

*Cette ressource est issue d'une publication du numéro 108 de La Revue 3EI d'avril 2022 et fait partie du « Dossier Intelligence Artificielle » [18] sur Culture Sciences de l'Ingénieur. Mélanie Pietri est en troisième année de l'ENS Paris-Saclay, en M2 FESUP PSEE (préparation à l'agrégation de science industrielle de l'ingénieur option ingénierie électrique).*

L'étude de la migration de cellules cancéreuses s'est développée ces dernières années avec les progrès en intelligence artificielle pour obtenir plus d'informations sur la formation de métastases (tumeurs cancéreuses). Les expériences mises en œuvre pour effectuer des observations nécessitent de recourir à des algorithmes de traitement d'images pour en tirer des conclusions. La variabilité de la géométrie des cellules étudiées est une difficulté pour le suivi cellulaire qui peut être abordée à l'aide d'algorithmes de traitement d'images s'appuyant sur des réseaux de neurones convolutifs à segmentation d'instance. La structure complexe du réseau de neurones Mask R-CNN permet d'obtenir des résultats avec une meilleure exactitude que pour d'autres structures et un grand gain de temps sur le traitement des données.

Cette ressource présente, dans le contexte de l'étude de cellules dans des dispositifs microfluidiques, une structure utilisée pour traiter les images et mesurer les grandeurs d'intérêt dans une approche utilisant des réseaux de neurones convolutifs avec apprentissage supervisé et modèles pré-entraînés. Cette ressource fait partie du « Dossier Intelligence Artificielle » [18].

## 1 – Contexte d'études

### 1.1 - Étude de la migration de cellules cancéreuses

Le phénomène d'électrotaxie a été découvert il y a une centaine d'années : lorsque des cellules biologiques vivantes sont placées dans un champ électrostatique, celles-ci ont tendance à migrer en s'alignant avec la direction du champ. Depuis ces dix dernières années, ce phénomène a été une source d'intérêt pour la communauté scientifique et en particulier dans le domaine biomédical en cancérologie. En effet, la migration cellulaire n'est pas la même pour des cellules saines et des cellules cancéreuses qui vont chercher à se regrouper et créer des métastases (amas de cellules ou tumeurs).

Les données collectées sur ce type de comportements révèlent qu'il existe un lien entre le taux d'invasivité des cellules cancéreuses (leur capacité à se regrouper pour former des tumeurs) et leur capacité électrotactique (leur faculté à se déplacer en présence d'un champ électrostatique extérieur). C'est ce lien qui est étudié pour tenter de contrôler le mouvement des cellules cancéreuses sous champs électrique et réaliser par exemple de la chimiothérapie localisée.

Afin d'étudier ce phénomène, des microsystèmes s'appuyant sur des dispositifs microfluidiques sont développés [1] pour observer les cellules dans des canaux d'épaisseur d'une centaine de micromètres (voir Figure 1). Les microscopes utilisés pour visualiser les cellules sont équipés de caméras afin d'enregistrer des vidéos des cellules se déplaçant dans le microcanal. On utilise un

type de microscopie particulière : la microscopie à contraste de phase, pour avoir un meilleur contraste entre les cellules et le substrat en arrière-plan.

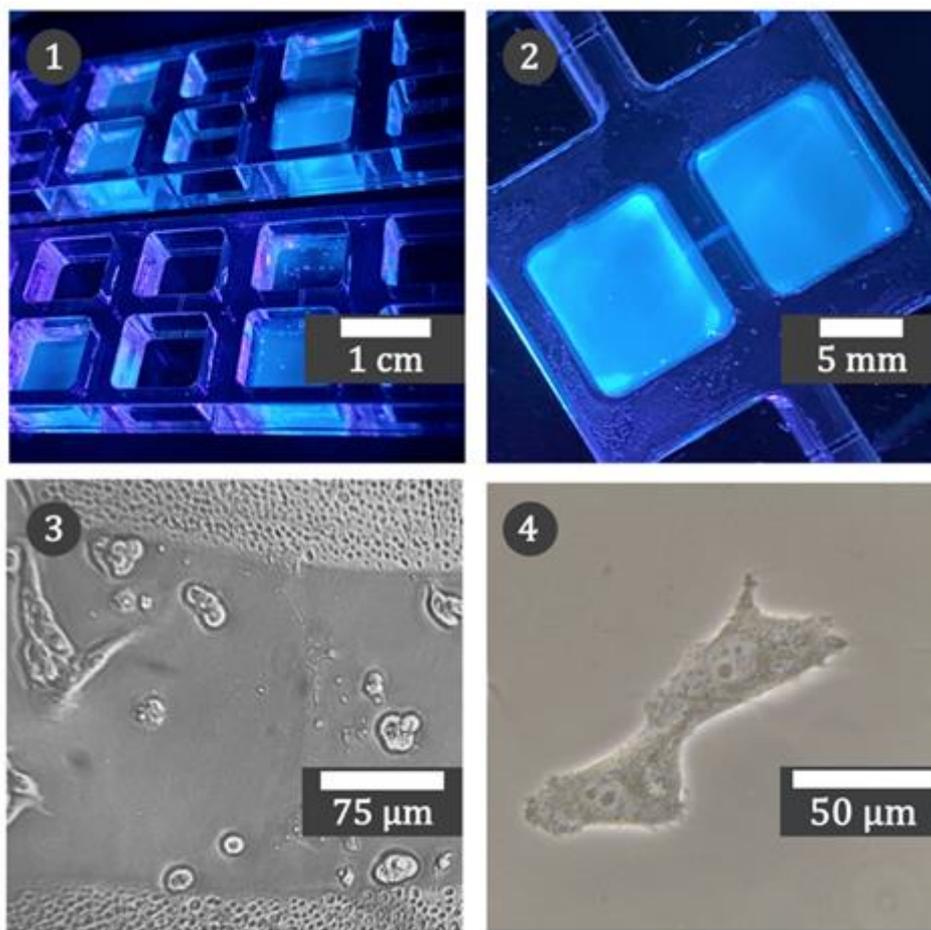


Figure 1 : 1) Dispositif microfluidique, les réservoirs cubiques sont reliés par un microcanal. 2) Zoom sur un microcanal reliant deux réservoirs. 3) Zoom sur le microcanal contenant les cellules. 4) Image de deux cellules cancéreuses.

Le traitement de données nécessite alors d'avoir recours à des algorithmes de traitement d'images. Les grandeurs d'intérêt pour étudier la migration cellulaire sont la vitesse moyenne de chaque cellule et la projection de la position de chaque cellule par rapport à l'horizontal. Ces paramètres sont décrits sur la Figure 2. Ces grandeurs sont mesurées sur toutes les cellules de l'image pour chaque image d'une vidéo. On étudie ensuite la relation entre la norme du champ électrique et ces deux paramètres.

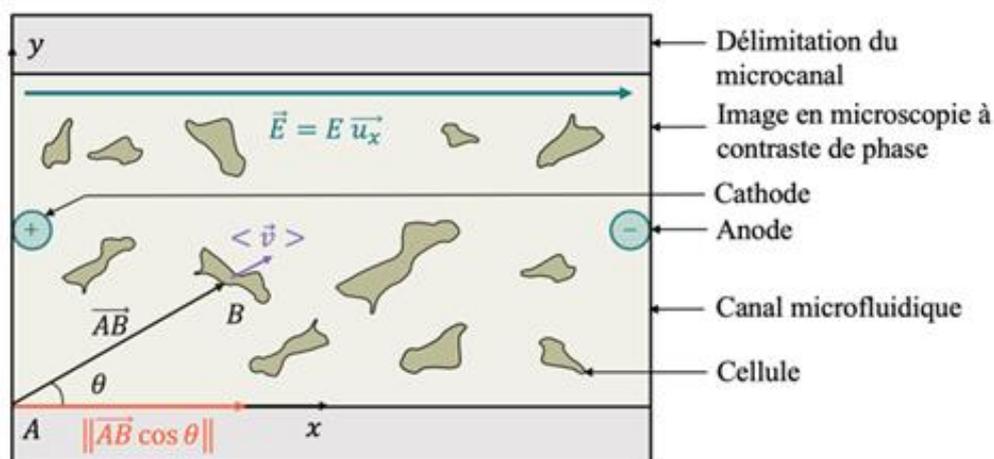


Figure 2 : Schéma des grandeurs d'intérêts pour l'étude de l'électrotaxie

## 1.2 - Traitement des données : intérêt du deep learning

Comme on peut le voir sur la Figure 1, les cellules cancéreuses qui adhèrent au microcanal ne sont pas de formes circulaires. Il est donc assez difficile de suivre ces cellules avec des algorithmes du type transformée de Hough (détection de cercles) [2]. D'autres algorithmes comme celui de Crocker-Grier [3] s'appuyant sur la physique du mouvement brownien permettent d'effectuer du suivi cellulaire plus précis mais les erreurs demeurent encore trop grandes pour des cellules avec ce type de géométrie. Ceux-ci sont mis en œuvre sur la Figure 3 pour des cellules circulaires.

Une autre difficulté s'ajoute à la complexité de la forme géométrique des cellules : celles-ci changent de forme d'une image à une autre en se déplaçant sur le substrat du canal.

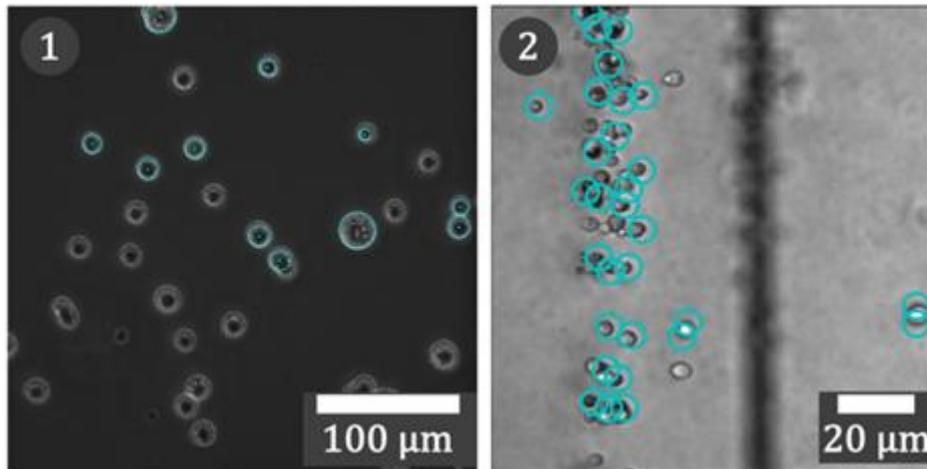


Figure 3 : 1) Détection de cellules en suspension par transformée de Hough.  
2) Détection de cellules en suspension par l'algorithme de Crocker-Grier

De plus il faut un certain nombre d'images pour pouvoir obtenir des données exploitables, ce qui est un point à prendre en compte concernant la complexité en temps et en mémoire des algorithmes utilisés.

Face à ces difficultés, des approches utilisant des réseaux de neurones convolutifs avec apprentissage supervisé et des modèles pré-entraînés ont été développés. Nous allons par la suite présenter une structure utilisée pour traiter les images et mesurer les grandeurs d'intérêt.

## 2 – Réseau de neurones convolutif et segmentation d'instance

### 2.1 - Présentation générale

L'objectif derrière l'utilisation de la structure de réseau de neurones est d'effectuer de la segmentation d'instance c'est-à-dire classer et distinguer des objets de la même instance « cellule » afin de suivre leur mouvement au cours du temps sur les différentes images d'une vidéo.

L'algorithme Usiigaci utilisé [4] fait appel à trois modules distincts : un réseau de neurones Mask R-CNN (Mask Regional Convolutional Neural Network) pour la détection des cellules et la segmentation, le module Python Trackpy permettant de déterminer les trajectoires des cellules et un module de traitement de données pour automatiser le traitement et visualiser les résultats. L'ensemble est codé en langage Python et accessible via Github [5].

Le réseau de neurones Mask R-CNN s'articule autour de deux structures combinées : un réseau à pyramide de caractéristiques (FPN : Features Pyramid Network) [6] et un réseau résiduel de 101

couches (Resnet-101) [7]. Nous allons ici montrer l'intérêt de cette structure assez complexe dans le suivi cellulaire.

## 2.2 - Données pour le pré-entraînement

Dans un premier temps il faut pré-entraîner le réseau de neurones pour la segmentation. Ici le réseau est pré-entraîné sur un problème de classification général pour ensuite en faire l'application sur un problème plus précis (*transfer learning*). Mask R-CNN a été pré-entraîné préalablement avec la base de données Microsoft COCO contenant de nombreuses images d'objets à détecter [8]. Le réseau pré-entraîné provenant de Matterport, Inc est documenté et disponible sur Github [9]. On ajoute ensuite de l'information par rapport au problème de suivi cellulaire avec notre pré-entraînement sur des images des cellules étudiées : un entraînement de 100 epochs pour le réseau de neurones à pyramide de caractéristiques et 500 epochs sur tout le réseau ResNet-101. Le principe de génération des données d'entraînement est décrit sur la Figure 4.

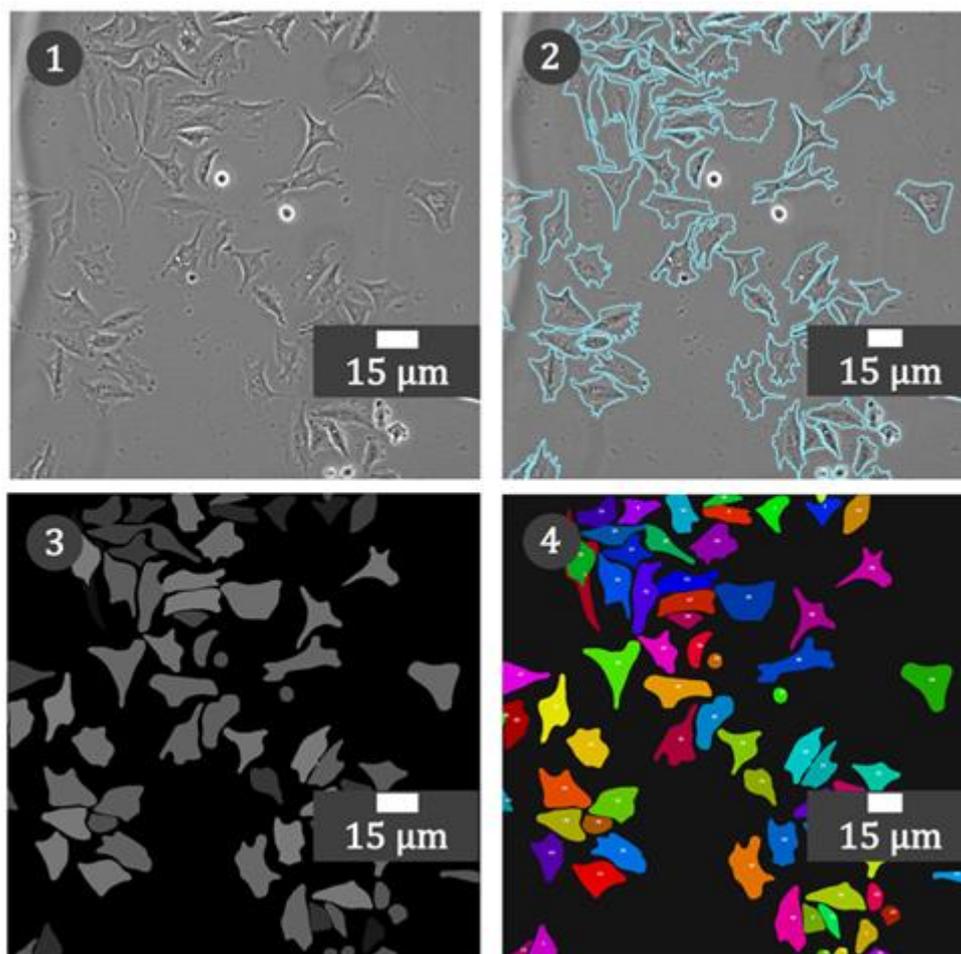


Figure 4 : Différentes étapes de la création des données d'entraînement avec la base de données de [5].  
1) Image en microscopie à contraste de phase. 2) Segmentation manuelle des cellules.  
3) Création de masques de segmentation. 4) Numérotation des masques de segmentation.

Un jeu de 50 images des cellules est utilisé. On segmente à la main les cellules en effectuant un détournage puis on génère des masques de segmentation à l'aide du logiciel ImageJ. On peut alors identifier de manière unique (par la couleur ou la numérotation) chaque cellule. 45 images de ce jeu de données sont utilisées pour l'entraînement et les 5 restantes permettent d'effectuer des tests de validation.

### 2.3 - Structure du réseau de neurone

La Figure 5 illustre la structure globale du réseau de neurones utilisé [10]. On peut dans un premier temps s'intéresser à la structure résiduelle. Il est notable que le nombre de couches utilisées ici (101) est assez élevé. Ceci permet d'avoir une meilleure précision sur les résultats obtenus.

Néanmoins cela mène à une structure de réseau complexe en couches qui, si elles sont empilées une à une successivement, peut conduire à une dégradation de la précision sur les résultats par rapport à des structures moins profondes.

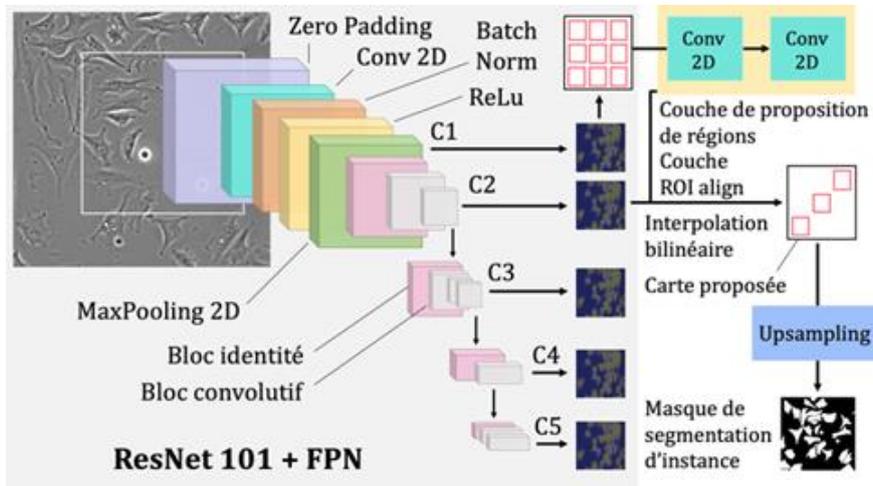


Figure 5 : Structure de Mask R-CNN, schéma adapté de [10]

Un problème connu pour les réseaux avec de nombreuses couches est la disparition du gradient (*gradient vanishing*) : lors de la rétropropagation du gradient de couches en couches ses valeurs diminuent rapidement ce qui entraîne son annulation totale et l'arrêt de l'apprentissage [11].

Pour pallier ce problème, les réseaux de neurones résiduels autorisent des sauts de connexions entre les couches. Avec cette structure l'apprentissage est plus rapide et moins vulnérable aux perturbations. Le réseau de neurones résiduel d'Usiigaci comprend deux types de blocs : des blocs identités et des blocs convolutifs décrits sur la Figure 6.

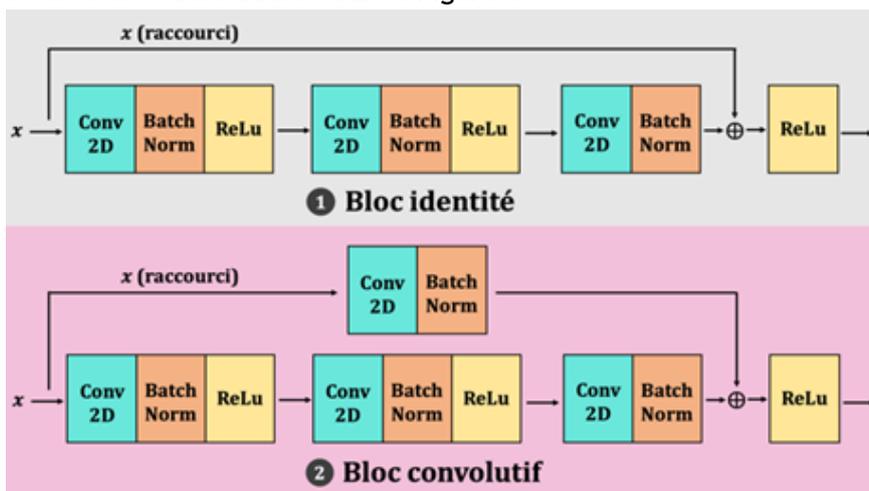


Figure 6 : Types de blocs du réseau de neurones résiduel

Le bloc identité est utilisé lorsque les dimensions de l'entrée et de la sortie sont identiques tandis que le bloc convolutif comprend une couche convolutive dans le chemin raccourci pour assurer une dimension égale entre l'entrée et la sortie. Ces deux types de blocs comprennent des fonctions identiques. Conv 2D effectue une convolution 2D. La fonction *Batch Normalization* rend le réseau plus rapide et plus stable en ajoutant des couches supplémentaires. Enfin la fonction ReLu

(*Rectifier Linear Unit*) est une fonction d'activation qui a pour but de « casser » la linéarité de l'image après les opérations linéaires de convolution. En effaçant certaines valeurs de l'image, les calculs sont accélérés et les caractéristiques mises en évidence par la convolution sont d'avantage surlignées.

La structure FPN combinée à la structure résiduelle permet la détection de cellules à plusieurs échelles. Cette structure fonctionne de telle sorte qu'à partir d'une image unique d'une taille donnée, des cartes de caractéristiques de tailles proportionnelles à plusieurs niveaux sont générées de manière convolutive [12]. Les pyramides ainsi créées sont alors invariantes avec l'échelle : lors d'une réduction ou d'une augmentation de la taille d'une cellule, le changement d'échelle est compensé par le déplacement dans la pyramide. Cette structure permet donc de détecter des cellules sur une gamme d'échelles plus étendue. Cette structure est schématisée sur la Figure 5 avec la succession de blocs roses et gris.

En sortie du réseau, une couche de propositions de régions génère des cadres qui délimitent les cellules potentiellement détectées. La couche ROI (Region Of Interest) align [13] permet ensuite d'aligner correctement les caractéristiques extraites des diverses opérations du réseau avec l'image d'entrée. Pour effectuer cela, une interpolation bilinéaire est utilisée, elle correspond à un redimensionnement de l'image [14]. La fonction Upsampling (suréchantillonnage) ramène enfin la résolution à celle de la couche précédente. On obtient finalement le masque de segmentation d'instance de la Figure 5.

### 3 – Détermination des paramètres d'intérêt

Une fois les masques de segmentation obtenus pour chaque image de la vidéo de la migration cellulaire, le module TrackPy calcule les paramètres d'intérêt décrits précédemment. Cela passe par l'utilisation d'un algorithme de recherche du plus proche voisin avec un arbre k-dimensionnel [15] qui ne sera pas détaillé ici.

On peut ainsi visualiser par exemple la trajectoire de chaque cellule de même que sur la Figure 7. Le module de traitement de données permet ensuite d'obtenir les graphes de la direction et la vitesse des cellules en fonction du champ électrique appliqué.

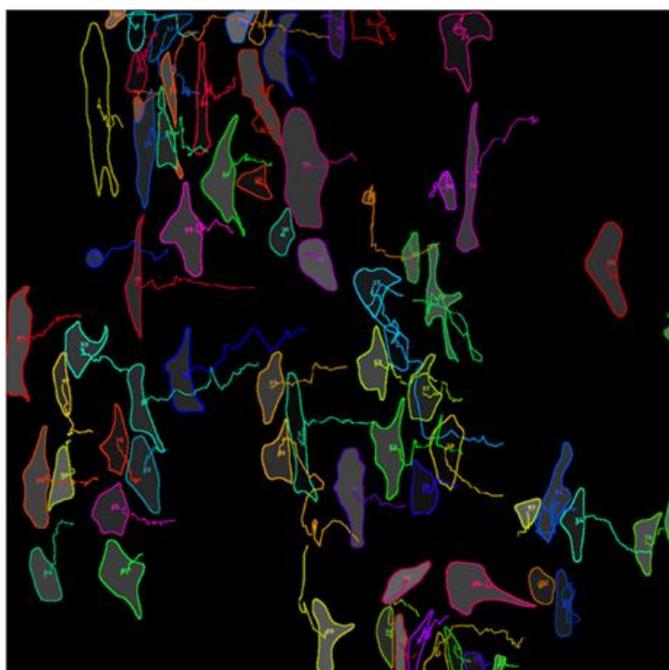


Figure 7 : Masques de segmentation et trajectoires de chaque cellule après traitement sous Trackpy, image issue de [5]

## 4 – Discussion

### 4.1 - Mise en œuvre du réseau de neurones

L'utilisation d'Usiigaci requiert l'accès à une carte graphique (GPU) pour effectuer des calculs parallélisés. Les différentes bibliothèques Python doivent être configurées en accord avec la configuration proposée dans [5] afin d'éviter les conflits entre bibliothèques et empêcher le fonctionnement de l'algorithme.

L'avantage d'Usiigaci est qu'il est basé sur un réseau Mask R-CNN pré-entraîné sur une base de données assez conséquente. De ce fait, l'entraînement requiert un jeu de données moins important que d'autres algorithmes comme Deepcell et cet entraînement prend moins de temps. Les *epochs* sont complétés plus rapidement mais un grand nombre est nécessaire pour obtenir une bonne précision. Des données issues de [10] concernant l'entraînement sont visibles sur la Figure 8.

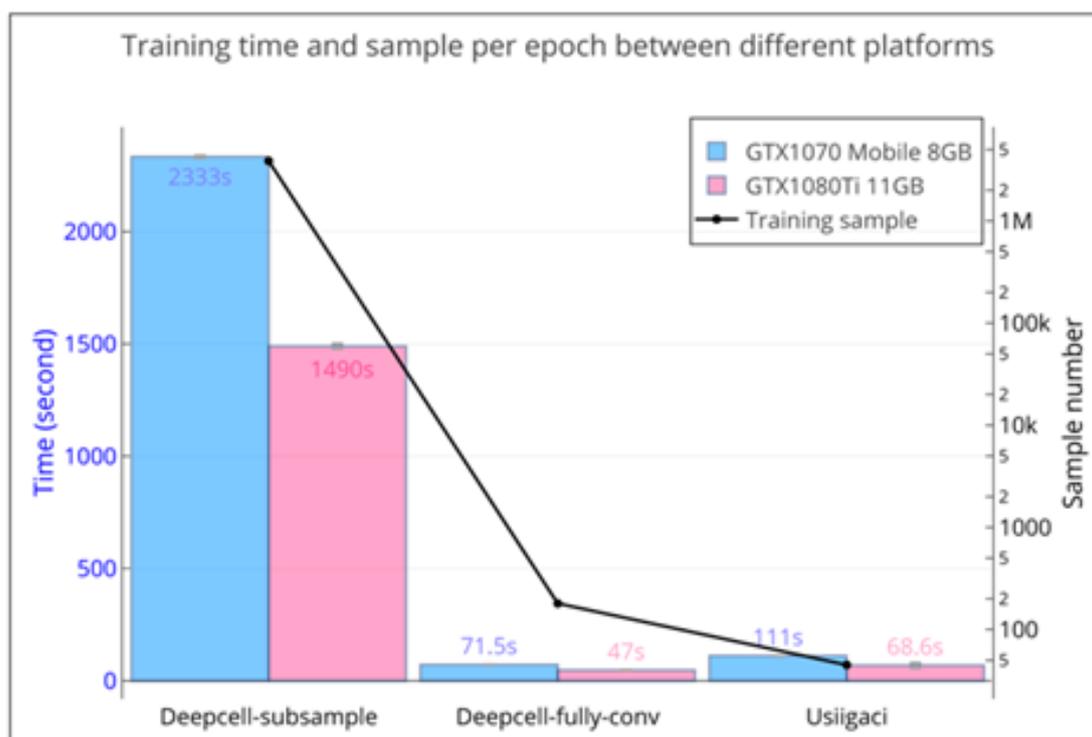


Figure 8 : Comparaison du temps d'entraînement et du nombre d'échantillons par epoch pour Usiigaci et Deepcell, avec deux modèles de carte graphique, graphe issu de [10]

En comparaison avec d'autres algorithmes, ce que nous verrons dans la partie suivante, la quantité de données utilisée (50 images) pour l'entraînement et les tests de validation est suffisante pour obtenir des résultats.

### 4.2 - Comparaison avec d'autres méthodes

On peut maintenant comparer l'algorithme Usiigaci à d'autres structures développées pour traiter ce type de problèmes. Pour effectuer cette comparaison plusieurs indicateurs ont été définis. L'un d'entre eux est l'indice de Jaccard qui permet d'évaluer la similarité entre l'ensemble d'images de départ et celles en sortie de l'algorithme. Cet indice est évalué dans [4] et représenté sur la Figure 9 pour différents algorithmes de segmentation.

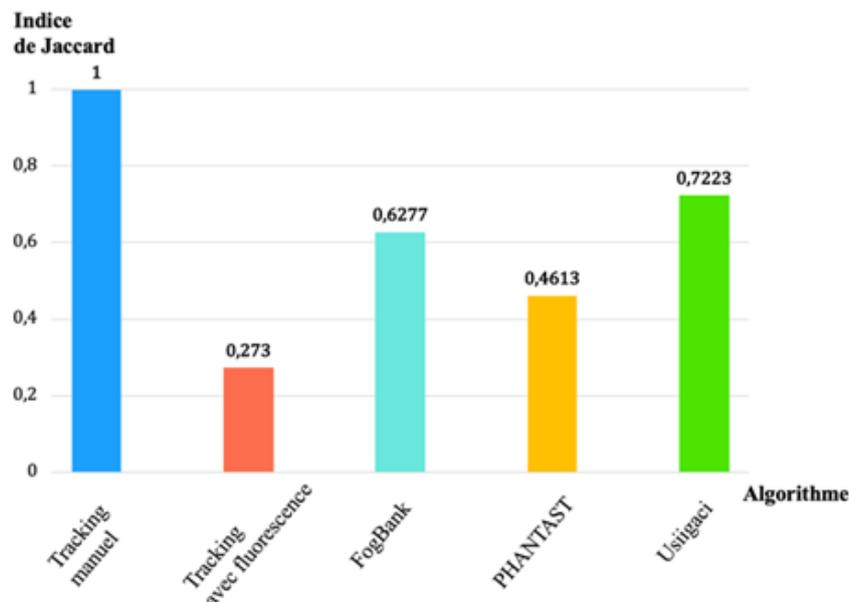


Figure 9 : Comparaison de l'indice de Jaccard pour différents algorithmes de segmentation, données issues de [4]

À première vue, une méthode qui semble simple à mettre en œuvre est le suivi manuel des cellules à l'aide d'un logiciel de *tracking* du type Tracker. Les résultats obtenus sont très satisfaisants du point de vue de l'indicateur considéré (indice de Jaccard de 1) mais le coût en temps peut être assez conséquent lorsqu'il faut traiter un grand nombre de vidéos des cellules afin que ces résultats soient représentatifs (environ 800 cellules pour 480 images). Usiigaci constitue donc un gain de temps considérable pour traiter les données comparé à cette méthode puisque la segmentation à la main n'est faite qu'une seule fois pour générer les données d'entraînement.

D'autres algorithmes de segmentation adaptés à la problématique de suivi cellulaire comme Fogbank [16] et PHANTAST [17] permettent d'obtenir des résultats en moins de temps que par du suivi manuel mais sont moins adaptés par rapport à Usiigaci pour le suivi cellule par cellule lorsque la densité cellulaire est élevée sur l'image. L'approche d'Usiigaci permet de traiter les données sans avoir recours à un traitement biologique particulier pour les cellules. Elles sont donc observées dans des conditions « naturelles ». D'autres méthodes de traitement d'images s'appuient sur l'utilisation de composés fluorescents pouvant être toxiques pour la cellule, qui ne sont pas requis ici pour obtenir des résultats satisfaisants.

Néanmoins comme beaucoup d'algorithmes de *deep learning*, si les images à traiter ont été prises dans des conditions trop éloignées de celles utilisées pour l'entraînement, des erreurs peuvent apparaître dans la segmentation et se répercuter dans le suivi cellulaire.

## 5 – Perspectives

Des perspectives qui pourraient être envisagées sont l'amélioration du dispositif expérimental (confection du système microfluidique, technologie d'électrodes utilisées ...) en le rendant le plus simple et portable possible, ainsi que la simplification de l'utilisation d'Usiigaci. L'idée serait d'automatiser les mesures afin que ce type de microsystèmes combiné à Usiigaci puissent être utilisés par des professionnels du milieu médical pour effectuer des diagnostics sur des patients de manière rapide et précise.

L'algorithme Usiigaci a été utilisé dans le cadre d'un stage de recherche en collaboration avec le LuMIn de l'ENS Paris-Saclay, l'Université de Freiburg en Allemagne et l'Université de Luleå en Suède.

## Références :

- [1]: J. Leal, N. Jedrusik, S. Shaner, C. Boehler, M. Asplund « SIROF stabilized PEDOT/PSS allows biocompatible and reversible direct current stimulation capable of driving electrotaxis in cells » *Biomaterials* 2021, Volume 275, August 2021, DOI: 10.1016/j.biomaterials.2021.120949
- [2]: A. Shehata, S. Mohammad, M. Sameer and M. Ehab Ragab « A Survey on Hough Transform, Theory, technique and Applications » *International Journal of Computer Science Issues*, Volume 12, Issue 1, No 2, January 2015, DOI: 10.48550/arXiv.1502.02160
- [3]: J. Crocker and D. Grier “Methods of Digital Video Microscopy for Colloidal Studies” *Journal of Colloid and Interface Science* 179, 298-310, 1996, DOI: 10.1006/jcis.1996.0217
- [4]: H-F. Tsai, J. Gadja, T. Sloan, A. Rares, A. Shen « Usiigaci: Instance-aware cell-tracking in stain-free phase contrast microscopy enabled by machine learning » *Software X*, Volume 9, 230-237 January-June 2019, DOI: 10.1016/j.softx.2019.02.007
- [5]: Usiigaci - Github: <https://github.com/oist/Usiigaci>
- [6]: J. Hui “Understanding Feature Pyramid Networks for object detection (FPN)” , Medium, 2018: <https://jonathan-hui.medium.com/understanding-feature-pyramid-networks-for-object-detection-fpn-45b227b9106c>
- [7]: K. He, X. Zhang, S. Ren, J. Sun “Deep Residual Learning for Image Recognition” , CVPR, 2016, DOI: 10.48550/arXiv.1512.03385
- [8]: Microsoft, COCO Common Objects in Context: <https://cocodataset.org/#home>
- [9]: Mask R-CNN for Object Detection and Segmentation - Matterport, Inc: [https://github.com/matterport/Mask\\_RCNN](https://github.com/matterport/Mask_RCNN)
- [10]: H. Tsai, J. Gadja, T. Sloan, A. Rares, A. Shen « Usiigaci: Instance-aware cell-tracking in stain-free phase contrast microscopy enabled by machine learning - Supplementary information » *Software X*, Volume 9, 230-237 January-June 2019, DOI: 10.1016/j.softx.2019.02.007
- [11]: C. Wang “The Vanishing Gradient Problem” , Medium, 2019: <https://towardsdatascience.com/the-vanishing-gradient-problem-69bf08b15484>
- [12]: T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan and S. Belongie “Feature Pyramid Networks for Object Detection”, CVPR, 2017, DOI: 10.48550/arXiv.1612.03144
- [13]: J. Chao “Understanding ROI Align”, SuperComputer’s Blog, July 2018 : <https://chao-ji.github.io/jekyll/update/2018/07/20/ROIAlign.html>
- [14]: J. Chao “Understanding Bilinear Image Resizing”, SuperComputer’s Blog, July 2018: <https://chao-ji.github.io/jekyll/update/2018/07/19/BilinearResize.html>
- [15]: « Arbres kd », Wikipédia, Juin 2019 : [https://fr.wikipedia.org/wiki/Arbre\\_kd](https://fr.wikipedia.org/wiki/Arbre_kd)
- [16]: J. Chalfoun, M. Majurski, A. Dima, C. Stuelten, A. Peskin and M. Brady “FogBank: a single cell segmentation across multiple cell lines and image modalities”, *BMC Bioinformatics*, 2014, DOI:10.1186/s12859-014-0431-x
- [17]: N. Jaccard, L.Griffin, A. Keser, R. Macown, A. Super, F. Veraitch and N. Szita “Automated Method for the Rapid and Precise Estimation of Adherent Cell Culture Characteristics from Phase Contrast Microcopy Images”, *Biotechnology and Bioengineering*, Vol. 111, No. 3, Mars 2014, DOI: 10.1002/bit.25115
- [18]: Dossier Intelligence Artificielle, juillet 2022, [https://eduscol.education.fr/sti/si-ens-paris-saclay/ressources\\_pedagogiques/dossier-intelligence-artificielle](https://eduscol.education.fr/sti/si-ens-paris-saclay/ressources_pedagogiques/dossier-intelligence-artificielle)