Segmentation d'images médicales semi-supervisée par curriculum via l'inférence des boîtes englobantes

par

Bruce CYUSA MUKAMA

MÉMOIRE PRÉSENTÉ À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
COMME EXIGENCE PARTIELLE À L'OBTENTION DE LA MAÎTRISE
AVEC MÉMOIRE EN GÉNIE DE LA PRODUCTION AUTOMATISÉE
M. Sc. A.

MONTREAL, LE 30 SEPTEMBRE 2021

ECOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

PRÉSENTATION DU JURY

CE MÉMOIRE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE :

M. José Dolz, directeur de mémoire
Département de génie logiciel et des TI à l'École de technologie supérieure

M. Ismail Ben Ayed, codirecteur de mémoire
Département de génie des systèmes à l'École de technologie supérieure

M. Marco Pedersoli, président du jury
Département de génie des systèmes à l'École de technologie supérieure

M. Rafael Menelau Cruz, membre du jury
Département de génie logiciel et des TI à l'École de technologie supérieure

IL A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 08 SEPTEMBRE 2021

A L'ECOLE DE TECHNOLOGIE SUPERIEURE

REMERCIEMENTS

La réalisation des travaux de recherche présentés dans ce mémoire a été possible grâce à la direction et aux habilitations fournies par plusieurs personnes et institutions. Je tiens à leur témoigner ma profonde gratitude à travers ces quelques lignes.

Je voudrais, en premier lieu, remercier mon directeur de recherche et le codirecteur, les professeurs José Dolz et Ismail Ben Ayed, pour leurs disponibilités. Leurs conseils expérimentés, et judicieux ont guidé et nourri mes réflexions. Je remercie aussi les membres du laboratoire d'imagerie, de vision et d'intelligence artificielle (LIVIA-ÉTS) ; en particulier M. Hoel Kervadec qui m'a beaucoup aidé dans la compréhension de ses travaux et dans l'utilisation des moyens matériels disponibilisés par le laboratoire.

Je voudrais aussi remercier l'Université de Technologie de Troyes (UTT-France) ainsi que l'École de Technologie Supérieure (ÉTS-Canada), pour avoir rendu possible cette formation bi-diplômante, à travers leurs programmes d'échange international.

J'exprime également mes remerciements à mes proches, pour leur soutien constant, et pour leurs encouragements.

Segmentation d'images médicales semi-supervisée par curriculum via l'inférence des boîtes englobantes

Bruce CYUSA MUKAMA

RÉSUMÉ

Cette recherche concerne les difficultés que pose la collection d'annotations, complètes et suffisantes, pour entrainer les réseaux de neurones en segmentation sémantique d'images médicales.

Nous adoptons l'axe de recherche qui consiste à exploiter les connaissances a priori de l'organe à segmenter, pour contraindre directement les segmentations prédites. Plus précisément, les images non annotées sont exploitées en utilisant des boîtes englobantes prédites par un réseau de neurones auxiliaire. Ces boîtes inférées sont ensuite utilisées pour imposer des contraintes directes sur la taille, la position globale ainsi que la topologie de l'organe, dans les segmentations prédites correspondantes.

Nous proposons une première évaluation des performances possibles, en utilisant un réseau de neurones auxiliaire entièrement convolutif pour l'inférence des boîtes englobantes. Les résultats de nos expérimentations sur deux jeux de données de petite taille montrent une augmentation des performances, de 0.3 à 4.4% (mesurée avec l'indice de Sørensen-Dice), par rapport à la supervision totale, lorsqu'on utilise le même nombre d'annotations.

Nous recommandons, entre autres, une étude qui remplace le modèle auxiliaire par une architecture de détection classique (RPN, SSD, etc.), afin de mieux cerner l'étendue des performances de cette stratégie.

Mots-clés : réseaux neuronaux convolutifs (CNN), segmentation d'images médicales, apprentissage par curriculum, boîtes englobantes inférées, a priori(s) et contraintes.

Curriculum semi-supervised segmentation of medical images using predicted bounding boxes

Bruce CYUSA MUKAMA

ABSTRACT

Although deep learning has nowadays become the de facto solution for medical image segmentation, one the limitations that remain is the need of large labelled dataset for training, which are hard to obtain. This research proposes a new semi-supervised training strategy, to alleviate the annotation burden.

Our focus consists in exploiting domain knowledge, about the organ of interest, to constrain directly the predicted segmentations. Specifically, we exploit unlabeled images, in a semi-supervised curriculum context, by using an auxiliary neural network to predict the bounding boxes of the organs that unlabeled images may contain. These predicted bounding boxes are then used to constrain the size, the global location and the topology of the corresponding predicted segmentations.

We present extensive experiments to showcase the benefits of the proposed methodology. In particular, the reported results demonstrate that by including our constrained formulation, a performance gain ranging from 0.3 to 4.4% is obtained compared to the fully supervised counterpart (in terms of Sorensen-Dice coefficient), when the same amount of annotations is used. These results are consistent across two different medical segmentation datasets, i.e., left-ventricle and prostate, which demonstrates the generalizability of our approach.

Among our recommendations, we advocate a more extensive study that uses a standard detection architecture (region based convolutional neural networks, single-shot detectors, etc.), to better access the extent of this supervision's performance.

Keywords: convolutional neural networks (CNN), medical image segmentation, curriculum learning, predicted bounding boxes, priors and constraints.

TABLE DES MATIÈRES

Page

LISTE DES TABLEAUX

Page

LISTE DES FIGURES

Page

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

ACDC	Automated Cardiac Diagnosis Challenge
CNN	Convolutional neural network
CRF	Conditional random field
CT	Computer tomography
DSC	Sørensen-Dice coefficient
GPU	Graphics Processing Unit
ENet	Efficient Net
EMA	Exponential moving average
HD	Hausdorff distance
IoU	Intersection over Union
IRM	Imagerie par resonance magnétique
MICCAI	Medical Image Computing and Computer Assisted Intervention Society
PRELU	Parametric Rectified Linear Unit
PROMISE 12	Prostate Magnetic resonance Image Segmentation 2012
RPN	Region proposal network
R-FCN	Region based fully convolutional network
SSD	Single shot detector
SVM	Support vector machine
TDM	Tomodensitométrie

LISTE DES SYMBOLES ET UNITÉS DE MESURE

h	le nombre d’heures
m	le nombre de minutes
s	le nombre de secondes
ms	le nombre de millisecondes
%	le pourcentage

INTRODUCTION

La segmentation sémantique d'images médicales est l'étape d'analyse d'images qui précède l'extraction d'informations ciblées par organe. Elle consiste à isoler des régions d'intérêt dans une image, suivant leurs différences conceptuelles : pour comprendre le contenu, les objets dans l'image sont dissociés sans différencier leurs instances. Cela se fait en déterminant les contours des régions ou les pixels en leurs seins.

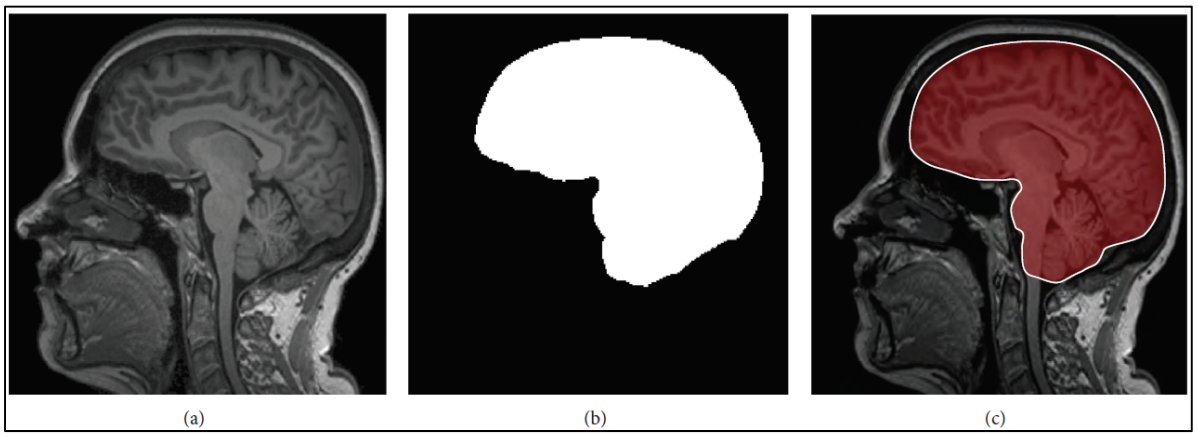

Figure 0.1.1 Illustration de la segmentation: (a) le scan d'un crâne humain, (b) le masque de segmentation du cerveau, (c) la superposition et le contour
Tirée de Despotović, Goossens, & Philips (2015, p. 10)

En ce qui concerne l'utilité pratique de cette technique, cela peut servir à mesurer des organes afin de planifier des interventions, calculer des dosages, détecter des anomalies, etc.

Actuellement, les meilleures performances s'obtiennent en utilisant des réseaux neuronaux convolutifs entrainés avec des grandes quantités d'exemples annotés par des humains, comme l'indiquent Dolz, Desrosiers, & Ayed (2018) ; et Chen, Dou, Yu, Qin, & Heng (2018). L'apprentissage profond aide à simplifier l'extraction des caractéristiques d'organes, et à automatiser le paramétrage de la chaîne de traitement, dans un temps beaucoup plus court, et de bout en bout.

Depuis l'utilisation de l'accélération matérielle (en particulier) par Krizhevsky, Sutskever, & Hinton (2017), les réseaux de neurones convolutifs profonds donnent les meilleures performances dans plusieurs domaines de la vision artificielle, comme rapporté par Minaee et al. (2020).

Le nœud de la problématique abordée dans cette recherche est la contradiction entre la quantité d'annotations nécessaires, pour atteindre les meilleures performances, et le manque d'une main d'œuvre qualifiée et disponible pour les créer.

En fait, l'annotation d'images médicales nécessitent une main d'œuvre experte (radiologues, etc.). Cette main d'œuvre est rare parce ce type de qualification nécessite des longues formations, contrairement à l'annotation d'images des scènes du quotidien qui est à la portée de la majorité d'adultes. De plus, les observations des experts peuvent varier et l'annotation d'un très grand nombre d'images reste une tâche très chronophage mais moins importante que le diagnostic et le soin de patients.

Pour diminuer la quantité d'images nécessaires, plusieurs axes de recherche ont été explorés, en particulier l'utilisation d'images non annotées (en semi-supervision) et l'utilisation d'annotations moins précises (en supervision faible).

En ce qui concerne la semi-supervision, cela consiste à utiliser une plus petite quantité de données complètement annotées ainsi qu'une grande quantité de données non annotées. Cet axe a suscité beaucoup d'intérêt, mais le mélange entre les deux axes n'est pas très exploré.

En ce qui concerne l'utilisation des annotations faibles, au lieu d'annoter chaque pixel d'un organe, seuls quelques pixels de l'organe sont annotés (par exemple). Comme le montre la figure sur la page qui suit, il existe plusieurs types d'annotations faibles. Par ailleurs, les propriétés de l'organe (taille, localisation, etc.) peuvent aussi être utilisé comme des faibles annotations. Il est connu que ces connaissances (a priori) du domaine médical peuvent être imposées avec des contraintes, pour améliorer les performances.

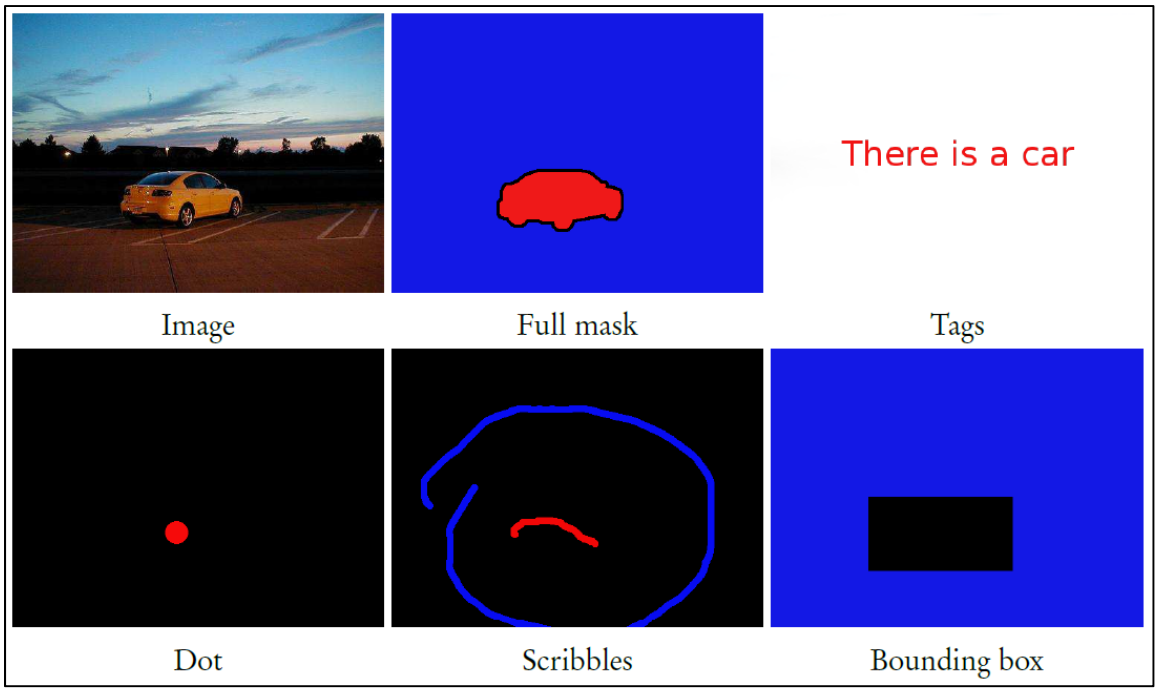

Figure 0.1.2 Illustration de quelques types d'annotations utilisées en semi supervision ou en supervision faible, où le bleu représente le fond de l'image
Tirée de Kervadec (2021, p. 15)

La segmentation semi-supervisée par curriculum mixe ces deux axes de recherche, mais l'état de l'art n'utilise pas le nouveau formalisme d'optimisation sous contrainte, développé par Kervadec (2021). Par ailleurs, seul un a priori global est utilisé.

De plus, selon Kervadec, Dolz, Wang, Granger, & Ayed (2020b), il est possible de se rapprocher considérablement des performances de la supervision totale, en utilisant plusieurs a priori ainsi que ce nouveau formalisme d'optimisation sous contraintes, pour contraindre directement les propriétés locales, globales et topologiques des segmentations prédites. Les annotations faibles utilisées sont des boîtes englobantes serrées, créées par des experts humains (non-inférées) : leur intervention est encore requise !

Alors que ces contraintes ont été utilisées dans un contexte de faible supervision, en notre connaissance, la possibilité d'utiliser des boîtes englobantes inférées, pour appliquer ces contraintes en semi-supervision, n'a pas encore été étudiée.

Il sied de noter que l'utilisation des boîtes englobantes inférées a déjà été explorée dans plusieurs contextes de supervision, en particulier par Kervadec et al. (2020b) ; Papandreou, Chen, Murphy, & Yuille (2015) ; Hsu, Hsu, Tsai, Lin, & Chuang (2019). Toutefois, une attention particulière doit être portée sur les différences majeures, qu'il existe entre les propositions de cette recherche et celles des autres travaux :

- les boîtes englobantes que nous utilisons sont inférées par un réseau auxiliaire ;
- ces boîtes servent à contraindre les prédictions du réseau principal, directement !

Eu égard à ce qui précède, l'exploration d'une telle stratégie d'apprentissage est importante dans la mesure où elle pourrait améliorer les performances actuelles, en alliant plusieurs types d'a priori (globaux, locaux, topologiques), dans un contexte d'apprentissage semi-supervisée par curriculum.

Les retombées académiques sont les conclusions et les recommandations tirées de cette recherche, pour guider des études connexes. Le code source est également disponibilisé. Par ailleurs, la réduction du nombre d'annotations nécessaires réduit les coûts en temps, en charge de travail et en argent, liés au temps que passent les experts médicaux à collectionner, organiser et annoter des jeux de données.

Bien que la détection d'objet soit le cadre propice à l'inférence des boîtes englobantes, nous considérons qu'un réseau de neurones entièrement convolutif peut être utilisé pour le modèle auxiliaire (reformuler le problème comme une segmentation), afin de donner un aperçu des performances. Par conséquent, les architecture de détection d'objets ne sont pas abordées.

Par ailleurs, même si deux réseaux de neurones sont utilisés, il ne s'agit pas d'apprentissage multitâche. Par conséquent, pour la comparaison aux méthodes préexistantes, nous excluons ce cadre de supervision et nous nous limitons aux méthodes qui utilisent un seul réseau auxiliaire, distinct.

Ainsi donc, nous explorons une stratégie d'apprentissage par curriculum qui utilise un modèle auxiliaire afin d'inférer des telles annotations, pour ensuite les utiliser dans l'expression des contraintes que doivent satisfaire les segmentations prédites.

Plus précisément :

- nous formalisons une stratégie d'apprentissage par curriculum, avec des boîtes englobantes inférées par un réseau de neurone entièrement convolutif ;
- nous évaluons les apports des composantes de la fonction de coût et la généralisation de la méthode par rapport aux architectures de segmentation usuelles ;
- nous évaluons empiriquement les performances lorsque la quantité des données annotées utilisées est variée, et nous comparons notre méthode à la supervision totale et à d'autres méthodes de semi-supervision (actuelles) ;
- nous démontrons les capacités de généralisation de cette méthode en étendant les expérimentations sur deux jeux de données, composés d'images médicales, et publiquement accessibles.

Ce document est organisé en cinq grandes parties principales. La revue de la littérature présente les fondements de l'apprentissage profond pour la segmentation d'images médicales, et les méthodes retenues pour l'évaluation comparative.

La méthodologie présente le formalisme utilisé pour entrainer notre modèle, le chapitre suivant détaille les protocoles et présente les résultats, pour chacune de nos expérimentations. La dernière partie présente l'interprétation des résultats, les conclusions tirées, et les recommandations de cette recherche.

CHAPITRE 1

RÉVUE DE LA LITTÉRATURE

1.1 Les notions préliminaires

Dans le but de définir les concepts et les notations utilisées dans ce document, les deux sections qui suivent présentent les particularités des images médicales ; les raisons qui poussent à l'utilisation de l'apprentissage profond ; ainsi que les techniques fondamentales qui aident à entrainer les réseaux de neurones (pour la segmentation sémantique).

1.1.1 Les particularités des images médicales

Les images médicales (scans par tomodensitométrie (TDM), par résonnance magnétique (IRM), par ultrasons, etc.) sont obtenues en utilisant des appareils cliniques qui fonctionnent suivant un principe commun : interagir avec la matière et catégoriser le type de tissue selon la façon dont la matière réagit. Les données que récoltent ces appareils peuvent être représentées dans un type à quatre dimensions : les données acquises sont souvent discrétisées dans l'espace et dans le temps.

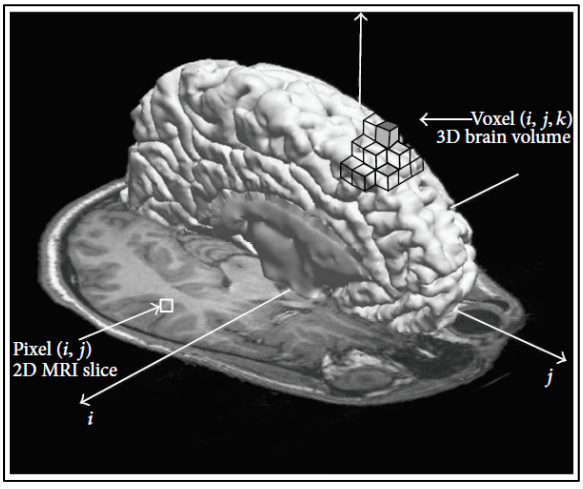

Figure 1.1 Illustration d'un voxel
Tirée de Despotović et al. (2015, p. 2)

Un voxel désigne un élément ponctuel du volume discrétisé (le scan). Une trame désigne tous les points acquis en un même instant, et une tranche désigne tous les points contenus dans un même plan (perpendiculaire à un axe donné).

En ce qui concerne l'aspect visuel d'images médicales, il dépend de la résolution spatiale mais surtout de la technologie utilisée. Pour la résolution spatiale, plusieurs scalaires caractérisent les espacements entre les tranches, entre les points dans une tranche, etc. Chaque technologie d'acquisition possède ses propres paramètres. Ces derniers sont contrôlés et enregistrés mais ils ne sont pas constants : ils peuvent varier d'un scan à l'autre, ou d'une clinique à l'autre.

Cette variabilité des protocoles d'acquisition explique en partie les difficultés que pose la conception des descripteurs classiques pour les organes humains : ils seraient spécifiques à un protocole, sinon le paramétrage du système prendrait trop de temps.

Mis à part les problèmes liés à l'acquisition, il existe un grand écart entre l'aspect visuel amorphe d'objets organiques et l'aspect cristallin d'objets industriels. Au sein d'une même catégorie d'objets industriels, les aspects visuels sont réguliers alors que l'aspect visuel de la matière organique est très irrégulier.

Comme mentionné précédemment, les méthodes par apprentissage profond aident à simplifier l'extraction des caractéristiques d'organes, et à automatiser le paramétrage de la chaîne de traitement, dans un temps beaucoup plus court, et de bout en bout. Leur utilisation dans les domaines médicaux vise à automatiser les tâches liées au traitement de scans, entre autres, comme évoqué par Kervadec (2021, p. xxvii). L'objectif est de réduire la charge des cliniciens, pour leur permettre de s'occuper des tâches plus obligeantes : le soin de patients.

Avant d'aborder les travaux pertinents par rapport à notre problématique, la section qui suit présente les bases de l'apprentissage profond pour la segmentation sémantique d'images médicales avec des réseaux neuronaux (entièrement) convolutifs.

1.1.2 L'apprentissage automatique

Pour aborder les principes de la segmentation avec des réseaux de neurones entièrement convolutifs, il sied de commencer par expliquer comment fonctionne l'opération de convolution discrète :

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) X_n(x-s\,, y-t) \tag{1.1}$$

Où w est le filtre de dimension (a,b), X_n est l'image filtrée, et $g(x,y)$ est la valeur du pixel aux coordonnées (x,y) dans la sortie produite.

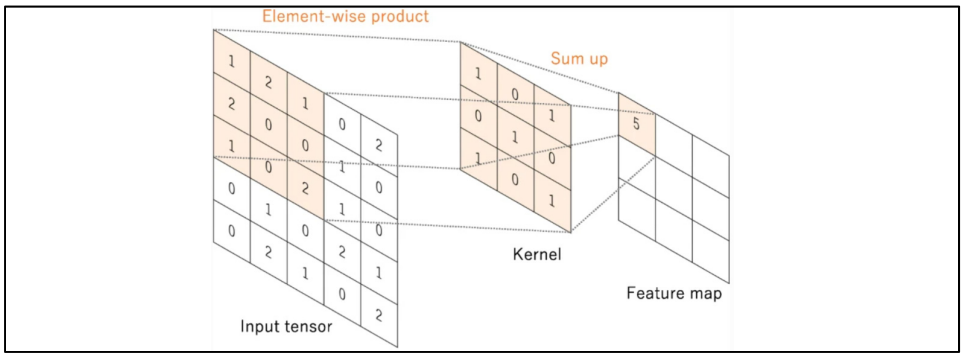

Figure 1.2 Illustration de l'opération de convolution, où le *kernel* dénote w
Tirée de Yamashita, Nishio, Do & Togashi (2018, p. 4)

Comme l'illustre la figure ci-dessus, l'opération convolution discrète calcule une sortie bidimensionnelle pour les images. Pour obtenir tous les pixels de la carte produite, le centre du filtre de convolution est glissé, tour à tour, sur les pixels de l'image en entrée. Chaque élément de cette sortie est la valeur d'une somme pondérée, calculée en centrant le filtre sur la position de l'élément, et en pondérant le voisinage avec les poids de la fenêtre (telle qu'elle se superpose).

Pour les méthodes de vision classique, cette opération trouve son utilité dans des usages variés, dont on peut citer l'implémentation d'opérations unitaires de filtrage : les filtres de

Sobel (détection de contours), les filtres gaussiens (filtrer de bruit), etc. Ici, les poids du filtre utilisé sont connus avant son insertion dans une étape de la chaîne de traitement.

Pour les méthodes de vision par apprentissage automatique, les usagers ne choisissent que les caractéristiques (les hyperparamètres) des filtres, et des algorithmes d'optimisation déterminent les valeurs idéales pour les poids (les paramètres) des filtres : c'est de là que vient l'appellation « apprentissage automatique ».

L'apprentissage profond est l'application des principes de l'apprentissage automatique à l'entrainement des réseaux de neurones profonds (ayant plus de trois couches).
Pour formaliser les principes de l'apprentissage profond, nous adaptons une notation unique qui est suivie partout dans ce document :

- $\{c_i | i = 1, \dots, C\}$ désigne l'ensemble d'organes que le traitement $\theta(\boldsymbol{X}_n)$ doit apprendre à segmenter ;
- $D_X = \{\boldsymbol{X}_n \in \mathbb{R}^{H \times W} | \, n = 1, \dots, N_1 + N_2\}$ désigne l'ensemble de toutes les tranches $\boldsymbol{X}_n$ disponibles pour l'entrainement. $H \times W$ désigne les dimensions des tranches ;
    - même si nous segmentons des volumes (scans), les prédictions sont faites tranche par tranche et réunies ensembles pour former la prédiction du volume, dans cette recherche ;
- $D_Y = \{\boldsymbol{Y}_n \in \{0,1\}^{H \times W \times C} | \, n = 1, \dots, N_1\}$ désigne les annotations disponibles pour l'entrainement, et $\boldsymbol{Y}_n^{(h,w,c)} \in \{0,1\}$ désigne l'annotation de la classe c pour le pixel aux coordonnées (h, w), dans la tranche désignée par X_n ;
- $\boldsymbol{S}_n \in \mathbb{R}^{H \times W \times C}$ désigne la normalisation des sorties du traitement, sous forme des probabilités, grâce à la fonction « *softmax* » (Bishop, 2006, p. 198) ;
- $\widehat{\boldsymbol{Y}}_n \in \mathbb{R}^{H \times W \times C}$ désigne l'estimation du masque binaire : la décision du traitement.

Il sied de noter que :

$$\widehat{\boldsymbol{Y}}_n^{(h,w,c*)} = 1 \Leftrightarrow c^* = \underset{c}{\operatorname{argmax}} \boldsymbol{S}_n{}^{(h,w,c)} \tag{1.2}$$

$$avec \quad \boldsymbol{S}_n{}^{(h,w,c)} = \frac{exp\left(\theta\left(\boldsymbol{X}_n^{(h,w,c)}\right)\right)}{\sum_{j=1}^{C} exp\left(\theta\left(\boldsymbol{X}_n^{(h,w,j)}\right)\right)} \tag{1.3}$$

En interprétant les sorties normalisées $\boldsymbol{S}_n$ comme des probabilités, de façon similaire à la classification, comme l'indique Bishop (2006, p. 164), on définit l'apprentissage de la segmentation comme un problème de minimisation de la dissimilarité entre la distribution définie par les annotations $(\boldsymbol{Y}_n)$ et celle des masques produits par $(\boldsymbol{S}_n)$.

Cette dissimilarité étant mesurée avec la divergence de Kullback-Leibler, entre autres, on peut caractériser le traitement optimal par :

$$\theta^* = arg\min_{\theta} -\sum_{n=1}^{N_1} \sum_{H\times W\times C} \boldsymbol{Y}_n^{(h,w,c)}\, log\big(\boldsymbol{S}_n{}^{(h,w,c)}\big) + f(\boldsymbol{Y}_n) \tag{1.4}$$

Le premier terme est l'entropie croisée $(\mathcal{L}_{ce})$ et c'est elle seule qui est minimisée, étant donné que le deuxième terme $f(\boldsymbol{Y}_n)$ est constant par rapport aux sorties du traitement $\theta(\boldsymbol{X}_n)$.

Plus précisément, lorsque $\theta(\boldsymbol{X}_n)$ est un réseau de neurones convolutif, il s'agit alors d'une longue suite composée de filtres, d'où la profondeur. C'est une représentation paramétrique, idéalement différentiable, de la chaîne de traitement d'images : filtrage, extraction des caractéristiques, décision, etc. La mise à jour des paramètres se fait itérativement grâce à deux règles principales :

$$\frac{\partial \mathcal{L}_{ce}}{\partial w_{(k)}} = \frac{\partial \mathcal{L}_{ce}}{\partial w'_{(k)}} \times \frac{\partial w'_{(k)}}{\partial w_{(k)}} \tag{1.5}$$

$$w_{(k+1)} = w_{(k)} - \alpha \left[\frac{\partial \mathcal{L}_{ce}}{\partial w_{(k)}}\right] \tag{1.6}$$

Où $w_{(k+1)}$ désigne la prochaine valeur d'un paramètre (un poids) d'un neurone dans le réseau, une unité de calcul dont la sortie est notée w'. L'hyperparamètre α désigne le taux d'apprentissage : le pas des itérations.

La première règle est connue sous plusieurs appellations (rétropropagation, règle de chaîne, théorème de dérivation des fonctions composées, etc.). Elle sert à déterminer le taux de variation de l'erreur suivant chaque paramètre du réseau.

La deuxième règle est l'optimisation suivant la descente du gradient, elle sert à déterminer dans quel sens il faut modifier les paramètres du réseau.

En pratique, la mise à jour ne se calcule pas en utilisant toutes les tranches. Selon Goodfellow, Bengio, & Courville (2016, p. 146), pour diminuer les coûts des calculs, à chaque itération, un lot composé de B tranches et tiré par hasard est utilisé pour évaluer la valeur de la fonction de coût : c'est le principe de l'algorithme du gradient stochastique.

De façon moins simpliste, un réseau de neurones n'est pas composé que de filtres de convolution. Il contient aussi des opérations de sous-échantillonnage pour diminuer les coûts des calculs, ainsi que des fonctions d'activation non linéaires pour étendre les types de fonctions représentables.

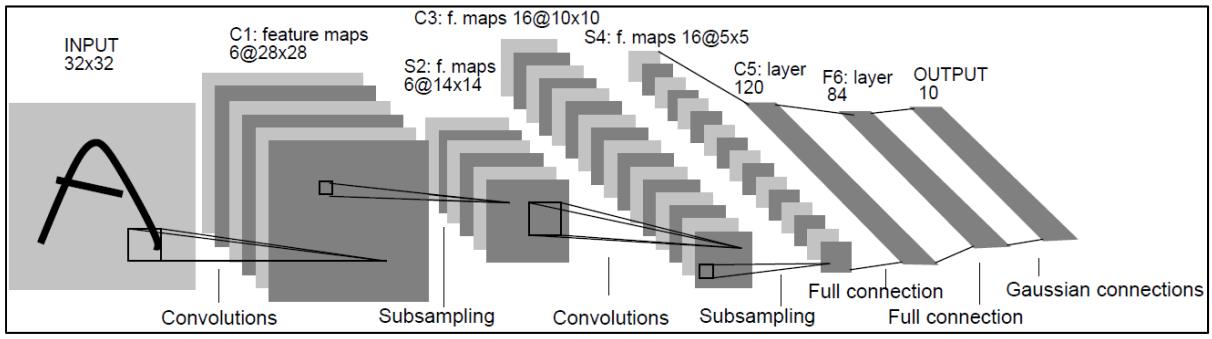

Figure 1.3 Illustration des modules architecturaux (couches et/ou opérations) avec LeNet, Tirée de Lecun, Bottou, Bengio, & Haffner (1998, p. 7)

1.1.3 Les limites de la supervision totale

Cette section résume les grandes lignes chronologiques qui ont conduit à l'état de l'art actuel en segmentation sémantique par apprentissage profond supervisé, et met en évidence les limites pertinentes des techniques de la supervision totale, en imagerie médicale.

En fait, l'une des avancées notables en segmentation sémantique a été l'avènement des réseaux neuronaux entièrement convolutifs, publiés par Long, Shelhamer, & Darrell (2015). Auparavant, pour segmenter une image, une région (aire) tiré de l'image (*patches*) était introduite dans un réseau de neurones pour classifier le pixel au centre de la région : il fallait donc effectuer les opérations du réseau de neurones autant de fois qu'il y avait des pixels (ou des groupements). Les réseaux neuronaux entièrement convolutifs diminuent les coûts des calculs en inférant les classes correspondant à tous les pixels de l'image, en une fois (un seul passage). Par ailleurs, ces architectures prennent en entrées des images de taille variable, et infèrent un masque de segmentation ayant la même taille.

Pour la segmentation par tranche (par pixels), les architectures utilisées en imagerie médicale sont (le plus souvent) des variantes de l'architecture UNet, publiée par Ronneberger, Fischer, & Brox (2015). Cette dernière est aussi entièrement convolutive et conçue pour tenir compte des détails de l'image, à plusieurs échelles, afin d'inférer des masques de segmentation plus détaillés. Pour la segmentation par volume (par voxels), il existe aussi des variantes et adaptations de UNet : 3D-UNet par Çiçek, Abdulkadir, Lienkamp, Brox, & Ronneberger, (2016) ; VNet par Milletari, Navab, & Ahmadi (2016) ; etc.

Toutefois, comme évoqué par Jurdi, Petitjean, Honeine, Cheplygina, & Abdallah (2020, p. 2), il est connu que UNet peut produire des masques de segmentation anatomiquement aberrants, i.e. avec des trous ou des mauvaises bordures, lorsqu'il est entraîné uniquement avec les masques d'annotation. Pour pallier ces lacunes, plusieurs travaux ont recouru à l'utilisation des fonctions de coûts qui expriment aussi les connaissances (a priori) du

domaine médical : la taille de l'organe, la forme de l'organe, la localisation de l'organe, et consorts, telles que revue par Jurdi et al. (2020).

D'autre part, même si l'entropie croisée (voir section 1.1.2) est la fonction de coût basique pour la segmentation sémantique, elle n'est pas toujours adaptée aux particularités des données (distributions, topologies, circonscriptions). Pour gérer les répartitions asymétriques des classes par rapport aux pixels, plusieurs variantes de l'entropie croisée ont été suggérées, en particulier l'entropie croisée équilibrée pour équilibrer les contributions des exemples négatifs et des exemples positifs, et la *Focal Loss* pour prioriser la contribution des exemples mal segmentés, durant l'apprentissage, comme l'indique Jadon (2020, p. 2).

D'autre part, il peut arriver que l'on désire optimiser directement une métrique de performance. Entre autres, la *Dice Loss* pour optimiser directement la superposition entre les régions d'intérêt inférées et les annotations ; et des variantes convexes de la distance de Hausdorff pour optimiser la superposition entre les frontières d'intérêt et les annotations, comme indiqué par Jadon (2020, pp. 2, 3).

Par ailleurs, comme évoqué dans l'introduction, ces techniques se heurtent souvent aux difficultés que posent la collection d'images médicales suffisamment annotées : la supervision totale est souvent irréaliste ou inefficace. Il sied de noter également que l'entrainement (voire même l'utilisation) des réseaux neuronaux convolutifs nécessite des calculateurs adaptés (mémoire vive, nombre de *core*(s), etc.), tels que les processeurs graphiques, pour accélérer matériellement le processus d'apprentissage (ou l'inférence). Ce matériel dédié est couteux, et la disponibilité des meilleurs équipements n'est pas assurée dans une clinique médicale.

1.2 Les méthodes semi-supervisées

La semi-supervision est le contexte d'apprentissage où l'algorithme exploite une petite quantité de données annotées ainsi qu'une grande quantité de données non annotées.

Du point de vue de la réduction de la charge liée à l'annotation d'images médicales, ce contexte de supervision est pertinent dans le sens où la semi-supervision peut se rapprocher des performances de la supervision totale (utilisant beaucoup d'annotations), tout en nécessitant moins d'images annotées.

Dans certains cas, la semi-supervision considère que les caractéristiques communes entre les données non annotées et les données annotées font qu'il existe des représentations où cela se traduit par une proximité : un réseau de neurones donnera des vecteurs similaires pour des images de la même catégorie, s'il extrait les caractéristiques pertinentes.

Cette considération est souvent dénommée *« smoothness assumption »* (Kostopoulos, Karlos, Kotsiantis, & Ragos, 2018, p. 14). Elle permet de traiter les données suivant les sous-ensembles de proximité qui apparaissent dans l'espace de représentions : le souhait est que les frontières de décisions soient moins brusques par rapport aux groupements.

Dans d'autres cas, la semi-supervision se base sur les connaissances du domaine au sujet de l'organe d'intérêt (a priori de taille, de proportion, de forme, etc.). Elle utilise ces caractéristiques globales pour vérifier la véracité des segmentations prédites, afin de superviser l'apprentissage sur le grand ensemble d'images non annotées.

Comme cela a été mentionné par Kervadec (2021, p. xxix), l'exploitation des connaissances du domaine médical est très pertinente par rapport à la problématique parce que l'acquisition d'images médicale est contrôlée et rapportée : les grandeurs mesurées (tailles, etc.) peuvent être aisément calculées à partir des métadonnées enregistrées avec les images, pour un grand nombre d'images.

Les deux sections qui suivent présentent les méthodes pertinentes par rapport à notre problématique, qui sont tirées de ces deux grands axes de recherche. La troisième section présente les concepts d'optimisation sous contraintes utilisés dans notre méthode, et la dernière section fait un court résumé de cette revue de littérature.

1.2.1 La génération des pseudo-étiquettes

Les méthodes basées sur les pseudo-étiquettes consistent à utiliser un réseau de neurones pour proposer des masques d'annotation sur le grand ensemble d'images non annotées. Cette revue de littérature présente les techniques souvent utilisées pour générer des pseudo-étiquettes : le *self-training*, le *co-training* et le *self-ensembling.* Ces deux dernières ne sont pas directement liées à notre proposition, mais elles traitent la même problématique.

1.2.1.1 Le self-training

Pour ce type de semi-supervision, le réseau de neurones est entrainé jusqu'à atteindre une certaine performance sur le petit ensemble annoté, ensuite deux étapes sont alternées :

- raffinement des masques proposés par le réseau, ses paramètres étant fixés ;
- utilisation des masques raffinés, pour corriger les paramètres du réseau de neurones.

Dans la littérature, plusieurs méthodes de raffinement ont été proposées. Les travaux de W. Bai et al. (*Medical image computing and computer-assisted intervention -- MICCAI*, 2017, p. 255), ont utilisé un modèle graphique, les champs aléatoires conditionnels (CRF), pour raffiner les segmentations du ventricule droit ; et les travaux de Hung, Tsai, Liou, Lin, & Yang (2018) ont utilisé l'apprentissage adverse pour le raffinement. En particulier, un premier réseau de neurones est utilisé pour segmenter les images, et un autre est utilisé pour différencier les pixels de vrais masques de segmentation et ceux des propositions du réseau. Selon Hung et al., le réseau discriminateur effectue un raffinement lorsqu'il indique les probabilités qu'un pixel ait été correctement segmenté. Il l'indique grâce à sa sortie, une carte de mêmes dimensions que son entrée. Le réseau de segmentation s'améliore en apprenant à tromper le discriminateur.

Dans la mesure où cette méthode a été retenue pour la comparaison aux méthodes pertinentes, nous présentons aussi sa formalisation :

- soient $D(\boldsymbol{S}_n)$ les sorties du réseau discriminateur, lorsqu'il prend en entrée les prédictions du réseau de segmentation ;
- soient $D(\boldsymbol{Y}_n)$ les sorties du réseau discriminateur pour les masques vrais.

La fonction de coût utilisée pour superviser l'apprentissage du discriminateur est l'entropie croisée binaire :

$$\mathcal{L}_D = -\sum_{h,w}(1 - \tilde{y}_n)log\big(1 - D(\boldsymbol{S}_n)^{(h,w)}\big) + \tilde{y}_n log\big(D(\boldsymbol{Y}_n)^{(h,w)}\big) \tag{1.7}$$

Adaptée de Hung et al. (2018).

Où $\tilde{y}_n$ est égal à 1 si le pixel est tiré d'un masque vrai (non prédit), sinon zéro.

Pour la supervision du réseau de segmentation, trois fonctions de coûts sont utilisées. La première s'applique sur le petit ensemble d'images annotées, c'est l'entropie croisée $\mathcal{L}_{ce}$. La deuxième fonction de coût s'applique sur les deux ensembles (c'est la fonction de coût adverse $\mathcal{L}_{adv}$), et la dernière ne s'applique qu'à l'ensemble sans annotations (c'est l'entropie croisée masquée) :

$$\mathcal{L}_{seg} = \sum_{n=1}^{N_1} \mathcal{L}_{ce} + \lambda_{adv_1}\mathcal{L}_{adv} + \sum_{n=N_1+1}^{N_2} \lambda_{semi}\mathcal{L}_{semi} + \lambda_{adv_2}\mathcal{L}_{adv} \tag{1.8}$$

$$\mathcal{L}_{adv} = -\sum_{h,w} log\big(D(\boldsymbol{S}_n)^{(h,w)}\big) \tag{1.9}$$

$$\mathcal{L}_{semi} = \sum_{h,w}\sum_{c\epsilon C} \mathbb{1}\big(D(\boldsymbol{S}_n)^{(h,w)} > T_{semi}\big)\hat{Y}_n^{(h,w,c)} log\big(\boldsymbol{S}_n{}^{(h,w,c)}\big) \tag{1.10}$$

Adaptées de Hung et al. (2018).

Où T_{semi} est le seuil qui permet d'éliminer les prédictions qui ne trompent pas le discriminateur ; $\mathbb{1}$ est la fonction indicatrice : elle est égale à 1 si le test est vrai, à zéro sinon.

1.2.1.2 Le co-training

Pour ce type de semi-supervision, l'exploitation des données non-annotées est faite en renforçant la cohérence des prédictions produites par deux modèles distincts. Ces modèles sont indépendamment entrainés, suivant des vues (descriptions des données) indépendantes. L'intuition est que même si les vues sont différentes, les prédictions doivent être cohérentes (suffisamment similaires). Ainsi donc, les divergences sont pénalisées pour l'entrainement avec les données non annotées.

Dans la littérature, les travaux de Zhou et al. (2019b) utilisent des tranches annotées provenant de trois axes de projection, pour entrainer le premier réseau de neurones. Ce dernier est ensuite utilisé pour créer des pseudo-étiquettes sur l'ensemble non annoté, afin d'entrainer le deuxième réseau (en semi-supervision). En reconstruisant les trois volumes, un pour chaque axe, le consensus est trouvé avec un vote de majorité à l'échelle des voxels.

Pour les travaux de Peng, Estrada, Pedersoli, & Desrosiers (2020), deux modèles sont entrainés sur deux ensembles annotées disjoints, et un ensemble non annoté commun. Le consensus est renforcé en minimisant la divergence de Jensen et Shannon, entre les distributions que représentent les prédictions des deux vues, mais l'aspect saillent de cette méthode est qu'elle renforce la diversité des prédictions (suivant les différentes vues), en introduisant des exemples adverses croisés (par rapport aux deux réseaux de neurones).

Intuitivement, il s'agit de forcer chaque modèle à prédire le contraire de ce que prédit erronément l'autre (sur ses exemples adverses) : la fonction de coût correspondante est maximale lorsque les deux réseaux de neurones sont identiques, comme mentionné par Peng et al. (2020, p. 8).

1.2.1.3 Le self-ensembling

Pour ce type de semi-supervision, il s'agit aussi de trouver une cohérence entre plusieurs réseaux de neurones, afin de générer des meilleures pseudo-étiquettes, mais ici les variantes utilisées proviennent du même modèle : elles ne sont plus indépendantes.

Pour les méthodes récentes, les variantes du modèle entrainé sont créées en utilisant une moyenne exponentielle mouvante (*exponential moving average,* EMA), tout au long des époques d'entraînement, comme proposé par Tarvainen & Valpola (2018).

Pour la segmentation, les travaux de Yu, Wang, Li, Fu, & Heng (2019) exploitent les données non-annotées en renforçant la cohérence entre les prédictions du modèle entrainé et celles du modèle trouvé par EMA. De plus, les voxels retenus pour les pseudo-étiquettes sont sélectionnés en prenant en compte l'incertitude liée à la prédiction. Pour ces travaux, l'incertitude est évaluée avec l'entropie des prédictions du modèle EMA, lorsqu'on varie les perturbations (bruits) ajoutées à son entrée.

Similairement, les travaux de Y. Wang et al. (2020) ont exploité les données non annotées, mais en prenant en compte l'incertitude liée aux descripteurs appris : l'ampleur de la contribution de la fonction de coût semi-supervisée est adaptée, suivant le nombre de régions d'activations qui varient peu (sous les perturbations), et l'incertitude des voxels prédits.

1.2.2 L'apprentissage par curriculum

Cette méthode est inspirée des travaux de Bengio, Louradour, Collobert, & Weston (2009), l'appellation de cette méthode vient du fait que l'apprentissage commence par une tâche plus simple avant d'aborder une tâche plus complexe.

Dans la littérature, les stratégies d'apprentissage par curriculum consistent souvent à trier et ordonner les exemples utilisés au cours de l'apprentissage d'une tache, de telle sorte que le réseau de neurones apprenne à traiter les exemples simples avant d'aborder des exemples

difficiles. Il existe aussi des stratégies qui s'intéressent à l'apprentissage de plusieurs tâches : les tâches faciles sont apprises préalablement pour faciliter l'apprentissage des tâches difficiles. Dans cette recherche, nous nous intéressons spécifiquement aux curriculum(s) dont la tâche simple consiste à inférer la valeur des propriétés (a priori) de la région à segmenter, pour améliorer l'apprentissage de la segmentation (tâche plus complexe), de façon graduelle.

L'apprentissage par curriculum est pertinent pour notre problématique parce qu'il améliore la qualité des segmentations produites, sans nécessiter plus d'annotations faites par des experts humains.

Dans les faits, les techniques d'apprentissage par curriculum de la segmentation consistent à utiliser un modèle auxiliaire, souvent séparé, pour apprendre à prédire des a priori(s). Ces derniers sont ensuite utilisés pour pénaliser les écarts observés dans les prédictions.

Dans le cas de la méthode de Zhang, David, & Gong (2017), un modèle de régression multinomiale infère la distribution des classes des pixels, pour indiquer comment modifier les segmentations. Un autre modèle auxiliaire, une machine à vecteur de support (SVM), indique où modifier les prédictions, en inférant un centaine de *superpixels*, développés par Li & Chen (2015).

Comme cela a été rapporté dans le Tableau 2 par Zhang et al. (2017, p. 8), les meilleurs résultats sont obtenus lorsqu'on utilise un a priori global ensemble avec un a priori local.

Dans le domaine médical, l'apprentissage par curriculum a été utilisée par Kervadec, Dolz, Granger, & Ayed (2019) pour segmenter le ventricule gauche du cœur humain. Dans cette implémentation, le réseau de neurones auxiliaire est utilisé pour inférer la taille de l'organe en question, des bornes sont calculées à partir de cette prédiction, pour imposer un intervalle dans lequel la taille de l'organe segmenté doit se trouver.

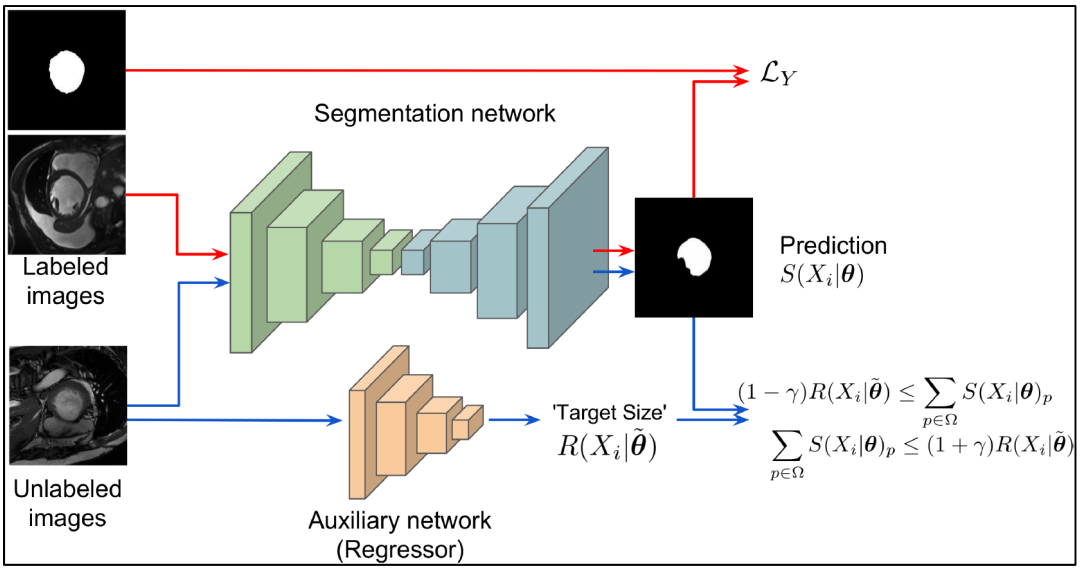

Figure 1.4 Illustration de la stratégie d'apprentissage par curriculum
Tirée de Kervadec et al. (2019, p. 4)

Il convient de noter que cette méthode a été choisie comme étalon, non seulement parce qu'elle segmente des images médicales, mais aussi parce qu'elle contraint directement les segmentations produites, contrairement aux autres méthodes qui contraignent les distributions des pixels : Zhang et al. (2017) ; Zhou et al. (2019a) ; etc. Notons que cette méthode n'a pas combiné un a priori local avec un a priori global.

Soient $R(\boldsymbol{X}_n)$ la prédiction de la taille de l'organe et Y_n le masque binaire vrai, le réseau auxiliaire est entraîné avec la somme des écart quadratiques :

$$\mathcal{L}_{size} = \sum_{n=1}^{N_1} \left(R(\boldsymbol{X}_n) - \sum_{\text{h,w}} \boldsymbol{Y}_n \right)^2 \tag{1.11}$$

Adapté de Kervadec et al. (2019a)

Après la convergence du modèle de régression, le modèle de segmentation (Θ_1) est entraîné en contraignant ses prédictions avec les tailles inférées (par l'auxiliaire) :

$$\min_{\Theta_1} \mathcal{L}_{ce}$$
$$t.q. \quad (1-\delta)R(\boldsymbol{X}_n) \leq \sum_{h,w} \boldsymbol{S}_n{}^{(h,w)} \leq (1+\delta)R(\boldsymbol{X}_n) \tag{1.12}$$

Adapté de Kervadec et al. (2019a)

Plus concrètement, les contraintes sont exprimées avec un terme additif $\mathcal{L}_{Naive}$:

$$\mathcal{L}_{seg} = \mathcal{L}_{ce} + \lambda_{Size}\mathcal{L}_{Naive} = \mathcal{L}_{ce} + \lambda_{size} \sum_{n=N_1+1}^{N_2} P\left(\sum_{h,w} \boldsymbol{S}_n{}^{(h,w)}\right) \tag{1.13}$$

$$avec \ P(t) \begin{cases} \left(t - (1-\delta_{Size})R(\boldsymbol{X}_n)\right)^2 si \ t \leq (1-\gamma)R(\boldsymbol{X}_n) \\ \left(t - (1+\delta_{Size})R(\boldsymbol{X}_n)\right)^2 si \ t \geq (1+\gamma)R(\boldsymbol{X}_n) \\ 0 \ sinon \end{cases} \tag{1.14}$$

Adaptées de Kervadec et al. (2019a)

Dans cette expression, P est la fonction de pénalité et l'hyperparamètre δ_{Size} contrôle le serrage de l'intervalle, selon Kervadec et al. (2019a, p. 4).

Sur le jeu de données *Automated Cardiac Diagnosis Challenge* (ACDC), publié par Bernard et al. (2018), en entrainant avec plusieurs proportions de scans annotés (sur un total de 75 scans), il a été démontré expérimentalement que cette méthode segmente mieux que les modèles entrainés en supervision totale, ou en semi-supervision par pseudo-étiquettes.

L'organe segmenté était le ventricule gauche du cœur, et à partir de 20 scans annotés sur 75, la méthode a montré des résultats comparables à ceux qu'on obtient lorsque les bornes sont exactes (non inférées). Par ailleurs, ce modèle nécessite que des augmentations d'images soient créées, pour l'entrainement de son réseau auxiliaire, afin de bien généraliser. De plus, comme mentionné précédemment, aucun a priori local n'est adjoint à l'a priori global.

1.2.3 Les réseaux de neurones contraints

Cette partie introduit les outils d'optimisation utilisés pour développer le modèle proposé par cette recherche. En fait, le terme additif ($\mathcal{L}_{Naive}$) vu dans la section 1.2.2 n'est pas la seule façon possible pour optimiser un réseau de neurones sous contraintes.

L'une des méthodes typiques d'optimisation sous-contraintes est le formalisme de Lagrange. Ce formalisme consiste à transformer le problème d'optimisation sous contraintes en deux problèmes d'optimisation sans contrainte, selon Walter (2014, p. 251). On a :

$$\max_{\boldsymbol{\lambda}} \min_{\theta} \; \mathcal{L}_{GT} \; + \sum_{i=1}^{P} \sum_{n=1}^{N} \lambda_i^n C_i(\boldsymbol{S}_n) \qquad (1.15)$$
$$s.t \quad \boldsymbol{\lambda} \epsilon \mathbb{R}^{P \times N}, \boldsymbol{\lambda} \succeq 0$$

Adapté de Kervadec, Dolz, Yuan, et al. (2020a)

Où P est le nombre de contraintes imposées et $\boldsymbol{\lambda}$ est le vecteur multiplicateur de Lagrange. Notons que la fonction $\mathcal{L}_{GT}$ utilise les quelques pixels annotés (le cas échéant). Le problème dual consiste à minimiser cette somme (le lagrangien), lorsqu'on fixe θ ; et le problème primal consiste à entrainer le réseau de neurones : optimiser θ en fixant $\boldsymbol{\lambda}$**.**

Intuitivement, cette formalisation signifie que le vecteur gradient de la fonction à optimiser est colinéaire avec le vecteur gradient des contraintes, au point optimal et faisable θ^*. L'appellation « multiplicateur de Lagrange » vient du fait que $\boldsymbol{\lambda}$ est le taux de variation de la fonction optimisée, par rapport aux variations des contraintes imposées.

Étant donné que l'optimisation se fait en entrainant tout un réseau de neurones, à chaque itération primale, la méthode n'est pas réaliste pour l'entrainement des réseaux de neurones parce que les coûts en temps de calcul sont trop importants, comme mentionné par Kervadec et al. (2020a, p. 3).

Cette méthode présente d'autres problèmes majeurs étudiés par Pathak et al. (2015), entre autres, mais en général des simples pénalités sont préférées à la méthode lagrangienne, pour l'apprentissage profond, en dépit de leurs multiples désavantages connus et évoqués par Márquez-Neila, Salzmann, & Fua, (2017, p. 1) et par Kervadec et al. (2020a, p. 4).

Tout de même, les travaux de Kervadec (2021) ont cherché à conserver les avantages de l'optimisation lagrangienne tout en évitant les itérations duales. Parmi ces avantages figure la garantie de la satisfaction des contraintes s'il existe des solutions faisables, selon Kervadec (2021, pp. 50, 51). En fait cette méthode est une extension des méthodes d'optimisation sous contrainte par fonction de barrière logarithmique (*log-barrier*). Contrairement aux pénalités, elle arrive à satisfaire (en même temps) plusieurs contraintes en compétition.

Il s'agit d'optimiser le problème suivant pour obtenir les paramètres d'un réseau de neurones optimisé sous contraintes :

$$\min_{\theta} \mathcal{L}_{GT} + \sum_{i=1}^{P} \sum_{n=N_1+1}^{N_2} \tilde{\psi}_t\big(C_i(\boldsymbol{S}_n)\big) \tag{1.16}$$

$$avec \quad \tilde{\psi}_t(z) = \begin{cases} -\frac{1}{t} log(-z) & si \;\; z \leq -\frac{1}{t^2} \\ tz - \frac{1}{t} log\left(\frac{1}{t^2}\right) + \frac{1}{t} & sinon \end{cases} \tag{1.17}$$

Adaptées de Kervadec, et al. (2020a, pp. 8, 9)

Selon cet article, la fonction $\tilde{\psi}$ est une approximation qui tend vers l'infinie, lorsque t tend vers l'infinie et que z est supérieur à zéro (contrainte insatisfaite). Elle tend vers zéro, sinon. La particularité est que le domaine de définition ne se limite pas à l'ensemble faisable : la méthode ne nécessite pas de point faisable initial, selon Kervadec et al. (2020a, p. 9). La figure à la page suivante illustre l'allure de cette fonction de coût.

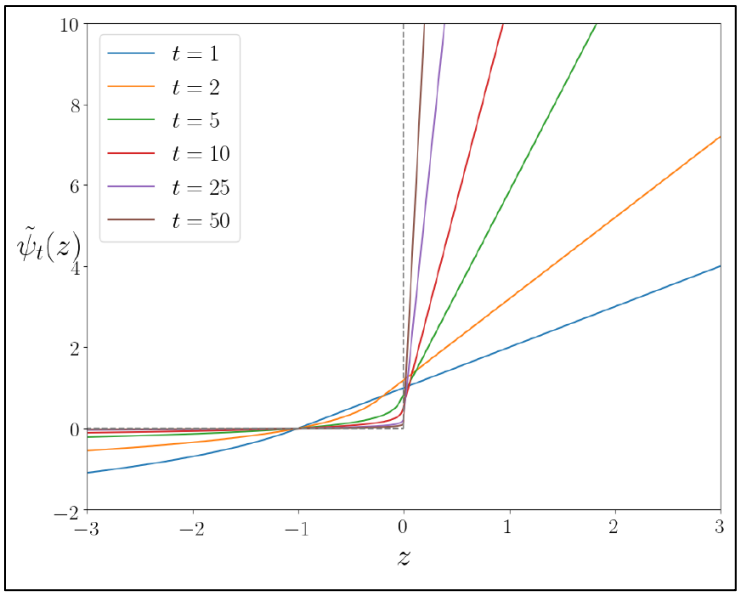

Figure 1.5 Illustration de la fonction de coût $\tilde{\psi}_t(z)$
Tirée de Kervadec et al. (2019b, p. 6)

1.2.3.1 Cas d'usage

Cette méthode a été utilisé par Kervadec, Dolz, Wang, et al. (2020b), pour contraindre les pixels de l'organe entre les limites spatiales de sa boîte englobante (serrée) :

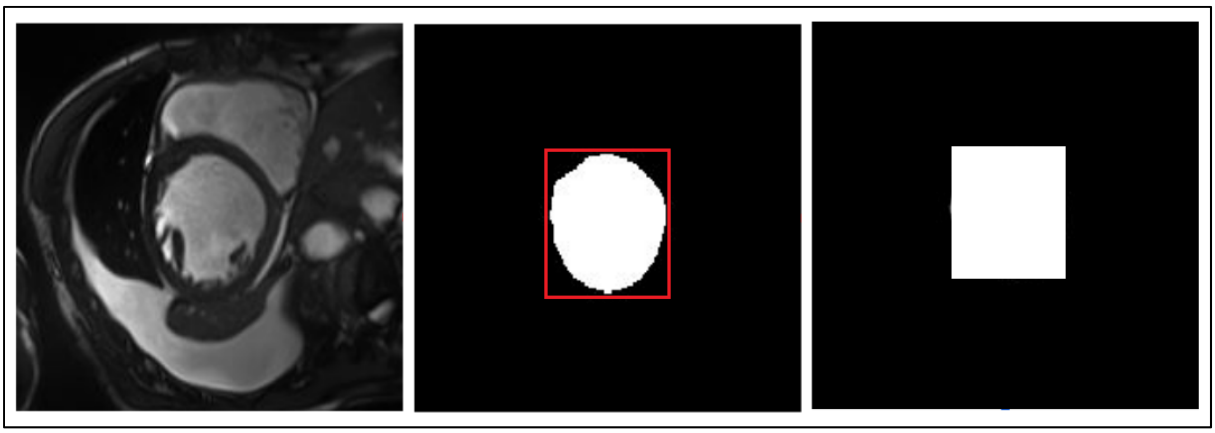

Figure 1.6 Illustration des limites spatiales qu'une boîte englobante serrée impose

La figure ci-dessus présente (à gauche) une tranche d'un scan tiré du jeu de données ACDC, publié par Bernard et al. (2018) ; au milieu se trouve le masque binaire du ventricule gauche, et le contour (en rouge) de la boîte qui englobe l'organe. Plus à droite, c'est le masque binaire de la boîte englobante : les pixels en son sein (en blanc) forment l'ensemble Ω_{In}, et les pixels à l'extérieur (en noir) forment l'ensemble Ω_{Out}.

L'entrainement se fait en approximant, avec des itérations simples, une solution du problème exprimé par l'équation suivante :

$$\min_{\theta} \sum_{n=1}^{N_1} \sum_{i=1}^{4} \tilde{\psi}_t\big(C_i(\boldsymbol{S}_n)\big) \tag{1.18}$$

Adapté de Kervadec et al. (2020b)

Où C_i représente une contrainte. À l'extérieur de la boîte qui englobe un organe, il n'y a aucun pixel de la même catégorie :

$$C_1(\boldsymbol{S}_n) = \sum_{(\mathrm{h,w})\epsilon\Omega_{out}} \boldsymbol{S}_n{}^{(h,w)} \quad \Leftrightarrow \quad \sum_{(\mathrm{h,w})\epsilon\Omega_{out}} \boldsymbol{S}_n{}^{(h,w)} \leq 0 \tag{1.19}$$

Adapté de Kervadec et al. (2020b)

Les bornes de l'intervalle où se trouve la taille de l'organe englobé dépendent de la taille de la boîte englobante :

$$C_2(\boldsymbol{S}_n) = \ \delta_{min}|\Omega_{In}| \ - \sum_{(\mathrm{h,w})\epsilon\Omega_{out}} \boldsymbol{S}_n{}^{(h,w)} \tag{1.20}$$

$$C_3(\boldsymbol{S}_n) = \sum_{(\mathrm{h,w})\epsilon\Omega_{out}} \boldsymbol{S}_n{}^{(h,w)} - \ |\Omega_{In}| \tag{1.21}$$

$$\Leftrightarrow \ \delta_{min}|\Omega_{In}| \leq \sum_{(\mathrm{h,w})\epsilon\Omega_{In}} \boldsymbol{S}_n{}^{(h,w)} \leq \ |\Omega_{In}| \tag{1.22}$$

Adaptées de Kervadec et al. (2020b)

Si l'on définit S_L comme étant l'ensemble des segments à l'intérieur de la boîte englobante qui sont orthogonaux par rapport à l'un de ses côtés ; le nombre d'intersections que les

segments (de largeur ω, appartenant à l'ensemble S_L) ont avec les pixels de l'organe englobé doit être supérieur ou égal à ω (en pixels) ;

$$C_4(\boldsymbol{S}_n) = \sum_{j=1}^{|S_L|} \left(\omega - \sum_{(h,w)\epsilon\Omega_{In}} \boldsymbol{AB_j S}_n \right) \quad \Leftrightarrow \quad |\overline{AB} \cap \boldsymbol{S}_n| \geq \omega, \forall\, \overline{AB}\, \epsilon\, S_L \qquad (1.23)$$

Adapté de Kervadec et al. (2020b)

Où $\boldsymbol{AB_j}$ est le masque binaire (une matrice) où seuls les pixels qui font partie du segment $\overline{AB_j}$ sont non nuls, et la notation $|\quad|$ désigne le cardinal de l'ensemble. La figure à la page qui suit illustre deux exemples de segments qui font partie de l'ensemble S_L.

Figure 1.7 Illustration de deux segments appartenant à S_L

Sur le jeu de données PROMISE 12, publié par Litjens et al. (2013), cette technique a montré des performances qui se rapprochent considérablement de celles d'un modèle complètement supervisé (avec le même nombre d'images), en utilisant l'a priori de la boîte englobante serrée comme la seule annotation disponible.

Toutefois, cette technique n'utilise pas des boîtes englobantes serrées qui prennent en compte l'angle de rotation de l'axe principal de la région à segmenter. Cette remarque est pertinente dans la mesure où la prise en compte de cet angle pourrait améliorer les performances.

Il convient de noter que les résultats dans le Tableau 1 de Kervadec et al. (2020b, p. 10) montrent aussi que l'utilisation d'a priori globaux, ensemble avec des a priori locaux, améliore les performances.

1.3 Le résumé de la revue de littérature

Somme toute, plusieurs méthodes de semi-supervision ont été explorées pour pallier le manque d'annotations suffisantes. Toutefois, l'état de l'art en apprentissage par curriculum, n'utilise pas encore les pénalités proposées par Kervadec et al. (2020a). De plus, la méthode de Kervadec et al. (2019a) n'adjoint pas d'apriori local à son apriori global, alors que les travaux de Zhang et al. (2017) et de Kervadec et al. (2020b) ont observé que l'utilisation d'un a priori global avec un a priori local améliore les performances.

En notre connaissance, l'a priori des boîtes englobantes serrées n'a pas encore été explorée dans un contexte d'apprentissage semi-supervisé par curriculum. Il existe donc un manque de connaissance, quant aux performances qu'aurait un modèle semi-supervisé par curriculum avec des boîtes englobantes inférées.

C'est pourquoi nos travaux proposent de combler ce gap, en explorant un modèle d'apprentissage par curriculum simple, avec les pénalités et les bornes (locales et globales) utilisées dans le travaux de Kervadec et al. (2020b).

Par ailleurs, l'une des limites connues de la génération des pseudo-étiquettes est la propagation des erreurs de prédiction, comme mentionné par Dolz, Desrosiers, & Ayed (2020, p. 4). En effet, lorsque ces dernières ne sont pas corrigées par le raffinement, le réseau de neurone s'auto-corrige avec des faussetés : les performances sont directement liées à la qualité des pseudo-étiquettes.

Comparativement, nous établissons les rapprochements et écarts pertinent suivants, entre les différentes méthodes de semi-supervision revues :

- l'entrainement avec un réseau auxiliaire (en *self-training*, en *co-training*, en *self-ensembling*), consomme des ressources additionnelles (mémoire vive et calculs) ;
- le *co-training* nécessite beaucoup de ressources étant donné que les multiples vues sont entrainées en même temps ;
- le *self-ensembling* nécessite moins de ressources pour la mise à jour du réseau auxiliaire (EMA), mais les besoins restent considérables étant donné que les architectures des deux réseaux de neurones doivent être identiques (au moins, deux fois plus de paramètres en mémoire) ;
- le *self-training* par apprentissage adverse consomme encore moins de ressource, même s'il utilise un réseau auxiliaire (discriminateur), parce que ce dernier est considérablement plus petit que le réseau qui effectue la segmentation ;
- comme évoqué par Kodali, Abernethy, Hays, & Kira (2017, p. 2), l'interaction entre deux réseaux de neurones adverses est enclin à des instabilités connues (*mode collapse & cycling),* pendant l'entrainement ;
- la segmentation par curriculum ne nécessite pas que les deux réseaux de neurones soient chargées en mémoire (les prédictions du réseau auxiliaire peuvent être sauvegardées, parce qu'elles ne varient pas) ;
- le *co-training* et le *self-ensembling* sont conceptuellement plus proches, parce qu'ils exploitent le consensus des vues ;
- contrairement aux autres méthodes, la segmentation par curriculum exploite directement les connaissances du domaine médicale, ce qui peut aussi pallier les aberrations anatomiques dans les segmentation produites (voir la section 1.1.3).

Ainsi donc, l'entrainement des méthodes semi-supervisées par *co-training* et par *self-ensembling* consomme des ressources qui ne sont pas forcément disponibles dans chaque clinique. Le consensus des vues multiples est en particulier plus exigeant quant à l'implémentation des algorithmes dans un processeur embarqué (par exemple), étant donnée les ressources nécessaires.

Eu égard à ce qui précède, pour la comparaison à l'état de l'art, nous avons retenu le *self-training* parce qu'il consomme le moins de ressources additionnelles, et parce que c'est l'un des étalons souvent utilisés pour évaluer les méthodes de segmentation semi-supervisées. La proposition de Hung et al., en particulier, parce que son raffinement est appris, selon Hung et al. (2018, p. 9), avec un modèle ayant une capacité de représentation supérieure aux CRF(s).

La partie qui suit étaye notre méthodologie, en présentant la formalisation de l'entrainement de deux réseaux de neurones qui constituent le model proposé.

CHAPITRE 2

MÉTHODOLOGIE

2.1 La méthodologie globale

Notre modèle est un réseau de neurones entièrement convolutif qui est semi-supervisé par curriculum, avec des boîtes englobantes. Ces boîtes sont inférées par un autre réseau entièrement convolutif : le réseau auxiliaire. Les contraintes sont exprimées suivant le formalisme publié par Kervadec et al. (2020a) : comme l'illustre la figure ci-dessous, les boîtes englobantes sont inférées sous forme de masques binaires.

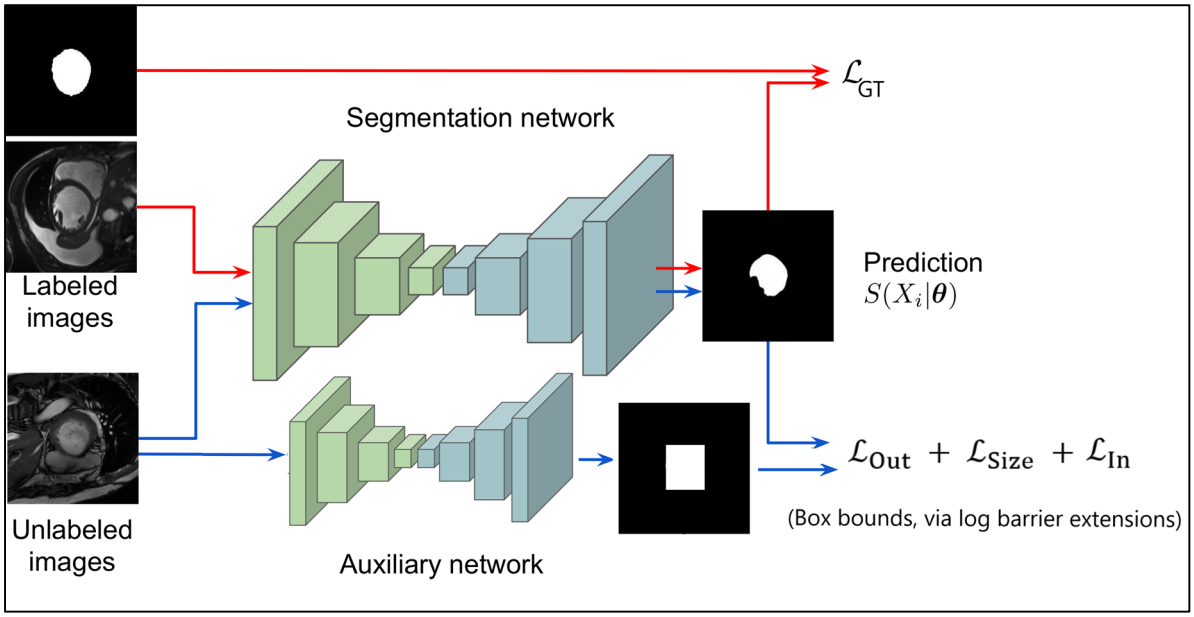

Figure 2.1 Illustration de l'utilisation d'un réseau auxiliaire entièrement convolutif, pour prédire les masques des boîtes englobantes ; reproduite et adaptée avec l'autorisation de Kervadec et al. (2019a, p. 4)

Sur la figure ci-dessus, les lignes rouges décrivent le flux des données annotées, et les lignes bleus décrivent le flux des données non-annotées. Notons que ce schéma d'entrainement (pour le réseau principal) utilise un réseau auxiliaire dont les paramètres ne varient pas.

Les sections qui suivent détaillent les fonctions de coût utilisées pour superviser les deux réseaux de neurones et l'algorithme utilisé pour simplifier l'inférence des boîtes englobantes (afin d'utiliser une architecture entièrement convolutive).

2.2 L'inférence des boîtes englobantes

Le réseau auxiliaire (entièrement convolutif) est entrainé avec les masques binaires des boites englobantes, calculées sur le petit ensemble annoté : $D_S = \{(\boldsymbol{X}_n, \boldsymbol{Y}_n)\}_{n=1,\dots,N_1}$. Il n'interagit avec les données non-annotées, qu'après la fin de son entrainement, pour inférer des pseudo-boîtes englobantes à utiliser dans la semi-supervision du réseau principal.

Ces pseudo-boîtes servent à créer des bornes non-exactes, pour caractériser la région de l'organe à segmenter. La gestion de l'incertitude sous-jacente est décrite dans la section 2.3.

Il ne s'agit pas d'apprentissage multitâche, parce qu'aucun paramètre n'est partagé entre les deux réseaux. Il ne s'agit pas non plus d'apprentissage en conjonction parce que le réseau auxiliaire et entrainé seul, jusqu'à la convergence : les paramètres du réseau auxiliaire ne changent pas pendant l'entrainement du réseau principal.

Pour associer à chaque organe dans l'image un masque binaire, où les seuls pixels égaux à un représentent l'aire de la boite qui englobe les composantes connectées de l'organe (voir la Figure 1.6), nous utilisons une entropie croisée. Dans le cas de la segmentation d'un seul organe, il n'est produit qu'un seul masque par image :

$$\mathcal{L}_{bbox} = -\sum_{n=1}^{N_1} (\boldsymbol{\Omega_{Out}})_n log(1 - B(\boldsymbol{X}_n)) + (\boldsymbol{\Omega_{In}})_n log(B(\boldsymbol{X}_n)) \tag{2.1}$$

Où $B(\boldsymbol{X}_n)$ représente les prédictions du réseau auxiliaire pour l'image $\boldsymbol{X}_n$, $(\boldsymbol{\Omega_{Out}})_n$ et $(\boldsymbol{\Omega_{In}})_n$ représentent, respectivement, les pixels à l'extérieur et les pixels à l'intérieur de la boîte englobante (voir la Figure 1.6). Pour alléger la notation, posons que $B(\boldsymbol{X}_n) = \boldsymbol{B}_n$.

Les annotations auxiliaires (qui servent à superviser le réseau auxiliaire) ne sont pas à confondre avec les masques de segmentation des organes : comme le montre la Figure 1.6, un pixel qui fait partie de la boîte englobante ne fait pas nécessairement partie de l'organe à segmenter.

Pour créer les masques des boîtes englobantes, l'algorithme publié par Wu, Otoo, & Shoshani (2005) est utilisé afin d'extraire les composantes connectées de l'organe d'intérêt. Ensuite, une boîte englobante est calculée pour chacun de ces groupes de pixels.

Cela se fait en déterminant la position du pixel le plus en haut à gauche, ainsi que celle du pixel le plus en bas à droite, par rapport au groupe en question. Il sied de noter que cette procédure vient de l'implémentation officielle de Kervadec et al. (2020b).

Pendant l'inférence, les boîtes englobantes sont retrouvées similairement, le pseudo masque des boîtes englobante $\left(\widehat{\boldsymbol{B}}_n\right)$ est créé comme suit :

Algorithme 2.1 Création du pseudo-masque des boîtes englobantes

Création du pseudo-masque des boîtes englobantes

Entrée : la tranche $\boldsymbol{X}_n$, le réseau auxiliaire B

Sortie : le pseudo-masque des boîtes englobantes ($\widehat{\boldsymbol{B}}_n$)

Initialiser $CC =$ composantes connectées de $(\boldsymbol{B}_n > 0.5)$; $\widehat{\boldsymbol{B}}_n = \boldsymbol{0}$
Pour i allant de 0 à $|CC| - 1$
    $\boldsymbol{BB_i} =$ masque binaire de boite la boîte englobante de $CC[i]$
    $\widehat{\boldsymbol{B}}_n = \widehat{\boldsymbol{B}}_n \; OU \; \boldsymbol{BB_i}$
sortir $\widehat{\boldsymbol{B}}_n$

Où $\boldsymbol{0}$ est la matrice nulle (de même dimension que $\boldsymbol{X}_n$), et l'opération OU fonctionne élément par élément. Ici, $|CC|$ représente le nombre de composantes connectées et $CC[i]$ représente la composante connectée à l'indice i.

Les composantes connectées sont gérées séparément, pour tenir compte des cas où l'organe à segmenter a des régions éloignées. Cet algorithme d'inférence fonctionne en ligne sur un processeur graphique (GPU), pour maintenir la rapidité, en dépit de la taille du jeu de données. L'implémentation de l'extraction des composantes connectées sur GPU est empruntée d'un code source accessible publiquement, basé sur les travaux de Allegretti, Bolelli, & Grana (2020).

2.3 La semi-supervision basée sur des boîtes englobantes inférées

Comme l'illustre la Figure 2.1, les annotations disponibles sont exploitées avec la fonction de coût $\mathcal{L}_{GT}$. En fait, nous utilisons une contrainte qui impose que le nombre de pixels prédits à l'extérieur du masque de l'organe soit nul (idem, pour le fond de l'image).

Nous avons préféré cette fonction de coût à l'entropie croisée en raison des observations publiées dans le Tableau 1 publié par Kervadec et al. (2020b, p. 10) : une utilisation mal équilibrée de l'entropie croisée, ensemble avec ces fonctions des contraintes, peut réduire les performances jusqu'à 31.6% (en moins).

Par conséquent, pour segmenter un seul organe :

$$\mathcal{L}_{GT} = \sum_{n=1}^{N_1} \tilde{\psi}_t \left(\sum_{(h,w)} [(\sim \boldsymbol{Y}_n) \boldsymbol{S}_n]^{(h,w)} \right) \tag{2.2}$$

Où $(\sim \boldsymbol{Y}_n)$ est l'opposé booléen du masque binaire de l'organe à segmenter.

En ce qui concerne l'exploitation d'images non annotées, le réseau auxiliaire est utilisé pour produire des pseudo-boîtes englobantes : il infère nos annotations auxiliaires.

Étant donné que l'inférence ne garantit pas l'exactitude, l'incertitude est gérée en introduisant des hyperparamètres (tous scalaires) dans les composante de l'équation (6) de Kervadec et al. (2020b), pour imposer des contraintes plus flexibles.

Ainsi donc, $\mathcal{L}_{Size}$, la contrainte qui impose l'intervalle où doit se trouver l'aire de la segmentation proposée, est adapté avec l'hyperparamètre δ_{max} :

$$\mathcal{L}_{Size} = \tilde{\psi}_t\left(\delta_{min}\sum_{(h,w)}\widehat{\boldsymbol{B}}_n{}^{(h,w)} - \sum_{(h,w)}\boldsymbol{S}_n{}^{(h,w)}\right) + \tilde{\psi}_t\left(\sum_{(h,w)}\boldsymbol{S}_n{}^{(h,w)} - \delta_{max}\sum_{(h,w)}\widehat{\boldsymbol{B}}_n{}^{(h,w)}\right) \tag{2.3}$$

De ce fait, avec des valeurs adéquates de δ_{min} et δ_{max}, l'apprentissage peut exploiter les prédictions qui s'écartent de l'intervalle vrai : on peut exploiter des boîtes englobantes trop larges ou trop serrées.

Il sied de noter qu'un δ_{max} non unitaire existe dans l'implémentation officielle de Kervadec et al. (2020b), même s'il ne figure pas dans les équations publiées.

De même, δ_{Out} est introduit dans la contrainte $\mathcal{L}_{Out}$, pour permettre qu'un certain nombre de pixels puissent se trouver à l'extérieur ($\sim$) de la pseudo-boîte englobante :

$$\mathcal{L}_{Out} = \tilde{\psi}_t\left(\sum_{(\mathrm{h,w})}\left(\sim\widehat{\boldsymbol{B}}_n{}^{(h,w)}\right)\boldsymbol{S}_n{}^{(h,w)} - \delta_{Out}\right) \tag{2.4}$$

Pour la contrainte qui impose le nombre d'intersections (la topologie) à l'intérieur de la boîte englobante, nous avons également introduit l'hyperparamètre δ_{In}. Cela est fait parce que la boîte ne vérifie pas nécessairement le caractère serré, telle qu'il est défini par Lempitsky, Kohli, Rother, & Sharp (2009, p. 3). On a donc :

$$\mathcal{L}_{In} = \sum_{\hat{s}_l \epsilon \hat{S}_L}\left[\tilde{\psi}_t\left((\omega - \delta_{In}) - \sum_{(h,w)\ \epsilon \hat{s}_l}\boldsymbol{S}_n{}^{(h,w)}\right)\right] \tag{2.5}$$

Où $\hat{S}_L$ est l'ensemble des droites orthogonales par rapport aux côtés de la pseudo-boîte englobante, et $(h,w)\ \epsilon \hat{s}_l$ signifie que le pixel en ces coordonnées fait partie du segment $\hat{s}_l$. Avec cet allègement, certaines droites dans $\hat{S}_L$ peuvent avoir moins de croisement que leurs largeurs (en pixels).

Les autres hyperparamètres introduits sont λ_{Size}, λ_{Out} et λ_{In}. Ils servent à équilibrer les contributions des composantes et à adapter les taux d'apprentissage, par rapport à la fonction de coût finale :

$$\mathcal{L}_{seg} = \mathcal{L}_{GT} + \left(\sum_{n=N_1+1}^{N_2} \lambda_{Size}\mathcal{L}_{Size} + \lambda_{Out}\mathcal{L}_{Out} + \lambda_{In}\mathcal{L}_{In} \right) \tag{2.6}$$

Pour illustrer ces trois contraintes, la figure ci-après met en valeur $\mathcal{L}_{Out}$ en haut à gauche, $\mathcal{L}_{Size}$ en haut à droite, et $\mathcal{L}_{In}$ en bas. Pour la case qui concerne $\mathcal{L}_{In}$, B_n représente une boîte englobante exacte (non-inférée).

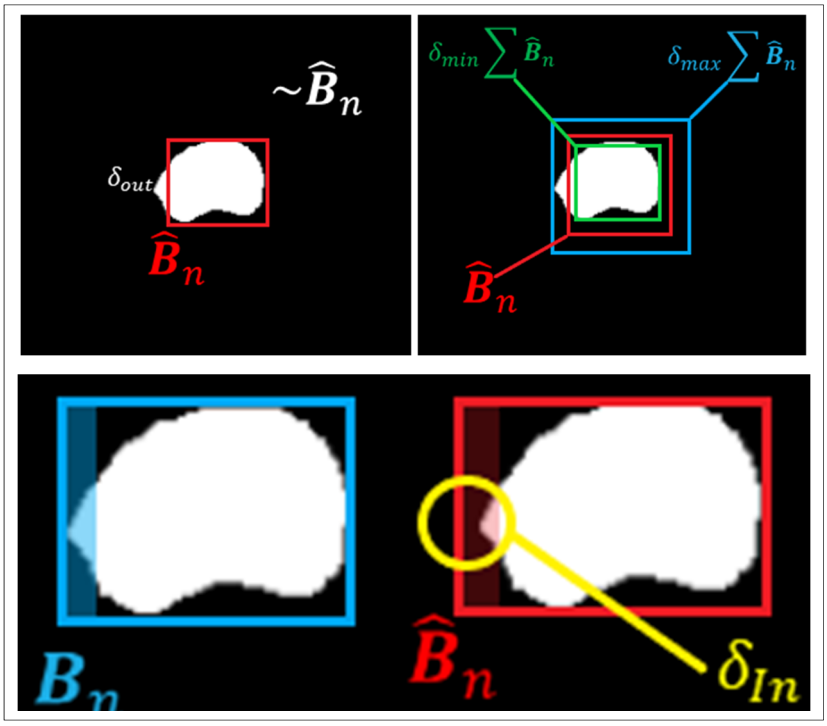

Figure 2.2 Illustration de l'adaptation des contraintes

Comme le montre visuellement la figure précédente, les hyperparamètres introduits peuvent être optimisés pour adapter ou alléger les contraintes, afin de pallier l'incertitude induite par l'utilisation des boîtes englobantes.

Le chapitre qui suit présente le protocole suivi pendant nos expérimentations, les détails de l'implémentation (les logiciels, le matériel, les jeux de données, et les configurations des modèles), ainsi que les résultats obtenus.

CHAPITRE 3

EXPÉRIMENTATIONS ET RÉSULTATS

3.1 Le protocole expérimental

Comme précisé dans l'introduction, nos expérimentations visent à comparer notre modèle avec les méthodes préexistantes, et à déterminer les contributions des composantes des fonctions de coût qu'utilise notre modèle, ainsi que la généralisation de notre méthode par rapport aux architectures de segmentation usuelles.

Nos expérimentations se basent sur le contrôle des initialisations des algorithmes randomisés pour pallier la variation des résultats, suivant les scissions des données et la variation des résultats de l'optimisation stochastique, entre autres. Le protocole expérimental enchaine les étapes suivantes :

- fixation d'une valeur d'initialisation pour tous les générateurs de nombres aléatoires ;
- scission aléatoire des données pour l'entrainement, la validation et le test ;
- validation des hyperparamètres pertinents (pour chaque modèle) et entrainement ;
- utilisation des paramètres enregistrés afin d'évaluer les performances sur l'ensemble de test (inférence).

Les valeurs rapportées sont la moyenne et l'écart-type obtenus lorsqu'on répète ces étapes trois fois (avec des graines d'initialisation différentes).

3.1.1 Les modèles comparés

Pour évaluer les performances de la semi-supervision basée sur l'inférence des boîtes englobantes, deux modèles pertinents ont été choisis parmi les modèles vus dans la revue de la littérature ; et deux autres modèles ont été créés pour définir les extrema.

Les deux algorithmes tirés de la revue de la littérature sont le modèle semi-supervisé par curriculum de Kervadec et al. (2019a), et le modèle adverse de Hung et al. (2018). Le premier modèle est choisi pour déterminer si la nouvelle méthode améliore l'état de l'art, établi par dans Kervadec et al. (2019a), et le deuxième sert à donner un placement par rapport aux méthodes de semi-supervision basées sur la génération des pseudo-étiquettes.

Pour les extrema (supremum et infimum), un modèle hybride a été créé. Il consiste à entrainer le réseau de neurones avec les masques de référence, sur le petit ensemble annoté. Sur le grand ensemble de données non annotées, il est supervisé avec les contraintes ayant des bornes exactes : c'est la borne supérieure, car il donne les meilleurs résultats possibles avec ce type de contraintes. Ces bornes exactes sont calculées à partir des masques de segmentation faits par les experts humains, et c'est pour le besoin des expérimentations que ces annotations sont considérées comme absentes.

Le deuxième modèle extrême (minorant) créé est complètement supervisé, mais il n'est entrainé qu'avec les annotations sur le petit ensemble : il n'exploite aucune information venant de l'ensemble de scans non-annotés, c'est pourquoi il constitue la borne inférieure.

3.1.2 Les variations effectuées

Pour comparer les modèles présentés dans la section précédente, nous avons varié la quantité des données annotés utilisées en entrainement, et nous avons observé l'influence de ce facteur sur les performances de chacun de ces modèles.

En ce qui concerne l'évaluation de la contribution de chaque composante de la fonction de coût utilisée, nous désactivons tour à tour chacune de ces dernières. De plus, nous évaluons les cas de figures où une seule composante de semi-supervision est utilisée.

Pour évaluer la généralisation des performances par rapport à l'architecture de segmentation choisie, le modèle a été entrainé en utilisant l'architecture *Efficient Net* (ENet) proposée par

Paszke, Chaurasia, Kim, & Culurciello (2016) ; UNet proposée par Ronneberger et al., 2015) ; DeepLabV3 proposée par Chen, Papandreou, Schroff, & Adam (2017) ; et FCN-ResNet50 proposée par Long et al. (2015). Elles ont été choisies en raison de leur utilisation récurrente dans la recherche, pour la segmentation par apprentissage profond.

Il convient de noter leurs différences comparatives pertinentes :

- FCN-ResNet50 est résiduelle, elle encode l'information à une seule échelle, et elle utilise une interpolation bilinéaire (paramètres fixes) dans les couches de décodage ;
- ENet est résiduelle, elle encode l'information à une seule échelle, et elle utilise des paramètres appris pour le décodage (la déconvolution) ;
- UNet est non-résiduelle, elle encode l'information à plusieurs échelles, et elle utilise l'interpolation suivie par des convolutions (paramètres appris) pour le décodage ;
- DeepLabV3-ResNet50 est résiduelle, elle encode l'information à plusieurs échelles, et elle utilise une interpolation (paramètres fixes) dans les couches de décodage.

Par ailleurs, ENet est l'architecture que nous utilisons principalement dans nos expérimentations. Ses caractéristiques essentielles sont la légèreté et la rapidité : suivant les expérimentations rapportées par Paszke et al. (2016), ENet était jusqu'à 18 fois plus rapide que l'état de l'art, avec 79 fois moins de paramètres. ENet est choisie dans cette recherche à cause de ces deux qualités, mais aussi parce que c'est elle qui est utilisée par les travaux de Kervadec et al. (2019a), et de Kervadec et al. (2020b). DeepLabV3 et FCN-ResNet50 sont également utilisés dans les domaines médicaux et non-médicaux, selon Minaee et al. (2020).

Comme évoqué dans la revue de la littérature, UNet est l'architecture typiquement utilisée pour la segmentation d'images médicales. UNet est en fait conçue pour apprendre à segmenter les images en utilisant des jeux de données de petite taille, tout en générant des détails fins, selon Ronneberger et al. (2015, p. 2).

Remarquons que ces architectures sont toutes de type encodeur-décodeur, et que UNet utilise des sauts de couche (*skip connections),* pour encoder l'information à plusieurs échelles ;

tandis que DeepLabV3 utilise des sous-échantillonnages avec des convolutions à trous, spatiales et pyramidales, selon Chen et al. (2017, p. 2).

Ici, les couches d'encodage désignent les groupes d'opérations (convolutions, fonctions d'activations, etc.) qui servent à sous-échantillonner ; et les couches de décodages désignent les groupes qui servent à reprojeter (suréchantillonner) à l'échelle de l'image prise entrée.

Le caractère résiduel désigne l'innovation proposée par He, Zhang, Ren, & Sun (2016), pour mieux entrainer les réseaux de neurones très profonds : les sauts en aval additifs et unitaires, entre les couches de convolution, illustrés dans la Figure 2 de He et al. (2016, p. 2).

3.1.3 Les métriques d'évaluation

En imagerie médicale, les performances de la segmentation sémantique peuvent être mesurées en caractérisant la précision des prédictions par voxel, la superposition du volume prédit avec le volume véridique de l'organe à segmenter, les différences topologiques, la fidélité des surfaces, etc.

Dans cette recherche, pour caractériser la superposition des volumes c'est l'indice de Sørensen-Dice (DSC) qui est utilisé, pour la fidélité des formes c'est la distance de Hausdorff (HD). Pour évaluer la superposition des masques des boîtes englobantes inférées, par rapport aux masques des boîtes englobantes vraies, nous utilisons le *Intersection Over Union* (IoU).

$$DSC = \frac{2\left|\hat{V}_p \cap V_p\right|}{\left|\hat{V}_p\right| + \left|V_p\right|} \tag{3.8}$$

$$IoU = \frac{\left|\hat{B}_n \cap B_n\right|}{\left|\hat{B}_n \cup B_n\right|} = \frac{2\left|\hat{B}_n \cap B_n\right|}{2\left(\left|\hat{B}_n\right| + \left|B_n\right| - \left|\hat{B}_n \cap B_n\right|\right)} \tag{3.9}$$

$$HD = max\left(h\left(\hat{V}_n, V_n\right), h\left(V_n, \hat{V}_n\right)\right) \tag{3.10}$$

$$avec \quad h(V_n, \hat{V}_n) = \max_{V_p^{(h,w,d)} \epsilon V_n} \min_{\hat{V}_p^{(h,w,d)} \epsilon \hat{V}_p} \left\| V_n^{(h,w,d)} - \hat{V}_n^{(h,w,d)} \right\|$$

Adaptées de Taha & Hanbury (2015, pp. 5, 9)

Où ‖ ‖ est une désigne la norme euclidienne, $V_p^{(h,w,d)}$ désigne l'annotation d'un voxel dans le volume V_p (un scan d'un unique patient donné) ; et $\hat{V}_p^{(h,w,d)}$ est la prédiction du réseau.

Comme le montre l'équation 3.9 à la page précédente, l'indice de Jacard (IoU) est toujours supérieur ou égale à l'indice de Sørensen-Dice, mais ces deux métriques caractérisent la superposition.

Pour l'interprétation, la superposition des volumes (des masques) est meilleure d'autant plus que l'indice DSC (l'indice IoU) se rapproche de la valeur un ou 100%, et les formes sont similaires d'autant plus que la distance HD se rapproche de la valeur zéro.

3.1.4 Les détails d'implémentation

3.1.4.1 Les logiciels et le matériel

En ce qui concerne les librairies dont dépend le code source utilisé, une liste exhaustive est disponible dans le dépôt du projet. Il s'agit principalement des librairies de calcul numérique usuelles (*NumPy*, *MedPy*, etc.), ainsi que de la librairie *PyTorch* dédiée aux calculs sur GPU. Le dépôt du projet se trouve à l'adresse suivante : https://github.com/oneCreativeUserName/Publications.computer-vision.master-thesis). La référence de l'implémentation de Allegretti et al. (2020), que nous avons mentionnée précédemment, est aussi listée dans les dépendances, dans le fichier *README.md* du projet.

En ce qui concerne le code source principal, il découle principalement des codes sources publiés dans les travaux de Kervadec et al. (2019a) et de Kervadec et al. (2020b). Ces emprunts et adaptations ont été effectués pour faciliter cette recherche, et reproduire les résultats des expérimentations de l'étalon choisi. Similairement, le code source de Hung et al. (2018) a été également emprunté, et adapté.

3.1.4.2 Les jeux de données

Pour les expérimentations, deux jeux de données constituées d'images médicales sont choisis afin de faciliter la comparaison avec les résultats publiés dans la littérature. Le premier jeu de données est ACDC, publié par Bernard et al. (2018), et utilisé par Kervadec et al. (2019a). Le second est jeu de données est *Prostate Image Segmentation 2012* (PROMISE 12), publié par Litjens et al. (2013). C'est le jeu de données utilisé par Kervadec et al. (2020b).

La partie du jeu de données ACDC utilisée est constituée de 100 scans (IRM), avec deux trames par scan, obtenus avec deux scanners cliniques différents, à l'hôpital universitaire de Dijon. Notre organe d'intérêt est le ventricule gauche du cœur, et seule une minorité représente des patients sans anomalie. Chaque scan contient deux trames ayant jusqu'à dix tranches.

En tout, nous avons dédié 70 scans aux entrainements (soient 1342 tranches), cinq à la validation (soient 84 tranches), et 20 aux tests (soient 476 tranches). Les prétraitements effectués sur ces données sont les suivant : les tranches ont été redimensionnées à 256x256 pixels, et les valeurs dans chaque volume ont été normalisées entre zéro et un.

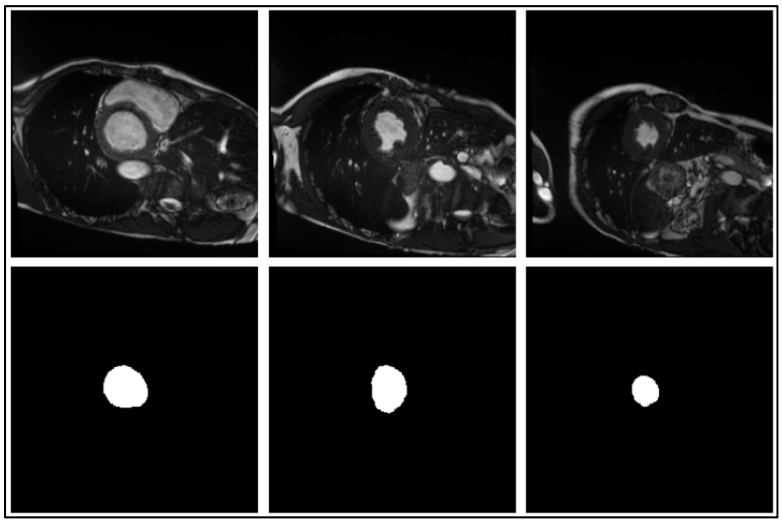

Figure 3.1 Aperçu des tranches d'un scan de ACDC,
ainsi que les masques de référence correspondants

En ce qui concerne le jeu de données PROMISE 12, la partie de ce jeu de données que nous avons utilisée est constituée de 50 scans IRM, obtenus suivant des protocoles différents, et dans des cliniques différentes. L'organe d'intérêt est la prostate, elle y est représentée avec des particularités qu'il sied de noter : certains scans contiennent un anneau endorectal, tandis que d'autres n'en ont pas. Certains scans n'ont que 15 tranches, alors que d'autres comptent jusqu'à 54 tranches. Tous les scans contiennent une seule trame.

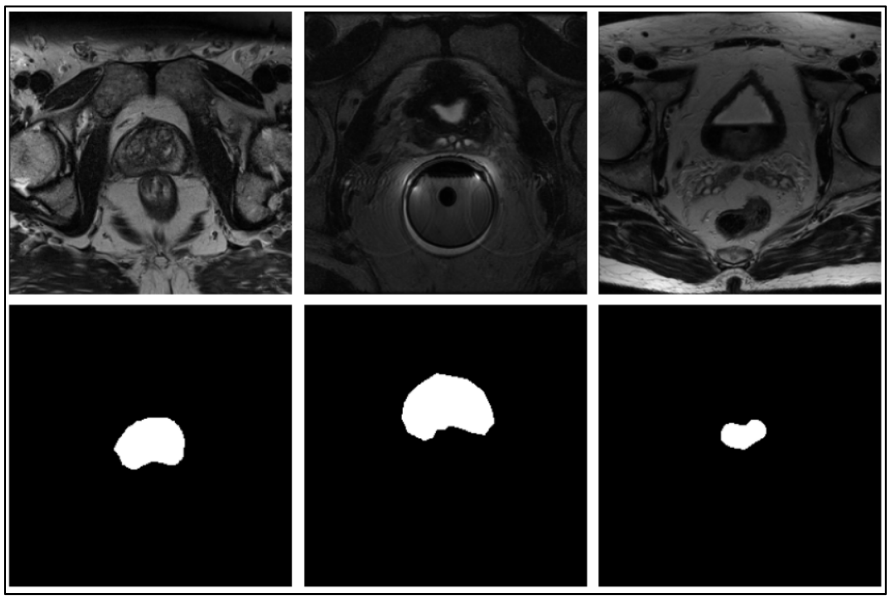

Figure 3.2 Aperçu des tranches d'un scan de ACDC,
ainsi que les masques de référence correspondants

Nous avons effectué les mêmes prétraitements que pour ACDC. 35 scans ont été dédiés à l'entrainement (soient 946 tranches), cinq à la validation (soient 156 tranches), et dix scans pour les tests (275 tranches). Aucun post-traitement n'est effectué, pour nos jeux de données.

3.1.4.3 Les configurations des modèles entrainés

Nous contrôlons la capacité brute des modèles comparés, en fixant une architecture (ENet) et un optimisateur uniques. Tous les modèles de segmentation sont entraînés pendant 100 époques. Ainsi donc, pour la borne inférieure (*Fs*), l'optimisateur *Adaptative Moment*

*Estimation (*ADAM), proposé par Kingma et al. (2015), est configuré suivant les valeurs expérimentales rapportées par Paszke et al. (2016, p. 7). Le *weight decay* est fixé à 2e-4, le couple bêta est fixé à (0.9, 0.99), et la taille des lots utilisés en entrainement est fixée à dix. Seule la valeur du taux d'apprentissage est optimisée (voir le Tableau-A II-1).

Pour le modèle entrainé avec des boîtes englobantes exactes, la borne supérieure (*Hybrid*), nous gardons la même configuration que le modèle *Fs* : seuls les hyperparamètres liés à la supervision hybride sont optimisés (voir le Tableau-A II-2).

Similairement, notre modèle (*Ours*) garde la même configuration que les modèles *Fs* et *Hybrid* : seuls les hyperparamètres liés aux potentielles erreurs de l'inférence des boîtes sont optimisés (voir le Tableau-A II-3). Ici, le réseau auxiliaire utilise aussi l'architecture ENet, mais avec une seule carte d'activation à la dernière couche (cas binaire). L'optimisateur utilisé est ADAM, configuré comme pour *Fs,* mais avec un taux d'apprentissage (re)optimisé (voir le Tableau-A II-4).

Pour la méthode préexistante (*size solus)* qui représente l'état de l'art établi par Kervadec et al. (2019a), nous utilisons la configuration du modèle *Fs* et nous optimisons les hyperparamètres liés à la semi-supervision (voir le Tableau-A II-5). Comme pour le modèle précédent, le taux d'apprentissage du réseau auxiliaire est aussi optimisé (voir le Tableau-A II-6), et ses autres hyperparamètres sont fixés suivant (Kervadec et al., 2019). De plus, nous normalisons la pénalité $\mathcal{L}_{Naive}$ (voir la section 1.2.2) par le nombre de tranches par lot. Ceci est fait parce que nous utilisons des lots composés par dix tranches, alors que la publication originale n'utilise qu'une seule tranche par lot.

Pour le modèle de la génération de pseudos-labels par apprentissage adverse (*self-training*), ce dernier est entrainé tel que spécifié par Hung et al. (2018), en utilisant la configuration du modèle *Fs*, et en optimisant les hyperparamètres liés à la semi-supervision (voir le Tableau-A II-7). De manière similaire, le taux d'apprentissage du réseau auxiliaire (le discriminateur)

est optimisé (voir le Tableau-A II-7). Notons que la normalisation des fonctions de coût est faite par défaut dans le code source officiel emprunté, suivant le nombre de tranches par lot.

3.2 La comparaison à l'état de l'art

Pour ces expérimentations, les variables dépendantes sont les moyennes et les écarts-types des métriques obtenues sur l'ensemble de test ; et la variable indépendante est la proportion des scans annotés utilisés en entrainement.

Dans les tableaux qui suivent, la notation N_1/N désigne la proportion de scans annotés utilisés, par rapport à la totalité des scans d'entrainement. Les valeurs mises entre les parenthèses sont les écarts-types du DSC et de la HD ; et les valeurs en caractères gras soulignent la méthode de semi-supervision qui donne les meilleures performances, pour la scission de données considérée.

3.2.1.1 Les résultats sur le jeu de données ACDC

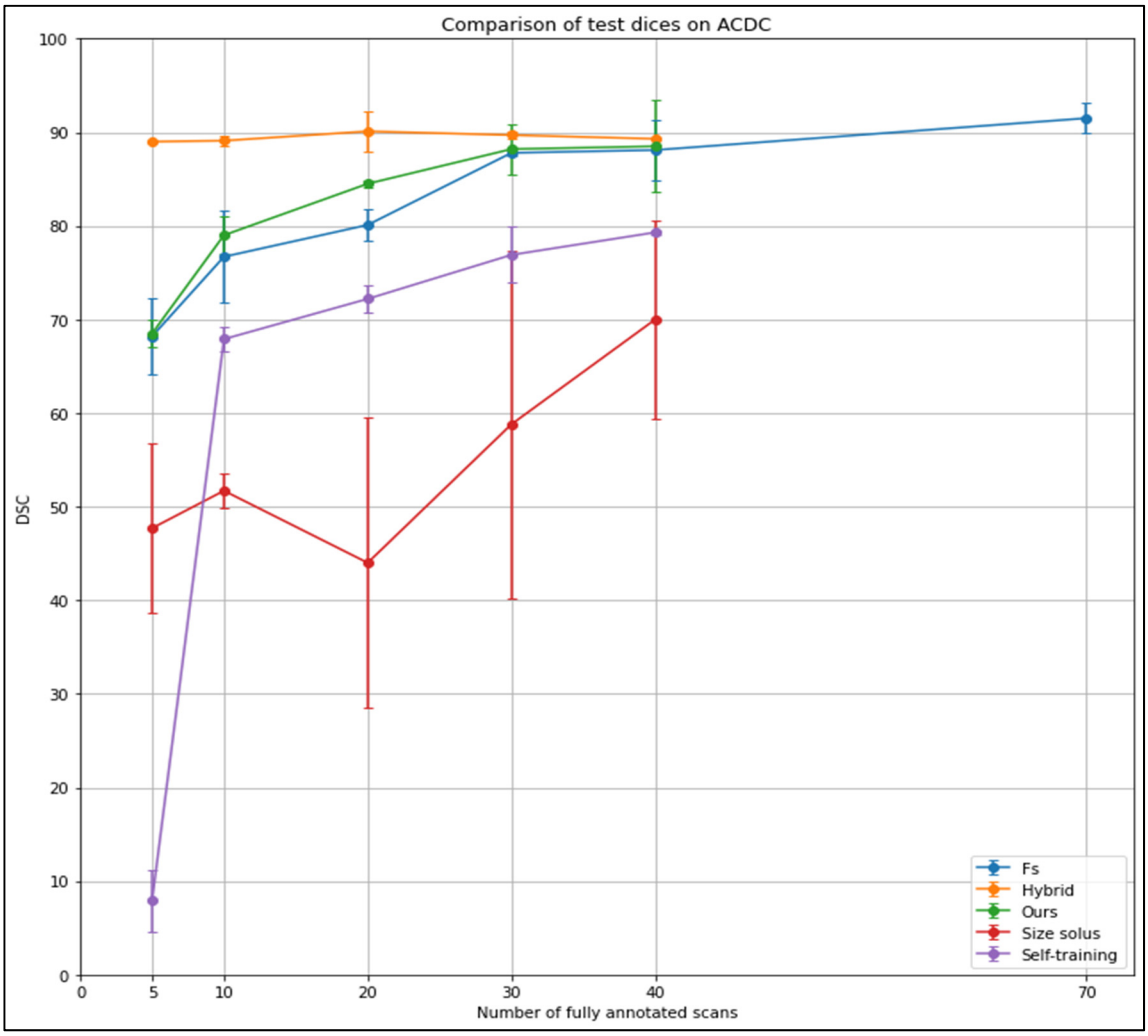

Figure 3.3 DSC(s) moyens, pour la comparaison des modèles,
lorsqu'on varie le nombre de scans annotés,
sur le jeu de données ACDC

Comme on peut l'observer sur la figure précédente, globalement, notre modèle donne un DSC moyen supérieur à ceux des autres modèles, excepté *Hybrid* ; mais il ne se rapproche de la borne supérieure qu'à partir de 30 scans annotés.

Tableau 3.1 Les résultats de la comparaison des modèles sur le jeu de données ACDC, suivant le DSC et la distance HD

$\frac{N_1}{N}$	*Fs*		*Hybrid*		*Ours*		*Size solus*		*Self-training*	
	DSC	HD	DSC	HD	DSC	HD	DSC	HD	DSC	HD
5/70	68.2 (4.0)	57.6 (33.7)	**89.0 (0.2)**	**27.7 (4.6)**	68.5 (1.4)	37.8 (0.1)	47.7 (9.1)	71.5 (50.7)	7.9 (3.3)	253.4 (56.0)
10/70	76.7 (4.9)	52.8 (57.8)	**89.1 (0.5)**	**26.2 (13.5)**	79.0 (2.0)	46.4 (10.4)	51.7 (1.8)	67.4 (63.7)	67.9 (1.3)	103.0 (0.6)
20/70	80.1 (1.7)	41.8 (37.6)	**90.1 (2.1)**	**18.3 (8.7)**	84.5 (0.4)	29.7 (5.1)	44.0 (15.5)	43.9 (32.6)	72.2 (1.4)	93.8 (0.5)
30/70	87.8 (0.4)	30.0 (6.6)	**89.7 (0.4)**	**10.1 (3.5)**	88.2 (2.7)	15.8 (4.5)	58.8 (18.6)	68.5 (53.1)	76.9 (3.0)	82.6 (0.6)
40/70	88.1 (3.2)	28.0 (12.3)	**89.3 (0.1)**	**13.0 (6.8)**	88.5 (4.9)	13.2 (5.8)	70.0 (10.6)	39.2 (21.6)	79.3 (0.2)	62.5 (0.5)
70/70	**91.5 (1.6)**	**11.6 (9.9)**	Ne s'applique pas							

En n'utilisant que 5 scans annotés, on remarque que le modèle hybride atteint 97.2% de performances du modèle complètement supervisé (avec 70/70 scans annotés).

Notre modèle reste à moins de 2% près des performances de la borne supérieure (*Hybrid),* à partir de 30 scans annotés. De plus, son DSC moyen est partout supérieur à ceux de tous les modèles (excepté la borne supérieure). Par ailleurs, l'écart entre le DSC moyen de notre modèle et celui du modèle Fs est plus grand pour la scission 20/70 : soit 4.4% de plus. On remarque aussi que la distance moyenne HD de notre modèle est partout inférieure à celle de *Fs,* et très proche de celle de la borne supérieure, pour la scission 40/70 (l'avant-dernière ligne du tableau ci-haut).

Pour ce jeu de données, les hyperparamètres optimisés suivant l'évolution du DSC moyen sur l'ensemble de validation (après chaque époque d'entrainement), sont présentés dans l'ANNEXE II. Les performances des réseaux auxiliaires sont mises dans l'ANNEXE III.

3.2.1.2 Les résultats sur le jeu de données PROMISE 12

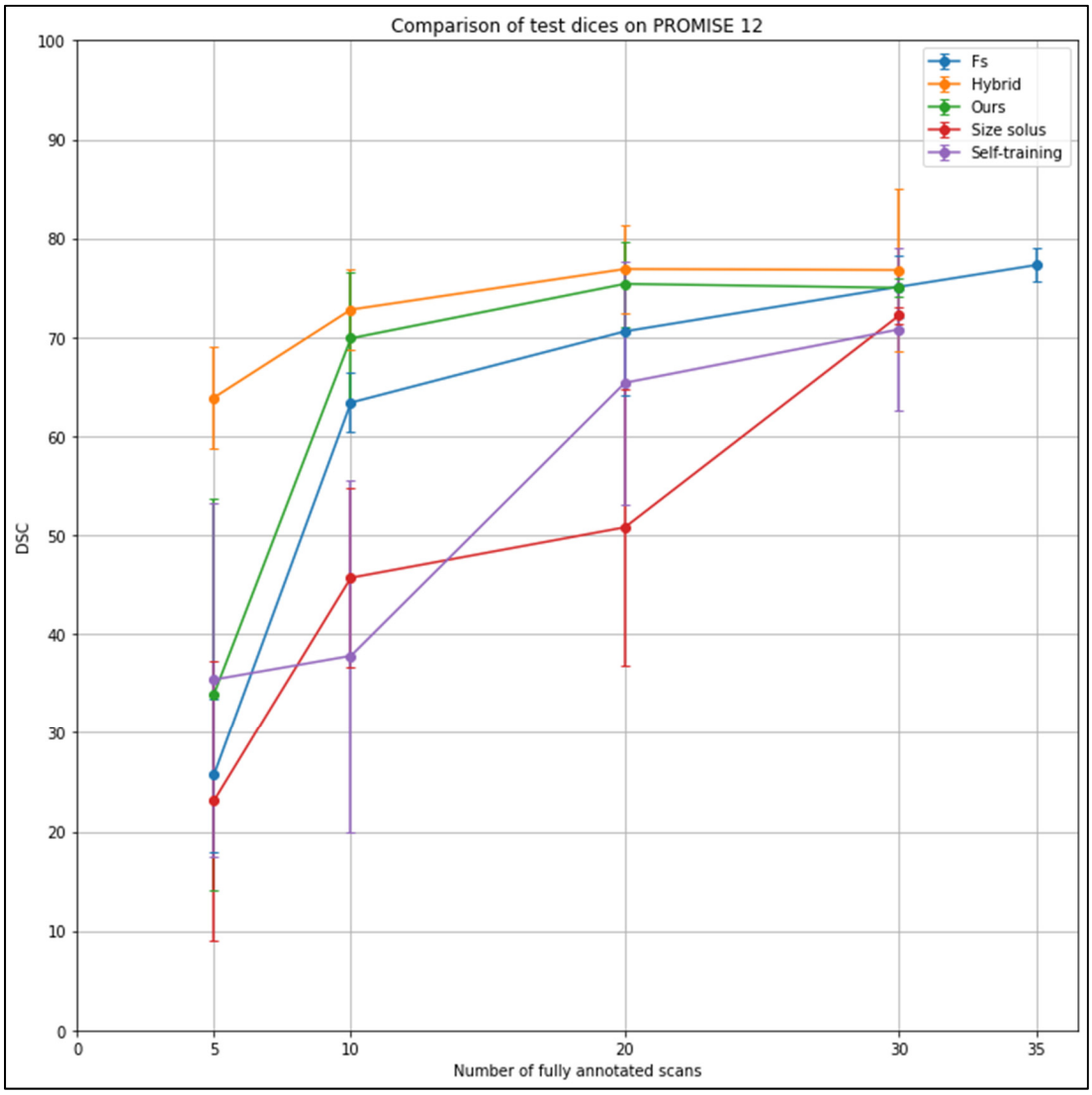

Figure 3.4 DSC(s) moyens, pour la comparaison des modèles,
lorsqu'on varie le nombre de scans annotés,
sur le jeu de données PROMISE 12

Suivant cette figure, notre modèle donne un DSC supérieur à ceux des autres modèles, excepté le modèle *Hybrid,* et le modèle *Fs* (pour la scission 30/35). L'écart entre notre modèle et la borne supérieur (*Hybrid*) est plus grand que sur le jeu de données ACDC. Notons que sur la scission 5/35, le modèle adverse (*Others 2*) donne des performances supérieures à celles de notre modèle.

Tableau 3.2 Les résultats de la comparaison des modèles sur le jeu de données PROMISE 12, suivant le DSC et la distance HD

$\frac{N_1}{N_2}$	*Fs*		*Hybrid*		*Ours*		*Size solus*		*Self-training*	
	DSC	HD	DSC	HD	DSC	HD	DSC	HD	DSC	HD
5/35	25.7 (7.7)	96.8 (23.9)	**63.9 (5.2)**	**22.6 (6.6)**	33.9 (19.8)	86.2 (42.3)	23.1 (14.1)	93.6 (14.3)	35.4 (17.9)	83.7 (10.7)
10/35	63.4 (3.0)	68.6 (22.7)	**72.8 (4.1)**	**57.7 (27.5)**	69.9 (6.7)	30.9 (13.2)	35.7 (9.0)	77.8 (47.7)	37.8 (17.8)	73.2 (55.7)
20/35	70.6 (6.4)	79.3 (2.6)	**76.9 (4.4)**	**23.8 (2.7)**	75.4 (4.3)	36.7 (3.1)	50.8 (14.0)	51.3 (31.6)	65.4 (12.3)	65.0 (3.6)
30/35	75.1 (3.1)	38.8 (7.5)	**76.8 (8.2)**	**27.8 (9.7)**	75.0 (0.9)	57.8 (24.1)	72.2 (0.8)	45.62 (4.66)	70.8 (8.2)	52.6 (2.8)
35/35	**77.3 (1.7)**	**32.5 (11.2)**	Ne s'applique pas							

De façon similaire aux observations faites sur le jeu de données ACDC, sur la scission 5/35, le modèle hybride atteint 82.6% de performances du modèle complètement supervisé. De plus, notre modèle s'approche à 2% près des performances de la borne supérieure à partir de la scission 20/35.

Des plus amples détails concernant l'optimisation des hyperparamètres sont présentés dans l'ANNEXE II. La section qui suit présente les résultats des expérimentations menées pour déterminer les contributions des composantes de la fonction de coût utilisée.

3.3 L'ablation des composantes de la fonction de coût

En gardant la même configuration que pour la comparaison à l'état de l'art. L'ablation de la fonction de coût de notre modèle est faite sur les deux scissions ayant cinq scans annotés (5/70 pour ACDC, et 5/35 pour PROMISE 12). Ces deux scissions sont les plus intéressantes dans cette recherche parce nous souhaitons réduire le nombre d'images annotées nécessaires pour l'entrainement : ce sont les scissions où le nombre d'annotations produites par des experts humains est le plus faible, parmi les cas de figure considérés. Les hyperparamètres λ_{Size}, λ_{Out} et λ_{In} sont modifiés pour cibler les composantes de la fonction de coût à

désactiver : ils prennent la valeur zéro, lorsque la composante en question est désactivée. Nous rapportons les mêmes métriques et statistiques que dans la section précédente.

Comme le montre le tableau suivant, les meilleures performances sont obtenues en utilisant les combinaisons numéro six et huit, sur le jeu de données ACDC ; tandis que ce sont les combinaisons cinq et sept qui l'emportent sur PROMISE 12.

Tableau 3.3 Les résultats de l'ablation de la fonction de coût, suivant l'indice DSC et la distance HD, sur les deux jeux de données

N°	Composantes actives	ACDC (5/75)		PROMISE 12 (5/30)	
		DSC	HD	DSC	HD
1	$\mathcal{L}_{GT}$	62.9 (0.7)	90.8 (0.6)	28.3 (24.4)	99.4 (10.4)
2	$\mathcal{L}_{GT} + \mathcal{L}_{Size}$	67.8 (1.8)	97.8 (8.9)	33.4 (19.2)	85.5 (67.0)
3	$\mathcal{L}_{GT} + \mathcal{L}_{Out}$	62.7 (7.4)	123.1 (5.6)	26.3 (26.7)	169.0 (23.2)
4	$\mathcal{L}_{GT} + \mathcal{L}_{In}$	65.2 (0.7)	100.0 (10.8)	30.5 (31.2)	131.2 (19.4)
5	$\mathcal{L}_{GT} + \mathcal{L}_{Size} + \mathcal{L}_{Out}$	66.0 (3.5)	68.7 (1.5)	39.2 (23.8)	**74.9 (52.7)**
6	$\mathcal{L}_{GT} + \mathcal{L}_{Out} + \mathcal{L}_{In}$	**70.1 (0.6)**	79.4 (9.5)	34.2 (23.7)	85.1 (9.8)
7	$\mathcal{L}_{GT} + \mathcal{L}_{Size} + \mathcal{L}_{In}$	68.8 (2.5)	111.6 (28.4)	**39.7 (29.8)**	90.6 (46.2)
8	Toutes les composantes	68.5 (1.4)	**37.8 (0.1)**	33.9 (19.8)	86.2 (42.3)

Notons que sur le jeu de données PROMISE 12, la combinaison numéro 7 ($\mathcal{L}_{GT}$, $\mathcal{L}_{Size}$ et $\mathcal{L}_{In}$) dépasse de 14% les performances (DSC) du modèle *Fs*, voir le Tableau 3.2. Cette combinaison dépasse aussi les performances qu'on obtient en utilisant toutes les composantes. Sur le jeu de données ACDC, c'est la combinaison 6 qui l'emporte.

La section qui suit présente les résultats de l'étude qui concerne la généralisation des performances, par rapport aux architectures de segmentation usuelles.

3.4 La variation de l'architecture de segmentation

La configuration utilisée est la même que pour la comparaison à l'état de l'art (section 3.2), et seule l'architecture de segmentation est variée. Nous présentons les mêmes statistiques que dans les sections précédentes, et nous marquons en gras les grandeurs optimales. Les durées affichées (Entrainement & Inférence) sont le temps qu'a duré le processus d'apprentissage pour le réseau de neurones concerné, et le temps moyen utilisé pour segmenter un scan composé d'une trame ayant dix tranches.

Tableau 3.4 Les performances des différentes architectures,
sur les scissions avec cinq scans annotés

	Architecture	DeepLabV3	ENet	FCN	UNet
ACDC (5/75)	DSC	67.1 (2.3)	68.5 (1.4)	68.4 (5.0)	**77.3 (3.4)**
	HD	**36.5 (28.1)**	37.8 (0.1)	61.9 (15.8)	84.8 (3.8)
	Entraînement	2h 33m 35s	2h 18m 24s	**1h 49m 05s**	3h 06m 50s
PROMISE (5/35)	DSC	34.9 (20.3)	33.9 (19.8)	42.4 (30.2)	**46.5 (27.9)**
	HD	**72.8 (63.0)**	86.2 (42.3)	80.8 (43.5)	105.3 (26.8)
	Durée	1h 59m 30s	1h 50m 39s	**1h 29m 44s**	2h 29m 29s
Temps d'inférence		195ms	29.5ms	5.32ms	345ms

Les différences en termes de performance (DSC) varient entre 1.3% et 10.2%. Remarquons que le plus haut DSC est obtenu avec UNet, et que la plus basse HD est obtenue avec DeepLabV3 (ResNet50). Pour mieux illustrer ces tendances, nous avons identifié les tranches (appartenant à l'ensemble de test) qui ont été remarquablement (très bien ou très mal) segmentées par l'architecture DeepLabV3 (ResNet50), suivant la distance HD, pour les deux jeux des données utilisés.

Ainsi donc, sur les figures à la page qui suit, la première ligne montre les tranches les moins bien segmentées par DeepLabV3, et la deuxième montre les mieux segmentées. Les contours en rouge sont les contours tracés par les experts humains, et les contours en bleu dénotent les

prédictions des différentes architectures : les colonnes désignent respectivement (de gauche à droite) les prédictions obtenues avec les architectures DeepLabV3, ENet, FCN, et UNet.

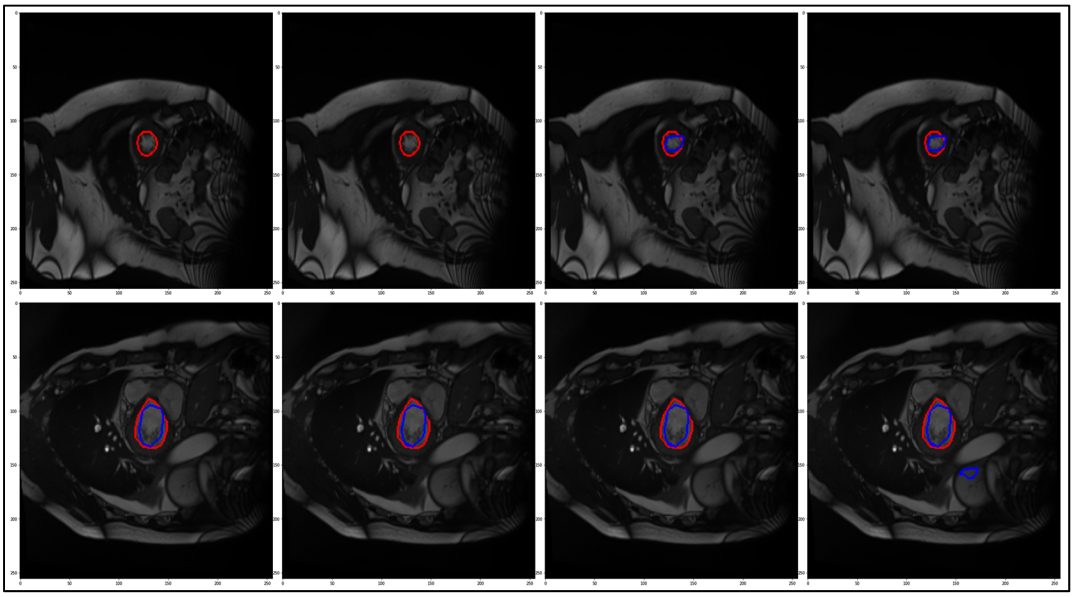

Figure 3.5 Illustration de quelques tranches remarquablement segmentées, suivant la distance HD, sur ACDC

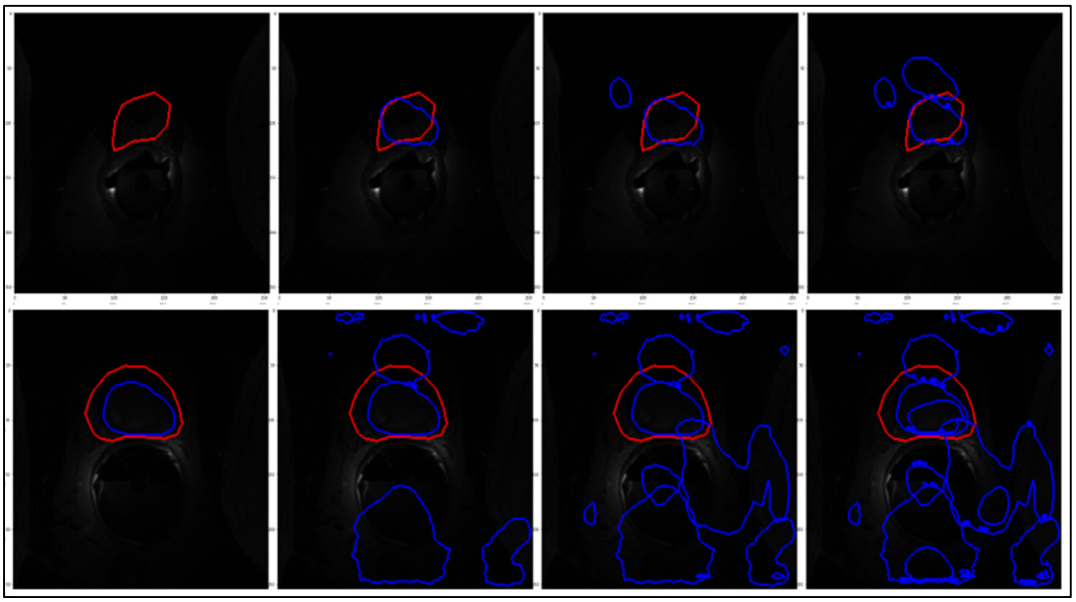

Figure 3.6 Illustration de quelques tranches remarquablement segmentées, suivant la distance HD, sur PROMISE 12

Comme on peut l'observer sur les figures précédentes, DeepLabV3 a tendance à ne détecter aucun pixel dans la région d'intérêt (pour les mauvaises segmentations), alors que UNet a tendance à détecter beaucoup des pixels éloignés de la région. Le chapitre qui suit est dédié à l'interprétation et à la discussion des résultats.

CHAPITRE 4

DISCUSSION DES RESULTATS

Comme précisé dans l'introduction, les expérimentations sont menées afin de déterminer si cette nouvelle stratégie d'apprentissage par curriculum est viable, avant de poursuivre avec une étude moins simplificatrice.

4.1 L'interprétation des résultats de la comparaison à l'état de l'art

En ce qui concerne la comparaison à l'état de l'art, globalement, les résultats expérimentaux montrent que la stratégie d'apprentissage donne des meilleurs résultats que ceux du modèle de l'apprentissage adverse. Toutefois, pour la scission avec cinq scans annotés sur 40 (voir le Tableau 3.2), les résultats du modèle de l'apprentissage adverse sont supérieurs.

Pour expliquer ce phénomène, nous pensons que la nature des données fait chuter les performances du modèle auxiliaire ; et que les fausses prédictions de ce dernier font baisser les performances de la segmentation :

- seul le jeu de données et varié lorsqu'on observe la chute, et suite à cela le modèle auxiliaire n'atteint pas 30% en IoU pour la scission 5/35 (voir la Figure-A III-4) : La relation de précédence est vérifiée ;

- dans toutes nos expérimentations, les performances du modèle principal varient dans le même sens et proportionnellement par rapport à celles du modèle auxiliaire : la relation de covariance est vérifiée ;

- nous contrôlons toutes les autres variables (fixées), et nous obtenons des gains lorsque les performances du modèle auxiliaire sont bonnes : la seule explication est que ce dernier n'arrive pas à bien généraliser sur les scissions 5/35 et 10/35.

En ce qui concerne la comparaison au modèle de l'apprentissage par curriculum de Kervadec et al. (2019a), nous limitons nos observations parce qu'on n'arrive pas à atteindre les résultats publiés. Les éventuelles causes sont le fait que nous utilisons des plus grands lots ainsi qu'un optimisateur avec un *weight decay* différent. De plus, leurs scissions (sur ACDC) utilisent 5 scans de plus pour l'entrainement.

Cela se traduit également par le fait que le réseau auxiliaire correspondant n'arrive pas à généraliser : la moyenne de la valeur absolue de la différence, entre les prédictions et la taille réelle, reste supérieure à 150 (voir la Figure-A III-1).

Si l'on se fie à la fonction de coût (la pénalité quadratique), voir la section 1.2.2, les mauvaises prédictions font croître le gradient brutalement. Cette remarque est appuyée par le Tableau 2.4 publié par Kervadec (2021, p. 41), parce qu'il indique que l'intervalle idéal des bornes imposées doit être à dix pour cent près de l'exacte.

4.2 L'interprétation des résultats de l'ablation de la fonction de coût

En ce qui concerne l'évaluation de ce qu'apporte chaque composante, sur le jeu de données ACDC, nous remarquons un gain pour chacune d'elles (voir le Tableau 3.3), par rapport à l'utilisation de $\mathcal{L}_{GT}$ seule, sauf pour la combinaison N° 3 ($\mathcal{L}_{GT} + \mathcal{L}_{Out}$).

Cela dit, même si l'utilisation de la combinaison N° 6 ($\mathcal{L}_{GT} + \mathcal{L}_{Out} + \mathcal{L}_{In}$) semble donner les meilleurs résultats (selon l'indice DSC), nous remarquons que la distance HD est plus petite pour la combinaison qui utilise toutes les composantes. Par ailleurs l'a priori de la taille est celle qui apporte le plus de gain, et l'utilisation de l'a priori des limites externes ($\mathcal{L}_{Out}$) diminue les performances lorsqu'aucun autre a priori n'est adjoint (ensemble).

Sur le jeu des données PROMISE 12, le résultat saillant est le fait que ce n'est plus la même combinaison de composantes qui donne les meilleurs résultats. Ici encore, l'apriori exprimée par $\mathcal{L}_{Out}$ dévient nuisible lorsqu'elle est adjointe seule.

Comme toutes les composantes sont susceptibles d'améliorer les performances, nous pensons que les apparentes nuisances de la contrainte $\mathcal{L}_{Out}$ peuvent être expliquées par le fait que nous utilisons une architecture simplifiée (pour inférer les boîtes englobantes).

En comparant les deux jeux de données, nous attribuons au contenu particulier de PROMISE 12 les différences notables par rapport à la magnitude des performances. Comme cela est mentionné par Litjens et al. (2013, p. 3), ce jeu de données a été assemblé pour poser un défi concernant la robustesse, en variant considérablement les aspects : tailles, éclairements, positions, cliniques (lieux d'acquisition), etc.

4.3 L'interprétation des résultats de la variation de l'architecture

Les résultats montrent qu'on obtient des performances relativement proches, pour les architectures ENet, FCN-ResNet50 et DeepLabV3-ResNes50 (voir Tableau 3.4). Toutefois, comme indiqué dans la section 3.4, l'architecture UNet donne un grand DSC (souhaité) ainsi qu'une grande distance HD (non-souhaitée).

Nous pensons que ces deux tendances peuvent être expliquées par le fait que l'indice DSC ne prend pas en compte la distance spatiale entre les voxels, contrairement à la distance HD. En effet, l'indice DSC peut avoisiner 80% si la zone d'intérêt est (presque) parfaitement segmentée en dépit des éventuels pixels faussement prédits comme positifs, tant que leur effectif ne dépasse pas la moitié du nombre des éléments qui composent la zone d'intérêt (voir les équations dans la section 3.1.3). L'image dans la quatrième colonne de la deuxième ligne illustre un tel cas, sur la Figure 3.5.

4.4 Les limites et l'importance

La limite interne principale est liée l'absence d'une gestion de l'angle de rotation de la boîte englobante, comme pour les travaux de Kervadec et al. (2020b). De plus le modèle auxiliaire n'effectue aucune détection d'organe, avant de proposer une boîte englobante. Par ailleurs, la multiplicité des hyperparamètres introduits nécessite un réglage (et donc un temps d'entrainement non négligeable), parce que les valeurs optimales dépendent des jeux de données (voir le Tableau-A II-3).

En ce qui concerne la méthodologie expérimentale, en particulier la variabilité des résultats, nous n'avons répété ces expérimentations que trois fois (avec trois initialisations aléatoires différentes).

Tout de même, le fait que la version simplifiée de cette méthode dépasse (globalement) les performances des méthodes actuelles (*baselines* retenues) est une première étape favorable à l'utilisation de la stratégie. La méthode peut sembler être un retour en arrière, par rapport à la supervision faible telle qu'étudiée par Kervadec et al. (2020b), mais l'étude de Bellver, Salvador, Torres, & Giro-i-Nieto (2019) a montré que l'apprentissage semi-supervisé donne des meilleurs résultats, si le nombre d'annotations est très limité, entre autres.

Par ailleurs, le modèle proposé dans cette recherche ne nécessite aucune augmentation des données, et l'entrainement ne prend que 100 époques, pour chacun de deux modèles. La partie qui suit présente les conclusions et les recommandations tirées de cette recherche.

CONCLUSION ET RECOMMANDATIONS

Dans ce projet de recherche, nous avons cherché à contribuer à la diminution du nombre d'images nécessaires pour l'entrainement des réseaux de neurones, en segmentation d'images médicales, afin de réduire les coûts (charge de travail, temps et argent) liés à l'annotations des jeux de données.

Notre modèle simplifié donne des résultats globalement supérieurs à ceux des autres méthodes considérées. En particulier, nous observons une augmentation du DSC moyen allant jusqu'à 4.4%, lorsqu'on utilise que 20 scans annotés sur le jeu de données ACDC. Par ailleurs, en n'utilisant que 5 scans annotés, notre borne supérieure atteint plus de 97.2% des résultats possibles avec la supervision totale.

Nous concluons que cette stratégie de semi-supervision par curriculum basée sur l'inférence des boîtes englobantes est viable. Plus précisément, dans nos expérimentations, cette stratégie donne des meilleures performances que les précédentes publications (en segmentation par curriculum avec un réseau auxiliaire), tout en nécessitant un temps d'entrainement inférieur.

Toutefois, en raison des variabilités observées et des simplifications effectuées, nous recommandons deux études plus étendues, pour cerner l'étendu des performances effectivement accessibles avec cette stratégie de supervision.

La première étude concerne la segmentation par curriculum avec un réseau de neurones auxiliaire séparé : en effet, le réseau de neurones entièrement convolutif utilisé comme modèle auxiliaire n'est pas la meilleure architecture, a priori. La recherche en détection d'objets préfère les *Region proposal network* (RPN), le *Region based fully convolutional network* (R-FCN) et les *Single shot detector* (SSD). Cela étant, une étude sur un plus grand jeu de données pourrait permettre de mieux cerner les performances effectivement accessibles. Cela se ferait en variant les architectures du modèle auxiliaire, la quantité d'annotations, et en optimisant les hyperparamètres pertinents : le nombre de régions

proposées, les tailles des régions proposées, la profondeur de l'extracteur des caractéristiques, etc. L'utilisation de telles architectures serait pertinente parce que l'organe à segmenter ne peut pas toujours être encadré par une seule boîte englobante serrée : plusieurs composantes connectées peuvent être spatialement éloignées. Tout de même, dans les cas où l'organe à segmenter serait toujours représenté par une seule composante connectée (ce qui n'est pas le cas pour les organes pairs, en particulier : les reins, les poumons, etc.) le réseau de neurones auxiliaire serait une simple architecture de régression (pour prédire deux coins de l'unique boîte englobante).

De plus, un plus grand jeu de données permettrait une validation croisée en *k-folds*, voire même une seconde boucle d'optimisation (*Nested cross-validation*), afin d'abstraire le choix des hyperparamètres des étalons et pour éliminer la différence entre les scissions qu'utilisent les différentes publications. En effet, même si la proportion d'annotations peut être est la même, les images dans la scission ne sont pas nécessairement les mêmes.

La deuxième étude que nous recommandons concerne l'architecture *Mask-RCNN* proposée par Hung et al. (2018), même si l'apprentissage multitâche est en dehors du cadre de ce mémoire. En effet, il serait a priori possible de n'utiliser qu'une seule architecture pour les deux tâches de notre stratégie de curriculum (en semi-supervision) : pour diminuer le temps d'entrainement et pour mieux régulariser (potentiellement).

Cette idée est à noter attentivement, étant donné que les travaux de Wang et al. (2021) ont rapporté un DSC de 69.4%, sur une scission de 5/75 (ACDC), alors que notre modèle obtient 77.3% avec l'architecture UNet. L'étude de Wang et al. (2021) utilise une version multitâche de UNet, dont la branche auxiliaire sert à prédire le centroïde et la taille, pour contraindre la taille et la localisation des segmentations, avec des simples pénalités quadratiques (voir L_{Naive}, dans la section 1.2.2).

ANNEXE I

VISUALISATION DES DONNÉES

Les tableaux suivants présentent les statistiques des a priori(s) pertinents, pour chacune des trois initialisations utilisées afin de scinder les données. La proportion des tranches positives fait référence au rapport entre le nombre de tranches qui contiennent des pixels de l'organe, par rapport au nombre (total) des tranches.

Les valeurs affichées dans les tableaux qui suivent sont les moyennes des a priori(s) calculées en réunissant toutes les tranches (sans distinguer les scans), ainsi que leurs écarts-types (mis entre les parenthèses).

Tableau-A I-1 Les statistiques des a priori(s) pour les scissions de données effectuées suivant la première initialisation sur ACDC

Scissions	Proportion des tranches positives	Coordonnée (x) du centroïde	Coordonnée (y) du centroïde	Taille de l'organe
5/70	0.98	127.39 (4.51)	129.31 (4.27)	1304.95 (577.81)
10/70	0.98	123.67 (8.57)	127.27 (6.23)	1453.91 (696.74)
20/70	0.94	121.44 (9.43)	127.53 (6.84)	1053.42 (732.95)
30/70	0.95	120.59 (8.73)	126.50 (6.98)	903.72 (675.37)
40/70	0.95	120.07 (10.78)	125.39 (11.79)	931.27 (635.98)
70/70	0.95	120.85 (11.86)	125.58 (10.82)	838.74 (573.67)

Tableau-A I-2 Les statistiques des a priori(s) pour les scissions de données effectuées suivant la première initialisation sur ACDC, (suite)

Scissions	Proportion des tranches positives	Coordonnée (x) du centroïde	Coordonnée (y) du centroïde	Taille de l'organe
Validation (5 scans)	0.95	122.11 (10.25)	130.84 (8.90)	1016.64 (674.75)
Test (25 scans)	0.96	123.09 (10.16)	125.34 (15.10)	957.93 (621.43)

Tableau-A I-3 Les statistiques des a priori(s) pour les scissions de données effectuées, suivant la première initialisation sur PROMISE 12

Scissions	Proportion des tranches positives	Coordonnée (x) du centroïde	Coordonnée (y) du centroïde	Taille de l'organe
5/35	0.53	122.05 (10.95)	124.84 (4.18)	3080.97 (1909.79)
10/35	0.54	121.03 (13.44)	125.98 (4.96)	2746.12 (1671.79)
20/35	0.6	122.47 (11.79)	127.95 (5.33)	2711.14 (1933.19)
30/35	0.59	122.95 (11.20)	127.97 (5.14)	2427.71 (1767.78)
35/35	0.59	123.21 (10.82)	128.13 (4.93)	2345.42 (1702.19)
Validation (5 scans)	0.5	113.75 (18.85)	127.40 (4.94)	1794.90 (738.11)
Test (10 scans)	0.52	115.44 (15.55)	126.99 (4.89)	2116.23 (1271.84)

Tableau-A I-4 Les statistiques des a priori(s) pour les scissions de données effectuées, suivant la deuxième initialisation sur ACDC

Scissions	Proportion des tranches positives	Coordonnée (x) du centroïde	Coordonnée (y) du centroïde	Taille de l'organe
5/70	0.96	121.31 (4.78)	130.62 (3.82)	1196.31 (515.66)
10/70	0.97	124.32 (7.82)	126.60 (9.78)	1357.10 (613.68)
20/70	0.96	124.84 (8.33)	128.09 (7.89)	1236.40 (738.06)
30/70	0.94	122.05 (9.31)	128.35 (8.90)	1031.59 (719.08)
40/70	0.95	120.64 (10.30)	126.35 (12.39)	995.47 (667.96)
70/70	0.95	121.42 (11.64)	126.89 (12.18)	875.23 (597.96)
Validation (5 scans)	0.97	122.25 (14.23)	119.58 (9.64)	1203.46 (642.26)
Test (25 scans)	0.95	121.46 (10.05)	123.55 (11.29)	814.58 (547.47)

Tableau-A I-5 Les statistiques des a priori(s) pour les scissions de données effectuées, suivant la deuxième initialisation sur PROMISE 12

Scissions	Proportion des tranches positives	Coordonnée (x) du centroïde	Coordonnée (y) du centroïde	Taille de l'organe
5/35	0.55	115.14 (12.90)	125.11 (3.90)	3035.71 (1726.18)
10/35	0.53	109.85 (16.89)	124.94 (4.75)	2811.58 (1580.02)

Tableau-A I-6 Les statistiques des a priori(s) pour les scissions de données effectuées, suivant la deuxième initialisation sur PROMISE 12, (suite)

20/35	0.58	116.05 (16.25)	126.76 (5.57)	2447.61 (1516.08)
30/35	0.56	118.47 (15.12)	127.19 (5.34)	2269.16 (1396.93)
35/35	0.56	119.48 (14.71)	127.68 (5.47)	2181.21 (1354.77)
Validation (5 scans)	0.56	128.42 (9.06)	128.02 (3.71)	3500.15 (2529.78)
Test (10 scans)	0.59	120.95 (7.40)	128.35 (3.25)	1699.22 (973.15)

Tableau-A I-7 Les statistiques des a priori(s) pour les scissions de données effectuées, suivant la troisième initialisation sur ACDC

Scissions	Proportion des tranches positives	Coordonnée (x) du centroïde	Coordonnée (y) du centroïde	Taille de l'organe
5/70	0.96	125.11 (7.50)	121.43 (16.77)	1307.30 (571.30)
10/70	0.97	123.61 (9.47)	124.61 (13.48)	1336.48 (570.88)
20/70	0.97	124.89 (8.93)	123.42 (11.15)	1304.28 (718.59)
30/70	0.94	121.15 (10.06)	125.43 (11.39)	1064.09 (730.05)
40/70	0.95	120.46 (11.62)	124.05 (13.72)	1023.95 (681.44)
70/70	0.95	121.40 (12.42)	125.08 (13.12)	907.80 (611.14)

Tableau-A I-8 Les statistiques des a priori(s) pour les scissions de données effectuées, suivant la troisième initialisation sur ACDC, (suite)

Validation (5 scans)	0.95	126.38 (10.93)	124.24 (8.24)	922.04 (639.13)
Test (25 scans)	0.94	120.83 (8.02)	127.79 (8.80)	785.16 (524.82)

Tableau-A I-9 Les statistiques des a priori(s) pour les scissions de données effectuées, suivant la deuxième initialisation sur PROMISE 12

Scissions	Proportion des tranches positives	Coordonnée (x) du centroïde	Coordonnée (y) du centroïde	Taille de l'organe
5/35	0.52	126.40 (8.17)	124.64 (4.27)	3047.61 (1930.53)
10/35	0.5	117.94 (17.39)	124.73 (4.04)	2943.88 (1663.39)
20/35	0.56	120.28 (14.70)	127.43 (5.50)	2451.54 (1570.47)
30/35	0.57	121.31 (13.46)	127.77 (4.98)	2251.13 (1441.89)
35/35	0.56	121.54 (12.98)	127.43 (4.91)	2162.72 (1401.16)
Validation (5 scans)	0.54	115.15 (16.90)	127.28 (6.21)	3168.47 (2711.45)
Test (10 scans)	0.59	120.97 (12.09)	129.51 (3.95)	2118.41 (1230.14)

ANNEXE II

OPTIMISATION DES HYPERPARMETRES

En évaluant le modèle sur l'ensemble de validation (après chaque époque), nous présentons les plus hautes valeurs des moyennes des métriques des performances (sans pourcentages).

Tableau-A II-1 La validation des hyperparamètres pertinents du modèle *Fs*, pour les différentes initialisations

lr	Initialisation	5e-2	5e-3	2.5e-4	5e-4	5e-5
DSC (Sur ACDC)	1	0.63	0.59	0.56	0.64	0.36
	2	0.67	0.64	0.53	0.67	0.60
	3	0.51	0.70	0.52	0.65	0.38
DSC (Sur PROMISE 12)	1	0.27	0.34	0.41	0.39	0.39
	2	0.49	0.48	0.55	0.46	0.55
	3	0.35	0.30	0.18	0.16	0.15

Tableau-A II-2 La validation des hyperparamètres pertinents du modèle *Hybrid*, pour les différentes initialisations

λ_{Size}		1.0	1e-1	1e-2	1.0	1.0	1.0	1.0	1.0	1.0	1.0
λ_{Out}		0.0	0.0	0.0	1.0	1e-1	1e-2	1.0	1.0	1.0	1.0
λ_{In}		0.0	0.0	0.0	0.0	0.0	0.0	1e-2	1e-3	1e-4	1e-5
DSC (Sur ACDC)	1	0.84	0.75	0.69	0.83	0.82	0.81	0.86	0.88	0.85	0.83
	2	0.85	0.78	0.69	0.83	0.86	0.84	0.87	0.88	0.82	0.84
	3	0.89	0.86	0.75	0.90	0.03	0.88	0.90	0.91	0.90	0.90
DSC (Sur PROMISE)	1	0.61	0.27	0.14	0.76	0.71	0.68	0.77	0.75	0.75	0.75
	2	0.58	0.57	0.51	0.66	0.66	0.65	0.69	0.71	0.65	0.63
	3	0.60	0.45	0.23	0.68	0.00	0.67	0.74	0.73	0.70	0.67

Tableau-A II-3 La validation des hyperparamètres pertinents de notre modèle, pour les différentes initialisations

$\delta_{minSize}$		0.1	0.5	0.5	0.5	0.5	0.5	0.5	0.5
$\delta_{maxSize}$		1.1	0.75	0.9	0.9	0.9	0.9	0.9	0.9
δ_{Out}		10	10	10	5	100	1000	10	10
δ_{In}		3	3	3	3	3	3	2	4
DSC (Sur ACDC)	1	0.72	0.72	0.75	0.74	0.74	0.77	0.74	0.77
	2	0.67	0.65	0.65	0.68	0.64	0.66	0.63	0.67
	3	0.80	0.82	0.81	0.78	0.80	0.81	0.82	0.82
DSC (Sur PROMISE)	1	0.41	0.42	0.42	0.39	0.41	0.40	0.42	0.40
	2	0.60	0.57	0.56	0.57	0.57	0.57	0.59	0.56
	3	0.21	0.18	0.42	0.46	0.40	0.51	0.39	0.31

Tableau-A II-4 La validation des hyperparamètres pertinents du réseau auxiliaire de notre modèle, pour les différentes initialisations

lr	Initialisation	5e-2	5e-3	2.5e-4	5e-4	5e-5
IoU (Sur ACDC)	1	0.47	0.49	0.35	0.56	0.48
	2	0.44	0.58	0.30	0.58	0.50
	3	0.35	0.58	0.34	0.56	0.48
IoU (Sur PROMISE 12)	1	0.20	0.31	0.32	0.36	0.29
	2	0.27	0.36	0.36	0.32	0.28
	3	0.23	0.30	0.16	0.28	0.11

Tableau-A II-5 La validation des hyperparamètres pertinents du modèle *size solus*, pour les différentes initialisations

λ_{Size}		1.0	0.0001	1e-5	1e-6
DSC (Sur ACDC)	1	0.03	0.03	0.03	0.52
	2	0.03	0.04	0.03	0.21
	3	0.03	0.03	0.06	0.28
DSC (Sur PROMISE)	1	0.03	0.03	0.04	0.32
	2	0.07	0.08	0.09	0.49
	3	0.06	0.06	0.08	0.40

Dans le tableau qui suit, la notation NaN signifie que le processus d'entrainement a divergé, à cause d'une magnitude du gradient inadéquate par exemple ; et *Abs (diff)* désigne la valeur moyenne maximale de la différence entre la traille réelle de l'organe et la taille prédite par le réseau auxiliaire.

Tableau-A II-6 La validation des hyperparamètres pertinents du réseau auxiliaire de *size solus*, pour les différentes initialisations

lr	Initialisation	5e-5	5e-6	5e-7	5e-8	5e-9
Abs (diff) (Sur ACDC)	1	NaN	533.80	NaN	572.05	NaN
	2	489.95	469.78	NaN	418.78	NaN
	3	489.52	492.60	NaN	611.27	NaN
Abs (diff) (Sur PROMISE 12)	1	NaN	NaN	845.21	1457.06	827.38
	2	NaN	NaN	2219.74	1835.12	1846.15
	3	NaN	NaN	1639.67	1982.24	1760.39

Dans le tableau ci-dessous, la variable lr_2 fait référence au taux d'apprentissage du réseau de neurones discriminateur, pour le modèle de l'apprentissage adverse. Les autres variables optimisées sont présentées dans la section 1.2.1. Le tableau ci-dessous ne rapporte que les hyperparamètres qui ont convergés correctement.

Tableau-A II-7 La validation des hyperparamètres pertinents du modèle du *self-training* (adverse), pour les différentes initialisations

λ_{adv_1}		1	0.01	0.01	0.1	0.1
λ_{adv_2}		0.001	0.001	0.01	0.001	0.001
λ_{Semi}		0.1	0.01	0.1	0.1	0.1
lr_2		0.01	1e-5	0.0001	1e-5	0.0001
DSC (Sur ACDC)	1	0.00	0.00	0.00	0.15	0.10
	2	0.00	0.00	0.00	0.04	0.08
	3	0.04	0.00	0.10	0.03	0.06
DSC (Sur PROMISE)	1	0.12	0.16	0.19	0.00	0.00
	2	0.39	0.00	0.00	0.06	0.07
	3	0.06	0.00	0.00	0.27	0.00

ANNEXE III

PERFORMANCES DES RÉSEAUX AUXILIAIRES

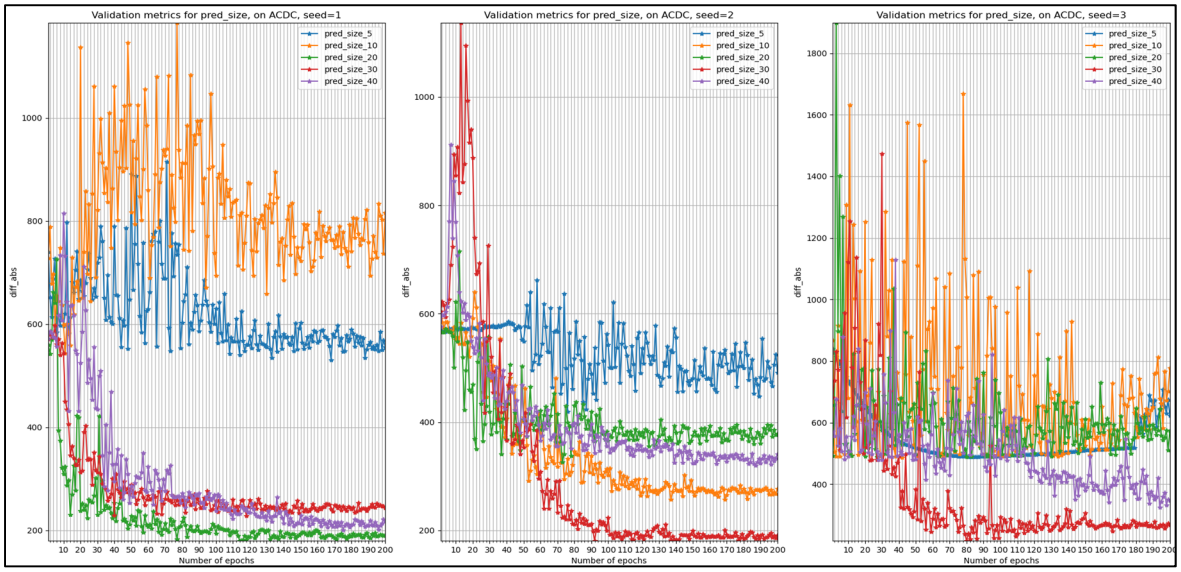

Figure-A III-1 Les valeurs absolues des différences entre les tailles vraies et les tailles prédites par le réseau auxiliaire de *size solus*, sur ACDC (5/70)

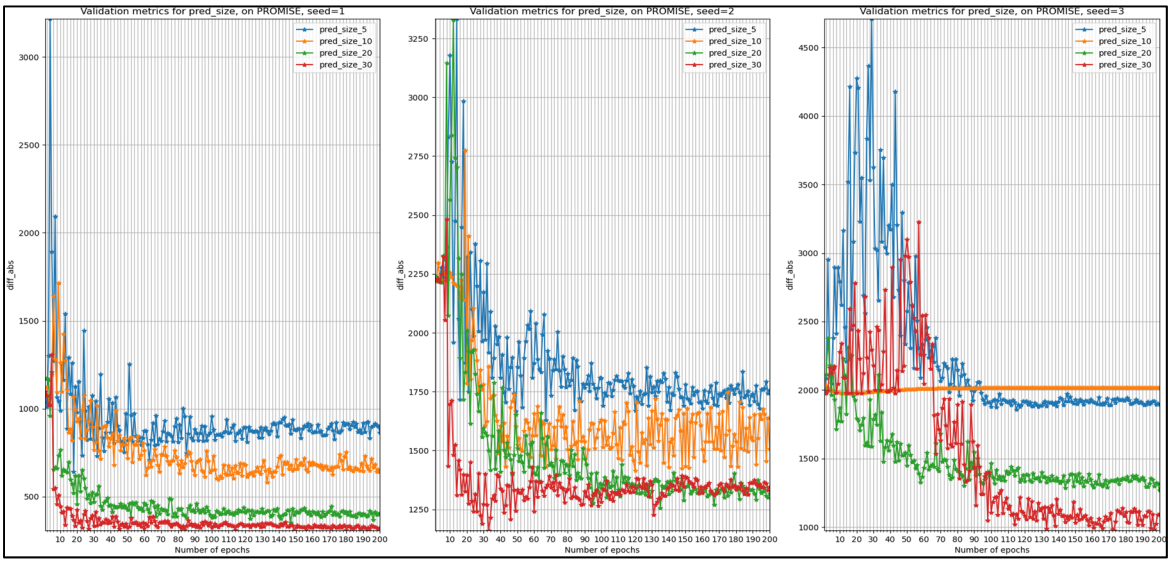

Figure-A III-2 Les valeurs absolues des différences entre les tailles vraies et les tailles prédites par le réseau auxiliaire de *size solus*, sur PROMISE 12 (5/35)

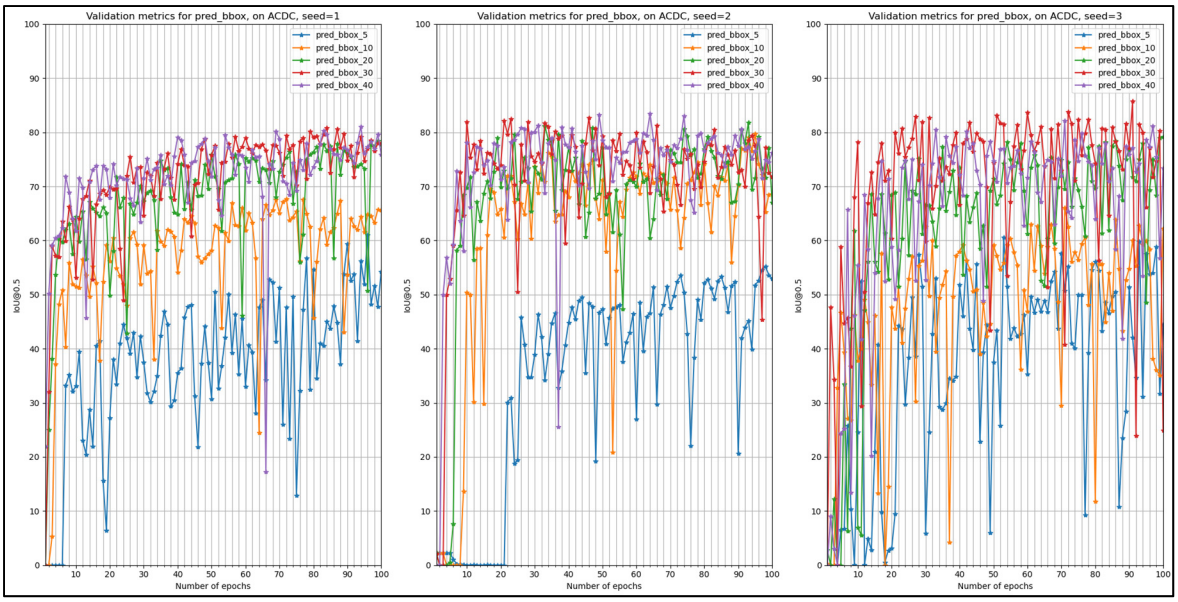

Figure-A III-3 Les valeurs moyennes de l'indice IoU entre les masques des boîtes englobantes vraies et ceux des pseudos boîtes englobantes, sur ACDC (5/70)

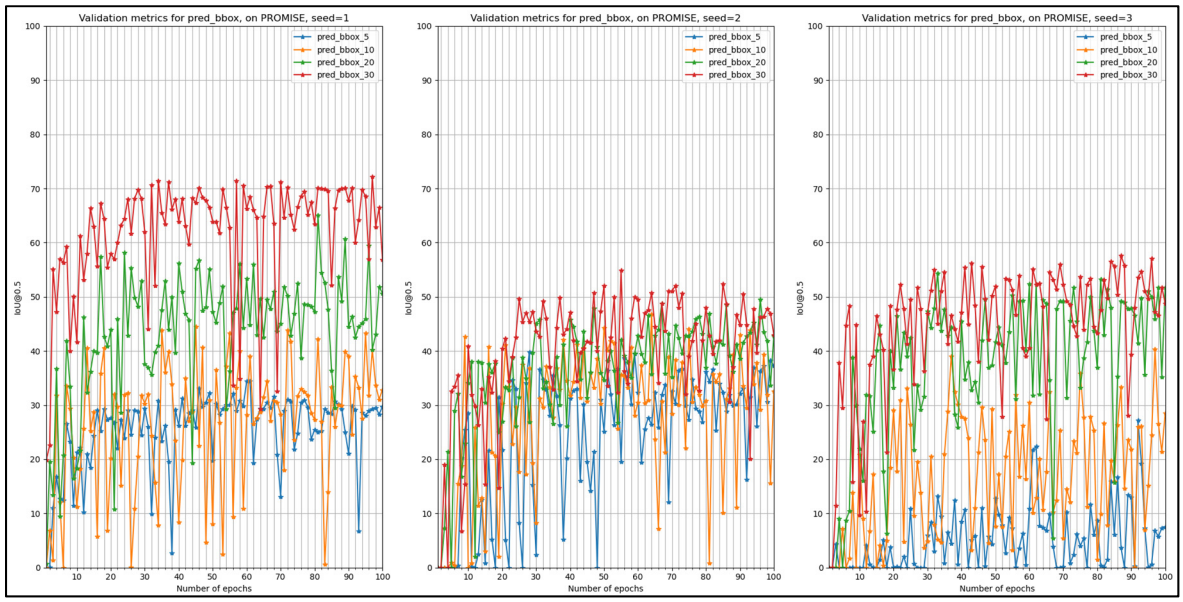

Figure-A III-4 Les valeurs moyennes de l'indice IoU entre les masques des boîtes englobantes vraies et ceux des pseudos boîtes englobantes, sur PROMISE 12 (5/35)

BIBLIOGRAPHIE

Allegretti, S., Bolelli, F., & Grana, C. (2020). Optimized Block-Based Algorithms to Label Connected Components on GPUs. *IEEE Transactions on Parallel and Distributed Systems*, *31*(2), 423-438. https://doi.org/10.1109/TPDS.2019.2934683

Bellver, M., Salvador, A., Torres, J., & Giro-i-Nieto, X. (2019). Budget-aware Semi-Supervised Semantic and Instance Segmentation. *arXiv:1905.05880 [cs]*. Repéré à http://arxiv.org/abs/1905.05880

Bengio, Y., Louradour, J., Collobert, R., & Weston, J. (2009). Curriculum learning. Dans *Proceedings of the 26th Annual International Conference on Machine Learning - ICML '09* (pp. 1-8). Montreal, Quebec, Canada : ACM Press. https://doi.org/10.1145/1553374.1553380

Bernard, O., Lalande, A., Zotti, C., Cervenansky, F., Yang, X., Heng, P.-A., … Jodoin, P.-M. (2018). Deep Learning Techniques for Automatic MRI Cardiac Multi-structures Segmentation and Diagnosis: Is the Problem Solved? *IEEE Transactions on Medical Imaging*, *37*(11), 2514-2525. https://doi.org/10.1109/TMI.2018.2837502

Bishop, C. M. (2006). *Pattern recognition and machine learning*. New York : Springer.

Chen, H., Dou, Q., Yu, L., Qin, J., & Heng, P.-A. (2018). VoxResNet: Deep voxelwise residual networks for brain segmentation from 3D MR images. *NeuroImage*, *170*, 446-455. https://doi.org/10.1016/j.neuroimage.2017.04.041

Chen, L.-C., Papandreou, G., Schroff, F., & Adam, H. (2017). Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv:1706.05587 [cs]*. Repéré à http://arxiv.org/abs/1706.05587

Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T., & Ronneberger, O. (2016). 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. Dans S. Ourselin, L. Joskowicz, M. R. Sabuncu, G. Unal, & W. Wells (Éds), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016* (pp. 424-432). Cham : Springer International Publishing. https://doi.org/10.1007/978-3-319-46723-8_49

Despotović, I., Goossens, B., & Philips, W. (2015). MRI Segmentation of the Human Brain: Challenges, Methods, and Applications. *Computational and Mathematical Methods in Medicine*, *2015*, e450341. https://doi.org/10.1155/2015/450341

Dolz, J., Desrosiers, C., & Ayed, I. B. (2020). Teach me to segment with mixed supervision: Confident students become masters. *Information Processing in Medical Imaging, 2021. arXiv:2012.08051 [cs]*. Repéré à http://arxiv.org/abs/2012.08051

Dolz, J., Desrosiers, C., & Ben Ayed, I. (2018). 3D fully convolutional networks for subcortical segmentation in MRI: A large-scale study. *NeuroImage*, *170*, 456-470. https://doi.org/10.1016/j.neuroimage.2017.04.039

Gonzalez, R. C., & Woods, R. E. (2018). *Digital image processing*. New York, NY : Pearson.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. Cambridge, Massachusetts : The MIT Press.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. Dans *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 770-778). https://doi.org/10.1109/CVPR.2016.90

Kervadec, H. (2021). *Constrained deep networks for medical image segmentation*.

Hsu, C.-C., Hsu, K.-J., Tsai, C.-C., Lin, Y.-Y., & Chuang, Y.-Y. (2019). Weakly Supervised Instance Segmentation using the Bounding Box Tightness Prior, 12.

Hung, W.-C., Tsai, Y.-H., Liou, Y.-T., Lin, Y.-Y., & Yang, M.-H. (2018). Adversarial Learning for Semi-Supervised Semantic Segmentation. *arXiv:1802.07934 [cs]*. Repéré à http://arxiv.org/abs/1802.07934

Jadon, S. (2020). A survey of loss functions for semantic segmentation. *2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, 1-7. https://doi.org/10.1109/CIBCB48159.2020.9277638

Jurdi, R. E., Petitjean, C., Honeine, P., Cheplygina, V., & Abdallah, F. (2020). High-level Prior-based Loss Functions for Medical Image Segmentation: A Survey. *Computer Vision and Image Understanding, 2021, 103248*. *arXiv:2011.08018 [cs]*. Repéré à http://arxiv.org/abs/2011.08018

Kervadec, H., Dolz, J., Granger, E., & Ayed, I. B. (2019a). Curriculum semi-supervised segmentation. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2019* (pp. 568-576). *arXiv:1904.05236 [cs]*. Repéré à http://arxiv.org/abs/1904.05236

Kervadec, H., Dolz, J., Wang, S., Granger, E., & Ayed, I. B. (2020b). Bounding boxes for weakly supervised segmentation: Global constraints get close to full supervision. *Medical Imaging with Deep Learning, 2020*. *arXiv:2004.06816 [cs]*. Repéré à http://arxiv.org/abs/2004.06816

Kervadec, H., Dolz, J., Yuan, J., Desrosiers, C., Granger, E., & Ayed, I. B. (2020a). Constrained Deep Networks: Lagrangian Optimization via Log-Barrier Extensions. *arXiv:1904.04205 [cs]*. Repéré à http://arxiv.org/abs/1904.04205

Kingma, D. P., & Ba, L. J. (2015). Adam: A Method for Stochastic Optimization. Repéré à https://dare.uva.nl/search?identifier=a20791d3-1aff-464a-8544-268383c33a75

Kodali, N., Abernethy, J., Hays, J., & Kira, Z. (2017). On Convergence and Stability of GANs. *Conference on Neural Information Processing Systems, 2017*. *arXiv:1705.07215 [cs]*. Repéré à http://arxiv.org/abs/1705.07215

Kostopoulos, G., Karlos, S., Kotsiantis, S., & Ragos, O. (2018). Semi-supervised regression: A recent review. *Journal of Intelligent & Fuzzy Systems*, *35*(2), 1483-1500. https://doi.org/10.3233/JIFS-169689

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84-90. https://doi.org/10.1145/3065386

Lempitsky, V., Kohli, P., Rother, C., & Sharp, T. (2009). Image segmentation with a bounding box prior. Dans *2009 IEEE 12th International Conference on Computer Vision* (pp. 277-284). Kyoto : IEEE. https://doi.org/10.1109/ICCV.2009.5459262

Litjens, G., Toth, R., van de Ven, W., Hoeks, C., Kerkstra, S., Ginneken, B., … Madabhushi, A. (2013). Evaluation of prostate segmentation algorithms for MRI: The PROMISE12 challenge. *Medical Image Analysis*, *18*, 359-373. https://doi.org/10.1016/j.media.2013.12.002

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully Convolutional Networks for Semantic Segmentation. *IEEE Conference on Computer Vision and Pattern Recognition, 2015*. *arXiv:1411.4038 [cs]*. Repéré à http://arxiv.org/abs/1411.4038

Márquez-Neila, P., Salzmann, M., & Fua, P. (2017). Imposing Hard Constraints on Deep Networks: Promises and Limitations. *arXiv:1706.02025 [cs]*. Repéré à http://arxiv.org/abs/1706.02025

Medical image computing and computer-assisted intervention -- MICCAI 2017. (2017). New York, NY : Springer Berlin Heidelberg.

Milletari, F., Navab, N., & Ahmadi, S.-A. (2016). V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. *arXiv:1606.04797 [cs]*. Repéré à http://arxiv.org/abs/1606.04797

Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., & Terzopoulos, D. (2020). Image Segmentation Using Deep Learning: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021*. *arXiv:2001.05566 [cs]*. Repéré à http://arxiv.org/abs/2001.05566

Papandreou, G., Chen, L.-C., Murphy, K. P., & Yuille, A. L. (2015). Weakly-and Semi-Supervised Learning of a Deep Convolutional Network for Semantic Image Segmentation. Dans *2015 IEEE International Conference on Computer Vision (ICCV)* (pp. 1742-1750). Santiago, Chile : IEEE. https://doi.org/10.1109/ICCV.2015.203

Paszke, A., Chaurasia, A., Kim, S., & Culurciello, E. (2016). ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation. *arXiv:1606.02147 [cs]*. Repéré à http://arxiv.org/abs/1606.02147

Pathak, D., Krähenbühl, P., & Darrell, T. (2015). Constrained Convolutional Neural Networks for Weakly Supervised Segmentation. Dans *2015 IEEE International Conference on Computer Vision (ICCV)* (pp. 1796-1804). https://doi.org/10.1109/ICCV.2015.209

Peng, J., Estrada, G., Pedersoli, M., & Desrosiers, C. (2020). Deep co-training for semi-supervised image segmentation. *Pattern Recognition*, *107*, 107269. https://doi.org/10.1016/j.patcog.2020.107269

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. Dans N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi (Éds), *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* (pp. 234-241). Cham : Springer International Publishing. https://doi.org/10.1007/978-3-319-24574-4_28

Kervadec, H., Dolz, J., Yuan, J., Desrosiers, C., Granger, E., & Ben Ayed, I. (2019). Log-barrier constrained CNNs.

Taha, A. A., & Hanbury, A. (2015). Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool. *BMC Medical Imaging*, *15*. https://doi.org/10.1186/s12880-015-0068-x

Tarvainen, A., & Valpola, H. (2018). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. *Conference on Neural Information Processing Systems, 2017*. *arXiv:1703.01780 [cs, stat]*. Repéré à http://arxiv.org/abs/1703.01780

Walter, É. (2014). *Numerical Methods and Optimization*. Cham : Springer International Publishing. https://doi.org/10.1007/978-3-319-07671-3

Wang, K., Zhan, B., Luo, Y., Zhou, J., Wu, X., & Wang, Y. (2021). Multi-Task Curriculum Learning For Semi-Supervised Medical Image Segmentation. Dans *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)* (pp. 925-928). https://doi.org/10.1109/ISBI48211.2021.9433851

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, *9*(4), 611-629. https://doi.org/10.1007/s13244-018-0639-9

Wang, Y., Zhang, Y., Tian, J., Zhong, C., Shi, Z., Zhang, Y., & He, Z. (2020). Double-Uncertainty Weighted Method for Semi-supervised Learning. Dans A. L. Martel, P. Abolmaesumi, D. Stoyanov, D. Mateus, M. A. Zuluaga, S. K. Zhou, … L. Joskowicz (Éds), *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020* (pp. 542-551). Cham : Springer International Publishing. https://doi.org/10.1007/978-3-030-59710-8_53

Wu, K., Otoo, E., & Shoshani, A. (2005). Optimizing connected component labeling algorithms. Dans J. M. Fitzpatrick & J. M. Reinhardt (Éds), (p. 1965). Communication présentée au Medical Imaging, San Diego, CA. https://doi.org/10.1117/12.596105

Yu, L., Wang, S., Li, X., Fu, C.-W., & Heng, P.-A. (2019). Uncertainty-aware Self-ensembling Model for Semi-supervised 3D Left Atrium Segmentation. *arXiv:1907.07034 [cs]*. Repéré à http://arxiv.org/abs/1907.07034

Zhang, Y., David, P., & Gong, B. (2017). Curriculum Domain Adaptation for Semantic Segmentation of Urban Scenes. Dans *2017 IEEE International Conference on Computer Vision (ICCV)*, 2039-2049. https://doi.org/10.1109/ICCV.2017.223

Zhengqin Li & Jiansheng Chen. (2015). Superpixel segmentation using Linear Spectral Clustering. Dans *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1356-1363). Boston, MA, USA : IEEE. https://doi.org/10.1109/CVPR.2015.7298741

Zhou, Y., Li, Z., Bai, S., Wang, C., Chen, X., Han, M., … Yuille, A. (2019a). Prior-aware Neural Network for Partially-Supervised Multi-Organ Segmentation. Dans *IEEE/CVF International Conference on Computer Vision (ICCV), 2019. arXiv:1904.06346 [cs]*. Repéré à http://arxiv.org/abs/1904.06346

Zhou, Y., Wang, Y., Tang, P., Bai, S., Shen, W., Fishman, E., & Yuille, A. (2019b). Semi-Supervised 3D Abdominal Multi-Organ Segmentation Via Deep Multi-Planar Co-Training. Dans *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)* (pp. 121-140). https://doi.org/10.1109/WACV.2019.00020

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324. https://doi.org/10.1109/5.726791