Universitat de Girona

MAIA

UNIVERSITAT DE GIRONA, MEDICAL IMAGE SEGMENTATION AND APPLICATIONS

Report: Atlas based segmentation | 2021-2022

OJEDA, Manuel; VASILE, Alexandru

November 21, 2021

1 INTRODUCTION

The image can be segmented with the atlas-based segmentation when there is no well defined relation between regions and pixels' intensities. Due to lack of the border or excessive noise or in the case when the objects of the same texture need to be segmented. If the information about difference between these object is incorporated in spatial relationship between them, other objects, or within their morphometric characteristics, the atlas-based segmentation is expected to work well. One more advantage of atlases is in their use in clinical practice, for computer aided diagnosis whereas they are often used to measure the shape of an object or detect morphological differences between patient groups.

2 PROBABILISTIC ATLAS

2.1 Introduction and problem definition

The primary goal of this part is to build a probabilistic atlas from a set of brain volumes with the available labels of three classes (WM, GM and CSF). For performing the registration, the tool of choice was elastix, an open source toolbox for rigid and nonrigid registration of images. The final result will be a probabilistic atlas: an intensity volume (used for registering new unsegmented volumes) and a probabilistic label volume (containing tissue probabilities at each voxel).

2.2 Algorithm analysis

- **Choose parameters to be used for elastix.**
- **ELASTIX:**
  - Register training volumes.
  - Transform training labels.
- **Compute and save the Probabilistic Atlas.**

2.3 Proposed solution

Several tests were run with a fixed image and a moving image to decide the best possible parameters to use with elastix for registration and transformation. Researching how Elastix works, it was known is possible to use more than one parameter file at the same time to get a better registration. The parameters files to run the tests were obtained from the Elastix Model Zoo GitHub repository which contains different type of registrations.
The final decision of the parameters to use are *affine and bspline transformation*, in two different parameters files. These two files contain each one different parameters which among the most important components are:

- *Type of interpolator:* for both transformations is used "BSplineInterpolator".
- *Metric:* for both transformations is used "AdvancedMattesMutualInformation".
- *Optimizer:* for the affine transformation is used "AdaptiveStochasticGradientDescent" and for the bspline transformations is used "StandardGradientDescent".
- *Number of resolutions:* for the affine transformation is set to 4 and for the bspline transformations is set to 5.
- *Maximum number of iterations for each resolution level:* for the affine transformation is set to 500 and for the bspline transformations is set to 2000.
- *How to combine transforms:* for both transformations is used "Compose".

After running the initial tests, it was observed the need of having a structure for all the folders to save all the results files according to every step of the pipeline: for registration, for transformation, to images, to labels, to MNI atlas.
The solution proposed considers using the file 1000 as the fixed image and all the other images as the moving image. The registration is done using elastix commands with the parameters mentioned before, saving all the files generated, so they can be used afterwards. The registration is generated using one single batch file, with the same parameters, the only difference is the patient image considered as moving image.
After the registration, we perform a transformation on the labels images using the transformation parameters generated from the registration, this is accomplished by using "Transformix", a tool from elastix. As before, all the results were saved with an organized structure to be used after, if needed. The steps mentioned were performed on the training set and also on the testing set; for both of the sets, their results files are saved in their corresponding folder so they can be used after.

The next steps are realized in a python script. To compute the probabilistic atlas, the files used are the labels transformations from the training set which are read and treated them as an array, easier to handle. All the transformations are managed one at the time, where each one of the classes (CSF, GM, WM) are saved in its corresponding variable along with their own labels with their original value. To accomplish this step, a copy of the label is generated, and per each class everything different to its value is set to zero and concatenated to a list.
The next step corresponds to get the average for each one of those lists and divide it by the maximum value so it is normalized. At the end of the algorithm, depending on the user, either the probabilistic atlas per each one of the classes can be returned or can be saved as a NIFTI file to be used in the future.

2.3.1 Proposed solution for MNI atlas

For the MNI atlas, the approach is different since the atlas is already done. The atlas was treated as an array, each one of the classes was divided by its maximum value. The result is either saved as a NIFTI file to be used in the future or it can be returned, depending on the user, in order to get the probabilistic atlas for all the three different classes.

3 PROBABILISTIC ATLAS INTO THE EM ALGORITHM.

3.1 Introduction and problem definition

The goal of this part is to integrate the use of a probabilistic atlas into the EM algorithm. As done in the previous assignment, the application will be the brain tissue segmentation task, where an atlas with the three probability maps per each tissue (WM, GM and CSF) will be available.

3.2 Algorithm analysis

- **Obtain the Probabilistic Atlas.**
- **Initialization:**
  - **K-Means initialization:**
    - Compute k-means clustering.
    - Obtain initial values for expectation and maximization per each class.
  - **Atlas initialization (both atlases):**
    - Read the result of the labels transformations obtained with elastix per each class.
    - Combined probabilities for all the classes to assign the maximum probability.
  - **Random:** the algorithm generates randomly a vector of 1, 2 or 3, which represents the classes: CSF, GM, WM. The length of the vector is the same as the length of the vectorized volume. This instance was only used as a debugging tool.
  - **Tissue models**
  - **Label propagation**
  - **Tissue models and label propagation**
- **EM:**
  - **EXPECTATION STEP:** Recompute labels for all the data set given the current cluster parameters.
  - **MAXIMIZATION STEP:** Use that classification to estimate the parameters again.
  - **Until CONVERGES to a local minimum:** it is checked if the parameters do not have any significant changes, if they do not, it stops.
- **Segmentation with atlas:**
  - Acquire as an array the probabilistic atlas per each one of the three classes. This step is either for the training atlas or the MNI atlas.
  - Calculate the intensity probability volume using the Tissue Models.
  - A predicted mask is obtained for each one of the classes based on the weights obtained from the convergence.
  - Combine the three previous masks for all the classes.
  - Assign the final label based on the maximum probability from the combined mask.

3.3 Proposed solution

The solution proposed begins with the registration of both atlases, MNI and training probabilistic atlas, to each one of the cases in the test set. This step is accomplished using elastix and transformix, with an organized structure for the folders so all the files can be saved and, if needed, used in the future. The process was accelerated by using batch files with all the necessary commands, which were reviewed to avoid problems when the files were saved or the file was ran. Everything was treated as an array, so the data is more manageable for all the needed operations. The probabilistic atlas can be loaded, since it was already saved from the first part, or can be generated at this step.

In case of the K-Means initialization, as in the previous assignment, the initial values for expectation and maximiziation are obtained from the result once the K-Means clustering process is applied. Where the mean and covariance for all the three classes is obtained, so the algorithm can iterate until it reaches convergence, everything from the centroids from the clusters generated from the K-Means algorithm.

In case of atlas initialization, the labels propagation is obtained with the probabilistic atlas and tissue models and it can be obtained using the training atlas or the MNI atlas. From that step, a vector per each one of the classes is obtained with the purpose of getting the initial values of the mean and covariance, for the EM algorithm to run until it reaches convergence.
For the tissue model initialization, the models are load in order to obtain three different vectors and the vectorized representation of the whole volume. From those vectors, the initial mean, covariance and prior probabilistic are computed for each one of the three classes (CSF, WM, GM). Those initial values are submit to the EM algorithm for it to converges.
For the label propagation initialization, the masks are already saved in their corresponding folders. To each one of the classes is computed its centroids, where it begins in order to compute the mean, the covariance and the prior probability for each one of the classes.
For the tissue model and label propagation initialization, both results from the previous two steps are multiplied and from those the initial values are obtained and submitted to the EM algorithm.

For the reconstruction of the result, it can be implemented by the only EM result or it can be used the probabilistic atlas from the training or the MNI.
In the first case, the predicted mask is reshape based on the weights obtained from the convergence, with the decision from the user if to return the final prediction or to save it.
In case of using the probabilistic atlases, we acquire as an array the corresponding labels transformed for all three classes, depending of the set up it can be from the training atlas or the MNI atlas. The next step is to calculate the intensity probability volume, which for this, the tissue models generated from the beginning are used to get the corresponding intensity probability. Using the weights, as before, the predicted masks are reshape and each one of the masks is combined with the atlas and the intensity from the tissue models. Before getting the final result, the final label is assigned by obtaining the maximum argument from the combination of the previous masks.

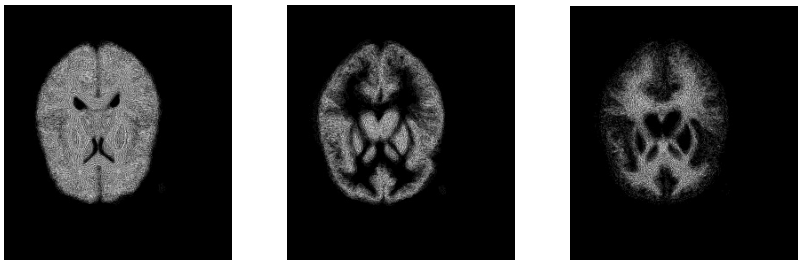
4 Experimental section and results

The experiments on the algorithm were realized in different ways and will be explained afterwards:

- **Segmentation without EM:**
  - **Tissue models:** segmentation using just intensity information.
  - **Label propagation:** segmentation using just position information.
  - **Tissue models and label propagation:** multiplying the two previous results: segmentation using intensity and position information.
- **Segmentation with EM:**
  - **Using K-Means initialization:** segmentation using intensity information.
  - **Using tissue models initialization:** segmentation using intensity information.
  - **Label propagation initialization:** segmentation using just position information.
- **Addition of atlas:**
  - **Best previous initialization.**
  - **Tissue models and label propagation initialization.**
- **Addition of MNI atlas into EM:**
  - **Choose the initialization which looks more feasible.**

4.1 Probabilistic Atlas

The figure 4.1 shows the results obtained from the probabilistic atlas algorithm implemented on the training set at slice 135.



(a) CSF. (b) Gray matter. (c) White matter.

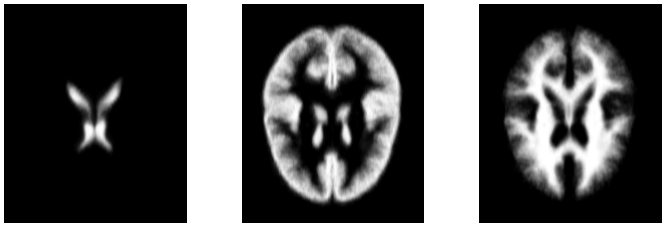
Figure 4.1: Probabilistic atlas.

4.2 Probabilistic MNI Atlas

The figure 4.2 shows the results obtained from the probabilistic atlas algorithm implemented on the MNI atlas at slice 91.

The figure 4.3 represents the blue cross-hairs mark the same voxel across all 3 volumes at slice 91 using the ITK Snap tool.

The figure 4.4 recovered from ITK Snap represents the final result from the registration using elastix. The fixed image, 1000, can be visualized; the moving image, 1008, and the result obtained from the registration and transformation. It can be observed for every one of the screens



(a) CSF. (b) Gray matter. (c) White matter.

Figure 4.2: Probabilistic MNI atlas.

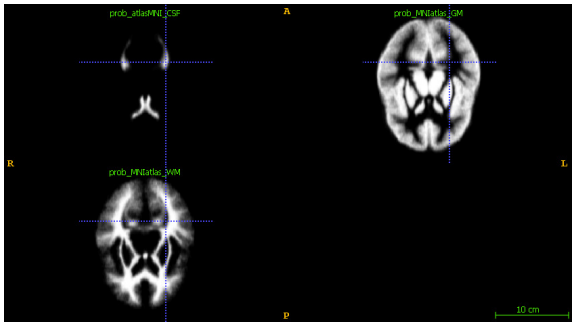


Figure 4.3: Use of ITK snap tool.

the number of the slice where the blue cross-hairs match. Another approach has also been pursued, in which the fixed image was an average of all the images in the trainining dataset, but the results were not so promising as to continue with that over the current method.

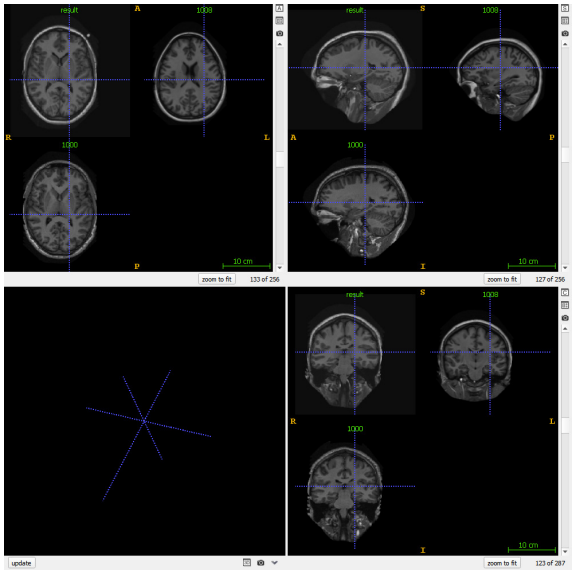


Figure 4.4: Registration results.

4.3 Tissue models

In the Figure 4.5, the tissue model can be visualized obtained for each one of the classes, where the intensity values above 2000 is due to division by very small small probabilities values.

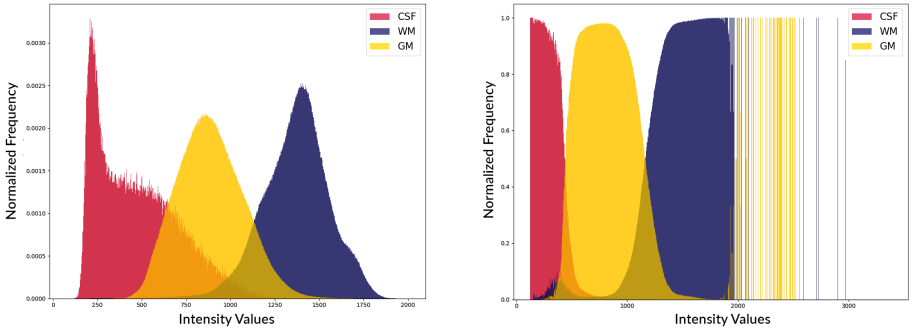


Figure 4.5: Tissue models.

4.4 Segmentation without EM

Table 4.1: Results of segmentation without EM for CSF class.

Dice coefficient and computational time (seconds)			
Patient	Tissue models	Labels propag	Tissue models & Labels propag
1003	0.0703/456.10	0.1622/456.10	0.1792/456.10
1004	0.0748/422.73	0.0574/422.73	0.0725/422.73
1005	0.0781/422.73	0.0357/422.73	0.0560/422.73
1018	0.1106/422.73	0.0162/422.73	0.0983/422.73
1019	0.0897/422.73	0.0587/422.73	0.0852/422.73
1023	0.1322/422.73	0.1156/422.73	0.1115/422.73
1024	0.1334/422.73	0.1100/422.73	0.1854/422.73
1025	0.0557/422.73	0.1286/422.73	0.1910/422.73
1038	0.1664/422.73	0.0566/422.73	0.0686/422.73
1039	0.1034/422.73	0.2704/422.73	0.1653/422.73
1101	0.2195/422.73	0.2394/422.73	0.1085/422.73
1104	0.1532/422.73	0.0750/422.73	0.0395/422.73
1107	0.1523/422.73	0.1077/422.73	0.2251/422.73
1110	0.1306/422.73	0.1258/422.73	0.2142/422.73
1113	0.2042/422.73	0.0353/422.73	0.0865/422.73
1116	0.0794/422.73	0.1307/422.73	0.1256/422.73
1119	0.1475/422.73	0.0216/422.73	0.0728/422.73
1122	0.0576/422.73	0.0633/422.73	0.0840/422.73
1125	0.1105/422.73	0.1246/422.73	0.0709/422.73
1128	0.0979/422.73	0.1340/422.73	0.1329/422.73

Table 4.2: Results of segmentation without EM for GM class.

Dice coefficient and computational time (seconds)			
Patient	Tissue models	Labels propag	Tissue models & Labels propag
1003	0.0641/456.10	0.1188/456.10	0.1257/456.10
1004	0.0659/422.73	0.0769/422.73	0.0830/422.73
1005	0.0672/422.73	0.0683/422.73	0.0764/422.73
1018	0.0802/422.73	0.0605/422.73	0.0933/422.73
1019	0.0718/422.73	0.0775/422.73	0.0880/422.73
1023	0.0888/422.73	0.1002/422.73	0.0986/422.73
1024	0.0893/422.73	0.0980/422.73	0.1281/422.73
1025	0.0583/422.73	0.1054/422.73	0.1304/422.73
1038	0.1025/422.73	0.0766/422.73	0.0814/422.73
1039	0.0773/422.73	0.1621/422.73	0.1201/422.73
1101	0.1238/422.73	0.1497/422.73	0.0974/422.73
1104	0.0972/422.73	0.0840/422.73	0.0698/422.73
1107	0.0969/422.73	0.0970/422.73	0.1440/422.73
1110	0.0882/422.73	0.1043/422.73	0.1396/422.73
1113	0.1176/422.73	0.0681/422.73	0.0886/422.73
1116	0.0677/422.73	0.1063/422.73	0.1042/422.73
1119	0.0950/422.73	0.0626/422.73	0.0831/422.73
1122	0.0590/422.73	0.0793/422.73	0.0876/422.73
1125	0.0802/422.73	0.1038/422.73	0.0823/422.73
1128	0.0751/422.73	0.1076/422.73	0.1071/422.73

Table 4.3: Results of segmentation without EM for WM class.

Dice coefficient and computational time (seconds)			
Patient	Tissue models	Labels propag	Tissue models & Labels propag
1003	0.8058/456.10	0.8175/456.10	0.8257/456.10
1004	0.8170/422.73	0.8371/422.73	0.8025/422.73
1005	0.8291/422.73	0.8172/422.73	0.8338/422.73
1018	0.7885/422.73	0.8334/422.73	0.8073/422.73
1019	0.8171/422.73	0.8234/422.73	0.8174/422.73
1023	0.8146/422.73	0.8110/422.73	0.8103/422.73
1024	0.8176/422.73	0.8411/422.73	0.8079/422.73
1025	0.8188/422.73	0.8171/422.73	0.8466/422.73
1038	0.8180/422.73	0.8224/422.73	0.7952/422.73
1039	0.8005/422.73	0.8138/422.73	0.8207/422.73
1101	0.8087/422.73	0.8363/422.73	0.8159/422.73
1104	0.8036/422.73	0.8194/422.73	0.8085/422.73
1107	0.7956/422.73	0.8016/422.73	0.7967/422.73
1110	0.8165/422.73	0.8122/422.73	0.8208/422.73
1113	0.8198/422.73	0.8283/422.73	0.8271/422.73
1116	0.7985/422.73	0.8195/422.73	0.8282/422.73
1119	0.8138/422.73	0.8255/422.73	0.8040/422.73
1122	0.8142/422.73	0.8159/422.73	0.8149/422.73
1125	0.8310/422.73	0.8153/422.73	0.8455/422.73
1128	0.8080/422.73	0.8241/422.73	0.8031/422.73

4.5 Segmentation with EM

Table 4.4: Results of segmentation with EM for CSF class.

Dice coefficient and computational time (seconds)			
Patient	K-Means	Tissue models	Labels propag
1003	0.0411/456.10	0.0847/456.10	0.1246/456.10
1004	0.1998/422.73	0.1808/422.73	0.0976/422.73
1005	0.1119/422.73	0.1483/422.73	0.2119/422.73
1018	0.1162/422.73	0.1283/422.73	0.2562/422.73
1019	0.1113/422.73	0.2088/422.73	0.1156/422.73
1023	0.2088/422.73	0.1623/422.73	0.0709/422.73
1024	0.1374/422.73	0.0259/422.73	0.1715/422.73
1025	0.0939/422.73	0.0973/422.73	0.3325/422.73
1038	0.2571/422.73	0.2614/422.73	0.1939/422.73
1039	0.0890/422.73	0.0976/422.73	0.1685/422.73
1101	0.0952/422.73	0.0772/422.73	0.0500/422.73
1104	0.0621/422.73	0.0605/422.73	0.0800/422.73
1107	0.1801/422.73	0.1640/422.73	0.1755/422.73
1110	0.2434/422.73	0.0710/422.73	0.3280/422.73
1113	0.1580/422.73	0.1607/422.73	0.0650/422.73
1116	0.2642/422.73	0.1534/422.73	0.1082/422.73
1119	0.1552/422.73	0.0407/422.73	0.2319/422.73
1122	0.1338/422.73	0.0606/422.73	0.1541/422.73
1125	0.1273/422.73	0.1765/422.73	0.1164/422.73
1128	0.0920/422.73	0.0500/422.73	0.1373/422.73

Table 4.5: Results of segmentation with EM for GM class.

Dice coefficient and computational time (seconds)			
Patient	K-Means	Tissue models	Labels propag
1003	0.0824/456.10	0.0938/456.10	0.1258/456.10
1004	0.1459/422.73	0.1323/422.73	0.1150/422.73
1005	0.1107/422.73	0.1193/422.73	0.1607/422.73
1018	0.1124/422.73	0.1113/422.73	0.1785/422.73
1019	0.1105/422.73	0.1435/422.73	0.1222/422.73
1023	0.1495/422.73	0.1249/422.73	0.1043/422.73
1024	0.1209/422.73	0.0703/422.73	0.1446/422.73
1025	0.1035/422.73	0.0989/422.73	0.2090/422.73
1038	0.1688/422.73	0.1645/422.73	0.1535/422.73
1039	0.1016/422.73	0.0990/422.73	0.1434/422.73
1101	0.1041/422.73	0.0909/422.73	0.0960/422.73
1104	0.0908/422.73	0.0842/422.73	0.1080/422.73
1107	0.1380/422.73	0.1256/422.73	0.1462/422.73
1110	0.1633/422.73	0.0884/422.73	0.2072/422.73
1113	0.1292/422.73	0.1243/422.73	0.1020/422.73
1116	0.1716/422.73	0.1213/422.73	0.1192/422.73
1119	0.1280/422.73	0.0762/422.73	0.1687/422.73
1122	0.1195/422.73	0.0842/422.73	0.1376/422.73
1125	0.1169/422.73	0.1306/422.73	0.1225/422.73
1128	0.1028/422.73	0.0800/422.73	0.1309/422.73

Table 4.6: Results of segmentation with EM for WM class.

Dice coefficient and computational time (seconds)			
Patient	K-Means	Tissue models	Labels propag
1003	0.8752/456.10	0.8846/456.10	0.9388/456.10
1004	0.8917/422.73	0.8517/422.73	0.9108/422.73
1005	0.8799/422.73	0.8496/422.73	0.9039/422.73
1018	0.8898/422.73	0.8583/422.73	0.9252/422.73
1019	0.8822/422.73	0.8447/422.73	0.9156/422.73
1023	0.8793/422.73	0.8375/422.73	0.9188/422.73
1024	0.8696/422.73	0.8523/422.73	0.9419/422.73
1025	0.8770/422.73	0.8391/422.73	0.9142/422.73
1038	0.8837/422.73	0.8524/422.73	0.9228/422.73
1039	0.8884/422.73	0.8566/422.73	0.9310/422.73
1101	0.8690/422.73	0.8629/422.73	0.9111/422.73
1104	0.8758/422.73	0.8599/422.73	0.9081/422.73
1107	0.8677/422.73	0.8689/422.73	0.9223/422.73
1110	0.8892/422.73	0.8468/422.73	0.9128/422.73
1113	0.8823/422.73	0.8483/422.73	0.9374/422.73
1116	0.8735/422.73	0.8268/422.73	0.9267/422.73
1119	0.8833/422.73	0.8444/422.73	0.9314/422.73
1122	0.8957/422.73	0.8519/422.73	0.9310/422.73
1125	0.8950/422.73	0.8671/422.73	0.9218/422.73
1128	0.8846/422.73	0.8525/422.73	0.9165/422.73

4.6 Atlas into EM

Table 4.7: Results of segmentation with Atlas into EM for CSF class.

Dice coefficient and computational time (seconds)		
Patient	Best previous init	Tissue models & Labels propag
1003	0.0597/615.117	0.1731/592.086
1004	0.1729/615.117	0.0516/592.086
1005	0.0766/615.117	0.1796/592.086
1018	0.1314/615.117	0.0745/592.086
1019	0.0873/615.117	0.0948/592.086
1023	0.1410/615.117	0.2612/592.086
1024	0.0685/615.117	0.0423/592.086
1025	0.0594/615.117	0.0936/592.086
1038	0.1387/615.117	0.0885/592.086
1039	0.0945/615.117	0.1737/592.086
1101	0.1481/615.117	0.1075/592.086
1104	0.4028/615.117	0.1100/592.086
1107	0.1139/615.117	0.0797/592.086
1110	0.1641/615.117	0.0870/592.086
1113	0.1658/615.117	0.1656/592.086
1116	0.1529/615.117	0.0632/592.086
1119	0.0947/615.117	0.1874/592.086
1122	0.1144/615.117	0.0180/592.086
1125	0.1844/615.117	0.2078/592.086
1128	0.1572/615.117	0.0673/592.086

Table 4.8: Results of segmentation with Atlas into EM for GM class.

Dice coefficient and computational time (seconds)		
Patient	Best previous init	Tissue models & Labels propag
1003	0.1338/615.117	0.1692/592.086
1004	0.1791/615.117	0.1206/592.086
1005	0.1406/615.117	0.1718/592.086
1018	0.1625/615.117	0.1298/592.086
1019	0.1449/615.117	0.1379/592.086
1023	0.1664/615.117	0.2044/592.086
1024	0.1374/615.117	0.1169/592.086
1025	0.1337/615.117	0.1374/592.086
1038	0.1655/615.117	0.1354/592.086
1039	0.1478/615.117	0.1695/592.086
1101	0.1692/615.117	0.1430/592.086
1104	0.2711/615.117	0.1440/592.086
1107	0.1555/615.117	0.1319/592.086
1110	0.1756/615.117	0.1348/592.086
1113	0.1763/615.117	0.1662/592.086
1116	0.1711/615.117	0.1253/592.086
1119	0.1479/615.117	0.1749/592.086
1122	0.1557/615.117	0.1072/592.086
1125	0.1837/615.117	0.1831/592.086
1128	0.1728/615.117	0.1269/592.086

Table 4.9: Results of segmentation with Atlas into EM for WM class.

Dice coefficient and computational time (seconds)		
Patient	Best previous init	Tissue models & Labels propag
1003	0.9509/615.117	0.9060/592.086
1004	0.9517/615.117	0.9134/592.086
1005	0.9427/615.117	0.9144/592.086
1018	0.9523/615.117	0.9177/592.086
1019	0.9414/615.117	0.9142/592.086
1023	0.9625/615.117	0.9121/592.086
1024	0.9439/615.117	0.9203/592.086
1025	0.9579/615.117	0.9111/592.086
1038	0.9487/615.117	0.9207/592.086
1039	0.9467/615.117	0.9196/592.086
1101	0.9509/615.117	0.9267/592.086
1104	0.9526/615.117	0.9040/592.086
1107	0.9398/615.117	0.9227/592.086
1110	0.9549/615.117	0.9196/592.086
1113	0.9547/615.117	0.9214/592.086
1116	0.9457/615.117	0.9317/592.086
1119	0.9536/615.117	0.9191/592.086
1122	0.9560/615.117	0.9124/592.086
1125	0.9386/615.117	0.8939/592.086
1128	0.9417/615.117	0.9149/592.086

4.7 MNI Atlas into EM

Table 4.10: Results of segmentation with MNI Atlas into EM for CSF class.

Dice coefficient and computational time (seconds)	
Patient	More feasible init
1003	0.6014/723.185
1004	0.6197/723.185
1005	0.6435/723.185
1018	0.7896/723.185
1019	0.5636/723.185
1023	0.6752/723.185
1024	0.6890/723.185
1025	0.6304/723.185
1038	0.5945/723.185
1039	0.6446/723.185
1101	0.6416/723.185
1104	0.6599/723.185
1107	0.5830/723.185
1110	0.6607/723.185
1113	0.6231/723.185
1116	0.5962/723.185
1119	0.6195/723.185
1122	0.6066/723.185
1125	0.6402/723.185
1128	0.6818/723.185

Table 4.11: Results of segmentation with MNI Atlas into EM for GM class.

Dice coefficient and computational time (seconds)	
Patient	More feasible init
1003	0.1005/723.185
1004	0.1079/723.185
1005	0.1174/723.185
1018	0.1758/723.185
1019	0.0854/723.185
1023	0.1300/723.185
1024	0.1356/723.185
1025	0.1121/723.185
1038	0.0978/723.185
1039	0.1178/723.185
1101	0.1166/723.185
1104	0.1239/723.185
1107	0.0932/723.185
1110	0.1243/723.185
1113	0.1092/723.185
1116	0.0984/723.185
1119	0.1078/723.185
1122	0.1026/723.185
1125	0.1160/723.185
1128	0.1327/723.185

Table 4.12: Results of segmentation with MNI Atlas into EM for WM class.

Dice coefficient and computational time (seconds)	
Patient	More feasible init
1003	0.8277/723.185
1004	0.8136/723.185
1005	0.8117/723.185
1018	0.8175/723.185
1019	0.8191/723.185
1023	0.8259/723.185
1024	0.8254/723.185
1025	0.8344/723.185
1038	0.8207/723.185
1039	0.8170/723.185
1101	0.8219/723.185
1104	0.8160/723.185
1107	0.8245/723.185
1110	0.8128/723.185
1113	0.8236/723.185
1116	0.8173/723.185
1119	0.8205/723.185
1122	0.8345/723.185
1125	0.8181/723.185
1128	0.8268/723.185

5 Project Management details

The plan for the first week of the task was for both team members, to look for information related to the algorithm. To understand what is expected to be submitted, to see the information provided and if it is enough or if is needed more and different information. The objective of the first week is to study and understand the basics of the software elastix, since it is a very complex tool with too many options to work with.

From the second week, the purpose is to start working on the implementation of the algorithm. To begin identifying the files created from elastix, to understand how to run in a efficient way all the necessary commands and to have the right structure with the folders so it can be correctly generated and saved, because the same files are going to be used for the next step.

Then the plan is to focus on the actual implementation for the python file to run and acquire all the data asked for. By looking at the information gathered during the first week, in order to have the best possible approach to implement the segmentation algorithm.

At the end, the real time dedicated was not as planed, due to the inconveniences. At the beginning, it was believed elastix worked along with Matlab or Python as a library, it was until long after when it was figured it out. It must be used using the command window and the way to run the commands was another step to figure out. The manual helped up until certain point, but it was necessary to try different things with the purpose of understand the software and get the files for both steps of the project. Because of different deadlines, sometimes it was not possible to work on this project during the lab classes. But more than enough time was dedicated in different days, different hours, with the purpose of submitting the project as expected.

6 Conclusions

All segmentation methods were applied for all cases as apparent from tables 4.1 through 4.12. For convenience, box plots have been computed. This way we can gain a visual indication of the qualitative aspects of different approaches and their outcomes expressed in DICE coefficients.

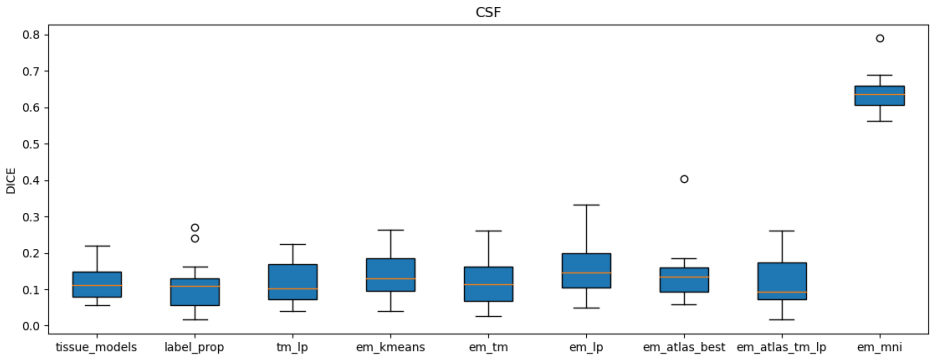


Figure 6.1: Box plots for the CSF class.

In the case of CSF, because of issues encountered in the initial processing of this region, the DICE scores were the lowest and least reliable. Problems were often encountered in the beginning of the project because of NaNs in the EM part of the pipeline, these combined with sparse volumes were often leading to singular matrices which would break the algorithm. These were overcome by careful case programming. This being said, out of the 3 segmentations, CSF still has the highest interquartile variance, further proving the lack of robustness of most approaches in this region. A further exploration of the data has revealed that the probabilistic atlas and tissue models that were computed have been the subject of an error that lead to the inaccurate processing of the CSF area in the pipeline. Unfortunately, in spite of rigurous debugging, the error was not identified and the results are precarious in every pipeline but the MNI segmentation. Although still presenting high variability, CSF segmentation with the MNI atlas produced a better DICE than the rest although lower than any segmentation in other regions such as WM and GM.

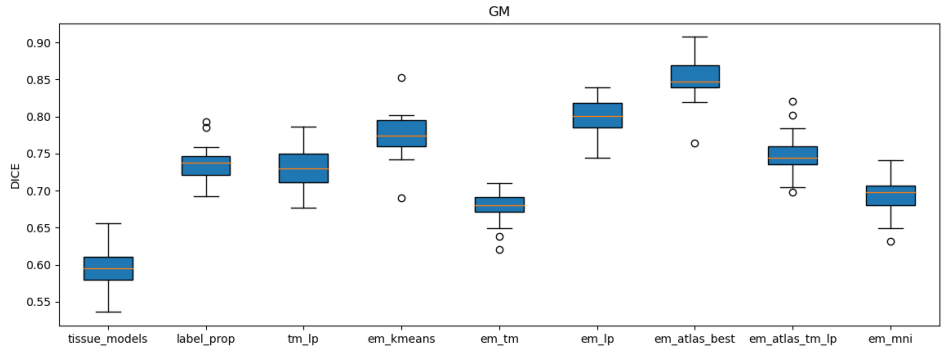


Figure 6.2: Box plots for the GM class.

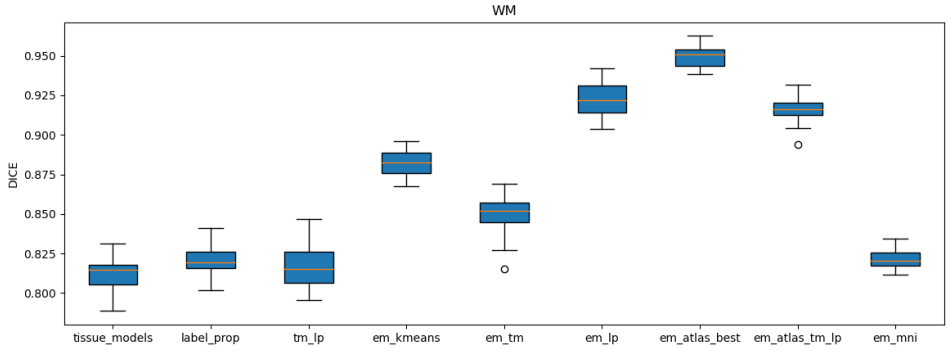


Figure 6.3: Box plots for the WM class.

In the case of Gray Matter and White Matter segmentation, a clear improvement can be seen in terms of both DICE coefficients and general interquartile variance. EM improves previous results when these are used for generating initial parameters for the algorithm. This is because of the additional optimization steps that are undergone until achieving the end result. Moreover, a new performance trend becomes apparent: the best segmentation results were achieved

when the EM algorithm was initialized with the probabilistic atlas constructed from the given dataset. As a sidenote, it was expected that this atlas would perform better than the MNI because it offers a more accurate, although probably biased, representation of the dataset at hand.

Outside the realm of EM, differences begin to appear between the performances of the segmentation pipelines. In the case of GM, we observe a worse overall performance with the segmentation of tissue models being the worst of them all and the one using label propagation appearing to be having a slightly higher median performance when compared to when we add tissue models into the method; doing so not only brings down the median performance but also increases the variability of the results.
In the case of WM segmentation, all non-EM approaches are similar. The trend with the variability introduced by the tissue models' usage still keeps true and one could argue that, statistically, the label propagation still did the best, but this time around by a very small and less reliable margin.

To conclude, this project has been a real challenge in terms of inter-technology operability with the elastix routines integration in the segmentation pipeline. Due to this aspect, the approach used was not a streamlined one, but a dynamic one. The main problems encountered were from a project management point of view as time quickly became a resource for which many projects are competing for in unison. Improvements to this project consist in streamlining the codebase, making it more executable-like and bringing more efficiency in terms of both computational time and coding methodology. The pipelines used are fragile to parameter changes even though the path system and folder structure used are kept consistent throughout the project. The relatively high number of working volumes that were used in the end prompted the use of special data-storing techniques.