

Détection des axones et des dendrites

Gabriel Leclerc, Antoine Dufour

Université Laval - GLO-7030

Travail remis à Philippe Giguère

Québec, Québec, Canada

1. Introduction

La compréhension du cerveau humain fascine l'Homme depuis déjà plusieurs siècles. Afin de mieux comprendre sa structure interne, plusieurs technologies ont été développées au cours du temps dans le but d'en étudier le fonctionnement. Plus récemment, la microscopie de fluorescence a permis de réaliser l'étude de diverses protéines à l'intérieur des neurones. Cependant, cette méthode est limitée par le nombre de protéines qui peuvent être observées simultanément. Afin de palier à cette limitation, il serait essentiel de développer une méthode alternative permettant de réaliser l'identification des protéines contenues dans les neurones. Ainsi, avec l'émergence des réseaux de neurones profonds et l'utilisation de ceux-ci dans la vision par ordinateur cette technologie semble la plus prometteuse afin d'automatiser l'identification et la localisation des composantes de la neurones telles que les axones et les dendrites. Le document suivant présente la réalisation d'un réseau de neurones profonds basé sur l'architecture U-net permettant de réaliser la prédiction du positionnement des neurones dans des images prises à l'aide d'un microscopes.

2. Revue de littérature

L'idée principale de la technique pour réaliser la localisation spatiale de diverses protéines dans les neurones humaines est d'utiliser un réseau de neurones à convolution basé sur l'architecture U-Net (Ronneberger et al., 2015). En effet, celui-ci ayant déjà fait ses preuves quand à ses performances sur plusieurs problèmes de segmentations d'images biomédicales, il semble pertinent de s'en inspirer dans le contexte de notre problèmes. Un des avantages principal de cette architecture réside dans sa capacité à faire passer l'information de bas niveau à travers les couches du réseau. Les modules des encodeurs et des décodeurs du réseau profond sont quand à eux inspirés de l'architecture pix-2-pix (Isola et al., 2016).

3. Jeu de données

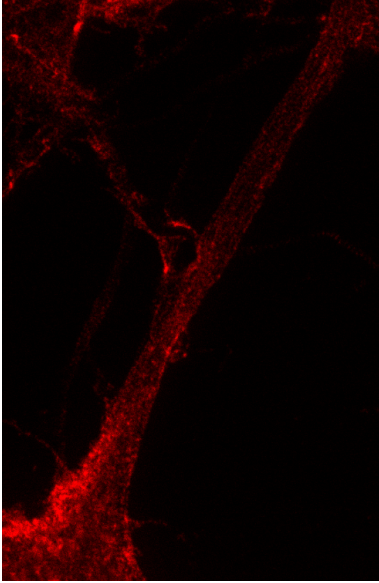
Le jeu de données utilisé est fourni par le CERVO, un centre de recherche en neuroscience et en santé mentale de Québec. Le jeu de données comporte 1041 images, de tailles variables, enregistré sous format tif sur 16-bit. Le jeu de données en entraînement comporte 832 images et celui en test comporte 209 images. 10% du jeu de données en entraînement est utilisé pour la validation. Le premier canal (figure 1a) contient l'image de la protéine d'intérêt, l'actine. Les deux autres canaux contiennent les prolongements neuronaux à détecter à partir de la protéine d'intérêt. Plus précisément, le deuxième et le troisième canal et contient l'image contenant les marquages pour les prolongements neuronaux de type axone (figure 1b) et dendrite (figure 1c) respectivement.

Les images originales sont enregistrées sur 16-bits (65 536 valeurs d'intensités). Par contre, la plage dynamique réel occupée par les images va d'environ 32 000 à 33 000. C'est pourquoi il est très important de normaliser les images pour entraîner le réseau. La moyenne et l'écart type calculés à partir du jeu de données en entraînement sont de $\mu = 32772.978$ et $\sigma = 8.194$.

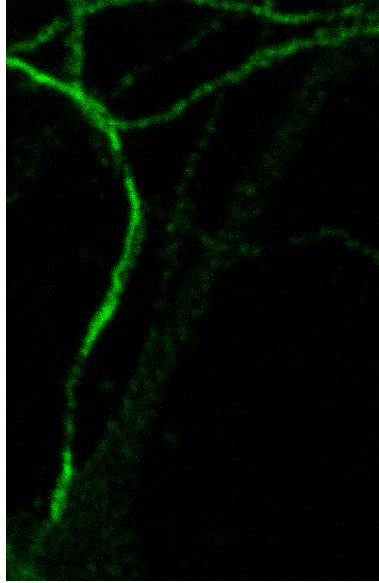
Les deux derniers canaux sont utilisés pour créer les masques pour entraîner le réseau. Afin de créer de bons masques, la valeur minimale de chaque canal y est soustraite. Ensuite, un filtre Gaussien de valeur $\sigma = 8$ et $\sigma = 10$ est appliqué sur les canaux axone et dendrites respectif. Le masque binaire est créé selon un seuillage de $I \geq 4 \times 10^{-5}$ et $I \geq 5 \times 10^{-5}$ pour chaque canal respectif.

4. Augmentation des données

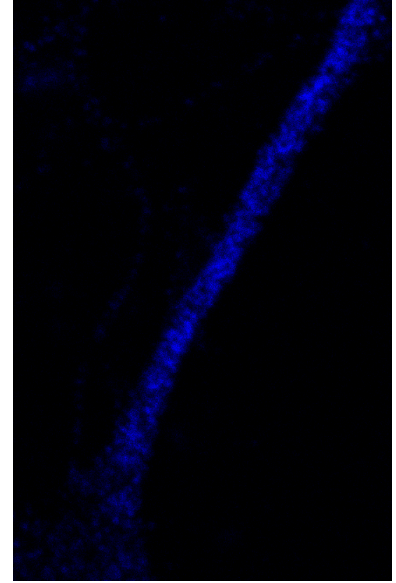
Il y a quatre contraintes majeurs avec le jeu de données : 1) le jeu de données comporte peu d'exemples, 2) les images sont de tailles différentes, 3) le redimensionnement n'est pas possible puisqu'il modifie la structure intrinsèque des images, pouvant ainsi altérer la prédiction selon le redimensionnement appliqué, 4) les classes à prédire (axone ou dendrite) peuvent être présente sur un même pixel.



(a) Canal actine



(b) Canal axone



(c) Canal dendrite

FIGURE 1: Exemple d'image du jeu de données

Afin de contourner les trois premières contraintes, l'utilisation du découpage aléatoire est utilisée. Celui-ci permet de générer des sous-images de 256x256 pixels. Le renversement aléatoire horizontal et vertical des images sont aussi utilisés pour générer des images additionnelles. Il est important de mentionner que les techniques d'augmentation des données doivent être simultanément appliquées sur l'image et sur le masque correspondant. En effet, le positionnement spatial des neurones dans chaque image et son masque se doit d'être identique afin de réaliser un bon entraînement du réseau sur les données. Puisque les images contiennent beaucoup d'espaces sans région d'intérêt, un tri automatique est réalisé afin de rejeter les images complètement noires découlant de l'application du découpage aléatoire. Un nouveau découpage est appliqué jusqu'à ce que le masque de l'image généré ne soit pas complètement noir.

Pour résoudre la dernière contrainte, deux réseaux sont entraînés pour prédire les axones et les dendrites respectivement. Les réseaux utilisent exactement la même architecture.

5. Architecture du réseau

L'ensemble du code de l'architecture est disponible sur le lien suivant : <https://github.com/PandaGab/detection-axones-dendrites>.

Le réseau reçoit en entrée une image 256x256 avec un canal uniquement. La section d'encodage du réseau réalise huit niveaux de convolutions ayant des fenêtres 4x4, un pas de 2. À chaque niveau la taille de l'image diminue d'un facteur 2 et le nombre de filtres quand à lui est

doublé jusqu'à concurrence de 512 filtres (1, 64, 128, 256, 512). Dans les quatre premiers niveaux, l'information spatiale est transférée vers un espace de caractéristique. Dans les quatre derniers niveaux, il y a une diminution de l'information afin d'aller chercher une certaine représentation haut niveau utile pour la segmentation. Après chaque convolution une "batchnorm" est appliquée ainsi qu'une "LeakyReLU". La section de décodage du réseau applique 7 déconvolutions. Chaque déconvolution est suivie d'une batchnorm puis d'une ReLU standard. Il est important de mentionner que du "dropout" avec probabilité de 50% est appliqué sur les trois premières couches de la section de décodage. Les "features maps" correspondant de même dimension (du côté encodeur et décodeur) sont concaténés ensemble avant d'être déconvolués. L'image en sortie du réseau comporte les mêmes dimensions que l'image en entrée.

6. Entraînement

L'entraînement est réalisé sur les images d'actine. Les poids du réseau sont initialisés de manière aléatoire selon une distribution Gaussienne de moyenne 0 avec une déviation standard de 0.02. Le réseau est entraîné pour un nombre de 300 epochs avant des "mini-batches" de 32 images. L'algorithme de descente du gradient stochastique (SGD) est utilisé. Un taux d'apprentissage de 0.1 et diminuant d'un facteur 10 à tous les 50 epochs est utilisé pour l'entraînement. Le "momentum" appliqué est quand à lui de 0.9. La fonction de perte utilisée est la "binary cross entropy" entre l'image prédite et le masque binaire.

Afin d'accélérer l'entraînement, le deuxième réseau à entraîner utilise les poids du premier réseau entraîné pour

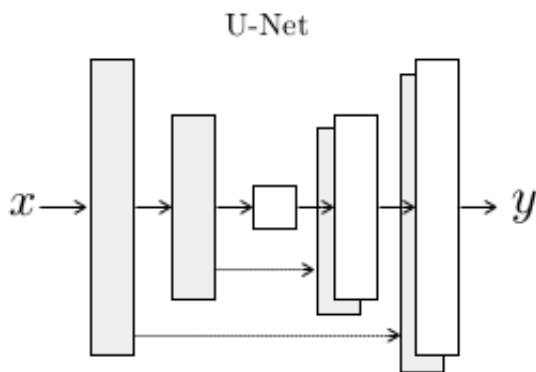


FIGURE 2: Exemple d'encodeur-décodeur U-Net tiré de (Isola et al., 2016)

détecter les axones. Aucune couche n'est gelée.

6.1. Inférence

Comme le réseau ne prend que des images 256x256 en entrée, l'image d'entrée doit être découpée en fenêtre de cette taille. Pour améliorer la détection, nous présentons au réseau des fenêtres qui se chevauchent de moitié (donc un 'stride' de la moitié de la largeur de la fenêtre). Cela permet d'avoir une meilleure estimation pour les éléments présents dans la bordure de l'image, puisque ces éléments se retrouvent environ au centre d'une autre image présentée au réseau. La moyenne est effectuée sur toutes les prédictions d'un pixel. Finalement, la décision si un pixel appartient ou non à la classe, est basée sur un seuillage de 0.5 sur la sortie du réseau.

7. Résultats

Les résultats sont présentés avec une image RGB. La prédiction pour les axones est dans le canal rouge et la prédiction pour les dendrites dans le canal bleu. Le canal vert est à zéro. Les régions jaunes sont donc là où il y a une prédiction des deux classes.

Sur la figure 3 et 4, l'image d'entrée est la protéine d'intérêt à partir de laquelle les axones et les dendrites doivent être identifiés. Leurs masques binaire respectif sont aussi présentés.

8. Analyse et discussions

8.1. Prédiction des axones

L'inférence sur les axones donne relativement de bons résultats selon la matrice de confusion de la figure 5. Par contre, en analysant les résultats sur le jeu de données en test, il semble que le réseau ne fait qu'effectuer un seuillage sur l'image d'entrée. En effet, comme on peut le voir dans les résultats présentés dans les figures 3 et 4, il y a du rouge

(prédiction pour les axones) sur tous les pixels non-nuls dans l'image d'entrée (la jaune signifie qu'il y a présence d'axone et de dendrite). Le réseau n'a pas appris à détecter les axones selon la structure de ceux-ci, mais seulement en effectuant un seuillage.

Ce comportement provient probablement de la méthode pour créer les masques d'entraînement pour les axones. Le fait de seuiller avec un seul seuil pour toutes les images du jeu de données en entraînement donnait des masques très variables. Par exemple, sur le masque 1b, on peut voir dans le canal axone la présence des dendrites qui peuvent être incluses dans le masque des axones si le seuillage n'est pas choisi adéquatement.

Comme, dans notre cas, la vérité terrain pour l'entraînement de la détection des axones n'était pas toujours cohérente, le réseau a déterminé que le seuillage était la meilleure méthode pour obtenir la plus petite erreur en moyenne. Il n'a pas appris à discerner la structure intrinsèque définissant un axone.

8.2. Prédiction des dendrites

Les résultats sur l'inférence des dendrites donnent des résultats nettement meilleurs. Sur les images 3d et 4d, la région en jaune (concrètement, la prédiction des dendrites est dans le canal vert et la prédiction des axones dans le canal rouge) est très semblable au masque de vérité terrain. Le réseau se fie sur la structure et non seulement sur l'intensité des pixels de l'image en entrée. En effet, il ignore les axones présents dans l'image d'entrée et prédit adéquatement les dendrites.

Les résultats pour ce type de prolongement de neurones sont dus à la qualité et la cohérence des masques qui ont été créés. En effet, la canal contenant l'image des dendrites (figure 1c) comporte une différence marquée d'intensité de pixels entre les dendrites et les autres structures.

9. Conclusion

L'architecture U-net est très puissante pour la segmentation d'image biologique. Avec des images d'une protéine d'intérêt, l'actine, nous avons été en mesure d'identifier les prolongements neuronaux de type dendrite. Malheureusement, la détection des prolongements de type axone n'a pas été apprise correctement par notre réseau en raison des masques de vérité terrain trop variables.

Afin d'améliorer les performances du réseau de détection des axones, la création des masques manuellement (choix du seuillage) améliorera les résultats.

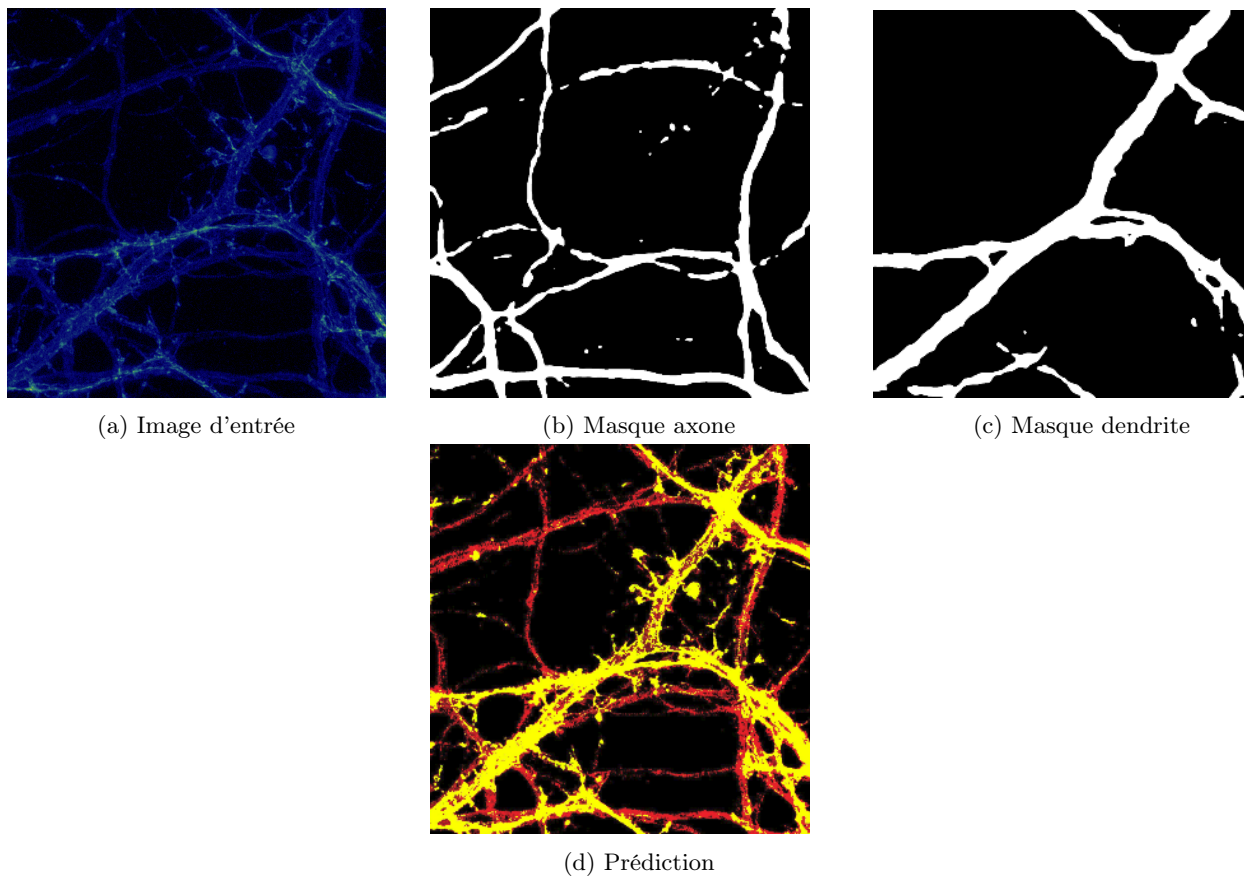


FIGURE 3: Exemple de résultats sur le jeu de données de test

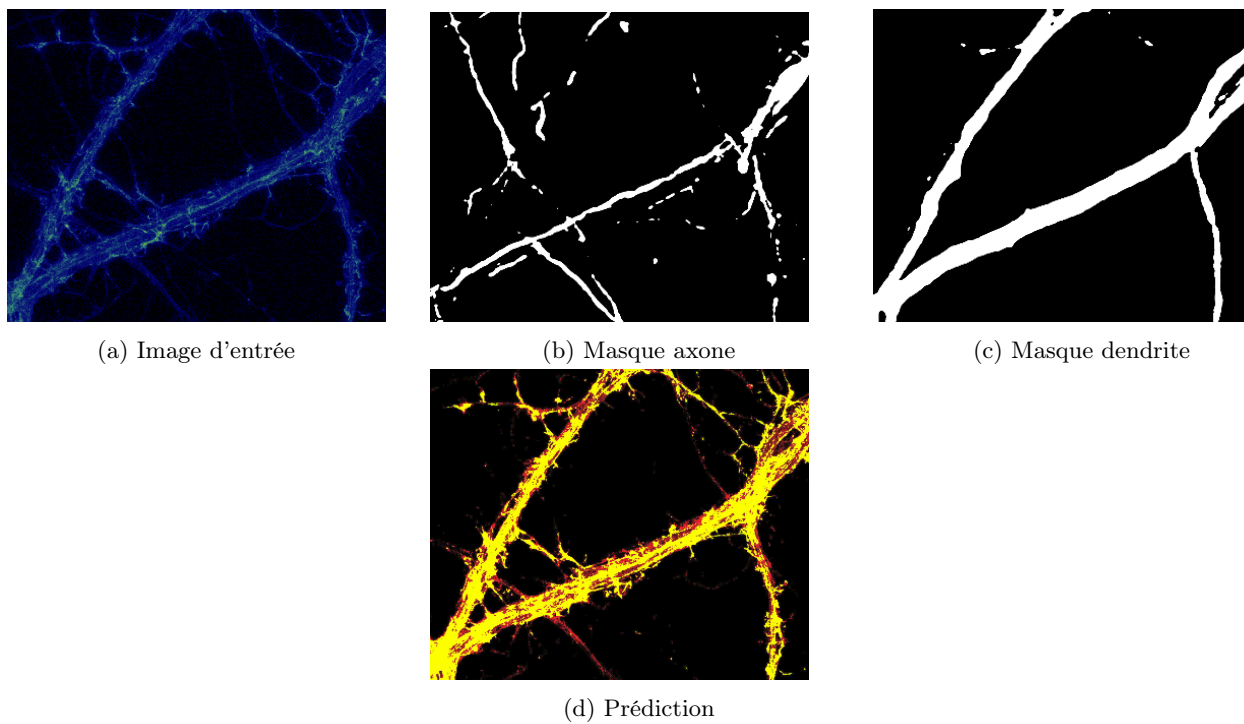


FIGURE 4: Exemple de résultats sur le jeu de données de test

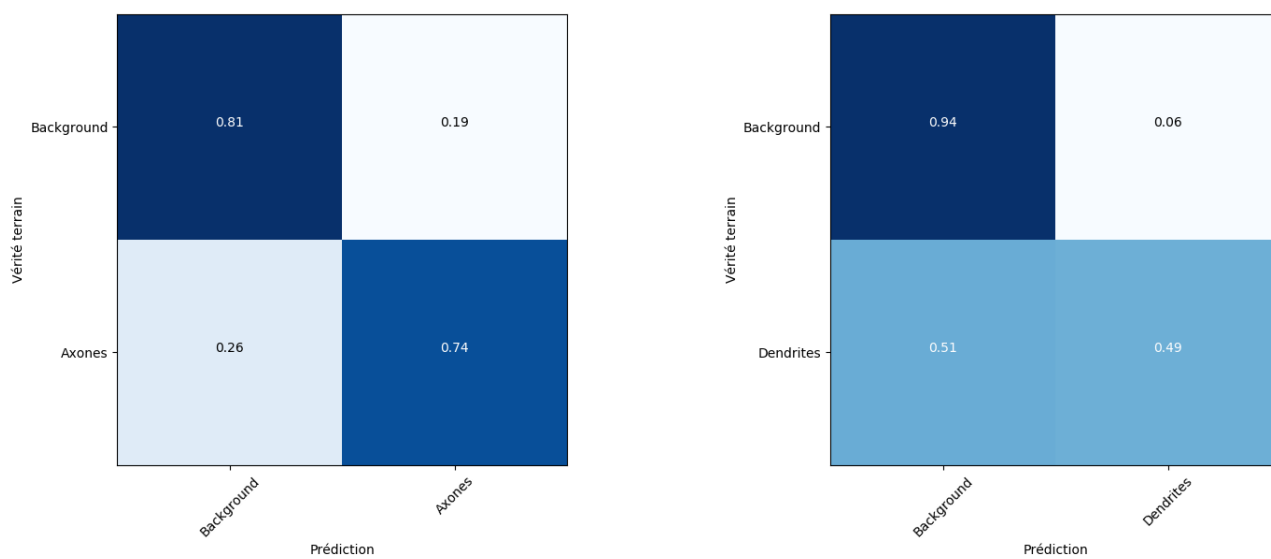


FIGURE 5: Matrices de confusion

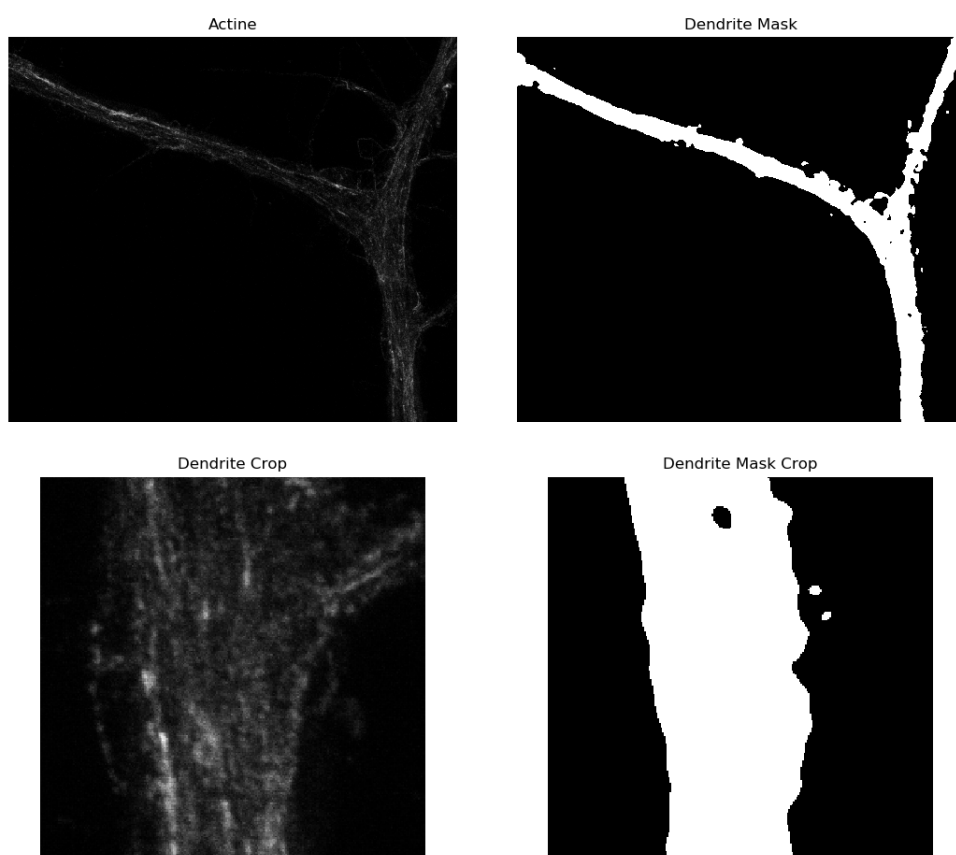


FIGURE 6: Exemple de découpage aléatoire réaliser sur une image de neurone et sur son masque

Références

Isola, P., Zhu, J., Zhou, T., Efros, A. A., 2016. Image-to-image translation with conditional adversarial networks. CoRR abs/1611.07004.

URL <http://arxiv.org/abs/1611.07004>

Ronneberger, O., Fischer, P., Brox, T., May 2015. U-Net : Convolutional Networks for Biomedical Image Segmentation. ArXiv e-prints.