

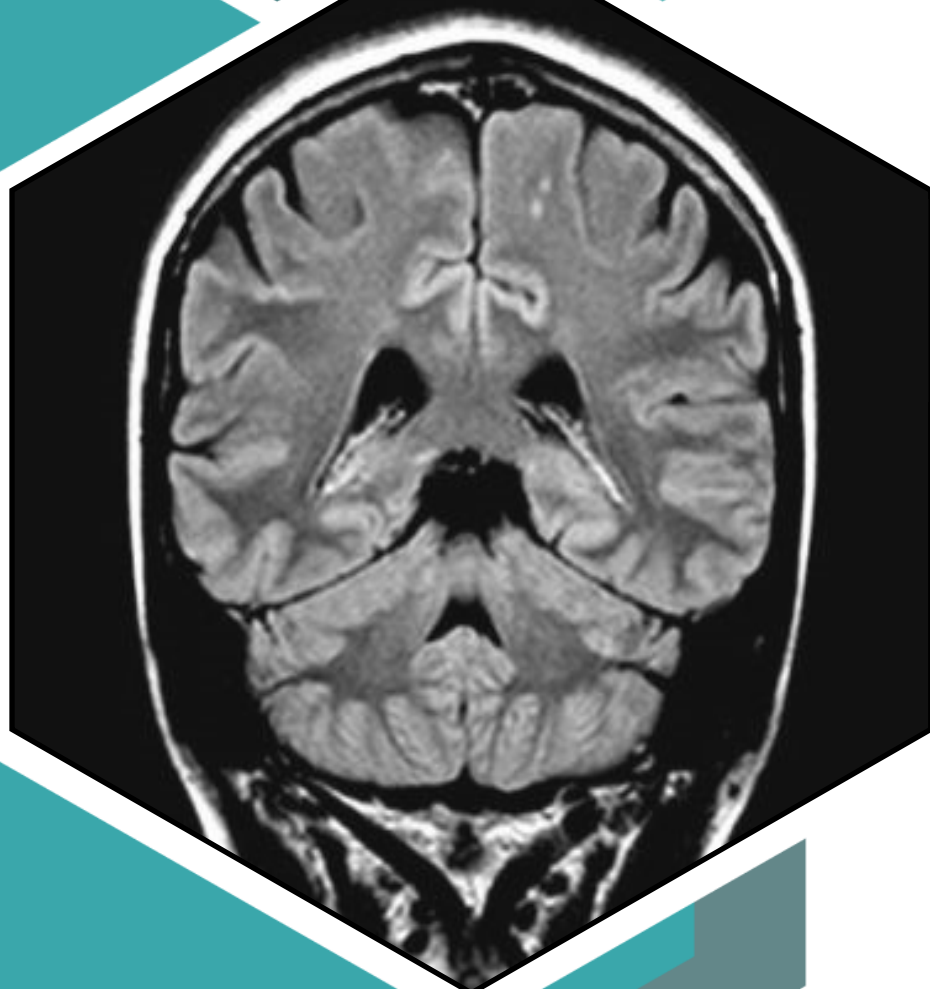
AUTOMNE 2019

PROJET DE SESSION



UNIVERSITÉ DE
SHERBROOKE

**QUANTIFICATION
DE TISSUS À
PARTIR D'IMAGES
D'IRM**



**IMN708- RECONSTRUCTION ET ANALYSE
D'IMAGES MÉDICALES
JULIEN BROSEAU**

Sommaire

Sommaire	1
Introduction	2
Revue des faits	3
L'imagerie par résonance magnétique	3
Problème actuel.....	3
Conception	4
Récupération des images obtenues par IRM.....	4
Débruitage des images.....	4
Segmentation des images	5
Quantification des images	5
Implémentation	7
Récupération des images obtenues par IRM.....	7
Débruitage des images.....	7
Segmentation des images.....	8
Quantification	11
Résultats	12
Conclusion	13
Bibliographie	14

Introduction

Lors de l'analyse des résultats après une IRM, le médecin/radiologue en charge du patient ne fait, généralement, qu'une analyse globale de la situation en observant les images. Rien ne permet de savoir précisément l'évolution d'une tumeur au sein d'un organisme.

Dans ce cadre, l'idée du projet était de quantifier un élément présent dans un organisme et dans mon cas, une tumeur. Plus précisément, la conception et l'implémentation se sont très vite portées sur la segmentation d'images. En effet, la segmentation est un domaine qui fait encore l'état de recherches puisque nous sommes toujours à la recherche d'algorithmes plus performants.

Par définition, une tumeur est une augmentation de volume d'un tissu sans précision de cause. C'est une néoformation de tissus corporels (ou néoplasie) qui se produit à la suite d'un dérèglement de la croissance cellulaire. Une tumeur peut être de type bénin ou malin. Dans le cadre d'une tumeur maligne, on parle de cancer.

De même, la segmentation d'images est une opération de traitement d'images qui a pour but de rassembler des pixels entre eux suivant des critères prédéfinis. Les pixels sont ainsi regroupés en régions, qui constituent un pavage ou une partition de l'image par exemple.

C'est donc dans ce cadre que je me suis mis à la recherche d'algorithmes de segmentation et ai cherché à comprendre leur fonctionnement. Pour ce faire, j'ai eu à ma disposition un ensemble d'images (T1, T2) provenant d'un patient.

Revue des faits

L'imagerie par résonance magnétique

L'imagerie par résonance magnétique (IRM) est un examen médical qui utilise un champ magnétique et des radiofréquences permettant de générer des images très précises, en 2D ou en 3D, des parties du corps et organes internes tels que le cerveau, la colonne, les articulations, l'abdomen, les seins et la prostate.

À l'aide d'un champ magnétique, d'ondes radio et d'ordinateurs qui génèrent des images des tissus internes, l'imagerie par résonance magnétique comporte un avantage par rapport à d'autres techniques d'imagerie. En effet, l'IRM permet aux médecins d'établir un diagnostic et de proposer des traitements précis sans faire appel à la radiation ou à du matériel radioactif.

Les examens d'IRM sont effectués lorsqu'un médecin soupçonne un problème médical difficilement détectable par un examen physique courant. Ils sont non invasifs, ce qui signifie qu'aucune procédure chirurgicale n'est nécessaire à la visualisation des systèmes du corps. L'IRM est aussi utilisée pour obtenir des données diagnostiques spécifiques que les autres technologies, comme l'échographie, la radiographie classique et la tomodensitométrie n'ont pas pu fournir.

Problème actuel

Aujourd'hui, de nombreux patients d'hôpitaux subissent régulièrement des examens pour diverses raisons et notamment une, le suivi de l'évolution d'une tumeur dans l'organisme.

Dans ce cadre, un patient est amené à réaliser, de façon périodique des examens IRM. A ce moment, des manipulateurs radio se chargent de faire l'acquisition des différentes images en fonction de ce qui est "recherché" chez le patient. Par la suite, ces images sont transférées à des radiologues. Ces derniers se chargent d'analyser les images en effectuant, entre autres, des mesures puis un compte rendu. Enfin, un médecin en charge du dossier se penche sur la question de l'évolution de la tumeur. Actuellement, le médecin ne fait qu'une constatation globale de la situation en observant les images récupérées, traitées et analysées lors de l'examen en constatant l'évolution mais sans savoir précisément son importance de manière quantitative. En effet, il est difficile, à l'œil nu de pouvoir déterminer le volume d'un tel organisme assez mobile et modulable.

Ayant une connaissance proche atteinte d'une tumeur bénigne et ayant déjà assisté à un rendez-vous avec le médecin en charge, il me semblait intéressant de me pencher sur la question d'un point de vue personnel mais aussi par curiosité. En effet, bien que le problème soit connu, il n'y a pas vraiment de système répandu, à ce jour, pour le résoudre.

Conception

Récupération des images obtenues par IRM

Avant d'entreprendre quoi que ce soit, il fallait s'assurer d'avoir les données permettant la réalisation du projet. Pour ce faire, j'ai récupéré des images obtenues par IRM d'un patient du CHU de Nantes (France). Comme nous le savons il y a des nombreux formats d'images, cependant nous avons travaillé essentiellement avec le format Nifti (.nii) durant la session.

Les images récupérées, quant à elles, étaient au format DICOM (.dcm). Ce format est de plus en plus répandu dans les hôpitaux. Néanmoins, il amène des inconvénients lorsqu'on s'intéresse au traitement des images. En effet, ce dernier n'a pas, à ce jour, de réelles normes qui lui sont propres. En d'autres termes les fichiers peuvent avoir le même format mais pas forcément la même structure ou architecture interne. Tout cela peut varier en fonction du constructeur de la machine utilisée. C'est pourquoi il est difficile de poser un code standard afin de traiter ce genre de fichier.

Dans le cadre du fichier, cela ne posait pas réellement de soucis, puisque je me suis concentré sur les images provenant d'un seul examen.

Le problème soulevé c'est plutôt tourné vers la conversion des images d'un format DICOM vers un format Nifti afin de traiter ces dernières. Pour ce faire, il y a plusieurs modules qui existe à ce jour afin de réaliser cette conversion. Pour ma part, je me suis tourné vers "dcm2nii".

Débruitage des images

Avant tout traitement, il faut traiter les images. Parmi les opérations courantes en imagerie, il est intéressant de procéder à un débruitage afin d'avoir une image la plus nette possible et, dans notre cas, permettre d'effectuer une segmentation plus facilement.

En effet, nous allons le voir plus en détail par la suite, mais la segmentation est basée, entre autres, sur la détection des contours. En d'autres termes, meilleurs sont les contours, meilleure sera la segmentation. Il donc important, d'un point de vue d'amélioration du système, de débruiter nos images afin qu'elle ait la meilleure qualité possible, notamment au niveau de la netteté et la prononciation des contours, afin d'obtenir de meilleurs résultats par la suite.

Segmentation des images

Le cœur du projet peut être considéré au niveau de la segmentation. C'est d'ailleurs sur ce point que je me suis attardé en testant plusieurs méthodes de segmentation, en les analysant, en les manipulant et en essayant de tirer le meilleur de chacune.

Pour rappel, l'idée est de pouvoir isoler un tissu, visible sur les images d'IRM, et de reconstituer et quantifier son volume. La segmentation, dans l'un de ces concepts, a pour but de rassembler des pixels entre eux suivant des critères prédéfinis.

Par analogie, si l'homme sait naturellement séparer des objets dans une image c'est grâce à des connaissances de haut niveau (compréhension des objets et de la scène). Néanmoins, mettre au point des algorithmes de segmentation du même niveau est encore un des thèmes de recherche les plus courants en traitement d'images.

La segmentation est une étape primordiale. À ce jour, il existe de nombreuses méthodes de segmentation. Pour ma part, dans le projet, deux concepts ressortent :

- Segmentation fondée sur les contours (méthodes "Inondation" et "Snake"). Succinctement, l'idée sous-jacente est de prendre un point dans l'image. Par la suite l'algorithme cherche à construire une aire autour de ce point par extension de proche en proche jusqu'au moment où il va être confronté à un certain seuil (souvent un contour) où il va s'arrêter.
- Segmentation fondée sur la classification des pixels en fonction de leur intensité (méthodes "Seuil", "K-Means" et "Mean Shift"). Ici, l'idée est de récupérer l'histogramme de l'image (qui donne la distribution des intensités de pixels de l'image). Ensuite, on regroupe par "paquet" les pixels semblables qui, pour "K-Means" se traduit par une courbe gaussienne. Ce regroupement est appelé "cluster".

Quantification des images

Après la segmentation il suffit de faire une quantification du volume récupéré. Pour ce faire il convient simplement de récupérer la taille d'un voxel. Ce dernier est un élément important en imagerie. En effet, il détermine un pixel en 3D (ayant donc une composante en Δx , en Δy et en Δz). Or, toutes les images traitées dans le cadre du projet et même dans la majeure partie des cas sont des images 3D. Il est donc important de définir un élément lié à ce type d'image.

Il faut savoir que dans un fichier Nifti, on peut récupérer toutes les informations liées à l'image (taille de l'image, taille des voxels, format de l'image, etc...). Nous avons donc en possession la taille des voxels. Ces données, que nous avons toujours eu au final, vont ainsi permettre d'effectuer plusieurs opérations dans le but de reconstituer l'image en 3D.

Il est important d'expliquer que les images utilisées pour ce projet ont des voxels anisotropes. Le concept d'un voxel anisotrope, ici, est d'avoir deux composantes égales (Δx et Δy) et une composante différente et plus grande (Δz).

Ce concept s'explique sur la base d'un souci physique et technique. En effet, lors d'une IRM, dans le but d'avoir des images nettes, il est nécessaire que le patient reste totalement immobile. Il est donc logique de chercher à réaliser une acquisition courte. C'est donc dans un souci de temps que les manipulateurs radio cherchent à optimiser ces acquisitions. On comprend alors pourquoi, si une seule coupe nous intéresse (axiale, coronale ou sagittale), on cherche à négliger une composante afin d'aller plus vite.

Dans le cadre d'un voxel anisotrope, il est plus simple de reconstruire une image 3D pour chaque coupe (axiale, coronale et sagittale). C'est pourquoi on cherche à récupérer la composante Δz afin de connaître la distance entre deux tranches. Par la suite, avec un lissage et une approximation entre les différentes tranches, on peut reconstituer l'image 3D, la quantifier et ainsi permettre un vrai suivi vis à vis de l'évolution du tissu.

Dans le cas de voxels anisotropes toujours, le but final, avant quantification, serait de combiner les trois modèles obtenus afin d'en concevoir un meilleur.

Implémentation

Récupération des images obtenues par IRM

Comme expliqué dans la section “conception”, les images d’origines étaient toutes au format DICOM. Il a donc fallu procéder à une conversion afin d’obtenir et de traiter par la suite des images au format Nifti.

Pour ce faire, j’ai tout simplement utilisé un convertisseur assez connu qui est “dcm2niix”. Le convertisseur est assez poussé puisque qu’il cherche à récupérer toutes les images DICOM issues d’une même acquisition (vis à vis du temps et de la coupe) et il nous transmet un fichier Nifti avec l’ensemble des images d’une coupe (axiale, sagittale ou coronale) ainsi un fichier JSON avec les détails.

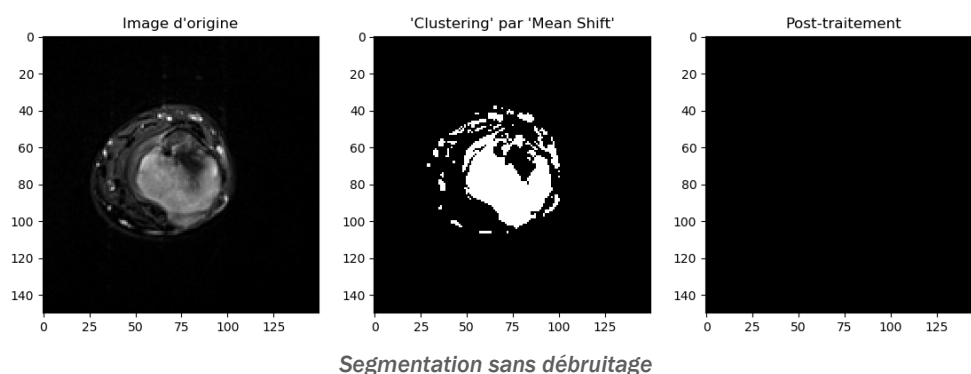
Débruitage des images

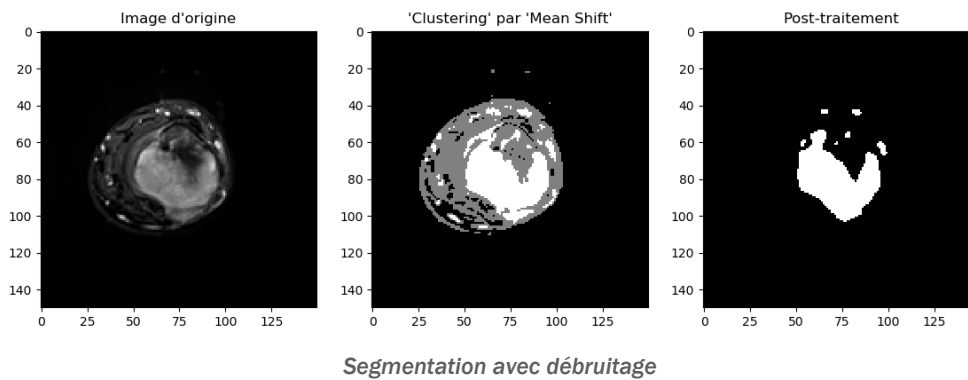
Une fois les images au bon format, il m’a suffi de suivre, dans les grosses lignes, l’ensemble des procédés vus durant les TP et plus précisément pour cette section, le débruitage.

Pour rappel, il m’a semblé essentiel de débruiter les images afin d’améliorer la qualité de ces dernières au niveau des contours notamment puisqu’une partie des algorithmes de segmentation que j’ai utilisé étaient basés sur la détection de contour. D’un point de vue de performance, il était donc logique de passer cette étape.

Pour ce faire, je me suis basé sur les implémentations réalisées lors du premier TP. En effet, une partie importante de ce dernier consistait au débruitage de diverses images à l’aide de plusieurs algorithmes. Je me suis penché, tout particulièrement, vers le “Non Local Means”.

Sans rentrer dans l’ensemble des détails concernant son fonctionnement, si on s’intéresse simplement aux résultats qu’il apporte, nous nous étions rendus compte qu’il était particulièrement performant pour les images d’IRM. Il m’a donc semblé logique de récupérer ce dernier, d’autant plus que son paramétrage est assez simple. Dans mon cas, il suffisait simplement de connaître le type de bruit (ricien ou gaussien) auquel les images étaient confrontées.



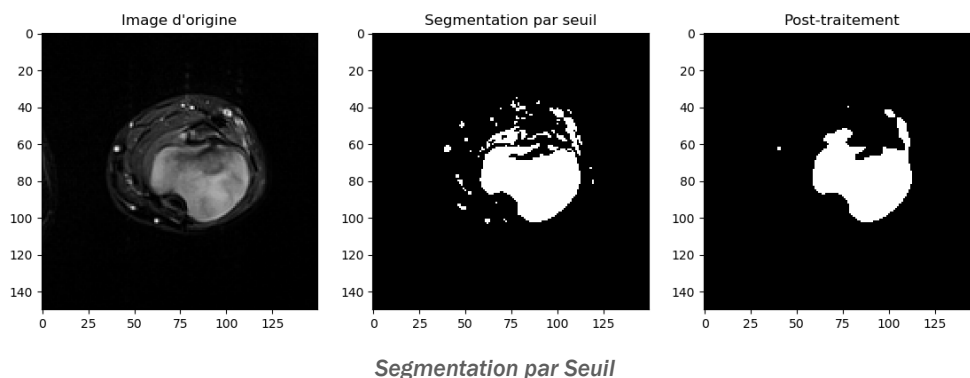


Segmentation des images

Méthode – Seuil

La méthode de seuil est la plus simple autant au niveau de la conception que de l'implémentation. Il s'agit d'une segmentation binaire. Soit l'intensité du pixel est au-dessus du seuil, soit elle est en dessous. Concernant le seuil, il est défini à chaque fois par l'utilisateur.

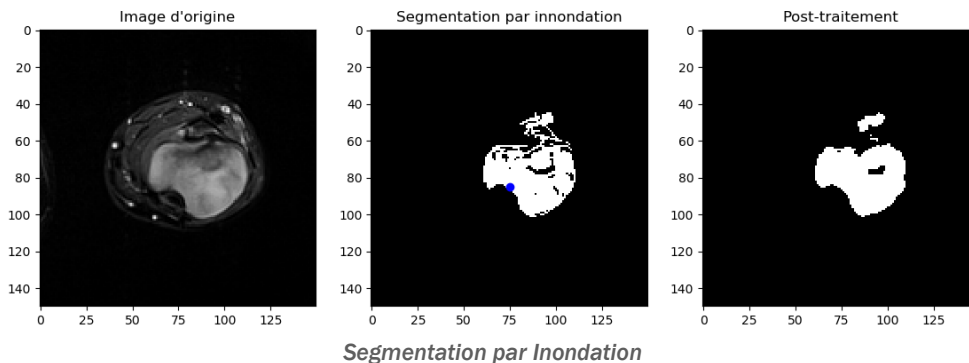
Dans mon cas, je me suis simplement basé sur l'histogramme des images. La méthode ne proposant qu'une classification à deux classes, il était impossible de bien isoler la tumeur. Cette implémentation relevait plus d'une mise en contexte sur la question de la segmentation. J'ai donc très vite abandonné ce cas.



Méthode - Inondation

La méthode d'inondation repose sur le concept de diffusion. En effet, le principe est de partir d'un pixel défini par l'utilisateur. Ensuite, la propagation rentre en jeu. De proche en proche, le but est de comparer les intensités de pixels en deux voisins. Si ces intensités sont semblables, soit, inférieures à un seuil défini, lui aussi, par l'utilisateur, le pixel voisin est considéré comme un élément de notre objet et va réitérer ce concept sur ses voisins.

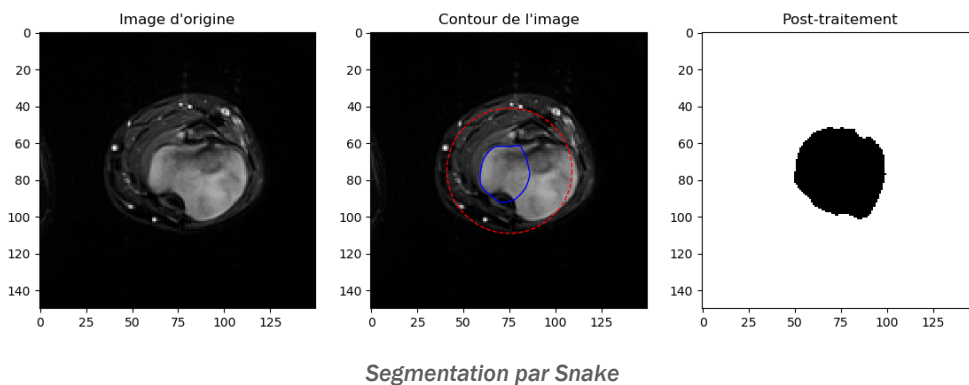
La diffusion s'arrête lorsque la différence d'intensité entre deux pixels voisins est trop grande. A ce moment le pixel voisin est considéré en dehors de notre objet, c'est concept de segmentation basée sur la détection de contours.



Méthode - Snake

La méthode snake, reprend exactement les concepts d'inondation. On pourrait dire que ce sont les mêmes, au point près, que pour la méthode inondation, j'ai eu recours à la bibliothèque "skimage" et globalement simplement utilisé la fonction "flood" afin d'obtenir une segmentation.

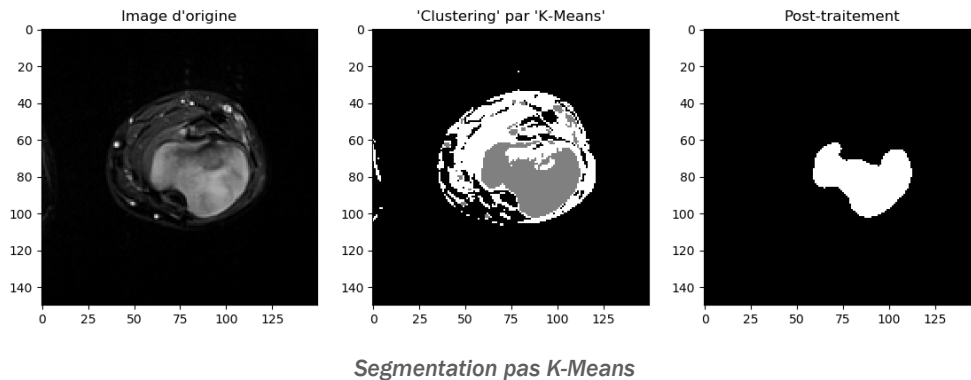
Concernant la méthode snake, je me suis permis de rentrer plus dans les détails et ne pas se limiter dans l'utilisation d'une seule fonction. Le but principal, par dernière, était d'avoir la possibilité de les manipuler avec plus de paramètres afin d'optimiser les résultats. Néanmoins, cela n'a pas été réellement convainquant.



Méthode - K-Means

La méthode K-Means repose sur le principe de classification par intensité de pixels. Le principe est d'initialiser au hasard les centres de classes. Par la suite, on associe l'ensemble des pixels au centre le plus près puis on déplace les centres sur la moyenne des données. On recommence ce processus jusqu'au moment où l'algorithme converge.

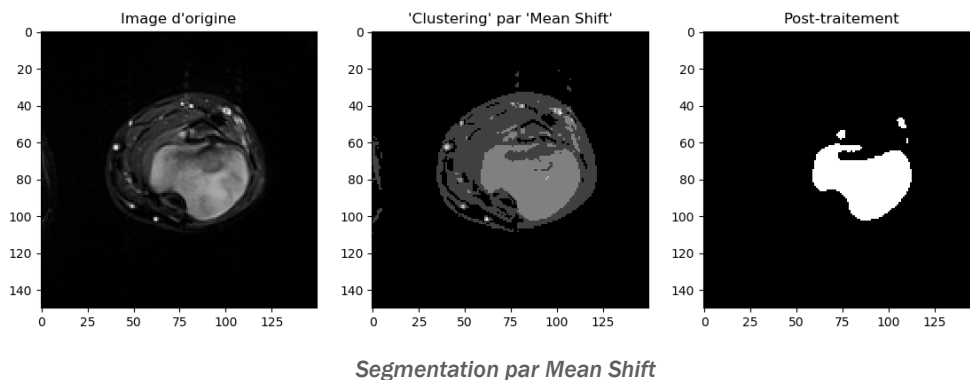
Du point de vue de l'implémentation, il a simplement suffi récupérer les fonctions associées à la méthode dans la bibliothèque "sklearn". Ensuite, il a fallu ajuster avec les paramètres afin d'obtenir la meilleure solution. Que ce soit en modifiant le nombre de clusters ou l'initialisation des centres de classes, les résultats obtenus ne convenaient pas entièrement lorsqu'on appliquait la configuration à l'ensemble des images. En effet, certaines segmentations mettaient à la lumière bien plus que la tumeur et d'autres simplement un bout de la tumeur.



Méthode - Mean Shift

La méthode Mean Shift peut être apparentée à l'amélioration de la méthode K-Means. En effet, cette méthode présente les mêmes fondements que précédemment mais avec quelques modifications tout de même. Le principe est donc d'initialiser un point au hasard. Ensuite, on cherche à calculer la densité locale, soit, selon un périmètre défini par l'utilisateur, soit on déplace le point en direction du gradient de densité maximum. Enfin, on répète cette opération jusqu'au moment où l'algorithme converge.

Concernant l'implémentation, il a aussi fallu récupérer les fonctions associées à la méthode dans la bibliothèque "sklearn". Par la suite j'ai simplement ajusté la largeur du périmètre afin d'obtenir les meilleurs résultats. De toutes les méthodes, c'est avec celle-ci que j'ai réussi, au mieux, à segmenter la tumeur sur chaque image.



Quantification

Une fois la segmentation effectuée, quelle qu'elle soit, il s'agissait de pouvoir mesurer la taille des différentes segmentations d'images. Pour rappel, les images récupérées sont constituées de voxels anisotropes. En d'autres termes, dans notre cas, il s'agit du fait que les composantes Δx et Δy sont de la même dimension mais la composante Δz , quant à elle, est bien plus grande.

Ainsi, cette section s'est divisée en deux étapes. La première consistait à récupérer l'aire de chaque tranche ayant été segmentée préalablement. La deuxième s'inscrivait dans le but de relier toutes ces tranches et donc de calculer le volume final de la structure nous intéressant.

Concernant le calcul d'aire, le principe sous-jacent était assez simple. L'idée était, au préalable, de récupérer les dimensions des voxels. Concrètement, la taille d'un voxel était de (0.76, 0.76, 6.5). Par la suite, il a juste fallu récupérer le nombre de pixel appartenant à la segmentation, c'est à dire, le nombre de pixels blancs ou plutôt, vis à vis de l'implémentation, le nombre de pixels différents de l'intensité zéro (synonyme du noir). Enfin, il m'a juste fallu effectuer une simple multiplication entre le nombre de pixels obtenus et l'aire d'un pixel, à savoir, 0.76×0.76 .

Par la suite, l'idée était donc de relier toutes ces aires. Pour ce faire, j'ai multiplié chaque aire par l'écart entre deux tranches. Le concept sous-jacent était d'implémenter une sorte de bloc ayant la forme d'une aire et enfin additionner tous ces blocs afin de reconstituer un volume.

```
Aire de la segmentation : 65.92 mm², soit : 0.66 cm².
Aire de la segmentation : 459.59 mm², soit : 4.60 cm².
Aire de la segmentation : 640.87 mm², soit : 6.41 cm².
Aire de la segmentation : 762.94 mm², soit : 7.63 cm².
Aire de la segmentation : 829.47 mm², soit : 8.29 cm².
Aire de la segmentation : 881.35 mm², soit : 8.81 cm².
Aire de la segmentation : 899.66 mm², soit : 9.00 cm².
Aire de la segmentation : 913.09 mm², soit : 9.13 cm².
Aire de la segmentation : 921.02 mm², soit : 9.21 cm².
Aire de la segmentation : 900.88 mm², soit : 9.01 cm².
Aire de la segmentation : 916.75 mm², soit : 9.17 cm².
Aire de la segmentation : 932.01 mm², soit : 9.32 cm².
Aire de la segmentation : 776.98 mm², soit : 7.77 cm².
Aire de la segmentation : 703.74 mm², soit : 7.04 cm².
Aire de la segmentation : 11.60 mm², soit : 0.12 cm².
Aire de la segmentation : 68.36 mm², soit : 0.68 cm².
Volume de la segmentation : 69447.33 mm³, soit : 69.45 cm³.
```

Quantification - Coupe axiale

```
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 393.75 mm², soit : 3.94 cm².
Aire de la segmentation : 650.39 mm², soit : 6.50 cm².
Aire de la segmentation : 2212.21 mm², soit : 22.12 cm².
Aire de la segmentation : 3670.31 mm², soit : 36.70 cm².
Aire de la segmentation : 3575.39 mm², soit : 35.75 cm².
Aire de la segmentation : 3084.96 mm², soit : 30.85 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Volume de la segmentation : 70312.79 mm³, soit : 70.31 cm³.
```

Quantification - Coupe coronale

```
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 55.39 mm², soit : 0.55 cm².
Aire de la segmentation : 1245.15 mm², soit : 12.45 cm².
Aire de la segmentation : 2049.41 mm², soit : 20.49 cm².
Aire de la segmentation : 2669.77 mm², soit : 26.70 cm².
Aire de la segmentation : 1262.14 mm², soit : 12.62 cm².
Aire de la segmentation : 3740.63 mm², soit : 37.41 cm².
Aire de la segmentation : 2652.04 mm², soit : 26.52 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Aire de la segmentation : 1979.25 mm², soit : 19.79 cm².
Aire de la segmentation : 412.84 mm², soit : 4.13 cm².
Aire de la segmentation : 1353.72 mm², soit : 13.54 cm².
Aire de la segmentation : 78.28 mm², soit : 0.78 cm².
Aire de la segmentation : 0.00 mm², soit : 0.00 cm².
Volume de la segmentation : 73494.20 mm³, soit : 73.49 cm³.
```

Quantification - Coupe sagittale

Résultats

Le projet a dévié petit à petit vers un concept de segmentation et comment concevoir un algorithme viable pour les différentes images à traiter.

Les algorithmes que j'ai utilisés sont globalement naïfs ou trop simplistes et ne marchent parfaitement que pour du cas par cas. Dès lors qu'on s'intéresse à segmenter plusieurs images avec la même configuration, on se rend vite compte qu'il n'y a rien de très efficace.

En effet, si on reprend le cas de "Mean Shift", la force de l'algorithme, par rapport à ceux que j'ai pu tester, est le fait de ne pas avoir besoin de définir un nombre de clusters. Tout passe simplement par la largeur de la bande d'intensité. Ainsi, les clusters sont définis en fonction de l'intensité des pixels de l'image. Cette force, en un sens, est aussi une faiblesse. En effet, d'une image à l'autre, on ne peut pas savoir le nombre de clusters définis et donc l'étiquette qui est attribué au(x) cluster(s) qui nous intéresse(nt). On ne peut donc pas prédéfinir une ou plusieurs étiquettes de clusters à récupérer.

D'un autre point de vue, si on prend le cas de "K-Means", on ne peut pas non plus définir un nombre de clusters par défaut, valable pour toutes les images. En effet, chaque image est différente. Elles ont toutes leurs intensités et leurs contrastes. Ainsi, si on prend le cas d'une tumeur, cette dernière n'a pas nécessairement la même plage d'intensité entre deux images. Cette plage est semblable certes, mais pas assez pour être sûr de pouvoir récupérer la totalité du tissu sans "déborder".

Le lien entre les deux points précédents est le concept du machine learning. Il est de plus en plus répandu, en traitement d'image, de faire appel à des techniques d'apprentissage car elles ont l'avantage d'entraîner un modèle (de segmentation dans notre cas) afin que ce dernier puisse être le plus performant possible par la suite (dans le cadre du projet, isoler une tumeur à la perfection sur n'importe quelle image). Cependant, là encore, il y a des inconvénients dans le concept le plus naïf. Les images obtenues par IRM ont plusieurs types (T1, T2, etc..). Les images provenant d'un cerveau, d'un bras ou d'une jambe ne se ressemblent en aucun point, les tumeurs ne sont pas "modélées" de la même façon par exemple. Par ces quelques exemples, on se rend compte qu'on ne peut pas entraîner notre modèle avec n'importe quelles données au risque d'obtenir des résultats catastrophiques.

De plus, si on suppose réussir à isoler ce qu'on pense être la tumeur, en réalité, il faudrait avoir un expert à côté de nous afin qu'il puisse juger la segmentation. En effet, il se peut qu'on soit amené à récupérer des tissus nécrosés par exemple, chose qui ne fait pas parti des éléments qui nous intéressent.

Enfin, l'idéal serait de combiner tous ces points et d'en tirer le meilleur. En d'autres termes, il faudrait un algorithme basé sur le machine learning (afin de l'entraîner et qu'il puisse être plus performant), ayant une combinaison entre la segmentation par contours et par classification des intensités de pixels, et enfin, en ayant un radiologue qui puisse superviser cette apprentissage.

Conclusion

Ce projet m'a permis de m'ouvrir sur le domaine médical. C'est sans doute l'un des domaines qui me tient le plus à cœur mais dans lequel je n'avais jamais eu de cours afin de l'étudier. Cela m'a permis de découvrir des concepts biologiques, des implémentations, des algorithmes, des questions ouvertes et les recherches liées au domaine. Pour en venir au projet, je me suis rendu compte que l'implémentation d'une segmentation et d'une quantification d'images médicales, notamment des tumeurs est loin d'être un concept trivial, parce qu'il est lié à beaucoup de problèmes d'implémentations.

C'est d'ailleurs en se lançant dans ce type de projet, que l'on est souvent amené à se demander "pourquoi ça n'existe pas ?" Cependant, en essayant d'amener une solution viable, on comprend très vite pourquoi.

Néanmoins, j'ai réussi à faire une ébauche d'implémentation permettant de calculer, avec une bonne marge d'erreur le volume d'une tumeur. J'ai pu aussi constater qu'il était vital de mélanger plusieurs algorithmes, plusieurs méthodes d'approches et avec l'aide de techniques d'apprentissage et d'experts, comme des radiologues, en imagerie, nous pourrions arriver à un résultat intéressant pour une première version.

Bibliographie

Documentation

- Explications sur les IRM

<https://irmquebec.com/lirm/quest-ce-quune-irm/>

- Explications de la segmentations

https://fr.wikipedia.org/wiki/Segmentation_d%27image

- Explications sur la tumeur

<https://fr.wikipedia.org/wiki/Tumeur>

Filtrage

- Moyenne non local

<https://dipy.org/documentation/1.0.0./reference/dipy.denoise/#dipy.denoise.nlmeans.nlmeans>

- Médian

https://docs.scipy.org/doc/scipy/reference/generated/scipy.ndimage.median_filter.html

Clustering

- K-Means

<https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html>

- Mean Shift

<https://scikit-learn.org/stable/modules/generated/sklearn.cluster.MeanShift.html>

<http://efavdb.com/mean-shift/>

- Snake

<https://towardsdatascience.com/image-segmentation-using-pythons-scikit-image-module-533a61ecc980>