Rapport Challenge Kaggle

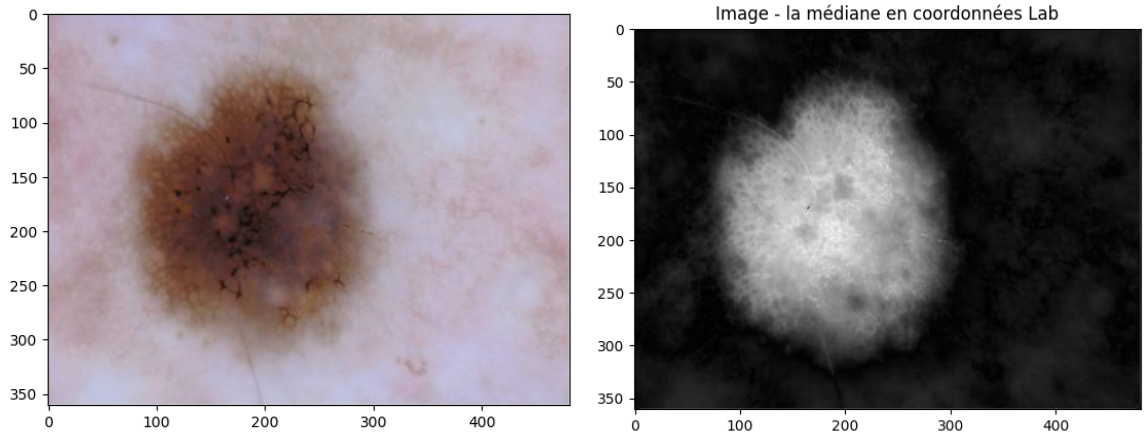
IMA205

Philippe Gratias-Quiquandon

Introduction

Pour réaliser ce challenge, j'ai lu quelques articles que j'ai mis en pièce-jointe et que je citerai tout au long de ce rapport. J'ai décidé de suivre la segmentation proposé par l'article « Segmentation of skin cancer images » [1] en ajoutant quelques étapes personnelles afin d'améliorer les résultats, en particulier pour ce qui est pré et post-processing.

Le principe de la segmentation que j'ai utilisé est assez simple : il s'agit de calculer les médianes en coordonnées LAB de l'image, en supposant que la peau est majoritaire sur l'image. Ainsi, les trois médianes calculées correspondraient à la peau. Enfin, pour extraire la lésion, on fait la norme de la différence de chaque pixel avec ces trois coordonnées. Ainsi, la peau devrait être de faible intensité tandis que la lésion ressort clairement : il ne nous reste plus qu'à faire un seuillage adaptatif.



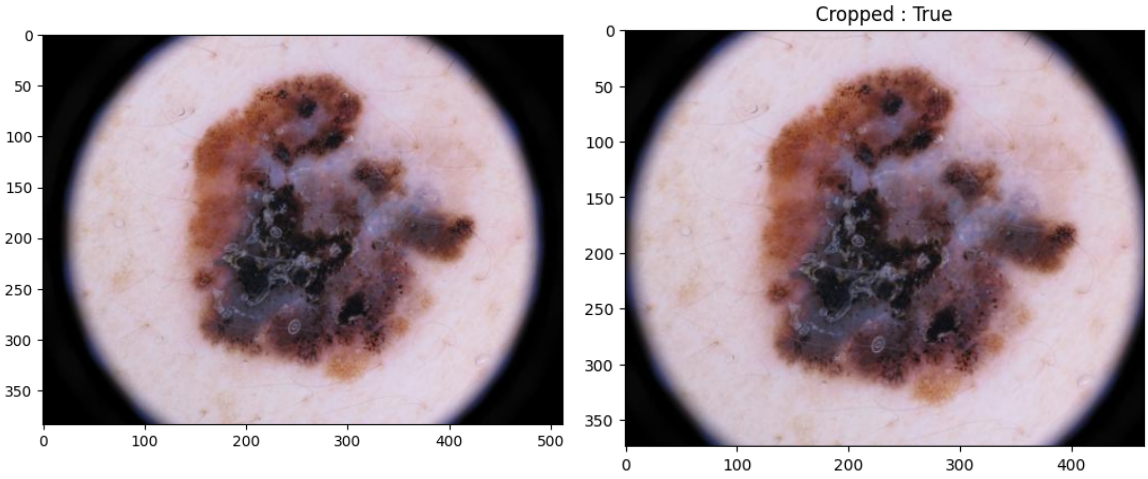
Pour ce qui est de l'extraction de features, je me suis basé sur plusieurs articles : « Combination of features from skin pattern and ABCD analysis for lesion classification » [2] et « Automatic detection of blue-white veil and related structures in dermoscopy images » [3]. Il y a également l'article « A survey, review, and future trends of skin lesion segmentation and classification » [4] qui faisait le récapitulatif des features les plus intéressantes et qui font le plus l'objet de recherches à l'heure actuelle.

Enfin, avec ces features extraites dans le fichier .csv, j'entraîne un SVM ce qui me donnera une soumission qui aura comme score 0.507 en public et 0.503 en privé.

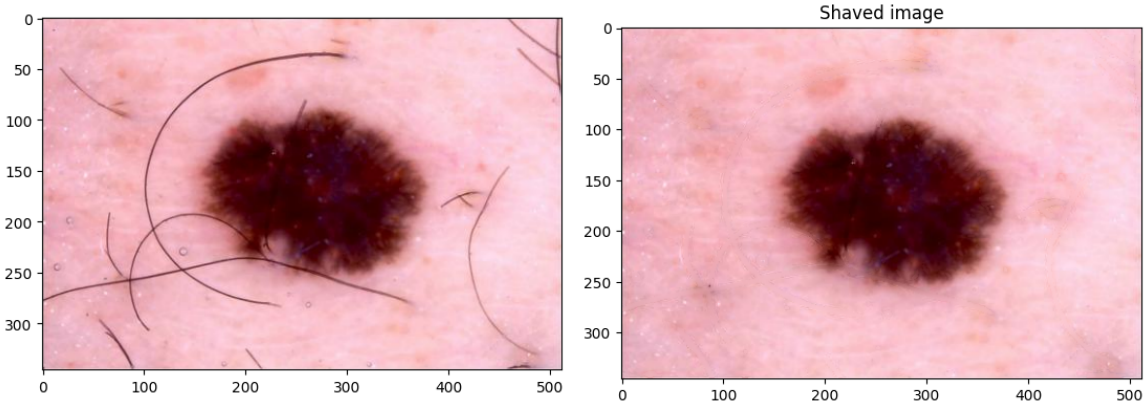
Il y a également une dernière partie sur l'entraînement d'un CNN en utilisant ResNet 101, ce qui me donnera au mieux un score de 0.473 en public et 0.420 en privé.

Pre-processing

Compte-tenu de la segmentation qui, pour rappel, prévoit une majorité de peau sur l'image, il faut retirer au maximum les bordures noires. Je commence donc par effectuer un crop si les bordures sont noires afin de recadrer au mieux l'image, l'idée étant d'éviter que la médiane soit le noir et pas la peau. Ici, à gauche l'image d'origine et à droite, le crop :



L'article revue « Computerized analysis of pigmented skin lesions: A review » [5] explique les pre-processings que l'on utilise fréquemment. En particulier, il est nécessaire de raser numériquement les images car les poils peuvent créer des parasites pour la création du masque. J'utilise donc l'algorithme du DullRazor comme illustré ici : https://github.com/BlueDokk/Dullrazor-algorithm

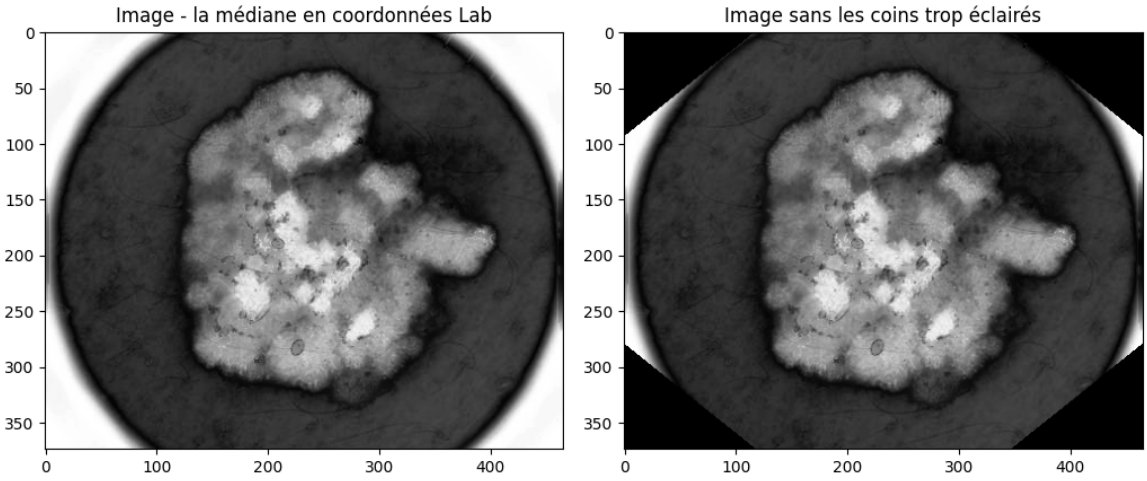


Segmentation

On filtre l'image avec un filtre médian avec un kernel_size de 10 pixels pour débruiter au mieux l'image et éviter de choisir des mauvaises médianes. Ensuite, on calcule L*, a* et b* les médianes des trois canaux et on calcule pour chaque pixel :

$$\Delta E = \sqrt{(L - L^*)^2 + (a - a^*)^2 + (b - b^*)^2}$$

Cela nous donne une image en niveaux de gris comme la figure en page 3. Si on a crop l'image, cela signifie qu'il nous reste des bordures qui vont avoir des plus grandes valeurs que la lésion, on les retire simplement :



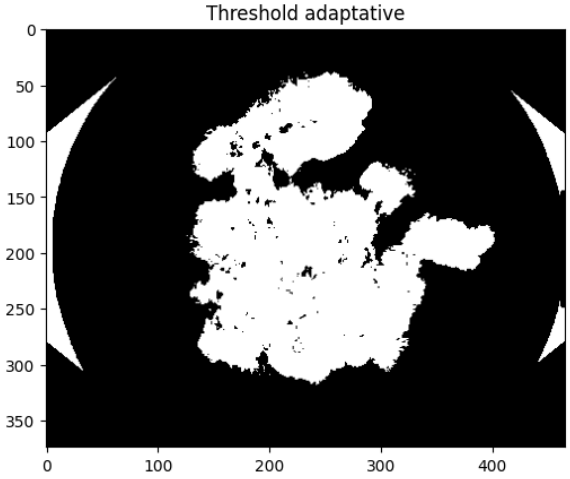
Pour avoir des meilleurs résultats, on aurait pu aussi trouver un masque en forme de disque qui écarteraient alors totalement les bordures noires. Néanmoins, cela ne gêne pas pour la suite.

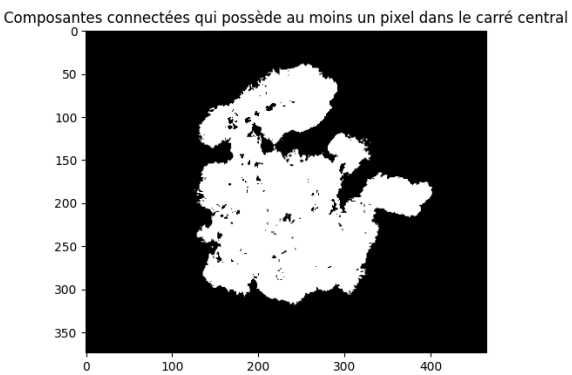
À partir de cette image, on applique un premier seuillage d'Otsu. La méthode d'Otsu cherche à maximiser la variance inter-classes en regardant l'histogramme. J'utilise la bibliothèque OpenCV qui l'implémente directement. Le problème avec cette méthode d'après l'article « A simple weighted thresholding method for the segmentation of pigmented skin lesions in macroscopic images » [6], est que la méthode d'Otsu fournit des seuillages surestimés et qu'on fait donc des segmentations incomplètes. La parade à cette difficulté est de faire la méthode SWOT que propose l'article.

La méthode SWOT consiste à sélectionner la valeur des pixels qui, dans l'histogramme, sont aux quantiles 0.05 et 0.5. On les nomme $\gamma_{0.05}$ et $\gamma_{0.5}$ et on retourne un nouveau seuillage : $\gamma_{0.5} + \beta\gamma_{0.05}$ ici on prend $\beta = 1$.

Enfin, on fait un compromis entre ces deux seuillages avec un paramètre α : $\alpha t_{otsu} + (1 - \alpha)t_s$

La dernière étape dans la segmentation est un ajout personnel pour corriger quelques problèmes qui pourraient survenir. On applique l'algorithme des composantes connectées et on regarde dans un carré central de l'image quels sont les labels présents. La segmentation sera les labels présents.

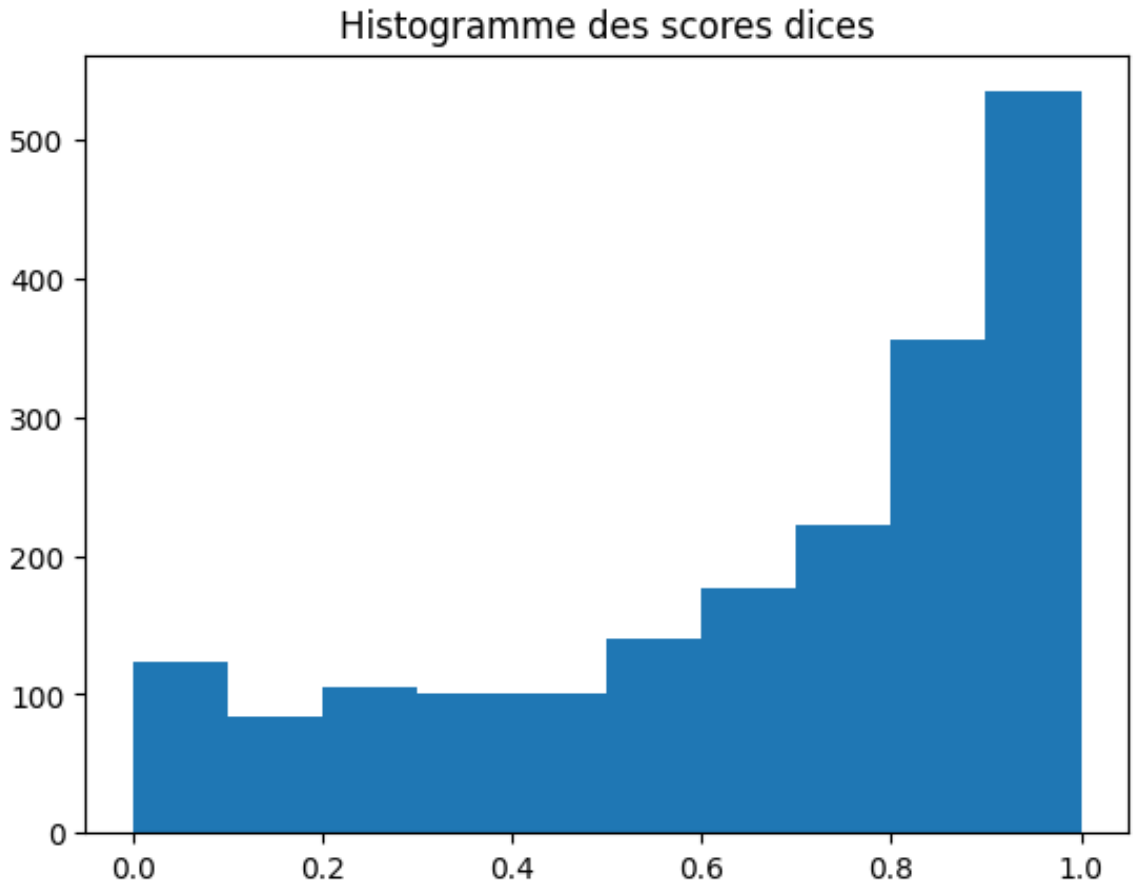




Post-processing et performance de la segmentation

Il est clair que l'on peut encore améliorer un petit peu la segmentation, par exemple en comblant les trous dans le masque. Pour ça, j'effectue une fermeture. J'ai ajouté également une deuxième étape permettant de prendre l'inverse du masque si celui-ci ne contient que de la peau. Ceci est pour éviter les segmentations erronées car il y a plus de zone de lésion que de peau sur l'image (donc la segmentation sélectionne la peau au lieu de la lésion).

Pour avoir une idée de la performance de la segmentation, j'ai calculé sur toutes les images pré-segmentées, le score dice que j'ai ensuite afficher sous forme d'histogramme :



La médiane de ce score est de 0.77 et sa moyenne est de 0.66 et d'après l'histogramme, la segmentation produit en général un résultat satisfaisant, mais

il reste beaucoup de segmentations qui sont complètement fausses, en particulier dû aux photos où la peau n'est pas majoritaire sur l'image.

Extraction des features

Voici un récapitulatif des features extraites sous forme de tableau :

Feature	Type
$R_L = \text{moyenne}(R \in \text{lesion})$	Couleur
$R_S = \text{moyenne}(R \in \text{skin})$	Couleur
$G_L = \text{moyenne}(G \in \text{lesion})$	Couleur
$G_S = \text{moyenne}(G \in \text{skin})$	Couleur
$B_L = \text{moyenne}(B \in \text{lesion})$	Couleur
$B_S = \text{moyenne}(B \in \text{skin})$	Couleur
$R_L - R_S$	Couleur
$G_L - G_S$	Couleur
$B_L - B_S$	Couleur
R_L / R_S	Couleur
G_L / G_S	Couleur
B_L / B_S	Couleur
$R_L / (R_L + G_L + B_L)$	Couleur
$G_L / (R_L + G_L + B_L)$	Couleur
$B_L / (R_L + G_L + B_L)$	Couleur
R_L / R_S / (somme des features X_L / X_S)	Couleur
G_L / G_S / (somme des features X_L / X_S)	Couleur
B_L / B_S / (somme des features X_L / X_S)	Couleur
$R_L - R_S$ / (somme des features $X_L - X_S$)	Couleur
$G_L - G_S$ / (somme des features $X_L - X_S$)	Couleur
$B_L - B_S$ / (somme des features $X_L - X_S$)	Couleur
$R_{max}, G_{max}, B_{max} = \max(R, G, B \in \text{lesion})$	Couleur
$R_{min}, G_{min}, B_{min} = \min(R, G, B \in \text{lesion})$	Couleur
x, y, a, b, θ = paramètres de l'ellipse	Asymétrie
Surface de l'ellipse	Diamètre / Taille
Diamètre de l'ellipse	Diamètre / Taille

Périmètre du masque	Bordure
Surface du masque	Diamètre / Taille
$(\Delta T/T) \times 100$, où T est la surface du masque, ΔT, le nombre de pixels appartenant au masque et pas à l'ellipse qui fit le masque	Asymétrie
$P^2/(4\pi T)$ où P est le périmètre du masque	Bordure
$\text{std}(X)/\max(X)$ où X est un channel et les pixels sont dans le masque	Couleur
Moyenne/var des positions des pixels dans le masque par rapport aux centroids (cf. [3])	Asymétrie / Bordure
Ellipticité de la lésion (cf. [3], feature S3)	Asymétrie / Bordure

Cela nous fait déjà une quarantaine de features mais j'ai essayé, sans succès, de rajouter des features liées à la texture comme, par exemple, les gray level co-occurence matrix ou les histogrammes de gradients orientés.

Pour ce qui est des données manquantes comme l'âge ou le sexe qui ne sont pas mentionnés à certains endroits, on utilise des imputer pour remplacer des valeurs non-existantes par la moyenne. Enfin, il y a aussi quelques valeurs qui valent inf que l'on remplace par un grand nombre pour que l'entraînement du SVM fonctionne.

Machine Learning : SVM

J'ai choisi d'utiliser un SVM sur les features du fait de sa popularité : d'après [4], le SVM est la méthode de machine learning la plus utilisée dans les articles entre 2011 et 2022. J'ai séparé le dataset des 18999 images en train set et test set pour vérifier le bon fonctionnement du SVM sur de nouvelles données.

J'applique sur ces données, deux pré-processings ; d'abord un StandardScaler() et ensuite un SimpleImputer() qui fait ce qui a été expliqué dans la partie précédente, c'est-à-dire, remplacer les valeurs manquantes.

Ensuite, il est indiqué que le SVM qui fournit les meilleurs résultats est celui avec un noyau gaussien (i.e kernel = radial basis function). On règle donc les deux paramètres C et γ afin d'avoir le plus grand test score. J'ai trouvé $C = 100$ et $\gamma = 0.1$. J'obtiens comme score de test 0.7 et en score de soumission environ 0.5.

J'ai ensuite essayé d'améliorer le modèle en utilisant un RandomOverSampler en me disant que l'on pouvait augmenter la performance du SVM en fournissant plus de classes minoritaires (typiquement les classes 6,7 et 8). Cela réduisait les performances sur les soumissions car le modèle prédisait alors trop de classes minoritaires par rapport à ce qu'il y avait réellement.

Je pense que j'aurai encore pu améliorer le SVM en ajoutant les features de texture et en réglant mieux la segmentation lorsqu'elle est défaillante. Ainsi, on aurait pu mieux détecter les outliers.

Convolutional Neural Network : ResNet 101

J'ai également entraîné un CNN en reprenant l'architecture de ResNet 101. J'ai utilisé la Cross-Entropy Loss pondéré avec les poids donnés dans le kaggle concernant les proportions de chaque classe au sein du dataset. En optimizer, j'ai utilisé Adam avec un learning rate de 0.001. Après avoir séparé le dataset en deux sous-ensembles de train et de validation, j'entraîne le réseau et au bout de 25 époques, j'ai soumis ces prédictions et obtenu au mieux un score de 0.473 en public et 0.420 en privé.

Bibliographie

[1] L. Xu, M. Jackowski, A. Goshtasby, D. Roseman, S. Bines, C. Yu, A. Dhawan, A. Huntley, Segmentation of skin cancer images.

[2] She Z, Liu Y, Damatoa A. Combination of features from skin pattern and ABCD analysis for lesion classification.

[3] Celebi ME, Iyatomi H, Stoecker WV, Moss RH, Rabinovitz HS, Argenziano G, Soyer HP. Automatic detection of blue-white veil and related structures in dermoscopy images.

[4] Md. Kamrul Hasan, Md. Asif Ahamad, Choon Hwai Yap, Guang Yang, A survey, review, and future trends of skin lesion segmentation and classification.

[5] Konstantin Korotkov, Rafael Garcia, Computerized analysis of pigmented skin lesions: A review

[6] Maciel Zortea, Eliezer Flores, Jacob Scharcanski, A simple weighted thresholding method for the segmentation of pigmented skin lesions in macroscopic images.