

**3A 2020**

## **Rapport Final**

**Filière Métier : R&D**

**« Segmentation des images par réseau U-Net »**

**Réalisé par :**

- BANI Hachim
- EL MEKKI Noufissa
- EDDAHMANI Ikram
- RADID Youness

**Encadré par :** M. BOURENANE Salah

## **Résumé :**

Dans cet article, nous allons présenter une approche de segmentation d'images médicales, et plus précisément, les images des noyaux de cellules basée sur l'apprentissage profond. Cette approche repose principalement sur l'architecture des réseaux de neurones convolutionnels U-Net. La méthode proposée fournit de meilleurs résultats de segmentation que les méthodes classiques utilisées auparavant.

## **Mots clés :**

Apprentissage profond, Segmentation, Réseau U-Net.

## **Abstract:**

In this article, we will present a deep learning approach for the segmentation of medical images, and more precisely, images of cell nuclei. This approach is mainly based on the U-Net convolutional neural network architecture. The proposed method provides better segmentation results than classical methods already used.

## **Keywords :**

Deep Learning, Segmentation, U-Net network.

### **1. Introduction et contexte :**

Le deep Learning fait partie du domaine de l'intelligence artificielle et c'est une technique qui nous permet de faire des prédictions à partir d'un ensemble de données. Pour pouvoir faire des prédictions correctes la machine doit apprendre, d'où le terme Learning. Prenons par exemple, un algorithme qui serait capable à partir d'une image de nous dire exactement ce dont il s'agit. Il est totalement normal qu'au début les résultats soient erronés vu que la machine n'a pas encore appris. Pour remédier à ce problème, l'apprentissage se fera manuellement à l'aide d'un training set (Apprentissage supervisé). Une fois la phase d'entraînement terminée on pourrait donc donner n'importe quel type d'image à notre machine et recevoir en sortie son contenu. Le deep Learning est basé sur le réseau de neurones, qui comme son nom l'indique est

composée de plusieurs neurones connectés entre eux qui appartiennent à différentes familles : la famille des “**inputs**”, la famille des “**hidden layers**” et la famille des “**outputs**”.

Or, avec l'avènement du deep Learning, on a su implémenter les différentes techniques de traitement d'image avec les différents réseaux de neurones. D'où, la segmentation s'est montrée efficace étant une technique de traitement d'images et qui consiste à définir des frontières permettant ensuite de labelliser des formes ou des figures bien précises.

Dans le cadre de la filière recherche et développement, nous nous sommes mis en collaboration avec notre encadrant M. BOURENANE Salah de l'institut Fresnel de partir sur un sujet basé sur l'apprentissage profond. Malheureusement, cette année à cause de ces conditions sanitaires nous n'avons pas pu travailler au sein d'une équipe au laboratoire alors nous avons suivi la méthodologie suivante :

- Etude bibliographique sur le sujet.
- Répartir les tâches.
- Réunion de groupe pour évaluer les recherches faites.
- Réunion avec M. BOURENANE après chaque phase pour mettre le point.

Notre but dans ce sujet consiste à utiliser le réseau U-net afin de segmenter les cellules présentes dans des images microscopiques. Nous avons choisi de se focaliser sur le réseau U-net, comme son nom l'indique ce dernier est basé sur une architecture de réseau convolutionnel sous formes de U. Ainsi, cette architecture nous permet dans un premier lieu d'effectuer une contraction de l'image afin de capturer le contexte ensuite vient la phase de l'expansion dont le rôle principal est la localisation.

Nous évoquerons dans la partie 1 de cet article les notions de bases du deep learning, les réseaux de neurones convolutionnels ainsi que l'architecture U-Net. Nous verrons ensuite les mécanismes derrière la segmentation par CNN tout en discutant l'intérêt du réseau U-net, tandis que les résultats de l'expérimentation effectuée seront exposés dans la partie 4. Enfin, la

dernière partie de cet article sera consacrée à la discussion de l'efficacité de cette approche.

## 2. Définitions et notions de bases :

### 2.1. Réseau de neurones à convolutif :

Un sous type de réseau de neurones dédié à la reconnaissance de formes ou analyse des images. En effet, le réseau prend à l'entrée une image qui sera renvoyé dans le réseau de neurone à convolution et renvoie comme sortie la probabilité d'appartenance à une catégorie bien défini.

### 2.2. Architecture d'un réseau de neurones à convolution :

- **La couche de convolution :**

L'idée derrière cette couche est de filtrer les images avant d'entraîner le réseau de neurones profond. Le principe est de définir un filtre convolutif appelé aussi kernel et calculer le produit de convolution entre ce filtre la et chaque portion de l'image balayé.

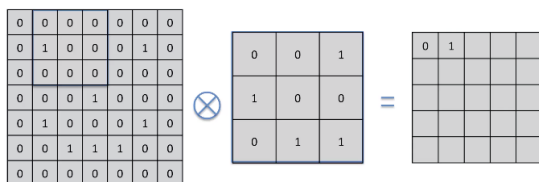


Figure 1 Convolution 3x3

On prend la matrice de convolution 3\*3 et on l'impose sur notre image 7\*7, puis on multiplie terme à terme et après on additionne tous les résultats, et on glisse le carré à droite d'une colonne.

On trouve comme résultat, une image réduite en termes de taille 5\*5, on perd un peu d'information mais on garde les pixels qui contiennent les caractéristiques importantes.

On réitère la même opération pour le second filtre et pour le n-ème filtre de convolution afin de garder que les pixels importants et donc extraire les caractéristiques d'une image puis l'enregistrer sur une image résultante.

- **Couche Relu :**

Après avoir complété l'étape de convolution, on va utiliser la fonction d'activation « Redresseur » pour que notre modèle soit non-linéaire.

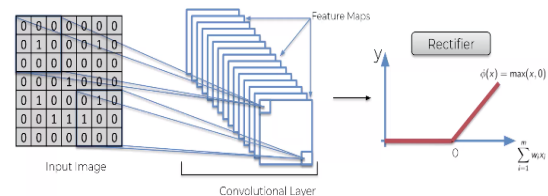


Figure 2 Fonction d'activation ReLu

- **Couche MaxPooling :**

Une forme de sous échantillonnage des images, qui consiste à faire passer sur l'image une fenêtre généralement 2à 2 et de la faire évoluer par un pas de 2. Pour chaque position de la fenêtre sur l'image le max pooling renvoie la valeur maximale parmi les 4 pixels. On trouve en sortie l'image réduite avec seulement les pixels significatifs.

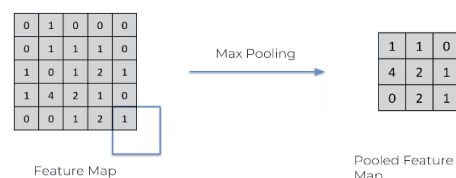


Figure 3 Max Pooling

- **Etape Flattening :**

Sert à aplatir complètement la matrice résultat de la couche de pooling parce que on va la placer dans un réseau de neurone.



Figure 4 Flattening

Ce nouveau vecteur, nous donne la couche d'entrée d'un nouveau réseau de neurones.

- **Couche Complètement connecté :**

On envoie le vecteur précédent dans un nouveau réseau de neurones artificielle complètement connecté, c'est à dire que toutes les couches cachées sont connectées avec les neurones de la couche précédente par des liaison synapses.

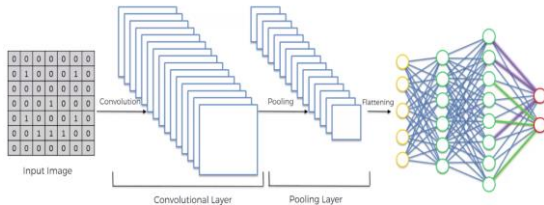


Figure 5 Couche Fully Connected

- **Comment fonctionner un CNNs :**

Après avoir compléter toutes les couches précédentes, on procède de la façon suivante :

On envoie une information d'entrée dans le réseau, elle va se propager via les synapses et le réseau va prédire une sortie.

La prédiction peut être correcte ou non, on compare la sortie obtenue par la sortie que l'on souhaitait obtenir, et on calcule l'erreur de prédiction.

L'erreur est propagée dans le réseau en utilisant l'algorithme de retro-propagation, afin de minimiser l'erreur en agissant sur la valeur des poids synaptiques et sur les filtres de convolutions.

### 2.3. Architecture U-Net :

Le réseau U-net est basé entièrement sur le réseau de neurone convolutionnel, et conçu de manière à donner des meilleurs résultats de segmentation des images.

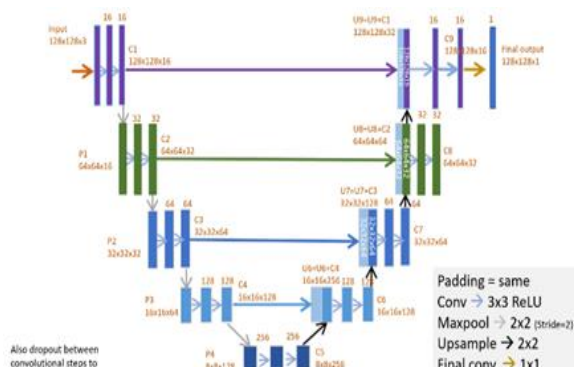


Figure 6 Architecture U-Net

L'architecture U-net est symétrique.

Il peut être réduit en 2 parties principales :

- **Une phase de Contraction** : sous-échantillonnage
- **Une Phase d'expansion** : sur-échantillonnage

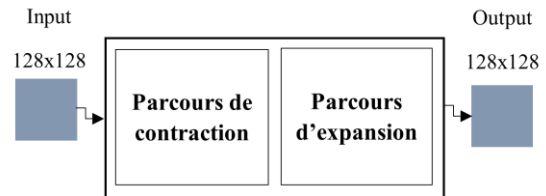


Figure 7 Schéma des deux phases U-Net

### 3. Expérimentation :

Dans cette section, nous détaillons l'expérimentation que nous avons menée. Afin d'analyser les performances de cette approche, nous avons choisi une base de données du site **Kaggle** contenant des images avec leurs masques exactes.

#### a. Acquisition de données :

La base de données utilisée contient un grand nombre d'images de noyaux segmentés. Les images ont été acquises dans diverses conditions et varient selon le type de cellule, le grossissement et le mode d'imagerie (fond clair ou fluorescence). L'ensemble de données est conçu pour mettre au défi la capacité d'un algorithme à généraliser à travers ces variations.

Chaque image est représentée par un identificateur associé. Les fichiers appartenant à une image sont contenus dans un dossier portant cet identificateur. Ce dossier contient deux sous-dossiers :

- **Images** contient le fichier image.
- **masks** contient les masques segmentés de chaque noyau. Ce dossier est uniquement inclus dans l'ensemble d'entraînement. Chaque masque contient un noyau. Les masques ne sont pas autorisés à se chevaucher (aucun pixel n'appartient à deux masques).

L'ensemble de données de la deuxième étape contiendra des images provenant de conditions expérimentales inconnues. Afin de décourager l'étiquetage manuel, il contiendra également des images qui ne sont pas prises en compte dans l'évaluation.

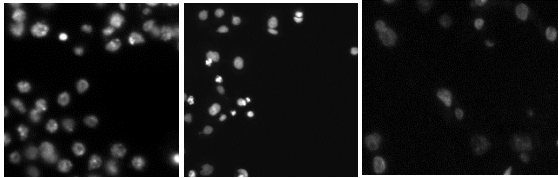


Figure 8 Exemple d'images à segmenter

### b. Architecture du réseau :

Nous construisons un réseau de neurones composé de deux parties : Une partie de contraction et une partie d'expansion.

La première partie suit l'architecture typique d'un réseau de neurones convolutionnel. En effet, c'est une répétition de deux **3x3 convolutions**, chacune est suivie par une unité linéaire redressée **ReLU** et une opération de **2x2 max pooling** avec un pas de 2 pour le sous-échantillonnage (C1, C2, C3, C4, C5). À chaque étape de sous-échantillonnage, nous doublons le nombre de canaux de caractéristiques.

Pour la deuxième partie, chaque étape consiste en un sur-échantillonnage suivi d'une convolution 2x2 (« **up-convolution** » ou « **trans-convolution** ») qui divise par deux le nombre de canaux de caractéristiques, ensuite une concaténation avec la carte de caractéristiques rognée en conséquence puis deux convolutions de 3x3, chacune suivie d'une ReLU (C6, C7, C8, C9).

Tout d'abord nous avons modifié la taille des images à pleine résolution à 128x128. Donc la sortie sera une image qui contient un masque des cellules dont la taille est toujours 128x128.

### c. Implémentation de l'approche :

Comme a été déjà mentionné, le réseau de neurones U-Net est une répétition de deux différentes opérations.

#### ➤ Contraction (C1, C2, C3, C4, C5) :

Cette partie commence par des convolutions 16x16.

```
#Contraction path
c1 = tf.keras.layers.Conv2D(16, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(s)
c1 = tf.keras.layers.Dropout(0.2)(c1)
c1 = tf.keras.layers.Conv2D(16, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(c1)
p1 = tf.keras.layers.MaxPooling2D((2, 2))(c1)
```

Figure 9 Code de la partie contraction

#### ➤ Expansion (C6, C7, C8, C9) :

Cette partie commence par une trans-convolution 128x128.

```
#Expansive path
u6 = tf.keras.layers.Conv2DTranspose(128, (2, 2), strides=(2, 2), padding='same')(c5)
u6 = tf.keras.layers.concatenate([u6, c4])
c6 = tf.keras.layers.Conv2D(128, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(u6)
c6 = tf.keras.layers.Dropout(0.2)(c6)
c6 = tf.keras.layers.Conv2D(128, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(c6)
```

Figure 10 Code de la partie expansion

U6 est temporaire qui effectue la concaténation entre C4 et la sortie de Trans-convolution.

L'entraînement et les prédictions de ce modèle ont été faits sur 670 images de cellules différentes en se basant sur les paramètres suivants :

- validation split=0.1
- batch size=16
- epochs=25

```
preds_train = model.predict(X_train[:int(X_train.shape[0]*0.9)], verbose=1)
preds_val = model.predict(X_train[int(X_train.shape[0]*0.9):], verbose=1)
preds_test = model.predict(X_test, verbose=1)

preds_train_t = (preds_train > 0.5).astype(np.uint8)
preds_val_t = (preds_val > 0.5).astype(np.uint8)
preds_test_t = (preds_test > 0.5).astype(np.uint8)
```

Figure 11 Code de la partie prédiction

## 4. Résultats et discussion :

### a. Résultats de la segmentation :

Après l'implémentation de l'approche choisie et l'entraînement du réseau U-Net, nous avons effectué le test sur quelques images cellulaires de la base de données.

Les résultats de prédictions nous ont permis de conclure que notre modèle est précis :

- **Loss:** 0.0921, **Accuracy:** 0.9640
- **Val loss:** 0.0872, **Val accuracy:** 0.9667

A partir d'une image originale en entrée du réseau avec un masque contenant les éléments à

segmenter/partitionner de l'image, nous avons obtenus les résultats suivants :

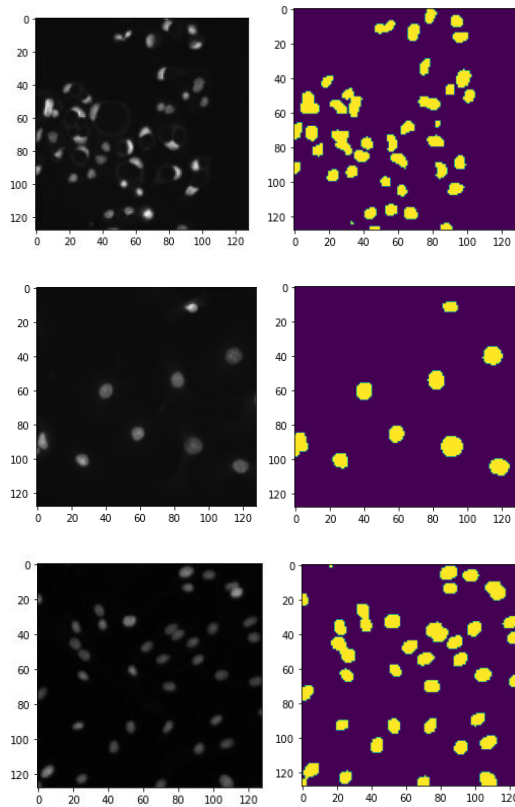


Figure 12 Images originales Images Segmentées

Nous remarquons que les images prédites sont pratiquement proches de ce que nous souhaitons avoir à la fin de la segmentation. Ceci, est dû à la robustesse du model U-Net.

#### b. Discussion :

L'architecture U-Net a donné de très bonnes performances au niveau de la segmentation des cellules. Néanmoins, cette approche a ses propres limitations et reste vulnérable vis-à-vis de précision du contour, de débruitage et de détection de petites cellules.

L'expérimentation présentée dans cet article nous confirme que le réseau U-Net peut échouer à représenter les cellules avec une forme elliptique comme l'image originale. Ceci est du parfois au bruit appliqué sur l'image et parfois non.

La figure ci-dessous, nous montre que le masque d'une image de la base de données contenait des cellules qui n'ont pas été présentées dans l'image résultante. En effet, ces

cellules, étant très petites par rapport aux autres, n'ont pas été prises en compte par le réseau.

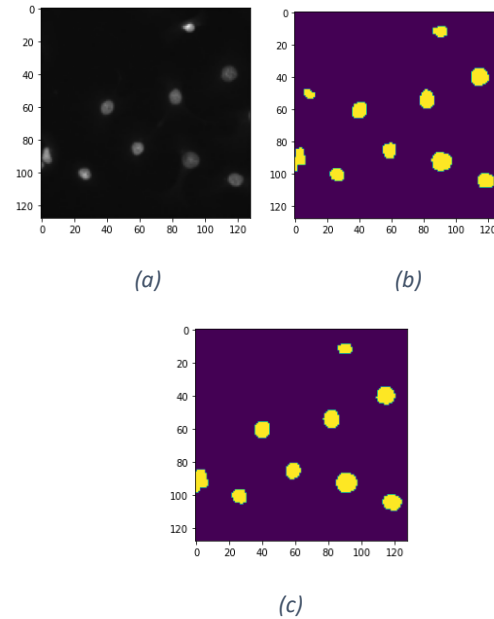


Figure 13 : (a) image d'entrée, (b) masque appliqué, et (c) image de sortie segmentée.

#### 5. Conclusion :

Nous avons présenté dans cet article une approche de segmentation basée sur les réseaux de neurones convolutionnels avec une nouvelle architecture plus élégante et plus rapide : **Réseau U-Net**. Cette approche a atteint de meilleurs résultats de segmentation sur la base de données des cellules médicales par rapport au réseaux convolutifs existants.

Afin de remédier aux limites de cette architecture U-Net classique, une nouvelle architecture DC-UNet (Dual Channel U-Net) a été mise en place avec un nombre réduit de paramètres. Ce modèle rend les résultats meilleurs grâce à ses caractéristiques multi résolution doublées. Cette nouvelle approche s'est avérée capable de relever le défi des images médicales nécessitant une segmentation à petite échelle.

#### Références :

<https://www.lebigdata.fr/reseau-de-neurones-artificiels-definition>

<https://stanford.edu/~shervine/l/fr/teaching/cs-230/pense-bete-reseaux-neurones-convolutionnels>

<https://deeplylearning.fr/cours-theoriques-deep-learning/fonction-dactivation/>

<https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net/>

<https://ichi.pro/fr/construire-une-architecture-u-net-pour-la-segmentation-d-images-biomedicales-83923601250600>

<https://www.kaggle.com/c/data-science-bowl-2018/overview>

<https://heartbeat.fritz.ai/deep-learning-for-image-segmentation-u-net-architecture-ff17f6e4c1cf>

<https://towardsdatascience.com/understanding-semantic-segmentation-with-unet-6be4f42d4b47>

<https://arxiv.org/ftp/arxiv/papers/2006/2006.00414.pdf>