INSA Lyon - CREATIS

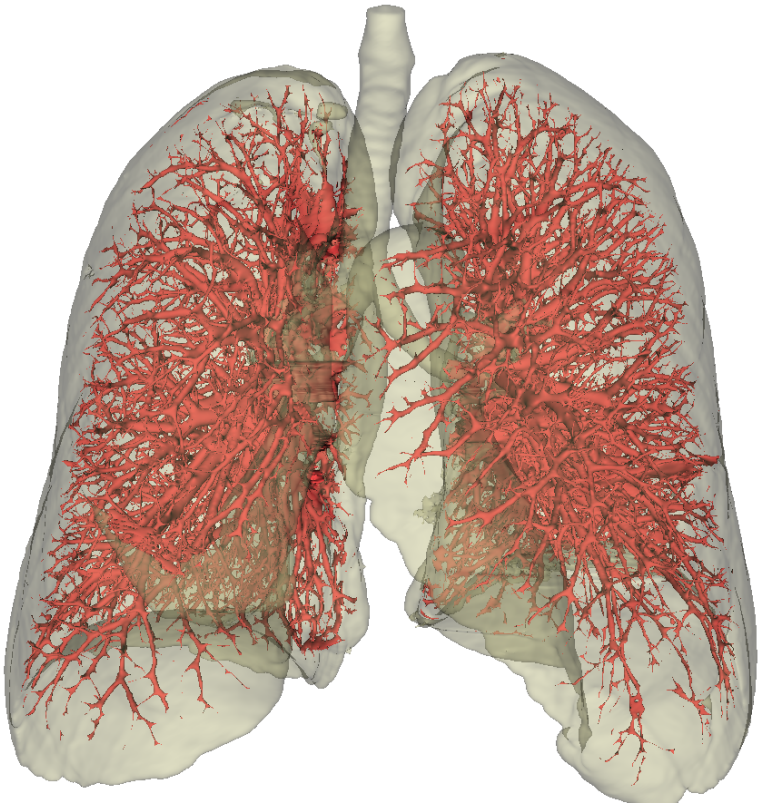
Stage de 4ème année

Juin 2020 - Septembre 2020

Segmentation de l'arbre vasculaire pulmonaire à partir d'images CT injectées pour une meilleure compréhension de sa détérioration chez des patients Covid-19

Auteur :
Titouan POQUILLON

Tutrices et référent :
Carole Frindel, Odyssée Merveille,
Sergio Peignier



1er septembre 2020

Table des matières

Introduction

1.1 CREATIS

Le Centre de Recherche en Acquisition et Traitement de l'Image pour la Santé (CREATIS) est une unité mixte de recherche en imagerie médicale basée à Lyon. Le laboratoire regroupe environ 200 personnes en 4 équipes de recherche :

— MYRIAD (modélisation et imagerie vasculaire thoracique et cérébrale),

— ULTIM (imagerie par ultrason),

— MAGICS (de la mesure au biomarqueur),

— TOMORADIO (imagerie tomographique et thérapie par radiation)

Avec le développement de la pandémie actuelle, le laboratoire CREATIS a réalisé un partenariat avec le CHU Saint-Etienne, qui s'est traduit par la construction d'une base de données d'images scanner de patients pour certain atteints par la Covid-19.

Un projet transverse a ainsi été créé au sein du Creatis réunissant une vingtaine de personnes pour travailler sur ces données autour de trois axes majeurs : l'étude des méta-données, l'étude de la ventilation pulmonaire et l'étude de l'arbre vasculaire pulmonaire (AVP).

C'est pour travailler sur ce dernier axe de recherche que j'ai été embauché par le laboratoire.

Dans le cadre de mon stage, bien qu'employé par l'équipe MYRIAD, j'ai travaillé essentiellement au sein de ce groupe de projet transverse en participant notamment aux réunions hebdomadaires. Mon stage s'étant déroulé entièrement en télétravail, je n'ai pas eu d'autres occasions d'interagir avec les membres extérieurs au projet, que par les réunion scientifiques, organisées toute les deux semaines.

Ces réunions permettaient de présenter les actualités scientifiques dans les différents domaines de l'imagerie médicale. Ce fut pour moi l'occasion de présenter une synthèse bibliographique des observations faites lors des autopsies sur des patients COVID.

1.2 Contexte

La maladie à coronavirus 2019 ou Covid-19 a déjà causé plus de 800 000 morts [3] et engendré de nombreux bouleversements économiques, politiques et sociaux sur l'ensemble du globe. Pour la recherche médicale, comprendre, diagnostiquer et guérir cette maladie sont des défis d'une ampleur inédite.

Dans le cadre de la Covid-19, les poumons, d'abord agressés par le virus et l'infection, subissent ensuite une agression inflammatoire majeure menant à un risque de fibrose pulmonaire et d'insuffisance respiratoire sur le long terme.

Au niveau microscopique, les cellules endothéliales des vaisseaux sanguins sont parmi les premières cibles du SARS-CoV-2 [1]. Cependant, on peut en observer les conséquences de la maladie sur l'ensemble de l'AVP des patients à des échelles bien plus importantes : embolies pulmonaires (obstruction d'une artère), occlusion partielle des vaisseaux et angiogénèse (génération de nouveaux vaisseaux sanguins) anormale [2]. Il s'agit donc d'un organe particulièrement intéressant à étudier, autant pour les diagnostiques que pour le traitement des malades.

L'AVP est une structure tri-dimensionnelle complexe à forte anisotropie. Les vaisseaux sanguins sont des structures fines, difficile à détecter dans les images. Cette problématique est renforcée dans

les zones de lésions présentes chez les patients malades, car celles-ci ont notamment tendances à être confondu avec les vaisseaux sanguins du fait de leur plus forte densité.

1.3 Objectifs du stage

Ce stage s'est articulé autour de 4 objectifs principaux :

1. La construction d'un jeu d'annotations manuelles sur un patient atteint de la Covid-19,
2. La mise en place d'un protocole de segmentation de l'AVP,
3. L'expérimentation de ce protocole, par la construction et l'évaluation d'un premier classifieur et la production de segmentation de l'AVP de plusieurs patients avec celui-ci,
4. la réalisation d'un code fonctionnel et transmissible en python permettant de reproduire ce protocole de segmentation.

Dans ce rapport on reviendra uniquement sur les 3 premiers objectifs. En effet le travail sur le quatrième n'est pas encore complètement fini, le stage se terminant le 11 septembre. La partie du code ayant permis de réaliser les expériences présenté dans ce rapport est terminée. La partie liée au déploiement et à la transmission de ce code est toujours en développement et sera présentée lors de la soutenance.

Au terme de ce stage, ce travail permettra au prochain stagiaire de construire des segmentations de l'AVP dans le but de détecter la présence d'embolies pulmonaires chez les patients COVID-19.

Travail préliminaire

En amont de ce stage, Il a fallu se familiariser avec les concepts et techniques utilisées dans le domaine de l'analyse d'images et plus particulièrement dans celui de la segmentation vasculaire. Cet apprentissage s'est fait autour de 4 grands axes : Le vocabulaire, le challenge Vessel12, RORPO et enfin Ilastik

2.1 Vocabulaire

Chaque discipline dispose de son vocabulaire et la segmentation ne fait pas exception. Celui-ci est indispensable pour bien communiquer et comprendre les différents articles sur le sujet.

2.1.1 Les images

— Image numérique : tableau de points à plusieurs dimensions spatiales (ici 3, l'image étant un scan CT). Chaque point possède une (généralement une intensité) ou plusieurs (c'est le cas pour les images en couleur) valeurs,

— Voxel : plus petit volume composant une image 3d, au même titre que le pixel pour une image 2d,

— Descripteur : un descripteur caractérise une image ou une partie d'une image. Par exemple l'intensité et le gradient sont deux descripteurs d'un voxel.

2.1.2 La segmentation

— Segmentation : une opération de traitement d'images qui a pour but de rassembler des éléments de celle-ci (ici des voxels) en classes, en fonction de leurs descripteurs. Il s'agit dans ce cas d'extraire les voxels des vaisseaux sanguins du reste de l'image,

— Classification : en analyse d'images, la classification consiste à attribuer une classe à une image en fonction de ses propriétés. Comme dans ce projet, on s'intéresse au composants de l'image et pas à l'image dans sa globalité, il s'agit d'un problème de segmentation et non de classification,

— Label : étiquette attribuée à un voxel indiquant à quelle classe il appartient,

— Annotation : Le fait d'attribuer un label à un voxel,

— Métrique : fonction attribuant une mesure de performance au résultat d'un traitement (e.g. une classification ou une segmentation) par rapport à une réalité connue (e.g. des annotations),

— vérité terrain : le résultat attendu d'une application. En traitement d'images il s'agit souvent d'une image annotée manuellement par des experts.

2.1.3 Les algorithmes

— Classifieur : l'algorithme dont le rôle est de réaliser la classification ou la segmentation.

— Apprentissage automatique : on parle d'apprentissage automatique lorsqu'un algorithme est capable d'apprendre à réaliser une tâche en s'entraînant à partir d'une base de données, sans que la méthode pour résoudre cette tâche (dans le cas présent, la segmentation) ne lui ai été explicitement donnée,

— Apprentissage supervisé : dans le cadre d'une classification, on parle d'apprentissage supervisé lorsqu'il est nécessaire que les données d'entraînement de l'algorithme soient annotées. Les forêts aléatoires sont des méthodes d'apprentissage supervisée,

— Apprentissage non supervisé : on parle d'apprentissage non supervisé lorsqu'il n'est pas nécessaire d'annoter les données d'entraînement. Le clustering est par exemple une méthode d'apprentissage non supervisé,

— Arbre de décision : structure de décision représentant un ensemble de classes sous la forme graphique d'un arbre. Les différentes classes possibles sont situées aux extrémités des branches (les « feuilles » de l'arbre), et sont atteintes en fonction de décisions prises à chaque étape (les « noeuds » de l'arbre), en fonction des descripteurs de l'objet à classifier. Un arbre peut être construit par apprentissage automatique à partir de données annotées,

— Forêt aléatoire ou forêt d'arbres décisionnels : type de classifieur basé sur un groupe d'arbres de décisions, chacun entraîné sur un sous-ensemble des données et des descripteurs du jeu de données d'entraînement. La décision finale est faite par vote majoritaire des arbres.

2.2 Challenge Vessel12

Avant de s'intéresser aux patients de la base de données CTPRED, l'étude du challenge Vessel-12 a été une bonne introduction au concept de segmentation vasculaire pulmonaire. Ce projet organisé par le Symposium International d'Imagerie Biomédicale (ISBI) 2012 avait pour objectif de fournir une comparaison objective des performances de segmentation de différents algorithmes sur des scanners pulmonaires injectés ou non injectés.

Ce challenge fournit un lot de données annotées qui ont permis de réaliser les premières expérimentations de segmentation. De plus, une analyse du challenge, des différents algorithmes ayant été pris en compte et des méthodes d'évaluation utilisées [4], a permis de construire la base des méthodes d'annotations et d'évaluation utilisées lors de ce stage, notamment avec l'utilisation des courbes ROC.

Enfin, les résultats de ce challenge, notamment la prévalence de méthodes utilisant des filtres Hessiens, ont orienté le choix des descripteurs.

2.3 RORPO

Développé en 2017, RORPO est une méthode d'analyse d'images basée sur la morphologie mathématique spécialisée dans la détection des structures curvilignes. Contrairement aux méthodes classiques de la litérature, RORPO est un filtre non linéaire avec une approche non locale [5].

Plutôt que d'étudier chaque point et son voisinage immédiat, RORPO utilise une approche plus globale à travers la constructions de chemins entres les points de l'image.

Ce filtre permet d'attribuer à chaque pixel ou voxel d'une image une intensité traduisant son appartenance à une structure curviligne. Cela permet de faire ressortir ces structure, par rapport à des structure planaires ou isotropes.

Cette méthode est donc particulièrement adaptée à la segmentation de l'AVP, les vaisseaux sanguins étant des structures curvilignes. Elle a d'ailleurs été développée dans le cadre de la détection des vaisseaux sanguin cérébraux.

Dans le cadre ce stage, RORPO a été utilisé comme un descripteur supplémentaire, pour compléter un corpus traditionnel de descripteurs linéaires locaux.

2.4 Ilastik

Ilastik est un logiciel de segmentation d'images interactif qui propose une outil de segmentation basé sur les forêts aléatoires [6]. Ce logiciel est particulièrement facile d'utilisation pour les personnes

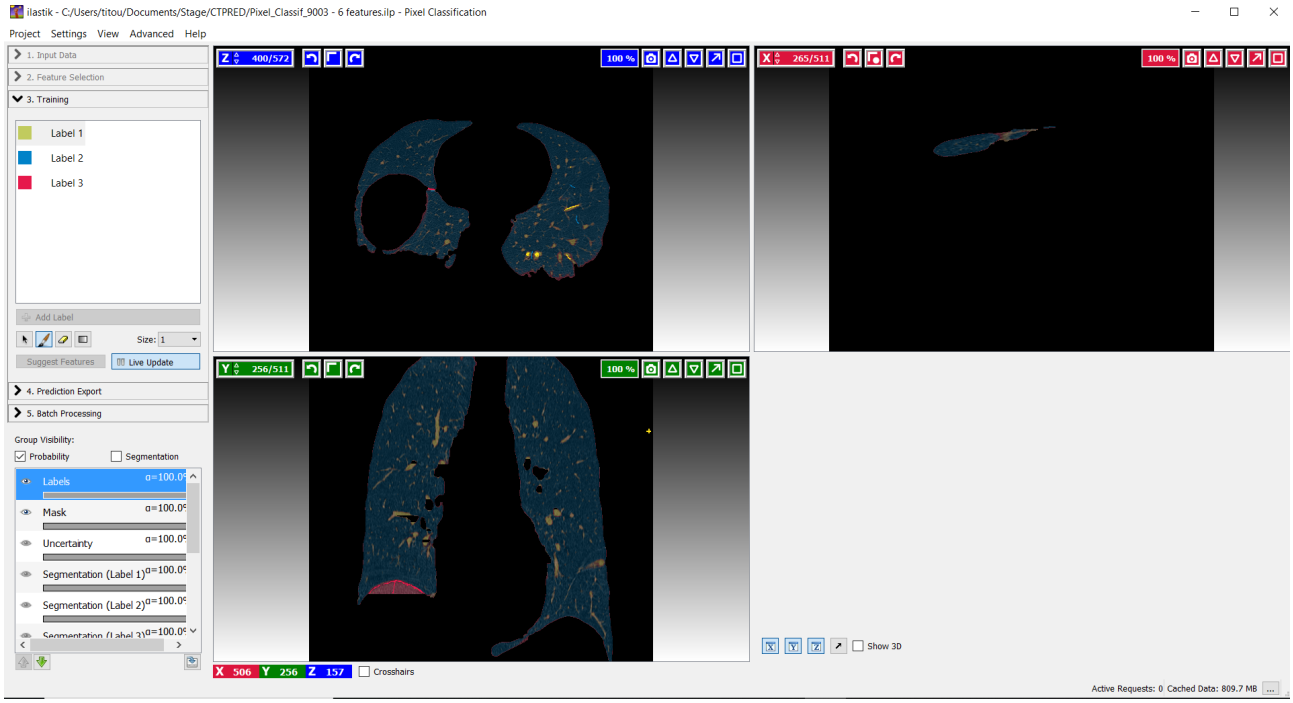


FIGURE 2.1 – Interface graphique du logiciel Ilastik. On peut réaliser une labélisation et observer la segmentation résultante en direct

sans expérience dans le domaine. Ce fut donc une excellente porte d'entrée pour ce stage.

Grâce à son interface graphique (figure 2.1), Ilastik permet de réaliser et de visualiser simultanément les différentes étapes d'un projet de segmentation :

1. L'importation des données
2. Le choix des descripteurs
3. Les annotations
4. La segmentation

Finalement, l'utilisation de ce logiciel a eu un impact fort sur celle de ce protocole, notamment sur le choix de l'algorithme de classification : les forêts aléatoires.

Bien qu'Ilastik permette facilement de réaliser des projets de segmentation, notamment sur des images en 3D, il est beaucoup moins pratique à utiliser de façon non interactive au sein d'un projet de programmation. Ilastik ne propose qu'un nombre limité de descripteurs et certains processus, comme leur sélection, ne sont pour le moment pas assez documentés pour être utilisés.

Pour ces raisons, au cours du stage, on a choisi de redévelopper sous python les fonctionnalités d'intérêt fournies par Ilastik pour s'abstraire du logiciel et pouvoir développer les fonctionnalités manquantes.

Méthodologie

Ce stage à été l'occasion de formaliser et de développer les différentes étapes d'un projet d'analyse d'image, des données brutes aux segmentations. L'enchaînement des différentes étapes du protocole est présenté dans le logigrame de la figure 3.1.

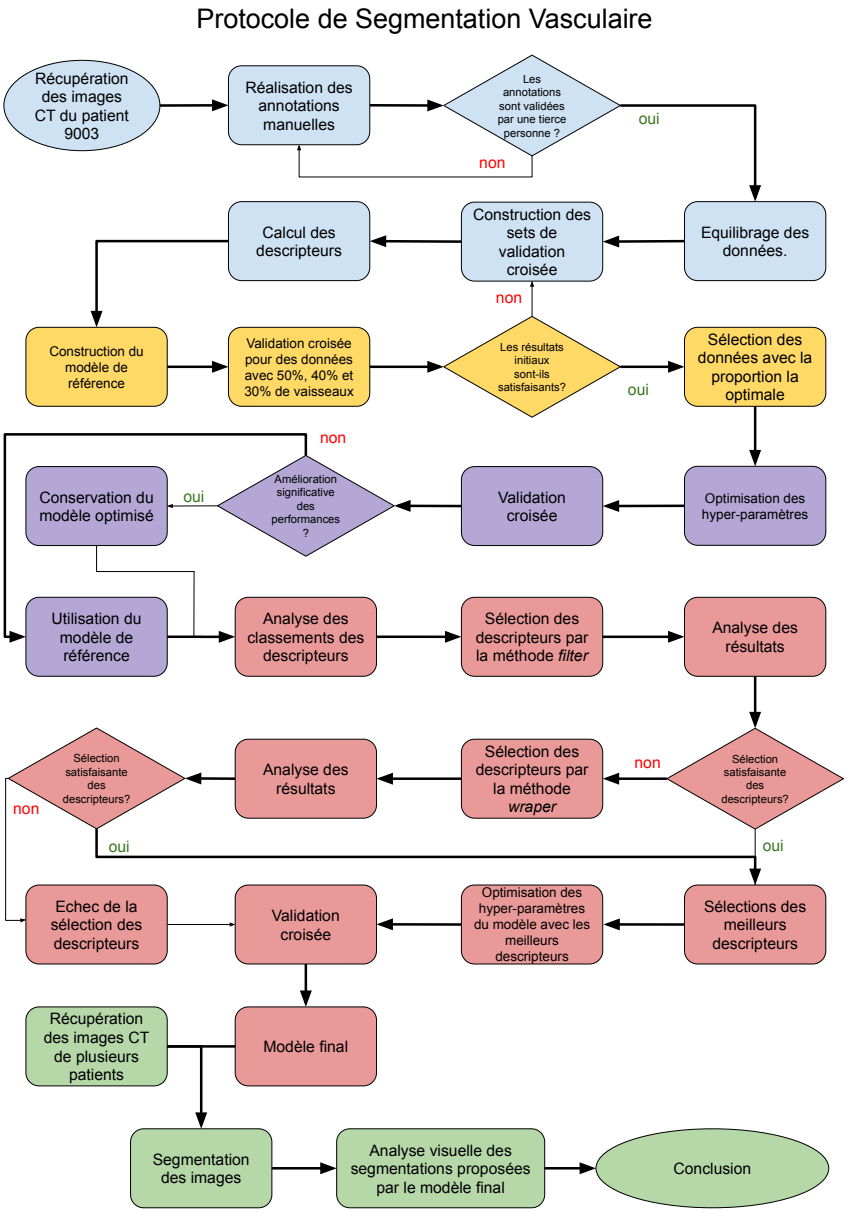
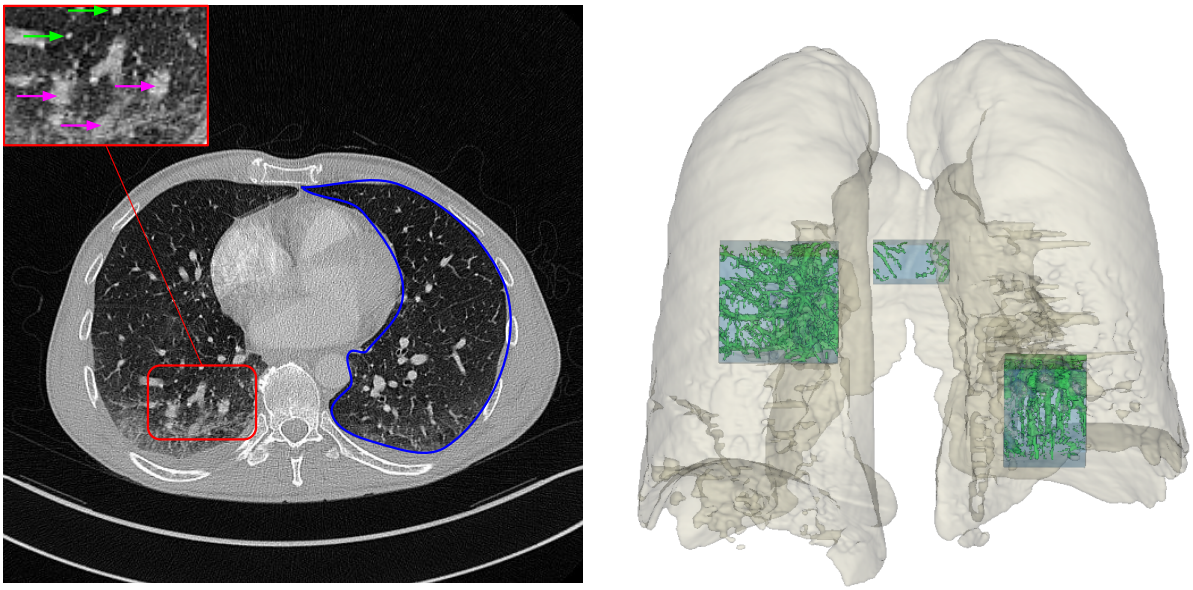


FIGURE 3.1 – Logigramme du protocole de segmentation vasculaire. En bleu : les étapes liées à la préparation des données, en jaune : celles liées à la première expérience de validation croisée et à la sélection des proportions de vaisseaux, en violet : celles de l'optimisation des hyper-paramètres, en rouge : les étapes de la sélection des descripteurs et en vert : celles liées à la segmentation et à l'évaluation du modèle final sur plusieurs patients.



(a) (b)

FIGURE 3.2 – (a) Coupe transversale issue d'un scan CT injecté du patient 9003 (malade de la Covid-19) de la base CTPRED. Les deux lobes sombres sont les poumons du patient. Le lobe de droite est contouré en bleu. On peut observer dans la partie dorsale (en base) du poumon de gauche une zone de lésion (en rouge). Les flèches vertes pointent sur des vaisseaux, et les flèches roses sur des lésions. (b) Annotations manuelles (en vert) réalisées dans chacune des boites (en bleu) pour le patient 9003 de la base CTPRED.

3.1 Les données

3.1.1 Scans CT

Les images utilisées pour ce stage sont produites par tomodensitométrie calculée par ordinateur ou scan CT (*Computerized Tomography*). Cette méthode consiste à mesurer l'absorption de rayons X par les tissus, et reconstruire ensuite numériquement une carte tridimensionnelle de la structure interne d'un patient.

Chaque voxel (l'équivalent d'un pixel en 3d) de l'image possède une valeur en unité Hounsfield, décrivant la radio-densité du tissu qu'il couvre. Grâce à l'étalonnage, les images obtenues par deux scanners différents sur le même patient sont normalement similaires. Les appareils qui ont produit les images que l'on utilise atteignent des résolutions de l'ordre du demi-millimètre qui peut légèrement varier en fonction des dimensions du patient. A titre d'exemple, le scan CT du patient 9003 est une images de 512*512*573 voxels de volumes $0.7*0.7*0.5^1$ mm^3. L'image totale fait donc environ $36*36*29$ cm^3

Le scanner CT permet d'étudier un volume sans discontinuité anatomique, avec un rehaussement des vaisseaux grâce à l'injection de produit de contraste. C'est donc un outil privilégié pour réaliser des images de l'AVP.

A l'exception des segmentations finales, l'ensemble des expériences présentées dans le cadre de ce rapport se base sur des images extraites du patient 9003 de la base CTPRED, atteint par la Covid-19. Travailler sur un seul patient est la source de biais, mais permet de travailler à une échelle bien plus abordable pour un stage de 3 mois.

3.1.2 Annotations

En l'absence de données annotées pré-existantes sur la base CTPRED, il m'a fallu les construire manuellement pour entraîner et évaluer le futur modèle. Ce processus demande du temps, une formation médicale et idéalement la reproduction des annotations par plusieurs personnes différentes. En l'absence des deux dernier, les annotations réalisées lors de ce stage sont donc probablement imparfaites.

Les poumons sont des organes vastes, et toutes leurs zones ne présentent pas la même complexité. Par complexité, on entend la difficulté à différencier les vaisseaux du reste du tissu pulmonaire. Ainsi il est plus facile de séparer une grosse artère bien contrastée par rapport au tissu environnant, qu'un petit vaisseau dans une lésion pulmonaire fortement contrastée.

Ces zones complexes sont, en proportion, bien moins représentées que les zones plus "classiques" du poumon. Il est donc très improbable que la sélection aléatoire de voxels dans l'image permette de construire un jeu de données équilibré où les points issus de zones de complexités différentes sont présents en quantités équivalentes. Cela pourra se traduire, à terme, par l'incapacité d'un modèle à réaliser des prédictions correctes dans ces zones. Or quelques unes, notamment autour des lésions, peuvent être particulièrement intéressantes. C'est pourquoi on a choisi de réaliser les annotations dans des "boîtes" représentant des zones de complexité différentes pour qu'un modèle soit capable d'identifier des vaisseaux de tailles diverses, et soit capable de les différencier des lésions :

— Boîte "healthy" comprenant environ 20000 voxels annotés comme vaisseaux. Zone du poumon normalement saine et avec peu de lésions. Cette zone comporte des vaisseaux larges et moyen,

— Boîte "unhealthy" comprenant environ 20000 voxels annotés comme vaisseau. Cette zone comporte des lésions,

— Boîte "edge" comprenant environ 1000 voxels annotés comme vaisseau. Cette zone en bordure du poumon présente essentiellement des vaisseaux fins.

3.1.3 Préparation des données

Les boîtes sont de tailles différentes, et la densité de vaisseaux dans les poumons est faible par rapport au reste du tissu pulmonaire. Cependant pour entraîner correctement un modèle, il faut des données équilibrées.

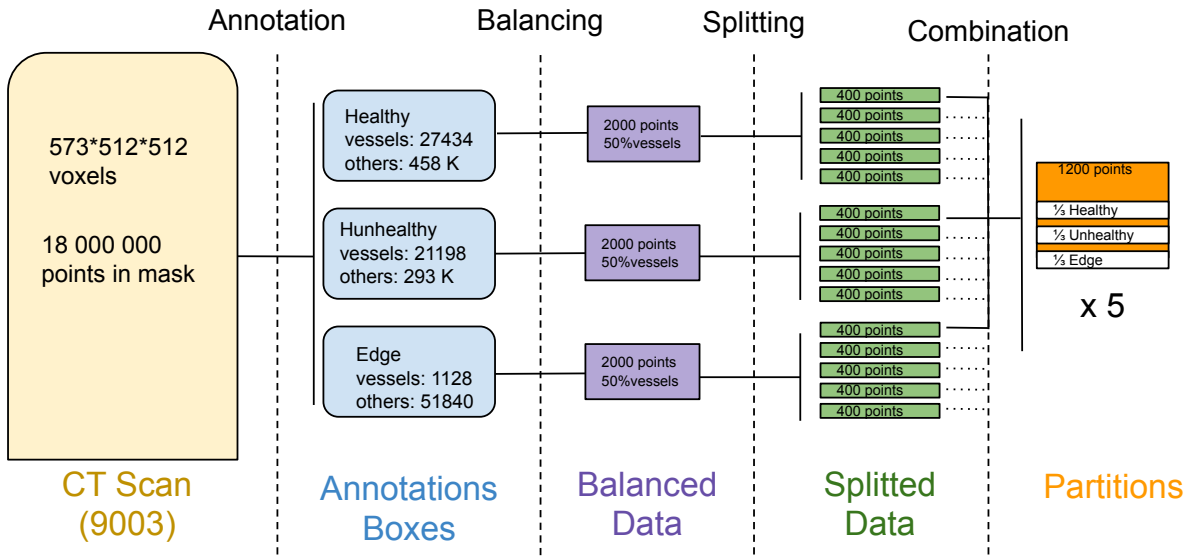


FIGURE 3.3 – Préparation des données de validation croisée. Exemple avec une proportion de vaisseaux de 50%

— en terme de proportion de vaisseaux

— en terme de complexité (boîtes)

Pour chaque boîte, on extrait de façon aléatoire un groupe de 2000 voxels : 50% vaisseaux, 50% non vaisseau. Chaque groupe est ensuite divisé en 5 ensembles de 400 points. Enfin, on construit, en combinant les ensembles issus de chacune des 3 boîtes, 5 partitions de 1200 points, pour lesquels la contribution de chaque boîte est équivalente et composée à 50% de vaisseaux. Ces jeux de données constitueront les partitions de la validation croisée (voir section 3.3.2). L'ensemble du processus de construction des jeux de données est résumé dans la figure 3.3.

Pour évaluer l'impact de la proportion de vaisseaux, on répétera ces opérations pour des jeux de données composés de seulement 40%, puis 30% de vaisseaux. Ils seront construits à partir de la même graine (même générateur aléatoire) que le jeu de données à 50%. L'ensemble des points labélisés "vaisseau" de ces jeux de données est donc inclus dans celui à 50%. Lors de l'évaluation de notre modèle de référence (voir section 3.2.3), on répétera l'évaluation pour chaque proportion de vaisseaux ce qui nous permettra de choisir la proportion optimale pour la suite des expériences.

3.2 Modèles

L'objectif de ce protocole est d'obtenir un modèle de classifieur spécialisé dans la segmentation de l'AVP. Un modèle est caractérisé par un ensemble de descripteurs et de paramètres permettant de construire un classifieur.

3.2.1 Classifieur

Chaque classifieur est une forêt aléatoire créée à partir du package Python Scikit-learn. Il doit être préalablement entraîné sur un jeu de données annotées avant de pouvoir réaliser des segmentions. Pour chaque point, une forêt aléatoire prend en entrée une série de descripteurs et renvoie la probabilité que ce point appartienne à la classe d'intérêt (ici la classe vaisseau sanguin). Une forêt aléatoire ne fonctionne que pour un unique ensemble de descripteurs qui lui est propre.

3.2.2 Descripteurs

Chacun des points du jeu de données est associé à une série de valeurs permettant de le décrire. C'est en fonction de celles-ci qu'un classifieur réalise une segmentation. On appellera par la suite ces valeurs des *descripteurs* de l'image. Ces descripteurs sont construits à partir des données brutes du scan CT et permettent d'ajouter à chaque point des informations supplémentaires sur leur environnement.

Il existe théoriquement une infinité de descripteurs différents. Dans le cadre de ce stage, 4 types de filtres traduisant des particularités structurales ont été utilisés :

1. le lissage Gaussien, qui donne une information sur l'intensité autour du point. Une valeur élevée indique la présence d'une structure en ce point,
2. le laplacien qui donne une information sur la discontinuité des intensités autour du point. Un fort laplacien indique que le point se trouve à la frontière entre deux structures différentes.
3. les valeurs propres (VP) de la matrice Hessienne calculées en chaque point de l'image. Il s'agit d'un descripteur multivarié permettant d'obtenir 3 valeurs caractérisant la géométrie de la structure autour du point :
   — tubulaire lorsque seule la première VP est importante,
   — surfacique lorsque les 2 premières VP sont importantes,
   — isotropique lorsque les trois VP sont importantes,
4. la valeur du filtre RORPO, qui donne une information sur l'appartenance du point à une structure tubulaire.

Ces filtres (sauf RORPO) sont calculés à différentes échelles, décrivant l'environnement du point plus ou moins proche : le lissage Gaussien qui est réalisé avant le calcul de chaque descripteur (voir figure 3.4) permet d'atténuer les structures de taille inférieure à l'échelle du filtre. Plusieurs échelles permettent donc de se concentrer sur des structures de tailles différentes. Les valeurs de ces échelles (en pixels) ont été choisies en fonction de la taille des vaisseaux : [0.1, 0.2, 0.3, 0.5, 1.0, 1.5, 2.0 , 3.0, 5.0, 10]. Le filtre RORPO est lui aussi calculé sur des échelles différentes, mais celles-ci ne sont pas basées sur le lissage gaussien. Les échelles utilisées (en pixels) sont : [20, 40, 60, 80, 100]

En combinant les types de filtres et les échelles, et en ajoutant les données brutes, on obtient un total de 56 descripteurs pour entraîner les modèles. On verra dans la section 3.4.2 qu'en fonction de l'importance de certains de ces descripteurs et de leur impact sur le modèle, on pourra être amené à diminuer ce nombre (voir figure 3.4).

3.2.3 Modèle de référence

Une fois les jeux de données construits et partitionnés, on produit un modèle de référence qui servira de base à la suite des expériences. Ce modèle utilise l'ensemble des 56 descripteurs et les paramètres d'une forêt aléatoire par défaut de Scikit-learn (n_estimator = 100, criterion = 'gini', max_depht = None, min_sample_split = 2, min_sample_leaf = 1).

3.3 Évaluation

Comme l'entraînement d'une forêt est un processus stochastique, deux classifieurs issus du même modèle peuvent avoir une structure et des performances différentes. Il est donc nécessaire d'entraîner et d'évaluer plusieurs classifieurs pour estimer les performances d'un modèle.

3.3.1 Métriques

Pour évaluer un classifieur sur un ensemble de points, on réalise la prédiction de ce classifieur pour chaque point, que l'on compare à la vérité terrain de l'image en question. Une prédiction est la probabilité (valeur entre 0 et 1) pour un point d'être un vaisseau sanguin. Cette prédiction doit être seuillée pour obtenir une segmentation comparable à la vérité terrain. Dans ce protocole, on a choisi d'utiliser 200 seuils différents. On évalue donc, non pas une, mais 200 segmentations.

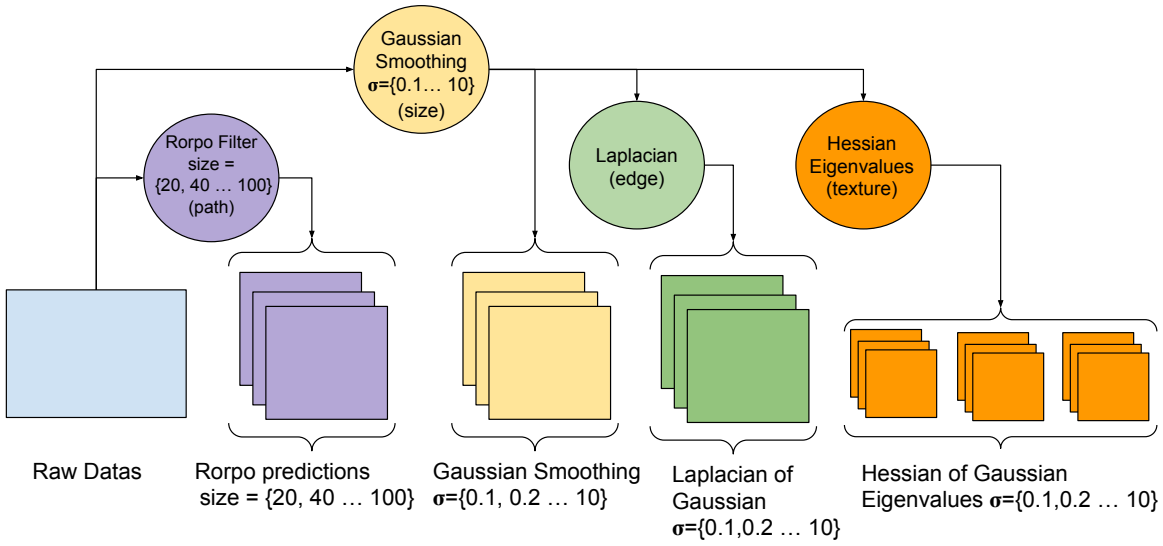


FIGURE 3.4 – Construction du jeu initial de 56 descripteurs pour une image 3D issue de scan CT

Pour chaque valeur de seuil, on mesure le nombre de vrais positifs (VP), faux positifs (FP), vrais négatifs (VN et faux négatifs (FN) entre la segmentation et les annotations. Ces mesures permettent de calculer les métriques suivantes :

— la sensibilité $\frac{TP}{TP+FN}$,
— la spécificité $\frac{TN}{TN+FP}$,
— le Matthews correlation coefficient (MCC) $\frac{TP*TN-FP*FN}{\sqrt{(TP+FP)*(TP+FN)*(TN+FP)*(TN+FN)}}$ compris entre -1 et 1, optimal lorsque égal à 1
— l'indice de Sorensen-Dice (Dice) : $\frac{2TP}{2TP+TN+FP}$ compris entre 0 et 1, optimal lorsque égal à 1

Une sensibilité élevée signifie qu'on détecte une majorité de vaisseaux de l'image, tandis qu'une spécificité élevé signifie on détecte peu de non vaisseau comme vaisseaux. Un MCC élevé est la marque d'une qualité générale de la classification. Enfin, Un Dice proche de 1 est la marque d'une bonne superposition entre les vaisseaux sanguins segmentés et les vaisseaux sanguins annotés.

Avec ces valeurs calculées pour l'ensemble des segmentations, on peut tracer les courbes de chaque métrique en fonction du seuillage (voir figure 3.5). Une représentation synthétique et visuelle des résultats consiste à tracer la courbe ROC (de l'anglais *Receiver Operating Characteristic*) qui affiche le taux de FP (le complémentaire 1 de la spécificité) en fonction du taux de VP (sensibilité). Cette courbe permet de représenter visuellement de façon efficace la qualité générale d'un modèle, et elle est particulièrement utile pour comparer deux classifieurs entre eux et pour trouver le seuil de segmentation optimal. Il s'agit de celui pour lequel la distance de la courbe au point (0, 1) du graphique est minimale.

Afin d'analyser les résultat on s'intéresse particulièrement aux métriques suivantes :

— la valeur du Dice maximale, et le seuil correspondant,
— la valeur du Dice au seuil 0.5 ,
— la valeur du MCC maximale, et le seuil correspondant,
— la valeur du MCC au seuil 0.5,
— la distance de la courbe ROC au point (0,1) minimal, et le seuil correspondant,
— la distance de la courbe ROC au point (0,1) au seuil 0.5,

3.3.2 Validation croisée

Les modèles sont évalués par validation croisée en 5 folds. Cette méthode particulièrement utilisée en apprentissage automatique permet, en plus d'évaluer les performances d'un modèle, d'étudier sa

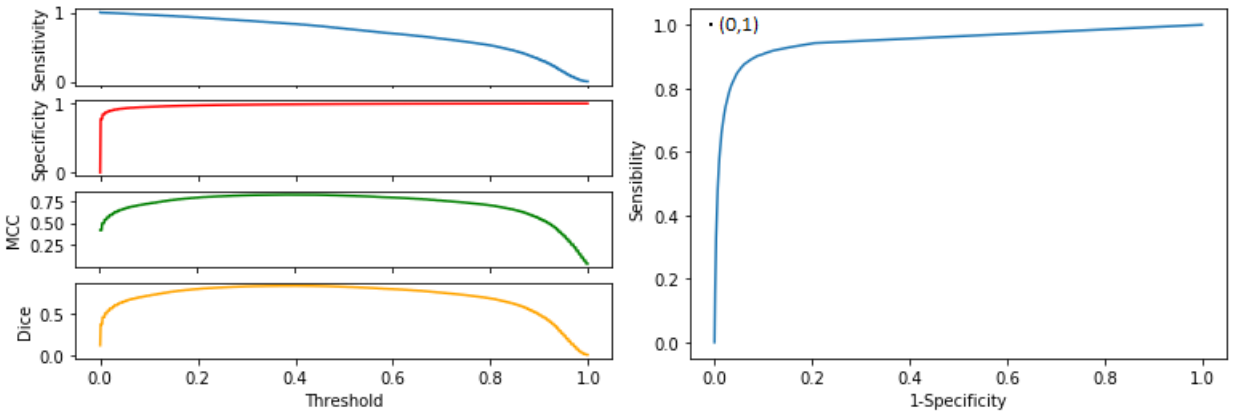


FIGURE 3.5 – L'évolution des différentes métriques utilisées pour évaluer la qualité d'une prédiction en fonction du seuillage

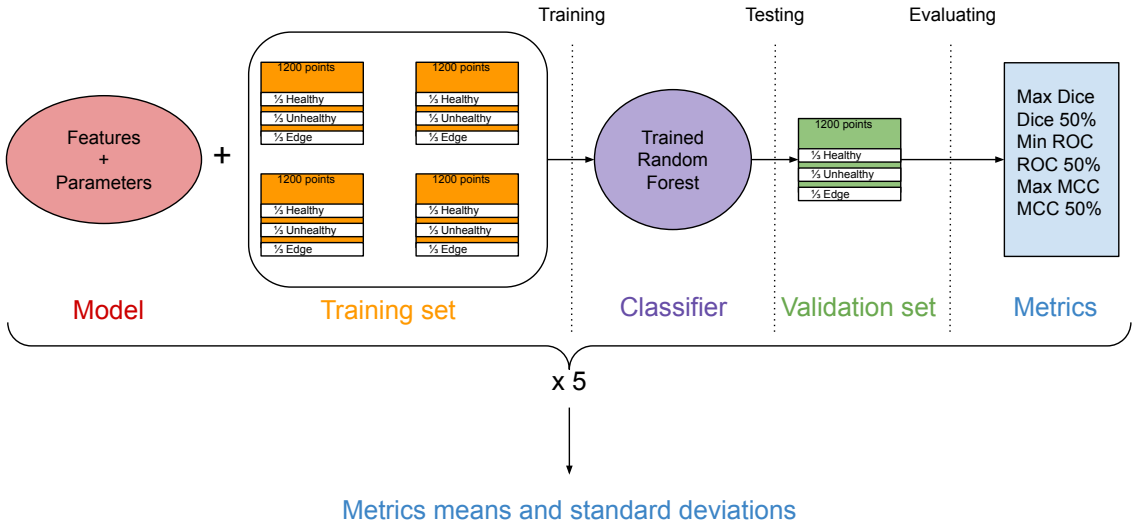


FIGURE 3.6 – Représentation schématique du processus de validation croisée à 5 folds

fiabilité.

Les données sont séparées 5 partitions. A partir de ces partitions, on construit 5 combinaisons différentes. Chaque combinaison est composée de 4 partitions qui servent à entraîner un classifieurs (données d'entraînement) et d'une partition servant à l'évaluer (données de validation). (voire fig 3.6). On ne s'intéresse plus à un classifieur unique, mais à un ensemble de classifieurs dont on récupère les valeurs moyennes de chaque métrique.

Chacune des étapes de la validation croisée fournie ces métriques. On évalue le qualité d'un modèle à partir de leur moyenne sur l'ensemble des étapes. On s'intéresse également à l'écart type de ces métriques, indiquant la variabilité des performances d'un classifieur.

3.4 Optimisation

On essaie d'améliorer le modèle de référence autour de 2 axes d'optimisation : les hyper-paramètres et les descripteurs. On cherche à obtenir après ces optimisations un modèle final :

— présentant les meilleures performances de segmentation,

— moins lourd et plus rapide (voir section3.4.2),

— plus facilement interprétable.

3.4.1 Choix des hyperparamètres

Il s'agit d'une étape très classique dans l'optimisation d'un modèle de forêt aléatoire. Trouver la bonne combinaison d'hyperparamètres permet d'améliorer les performances d'un modèle en l'adaptant à la complexité de son sujet. Ces améliorations viennent parfois avec un coût, puisque certains hyperparamètres entraînent une augmentation du temps de segmentation. De plus, plus le nombre de paramètres optimisés est grand, plus le temps de calcul de l'algorithme d'optimisation augmente.

Dans ce projet on a optimisé les 5 hyperparamètres principaux d'une forêt aléatoire :

— *criterion* ("gini" ou "entropy"), il s'agit de la fonction de coût utilisée lors de la séparation des branches lors de la construction d'un arbre,

— *max_depth* (10, 20, 30, 40, 50, 60, 70, 80, 90 ou 100), la taille maximale d'un arbre),

— *max_features* ('auto' ou 'sqrt'), le nombre maximum de descripteurs utilisés pour construire chaque arbre),

— *n_estimators* (100, 200, 400, 800, 1000), le nombre d'arbres de la forêt. Le nombre d'arbres augmente de façon proportionnelle le temps de calcul,

— *min_samples_leaf* (1, 2, 4) et *min_samples_split* (2, 5, 10) le nombre d'échantillons nécessaires à la séparation des noeuds internes d'un arbre et à la création de ses feuilles

Tester l'ensemble des 1800 combinaisons possibles avec ces paramètres est relativement long. On préférera utiliser l'outil de Scikit-learn *RandomizedSearchCV* qui permet de réaliser seulement 300 combinaisons aléatoires pour étudier les effets des hyperparamêtres sur un modèle.

3.4.2 Sélection des descripteurs

Comme le modèle initial utilise 56 descripteurs, leur calcul prend du temps (de l'ordre de la dizaine de seconde pour chaque descripteur classique, et de la minute pour RORPO) et de la place (31Go pour les descripteurs du patient 9003). De plus, ils apportent probablement une information redondante sur les voxels. Enfin, plus le nombre de descripteurs est grand plus il est difficile d'interpréter les résultats obtenus. De plus, l'interprétabilité d'un modèle est particulièrement importante dans le monde médical

Pour comprendre quel est l'impact de chaque descripteur dans un classifieur, on utilise la methode des Shape-Values [7]. Ces valeurs permettent d'avoir un aperçu de l'action de chaque descripteur sur le modèle. De plus, elles permettent de classer les descripteurs d'un même classifieur par ordre d'importance. Un tel classement est aussi possible directement depuis Scikit-learn avec l'*out of bag score* (OOB) qui est calculé après l'entraînement d'une forêt. Il est calculé pour chaque descripteur en mesurant l'erreur de prédiction moyenne des arbres entraînés sans le dit descripteur.

De tels classements permettent de savoir quels sont les descripteurs importants dans ces forêts aléatoires. Après une validation croisée, on s'intéresse au rang moyen et à son écart type pour chacun descripteur. Le premier donne un aperçu de l'importance générale de chaque descripteur, le second permet de savoir si pour un même modèle, le classement change beaucoup entre les classifieurs.

Cependant chaque classement ne vaut que pour un classifieur donné. Il n'est donc pas suffisant pour les sélectionner efficacement : un descripteur sans importance dans un modèle à 50 descripteurs pourrait devenir très important dans un modèle à 20 descripteurs.

Il existe deux catégories populaires de méthodes produisant une sélection de descripteurs intéressante : *filter* et *wraper* [8]. La première consiste à classer les descripteurs d'un modèle à l'aide d'une fonction traduisant leur importance et à supprimer les moins importants, la seconde à tester les performances de plusieurs sous ensembles de descripteurs et à garder le meilleur. Cette dernière catégorie de méthode donne généralement de meilleurs résultats, mais est plus gourmande en temps d ecalcul.

Dans le cadre de ce projet, on utilise des methodes *filter* et *wraper* dont on compare les performances. Pour savoir si une sélection est satisfaisante, on trace la courbe de l'évolution des métriques des modèles en fonction de leur nombre de descripteurs. On s'attend ainsi à obtenir un profil de courbes d'abord plan puis à observer une diminution de plus en plus rapide de la qualité des métriques lorsque le nombre de descripteur devient plus faible. Il est possible que l'on observe une légère amélioration de la qualité des résultats avec la diminution du nombre de descripteurs. C'est ce qu'on a pu constater dans un travail préliminaire avec Ilastik, pour lequel la méthode de sélection des descripteurs du logiciel donnait les meilleurs résultats pour un ensemble de descripteurs plus petit que celui de départ.

Méthodes *filter* :

Cet algorithme de sélection consiste à réaliser une série de validation croisées, pour lesquels à chaque étape :

1. on réalise une validation croisée du modèle,
2. on réalise un classement des descripteurs de chaque classifieur en fonction de l'OOB score, ou des Shape-Values (on testera les deux),

3. on calcule le classement moyen de chaque descripteur sur l'ensemble des classifieurs de la validation croisée,
4. on enlève le descripteur avec le rang moyen le plus faible,
5. on reprend à l'étape 1 avec l'ensemble des descripteurs restants.

Cette méthode est relativement rapide, mais très dépendante de la précision du classement des descripteurs.

Méthode *wraper* :

Si la méthode *filter* ne donne pas de résultats satisfaisant quel que soit le classement utilisé, on réalise l'algorithme suivant :

1. on réalise une validation croisée du modèle,
2. on réalise un classement des descripteurs de chaque classifieur,
3. on calcule le classement moyen sur l'ensemble des classifieurs de la validation croisée,
4. pour chacun des 4 descripteurs les moins bien classés, on construit un sous-modèles en enlevant ce descripteur,
5. on réalise une validation croisée sur chaqun de ces 4 sous-modèle et on sélectionne le meilleur (celui avec la plus faible distance ROC moyenne),
6. On reprend à l'étape 1 avec les descripteurs restants.

Si cette méthode ne permet pas de tester toutes les combinaisons de descripteurs, elle offre un bon compromis entre son efficacité et son temps de calcul. En effet, on étudie à chaque étape un petit ensemble de sous-modèles, et non l'ensemble des possibilités, ce qui prendrait beaucoup plus de temps.

3.5 Segmentation

Une fois un modèle optimal obtenu, on réalise une segmentation à partir de celui-ci sur plusieurs scan CT de patients différents pour analyser visuellement la qualité des prédictions.

Résultats

L'ensemble des résultats présentés ont été réalisés à partir des annotations réaliser entre juin et juillet 2020 sur le patient 9003. Les calculs ont été faits sur mon ordinateur personnel (Windows 10, 4 coeurs, 8 threads, 2.4GHz et 16Go de Ram) depuis l'environnement de développement Jupyter Notebook à partir du code Python qui constitue le livrable de ce stage.

4.1 Modèle de référence

Les résultats d'une première validation croisée (voir table 4.1) réalisée sur le modèle de référence sont encourageants : les erreurs de classification mesurées par les métriques sont de l'ordre de 5% et les écarts-types sont autour de 1%.

On observe une dégradation légère des résultats avec la diminution de la proportion de vaisseaux. On conservera ainsi par la suite le jeu de données à 50%.

	Modèle à 50% de vaisseaux	Modèle à 40% de vaisseaux	Modèle à 30% de vaisseaux
Dice max	0.966 ± 0.004	0.957 ± 0.004	0.937 ± 0.010
Dice 0.5	0.965 ± 0.005	0.954 ± 0.004	0.933 ± 0.012
ROC min	0.050 ± 0.006	0.052 ± 0.007	0.055 ± 0.008
ROC 0.5	0.053 ± 0.005	0.057 ± 0.006	0.068 ± 0.015
MCC max	0.932 ± 0.008	0.929 ± 0.007	0.910 ± 0.014
MCC 0.5	0.930 ± 0.009	0.924 ± 0.007	0.905 ± 0.017

TABLE 4.1 – Métriques moyennes et écarts-types obtenus pour différentes proportions de vaisseaux sanguins. Les modèles sont non optimisés et utilisent l'ensemble des descripteurs.

4.2 Optimisation

4.2.1 Hyper-paramètres

L'optimisation des hyperparamètres du modèle n'entraîne qu'une très légère augmentation de la qualité des résultats (voir table 4.2), inférieure aux écarts-types. L'étude de l'impact de chaque paramètre individuellement (voir figure 4.1) ne montre également pas d'amélioration significative, avec une exception pour la fonction de coût d'entropie qui semble significativement plus intéressant que *"gini"*. On utilise donc le modèle de référence pour la sélection des descripteurs.

4.2.2 Sélection des descripteurs

L'étude préliminaire des classements des descripteurs par Shape-Values et par out-of-bag score lors d'une validation croisée sur le modèle de référence montre des écart types pouvant monter à plus d'une dizaine de rang pour certains descripteurs. Cette instabilité traduit une forte redondance dans l'information apportée par les métriques et va poser problème pour la méthode *"filter"*.

En effet comme on peut le voir sur la figure 4.2(a), la courbe de l'évolution des métriques en fonction du nombre de descripteurs est très irrégulière quelle que soit la méthode de classement. On observe de

	Modèle de référence [2]	Modèle optimisé	Modèle final à 11 descripteurs
Dice max	0.966 ± 0.006	0.967 ± 0.006	0.969 ± 0.005
Dice 0.5	0.964 ± 0.006	0.964 ± 0.006	0.966 ± 0.005
ROC min	0.052 ± 0.010	0.051 ± 0.009	0.047 ± 0.009
ROC 0.5	0.057 ± 0.008	0.057 ± 0.009	0.054 ± 0.009
MCC max	0.931 ± 0.011	0.933 ± 0.010	0.937 ± 0.010
MCC 0.5	0.926 ± 0.011	0.928 ± 0.012	0.931 ± 0.010

TABLE 4.2 – Métriques moyennes et écarts-types obtenus à partir de trois validations croisées avec les modèles de références, optimisés et après sélection des descripteurs

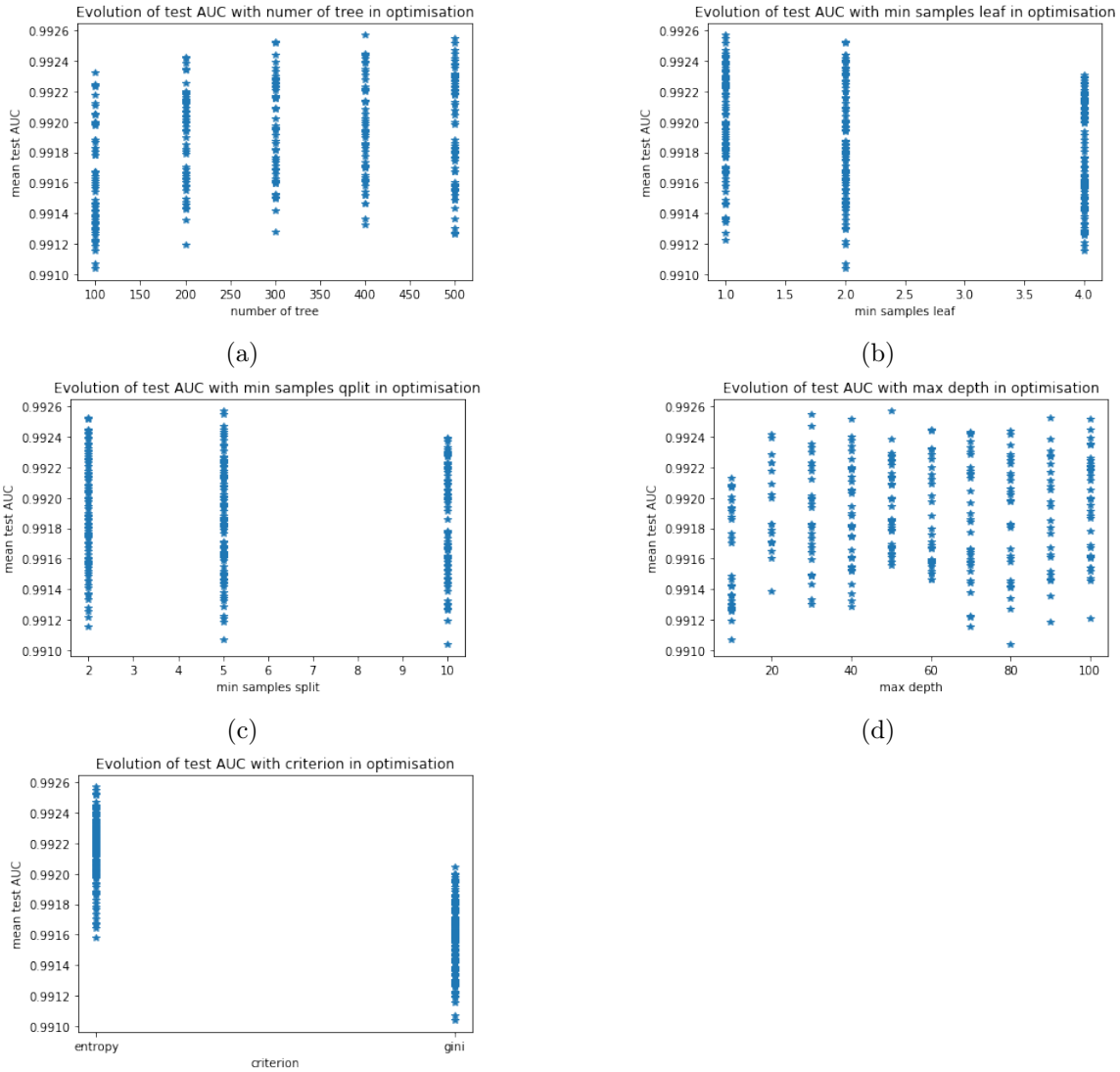
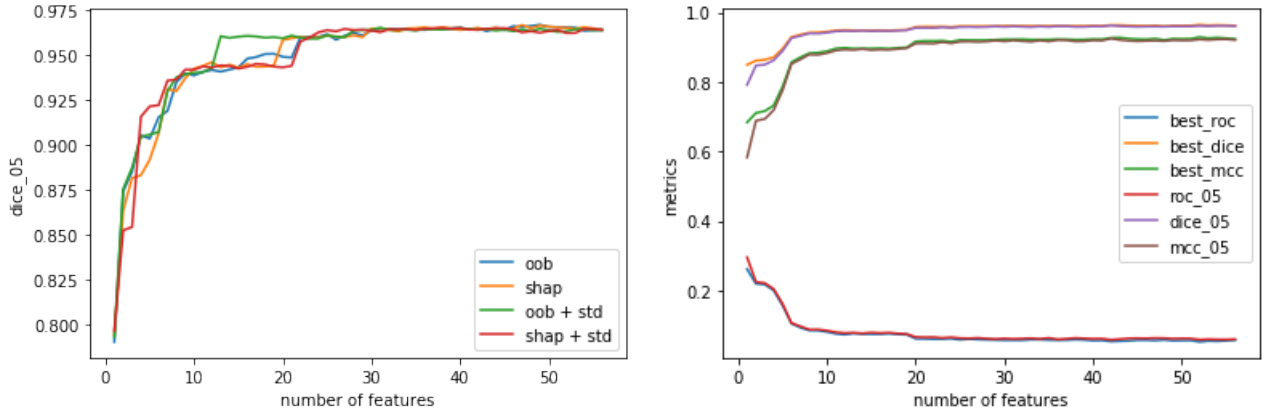


FIGURE 4.1 – Évolution de l'AUC (aire sous la courbe ROC) en fonction des hyper-paramètres : (a) Nombre d'arbres, on observe une très légère influence. (b) Nombre d'échantillons nécessaire à la séparation des noeuds, on observe une très légère influence. (c) Nombre d'échantillon de création des feuilles, on n'observe pas de réelle influence. (d) Profondeur maximale, on n'observe pas de réelle influence. (e) Critère de séparation : l'entropie semble être le meilleur critère de séparation.

plus une baisse de qualité brutale entre deux zones plus ou moins planes montre la suppression d'un descripteur important.



(a) (b)

FIGURE 4.2 – (a) Évolution de la valeur du Dice 0.5 moyen en fonction du nombre de descripteurs pour différents types de classement lors de la sélection des descripteurs par la méthode *filter*. On observe dans tout les cas des courbes irrégulière. Le modèl initiale à 56 descripteurs est le modèle de référence. (b) Évolution des valeurs moyennes des métriques en fonction du nombre de descripteurs pour le classement basé sur les Shape-Values. On observe que toute les métriques ont un comportement similaire.

Ce résultat n'est donc pas satisfaisant. De plus, quelle que soit la méthode utilisée, les six premiers descripteurs sélectionnés sont systématiquement des filtres Gaussiens. Ce manque de diversité semble contradictoire avec l'objectif de sélection de descripteurs et avec la prévalence des filtres Hessiens trouvée dans la littérature [4].

On a donc dû utiliser la méthode *wraper* pour essayer d'obtenir une sélection satisfaisante, avec comme inconvénient un temps de calcul environ 4 fois plus long. Comme aucun des classements de descripteurs utilisés avec *filter* n'a donné de résultat satisfaisant, on a choisi le plus simple pour la suite : l'out-of-bag score.

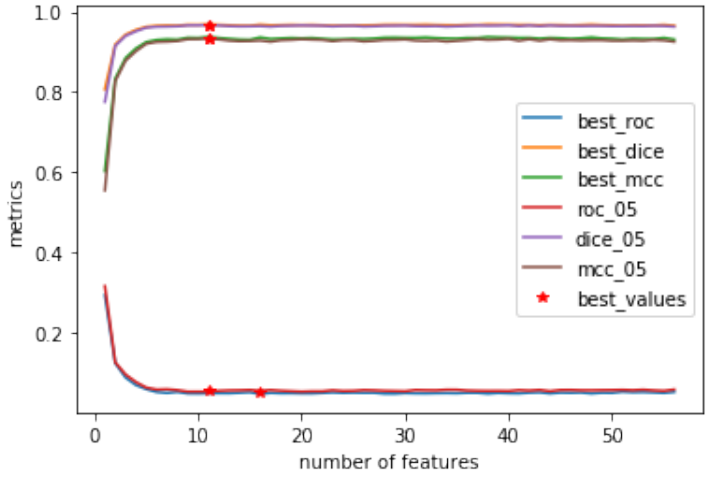
Les résultats pour la méthode *wraper* sont plus intéressants. La courbe d'évolution des métrique en fonction du nombre de descripteurs à une allure bien plus satisfaisante (figure 4.3.(a)). On observe même une amélioration de la qualité des résultats puisqu'on obtient une qualité moyenne optimale avec 11 descripteurs (table 4.2).

Cette amélioration de la qualité reste inférieur à 1%, tout en étant plus importante que celle que l'on a obtenu avec l'optimisation des hyper-paramètre. De plus cette sélection permet de diviser par 5 le nombre de descripteurs nécessaires. Cela diminue d'autant le poids en mémoire et double la rapidité d'exécution de la forêt aléatoire.

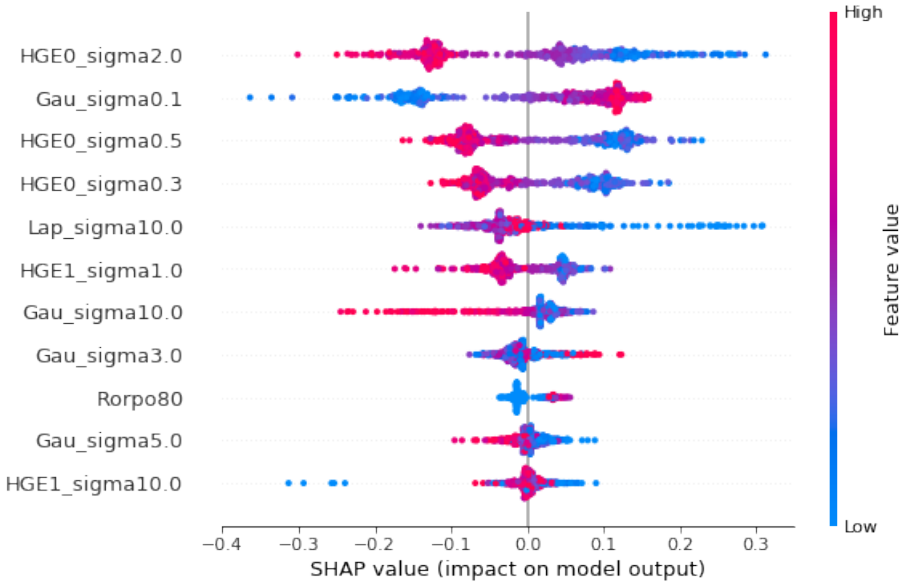
Enfin, le nombre réduit de descripteurs permet de faire une analyse visuelle de l'effet qu'ont chacun d'entre eux sur la classification des points à l'aide des Shape-Values (figure 4.3.(b)), ce qui aurait été impossible avec 56 descripteurs.

On retrouve dans le modèle à 11 descripteurs des filtres Hessien, Gaussiens, Laplacien et un filtre RORPO. La diversité de cet échantillon est représentative de l'ensemble de départ. L'analyse plus en détail des Shape-Values permet de retrouver la prévalence des filtres Hessien signalée dans la littérature.

Cette sélection des descripteurs par la méthode *wraper* est satisfaisante. Elle permet de construire un modèle optimisé pour réaliser des segmentations sur plusieurs images complètes pour une analyse visuelle.



(a)



(b)

FIGURE 4.3 – (a) Évolution des valeurs moyennes des métriques en fonction du nombre de descripteurs pour la méthode *wraper*. La courbe est bien plus régulière que dans le cas précédent. La qualité est proche de son maximum même en dessous de 10 descripteurs. Celui-ci est atteint pour un modèle de 11 descripteurs. (b) Importance relative et influence des descripteurs dans le modèle à 11 descripteurs en fonction de leurs valeurs (Shape-Values)

4.3 Segmentations

On a appliqué le classifieur optimal à 4 images CT (le scan d'entraînement 9003 ainsi que 3 autres scanners : 0015, 0005 et 0001). L'évaluation des résultats montre une segmentation visuellement satisfaisante : la plupart des vaisseaux sanguins sont correctement identifiés par rapport au reste des poumons. Cependant, on observe dans certaines zones particulières (jonction des lobes pulmonaires et lésions) une moins bonne qualité de segmentation.

Celle-ci est particulièrement compliqué pour le patient 0015 dont les poumons sont très endommagés. Certaines lésions y ont été classées comme des vaisseaux sanguins. Même sur le patient 9003 certaines régions comme les jonctions des lobes pulmonaires sont parfois identifiées incorrectement comme des vaisseaux sanguins. sanguins.

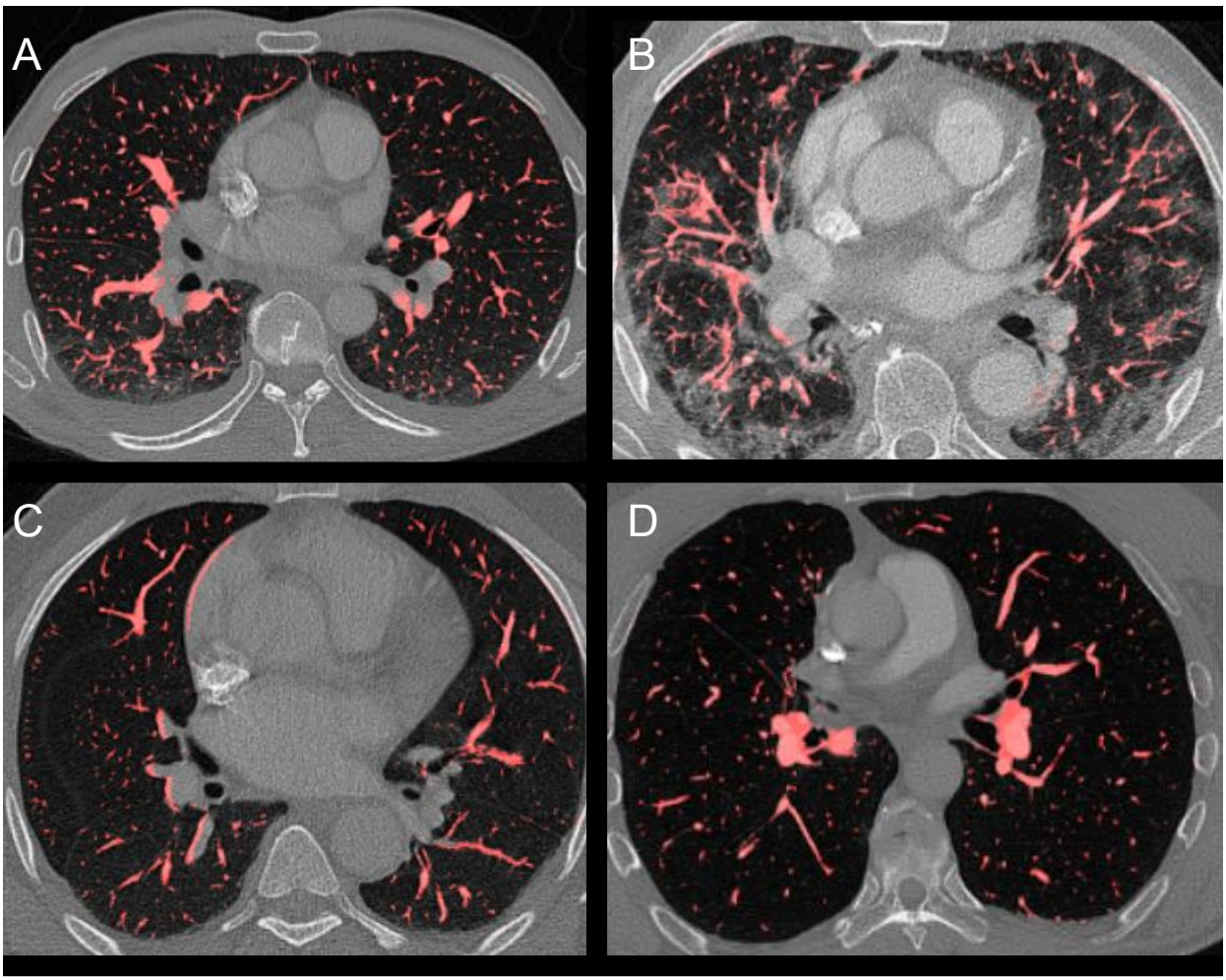


FIGURE 4.4 – Analyse visuelle des segmentations du modèle à 11 descripteurs. La segmentation, en rouge, est superposée à l'image initiale, en niveau de gris. (A) Patient d'entraînement 9003. (B) Patient 0015 avec de fortes lésions pulmonaires. (C) Patient 0005. (D) Patient 0001

Conclusion

Les objectifs de ce stage étaient de réaliser un protocole de segmentation de l'arbre vasculaire pulmonaire et de le tester avec le patient 9003. Il aura permis de produire un modèle final plus performent, interprétable, significativement moins lourd et plus rapide que le modèle de référence.

Parmi les résultats obtenus, on aura notamment montré l'importance d'un jeu de données équilibré et d'une méthode de sélection de descripteurs robuste. C'est une méthode *wraper* qui aura permis une sélection efficace

La diversité des descripteurs dans le modèle final montre l'importance d'utiliser plusieurs types de filtres différents. On pourra donc envisager, pour aller plus loin, d'utiliser plus de filtres différents.

L'analyse visuelle sur plusieurs scans montre les limites actuelles du modèle final. On observe des faux-positifs dans des zones de lésions et des zones en bordure du parenchyme, qui sont plus complexes à segmenter que le fond sombre. De plus, les plus petits vaisseaux ne sont pas systématiquement segmentés.

Il est probable que la qualité imparfaite des annotations initiales, dans lesquelles certains vaisseaux ont été oubliés, soit la principale source d'erreurs. Par ailleurs, certaines zones de tissu non vasculaire sont plus complexes à identifier. Cela peut être une source d'erreur supplémentaire car elles sont très peu représentées dans les jeux de données à notre disposition. Elle sont en effet proportionnellement bien moins présent que le fond.

La segmentation de l'AVP reste une tache compliquée chez les individus aux poumons endommagés, notamment par des maladies pulmonaires comme la Covid-19. Ce stage aura permis de construire un outil fiable pour travailler sur le sujet, mais qui peut encore évoluer. L'amélioration des annotations et le rajout d'une ou plusieurs classes supplémentaires décrivant les zones les plus complexes sont des pistes d'amélioration à explorer.

Bibliographie

[1] Teuwen, L., Geldhof, V., Pasut, A. et al. *COVID-19 : the vasculature unleashed*. Nat Rev Immunol 20, 389–391 (2020). https ://doi.org/10.1038/s41577-020-0343-0

[2] Maximilian Ackermann, M.D., Stijn E. Verleden, Ph.D., Mark Kuehnel, Ph.D., Axel Haverich, M.D., Tobias Welte, M.D., Florian Laenger, M.D., Arno Vanstapel, Ph.D., Christopher Werlein, M.D., Helge Stark, Ph.D., Alexandar Tzankov, M.D., William W. Li, M.D., Vincent W. Li, M.D., Steven J. Mentzer, M.D., and Danny Jonigk, M.D. *Pulmonary Vascular Endothelialitis, Thrombosis, and Angiogenesis in Covid-19*. July 9, 2020 N Engl J Med 2020 ; 383 :120-128 DOI : 10.1056/NEJMoa2015432

[3] *"COVID-19 Dashboard by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University (JHU)"*. ArcGIS. Johns Hopkins University. Retrieved 23 August 2020.

[4] Rina D. Rudyanto,Sjoerd Kerkstra,Eva M. van Rikxoort,Catalin Fetita,Pierre-Yves Brillet,Christophe Lefevre,Wenzhe Xue,Xiangjun Zhu,Jianming Liang,İlkay Oksüz,Devrim Ünay,Kamuran Kadipaşaoglu,Raúl San José Estépar,James C. Ross,George R. Washko et al. *Comparing algorithms for automated vessel segmentation in computed tomography scans of the lung : the VESSEL12 study*. Medical Image Analysis. Elsevier. October 2014

[5] Odyssée Merveille. *RORPO : A morphological framework for curvilinear structure analysis. Application to the filtering and segmentation of blood vessels*. Image Processing [eess.IV]. Université Paris Est, 2016.

[6] C. Sommer, C. Straehle, U. Köthe and F. A. Hamprecht, *"Ilastik : Interactive learning and segmentation toolkit"*, 2011 IEEE International Symposium on Biomedical Imaging : From Nano to Macro, Chicago, IL, 2011, pp. 230-233, doi : 10.1109/ISBI.2011.5872394.

[7] Lundberg, S.M., Erion, G., Chen, H. et al. *From local explanations to global understanding with explainable AI for trees*. Nat Mach Intell 2, 56–67 (2020). https ://doi.org/10.1038/s42256-019-0138-9

[8] L.Ladha et al. *Feature selection methods and algorithms*. International Journal on Computer Science and Engineering (IJCSE) Vol. 3 No. 5 May 2011

Notes

[1] compte tenu des outils utiliser dans l'ensemble du rapport on utilisera le point comme séparateur décimal

[2] les métriques pour ce modèle sont légèrement différentes de celle du modèle à 50% de vaisseaux du tableau précédent car on a du régénérer notre jeu de donnée entre les quelques jours qui ont séparé ces expériences.