Medical Image Segmentation and Applications
Lab 3

Leonardo Pestana Legori, Lisle Faray de Paiva

1 Introduction and problem definition

The aim of this lab project is to implement and evaluate multiple atlas-based segmentation methods for brain tissue segmentation, with and without the use of the expectation-maximization (EM) algorithm. Additionally, we will also use the tissue models created during MIRA lab 2 assignment for the segmentation. The tissue models will be used for segmentation of the brain volumes using only intensity information and also it will also be used as one type of initialization for the EM algorithm.

We will use two different probabilistic atlases in this assignment: one that we created from a training set of images in MIRA lab 2, and another one that is provided by the Montreal Neurological Institute (MNI). We will register both atlases to each test image using elastix, and then use them to segment the test images using the EM algorithm. By incorporating the probabilistic atlas, we can improve the accuracy and robustness of the EM algorithm, especially in cases where the image quality is low or the intensity distributions are overlapping.

We will compare the results of the different segmentation methods and evaluate the their performance using the Dice similarity coefficient.

2 Algorithm analysis / Design and implementation of the proposed solution

The first step was to register both MIRA atlas and the MNI atlas of intensities to each image in the test dataset and to propagate the transformations to the atlas of labels. The registration itself was done using elastix. Same as when we registered the images during MIRA lab to create the MIRA atlas, this time we are also using the parameters from Par0010 at the Model Zoo, which included files to perform affine and B-spline transformations (two separated .txt files) using mutual information as metric. The same modifications were applied to the parameter files for the registration step, but this time, for the transformation, besides setting FinalBSplineInterpolationOrder to 0, we also set ResultImagePixelType to ‘float’. That makes sure that the propagated labels maintain their range of [0-1].

Next we implemented the segmentation pipeline. In this lab our objectives were to develop and perform the segmentation for the following scenarios:

1. Segmentation without EM:

1.1. Tissue models: segmentation using just intensity information

The segmentation using only tissue models is implemented in the function *process_tm_image*. It works as follows: first, we scale the intensities of the input image (the image we want to segment) to be in the [0-255] range. Next, using only the voxels of the brain tissues, we match the intensities of each voxel with the probabilities of that intensity value being CSF, WM or GM, which were retrieved from the tissue models we created in the MIRA lab. We create, as a result of that, a new array with 3 columns, where each column represents the probability of each tissue type for all the voxels in the brain. This array or probabilities will be returned by the function because later we will use it for the initialization of the EM algorithm. Next, to proceed with the segmentation we simply used *np.argmax* on the array, which determines which type of tissue has the highest probability for that voxel. That means that, for each voxel we will have a value of 0 (CSF), 1 (WM) and 2 (GM). Finally, to obtain a valid segmented image we reshape the resulting *np.argmax* array back to its original dimensions.

1.2. Label propagation: segmentation using just position information

The segmentation using the propagated atlas labels was implemented in the function *process_image_atlas*. It takes as input the propagated atlas labels (one for CSF, WM and GM) for a certain test image and simply reshapes each propagated label to be an 1D array, while also excluding voxels that are not in the brain region. After that it concatenates them together, creating a 3 column array with the probabilities for each tissue type for each brain voxel. As with the tissue models, the function will later return this array for use in the EM. Next, to proceed with the segmentation we simply used *np.argmax* on the array, which determines which type of tissue has the highest probability for that voxel. That means that, for each voxel we will have a value of 0 (CSF), 1 (WM) and 2 (GM). Finally, to obtain a valid segmented image we reshape the resulting *np.argmax* array back to its original dimensions.

1.3. Tissue models & label propagation: multiplying both results: segmentation using intensity & position information

This is implemented in the lines [45-50] of the function *get_results_case*. After running *process_tm_image* and *process_image_atlas* for a certain test case we get two 3 columns arrays, one with the intensity and another with the positional probabilities for each tissue type. To get one array taking both probabilities into account we simply multiply those two together. With that we can proceed with the segmentation using *np.argmax* like described above.

2. **Segmentation by EM from the previous session. We will refine using just intensity information an initial segmentation obtained in different ways:**

2.1. Using k-Means initialization (algorithm from last session): segmentation using intensity information

There are a couple of changes since the implementation of the last lab session (in the function *em_segmentation*). Now, in order to make the results repeatable, we set the random state for the k-Means algorithm. Also, for each cluster, we now calculate the covariance matrix and the alpha (mixture weight) taking into account only the samples in each respective cluster.

2.2. Using tissue models initialization (1.1): segmentation using intensity information

For this initialization, we use the tissue models probabilities we calculated in the function *process_tm_image* in a maximization step before starting the main EM algorithm loop. That allows us to initialize the cluster parameters more accurately using the soft probabilities.

2.3. Using label propagation initialization (1.2): segmentation obtained just using position information

Here we do the same procedure as in (2.2), but this time we use the label propagation probabilities obtained from the function *process_image_atlas*.

3. **Add atlas (from MIRA) information after/into EM. We will refine using intensity & position information an initial segmentation obtained by:**

3.1. Choosing the best previous (2.i) initialization

Using the initialization that gives the best results in the previous experiment, we implemented the option to integrate the atlas positional information into the EM algorithm. To do that we modified the *em_segmentation* function to accept as input (in the *into_or_after* parameter) the type of procedure we want to do: not using the atlas information ('none'), using the atlas information to calculate the membership weights in every loop ('into') or using the atlas information to calculate the membership weights after the last loop ('after'). For both into and after procedures we simply multiply the current membership weights with the atlas position probabilities, to get the updated membership weights. In this experiment we use the positional probabilities obtained using the MIRA atlas, which is passed as the parameter *em_atlas_weights*.

3.2. Using the tissue models & label propagation (1.3) initialization

For this, we do the same as in (3.1) but changing the initialization to use both tissue models and the label propagation.

4. Add atlas (MNI) information into EM. We will refine using intensity & position information an initial segmentation obtained by:

4.1. Using the tissue models & label propagation (1.3) initialization

We do the same as in (3.2), but this time we use the probabilities calculated using the MNI atlas to provide the positional information into the EM.

For all the EM segmentations we perform label matching with the function *reorder_labels* as described in the last lab's report.

Another important aspect of our implementation is that, since we have many experiments to run for the EM segmentation, we parallelized the EM segmentation procedure code using the Delayed feature from Dask library, which allowed us to run each experiment in parallel (using 8 workers and 2 threads per worker).

The segmentation pipeline workflow itself works as follows:

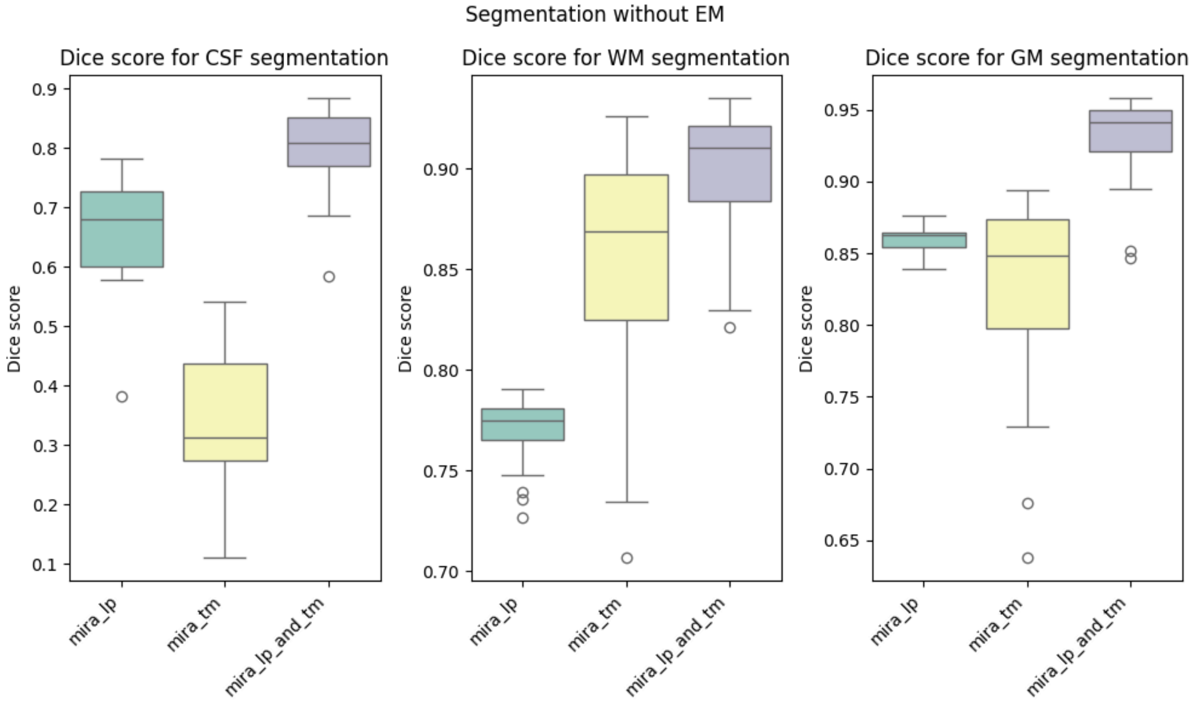
- First, we load the test images and the ground truth labels for all the cases. We also use the ground truth labels to get the tissue mask for each case.
- Next we pass the data we loaded to the *get_results* function. Inside this function we load the MIRA and MNI propagated atlas labels. We also load the tissue models.
- For each case in the test dataset we call the function *get_results_case*. In this function we segment and get the results (Dice score) for each method: label propagation, tissue model, both together, and EM.
- All the results are returned in a list of dictionaries. The individual case results are, in turn, appended to a list containing the results for all cases.
- With this *get_results* finishes execution and returns the list with all the results, which we later transform into a Pandas dataframe and save as a compressed pickle file for future access.

3 Experimental section and result analysis

The test dataset had 20 cases. That means that we have 20 sets of Dice results for CSF, WM and GM segmentation for each of the methods evaluated. Due to that, in order to facilitate the interpretation, all results will be shown with a boxplot and tables, showing the mean and the standard deviation (SD). For the plot labels names we follow this nomenclature: atlas_method. Atlas can be 'mira' or 'mni'; method can be 'lp' (label propagation), 'tm' (tissue models), 'lp_and_tm' (label propagation and tissue models) or 'em+initialization_type+into/after/none'.

3.1 Segmentation without EM

First, we have the results for the segmentation methods without EM using the MIRA atlas:



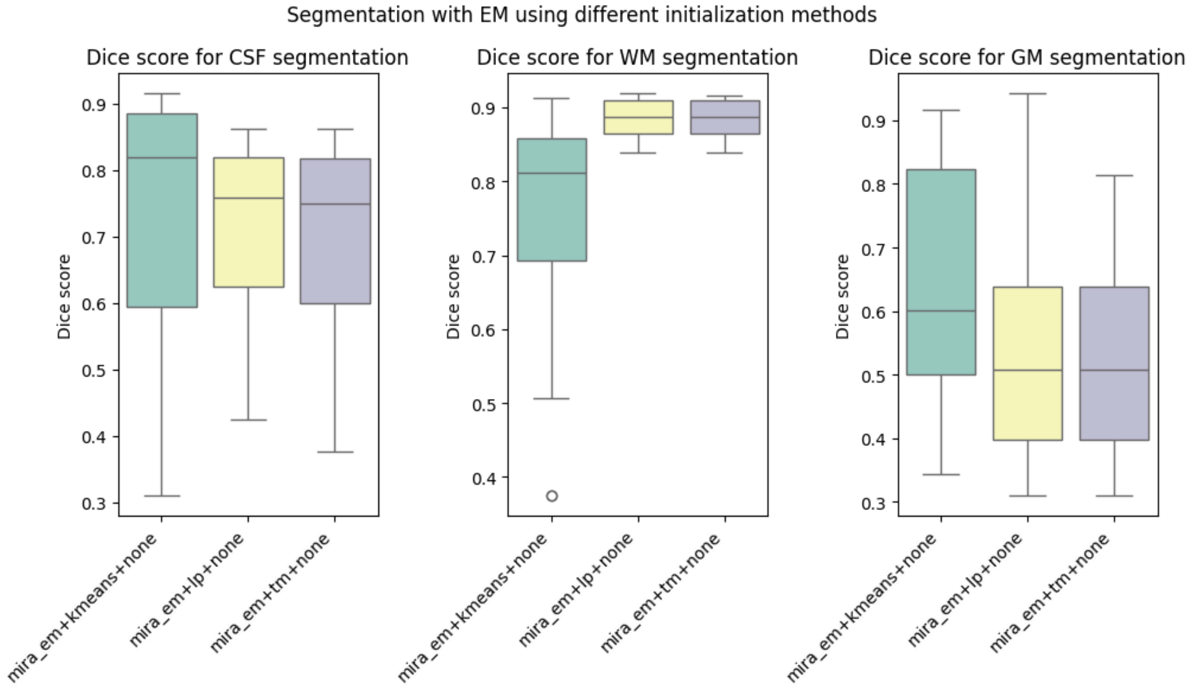
The results in a table are as follow:

Method	Dice CSF		Dice WM		Dice GM	
	Mean	SD	Mean	SD	Mean	SD
lp	0.661105	0.091802	0.769132	0.018490	0.858585	0.010414
tm	0.340959	0.127282	0.852397	0.061667	0.821551	0.072105
lp_and_tm	0.796505	0.071695	0.898175	0.032288	0.928293	0.032890

Using label propagation and tissue models probabilities together produced the best segmentation results in this experiment. By themselves, the label propagation had higher results on CSF and GM and tissue models had better score on WM. This shows that combining positional and intensity information together provides a more robust segmentation.

3.2 Segmentation with EM using different types of initialization

The results with the EM algorithm using different types of initialization can be seen bellow:



The results in a table are as follow:

Init Type	Dice CSF		Dice WM		Dice GM		Time (sec)		Iterations	
	Mean	**SD**	**Mean**	**SD**	**Mean**	**SD**	**Mean**	**SD**	**Mean**	**SD**
kmeans	0.7157	0.2053	0.7466	0.1677	0.6414	0.1956	125.95	45.33	126.30	42.60
lp	0.7188	0.1317	0.8848	0.0256	0.5288	0.1648	139.52	51.01	150.95	50.64
tm	0.6966	0.1501	0.8847	0.0255	0.5223	0.1496	138.22	50.00	146.05	50.97

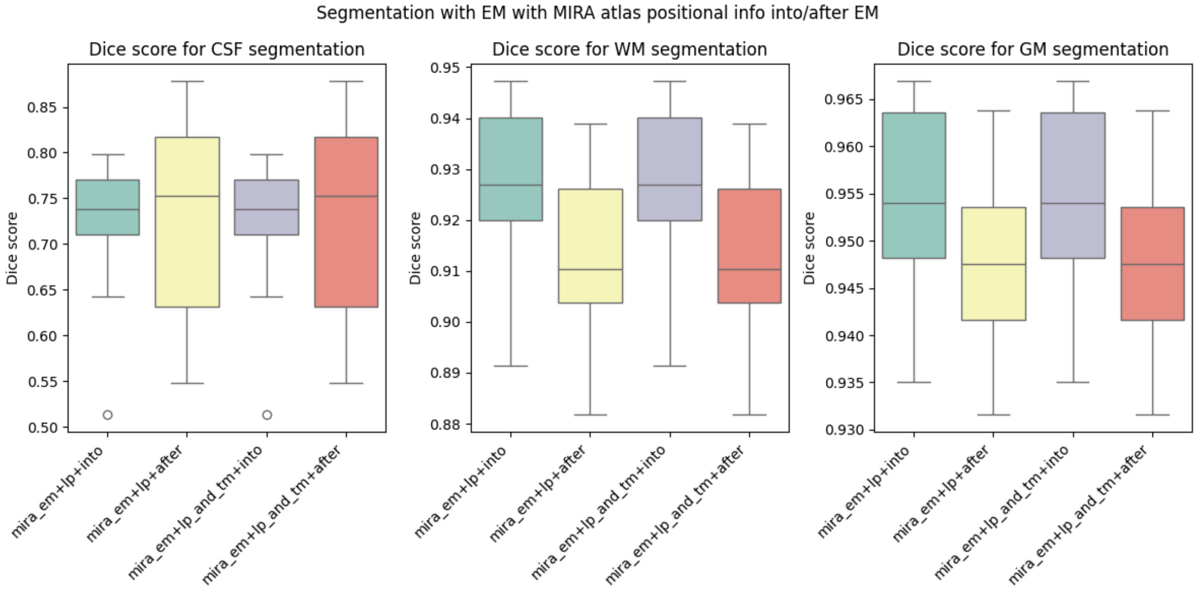
The results show that, overall, k-Means initialization had the worst Dice score results. It had a similar performance to other methods on CSF and a better one on GM, with a difference of 0.1126 in regard to the second best results, while it had a worse performance on WM, with a difference of 0.1382 in comparison with the best results. It also showed much higher SD in comparison with other methods.

In contrast, the Dice score results with label propagation initialization were the best. Following close behind, tissue models initialization had very similar performance, being only slightly worse on CSF segmentation.

3.3 Segmentation with EM using MIRA atlas positional information into/after the EM algorithm

For this section we performed the experiments with the best initialization type determined in the previous experiment, label propagation, and also using label propagation & tissue models together as the initialization method.

The results are shown below:



The results in a table format are:

Init Type	Into or After	Dice CSF		Dice WM		Dice GM		Time (sec)		Iterations	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
lp	into	0.7301	0.0657	0.9279	0.0153	0.9551	0.0100	64.03	15.30	49.30	4.51
lp	after	0.7276	0.1047	0.9129	0.0157	0.9480	0.0090	139.00	49.90	150.95	50.64
lp_and_tm	into	0.7301	0.0657	0.9279	0.0153	0.9551	0.0100	62.83	14.40	48.45	4.62
lp_and_tm	after	0.7276	0.1047	0.9129	0.0157	0.9480	0.0090	137.60	50.20	149.70	52.27

The results show that using the positional information of the MIRA atlas into the EM algorithm produces improved results. We had a substantial improvement on WM and GM segmentation results and a minor one on CSF. Both strategies (into and after) produced similar Dice score results with 'into' being slightly better. The biggest difference, though, is on the number of iterations and, consequently, the time it took for the EM algorithm to converge. Using the atlas positional information into the EM allowed it to converge about 3 times faster than when only using it after the last EM loop.

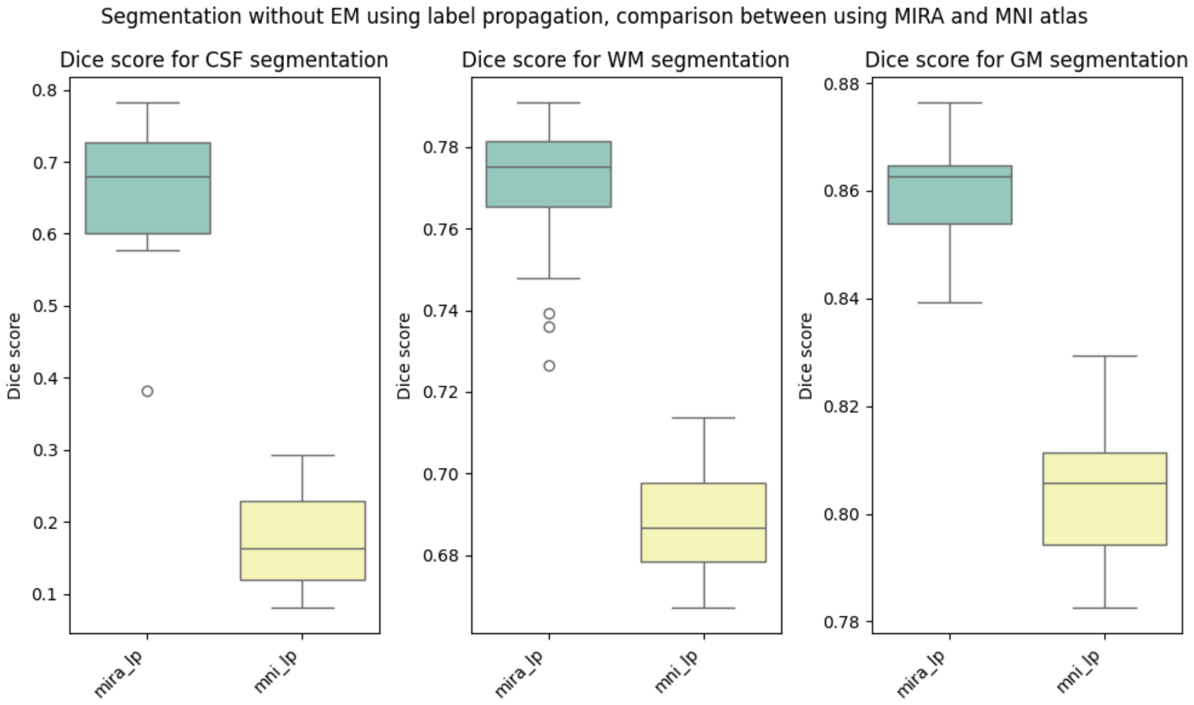
Basically, the use of atlas positional information serves as a form of regularization by imposing spatial constraints on the parameter space. This incorporation guides the optimization process to favor solutions that align with the expected spatial distribution outlined by the atlas, leading to a faster convergence and a more robust model.

Regarding the types of initialization, there is no difference in performance between them. That said, going forward with the next experiment the label propagation & tissue models initialization will be preferred due to it allowing the EM algorithm to converge slightly faster, needing on average about 1 less iteration to converge.

3.4 Segmentation with EM using MNI atlas positional information into the EM algorithm

For this experiment, we compare MIRA and MNI atlas performance using the best predetermined parameters for the EM: using label propagation & tissue models initialization and using the atlas positional information into the EM.

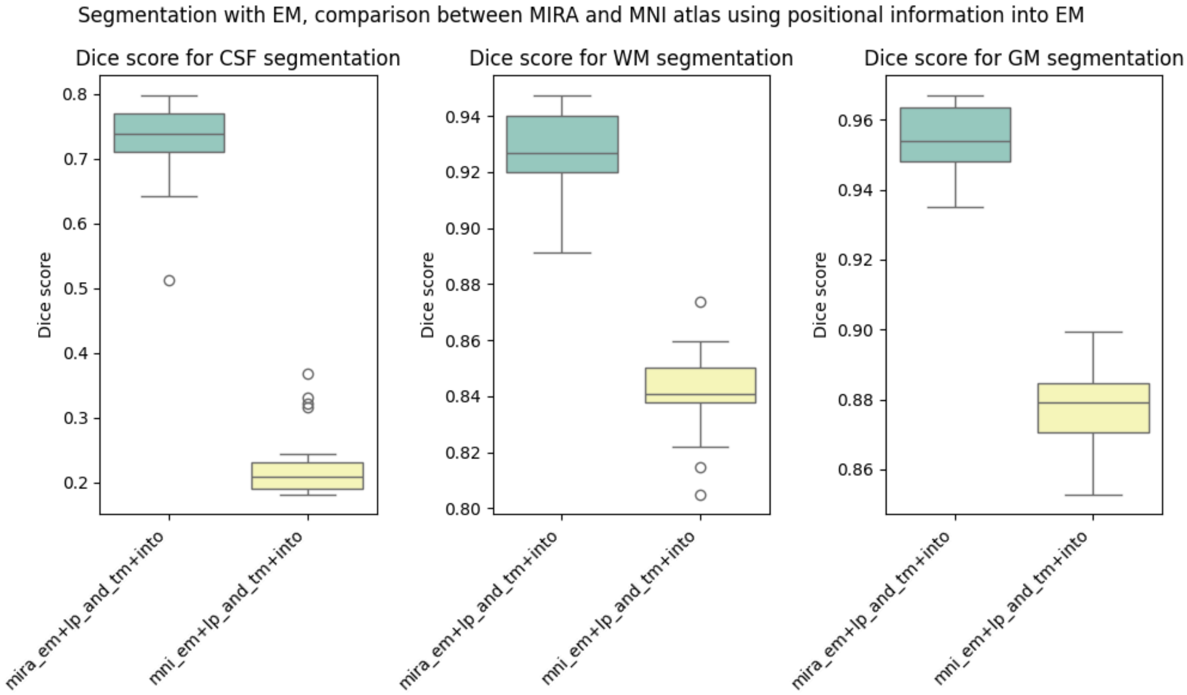
It's important to note that for all the runs of the algorithm, the label propagation & tissue models initialization is computed using only the probabilities extracted from the MIRA probabilistic atlas. The MNI atlas is not used for initialization. We made this choice due to the fact that the performance of the positional information of MNI atlas is low, which we determined empirically by segmenting the test images using only label propagation (without EM) and comparing it to the MIRA atlas:



The tabulated results are:

Atlas	Method	Dice CSF		Dice WM		Dice GM	
		Mean	SD	Mean	SD	Mean	SD
mira	lp	0.6611	0.0918	0.7691	0.0185	0.8586	0.0104
mni	lp	0.1695	0.0617	0.6889	0.0151	0.8049	0.0122

Now, we can proceed to the main results of this experiment:



The results are also shown in the following table:

Atlas	Method	Init Type	Into or After	Dice CSF		Dice WM		Dice GM		Time (sec)		Iterations	
				Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
mira	em	lp_and_tm	into	0.7301	0.0657	0.9279	0.0150	0.9550	0.0100	62.83	14.44	48.45	4.62
mni	em	lp_and_tm	into	0.2294	0.0571	0.8416	0.0160	0.8780	0.0100	58.79	11.71	44.90	5.67

The results show a significant performance drop when using the MNI atlas positional information into the EM algorithm, especially in the case of CSF segmentation. This could be attributed to differences in the quality and spatial accuracy of the atlases. The MNI atlas may not be well-aligned with the specific characteristics of the test dataset, leading to inaccurate positional priors during the EM algorithm. Additionally, variations in image acquisition protocols, anatomical differences, or registration inaccuracies between the MNI atlas and the test images could contribute to a mismatch in spatial information.

Consequently, the EM algorithm, when guided by MNI atlas information, may struggle to converge effectively and produce suboptimal segmentation results, particularly evident in the decreased Dice scores for CSF segmentation. Based on that, it can be said that the choice of the atlas for spatial priors is crucial, and in this case, the MNI atlas appears less suitable for the specific characteristics of the test dataset compared to the MIRA atlas.

3.5 All the results together

To allow for easier comparison, we show the results for all the methods together in a table:

Atlas	Method	Init Type	Into or After	Dice CSF		Dice WM		Dice GM		Time (sec)		Iterations	
				Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
mira	lp	-	-	0.6611	0.0918	0.7691	0.0185	0.8586	0.0104	-	-	-	-
mira	tm	-	-	0.3410	0.1273	0.8524	0.0617	0.8216	0.0721	-	-	-	-
mira	lp_and_tm	-	-	0.7965	0.0717	0.8982	0.0323	0.9283	0.0329	-	-	-	-
mira	em	kmeans	none	0.7157	0.2053	0.7466	0.1677	0.6414	0.1956	125.95	45.33	126.30	42.60
mira	em	tm	none	0.6966	0.1501	0.8847	0.0255	0.5223	0.1496	138.22	50.00	146.05	50.97
mira	em	lp	none	0.7188	0.1317	0.8848	0.0256	0.5288	0.1648	139.52	51.01	150.95	50.64
mira	em	lp	into	0.7301	0.0657	0.9279	0.0153	0.9551	0.0095	64.03	15.29	49.30	4.51
mira	em	lp	after	0.7276	0.1047	0.9129	0.0157	0.9480	0.0086	139.00	49.88	150.95	50.64
mira	em	lp_and_tm	into	0.7301	0.0657	0.9279	0.0153	0.9551	0.0095	62.83	14.44	48.45	4.62
mira	em	lp_and_tm	after	0.7276	0.1047	0.9129	0.0157	0.9480	0.0086	137.59	50.25	149.70	52.27
mni	lp	-	-	0.1695	0.0617	0.6889	0.0151	0.8049	0.0122	-	-	-	-
mni	em	lp_and_tm	into	0.2294	0.0571	0.8416	0.0158	0.8776	0.0125	58.79	11.71	44.90	5.67

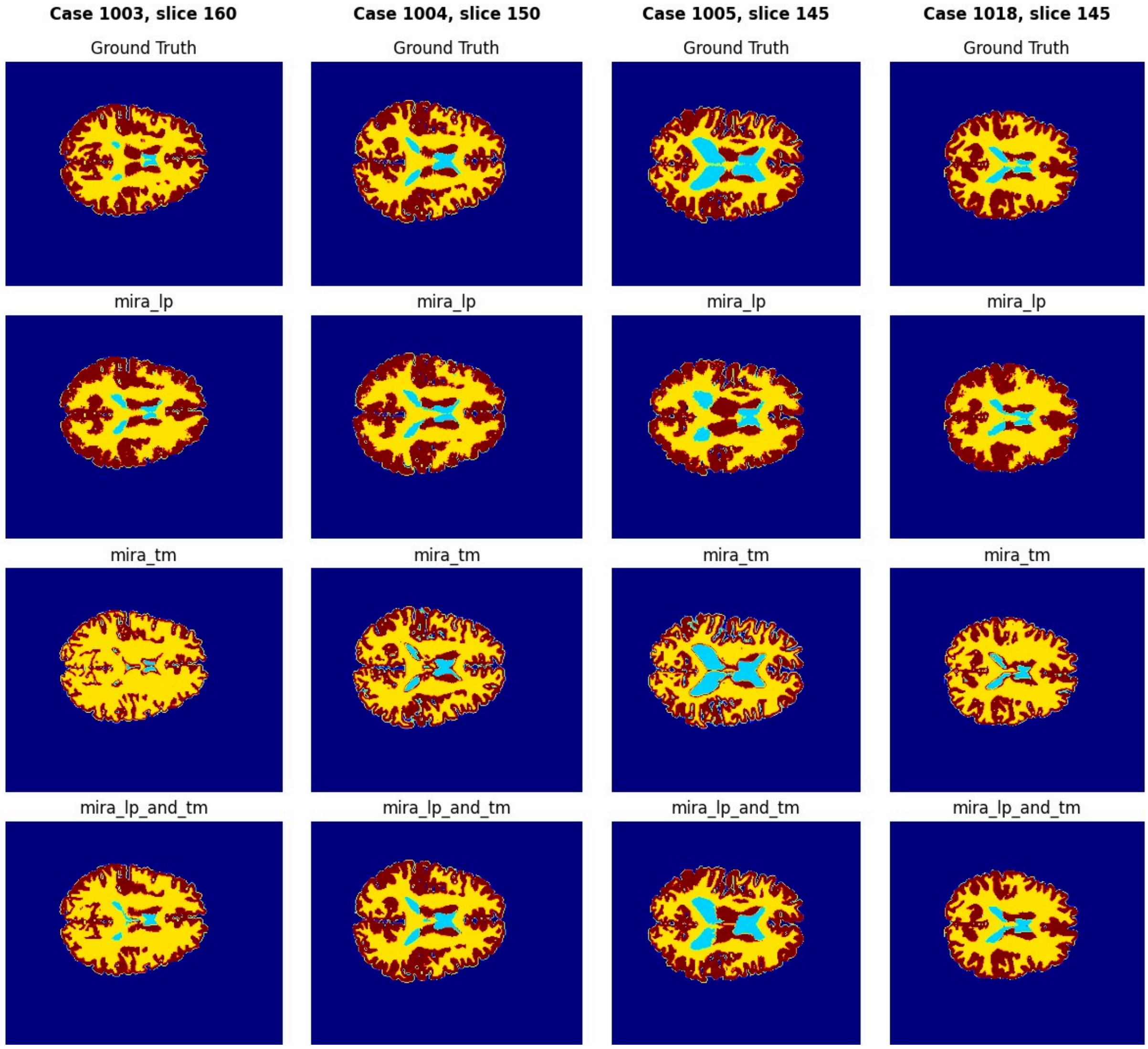
Here we highlight the best segmentation results (Dice score) and the respective methods for each type of tissue:

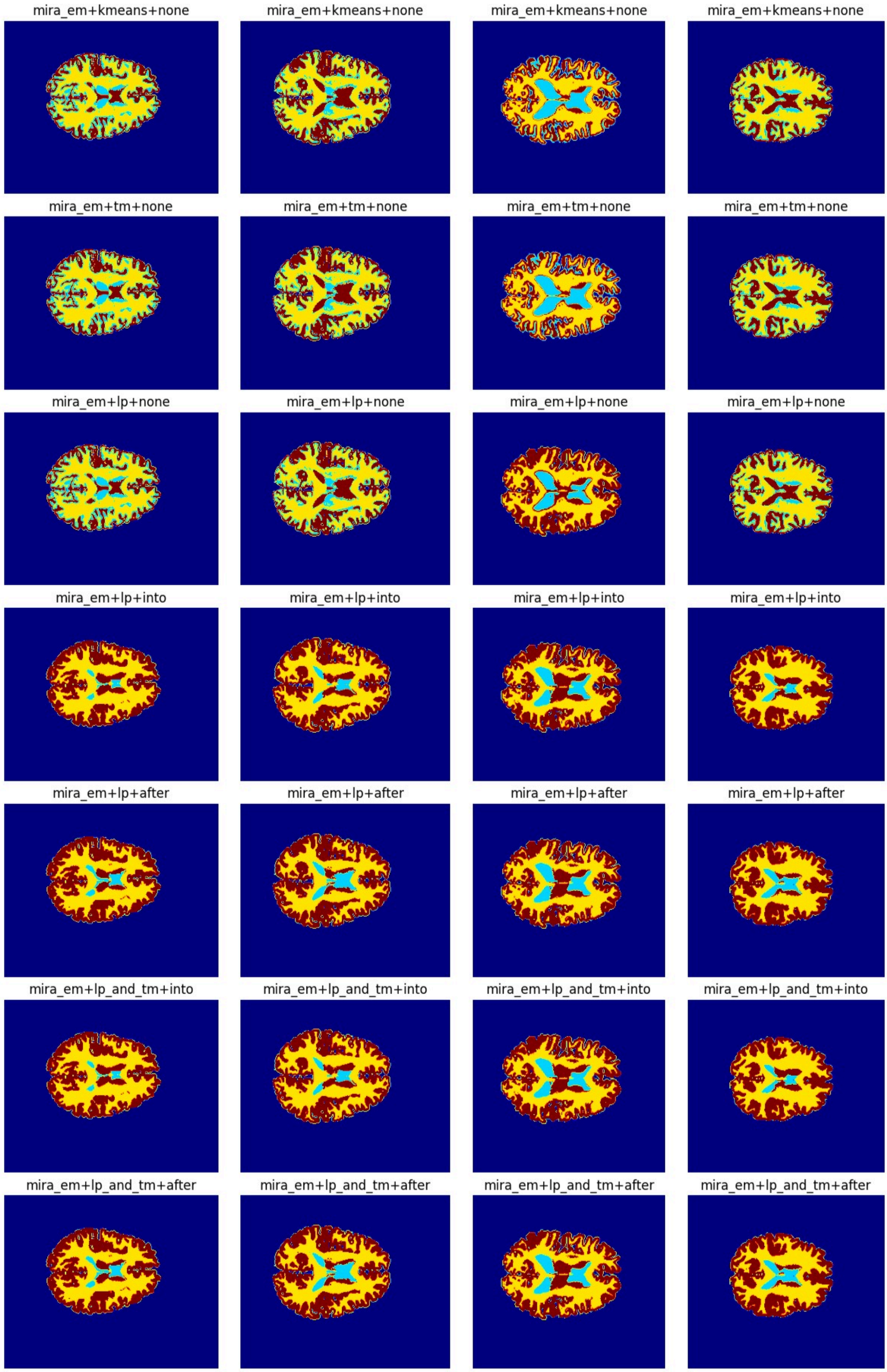
- For CSF the best method was segmentation without EM using both propagated labels (positional information) and tissue models (intensity information) together, achieving a Dice score of 0.7965.
- For GM and WM the best method was segmentation with EM using using label propagation & tissue models initialization and using MIRA atlas positional information into the EM algorithm, producing Dice scores of 0.9279 and 0.9550, with

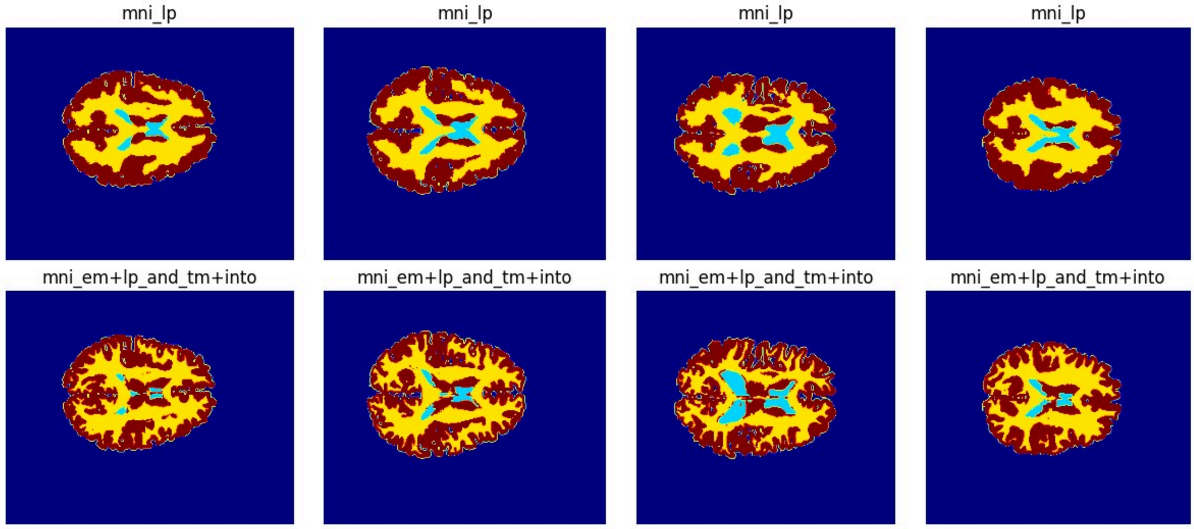
a mean of 137.59 seconds per iteration, and 149.70 mean interaction per segmentation.

3.4 Qualitative results

Below we show the ground-truth and segmented labels of all the methods developed for the first 4 cases in the test dataset. Each case shows the optimal slice in which all the 3 types of tissue (CSF, WM, GM) are clearly visible.




mira_em+kmeans+none
mira_em+kmeans+none
mira_em+kmeans+none
mira_em+kmeans+none
mira_em+tm+none
mira_em+tm+none
mira_em+tm+none
mira_em+tm+none
mira_em+lp+none
mira_em+lp+none
mira_em+lp+none
mira_em+lp+none
mira_em+lp+into
mira_em+lp+into
mira_em+lp+into
mira_em+lp+into
mira_em+lp+after
mira_em+lp+after
mira_em+lp+after
mira_em+lp+after
mira_em+lp_and_tm+into
mira_em+lp_and_tm+into
mira_em+lp_and_tm+into
mira_em+lp_and_tm+into
mira_em+lp_and_tm+after
mira_em+lp_and_tm+after
mira_em+lp_and_tm+after
mira_em+lp_and_tm+after



4 Project management details

The tasks were done in conjunction and with the same amount of effort by both group members.

5 Conclusions

In this lab project, we implemented and evaluated multiple segmentation methods for brain tissue segmentation, encompassing both atlas-based approaches with and without the EM algorithm, as well as tissue models segmentation. We used two different probabilistic atlases (MIRA and MNI) and integrated them into the EM algorithm in different ways. We also used different types of initialization for the EM algorithm, namely k-Means, tissue models, label propagation, and label propagation & tissue models together.

We found that using the MIRA atlas positional information into the EM algorithm improved the segmentation accuracy and robustness, especially for WM and GM tissues. We also found that using label propagation and tissue models together as the initialization method produced the best results among the different initialization methods. We observed that using the MNI atlas positional information into the EM algorithm resulted in a significant performance drop, especially for CSF segmentation. This highlights that one of the main limitations of our approach is that it relies on the quality and accuracy of the probabilistic atlases, which may vary depending on the registration method and the training set.

Notably, it is also worth mentioning that using the MIRA atlas positional information and the tissue models intensity information together for segmentation without EM produced the best CSF segmentation result (Dice score of 0.7965), with the EM algorithm using label propagation & tissue models initialization and using MIRA atlas positional information into the EM algorithm producing the best results for WM and GM (Dice scores of 0.9279 and 0.9550, respectively).