SCONet : Réseaux d'Occupation Convolutifs pour la Segmentation Multi-Organes

Maylis JOUVENCEL[1] Razmig KÉCHICHIAN[1] Julie DIGNE[2] Sébastien VALETTE[1]

[1]CREATIS, CNRS UMR 5220, INSERM U1206, Université Claude Bernard Lyon 1, INSA Lyon, 69100 Villeurbanne, France

[2]LIRIS, CNRS UMR 5205, Université Claude Bernard Lyon 1, 69100 Villeurbanne, France

Résumé – Les réseaux de neurones convolutifs sont la norme pour la segmentation multi-organes en 3D, mais présentent des limitations en termes de coût computationnel. Pour surmonter ces limitations, nous proposons de remplacer la grille de voxels par une représentation en nuage de points plus compacte. Nous proposons SCONet, un réseau léger basé sur ConvONet adapté à la segmentation multi-organes. SCONet prend en entrée un nuage de points extrait du volume d'origine et l'enrichit avec des caractéristiques géométriques et photométriques. La capacité de SCONet à interroger les probabilités d'occupation pour tout point dans l'espace lui permet de prédire une carte de segmentation multi-organes à une résolution arbitraire. Nous évaluons notre méthode sur un dataset d'images CT abdominales et comparons ses performances avec des méthodes performantes de l'état de l'art. Nous étudions aussi les bénéfices d'utiliser une fenêtre glissante pour l'apprentissage. Notre implémentation est disponible à l'adresse `https://github.com/maylis-j/SCONet`.

Abstract – Convolutional neural networks are the standard for 3D multi-organ segmentation, but have limitations in terms of computational cost. To overcome these limitations, we propose to replace the image voxel grid with a more compact point cloud representation. We propose SCONet, a lightweight ConvONet-based network adapted to the specific task of multi-organ segmentation. SCONet takes as input a point cloud extracted from the original volume and enriches it with geometric and photometric features. Thanks to its ability to query occupancy probabilities for any point in space, SCONet can be used to predict a multi-organ segmentation map at arbitrary resolution. We evaluate our network on a dataset of abdominal CT images and compare its performance with efficient state-of-the-art methods. We also study the impact of using a sliding window in the learning process. Our implementation is available at `https://github.com/maylis-j/SCONet`.

1 Introduction

Les méthodes récentes de segmentation automatique multi-organes ont obtenu des résultats impressionnants en utilisant des réseaux de neurones convolutifs (CNN), parmi lesquels UNet [12], qui est devenu la référence pour cette tâche. Cependant le coût en calcul de tels réseaux peut être excessif. Nous proposons de travailler sur un nuage de points extrait du volume d'origine pour réduire le coût computationnel. L'utilisation de nuages de points a récemment été proposé pour segmenter des images médicales [6], mais les solutions développées consistent souvent à simplement ajouter un module basé sur les nuages de points pour affiner une segmentation produite par un CNN, ce qui ne garantit pas toujours un coût de calcul inférieur. Les nuages de points sont également utilisés comme entrée pour les réseaux apprenant une représentation neuronale implicite (INR). Ces réseaux apprennent généralement une fonction de probabilité d'occupation comme ONet [9] et peuvent représenter des formes complexes à des résolutions arbitraires. De tels réseaux implicites ont déjà été utilisés pour la segmentation d'images médicales [8] et obtiennent des résultats prometteurs en utilisant ONet avec un encodeur basé sur un CNN. Dans ce travail, nous tirons parti d'une approche plus récente ConvONet [10] qui a une plus grande capacité de représentation locale grâce à une grille latente de caractéristiques de taille 32^3 qui permet d'apprendre des structures plus complexes que le vecteur latent de taille 512 de l'architecture ONet originale.

Nous proposons donc : (1) Une chaîne de traitement pour la segmentation multi-organes basée sur une représentation en nuage de points. (2) SCONet, un réseau basé sur ConvONet, conçu pour reconstruire plusieurs objets 3D à partir d'un nuage de points enrichi. Nous comparons notre méthode avec des réseaux discrets et implicites et montrons les avantages de SCONet en termes d'utilisation de la mémoire et de complexité computationnelle. Cette étude complète un travail précédent [7] en étudiant l'impact de l'utilisation d'une fenêtre glissante sur les résultats de SCONet.

2 Méthode proposée

2.1 Extraction du nuage de points

La première étape de notre chaîne de traitement, illustrée dans la Figure 1, consiste à extraire le nuage de N points de contours à partir de l'algorithme standard Canny [3]. Pour chaque point, en plus de ses coordonnées en x, y et z, nous calculons le descripteur SURF correspondant qui encode les informations locales dans un voisinage de 10^3 voxels autour du point. Ce descripteur est calculé avec l'implémentation 3D [1] de l'algorithme original [2] et produit un vecteur de 48 valeurs pour chaque point. Le nuage de points résultant est une matrice $(N \times 51)$ qui encode les caractéristiques géométriques et photométriques locales. Nous translatons et normalisons les coordonnées de manière isotrope pour les faire entrer dans le cube unité.

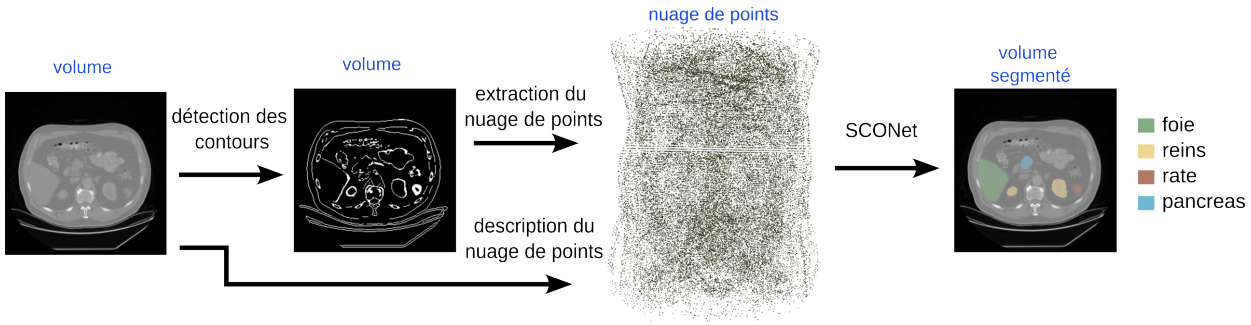

FIGURE 1 : Notre workflow complet de segmentation pour les images 3D.

2.2 SCONet

Pour traiter ce nuage de points, nous proposons SCONet, un réseau basé sur ConvONet adapté à la segmentation multi-organes. Le réseau proposé est illustré dans la Figure 2 et possède l'architecture encodeur-décodeur suivante :
Encodeur : L'encodeur calcule une grille latente de caractéristiques à partir du nuage de points d'entrée. Les coordonnées du nuage de points sont fournies à un encodeur PointNet [11] qui calcule des caractéristiques géométriques de dimension 32 pour chaque point. Ce vecteur de caractéristiques est ensuite concaténé au descripteur SURF de dimension 48. Les caractéristiques sont projetées sur une grille 32^3 par max pooling, résultant en une grille de caractéristiques de taille $(32^3 \times 80)$. Celle-ci est traitée par un réseau U-Net 3D à 3 niveaux qui prédit une grille latente de caractéristiques de dimension $32^3 \times 32$.
Décodeur : Le décodeur utilise la grille latente de caractéristiques apprise pour prédire les probabilités d'occupation d'un point de requête. Le vecteur de caractéristique en ce point est calculé par interpolation trilinéaire sur la grille de caractéristiques apprise par l'encodeur. Ce vecteur est fourni à un réseau d'occupation ONet basé sur ResNet. Une dernière couche linéaire suivie d'une couche Softmax permet de prédire par organe les probabilités d'occupation au point interrogé.

2.3 Entraînement et inférence

Entraînement : Lors de l'entraînement, nous échantillonnons des points de requête aléatoirement dans l'espace 3D et prédisons leurs valeurs d'occupation, optimisant une fonction de perte combinant Dice et entropie croisée.
Inférence : Lors de l'inférence, nous prédisons la carte de segmentation pour l'ensemble du volume en interrogeant les points correspondant aux coordonnées de tous les voxels du volume original.

3 Expériences

3.1 Dataset

Nous évaluons notre méthode sur le dataset AbdomenCT-1K, qui contient 1000 scans CT abdominaux. Les volumes ont une taille de $(512 \times 512 \times z)$. Nous ré-échantillonnons les images originales le long des axes x et y pour obtenir des volumes de taille $(256 \times 256 \times z)$. Les cartes de segmentation de référence sont fournies pour le foie, les reins, la rate et le pancréas. Pour l'entraînement nous séparons la base de données en des sous-ensemble train/val/test de 806/91/100 images.

3.2 Détails d'entraînement

Nous implémentons SCONet avec PyTorch et réalisons toutes les expériences sur un GPU NVIDIA Tesla V100-SXM2-32GB. Nous entraînons le réseau pendant 300 époques avec l'optimiseur AdamW et un taux d'apprentissage de 10^{-4}. La taille du mini-batch est fixée à un nuage de points d'entrée représentant un sujet. Lors de l'entraînement, le nombre de points de requête est fixé à $50k$. Nous fixons également la taille du nuage de points d'entrée à $N = 50k$ pour l'entraînement et l'inférence.

3.3 Comparaison avec d'autres méthodes

Pour évaluer les avantages de travailler avec une représentation en nuage de points plus légère et un réseau implicite, nous comparons nos performances avec celles de méthodes discrètes et implicites. Nous choisissons les U-Net 2D et 3D comme références CNN. Pour les deux réseaux, nous implémentons une architecture à 4 niveaux que nous entraînons avec une fonction de perte Dice pendant 30 époques. Les tailles de mini-batch utilisées sont 64 et 2 pour U-Net 2D et U-Net 3D, respectivement. De plus, nous entraînons le U-Net 3D sur des patchs d'images de taille $(256, 256, 30)$. Nos U-Net 2D et 3D sont basés sur l'implémentation PyTorch de [4] ([13]). SwinUNETR-V2 [5] sert de référence transformer, et est entraîné pendant 100 époques avec une fonction de perte combinée Dice et entropie croisée. Nous comparons également notre réseau avec ImPulSe, une référence INR composée d'un encodeur CNN et d'un décodeur ONet. Ce réseau est entraîné pendant 50 époques avec une fonction de perte combinant Dice et entropie croisée.

4 Résultats

Le premier critère pour évaluer notre méthode est la qualité des cartes de segmentation prédites. Les résultats qualitatifs présentés dans la Figure 3 confirment que SCONet peut produire des cartes de segmentation cohérentes : les quatre organes cibles sont globalement bien segmentés. On note aussi que les deux méthodes implicites que sont ImPulSe et SCONet permettent de générer des cartes de segmentations moins bruitées que les méthodes discrètes. Les segmentations prédites par SCONet manquent cependant de détails et ont des bordures parfois imprécises et des organes globalement très lisses, en particulier au niveau des reins.

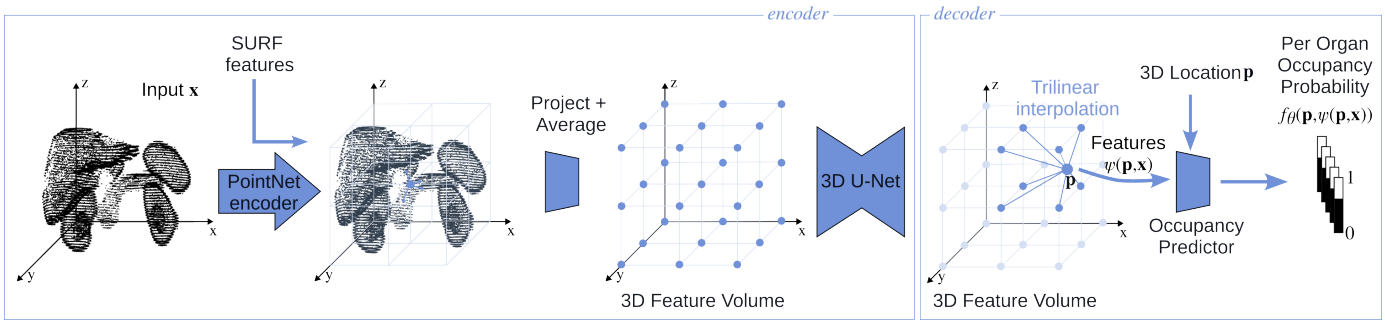

FIGURE 2 : Chaîne de traitement de SCONet.

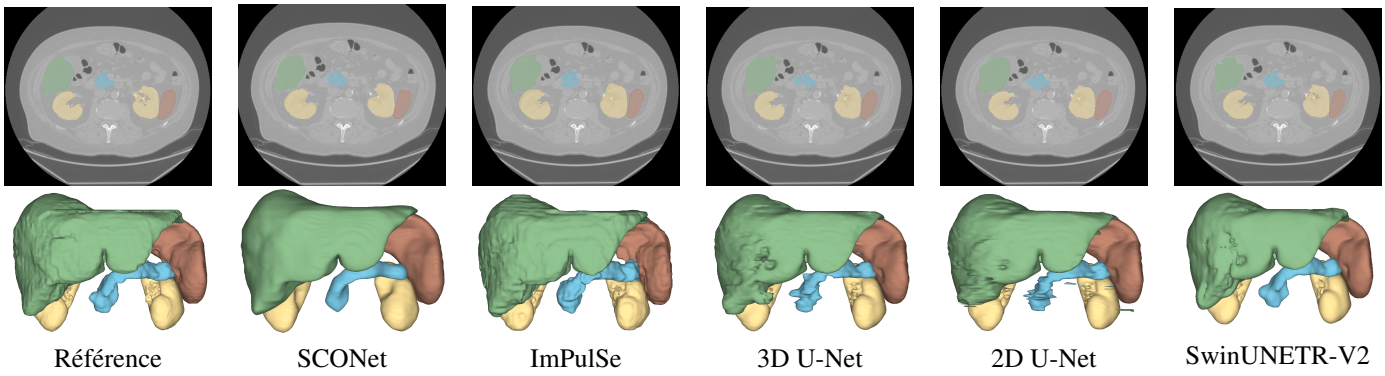

FIGURE 3 : Comparaison des résultats de segmentation obtenus par SCONet, ImPulSe, 3D et 2D U-Net et SwinUNETR-V2 sur une coupe axiale CT (en haut) et en 3D (en bas) pour le foie (vert), les reins (jaune), le pancréas (bleu) et la rate (rouge).

TABLE 1 : Comparaison des performances avec les références. La version (A) de SCONet apprend une grille de caractéristiques de taille 32^3, tandis que la taille de la grille pour la version (B) est 64^3.

Modèle	**Dice**					**HD**	**Inf. GPU**	**#Param**	**GFLOPs**
	Foie	*Reins*	*Rate*	*Panc.*	Moy.	Moy.	**(GB)**		
2D U-Net	0.961	0.943	0.950	0.794	0.912	91.60	**0.77**	1.36M	960
3D U-Net	0.956	0.938	0.948	0.808	0.913	46.29	4.48	4.08M	2,980
SwinUNETR-V2	0.950	0.936	0.954	0.830	**0.918**	139.16	7.24	18.35M	2,961
ImPulSe	0.957	0.929	0.930	0.763	0.895	20.32	16.25	33.28M	2,504
SCONet (A)	0.958	0.916	0.938	0.755	0.892	19.12	0.83	**1.26M**	**120**
SCONet (B)	0.965	0.929	0.949	0.801	0.911	**17.94**	1.27	**1.26M**	218

Nous rapportons aussi les résultats quantitatifs des différentes méthodes testées selon le coefficient Dice et la distance de Hausdorff (HD) dans le Tableau 1. Les caractères en gras indiquent des différences de performance significatives avec d'autres méthodes selon le test de Wilcoxon. Nous étudions également l'influence de la taille de la grille de caractéristiques pour SCONet : (A) correspond à une grille de 32^3 et (B) à 64^3. Les valeurs de Dice de SCONet sont similaires à celles de l'état de l'art, en particulier quand on augmente la résolution de la grille de caractéristiques. On note aussi que, comme c'est le cas pour les méthodes comparées, le pancréas est particulièrement difficile à segmenter par rapport aux autres organes. Pour SCONet, ce résultat peut s'expliquer par la grande diversité de formes de cet organe mais aussi par sa texture locale qui résulte en des contours mal définis dans les nuages de points extraits, rendant la reconstruction plus difficile par rapport aux trois autres organes. En ce qui concerne la précision de surface de segmentation, les méthodes implicites produisent une meilleure distance de Hausdorff, avec SCONet donnant les meilleurs résultats. Les valeurs de HD plus faibles des méthodes discrètes peut s'expliquer par leur restriction à travailler sur des coupes ou des patchs, tandis que SCONet et ImPulSe profitent d'une vision plus globale de la zone à segmenter.

Le deuxième critère d'étude est le coût de calcul de toutes les méthodes, que nous rapportons dans le Tableau 1. En termes d'utilisation de la mémoire GPU et du nombre de paramètres, notre réseau est comparable au U-Net 2D et quatre fois plus efficace que le U-Net 3D. Nous sommes également plus de 10 fois plus efficaces en termes de mémoire par rapport à ImPulSe et SwinUNETR-V2. Un autre avantage de SCONet est qu'il peut produire des cartes de segmentation avec la même empreinte mémoire à n'importe quelle résolution grâce à son décodeur basé sur des points de requête. Pour évaluer la complexité computationnelle, le nombre d'opérations, ou FLOPs, est rapporté pour un volume de taille fixe de $(256 \times 256 \times 100)$. Nous constatons que SCONet est le réseau le plus léger, avec quatre fois moins de FLOPs que U-Net 2D pour les deux configurations testées. Nous notons également que le nombre

d'opérations requises par SCONet dépend principalement du nombre de points de requête puisque l'encodeur a un nombre fixe d'opérations, qui est de 15 GFLOPs pour la configuration (A) et 113 GFLOPs pour (B).

4.1 Ajout d'une fenêtre glissante

Afin d'augmenter la capacité de représentation de la grille latente, qui peut être limitée par l'agrégation de plusieurs points dans une seule cellule, on propose d'étudier l'utilisation d'une fenêtre glissante. Cette fenêtre est de taille fixe pour l'entraînement et l'inférence de SCONet. À l'entraînement, la position de la fenêtre est choisie de manière aléatoire. Pour générer une carte de segmentation complète, plusieurs inférences sont effectuées afin que tout le volume soit traité. Au lieu d'échantillonner le nuage de points d'entrée dans son ensemble, on ne procède à l'échantillonnage que dans la zone inclue dans la fenêtre. Les coordonnées sont ensuite normalisées pour les faire correspondre au cube unité.

TABLE 2 : Résultats de segmentation de SCONet en utilisant une fenêtre glissante.

Configuration	**Dice** Moy	**HD** Moy	**GFLOPs**
SCONet (A)	0.900	18.98	409
SCONet (B)	0.913	21.43	800

On rapporte dans le Tableau 2 les résultats de l'utilisation d'une fenêtre glissante de taille $0.8 \times 0.8 \times 0.8$ dans le cube unité. Les scores de Dice et HD sont améliorés par rapport à la version standard mais au prix d'un coût computationnel plus élevé. L'utilisation de la mémoire reste constante.

5 Conclusions et travaux futurs

Nous avons présenté SCONet, un réseau léger basé sur ConvONet qui produit une carte de segmentation à une résolution arbitraire à partir d'un nuage de points enrichi avec des caractéristiques photométriques. Comparé aux références discrètes ainsi qu'à un réseau implicite similaire, SCONet montre un bon compromis entre des cartes de segmentations de qualité et un faible coût de calcul. Utiliser une fenêtre glissante ou augmenter la dimension de la grille latente permet aussi d'améliorer les carte de segmentation prédites, au prix d'une légère augmentation du coût de calcul.

Nos expériences montrent que les résultats de segmentation de SCONet dépendent de la qualité du nuage de points d'entrée. Les points manquants sur les bordures des organes peuvent ainsi diminuer les performances de segmentation. Cela pourrait être adressé en utilisant une extraction apprises des points d'entrée. Dans le futur, nous aimerions également explorer l'utilité de cette méthode pour la multi-modalité.

6 Remerciements

Ce travail a été financé par le projet TOPACS ANR-19-CE45-0015 de l'Agence Nationale de la Recherche (ANR). Il utilise des ressources HPC de GENCI-IDRIS (Grant 2024-AD011013983R1).

Références

[1] Rémi AGIER, Sébastien VALETTE, Razmig KÉCHICHIAN *et al.* : Hubless keypoint-based 3d deformable groupwise registration. *Medical image analysis*, 59: 101564, 2020.

[2] Herbert BAY, Tinne TUYTELAARS et Luc VAN GOOL : Surf : Speeded up robust features. *In ECCV 2006*, pages 404–417. Springer, 2006.

[3] John CANNY : A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986.

[4] Özgün ÇIÇEK, Ahmed ABDULKADIR, Soeren S LIENKAMP, Thomas BROX *et al.* : 3d u-net : learning dense volumetric segmentation from sparse annotation. *In MICCAI 2016*, pages 424–432. Springer, 2016.

[5] Yufan HE, Vishwesh NATH, Dong YANG *et al.* : Swinunetr-v2 : Stronger swin transformers with stagewise convolutions for 3d medical image segmentation. *In MICCAI 2023*, pages 416–426, 2023.

[6] Ngoc-Vuong HO, Tan NGUYEN, Gia-Han DIEP, Ngan LE *et al.* : Point-unet : A context-aware point-based neural network for volumetric segmentation. *In MICCAI 2021*, pages 644–655. Springer, 2021.

[7] Maylis JOUVENCEL, Razmig KÉCHICHIAN, Julie DIGNE et Sébastien VALETTE : Sconet : Convolutional occupancy networks for multi-organ segmentation. *In IEEE ISBI*, 2025.

[8] Kaiming KUANG, Li ZHANG, Jingyu LI, Hongwei LI *et al.* : What makes for automatic reconstruction of pulmonary segments. *In MICCAI 2022*, pages 495–505. Springer, 2022.

[9] Lars MESCHEDER, Michael OECHSLE, Michael NIEMEYER, Sebastian NOWOZIN *et al.* : Occupancy networks : Learning 3d reconstruction in function space. *In 2019 IEEE CVPR*, pages 4460–4470, 2019.

[10] Songyou PENG, Michael NIEMEYER, Lars MESCHEDER, Marc POLLEFEYS *et al.* : Convolutional occupancy networks. *In ECCV 2020*, pages 523–540. Springer, 2020.

[11] Charles R QI, Hao SU, Kaichun MO et Leonidas J GUIBAS : Pointnet : Deep learning on point sets for 3d classification and segmentation. *In 2017 IEEE CVPR*, pages 652–660, 2017.

[12] Olaf RONNEBERGER, Philipp FISCHER et Thomas BROX : U-net : Convolutional networks for biomedical image segmentation. *In MICCAI 2015*, pages 234–241. Springer, 2015.

[13] Adrian WOLNY, Lorenzo CERRONE, Athul VIJAYAN, Rachele TOFANELLI *et al.* : Accurate and versatile 3d segmentation of plant tissues at cellular resolution. *eLife*, 9:e57613, jul 2020.