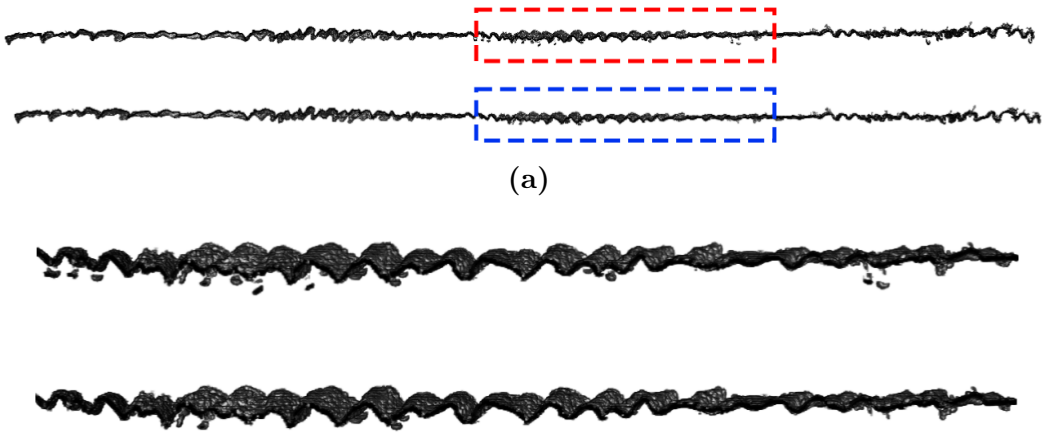
contains the rejected voxels in the space pruning process; C_{lame_1} contains all the voxels extracted as the first lamella; c_i is a component in C_{lame_1}. For each component smaller than the prefixed size, we checked the highest voxel (with the smallest y in each column, x, z fixed), if it was just below a voxel of the rejected region, then the component that this voxel belonged to would be removed. Figure 4.23 shows a comparison of a frame before and after this post-processing, with a segmented enlarged. Figure 4.24 is the result of the corresponding 3D substack.

Algorithm 2: Elimination of falsely extracted components.

Data: $C_{lame_1}, R_{rejected}$
Result: C'_{lame_1}
$C'_{lame_1} \leftarrow C_{lame_1}$;
for $c_i \in C'_{lame_1}$ **do**
  if $Size(c_i) < 5000$ **then**
    $Eliminating \leftarrow False$;
    $Searching \leftarrow True$;
    while $Eliminating = False$ ***and*** $Searching = True$ **do**
      while $\exists v \in c_i$ *s.t.* $\nexists v' \in c_i$ *with* $x_{v'} = x_v, z_{v'} = z_v, y_{v'} <$ y_v ***and*** v *not checked* ***and*** $Searching = True$ **do**
        if $\exists v_{above} \in R_{rejected}$ *with* $x_{v_{above}} = x_v, z_{v_{above}} = z_v, y_{v_{above}} = y_v - 1$ **then**
          $Eliminating \leftarrow True$;
          $Searching \leftarrow False$;
        end
      end
    end
    if $Eliminating = True$ **then**
      Remove c_i *from* C'_{lame_1};
    end
  end
end

4.2.3 Restoration of elastic lamellae to original shapes

The restoration of the lamella shape is the reverse of the arterial wall unfolding process described in Chapter 3. Having the correspondence between the positions of the rotation centers in the unfolded image and in the initial image as well as the rotating angles calculated according to the contour, we are able to map each pixel to its initial position. Figure 4.25 shows an image directly restored from the



(a)

(b)

Figure 4.24: A 3D view of extracted first elastic lamella before (row 1) and after (row 2) the post-processing. (a) shows the entire lamellae, (b) are enlarged segments marked in (a).

extracted elastic lamella; it is composed of sparse pixels. We performed a dilation with a round structural element of radius 1 and preserved the intersection of this dilated result and the corresponding binarized image obtained in the pre-processing step (Chapter 2). Figure 4.26 shows a processed 2D result and Figure 4.27a is its superimposition on the adjusted grayscale image. The 3D reconstruction composed of 512 frames of this extracted first elastic lamella is shown in Figure 4.27b.

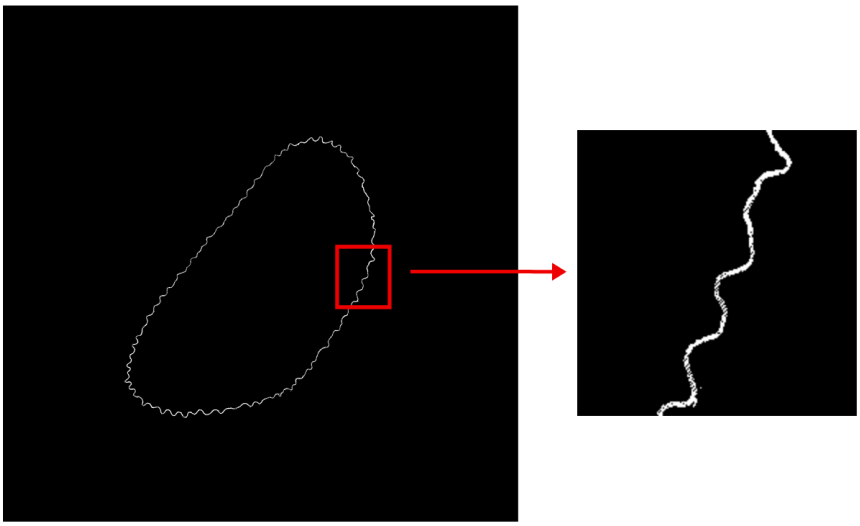


Figure 4.25: A frame of restored first elastic lamella; pixels are sparsely mapped.

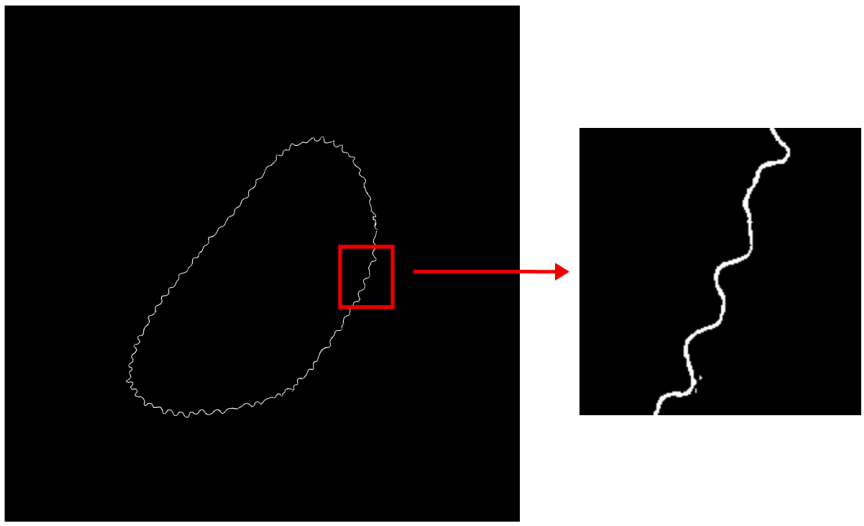
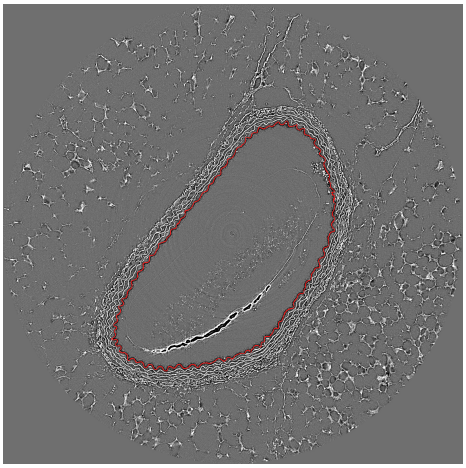


Figure 4.26: The restored elastic lamella after processing.



(a) Superimposition of the extracted lamella on an initial image (after contrast adjustment).

(b) 3D reconstruction of the extracted first lamella with 512 frames.

Figure 4.27: Post-processed restored elastic lamella illustrated in 2D and in 3D.

4.3 Texture classification to find the *media-adventitia* border

Compared to the inner side of the *tunica media*, the outer side of this structure is more complex to describe since it is in contact with the *tunica adventitia* which has similar densities and intensities as the *tunica media* and the thickness of the *media* varies in stacks and in regions of the same stack. The border between these

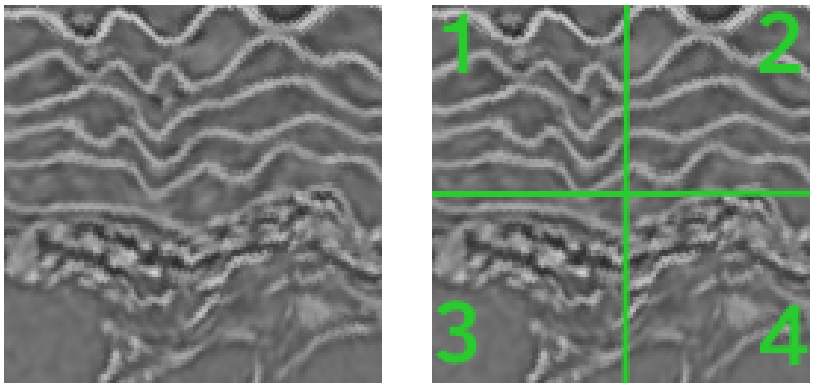
two layers can be found with eyes by distinguishing the difference in texture. This inspired us to use texture analysis methods to separate these two structures. Deep learning has been widely used to extract image features by training neural networks with a number of filters to obtain parameters adjusted to the classification goal instead of using manually designed filters. In our case, there are no labeled images for supervised learning. For this reason, the information that can be passed to the neural network is limited. However, after being unfolded, the structures in the arterial wall are placed in an order: the lamellae that are closer to the lumen are above those that are further from the lumen. We have used this information to extract inner elastic lamellae. In this section, with the help of this positional relationship between the *tunica media* and the *tunica adventitia* as well as the result of the extraction of the first elastic lamella, we propose a deep learning-based segmentation method using a Siamese network to find the boundary between the two textures.

4.3.1 Method

A Siamese network is able to distinguish a pair of inputs by generating pairs of descriptors between which the distance indicates the similarity of the inputs. With the arterial wall unfolded, we can observe that the *tunica media* is always above the *tunica adventitia*. This positional relationship can be used as a label indicating the similarity of a pair of inputs. Figure 4.28 is a patch of the unfolded grayscale arterial wall image. We can subdivide this patch into 4 parts and classify each pair of 2 parts based on their similarity. For example, patch 1 and patch 2 can be considered two similar regions, while patch 1 and patch 3 are considered dissimilar regions. The information on different textures is contained in the descriptors. Instead of passing a label of the classification of the texture around each pixel to the network and obtaining an output, we rely on the similarity of a pair of inputs. We let the network learn this similarity by minimizing the distance between outputs generated by similar inputs and maximizing the distance between those generated by dissimilar patches. The results that will be used for texture classification are the descriptors generated during this similarity learning process.

4.3.2 Dataset

Patches of size 128×128 were used for training. With the help of the result of the first elastic lamella extraction, we could localize the first layer and unfold the arterial wall with this layer on the top of the image. In Chapter 3, we used the mask



(a) **(b)**

Figure 4.28: A patch of size 128 × 128 with *tunica media* above *tunica adventitia* (a) and the same patch divided into 4 regions (b). Similarities can be distinguished from different pairs.

of the general shape of the lumen to unfold an image along its contour. To unfold an image along the first layer, we also need a mask enclosed by a closed contour. We used the restored first elastic lamella to generate these masks. However, the first lamella is not always a closed structure. To generate masks as wall as closed contours, with the lumen mask I_{lumen} that we obtained in the process in Chapter 3 and the image of the first elastic lamella I_{el}, we perform the operation described in Algorithm 3.

Algorithm 3: Generation of mask and contour for single elastic lamella.

Data: $I_{la}, I_{lu}, r_d, r_e, r^1_{Area}, r^2_{Area}$
Result: $I_{mask}, I_{contour}$
$increase\_radius \leftarrow True$;
$r_d \leftarrow r_{d_{init}}$;
$r_e \leftarrow r_{e_{init}}$;
while $increase\_radius = True$ **do**
  $(I_{mask}, I_{contour}) \leftarrow f_{in}(I_{la}, I_{lu}, r_e, r_d)$;
  if $Area(I_{mask}) < \frac{Area_{total}}{r^1_{Area}}$ ***or*** $Area(I_{mask}) > \frac{Area_{total}}{r^2_{Area}}$ **then**
    $r_d \leftarrow r_d + \Delta r_d$;
    if $r_d > r_e - \Delta r_{d,e}$ **then**
      $r_e \leftarrow r_d + \Delta r_{d,e}$;
    end
  else
    $increase\_radius \leftarrow False$;
  end
end

The function $f_{in}(I_{la}, I_{lu}, r_d, r_e, r^1_{Area}, r^2_{Area})$ in Algorithm 3 is the process of generating an outer contour of a single elastic lamella and the mask of the region enclosed by it with an image of the lamella I_{la} and the mask of the lumen I_{lu} that we obtained in the lumen extraction process in Chapter 3. $r_{e_{init}}$ and $r_{d_{init}}$ are values set to initialize r_e and r_d; r^1_{Area} and r^2_{Area} are parameters that verify if the mask corresponds to the lumen; their values depend on the operated stack. This process is shown in Algorithm 4.

Algorithm 4: Process of generating a mask and a contour of the space enclosed by the first elastic lamella.

Function $f_{in}(I_{la}, I_{lu}, r_d, r_e)$**:**

$I^d_{la} \leftarrow Dilation(I_{la}, r_d)$ `/* Dilation in image` I_{la} `with structuring element of radius` r_d `*/`

$I^e_{lu} \leftarrow Erosion(I_{lu}, r_e)$ `/* Erosion in image` I_{lu} `with structuring element of radius` r_e `*/`

Select a point(s) $p \in I^e_{lu}$ and $p \notin I^d_{la}$

$I^{rg}_{lu} \leftarrow RG(I^d_{la}, p)$ `/* Region growing in image` I^d_{la} `with seed point p */`

$I^{rg}_{lu} \leftarrow Dilation(I^{rg}_{lu}, r_d - 1)$

$I^{rg}_{lu} \leftarrow Closing(RG(I^{rg}_{lu}, p), 3)$ `/* Morphological closing with structuring element of radius 3 */`

$I_{contour} \leftarrow$ Largest closed contour of $I^{rg}_{lu} + I_{la}$

$I_{mask} \leftarrow$ Space enclosed by $I_{contour}$

return I_{mask}, $I_{contour}$

The regions covered by these masks are those enclosed by the outer contour of the first elastic lamella. With these masks, we can unfold the image again more accurately so that the first elastic lamella is just on the top of each unfolded image. The extracted contour is on the first lamella. For unfolding, the rotation center should be near the middle of the arterial wall. Thus we applied a dilation with a round structural element of radius $r_{el_{dilate}}$ on the masks and got their contours. Images were unfolded with these dilated masks and the contours. These images were then denoised by NL-means filtering. Figure 4.29 illustrates an unfolded grayscale image with the first lamella on the top. The height of the image is 128, and the width is the length of the contour extracted from the dilated mask. The radius $r_{el_{dilate}}$ was set to 50. A denoised unfolded image was cut into patches of size 128×128 by a moving window sliding from the left to the right of the image with step 64. In these patches, the upper part is filled mainly by elastic lamellae and the lower part contains tissues of the *tunica adventitia*. We cut the subpatches as shown in Figure 4.30: the middle regions of the upper and the lower parts of each patch were

cut and combined as similar and dissimilar pairs to enter the Siamese network.

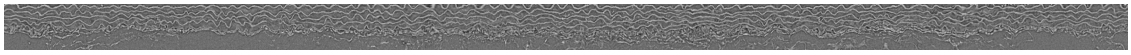


Figure 4.29: An unfolded grayscale image with the contour of the dilated mask. The first elastic lamella is on the top of the image.

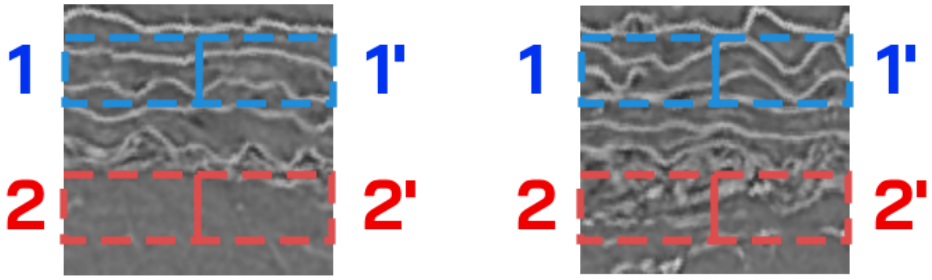


Figure 4.30: Two patches with the 4 subpatches marked. Subpatches 1 and $1^{'}$ represent the texture of the *tunica media*. Subpatches 2 and $2^{'}$ represent the texture of non *tunica media*; it can be the *tunica adventitia* or the structure outside it. Subpatches are cropped from the middle of the upper half and the lower half of the whole patch. The patch on the left contains less texture of the *tunica adventitia* than the one on the right.

4.3.3 Neural network architecture

The architecture of this Siamese network is illustrated in Figure 4.31. The input is a subpatch of size $(H, W) = (32, 64)$, $ConvRelu$ layers in the figure are convolutional layers followed by a ReLU function, $ConvRelu1$ to $ConvRelu6$ are groups composed of two "convolutional layer-ReLU" structures with convolution kernels of size 3×3. The numbers of channels are 64-64-64-64-128-128-128-128-256-256-256-256. $ConvRelu7$ is a convolutional layer with a kernel size of 1×1 and 256 channels followed by a ReLU function. Each one of $ConvRelu1$ to $ConvRelu6$ is followed by a max pooling; the last pooling is applied on the X direction. The last layer is also a fully convolutional layer with kernel size 1×1 and it outputs a descriptor of size 16.

4.3.4 Loss function

The 4 subpatches P_{ij} obtained from one patch P enter the same network. This leads to 4 output (normalized) descriptors d_{ij} of length 16, combined as 6 pairs of

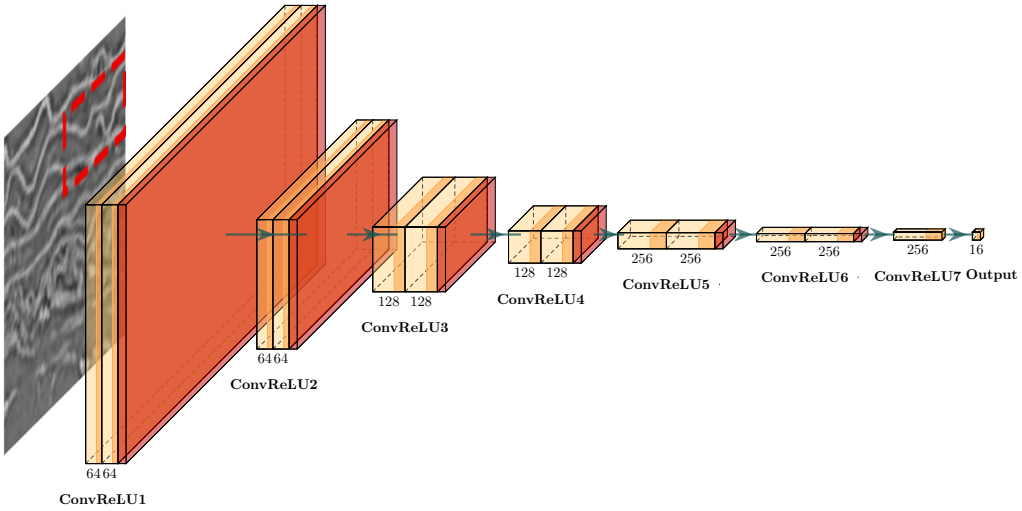


Figure 4.31: Architecture of the network. The input is of size $(H, W) = (32, 64)$, the output is a vector of size 16.

(distinct) descriptors. The loss associated with a pair of subpatches is defined as:

$$\mathcal{L}(d_{ij}, d_{kl}) = \alpha.\delta_{ik}.\Delta^T.\Delta \ + \beta.(1 - \delta_{ik}).[\max\{0, m_p - (\Delta^T.\Delta)^{1/2}\}]^2 \tag{4.27}$$

where $\Delta = d_{ij} - d_{kl}$, δ is the Kronecker symbol and $\alpha = 2$, $\beta = 1$, $m_p = 2$. This is a contrastive loss. The descriptor is normalized to a Euclidean distance of 1. The value of δ is 1 when the inputs are similar patches. In this case, the product $\Delta^T.\Delta$ gives a smaller result of the loss function when the descriptors are close to each other. With dissimilar inputs, δ is set to 0, and $\Delta^T.\Delta$ gives a smaller result when the descriptors are not similar. The use of $\max\{0, m_p - (\Delta^T.\Delta)^{1/2}\}$ is to control the range of variation of different descriptors. In our case with $m_p = 2$ and d_{ij} as well as d_{kl} normalized to size 1, this term equals $m_p - (\Delta^T.\Delta)^{1/2}$.

4.3.5 Results

A stack of 2048 frames generates about 100000 patches. 10 epochs were performed on each stack for generating its own model. In each epoch, 80% of the patches were used for training and the rest were for validation. The training process was done on the images that need to segment so we did not test the similarity classification (with results 0 or 1) between the 4 subpatches because that was what the training did on the same dataset. We evaluated the performance of a model directly by pixel-level prediction. The method is illustrated in Figure 4.32. For

a patch cropped from an unfolded grayscale image, the *media-adventitia* border is assumed to be between $y = \frac{L}{4}$ and $y = \frac{3L}{4}$ in a patch of size $L \times L$ ($L = 128$ in our case). To predict the classification of the pixel p at (y_p, x_p), two sub patches of reference of size $(H, W) = (\frac{L}{4}, \frac{L}{2})$ (H: height, W: width) with centers at $(\frac{L}{4}, x_p)$ and $(\frac{3L}{4}, x_p)$ are cropped in order to generate two descriptors representing the *media* and the non-*media* textures. Pixels p with $y_p \in (\frac{L}{4}, \frac{3L}{4})$ and $x_p \in (\frac{L}{4}, \frac{3L}{4})$ are predicted by comparing the descriptor Des_p generated by the subpatch $P_{(y_p,x_p)}$ centered by (y_p, x_p) with the two descriptors Des_{r1} and Des_{r2} generated by the subpatches of reference $P_{sub}(\frac{L}{4}, x_p)$ and $P_{sub}(\frac{3L}{4}, x_p)$, respectively, with the following rule:

$$T_p = \begin{cases} media & if\ \|Des_p - Des_{r1}\| < \|Des_p - Des_{r2}\| \\ non\text{-}media & otherwise \end{cases} \tag{4.28}$$

where T_p is the texture class to which p belongs.

Figure 4.33 shows several predictions on patches; the two classes of textures are colored in red and green. The classification of a pixel p is decided by the comparison of similarities between the subpatch $P_{sub}(p)$ of p and the two subpatches of reference by which $P_{sub}(p)$ is bounded. The similarity is measured by the Euclidean distance between the descriptors generated by the trained neural network.

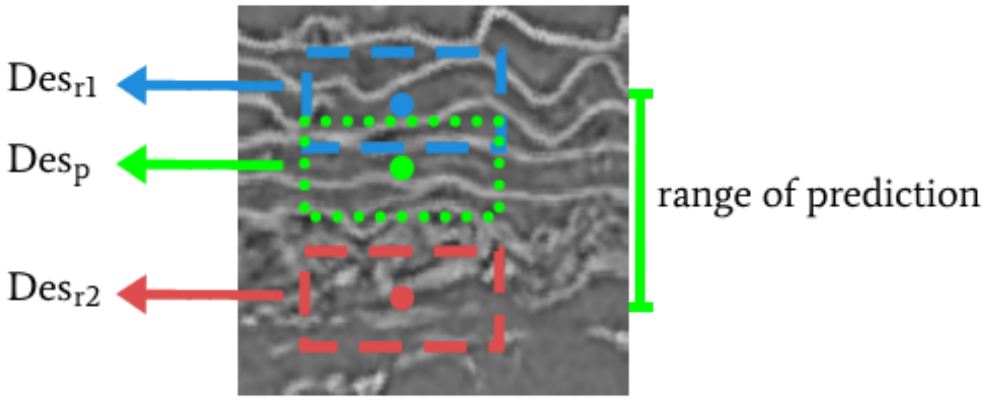


Figure 4.32: Pixel-level prediction method. Des_p generated by the subpatch centered at the green pixel is compared with those generated by the two reference patches above and below it. The range of prediction is $y \in (\frac{L}{4}, \frac{3L}{4})$.

We applied this predicting method on entire unfolded images. Pixels with $y \in (\frac{L}{4}, \frac{3L}{4})$ were predicted by comparing their descriptors with those generated by the subpatches of reference above and below them. Figure 4.34 illustrates the result of a such prediction.

Since the prediction is pixel-level, the pixels classified as *media* are not all above

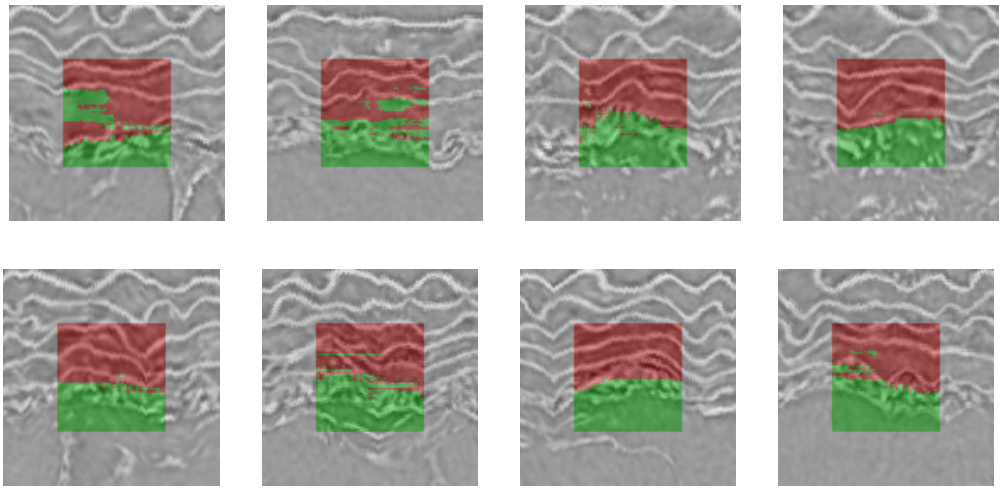


Figure 4.33: Eight patches predicted at the pixel level. Red pixels are *media* texture and green pixels are non-*media* texture.

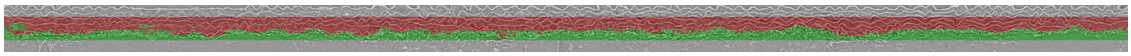


Figure 4.34: A prediction result on an unfolded arterial wall. Red pixels are *media* texture and green pixels are non-*media* texture.

those classified as non-*media*, as shown in the previous figures. According to the fact that the *media* is always above the *adventitia*, to accelerate the prediction process, we checked pixels from bottom to top in each column and once pixels classified as *media* continuously appeared more than twice, the rest of the column was no longer checked and was classified as *media*. Figure 4.35 illustrates the result obtained by this method, corresponding to the same unfolded image in Figure 4.34.

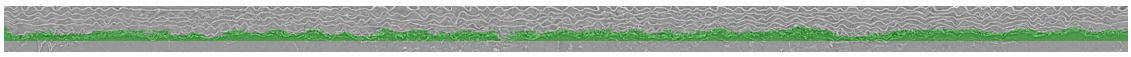


Figure 4.35: A prediction result with non-*media* textures colored in green. Pixels are predicted from bottom to top.

To get a continuous smooth *media-adventitia* border, we approximated a closing contour by selecting the highest pixel classified as non *media* every 25 columns of the unfolded image and mapped them to their original positions in the image before unfolding. These points are used to generate a parametric closed spline with the starting and ending points at the same position. A frame of contrast-adjusted image superimposed with the spline is illustrated in Figure 4.36.

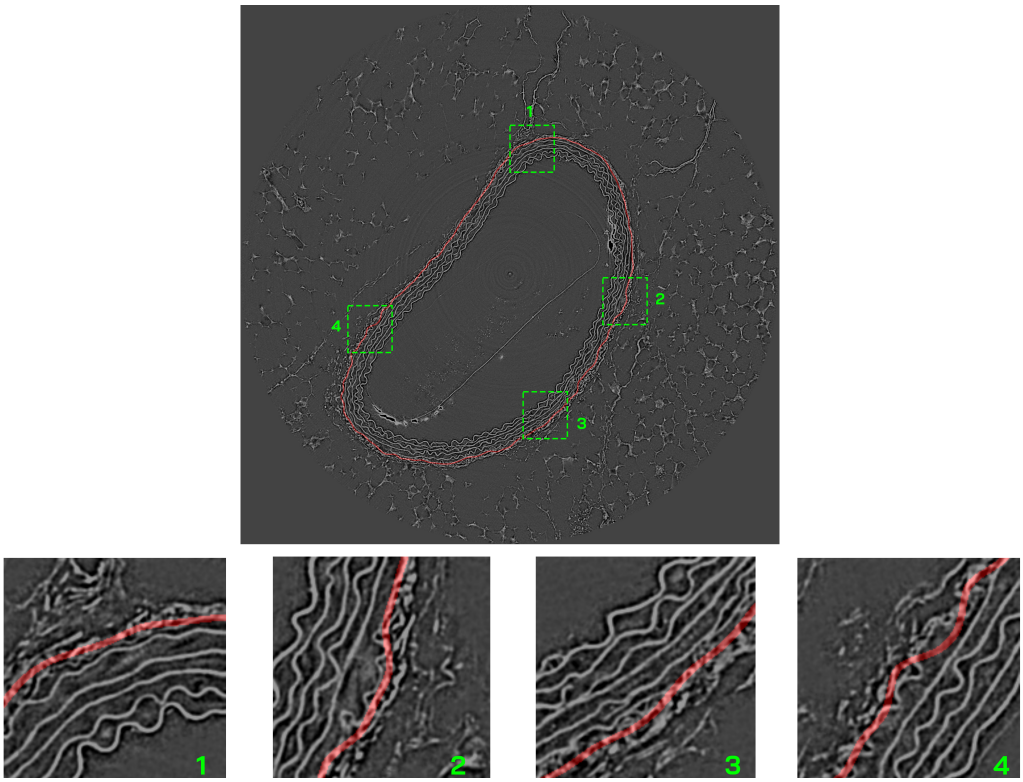


Figure 4.36: *Media-adventitia* border (red) approximated by a spline from the top points selected every 25 columns in the unfolded image.

4.4 Conclusion

In this chapter, two types of segmentation are described: extraction of the first elastic lamella and classification of the *tunica media* and the *tunica adventitia*. Both operations are applied to unfolded images, one on binary images and one on grayscale images. The first segmentation uses a region growing-based method. The idea is to extract the space above the first elastic lamella. A 3D region growing is first applied on segments along the unfolded lumen contour in the pre-processed binary image. This enables us to find the space hidden in the curves. Then, space pruning is performed to reject the falsely grown regions by measuring the weight of voxels under the potential leak points and the size of these points. The 3D components of which the ratio between these two values is higher are rejected. In this way, the space above and below the first lamella can be distinguished and thus the frontier between them, where lies the first lamella, can be found by searching for the voxels in the binarized image that contain both the upper and the lower space in the neighborhood. This first lamella extracted from the unfolded image can be restored to its initial shape, which is more accurate than the lumen contour extracted in the

lumen extraction step (Chapter 3). The outer contour of the lamella enables us to unfold the arterial wall again to perform the second segmenting aiming to find the border between the *tunica media* and the *tunica adventitia*. A Siamese network is built with a training set composed of patches cropped from the unfolded grayscale image and labeling of input patches according to their spatial information. This step performs a texture segmentation by learning the similarities and dissimilarities between patches that can be regarded as similar and dissimilar pairs. The pixel-level prediction is realized by comparing the similarities between the patch centered at the pixel to predict and two patches of which the classifications have been assumed. The border is then approximated by a parametric closing spline from the points selected in the unfolded predicted image and mapped back to their initial positions. The region of the *media* is then segmented. These two types of results lead us to different analyses of elastic lamellae in the next chapter.

Chapter 5

Elastic lamellae analysis

In the work described in the previous chapter, we extracted the 3D structure of the first three elastic lamellae and the border between the *tunica media* where lie the elastic lamellae and the *tunica adventitia*. In this chapter, the results of different measurements calculated on the segmented images will be shown and some analysis of these features will be presented. The analysis is divided into two main parts: a 3D curvature analysis on the first elastic lamellae and a 2D waviness analysis on the fragments throughout all the elastic lamellae.

5.1 Related works

Several works are dedicated to analyzing the geometric properties of elastic lamellae, some of which were already mentioned in Chapter 1. Walton *et al.* [51] visualized the structures in unpressurized and pressurized arteries and found that when intra-luminal pressure increased, the lumen cross-sectional area increased, the medial elastic lamellae were straightened, and profound remodeling of the *tunica adventitia* was induced. In the work of Yu *et al.* [50], the centerlines of the first three elastic lamellae were extracted, with each lamella fitted by an arc. A straightness parameter is defined as:

$$P_s = \frac{L_a}{L_c} \tag{5.1}$$

where L_a is the length of the arc and L_c is the contour length of the lamella, as shown in Figure 5.1. Interlamellae distance can then be evaluated by the radii difference between arcs. A mechanical test to characterize the pressure-diameter response was performed by increasing intraluminal pressure. The tissue deformation was decom-

posed by lamellar unfolding and lamellar stretching. Trachet *et al.* [54] carried out a similar analysis as in [50]. They studied the evolution of lamellar diameter, length, and straightness with increasing pressure. Besides, the lamellar, interlamellar, and *adventitia* thicknesses were also analyzed. In the pressure-inflation experiments performed on the carotid artery of a mouse (three lamellae and no side branches), they found that as pressure increased, all three layers were unfolded and simultaneously stretched out, with the lamellar length increasing quasi-linearly. During this process, a decrease in the thickness of the wall was mostly observed in the outer layers. At the same pressure, the *tunica adventitia* had a greater percentage of thickness reduction than the *tunica media*. López-Guimet *et al.* [55] analyzed elastic lamellae injuries of the ascending aorta with Marfan syndrome. An analysis performed in this study that was not mentioned above is that of fenestrae. The change of fenestrae density and size of the internal elastic lamella (the frontier between intimal endothelium and the *tunica media*) and the successive second lamella were analyzed. These two layers were injured to a different extent in murine Marfan aortae, with the fenestrae density and size changed. Instead of occurring uniformly in the entire ascending *media*, these changes were mostly restricted to the proximal and convex regions (as defined in Figure 5.2). Logghe *et al.* [134] analyzed the aortic dissection in BAPN/AngII-infused mice, a mouse model for aortic dissection. Ruptures on the elastic lamellae were found in 3D images and were quantified both in size and amount. A steep increase in ruptures occurred after 3 days of BAPN/AngII-infusion. The numbers of ruptures affecting only the first lamella and multiple layers were 133 and 135, respectively. Medial ruptures through all the layers, which lead to intramural hematoma, were only found in the thoraco-abdominal aorta; interlamellar hematoma formation in the ascending aorta could be directly associated with the innermost lamellar ruptures.

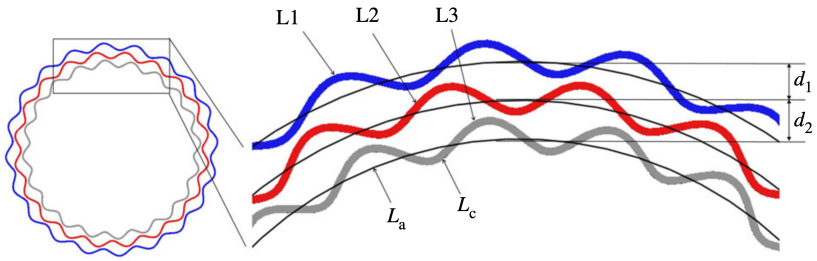
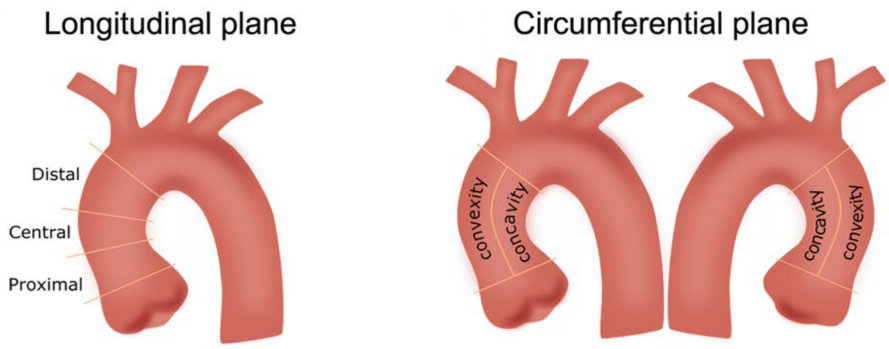


Figure 5.1: Analysis of lamellar straightness in the work of Yu *et al.* [50].



(a) **(b)**

Figure 5.2: Ascending aorta anatomical regions defined in the work of López-Guimet *et al.* [55] in longitudinal (a) and circumferential (b) planes.

5.2 Analysis of curvatures of the first elastic lamella

In Chapter 4, we extracted the first elastic lamellae. Each entire layer forms a 3D structure containing geometric information that enables us to extract various features and analyze their deformations from a biological point of view. We calculated 3D curvatures all around an elastic lamella and generated histograms of these curvatures in order to compare this distribution between different samples and analyze the geometric deformations related to pathologies.

5.2.1 Method

According to the work [18] built upon the previous works of differential estimators on smooth surfaces based on integral invariant [19] [20], we estimated the 3D curvature H on each voxel x of the first lamella by:

$$H_r(X, x) = \frac{8}{3r} - \frac{4V_r(x)}{\pi r^4} \tag{5.2}$$

where X is the surface where x lies, r is the radius of the sphere centered at x and $V_r(x)$ is the volume inside the space enclosed by the surface and inside the sphere. Figure 5.3 illustrates the positivities of curvatures at different points calculated in this way.

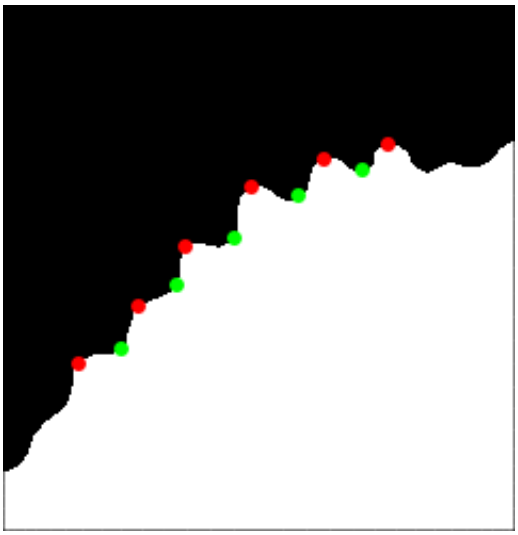


Figure 5.3: 2D illustration of curvature positivities. The image is a part of the mask of the space inside the outer contour of the first elastic lamella. Curvatures on the border are calculated, green points have negative curvatures and red points have positive curvatures.

After several experiments, we chose $r = 15$ (corresponding to a real radius of 9.75 μm) as the sphere radius and generated a histogram of the frequency of curvature range for each stack. $V_r(x)$ was the counted number of voxels inside the sphere and the enclosed space, the bin size was set to 0.005.

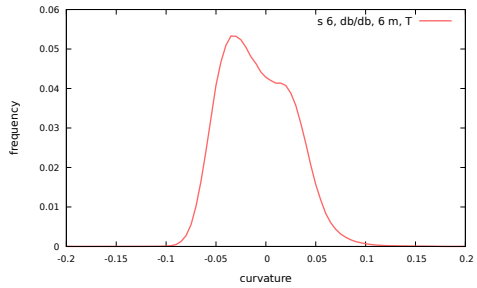
5.2.2 Dataset

To apply this calculation of curvature, we need images of extracted elastic lamellae with the inside filled with white pixels in order to distinguish positive curvatures and negative curvatures. These images were already generated in the step described in Section 4.3, for the preparation of training samples.

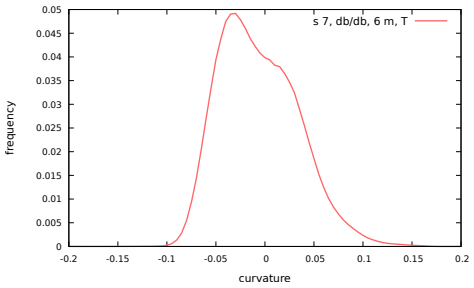
Since the first lamella is attached with impurities in some stacks, we used only those that have a relatively clean first lamella of which the extracted curves can show precisely the shape of the lamella without introducing much error. 18 stacks (10 diabetic and 8 normal) were used for this analysis. We calculated the curvatures of the points on the mask contours that also existed on the extracted first lamella since the masks had enclosed contours on bifurcations where there is no extracted lamella. We rejected the frames where the lamellae shapes were not applicable in the original images or the mask borders were not accurate enough to form a smooth and continuous 3D border with consecutive frames .

5.2.3 Results

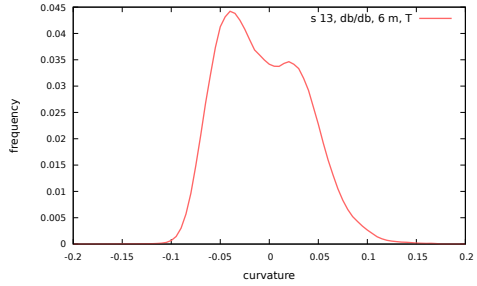
Figure 5.4, 5.5 and Figure 5.6, 5.7 show the histograms of 10 curvature histograms of diabetic mice and 8 curvature histograms of normal mice, respectively. Figure 5.8 shows the superimposition of the histogram of each group.



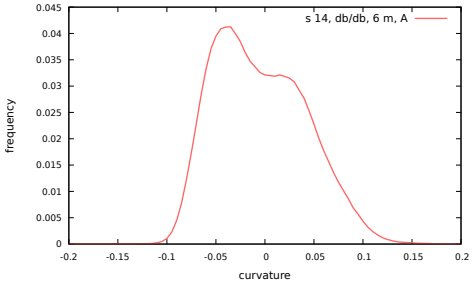
(a) Stack 6, 6 months, thoracic.



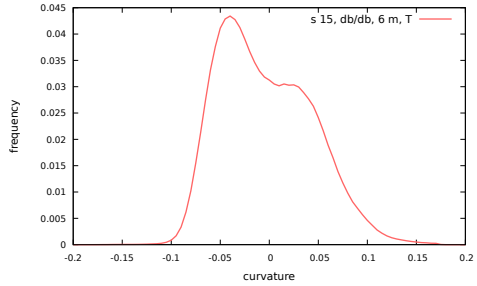
(b) Stack 7, 6 months, thoracic.



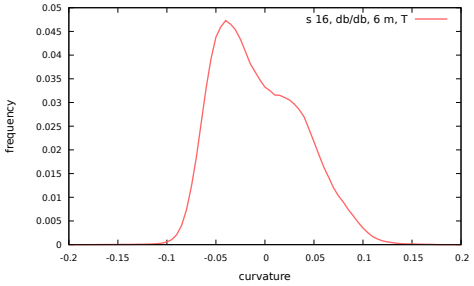
(c) Stack 13, 6 months, thoracic.



(d) Stack 14, 6 months, abdominal.

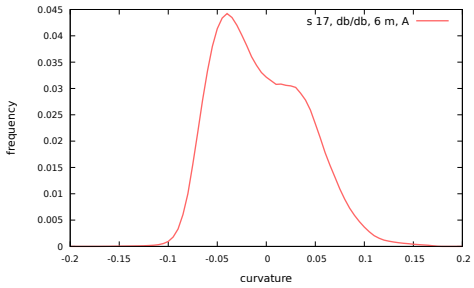


(e) Stack 15, 6 months, thoracic.

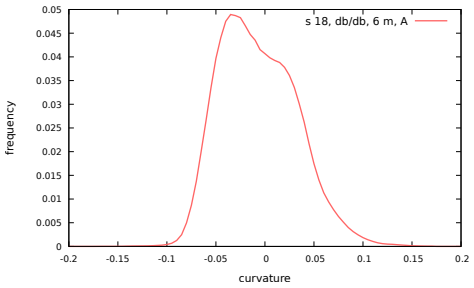


(f) Stack 16, 6 months, thoracic.

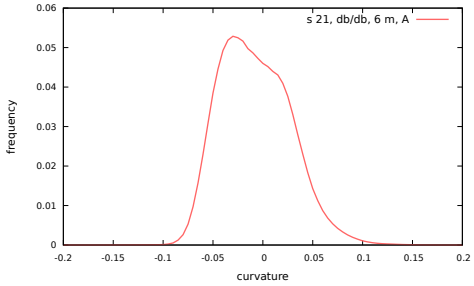
Figure 5.4: 3D curvature histograms of 10 diabetic stacks (part 1).



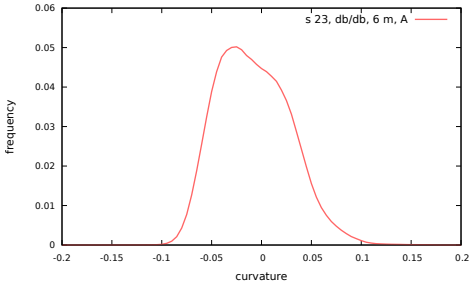
(a) Stack 17, 6 months, abdominal.



(b) Stack 18, 6 months, abdominal.

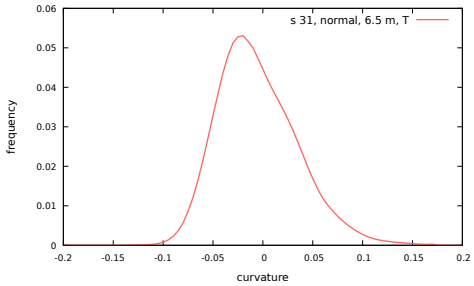


(c) Stack 21, 6 months, abdominal.

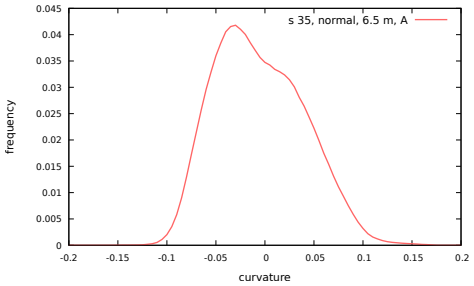


(d) Stack 23, 6 months, abdominal.

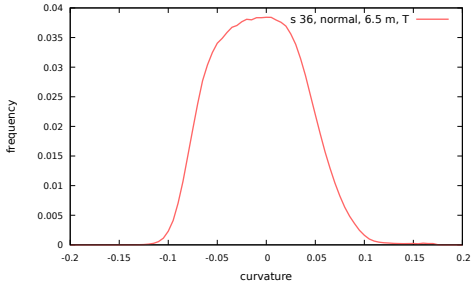
Figure 5.5: 3D curvature histograms of 10 diabetic stacks (part 2).



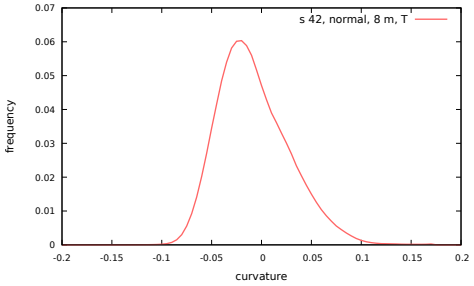
(a) Stack 31, 6.5 months, thoracic.



(b) Stack 35, 6.5 months, abdominal.

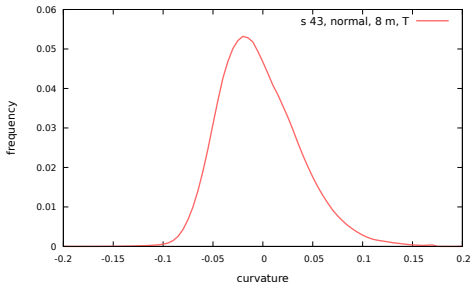


(c) Stack 36, 6.5 months, thoracic.

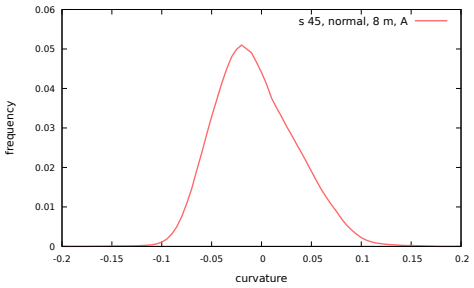


(d) Stack 42, 8 months, thoracic.

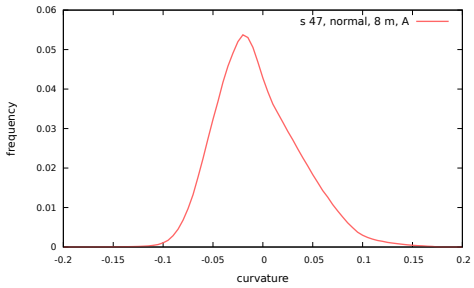
Figure 5.6: 3D curvature histograms of 8 normal stacks (part 1).



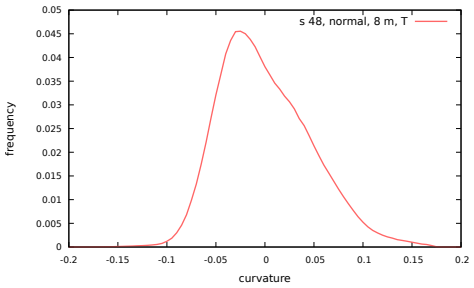
(a) Stack 43, 8 months, thoracic.



(b) Stack 45, 8 months, abdominal.

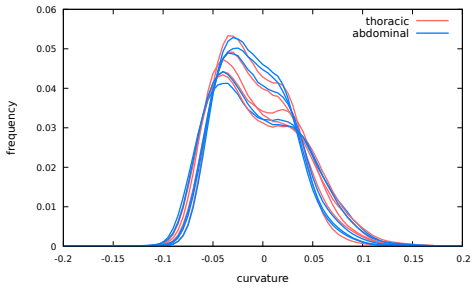


(c) Stack 47, 8 months, abdominal.

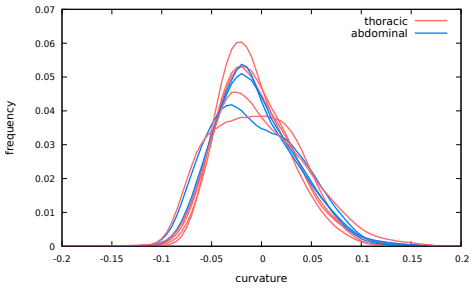


(d) Stack 48, 8 months, thoracic.

Figure 5.7: 3D curvature histograms of 8 normal stacks (part 2).



(a) Diabetic stacks.



(b) Normal stacks.

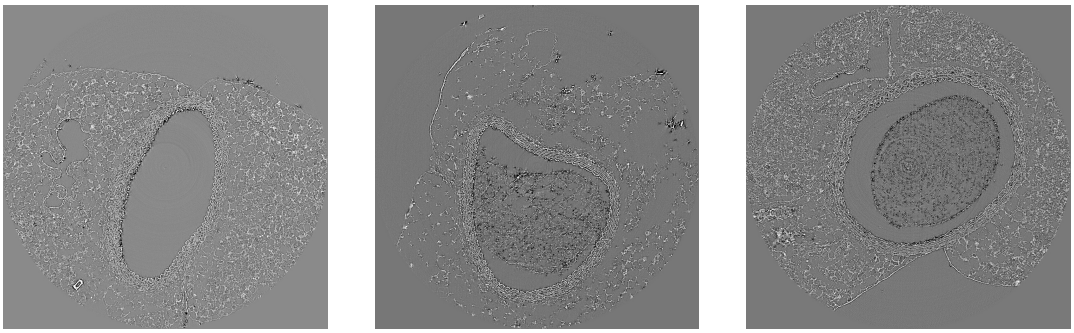
Figure 5.8: Superimpositions of 3D curvature histograms of diabetic (a) and normal (b) stacks. Thoracic and abdominal stacks are distinguished by orange and blue curves respectively.

We observed from these results that, for most of the histograms of normal mice, the distribution of the 3D curvatures has a peak at a negative value. In contrast, in those of diabetic mice, we can see another peak (or a shoulder) on the right, at a positive value. Except for stack 36, the histograms could be decomposed into two modes with one centered at a negative value, and one centered at a positive value. No difference is observed between thoracic and abdominal samples in the curvature aspect. Variability exists in the samples and, in general, the difference between the two groups is that the diabetic stacks have a more visible mode centered at the

positive side, which generates a second peak or shoulder. This shows that in diabetic stacks, the first elastic lamella has two dominant curvature distributions in ranges around a negative value and a positive value respectively. In the normal stacks, the distribution around the positive value is relatively close to the one around the negative value or its height is not as marked as in diabetic stacks. Further analyses are needed to find the cause of this distribution difference and possibly interpret its biological meaning.

5.3 Analysis of the fragment waviness

From the second to the outermost elastic lamella, the structures are not always complete as the first layer which can form a circle. As was analyzed in Chapter 2, the number of elastic lamellae is not 5 everywhere. The further outward the lamellae go, the more complex their structure and position relationships become. We are not always able to precise the exact number of a part of a lamella and divide all the lamellae into 5 groups. Thus, for all these layers in the *tunica media*, we analyzed their geometric properties with the fragments instead of an entire one as the previous analysis. By direct observation, we can find that the elastic lamellae of diabetic mice are smoother than those of normal mice (Figure 5.9), which means that there may be a difference in curvature between these two groups of mice. In addition, the smoothness of the mice of great age is higher, so it is worth comparing this smoothness/waviness of groups of mice divided by their health condition and by their age to measure the correlation between these factors and their geometric properties.



(a) **(b)** **(c)**

Figure 5.9: Three images for comparison (with contrast enhancement for visibility). (a), (b) and (c) are from a normal mouse of 6.5 months, a diabetic mouse of 6 months and a normal mouse of 12 months respectively. Smooth parts of elastic lamellae can be observed in (b) and (c) (in the upper right part and the lower right part of the arteries respectively).

To quantify this waviness, we proposed a measurement for elastic lamellae fragments with various lengths, which measures the extent to which the length is "hidden" or "retained". For a curve obtained by skeletonizing a fragment of lamella, its waviness was measured by comparing the curve itself with a fitted open polygon composed of straight lines, the ratio of the length of the curve and the polygon indicates the length hidden in the curvy parts. The closer the ratio is to 1, the smoother the curve is.

5.3.1 Pre-processing

The images for such an analysis were skeletonized fragments of elastic lamellae. These images were obtained from binary images of the lamellae with thickness. After the skeletonization, fragments were of thickness 1, and then the path connecting the start and the end points of each fragment was easy to find. A pre-processing was applied to these skeletonized images to remove small branches. Figure 5.10 shows two cases of connections of branches. We detected different cases of bifurcations and removed them with the kernels in Figure 5.11.

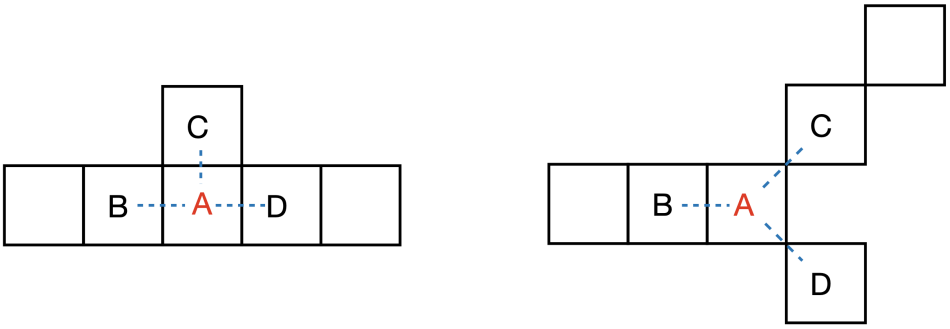


Figure 5.10: Two cases of branches in a skeletonized image. In the first image, pixels B, C and D are vertically or horizontally connected to A; in the second image, B is horizontally connected to A, C and D are diagonally connected to A.

In this method, pixel values in the skeletonized image (denoted $I_{skeleton}$) are transformed to 1 (white) and 0 (black) and the image is convolved with these kernels. The process is divided into four steps:

1. Detection of branch points. In the image of $I_{skeleton}$ convolved with kernel a, for a pixel p that has value 0 in $I_{skeleton}$, its value in this convolved image is turned to 0 too. This result is denoted as I_a. Then for pixels in the image convolved with kernel b_1, positive values are turned to 1, the result is denoted as I_{b_1}. Then the image $I_{skeleton}$ is convolved with kernel b_2 (denoted I_{b_2}). Pixels

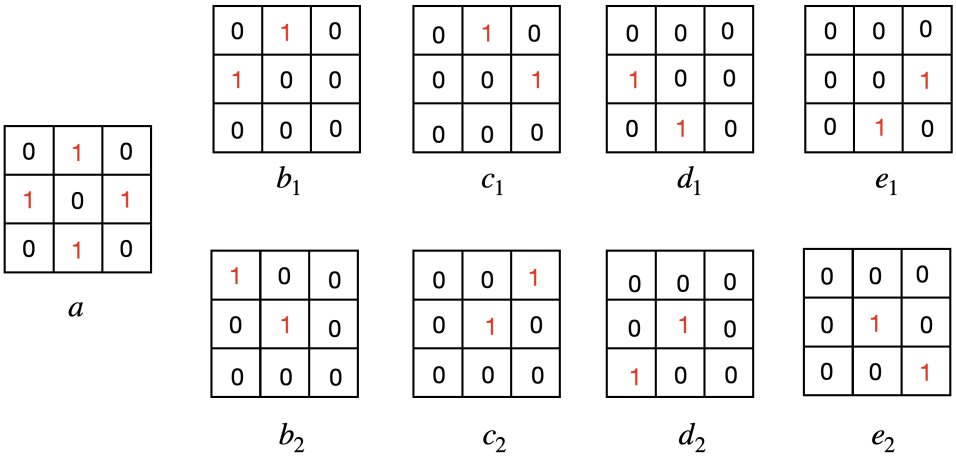


Figure 5.11: Kernels used for branch detection.

with value of 1 in I_{b_1} and with value of 2 in I_{b_2} retain their value of 1, while others are turned to 0. This result is denoted as I_b. The same operation is applied to $I_{skeleton}$ with kernels c_1, c_2, d_1, d_2, e_1 and e_2, with results denoted as I_c, I_d, and I_e, respectively. The sum of these 5 images is denoted I_{branch}:

$$I_{branch} = I_a + I_b + I_c + I_d + I_e \tag{5.3}$$

For a pixel p, if $I_{branch}(p) \geq 3$, it is noted as a bifurcation point.

2. Detection of branch neighbors. With the branch points, branch neighbors are found by 8-connectivity. As in Figure 5.10, B, C and D are branch neighbors of A.

3. Detection of end points. The operations in step (1) are applied, pixels with $I_{branch}(p) = 1$ are noted as end points.

4. Detection and removal of small branches. The curves that are potentially branches to be cleaned are the curves between an end point and a neighbor of a branch point (included). The detection starts with an end point and goes forward pixel by pixel until a bifurcation neighbor is reached. It is realized by the following steps:

   (a) Find points that are not only end points but also branch neighbors; then delete them.

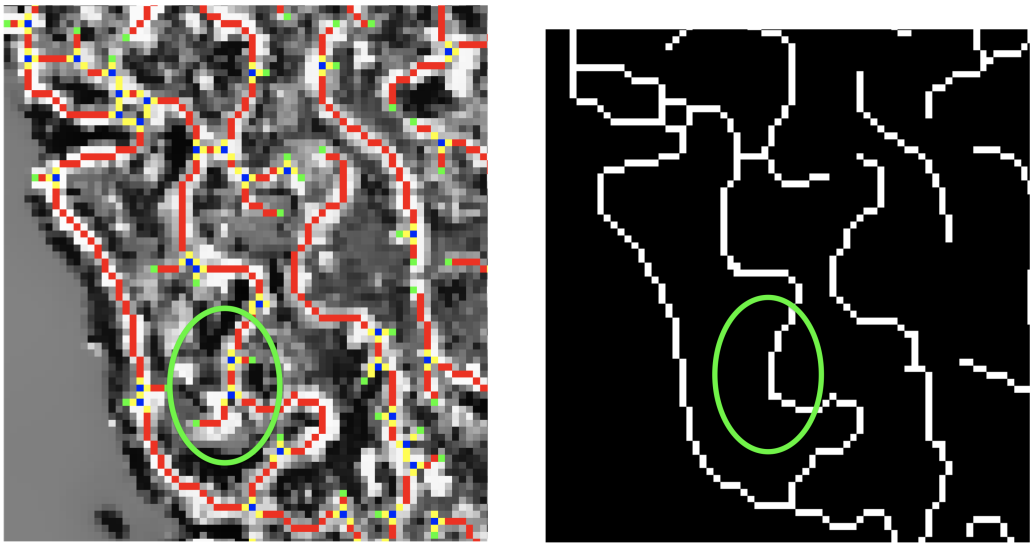
   (b) Find branch points, branch neighbors and extremities.

   (c) For each end point, starting from that point, we move pixel by pixel in

the opposite direction to the end, counting the length of the path. If the path is not adjacent to a branch neighbor or another end point, but the path length has reached L_{branch}, then the path is not noted. When the path length has not been noted, if the point to be touched (the next point to be advanced) is a branch neighbor, the path from the end to this branch neighbor is noted as a branch. When the length of the path does not reach L_{branch}, but the point to be touched is another end point, this path is not noted. L_{branch} is a fixed length; we set it to 10 in our implementation.

(d) Noted branches are removed from $I_{skeleton}$.

Figure 5.12 shows such a branch-removing process and the result. Red pixels form a part of a skeletonized image superimposed on an initial grayscale image of elastic lamellae. Blue pixels are branch points, yellow pixels are branch neighbors and green pixels are end points. Curves between yellow and green points of lengths under L_{branch} are removed. For example, in the result on the right, two branches in the circle are removed.



(a) **(b)**

Figure 5.12: A part of a skeletonized image superimposed on a grayscale image of elastic lamellae (a) with different types of pixels marked and the results of skeleton after a branch removal (b).

5.3.2 Waviness measurement

The measurement of waviness is trivially defined by the relation:

$$\rho = \frac{L_{Euc}}{L_{geo}} \tag{5.4}$$

where L_{Euc} is the Euclidian length defined by:

$$L_{Euc} = \sum_{0 \le j < L_{geo}/\sigma,\ (j+1)\sigma \le L_{geo}} \|V_{(j+1)_\sigma} - V_{j_\sigma}\|_2 \tag{5.5}$$

and L_{geo} is the geodesic length defined by:

$$L_{geo} = \sum_{0 \le i < k} \|V_{i+1} - V_i\|_2 \tag{5.6}$$

In these two equations, k is the number of pixels and V_i are vertices of the curve.

Figure 5.13 is an illustration of the measurement. White pixels are the initial skeletonized curve with red points added. These red pixels subsampled points that will be explained later. The green lines form a fitted open polygon to calculate the L_{Euc}. This length is calculated based on L_{geo} since the selection of vertices V_i depends on L_{geo}.

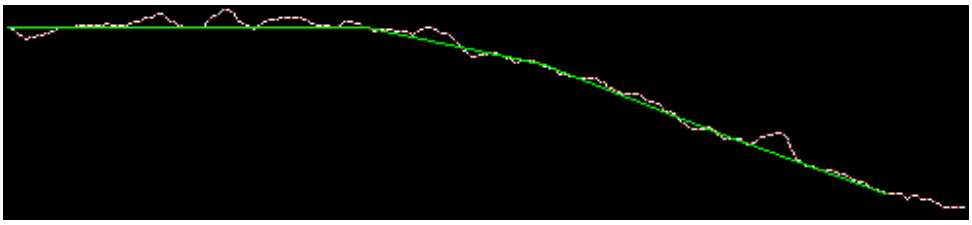
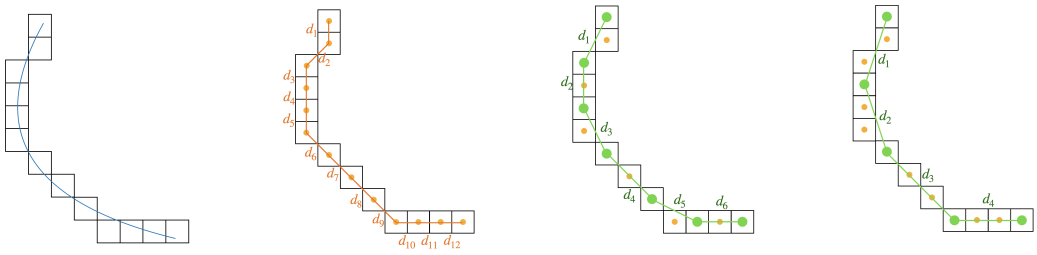


Figure 5.13: Illustration of the waviness measurement. An initial curve is shown in white. Red pixels are subsampled vertices. An open polygon (in green) is generated by connecting a subset of the vertices.

L_{geo} is the geodesic distance representing the "real" curve. In the skeletonized image, curves are shown by discrete pixels, while the real curve is a continuous structure. The L_{geo} is thus an estimated length of the real curve. As is shown in Figure 5.14, the length of a curve can be calculated by adding consecutive distances between pixels:

$$L = \sum_{i=1}^{(N-1)/s} d_i \tag{5.7}$$

where N is the number of pixels on the curve, s is the step of selecting pixels as end points of consecutive lines, d_i is the Euclidean distance between the pixels p_{is} and $p_{(i-1)s}$. There is an error between the Euclidean distance calculated in this way and the true length of the curve. This error depends on the choice of the parameter s. To choose an s that generates a smallest error, we used circle segments with different radii to calculate their theoretical perimeters and the estimated length with different s. We carried out experiments on synthetic digital curves of constant curvature $1/R$, i.e. circle segments of radius R of known length $R\theta$ where θ is the angular sector associated with the segment, with an algorithm previously used in radio astronomy [135]. We computed the mean ratio α between the estimated geodesic length L_{geo} and the true length of the segments, for various radii $R \in [10, 30]$. Table 5.1 summarizes the mean ratio α with different steps. We also investigated this ratio for straight line segments of various angles and of size $S \in [20, 50]$ and we found that $|\alpha - 1| < 0.01$ when $s \geq 2$. These facts led us to choose $s = 4$ as the sampling parameter.



(a) **(b)** **(c)** **(d)**

Figure 5.14: Illustration of length counting of a curve. (a) illustrates the relationship between a curve and its pixel-level visualization. (b), (c) and (d) are their length counting processes with different steps s in Equation 5.7.

Table 5.1: Mean value α of the ratios (computed for radii $R \in [10, 30]$) between the estimated geodesic length L_{geo} and the true length of circular segments.

s	1	2	3	4	5	6	7	8	9
α	1.0551	1.0128	1.0046	0.9999	0.9955	0.9939	0.9907	0.9890	0.9842

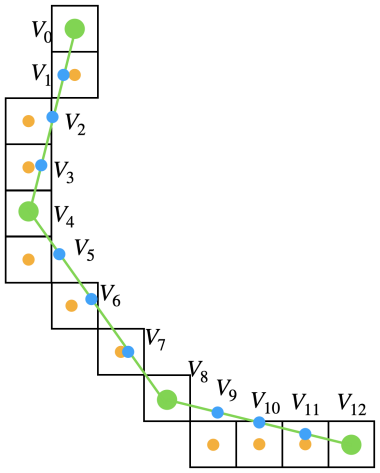


Figure 5.15: Virtual vertices created after choosing $s = 4$ as sampling parameter.

After choosing $s = 4$ for geodesic distance calculation, we chose subsampled vertices to accumulate the distances in between in order to get the Euclidean distance L_{Euc}. We upsampled the selected pixels to have the same number of vertices as the original pixels. We use the case $s = 4$ as an example. When having 2 adjacent sampled vertices p_1 of coordinate (x_1, y_1) and p_2 of coordinate (x_2, y_2), we insert $(x_1 + \frac{k(x_2-x_1)}{4}, y_1 + \frac{k(y_2-y_1)}{4})$ between p_1 and p_2, with $k = 1, 2, 3$, as shown in Figure 5.15. The number of vertices was then augmented to that of the original pixels. Each of them has a cumulative distance to the start point. This cumulative distance was used for choosing vertices of the fitted polygon.

Curves are of different lengths. We classified them into 3 types according to their geodesic lengths:

$$curve\ type = \begin{cases} short, & L_{geo} \leq 100 \\ medium, & 100 \leq L < 300 \\ long, & L > 300 \end{cases} \tag{5.8}$$

For short curves, we selected its two end points, as the start and the end of the polygon. For medium-length curves, we select its two end points and the upsamples vertices having a cumulative distance closest to $L_{geo}/2$. For long curves, we chose one vertex every 100 pixels of cumulative distance to form the polygon. Each case is illustrated on Figure 5.16. The sum of the lengths of the selected upsampled vertices was then the L_{Euc} of the curve.

Fragment images may have branches since elastic lamellae may have contact with each other. In the length counting process, curves have been defined with a

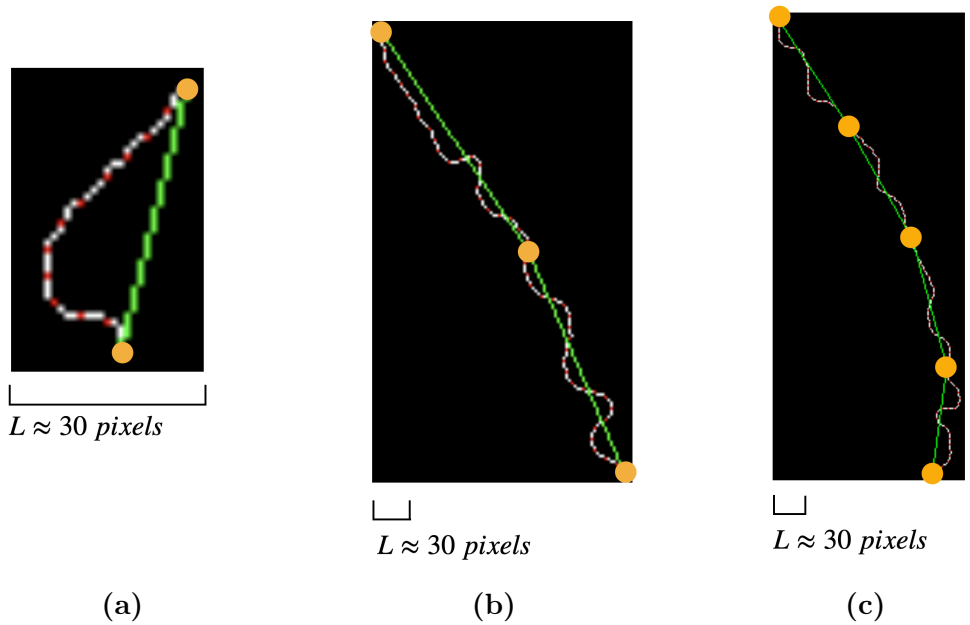


Figure 5.16: Three types of curve (in white) with the sampled points to calculate L_{Euc} (in red) and the polygons (in green) as well as their vertices (in yellow) ((a) short curve; (b) medium-length curve; (c) long curve).

start point and an end point, the start point was chosen randomly from the end points of the curves. A path from this point was found pixel by pixel. When there is a cross point where different branches meet, there is more than one choice of the next pixel of the path. If we chose the next one randomly, then a wrong path with a bad end point can be generated, as illustrated in Figure 5.17. To deal with this issue, before calculating L_{geo}, we detected the existence of cross points in each image and eliminated such points as well as their neighbor pixels to split the branches. After this operation, separated curves can be calculated with the described method independently.

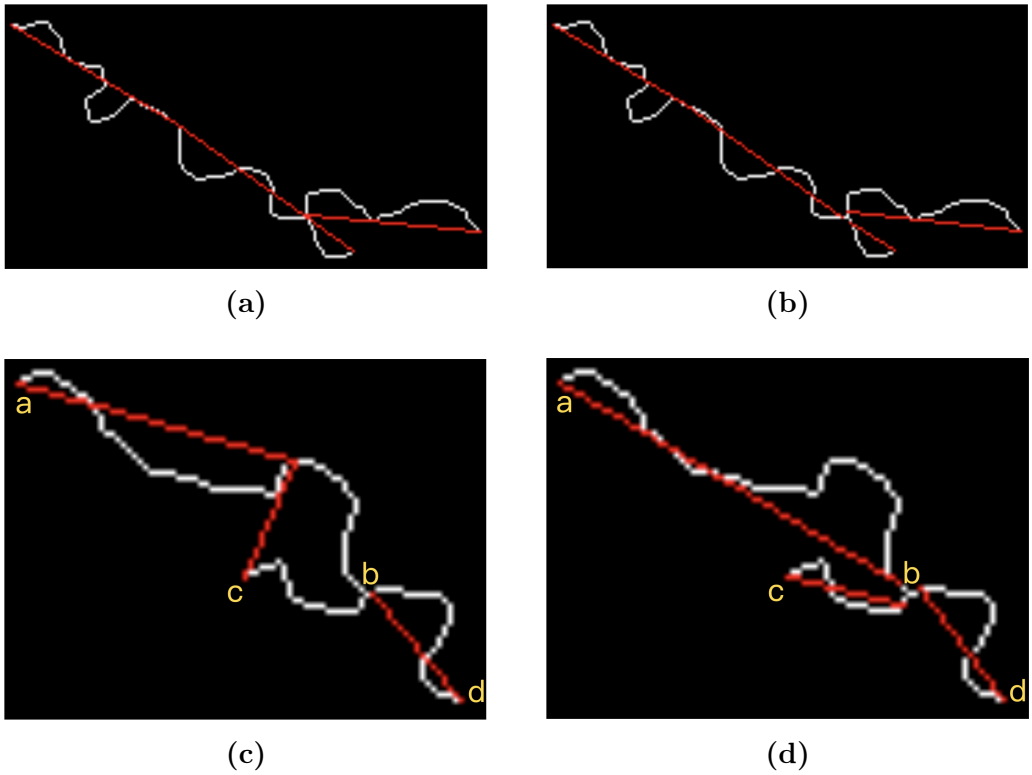


Figure 5.17: Two images with cross points. (a) and (b) show the fitted polygons without and with a separation of branches. The right path happens to be chosen in (a). (c) and (d) show another case; in (c), without the cutting operation, two different branches a-b and b-c are considered to be in the same elastic lamella, as a path a-b-c.

5.3.3 Results

Dataset 1

The first waviness analysis was implemented before finishing the implementation of the segmentation methods in Chapter 4. So our first analysis was performed on a dataset generated by a method based on Otsu's thresholding method with a selection of fragments. The binary results were skeletonized and processed as described in Section 5.3.1 in order to generate 1-pixel-width digital objects.

We applied this method on 21 3D stacks that were divided into two groups: 11 stacks of diabetic mice and 10 stacks of control mice, at the age of 6 to 6.5 months. 2000 to 3000 fragments were generated by each stack. The results of the ratio ρ are illustrated in Table 5.2, with the part that the stack belongs to marked (abdominal or thoracic). Figure 5.18 shows their box plots. For the chosen parameters (in particular, s = 4), the p-value of Mann-Whitney U test is $0.0035 < 0.05$. The

difference between the mean values $\bar{\rho}_{diabetes} - \bar{\rho}_{control}$ is equal to 0.0662. From these results, we can conclude that the elastic lamellae in diabetic mice are globally flatter than those in control (i.e. healthy) mice.

Table 5.2: First results of the waviness measurement. p-value$_1$ are separately calculated on thoracic and abdominal samples, p-value$_2$ is calculated on all the 21 samples.

Normal	Thoracic (6 samples)	Abdominal (4 samples)
ρ	0.8929 0.8993 0.7578 0.8324 0.8557 0.7968	0.7701 0.8379 0.8380 0.8303
Diabetic	Thoracic (6 samples)	Abdominal (5 samples)
ρ	0.8815 0.8769 0.8864 0.9301 0.9268 0.9236	0.8965 0.9303 0.8621 0.8804 0.8749
p-value$_1$ (normal vs. diabetic)	Thoracic: 0.06555	Abdominal: 0.01996
p-value$_2$ (normal vs. diabetic)	0.003474	

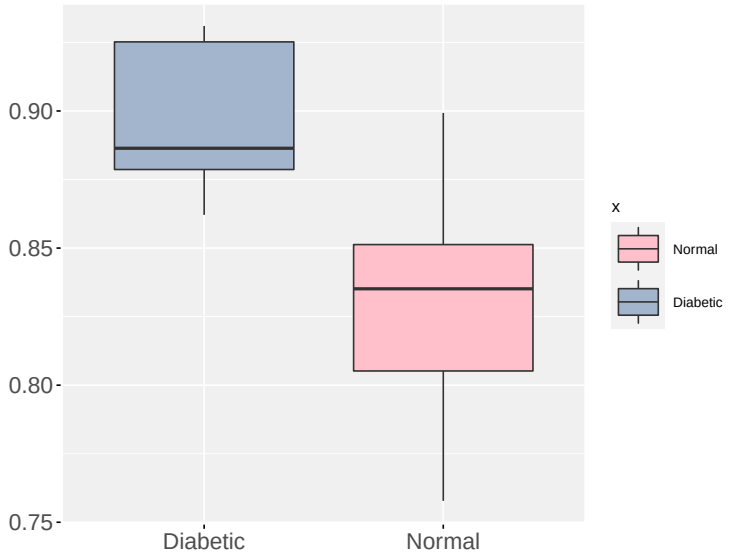
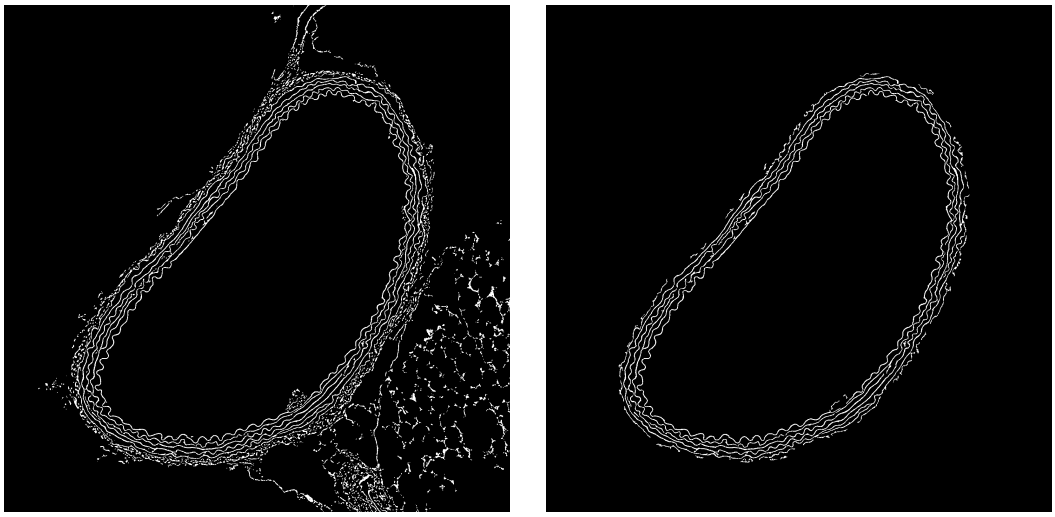


Figure 5.18: Box plots of wavinesses of two groups in dataset 1, with 10 normal samples and 11 diabetic samples.

Dataset 2

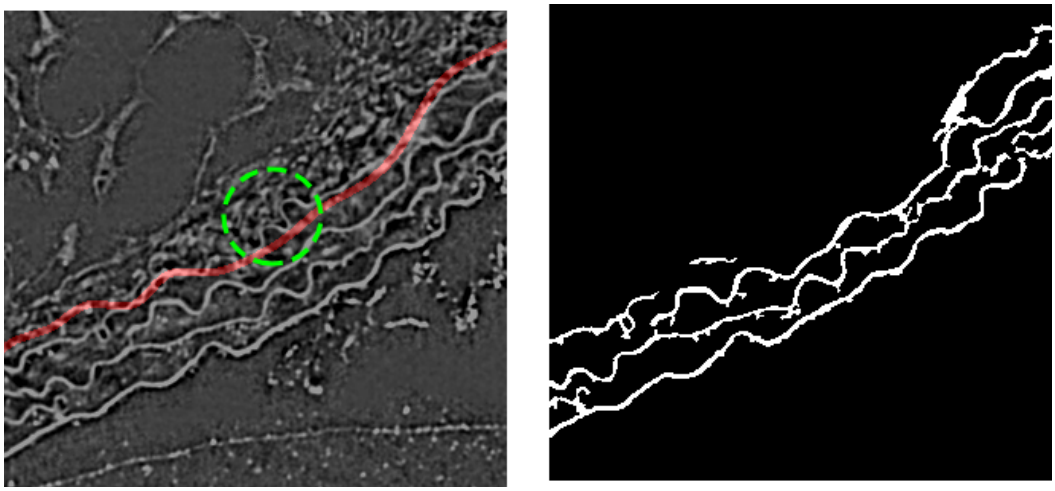
We applied the preprocessing process in an iterative way and the same analysis process to the whole images of elastic lamellae generated with the help of the

extraction of the first lamella and the *media-adventitia* border. Figure 5.19b shows an image of the elastic lamellae. The original image I_{ori} is the binary image obtained from the last result in Chapter 2, the largest connected component of every 32 frames. We denote the binary image mainly containing the *media* as I_{media}. The inner contour of I_{media} is found by eroding the mask of the space enclosed by the first lamella (Section 4.2) and extracting its contour. The outer contour of I_{media} is the approximated spline of the *media-adventitia* border (Chapter 4). We preserved the connected components larger than 50 pixels and skeletonized such images. The *media-adventitia* border that we found were splines crossing the outermost lamellae (Figure 5.20). This lamella is usually connected to *tunica adventitia* textures; so our waviness analysis was performed on almost all the lamellae except for the outermost one and a part of the outermost one that is inside the spline. For each stack, one image was adopted every 256 frames that were validated (frames without abnormal structures).



(a) **(b)**

Figure 5.19: A frame of the largest connected component of 32 consecutive frames (a), as mentioned in Chapter 2, and the same frame of the extracted *tunica media* with the help of the extraction of the first lamella and the *media-adventitia* border (b), with a selection of component size. Images are cropped to focus on the arterial wall part.



(a) (b)

Figure 5.20: A part of a grayscale image superimposed by the spline (a) and the same part of the binary image with the part inside the *tunica-media* border preserved (b). The spline crosses the outermost lamella. Thus after the extraction of the *media*, a part of this lamella is eliminated, as marked by the green circle in (a).

Table 5.3: Second results of the waviness measurement, with stack information marked. T: thoracic; A: abdominal.

Normal (T)	Normal (A)	Diabetic (T)	Diabetic (A)
#0, 6.5 months $\rho = 0.7854$	#2, 6.5 months $\rho = 0.7949$	#6, 6 months $\rho = 0.8919$	#8, 6 months $\rho = 0.9290$
#1, 6.5 months $\rho = 0.7910$	#33, 6.5 months $\rho = 0.8256$	#7, 6 months $\rho = 0.8558$	#11, 6 months $\rho = 0.9467$
#30, 6.5 months $\rho = 0.8542$	#35, 6.5 months $\rho = 0.7984$	#9, 6 months $\rho = 0.9582$	#12, 6 months $\rho = 0.9416$
#31, 6.5 months $\rho = 0.8783$	#37, **3 months** $\rho = 0.8728$	#10, 6 months $\rho = 0.9564$	#14, 6 months $\rho = 0.8044$
#32, 6.5 months $\rho = 0.8563$	#38, **12 months** $\rho = 0.8674$	#13, 6 months $\rho = 0.8424$	#17, 6 months $\rho = 0.7963$
#36, 6.5 months $\rho = 0.8025$	#39, **12 months** $\rho = 0.8459$	#15, 6 months $\rho = 0.8466$	#18, 6 months $\rho = 0.8600$
#40, **12 months** $\rho = 0.8725$	#44, 8 months $\rho = 0.9266$	#16, 6 months $\rho = 0.8169$	#21, 6 months $\rho = 0.8823$

#41, **12 months** $\rho = 0.8803$	#45, 8 months $\rho = 0.8484$	#19, 6 months $\rho = 0.9031$	#23, 6 months $\rho = 0.8795$
#42, 8 months $\rho = 0.9025$	#47, 8 months $\rho = 0.8654$	#20, 6 months $\rho = 0.8935$	#27, 6 months $\rho = 0.9191$
#43, 8 months $\rho = 0.8926$		#22, 6 months $\rho = 0.9084$	#54, 6 months $\rho = 0.9032$
#46, 8 months $\rho = 0.8840$		#24, 6 months $\rho = 0.9249$	
#48, 8 months $\rho = 0.8490$		#25, 6 months $\rho = 0.8985$	
#49, 8 months $\rho = 0.8606$		#26, 6 months $\rho = 0.9062$	
#50, 8 months $\rho = 0.8711$		#28, 6 months $\rho = 0.9285$	
		#29, 6 months $\rho = 0.9180$	
		#51, 6 months $\rho = 0.9485$	
		#52, 6 months $\rho = 0.9554$	
		#53, 6 months $\rho = 0.9444$	
$\bar{\rho} = 0.8523$	$\bar{\rho} = 0.8432$	$\bar{\rho} = 0.9054$	$\bar{\rho} = 0.8862$
$\bar{\rho} = 0.8493$		$\bar{\rho} = 0.8986$	

We calculated the wavinesses on 51 of the 55 samples (3 $ApoE^{-/-}$ samples and 1 normal sample that failed to generate a classification model in the training process were not used). 23 are normal samples and 28 diabetic samples. Their wavinesses are shown in Table 5.3, divided into four groups. The averages in the last two lines were calculated without the five samples of 3 months and 12 months (marked in red and blue). The statistics were performed on the remaining 46 samples, 18 normal and 28 diabetics. Figure 5.21 shows the box plots of each column (except for the 5 marked groups in Table 5.3) and the combination of two columns of diabetic and normal stacks.

The samples from 6 to 8 months were considered as in the same age range. We compared their wavinesses without considering their age difference. The average

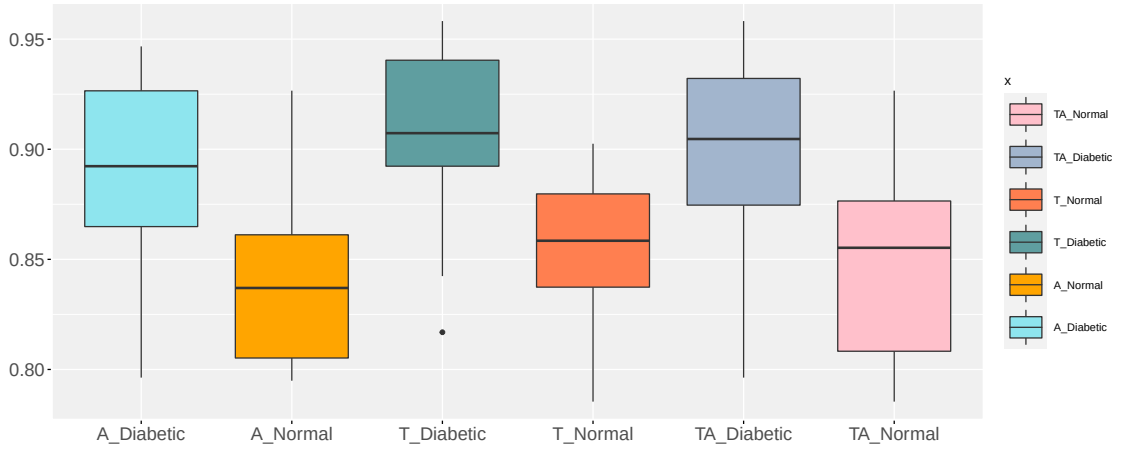


Figure 5.21: Box plots of wavinesses of two groups in dataset 2. A: abdominal; T: thoracic; TA: thoracic and abdominal.

waviness of diabetic samples is greater than that of normal samples, both in the thoracic and the abdominal parts. The *p-value* of diabetic and normal samples in the abdominal part is 0.1158, that in the thoracic part is 0.00325, and that in the two parts is 0.000976, which shows a significant difference between these two groups. This difference is mainly from the thoracic samples. One of the reasons is that the number of abdominal samples is lower than thoracic samples, and this number may be not enough to reveal the difference between the two types of samples. On the other hand, it may be due to the difference in the impacts of pressure on the thoracic part and the abdominal part. The thoracic segment is more exposed to pressure than the abdominal part. Thoracic part is mechanically much more stressed than the abdominal part because of its proximity to the heart.

We observe that the four stacks of 12 months have greater wavinesses than the average of their corresponding columns, but still lower than the average of diabetic samples. The sample of 3 months has a higher waviness than the two 12-month-old samples. The assumption we made before obtaining these results was that aging has similar impacts on arteries as pathologies. Since these samples are not enough to verify this assumption, we still need further analysis with more samples at very young and very old ages to reduce the error due to the variabilities between individuals.

If we combine this waviness analysis and that of Section 5.2, it can be observed that with similar wavinesses, a diabetic sample and a normal sample can show different shapes in curvature histograms of the first lamella. Based on this fact, we may make an assumption that the impact of pathologies can be manifested in different aspects. The waviness change shows its impact of smoothing the elastic lamellae, and the curvature distribution shows its impact of changing the way that

the lamellae curl, which is probably related to the way the vessel reacts to forces.

5.4 Conclusion

In this chapter, we described two analyses of the segmented elastic lamellae. The first analysis was based on the extraction of the first lamella as shown in Section 4.2 of Chapter 4. 3D surfaces of the outer border of the first lamellae were built, 3D curvatures were calculated with spheres centered at the voxels on the surfaces. 18 curvature histograms were generated and analyzed. In general, besides a peak on the negative side, those of diabetic samples show an obvious second height at a positive value, which is not the same in most cases of the normal samples. Thoracic and abdominal samples do not show different features. The second analysis was performed on fragments, based on the result of Section 4.3. The texture classification helped us find the outer border of the *tunica media*. Thus we could extract the pixels inside this border and skeletonize them into curves of thickness one. A waviness measurement was proposed to estimate the ratio between the curve length and the approximated polygon length, which reveals the level that the lamellae are straightened. We applied this analysis twice. One was performed on fragments segmented in a relatively simple way based on a thresholding method. The second was performed on more stacks obtained with the segmentation method proposed in this work. Both results show a significant difference in waviness between the normal and diabetic samples: the elastic lamellae in the diabetic mice are more straightened than in the healthy mice. We combined these two analyses and found that samples having similar wavinesses show noticeable differences in curvature histogram shapes. This observation led us to make an assumption of decomposing the impact of pathologies into different aspects including straightening the lamellae and changing the way they curl.

Chapter 6

Conclusion

6.1 Conclusion of the works

Cardiovascular diseases is a leading cause of death in the world. An intervention of such diseases before they get severe is thus important. These diseases are highly related to metabolic disorders such as diabetes. In our work, arterial images of healthy and diabetic mice were obtained by synchrotron X-ray microtomography with a size 2048^3 voxels of each stack, which enabled us to visualize very fine structures of the mice arterial wall. The work can be divided into two main parts: segmentation (Chapters 2, 3, 4) and analysis (Chapter 5). The segmentation was realized by first preprocessing the stacks and then extracting the general shape of the lumen in order to perform a finer segmentation on unfolded images in the following steps. We implemented two segmentation processes: extraction of the first elastic lamella and classification of the *tunica media* and the *tunica adventitia* textures. These two steps correspond to the extraction of the inner border and the outer border of the *tunica media*, where lie the elastic lamellae, the important structure that alters during aging and the development of diseases. The first segmentation method is based on region growing, with a space pruning process that removes the falsely grown regions caused by the leakages due to the holes or low-intensity points on the lamellae. The second segmentation is based on deep learning, aiming to perform the texture classification without a manual labeling process. With the help of the extraction of the first lamella, the arterial wall can be unfolded more accurately, with the *tunica media* starting from the top rows and the *tunica adventitia* placed below it. The positional information of these two types of texture can be then used for labeling the similarities between subpatches. Having similarities between subpatches, we trained one Siamese network for each stack. The pixel-level segmentation was

realized by comparing the descriptors generated by the subpatch surrounding the pixel to predict and those generated by the subpatch above and below it. The results of this pixel-level prediction were discontinuous pixels between the two textures. We selected the horizontally equidistant points from them and generated closing splines to approximate the continuous outer border of the *tunica adventitia*. Figure 6.1 illustrates the segmentation steps, of which the results correspond to the two types of analyses. The analyses were performed on 3D and 2D images. The 3D curvatures of the first elastic lamellae from different stacks were visualized in a statistical way by generating their histograms. Histograms generated by diabetic samples show a second height at a positive value, compared to those of normal samples, which mostly have one peak at a negative value. The 2D waviness was calculated on two datasets of 2D fragments. The first dataset was generated by a thresholding method with a selection of fragments. These fragments were generated from entire 2D images. The second dataset was generated by our proposed segmentation method, with the inner and outer borders of the *media* found. The two datasets were post-processed and analyzed the same way, with a waviness measure estimating the ratio between the approximated open polygon length and the curve length. Both datasets show the fact that the elastic lamellae in the arterial wall of diabetic mice are flatter/smoother than those of healthy mice. The 2D and 3D measurements could be indicators of the functional age of the arterial system rather than its chronological age.

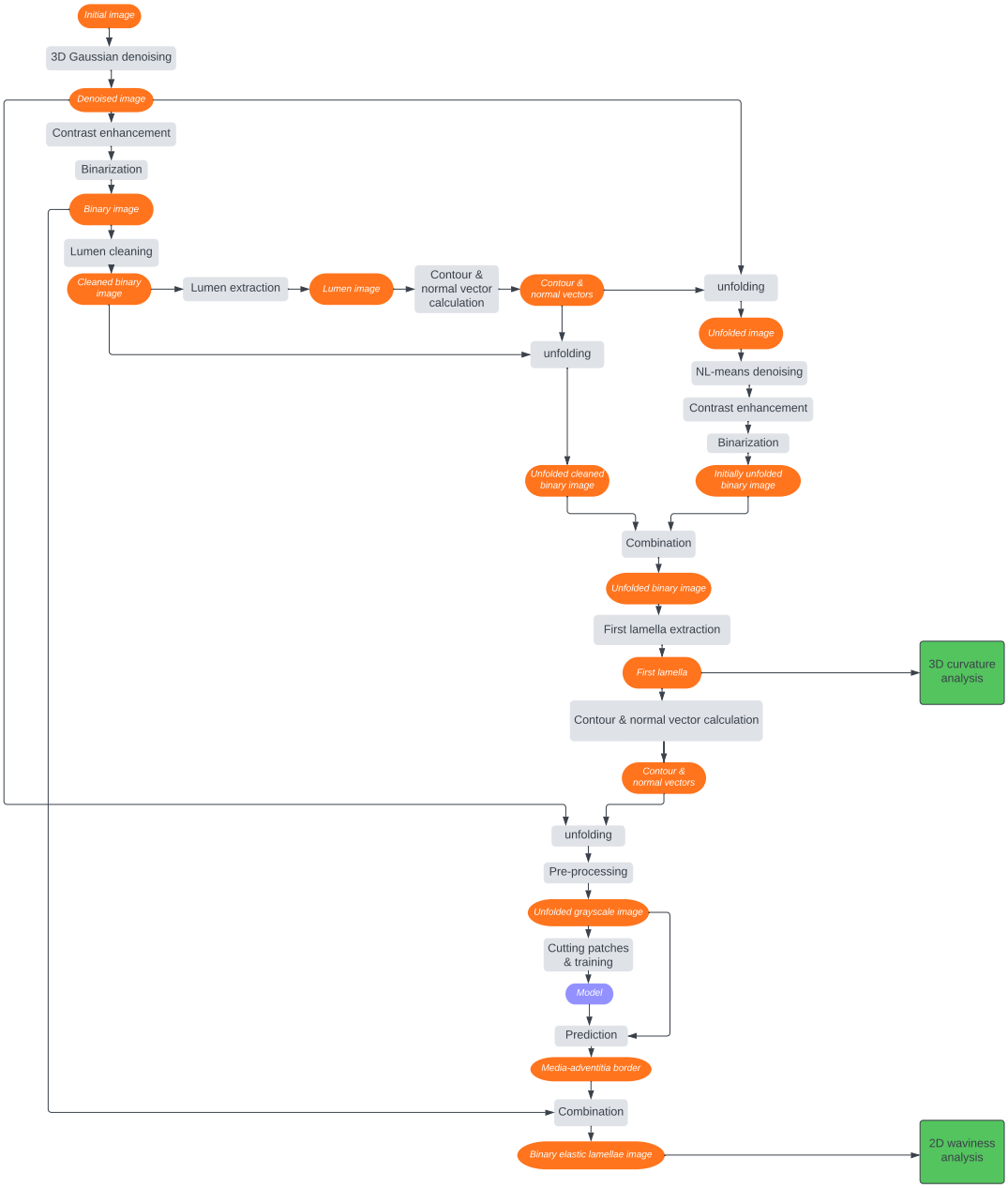


Figure 6.1: Diagram of segmentation steps, of which the results correspond to two types of analyses.

The segmentation pipeline was designed with constantly revising previous steps. The images we obtained are various in shape and components as summarized in Appendix A. In addition, only when we got the results of step i and had a concrete perception of that could we consider the appropriate operation of step $i + 1$. After

designing and testing the result of step $i+1$, problems of step i were exposed and that enabled us to go back and make improvements. Thus, the further step we were, the more steps we needed to backtrack when a problem was found, and these problems were usually not predictable before getting the results of further steps. For example, before designing the unfolding process of arterial walls, we could not know what kind of denoising method would be appropriate for getting the final elastic lamellae segmentation result; before the extraction of the first elastic lamella by the region growing-based method, we did not know how far we needed to clean the impurities and how precise should the binarisation on the whole image achieve. During the whole process of designing every step $i + 1$ of the segmentation pipeline, we first obtained the results of step i without modifying this step until it seemed "perfect" because the evaluation of these results depended highly on how it would work later. The idea of the texture classification came before that of the segmentation of the first elastic lamella and the 3D lumen extraction. The first implementation of the texture classification was realized with the lumen contour extraction based on relatively simple 2D image processing methods including GLCM and classic region growing. The first prediction of the *media-adventitia* border was performed on rotated patches without unfolding the whole arterial wall pixel by pixel. The result was a stitched image of the results on patches. This first implementation was realized based on the results that were relatively less precise and only for "regular" images. After this implementation, we got a more global view of the pipeline and that enabled us to develop and improve the previous steps that should serve the final results.

6.2 Discussion & perspectives

The segmentation pipeline is based on general processing methods with possible human interventions after each step. The general processing methods were built upon the common features of the stacks. Specific features that exist only in a part of the stacks were dealt with by human interventions. The operations start from preprocessing by a basic contrast enhancement method but with manual parameter adjustments on some of the stacks. Human intervention is also needed in the lumen cleaning and lumen extraction process. After observing the result of the lumen cleaning, we could know if there exist impurities that will interrupt the lumen extraction and lamella extraction processes. The continuity between consecutive frames can then be used to eliminate more impurities. The range of the frames can also be a constraint to cleaning the impurities in a specific segment. For stacks that contain

air bubbles, another processing is needed. The initialization of the lumen extraction concerns a localization of the lumen on one frame for each stack. Sometimes, the extracted lumen in the following step contains more than one component. This also requires human intervention to eliminate unwanted components with the help of frame continuity. The unfolded binary process contains an extraction of the largest 3D connected component. When there are bifurcations, the number of the selected components needs to be augmented, which introduces another human intervention. After the extraction of the first elastic lamella, the generation of the mask needs a process of closing the contour of the lamella, where many parameters are needed and these parameters are not the same for stacks with very different shapes. More adaptive methods in these steps are expected. Besides improving the robustness of each step itself, it is also possible to add a feature-detecting process before each step in order to choose the appropriate methods automatically before the operations.

Numerous stacks contain high-contrast particles on the first elastic lamellae. Sometimes these particles are superimposed on and hide the first lamella; sometimes they touch but do not hide the lamella. A possible improvement in the lamella extraction step is to restore the shape of the lamella part disturbed by these particles (for example, with manually designed filters). This would enable us to realize a more complete analysis of the 3D geometry of the first lamella.

In the proposed pipeline, a deep learning-based method was applied to the texture classification step, which is the second segmentation process. We also tested a deep learning model based on SegNet [136] with a weighted loss function as in an application for extracting the guidewire structure in medical images [137]. We trained two models with the ground truth of the first lamella extracted by the region growing-based method: one for the whole image and one for the unfolded arterial wall part. In the first case, besides the first lamella, a number of perivascular tissues were also extracted, while in the second case, the model was capable of extracting the first lamella. This result inspires us to consider integrating more deep learning steps in the segmentation process. Since a direct extraction of the first lamella from the initial image is more difficult than in the unfolded image due to the structures outside the artery, we could split this into a model for the extraction of the lumen and another model for the extraction of the first lamella on the unfolded image obtained with the extracted lumen.

The extraction method applied to the first elastic lamella can also be used to extract other lamellae layer by layer after the elimination of the already extracted ones. We tested this extraction on several images. The complexity and the difficulty

grow with the number of the layers extracted since the completeness of each layer decreases (some of them have only a part) and the error of a false extraction of the pixels on the layer below will propagate to the next extraction. So, possible improvements could be focused on the post-processing of each extracted layer to limit the error propagated to the next one and a rearrangement of the extracted layers to find the most probable order of them.

The *media-adventitia* border we found crosses the outermost lamella and this layer is in contact with the *adventitia*. Thus its accurate extraction is more difficult than the others. We generated a spline to approximate the border. One possibility to improve the result is to use it as an initialization of an active contour and let the contour deform in a certain range.

In the segmentation pipeline, some methods are performed in 2D while some are in 3D. All the pre-processing steps except for the very first 3D Gaussian filtering are 2D methods. More implementations of filtering methods are expected, especially the 3D ones, for example, 3D NL-means filtering [138] and BM4D filtering [139]. The elastic lamellae in our images have special geometry features. Thus it is also worth exploring methods such as nonlinear diffusion filtering [140] that denoise an image with the *a priori* knowledge taken into account and combined methods such as the NL-means filtering with the diffusion tensor [141]. An important problem in such implementations is the computation cost. With stacks of size 2048^3, optimization strategies will be much needed. The texture classification was implemented on 2D images. We also tried an implementation with a 3D Siamese network and 3D patches. However, the results did not show obvious improvements, and the computation cost augmented much. Nonetheless, such 3D deep learning and texture analysis methods are still worth discovering more about, since there exists much potential textural information in the Z direction.

In the analysis section, we calculated the 3D curvatures with spheres of the same size and counted the number of voxels inside the 3D mask to use as the volume. In the work [18], factors including the sphere radius and the volume counting method were studied to achieve an accurate calculation and a library [142] with such implementations has been developed. However, with the large size of our dataset, the computation is too time-consuming. Thus one of the future works will be implementing a more accurate 3D curvature estimation adapted to our dataset. Besides, we would also analyze the local curvature distributions in each stack, which means comparing the curvatures of fixed segments instead of performing an overall statistic on the whole stack. Such analyses would enable us to figure out the lamellar

curvature features in various regions of an artery and to compare different samples in detail, in order to give biological interpretations.

In the 2D analysis, we proposed a waviness measurement to estimate the folding level of the elastic lamella fragments. This can also be applied to complete lamellae and decompose the waviness into the part due to the lamella itself and the part due to the circular structure of the vessel, as realized in the work of Yu *et al.* [50].

Besides the elastic lamellae, other structures are also affected by aging/pathological processes. For example, perivascular tissue is important for vessel function, and it is worth analyzing its structure.

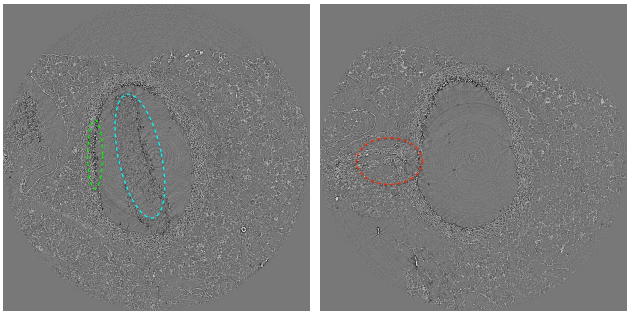
In the image acquisition process, different segments of one artery generated more than one stack. If we stitch these segments into one long segment with a proper alignment method, the more complete structure may give us more information about the sample.

We also acquired another batch of data after the acquisition of those used in this work. This second dataset contains images of larger sizes in X and Y directions since these arteries have greater diameters. So in future works, we will not only improve the methods as discussed above but also adapt them to other data that may contain new features.

Appendix A

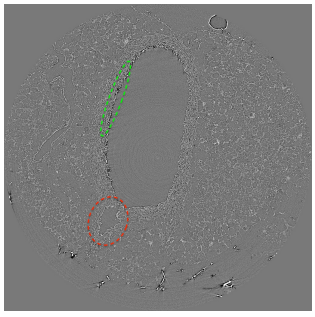
Illustration of raw data

2D frames corresponding to Table 2.2 in every stack are illustrated below, the captions indicate the number of each stack and the subcaptions indicate the types of features in this frame. Features of type 1 (blue), 2 (green), 3 (red), 5 (orange) and 6 (yellow) are marked by circles of different colors. The correspondence of features and type numbers are shown in Table 2.2. Frames that have feature type 4 are directly shown. Frames are illustrated in the order they appear in the corresponding stack. For each stack, features are not repeatedly marked. Images labeled as (a) and (b) may contain common features, these features are not marked again in (b).



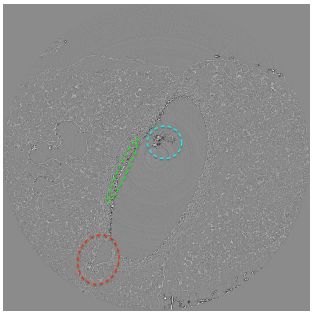
(a) Type 1, 2 **(b)** Type 3

Figure A.1: Stack 0.



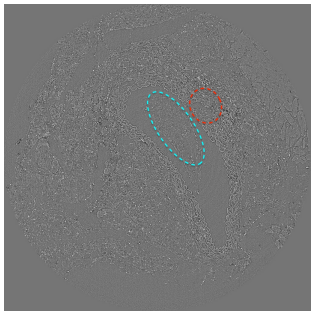
(a) Type 1, 3

Figure A.2: Stack 1.

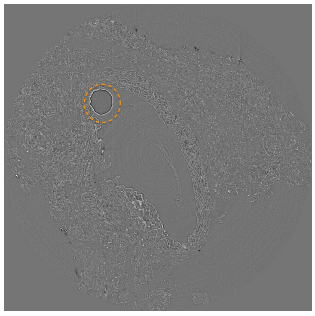


(a) Type 1, 3

Figure A.3: Stack 2.

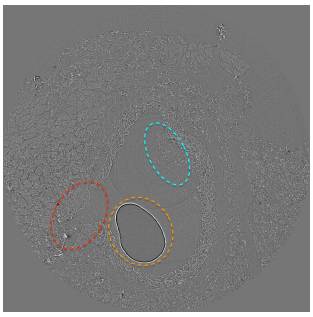


(a) Type 1, 3



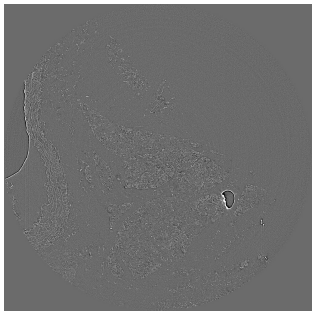
(b) Type 5

Figure A.4: Stack 3.

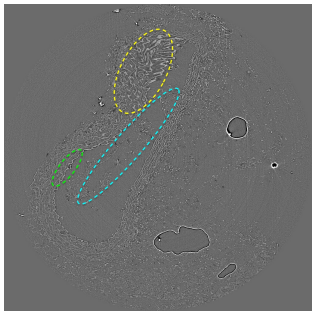


(a) Type 1, 3, 5

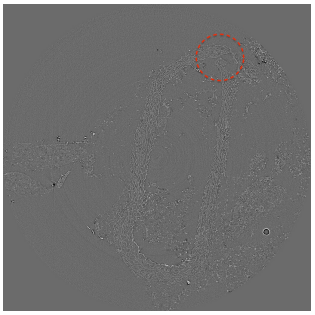
Figure A.5: Stack 4.



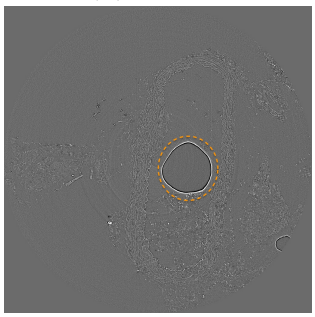
(a) Type 4



(b) Type 1, 2, 6

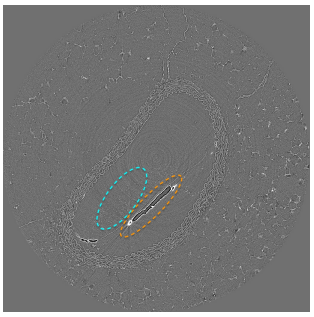


(c) Type 3

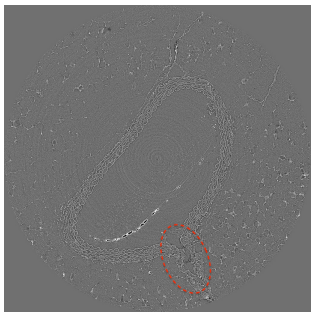


(d) Type 5

Figure A.6: Stack 5.

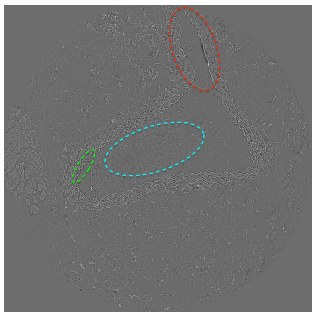


(a) Type 1, 5



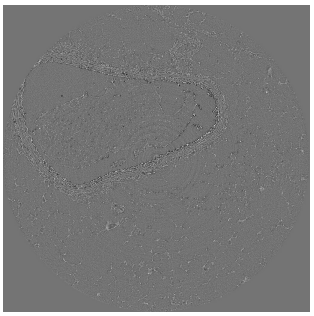
(b) Type 3

Figure A.7: Stack 6.

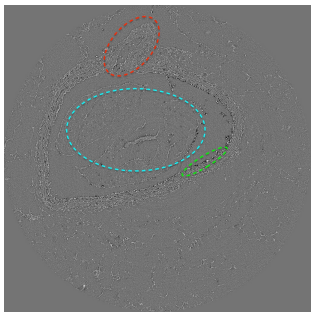


(a) Type 1, 2, 3

Figure A.8: Stack 7.

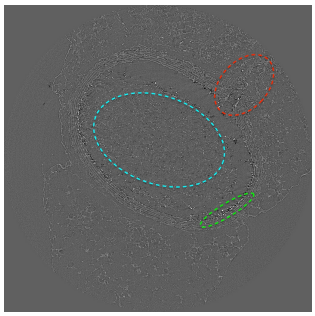


(a) Type 4



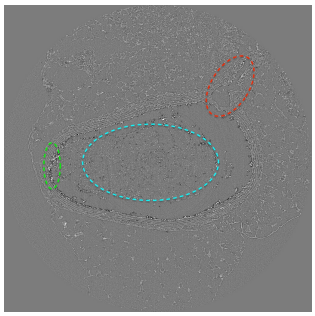
(b) Type 1, 2, 3

Figure A.9: Stack 8.



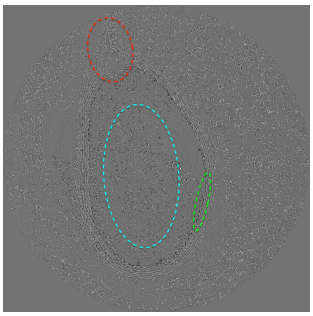
(a) Type 1, 2, 3

Figure A.10: Stack 9.



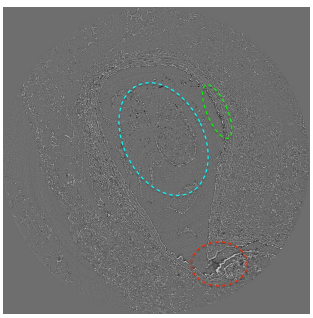
(a) Type 1, 2, 3

Figure A.11: Stack 10.



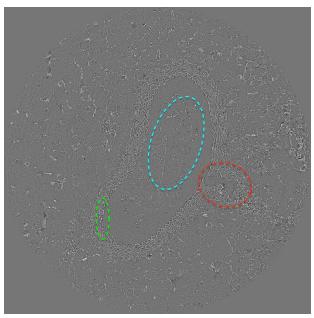
(a) Type 1, 2, 3

Figure A.12: Stack 11.



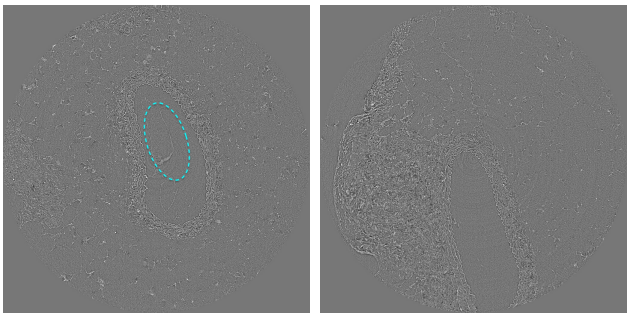
(a) Type 1, 2, 3

Figure A.13: Stack 12.



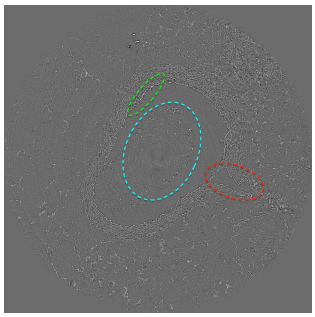
(a) Type 1, 2, 3

Figure A.14: Stack 13.



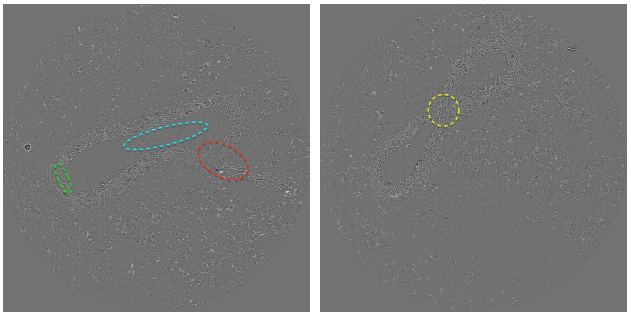
(a) Type 1 **(b)** Type 4

Figure A.15: Stack 14.



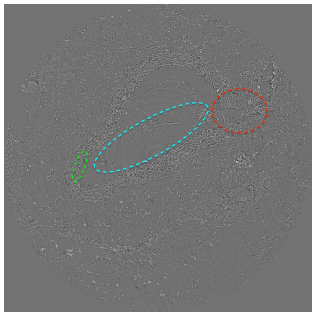
(a) Type 1, 2, 3

Figure A.16: Stack 15.



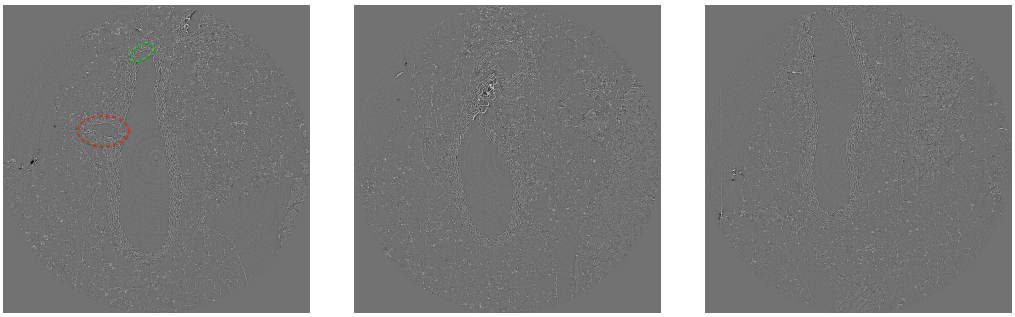
(a) Type 1, 2, 3 **(b)** Type 6

Figure A.17: Stack 16.



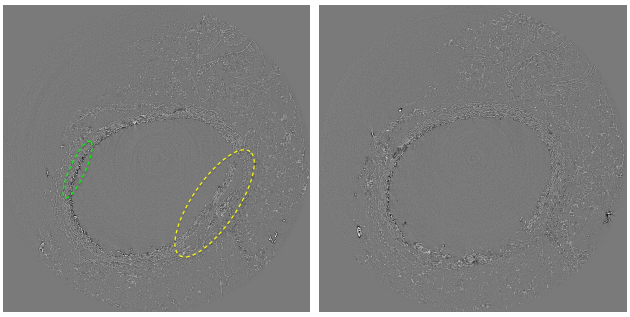
(a) Type 1, 2, 3

Figure A.18: Stack 17.



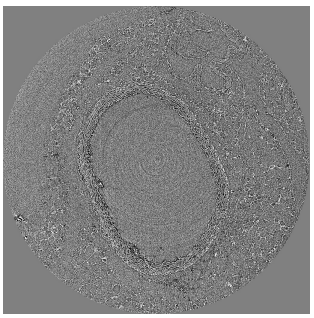
(a) Type 2, 3 **(b)** Type 6 **(c)** Type 4

Figure A.19: Stack 18.

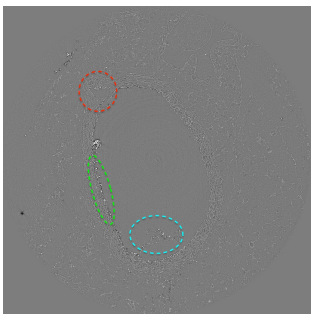


(a) Type 2, 3 **(b)** Type 6

Figure A.20: Stack 19.

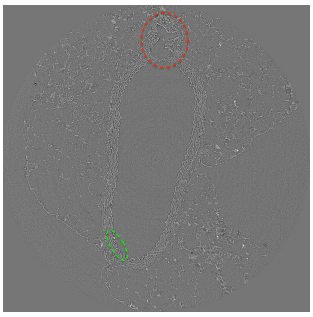


(a) Type 6



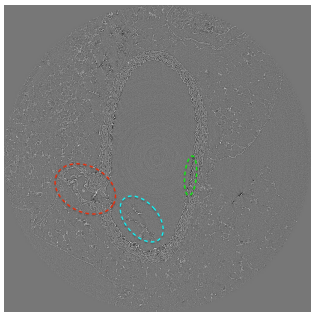
(b) Type 1, 2, 3

Figure A.21: Stack 20.



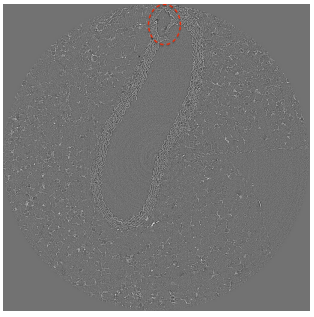
(a) Type 3

Figure A.22: Stack 21.

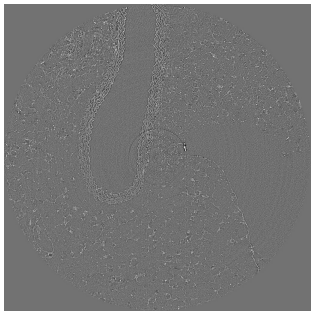


(a) Type 1, 2, 3

Figure A.23: Stack 22.

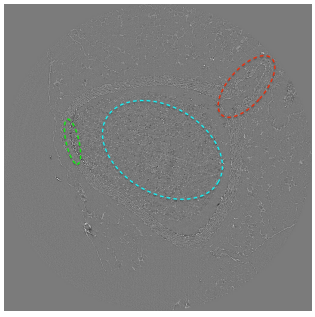


(a) Type 3



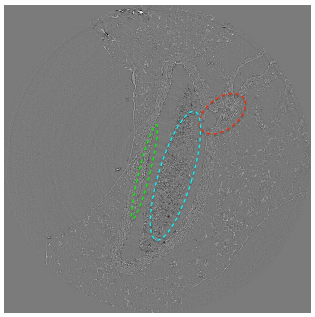
(b) Type 4

Figure A.24: Stack 23.



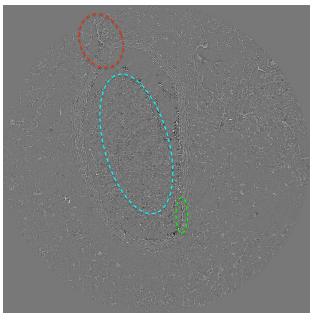
(a) Type 1, 2, 3

Figure A.25: Stack 24.



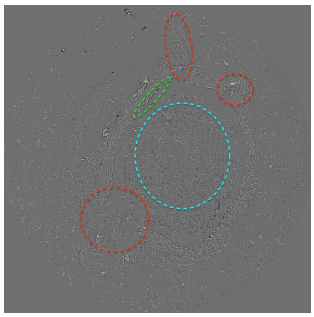
(a) Type 1, 2, 3

Figure A.26: Stack 25.



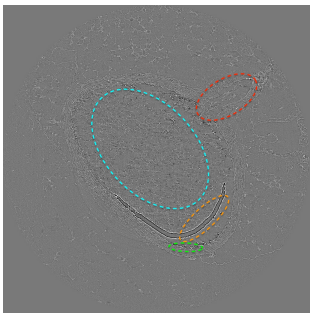
(a) Type 1, 2, 3

Figure A.27: Stack 26.



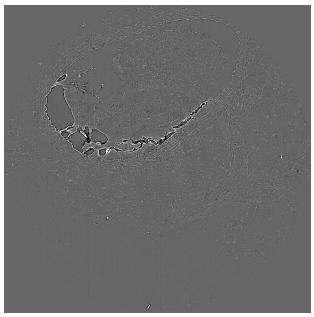
(a) Type 1, 2, 3

Figure A.28: Stack 27.

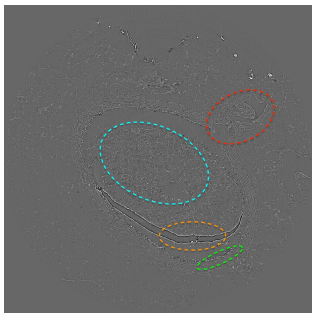


(a) Type 1, 2, 3, 5

Figure A.29: Stack 28.

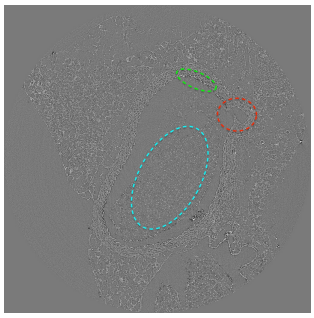


(a) Type 4



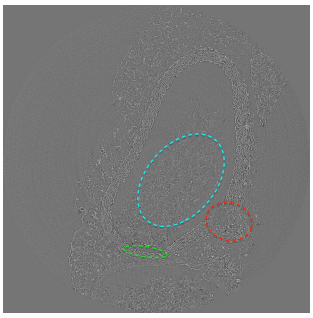
(b) Type 1, 2, 3, 5

Figure A.30: Stack 29.



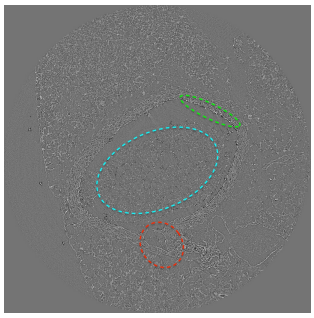
(a) Type 1, 2, 3

Figure A.31: Stack 30.



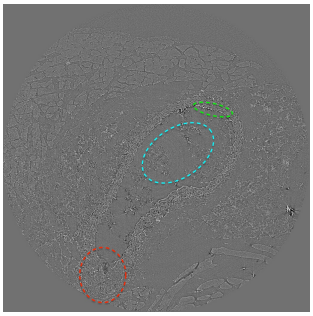
(a) Type 1, 2, 3

Figure A.32: Stack 31.



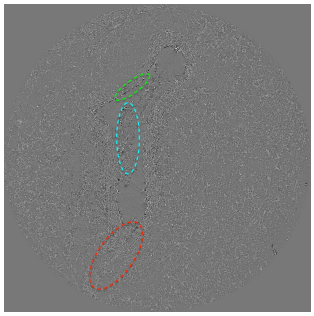
(a) Type 1, 2, 3

Figure A.33: Stack 32.

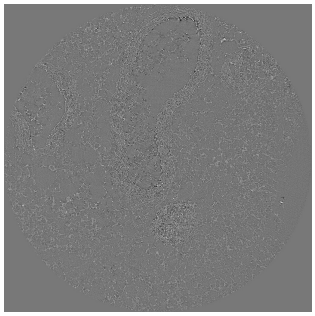


(a) Type 1, 2, 3

Figure A.34: Stack 33.

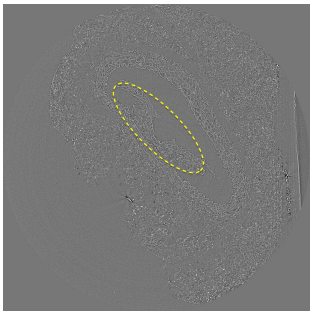


(a) Type 1, 2, 3

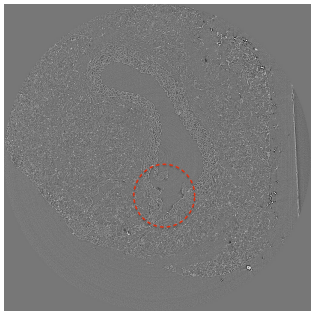


(b) Type 4

Figure A.35: Stack 34.

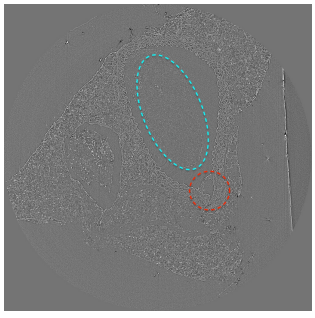


(a) Type 6



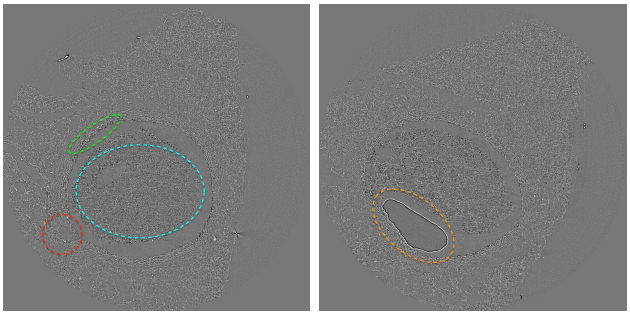
(b) Type 3

Figure A.36: Stack 35.



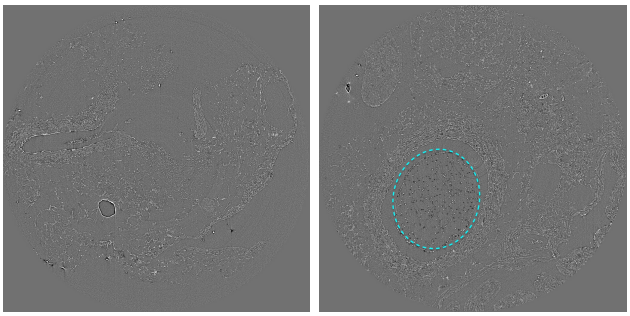
(a) Type 1, 3

Figure A.37: Stack 36.



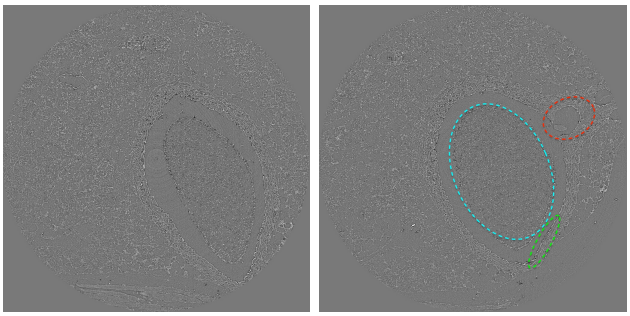
(a) Type 1, 2, 3 **(b)** Type 5

Figure A.38: Stack 37.



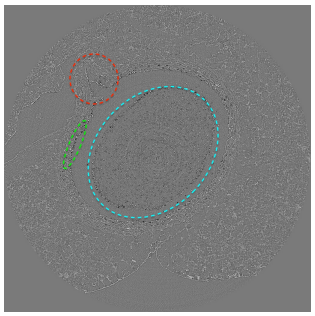
(a) Type 6 **(b)** Type 1

Figure A.39: Stack 38.



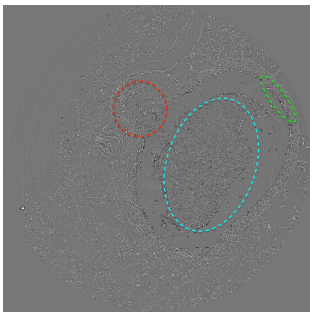
(a) Type 4 **(b)** Type 1, 2, 3

Figure A.40: Stack 39.



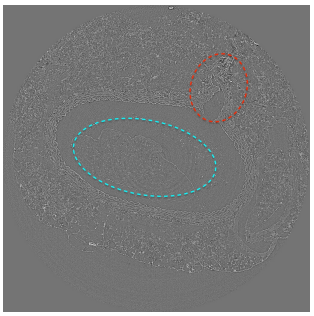
(a) Type 1, 2, 3

Figure A.41: Stack 40.



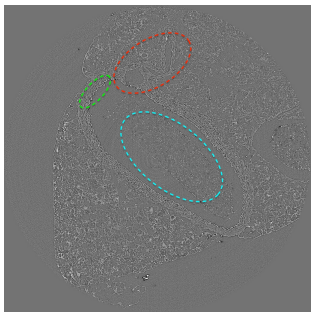
(a) Type 1, 2, 3

Figure A.42: Stack 41.



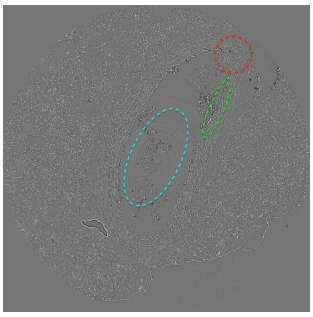
(a) Type 1, 3

Figure A.43: Stack 42.



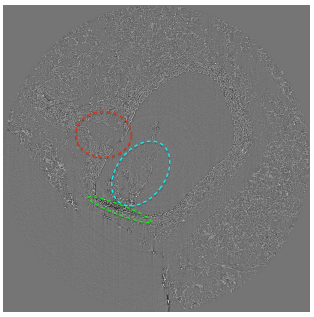
(a) Type 1, 2, 3

Figure A.44: Stack 43.



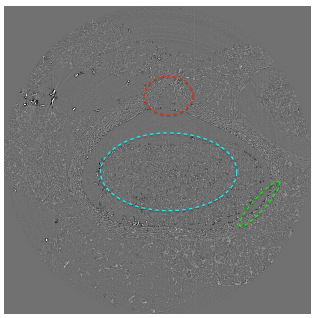
(a) Type 1, 2, 3

Figure A.45: Stack 44.



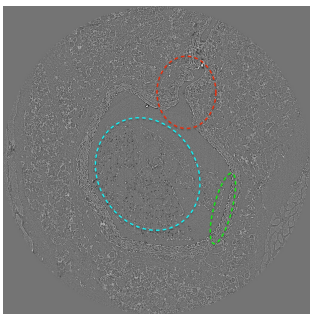
(a) Type 1, 2, 3

Figure A.46: Stack 45.



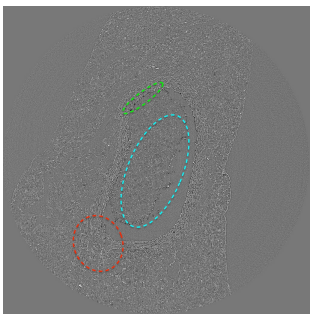
(a) Type 1, 2, 3

Figure A.47: Stack 46.



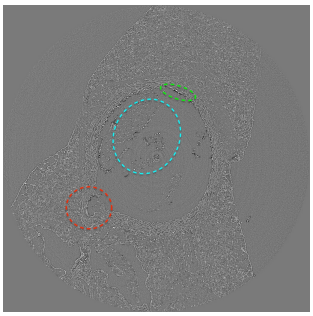
(a) Type 1, 2, 3

Figure A.48: Stack 47.



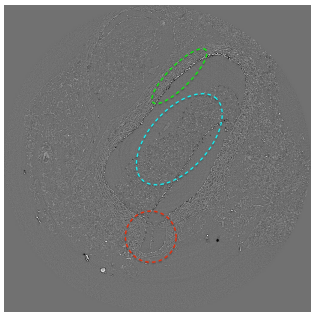
(a) Type 1, 2, 3

Figure A.49: Stack 48.

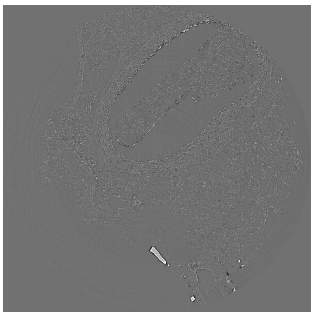


(a) Type 1, 2, 3

Figure A.50: Stack 49.

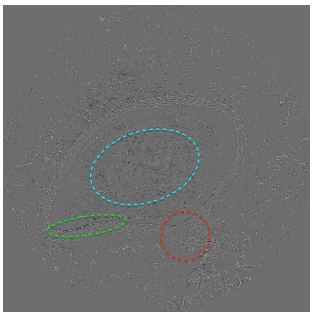


(a) Type 1, 2, 3

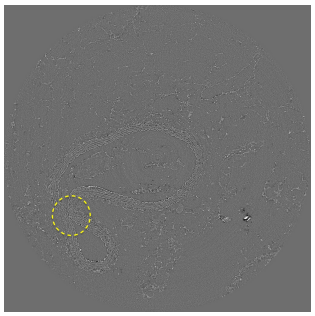


(b) Type 4

Figure A.51: Stack 50.

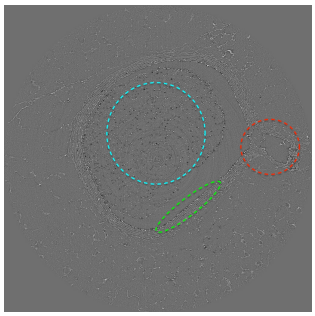


(a) Type 1, 2, 3



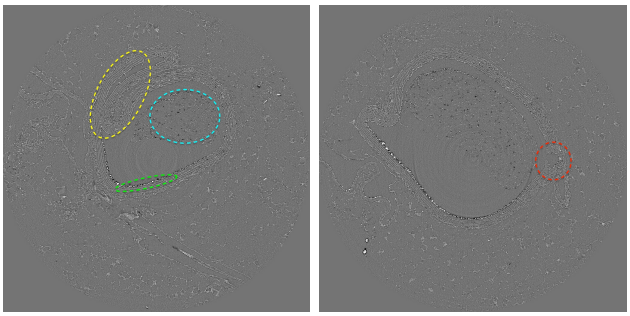
(b) Type 6

Figure A.52: Stack 51.



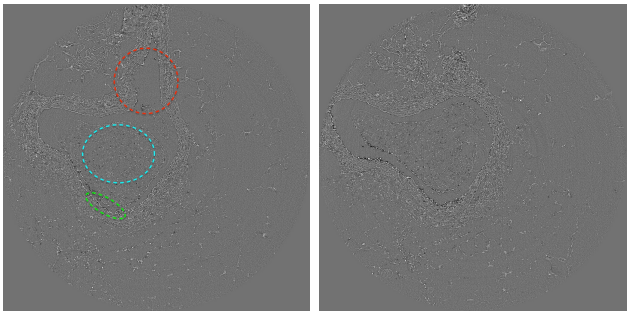
(a) Type 1, 2, 3

Figure A.53: Stack 52.



(a) Type 1, 2, 6 **(b)** Type 3

Figure A.54: Stack 53.



(a) Type 1, 2, 3 **(b)** Type 6

Figure A.55: Stack 54.

Appendix B

Publications & communications

International journal

Ben Zemzem, A., Liang, X., Vanalderwiert, L., Bour, C., Romier-Crouzet, B., Blaise, S., Sherratt, M. J., Weitkamp, T., Dauchez, M., Baud, S., Passat, N., Debelle, L., Almagro, S. (2022). Early alterations of intra-mural elastic lamellae revealed by synchrotron X-ray micro-CT exploration of diabetic aortas. International Journal of Molecular Sciences, 23(6), 3250.

International conferences with proceedings

Liang, X., Ben Zemzem, A., Almagro, S., Weitkamp, T., Debelle, L., Passat, N. (2022). Analysis of diabetes alteration of the mouse aorta from synchrotron micro-CT imaging. In Computational and Mathematical Biomedical Engineering (CMBE), pp. 609-612.

Liang, X., Ben Zemzem, A., Almagro, S., Boisson, J. C., Steffenel, L. A., Weitkamp, T., Debelle, L., Passat, N. (2022). Mouse arterial wall imaging and analysis from synchrotron X-ray microtomography. In 2022 IEEE International Conference on Image Processing (ICIP), pp. 1731-1735.

National conferences with proceedings

Liang, X., Ben Zemzem, A.. Imagerie et analyse de la paroi artérielle de souris à partir de la microtomographie par rayons X synchrotron. 19ème journées franco-

phones des jeunes chercheurs en vision par ordinateur (ORASIS 2023), May 2023, Carqueiranne, France.

Liang, X., Ben Zemzem, A., Vanalderwiert, L., Romier-Crouzet, B., Blaise, S., Weitkamp, T., Dauchez, M., Baud, S., Debelle, L., Passat, N., Almagro, S.. Exploring damaged elastic arteries by Synchrotron X-ray Micro-CT. Congrès de la Société Française de Biomécanique, 2023, Grenoble, France.

National conferences

Liang, X., Ben Zemzem, A., Almagro, S., Boisson, J. C., Steffenel, L. A., Weitkamp, T., Debelle, L., Passat, N.. Mouse arterial wall imaging and analysis from synchrotron X-ray microtomography. SOLEIL Users' Meeting (SUM), Jan 2023, Saint-Aubin, France.

Liang, X., Ben Zemzem, A., Almagro, S., Boisson, J. C., Steffenel, L. A., Weitkamp, T., Debelle, L., Passat, N.. Mouse arterial wall analysis from synchrotron micro-CT imaging. Journée ISS (International Society for Stereology) France, Feb 2023, Paris, France.

Liang, X., Ben Zemzem, A., Almagro, S., Boisson, J. C., Steffenel, L. A., Weitkamp, T., Debelle, L., Passat, N.. Mouse arterial wall imaging and analysis from synchrotron X-ray microtomography. Journée Scientifique autour de l'IA (JSIA), Feb 2023, Nancy, France.

Liang, X., Ben Zemzem, A., Almagro, S., Boisson, J. C., Steffenel, L. A., Weitkamp, T., Debelle, L., Passat, N.. Imagerie et analyse des parois d'artères par microtomographie aux rayons X synchrotron. Colloque Français d'Intelligence Artificielle en Imagerie Biomédicale (IABM), Mar 2023, Paris, France.

Local conference

Liang, X., Ben Zemzem, A., Almagro, S., Boisson, J. C., Steffenel, L. A., Weitkamp, T., Debelle, L., Passat, N.. Imagerie et analyse des parois d'artères avec la microtomographie aux rayons X synchrotron. Journée Rémoise des Jeunes Chercheurs en Santé (JRJCS), Oct 2022, Reims, France.

Bibliography

[1] M. F. O'Rourke, M. E. Safar, and V. Dzau. The cardiovascular continuum extended: aging effects on the aorta and microvasculature. *Vascular Medicine*, 15(6):461–468, 2010.

[2] H. L. Birch. *Extracellular Matrix and Ageing*, pages 169–190. Springer Singapore, Singapore, 2018.

[3] L. Duca, S. Blaise, B. Romier, M. Laffargue, S. Gayral, H. El Btaouri, C. Kawecki, A. Guillot, L. Martiny, L. Debelle, and P. Maurice. Matrix ageing and vascular impacts: focus on elastin fragmentation. *Cardiovascular research*, 110(3):298–308, 2016.

[4] Blausen.com staff (2014). "Medical gallery of Blausen Medical 2014". WikiJournal of Medicine 1 (2). DOI:10.15347/wjm/2014.010. issn 2002-4436.

[5] Classification & Structure of Blood Vessels. SEER training. `https://training.seer.cancer.gov/anatomy/cardiovascular/blood/classification.html#:~:text=The%20wall%20of%20an%20artery,is%20usually%20the%20thickest%20layer.`

[6] A. M. Robertson and P. N. Watton. Mechanobiology of the arterial wall. *Transport in biological Media*, 2013:275–347, 2013.

[7] M. P. Jacob. Extracellular matrix remodeling and matrix metalloproteinases in the vascular wall during aging and in pathological conditions. *Biomedicine & pharmacotherapy*, 57(5–6):195–202, 2003.

[8] R. P. Mecham and F. Ramirez. Extracellular determinants of arterial morphogenesis, growth, and homeostasis. *Current Topics in Developmental Biology*, 130:193–216, 2018.

[9] Diamond Light. About synchrotrons. `hhttps://www.diamond.ac.uk/Home/About/FAQs/About-Synchrotrons.html`.

[10] M. W. Westneat, J. J. Socha, and W. K. Lee. Advances in biological structure, function, and physiology using synchrotron x-ray imaging. *Annual Review of Physiology*, 70:119–142, 2008.

[11] Synchrotron SOLEIL. Beamlines. `https://www.synchrotron-soleil.fr/en/beamlines`.

[12] T. Weitkamp, M. Scheel, J. L. Giorgetta, V. Joyet, V. Le Roux, G. Cauchon, T. Moreno, F. Polack, A. Thompson, and J. P. Samama. The tomography beamline ANATOMIX at

Synchrotron SOLEIL. In *Journal of Physics: Conference Series*, volume 849, page 012037. IOP Publishing, 2017.

[13] N. Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979.

[14] A. Buades, B. Coll, and J. M. Morel. A non-local algorithm for image denoising. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, volume 2, pages 60–65. Ieee, 2005.

[15] A. Buades, B. Coll, and J. M. Morel. Non-local means denoising. *Image Processing On Line*, 1:208–212, 2011.

[16] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Säckinger, and R. Shah. Signature verification using a" siamese" time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(4):669–688, 1993.

[17] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille, 2015.

[18] D. Coeurjolly, J. O. Lachaud, and J. Levallois. Integral based curvature estimators in digital geometry. In *Discrete Geometry for Computer Imagery: 17th IAPR International Conference, DGCI 2013, Seville, Spain, March 20-22, 2013. Proceedings 17*, pages 215–227. Springer, 2013.

[19] H. Pottmann, J. Wallner, Y. L. Yang, Y. K. Lai, and S. M. Hu. Principal curvatures from the integral invariant viewpoint. *Computer Aided Geometric Design*, 24(8-9):428–442, 2007.

[20] H. Pottmann, J. Wallner, Q. X. Huang, and Y. L. Yang. Integral invariants for robust geometry processing. *Computer Aided Geometric Design*, 26(1):37–60, 2009.

[21] D. H. Thijssen, S. E. Carter, and D. J. Green. Arterial structure and function in vascular ageing: are you as old as your arteries? *The Journal of physiology*, 594(8):2275–2284, 2016.

[22] A. Tsamis, J. T. Krawiec, and D. A. Vorp. Elastin and collagen fibre microstructure of the human aorta in ageing and disease: a review. *Journal of the royal society interface*, 10(83):20121004, 2013.

[23] Y. Hosoda, K. Kawano, F. Yamasawa, T. Ishii, T. Shibata, and S. Inayama. Age-dependent changes of collagen and elastin content in human aorta and pulmonary artery. *Angiology*, 35(10):615–621, 1984.

[24] T. Halme, T. Savunen, H. Aho, T. Vihersaari, and R. Penttinen. Elastin and collagen in the aortic wall: changes in the marfan syndrome and annuloaortic ectasia. *Experimental and molecular pathology*, 43(1):1–12, 1985.

[25] E. J. Miller, G. T. Malcom, C. A. McMahan, and J. P. Strong. Atherosclerosis in young white males: arterial collagen and cholesterol. *Matrix*, 13(4):289–296, 1993.

[26] What to know about the cardiovascular system. https://www.medicalnewstoday.com/articles/cardiovascular-system#organs-and-tissues.

[27] G. S. Kassab. Biomechanics of the cardiovascular system: the aorta as an illustratory example. *Journal of the Royal Society Interface*, 3(11):719–740, 2006.

[28] Anatomy of the Cardiovascular System. https://simplemed.co.uk/subjects/cardiovascular/anatomy-of-the-cardiovascular-system.

[29] Wapcaplet. Diagram of the human heart (cropped). https://commons.wikimedia.org/w/index.php?curid=830253.

[30] Department of Internal Medicine. What is Aortic Disease? https://med.uth.edu/internalmedicine/medical-genetics/john-ritter-research-program/what-is-aortic-disease/.

[31] M. Häggström. Using source image by edoarado. https://commons.wikimedia.org/w/index.php?curid=59525246.

[32] World Health Organization: WHO. Cardiovascular diseases. https://www.who.int/health-topics/cardiovascular-diseases#tab=tab_1.

[33] NHS Inform. Cardiovascular disease. illnesses & conditions. https://www.nhsinform.scot/illnesses-and-conditions/heart-and-blood-vessels/conditions/cardiovascular-disease.

[34] N. Sakalihasan, R. Limet, and O. D. Defawe. Abdominal aortic aneurysm. *The Lancet*, 365(9470):1577–1589, 2005.

[35] J. Golledge and P. E. Norman. Atherosclerosis and abdominal aortic aneurysm: cause, response, or common risk factors? *Arteriosclerosis, thrombosis, and vascular biology*, 30(6):1075–1077, 2010.

[36] J. Martel. Atherosclerosis. https://www.healthline.com/health/atherosclerosis.

[37] R. Ross. The pathogenesis of atherosclerosis: a perspective for the 1990s. *Nature*, 362(6423):801–809, 1993.

[38] R. Erbel. Diseases of the thoracic aorta. *Heart*, 86(2):227–234, 2001.

[39] B. S. Brooke, A. Bayes-Genis, and D. Y. Li. New insights into elastin and vascular disease. *Trends in cardiovascular medicine*, 13(5):176–181, 2003.

[40] H. Wolinsky and S. Glagov. Structural basis for the static mechanical properties of the aortic media. *Circulation research*, 14(5):400–413, 1964.

[41] E. G. Lakatta and D. Levy. Arterial and cardiac aging: major shareholders in cardiovascular disease enterprises: Part I: aging arteries: a "set up" for vascular disease. *Circulation*, 107(1):139–146, 2003.

[42] P. Lu, K. Takai, V. M. Weaver, and Z. Werb. Extracellular matrix degradation and remodeling in development and disease. *Cold Spring Harbor perspectives in biology*, 3(12):a005058, 2011.

[43] O. Fritze, B. Romero, M. Schleicher, M. P. Jacob, D. Y. Oh, B. Starcher, K. Schenke-Layland, J. Bujan, and U. A. Stock. Age-related changes in the elastic tissue of the human aorta. *Journal of vascular research*, 49(1):77–86, 2012.

[44] M. F. O'Rourke. Arterial aging: pathophysiological principles. *Vascular medicine*, 12(4):329–341, 2007.

[45] J. B. Michel, D. Heudes, O. Michel, P. Poitevin, M. Philippe, E. Scalbert, B. Corman, and B. I. Levy. Effect of chronic ANG I-converting enzyme inhibition on aging processes. II. Large arteries. *American Journal of Physiology-Regulatory, Integrative and Comparative Physiology*, 267(1):R124–R135, 1994.

[46] S. E. Greenwald. Ageing of the conduit arteries. *The Journal of Pathology: A Journal of the Pathological Society of Great Britain and Ireland*, 211(2):157–172, 2007.

[47] L. de Figueiredo Borges, R. G. Jaldin, R. R. Dias, N. A. G. Stolf, J. B. Michel, and P. S. Gutierrez. Collagen is reduced and disrupted in human aneurysms and dissections of ascending aorta. *Human pathology*, 39(3):437–443, 2008.

[48] M. A. Cattell, P. S. Hasleton, and J. C. Anderson. Increased elastin content and decreased elastin concentration may be predisposing factors in dissecting aneurysms of human thoracic aorta. *Cardiovascular research*, 27(2):176–181, 1993.

[49] M. R. Ward, G. Pasterkamp, A. C. Yeung, and C. Borst. Arterial remodeling: mechanisms and clinical implications. *Circulation*, 102(10):1186–1191, 2000.

[50] X. Yu, R. Turcotte, F. Seta, and Y. Zhang. Micromechanics of elastic lamellae: unravelling the role of structural inhomogeneity in multi-scale arterial mechanics. *Journal of The Royal Society Interface*, 15(147):20180492, 2018.

[51] L. A. Walton, R. S. Bradley, P. J. Withers, V. L. Newton, R. E. Watson, C. Austin, and M. J. Sherratt. Morphological characterisation of unstained and intact tissue micro-architecture by x-ray computed micro-and nano-tomography. *Scientific reports*, 5(1):1–14, 2015.

[52] J. A. Sethian. A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences*, 93(4):1591–1595, 1996.

[53] G. Schena and S. Favretto. Pore space network characterization with sub-voxel definition. *Transport in Porous Media*, 70:181–190, 2007.

[54] B. Trachet, M. Ferraro, G. Lovric, L. Aslanidou, G. Logghe, P. Segers, and N. Stergiopulos. Synchrotron-based visualization and segmentation of elastic lamellae in the mouse carotid artery during quasi-static pressure inflation. *Journal of the Royal Society Interface*, 16(155):20190179, 2019.

[55] J. López-Guimet, J. Andilla, P. Loza-Alvarez, and G. Egea. High-resolution morphological approach to analyse elastic laminae injuries of the ascending aorta in a murine model of marfan syndrome. *Scientific Reports*, 7(1):1–13, 2017.

[56] W. Niblack. *An introduction to digital image processing*. Strandberg Publishing Company, 1985.

[57] J. Sauvola and M. Pietikäinen. Adaptive document image binarization. *Pattern recognition*, 33(2):225–236, 2000.

[58] F. Ciompi, O. Pujol, C. Gatta, M. Alberti, S. Balocco, X. Carrillo, J. Mauri-Ferre, and P. Radeva. Holimab: A holistic approach for media–adventitia border detection in intravascular ultrasound. *Medical image analysis*, 16(6):1085–1100, 2012.

[59] I. Fogel and D. Sagi. Gabor filters as texture discriminator. *Biological cybernetics*, 61(2):103–113, 1989.

[60] T. Ojala, M. Pietikäinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59, 1996.

[61] M. Demi. On the gray-level central and absolute central moments and the mass center of the gray-level variability in low-level image processing. *Computer Vision and Image Understanding*, 97(2):180–208, 2005.

[62] C. Gatta, E. Puertas, and O. Pujol. Multi-scale stacked sequential learning. *Pattern Recognition*, 44(10-11):2414–2426, 2011.

[63] A. Ben Zemzem, A. Genevaux, A. Wahart, A. J. Bodey, S. Blaise, B. Romier-Crouzet, J. Jonquet, C. Bour, R. Cogranne, P. Beauseroy, Dauchez M., Sherratt M. J., Debelle L., and Almagro S. X-ray microtomography reveals a lattice-like network within aortic elastic lamellae. *The FASEB Journal*, 35(10):e21844, 2021.

[64] X. Zhou and G.E. Thompson. Electron and photon based spatially resolved techniques. In *Reference Module in Materials Science and Materials Engineering*, pages 1–30. Elsevier, 2017.

[65] E. N. Landis and D. T. Keane. X-ray microtomography. *Materials characterization*, 61(12):1305–1316, 2010.

[66] J. Hsieh. Computed tomography: principles, design, artifacts, and recent advances. 2003.

[67] A. C. Kak and M. Slaney. *Principles of computerized tomographic imaging*. SIAM, 2001.

[68] L. Quenot, S. Bohic, and E. Brun. X-ray phase contrast imaging from synchrotron to conventional sources: A review of the existing techniques for biological applications. *Applied Sciences*, 12(19):9539, 2022.

[69] A. Bravin, P. Coan, and P. Suortti. X-ray phase-contrast imaging: from pre-clinical applications towards clinics. *Physics in Medicine & Biology*, 58(1):R1, 2012.

[70] Synchrotron SOLEIL. Soleil in 3 questions. https://www.synchrotron-soleil.fr/en/about-us/what-soleil/soleil-3-questions#3.

[71] EPSIM 3D/JF Santarelli. Synchrotron Soleil. https://commons.wikimedia.org/w/index.php?curid=376907.

[72] LEAPS. Synchrotrons. https://leaps-initiative.eu/synchrotrons/.

[73] R. C. Gonzales and P. Wintz. *Digital Image Processing*. Addison-Wesley Longman Publishing Co., Inc., 1987.

[74] K. Singh and R. Kapoor. Image enhancement using exposure based sub image histogram equalization. *Pattern Recognition Letters*, 36:10–14, 2014.

[75] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. B. Zimmerman, and K. Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987.

[76] Y. T. Kim. Contrast enhancement using brightness preserving bi-histogram equalization. *IEEE transactions on Consumer Electronics*, 43(1):1–8, 1997.

[77] Y. Wang, Q. Chen, and B. Zhang. Image enhancement based on equal area dualistic sub-image histogram equalization method. *IEEE transactions on Consumer Electronics*, 45(1):68–75, 1999.

[78] S. D. Chen and A. R. Ramli. Minimum mean brightness error bi-histogram equalization in contrast enhancement. *IEEE transactions on Consumer Electronics*, 49(4):1310–1319, 2003.

[79] S. D. Chen and A. R. Ramli. Contrast enhancement using recursive mean-separate histogram equalization for scalable brightness preservation. *IEEE Transactions on consumer Electronics*, 49(4):1301–1309, 2003.

[80] M. Abdullah-Al-Wadud, M. H. Kabir, M A. A. Dewan, and O. Chae. A dynamic histogram equalization for image contrast enhancement. *IEEE transactions on consumer electronics*, 53(2):593–600, 2007.

[81] H. Ibrahim and N. S. P. Kong. Brightness preserving dynamic histogram equalization for image contrast enhancement. *IEEE Transactions on Consumer Electronics*, 53(4):1752–1758, 2007.

[82] S. C. Huang, F. C. Cheng, and Y. S. Chiu. Efficient contrast enhancement using adaptive gamma correction with weighting distribution. *IEEE transactions on image processing*, 22(3):1032–1041, 2012.

[83] K. Somasundaram and P. Kalavathi. Medical image contrast enhancement based on gamma correction. *International Journal of Knowledge Management and E-Learning*, 3(1):15–18, 2011.

[84] S. Rahman, M. M. Rahman, M. Abdullah-Al-Wadud, G. D. Al-Quaderi, and M. Shoyaib. An adaptive gamma correction for image enhancement. *EURASIP Journal on Image and Video Processing*, 2016(1):1–13, 2016.

[85] K. F. Riley, M. P. Hobson, and S. J. Bence. *Mathematical Methods for Physics and Engineering: A Comprehensive Guide.* Cambridge University Press, 2006.

[86] M. Sezgin and B. L. Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic imaging*, 13(1):146–168, 2004.

[87] J. Kittler and J. Illingworth. Minimum error thresholding. *Pattern recognition*, 19(1):41–47, 1986.

[88] J. Bernsen. Dynamic thresholding of gray level image. *Proceedings of International Conference on Pattern Recognition*, pages 1251–1255, 1986.

[89] D. Lesage, E. D. Angelini, I. Bloch, and G. Funka-Lea. A review of 3D vessel lumen segmentation techniques: Models, features and extraction schemes. *Medical image analysis*, 13(6):819–845, 2009.

[90] T. Boskamp, D. Rinck, F. Link, B. Kummerlen, G. Stamm, and P. Mildenberger. New vessel analysis tool for morphometric quantification and visualization of vessels in CT and MR imaging data sets. *Radiographics*, 24(1):287–297, 2004.

[91] C. Metz, M. Schaap, A. Van Der Giessen, T. Van Walsum, and W. Niessen. Semi-automatic coronary artery centerline extraction in computed tomography angiography data. In *2007 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 856–859. IEEE, 2007.

[92] J. Yi and J. B. Ra. A locally adaptive region growing algorithm for vascular segmentation. *International Journal of Imaging Systems and Technology*, 13(4):208–214, 2003.

[93] Y. Masutani, T. Schiemann, and K. H. Höhne. Vascular shape segmentation and structure extraction using a shape-based region-growing model. In *Medical Image Computing and Computer-Assisted Intervention-MICCAI'98: First International Conference Cambridge, MA, USA, October 11–13, 1998 Proceedings 1*, pages 1242–1249. Springer, 1998.

[94] P. A. Brathwaite, K. B. Chandran, D. D. McPherson, and E. L. Dove. Lumen detection in human IVUS images using region-growing. In *Computers in Cardiology 1996*, pages 37–40. IEEE, 1996.

[95] F. Meyer and S. Beucher. Morphological segmentation. *Journal of visual communication and image representation*, 1(1):21–46, 1990.

[96] J. Rivera-Rovelo and E. Bayro-Corrochano. Medical image segmentation, volume representation and registration using spheres in the geometric algebra framework. *Pattern recognition*, 40(1):171–188, 2007.

[97] R. Adams and L. Bischof. Seeded region growing. *IEEE Transactions on pattern analysis and machine intelligence*, 16(6):641–647, 1994.

[98] A. Mehnert and P. Jackway. An improved seeded region growing algorithm. *Pattern Recognition Letters*, 18(10):1065–1071, 1997.

[99] Y. L. Chang and X. Li. Adaptive image region-growing. *IEEE transactions on image processing*, 3(6):868–872, 1994.

[100] S. A. Hojjatoleslami and J. Kittler. Region growing: a new approach. *IEEE Transactions on Image processing*, 7(7):1079–1084, 1998.

[101] C. Revol and M. Jourlin. A new minimum variance region growing algorithm for image segmentation. *Pattern Recognition Letters*, 18(3):249–258, 1997.

[102] C. Revol-Muller, F. Peyrin, C. Odet, and Y. Carillon. Automated 3D region growing algorithm governed by an evaluation function. In *Proceedings 2000 International Conference on Image Processing*, volume 3, pages 440–443. IEEE, 2000.

[103] H. Singh, M. Crawford, J. Curtin, and R. Zwiggelaar. Automated 3D segmentation of the lung airway tree using gain-based region growing approach. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2004: 7th International Conference, Saint-Malo, France, September 26-29, 2004. Proceedings, Part II 7*, pages 975–982. Springer, 2004.

[104] H. T. Yau, Y. K. Lin, L. S. Tsou, and C. Y. Lee. An adaptive region growing method to segment inferior alveolar nerve canal from 3D medical images for dental implant surgery. *Computer-Aided Design and Applications*, 5(5):743–752, 2008.

[105] J. A. Martínez-Mera, P. G. Tahoces, J. M. Carreira, J. J. Suárez-Cuenca, and M. Souto. A hybrid method based on level set and 3D region growing for segmentation of the thoracic aorta. *Computer Aided Surgery*, 18(5–6):109–117, 2013.

[106] M. Tuceryan and A. K. Jain. Texture analysis. *Handbook of pattern recognition and computer vision*, pages 235–276, 1993.

[107] R. M. Haralick, K. Shanmugam, and I. H. Dinstein. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6):610–621, 1973.

[108] D. Harwood, T. Ojala, M. Pietikäinen, S. Kelman, and L. Davis. Texture classification by center-symmetric auto-correlation, using kullback discrimination of distributions. *Pattern Recognition Letters*, 16(1):1–10, 1995.

[109] P. Cavalin and L. S. Oliveira. A review of texture classification methods and databases. In *2017 30th SIBGRAPI Conference on graphics, patterns and images tutorials (SIBGRAPI-T)*, pages 1–8. IEEE, 2017.

[110] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987, 2002.

[111] M. Idrissa and M. Acheroy. Texture classification using Gabor filters. *Pattern Recognition Letters*, 23(9):1095–1102, 2002.

[112] L. Armi and S. Fekri-Ershad. Texture image analysis and texture classification methods–A review. *arXiv preprint arXiv:1904.06554*, 2019.

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

[114] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60:91–110, 2004.

[115] T. Ojala, M. Pietikainen, and D. Harwood. Performance evaluation of texture measures with classification based on kullback discrimination of distributions. In *Proceedings of 12th international conference on pattern recognition*, volume 1, pages 582–585. IEEE, 1994.

[116] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[117] I. R. I. Haque and J. Neubert. Deep learning approaches to biomedical image segmentation. *Informatics in Medicine Unlocked*, 18:100297, 2020.

[118] Z. Li, F. Liu, W. Yang, S. Peng, and J. Zhou. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE transactions on neural networks and learning systems*, 2021.

[119] P. Simon and V. Uma. Deep learning based feature extraction for texture classification. *Procedia Computer Science*, 171:1680–1687, 2020.

[120] V. Nair and G. E. Hinton. Rectified linear units improve restricted Boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[121] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[122] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, and Chen T. Recent advances in convolutional neural networks. *Pattern recognition*, 77:354–377, 2018.

[123] V. Andrearczyk and P. F. Whelan. Using filter banks in convolutional neural networks for texture classification. *Pattern Recognition Letters*, 84:63–69, 2016.

[124] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[125] P. H. Le-Khac, G. Healy, and A. F. Smeaton. Contrastive representation learning: A framework and review. *IEEE Access*, 8:193907–193934, 2020.

[126] B. Lake, R. Salakhutdinov, J. Gross, and J. Tenenbaum. One shot learning of simple visual concepts. In *Proceedings of the annual meeting of the cognitive science society*, volume 33, 2011.

[127] I. Melekhov, J. Kannala, and E. Rahtu. Siamese network features for image matching. In *2016 23rd international conference on pattern recognition (ICPR)*, pages 378–383. IEEE, 2016.

[128] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, volume 1, pages 487–495, 2014.

[129] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.

[130] R. Yamada, H. Ide, N. Yudistira, and T. Kurita. Texture segmentation using siamese network and hierarchical region merging. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 2735–2740. IEEE, 2018.

[131] T. Kurita. An efficient clustering algorithm for region merging. *IEICE Transactions on Information and Systems*, 78(12):1546–1551, 1995.

[132] D. DeTone, T. Malisiewicz, and A. Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 224–236, 2018.

[133] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[134] G. Logghe, B. Trachet, L. Aslanidou, P. Villaneuva-Perez, J. De Backer, N. Stergiopulos, M. Stampanoni, H. Aoki, and P. Segers. Propagation-based phase-contrast synchrotron imaging of aortic dissection in mice: from individual elastic lamella to 3d analysis. *Scientific reports*, 8(1):2223, 2018.

[135] P. Zhang, P. Zucca, C. Wang, M. M. Bisi, B. Dabrowski, R. A. Fallows, A. Krankowski, J. Magdalenic, G. Mann, D. E. Morosan, and C. Vocks. The frequency drift and fine structures of Solar S-bursts in the high frequency band of LOFAR. *The Astrophysical Journal*, 891(1):89, 2020.

[136] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

[137] M. G. Wagner, P. Laeseke, and M. A. Speidel. Deep learning based guidewire segmentation in x-ray images. In *Medical Imaging 2019: Physics of Medical Imaging*, volume 10948, pages 1002–1008. SPIE, 2019.

[138] G. Palma, M. Comerci, B. Alfano, S. Cuomo, P. De Michele, F. Piccialli, and P. Borrelli. 3D non-local means denoising via multi-GPU. In *2013 Federated conference on computer science and information systems*, pages 495–498. IEEE, 2013.

[139] M. Maggioni, V. Katkovnik, K. Egiazarian, and A. Foi. Nonlocal transform-domain filter for volumetric data denoising and reconstruction. *IEEE transactions on image processing*, 22(1):119–133, 2012.

[140] J. Weickert. A review of nonlinear diffusion filtering. In *Scale-Space Theory in Computer Vision: First International Conference, Scale-Space'97 Utrecht, The Netherlands, July 2–4, 1997 Proceedings 1*, pages 1–28. Springer, 1997.

[141] F. Romdhane, F. Benzarti, and H. Amiri. 3D medical images denoising. In *International Image Processing, Applications and Systems Conference*, pages 1–5. IEEE, 2014.

[142] `DGtal`: Digital geometry tools and algorithms library. `http://libdgtal.org`.

Analyse d'images haute performance pour les données biomédicales très haute résolution acquises en imagerie synchrotron

Les maladies cardiovasculaires sont la première cause de mortalité dans le monde. Le remodelage vasculaire est une conséquence du vieillissement moléculaire des composants de la paroi artérielle. Le vieillissement des fibres (collagène et élastine) de la matrice extracellulaire (MEC) est critique. Nous avons étudié les altérations des lames élastiques, une structure importante de la MEC, en traitant et en analysant les images de la paroi artérielle de souris acquises par imagerie à rayons X synchrotron. Des échantillons d'images de souris saines et diabétiques ont été utilisés dans notre travail. Nous avons proposé un pipeline de segmentation composé d'une extraction de la lumière et d'une segmentation fine basée sur le résultat de la première extraction. La segmentation fine comprend une extraction de la première lame élastique basée sur une croissance de région, et une extraction de l'ensemble des lames élastiques basée sur l'apprentissage profond. Nous avons appliqué une analyse 3D au premier résultat et une analyse 2D au second résultat de la segmentation fine. Nous avons comparé les caractéristiques géométriques, y compris la courbure 3D d'une lamelle entière et l'ondulation des fragments 2D, entre les deux groupes de souris et avons observé des différences significatives entre eux. Ces résultats peuvent contribuer à identifier des biomarqueurs susceptibles de caractériser l'apparition de maladies vasculaires.

Traitement d'images, apprentissage profond, imagerie biomédicale, vieillissement vasculaire

High performance image analysis for very high resolution biomedical data acquired with synchrotron imaging

Cardiovascular diseases are the world's leading cause of death. Vascular remodeling is a consequence of molecular aging of the arterial wall components. Extracellular matrix (ECM) fibers (collagen and elastin) aging is critical. We studied alterations of the elastic lamellae, an important structure in ECM, by processing and analyzing the images of mice arterial wall acquired with synchrotron X-ray imaging. Image samples from healthy and diabetic mice were used in our work. We proposed a segmentation pipeline composed of a lumen extraction and a fine segmentation based on the first extraction result. The fine segmentation includes an extraction of the first elastic lamella based on region growing, and an extraction of all the elastic lamellae based on deep learning. We applied a 3D analysis to the first result and a 2D analysis to the second result of the fine segmentation. We compared the geometric features, including 3D curvature of an entire lamella and waviness of 2D fragments, between the two groups of mice and observed significant differences between them. These results may help identify biomarkers that could characterize the occurrence of vascular disease.

Image processing, deep learning, biomedical imaging, vascular aging

Discipline : INFORMATIQUE

Spécialité : Traitement d'images

Université de Reims Champagne-Ardenne

CRESTIC - EA 3804

UFR Sciences Exactes et Naturelles