Universitat De Girona

MIRA & MISA

Atlas Based Segmentation (Integration to the EM algorithm)

Authors:
Ali Berrada
Ama Katseena Yawson

Supervisors:
Robert Martí
Xavi Lladó

December 11, 2018

Contents

List of Figures

List of Tables

Part I

MIRA

1 Introduction

1.1 Overview

Image registration is a commonly used approach in medical image processing. It is the task of finding the correspondence of two or more images taken at different times or using different imaging modalities. This technique is usually applied in areas such as atlas - based segmentation, alignment of pre- and post-contrast images and many others. The most intuitive use of registration in the medical domain is to correct for different patient positions between scans [2]. An example of medical image registration is shown in Figure 1.

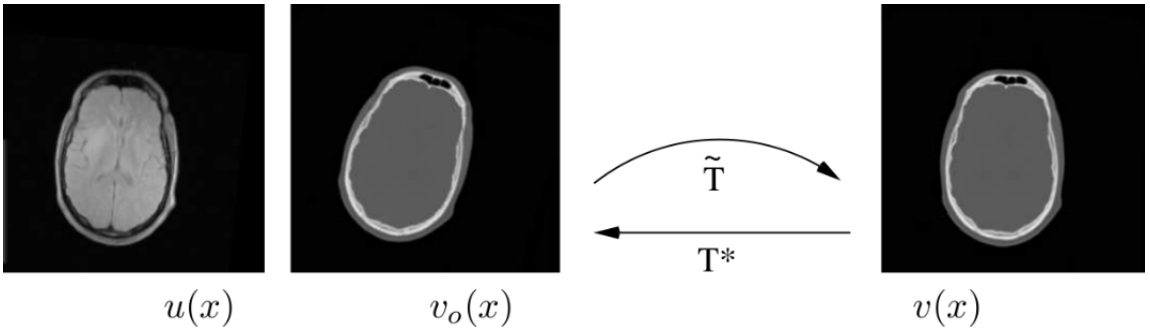


Figure 1: Illustration of Medical Image registration

1.2 Problem Definition

A more accurate image registration technique requires many good choices to be made in the registration framework such as the optimization method, the multiresolution strategy, the method of image interpolation to evaluate, the coordinate transformation model, and the definition of the cost function. Whereas in many applications it may be sufficient to consider only rigid transformations(translation and rotation), usually a more flexible transformation model is needed, allowing for local deformations known as the non - rigid transformation [3]. In medical image processing research, there is the need to validate several options for each of the registration components. This makes the registration process very tedious. To facilitate the research on medical image registration and to simplify its application, Stefan Klein and et al [3] have developed an open source software package called elastix. In this exercise, the underlying objective were as follows:

- To understand and become familiar with the elastix sofware by performing both rigid and non-rigid image registration on the 3D brain images.
- To build a probabilistic atlas from a set of training data for the 3 tissue models(CSF, White matter and Gray matter).
- To display some registration results of the final probabilistic atlas (intensities and label probabilities) and the tissue models for each tissue class by building a histogram distribution.

2 Implementation

Registration was performed using the Elastix open source software over the provided training set consisting of 15 brain volumes with their corresponding labels and mask. The Matlab R2018b environment was used to automate the registration process and also generate the atlas' intensity template and probabilistic label volumes. The ITK-snap visualization tool was used to visually assess the results.

To ease the use of our code, we provided a "main.m" file where the user can choose which scenario to run among 4 options:

1. Register train data to a reference image.
2. Build the atlas intensity volume.
3. Build the atlas probabilistic label volumes.
4. Compute the tissue models.

One of the helper functions we wrote is *niftiwriteWrapper()*. Before leveraging Matlab's *niftiwrite()* to save a volume, our function builds the correct NifTi metadata for the volume to be generated given another volume as reference. Building the correct metadata is important to obtain correctly oriented volumes. This is also part of our effort to automatize the whole process.

2.1 Registering Train Data

After understanding and familiarizing with the elastix's capabilities and syntax, we performed a simple registration on the "cmd" console with a command such as the below:

```
elastix -f "1000.nii.gz" -m "1001.nii.gz" -out "./out" -p "p1.txt"
```

We extended this command to perform 2-stages registration by providing an additional parameter file. The first file *p1.txt* defines an affine (rigid) registration while the second file *p2.txt* continues the registration with a B-spline (non-rigid) transformation. The elastix's command becomes:

```
elastix -f "1000.nii.gz" -m "1001.nii.gz" -out "./out" -p "p1.txt" -p "p2.txt"
```

The parameter files we used were originally developed for [3].

The found transformation is saved in a text file by elastix. We use it to transform the label (segmentation) volume of the moving image using transformix as follows:

```
transformix -in "1001_3C.nii.gz" -out "Trans1" -tp "TransformParameters.txt"
```

Before executing this command, we changed the order of the B-Spline interpolation used at the final deformation to 0 in the parameter file (parameter *FinalBSplineInterpolationOrder*). This is the suitable order to preserve the values of the label volume after deformation. This has been explained in the lab and is also mentioned on the elastix manual.

As we have to register the train data to a reference train image (a tedious and error-prone work to do manually), we developed Matlab scripts to automatize the whole process. The first train image was set as fixed and the rest of images were registered to it. The function *registerIntensityData()* runs the elastix command and automatically modifies the *FinalBSplineInterpolationOrder* parameter in the generated transformation parameter file. The function *transform-*

LabelData() does the transformix part. The elastix and transformix commands were executed from Matlab with the help of the *system()* function.

The registration of intensity data takes about 2.5 minutes per volume while the transformation of the label volume takes around 30 seconds.

2.1.1 Parameter Files

In our implementation, we combined both the rigid and non rigid registration approaches. The foremost quickly finds a rough alignment while the second, which time consuming, helps deals with the very fine details of the deformation which is much needed in certain types of registration such as brain images from different patients where the inner structure shapes differ. Different parameter files were used for this cause. Some of the important components for affine and B-spline registration are shown in Table 1 and Table 2 respectively.

Registration framework	4-levels multiresolution
Transform	Affine
Metric	Mutual information
Interpolator	B-spline order 1
Resample Interpolator	B-spline order 3
Optimizer	Adaptive SGD

Table 1: Registration components for affine transform

Registration framework	4-levels multiresolution
Transform	B-spline
Metric	Mutual information
Interpolator	B-spline order 1
Resample Interpolator	B-spline order 3
Optimizer	Gradient Descent

Table 2: Registration components for B-spline transform

2.2 Building the Probabilistic Atlas

2.2.1 Intensity Template

In building the atlas template, we initialized the template with the reference image ("1000.nii.gz") and the 14 registered images were added to the initialized image iteratively. The atlas template was built by dividing the sum of all images by the total number of images.

2.2.2 Probabilistic Label Volumes

The function *buildProbabilisticAtlas()* computes 3 label maps. First they are initialized with 0. Then in each transformed label volume we search for the indices of label i. The label map i is incremented by 1 at those indices. After iterating through the 15 label volumes, the values of the label maps are divided by 15.

2.3 Computing the Tissue Models

To speed up the process, all training intensity volumes and label volumes were each concatenated into one array with the help of Matlab's *horzcat()* function. From the label array, we find the indices where there is label i. Using these indices in the image array, we retrieve the intensity values which are then passed to *histcounts()* to build a histogram distribution for label i. Finally we normalize the histogram by dividing by the total number of non-zero values in the label array. Figure 2 is illustration of the histogram distribution for each tissue model.

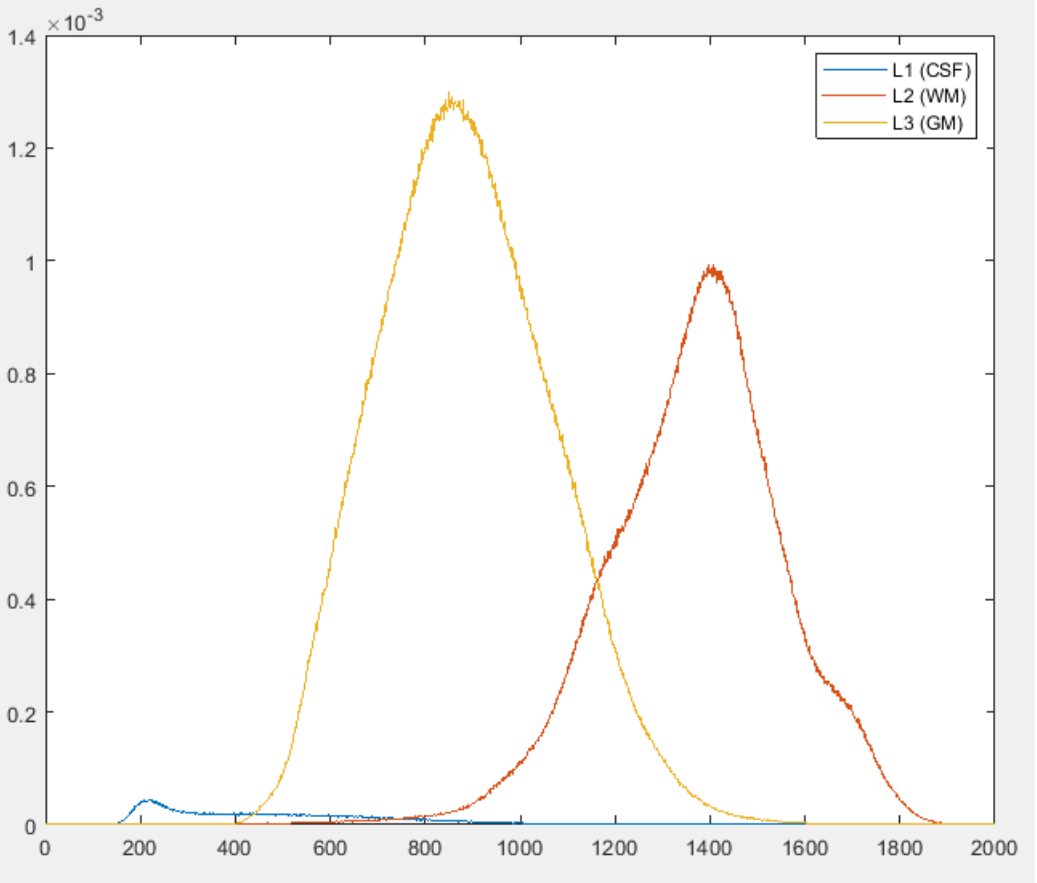
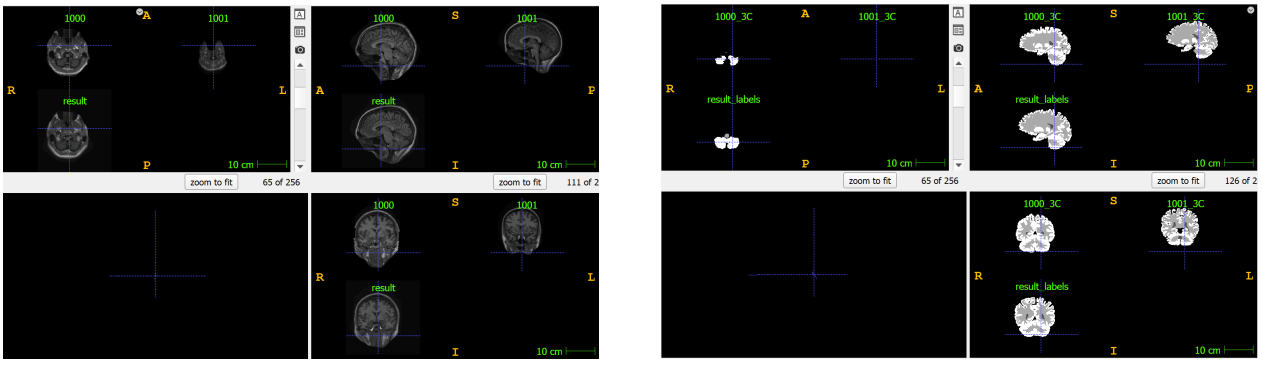


Figure 2: Tissue Model for each class

3 Results and Discussion

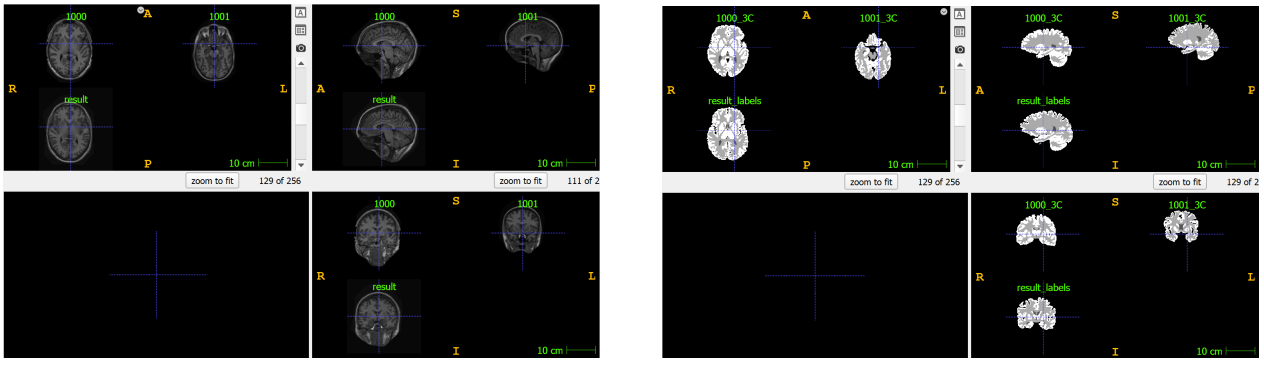
3.1 Registration results

As way of evaluating the registration results, we utilised the ITK-snap visualization tool. From Figure 3a and 4a, it can be observed that from the reference image '1000' and moving image '1001' are at different locations at the same points. The 'result' represents the registered image. By comparing the registered image to the reference image, they both point at the same location at different times. Although the affine (rigid) registration yields good results in some cases, we found out that affine + bspline(non-rigid) registration was more robust with better alignment and deformation produced; yet, it is computationally expensive (about 150sec per image).



(a) Intensity Image (b) Label Image

Figure 3: Registration results for rigid transformation

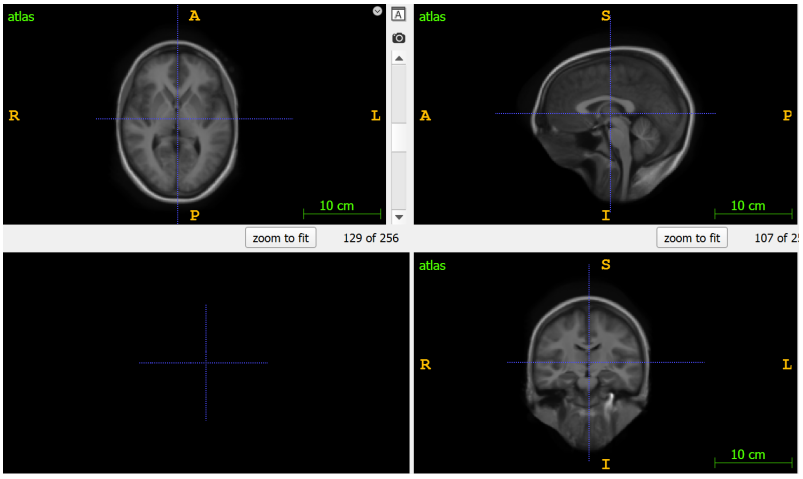


(a) Intensity Image (b) Label Image

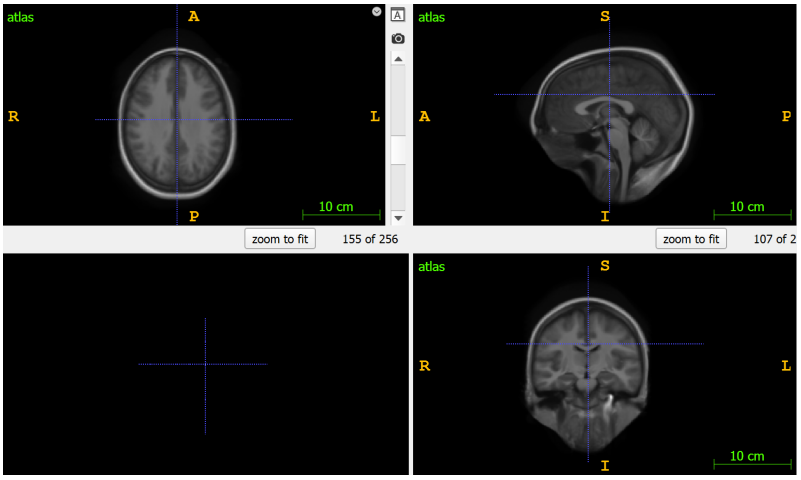
Figure 4: Registration results for rigid and non-rigid transformation

3.2 Final Probabilistic Atlas

After registering all the training images with their corresponding label images, the final atlas (intensity template and 3 probability maps) were generated. The time taken to generate the intensity atlas template and the 3 probability maps were approximately 33.4 sec and 24.9 sec respectively. Figure 5a and Figure 5b are some samples of the intensity template of the final atlas. Figure 6 is the probability map for the 3 tissue models.



(a) Intensity Template at slide 129



(b) Intensity Template at slide 155

Figure 5: Final Atlas - Intensity Template

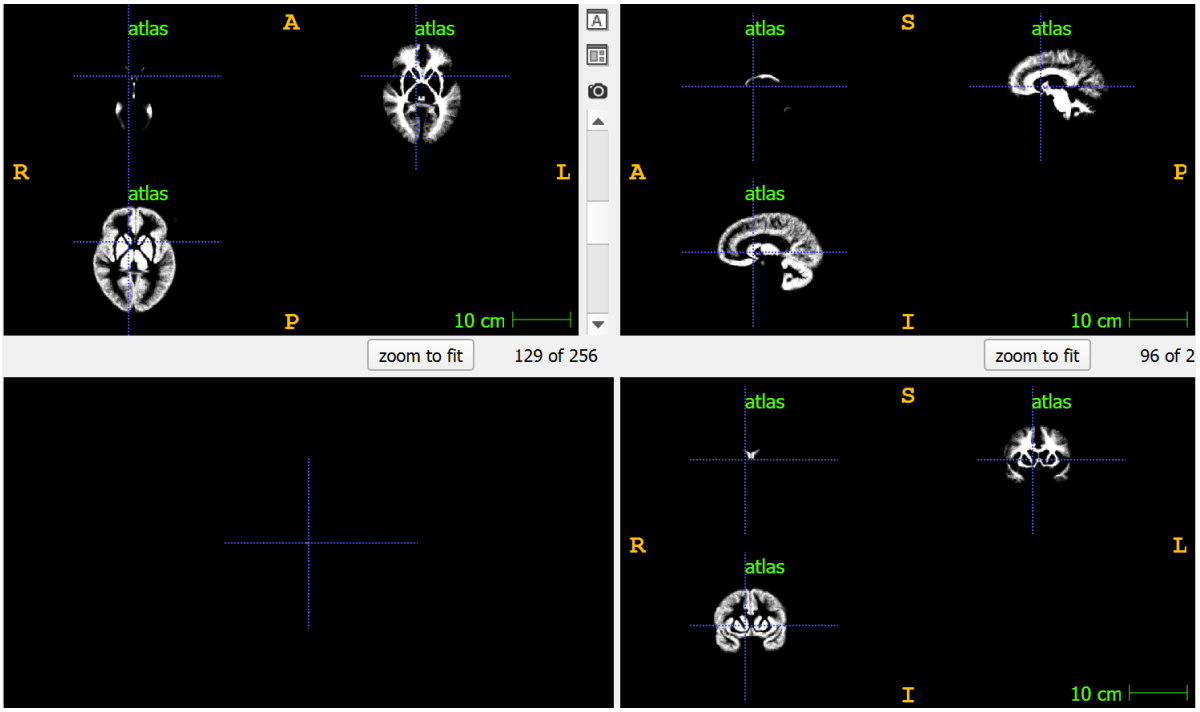


Figure 6: Final Atlas - Probabilistic maps

4 Project Management

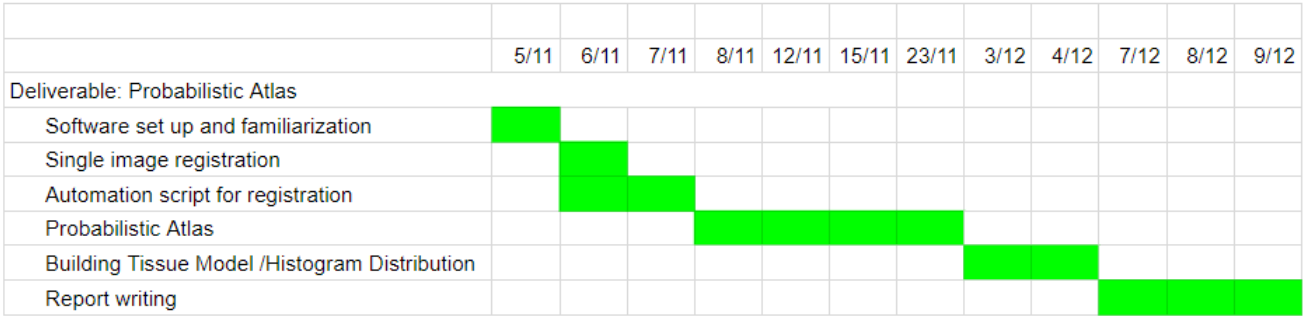


Figure 7: Gantt chart

5 Conclusion

Image registration is a commonly used approach in medical image processing. In this coursework, our primary goal was to build a probabilistic atlas from a set of brain volumes with their available labels (segmentation) of three classes (WM, GM and CSF). To carry out this task, we used the elastix software environment to register images to a reference image. Once registration was completed, the transformation parameters were used to transform the label images. We developed an algorithm from scratch which uses the registered images and label images to generate an atlas intensity template and probability maps for the 3 tissue types. As way of evaluating the registration results, the ITK-snap visualization tool was used to visually access the results. Although the registration results of affine (rigid) gave good results in some cases, we observed that affine + bspline(non-rigid) registration was more robust. The success of our atlas will be quantitatively assessed in Part II.

References

[1] Rolf Heckemann. Segmenting brain images with maper. `https://soundray.org/maper/`. Accessed: 2018-11-02.

[2] William R Crum, Thomas Hartkens, and DLG Hill. Non-rigid image registration: theory and practice. *The British journal of radiology*, 77(suppl_2):S140–S153, 2004.

[3] Stefan Klein, Marius Staring, Keelin Murphy, Max A Viergever, and Josien PW Pluim. Elastix: a toolbox for intensity-based medical image registration. *IEEE transactions on medical imaging*, 29(1):196–205, 2010.

[4] Anita Khanna, Meenakshi Sood, and Swapna Devi. Us image segmentation based on expectation maximization and gabor filter. *International Journal of Modeling and Optimization*, 2(3):230, 2012.

Part II

MISA

6 Introduction

6.1 Overview

Segmentation of non-trivial images is one of the most difficult tasks in image analysis. It can be defined as the partition of an image into several regions which exhibit same characteristics. This image processing technique is often the first step for image analysis and is a key basis of many higher-level activities such as visualization, compression, medical diagnosis and other imaging applications [4]. Figure 8 is an example of image segmentation of the brain. There exist several approaches to segment an image. Some of these techniques include edge based, region based, clustering based and many others.

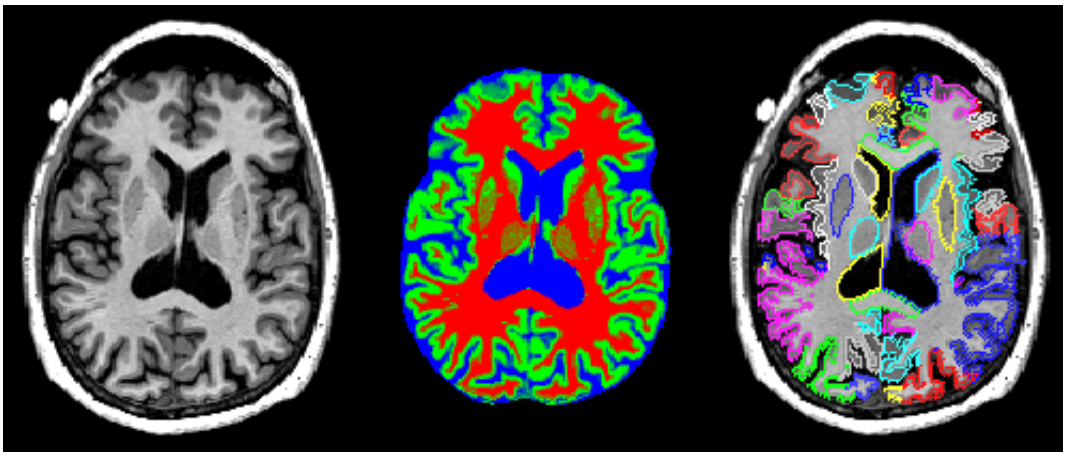


Figure 8: An example of Brain Image Segmentation [1]

6.2 Problem Definition

The quantitative evaluation of brain MRI images are associated with an estimation of tissue density and partitioning them into regions of interest. In the last decades, this field of interest has received massive attention as a result of improved accuracy and resolution of MRI systems. Additionally, MRI images provide high contrast between soft tissues in 3D. However, the amount of data obtained is very numerous for manual analysis especially in the case of diagnosis. Hence, there is a need for automated techniques of image analysis to perform segmentation of Brain MRI images into 3 different tissue classes which includes gray matter, white matter and cerebrospinal fluid. In this exercise, our main goal is to develop an atlas based segmentation approach by integrating the use of a probabilistic atlas into the EM algorithm to segment 3D brain MRI images into the 3 main tissue models: white matter (WM), gray matter (GM) and cerebrospinal fluid (CSF) and evaluate the dice coefficient.

7 Implementation

The starting point for running our program is the "main.m" file. In there, the user can make some choices as how to run the experiments. They are as follows:

1. id: the number of the image to be segmented. The value should be from 1 to 20. If the value is 0, then all test images will be segmented.
2. atlasType: which atlas to use. The value should be conform to the the subfolder names under the "./atlas" folder, e.g. "bspline1" or "mni".
3. scenario: which type of segmentation to use. It accepts values from 1 to 4: 1 for Atlas + EM, 2 for Atlas + TM, 3 for Atlas only and 4 for EM only.
4. printResults: prints the DSC values if the value is 1.
5. saveResults: saves the DSC values as a .mat file if the value is 1.
6. saveSegmentation: generate the segmented volume if the value is 1.

To make the segmentation, the probability table is first computed. This table is a matrix of N rows and 3 columns. The rows represent the image pixels where the corresponding mask image has a value of 1. The columns refers to the 3 labels. The matrix element at row i and column j gives the probability that pixel i comes from cluster j. This probability matrix is computed by combining the probability matrices provided by getAtlasProbs(), getEMProbs() and getTMProbs() functions using element-wise multiplication like follows:

$$Probs_{\text{final}} = Probs_{\text{Atlas}} * Probs_{\text{EM}} * Probs_{\text{TM}}$$

The "main" script is able to handle all types of segmentation with the above formula by initializing $\text{Probs}_{\text{Atlas}}$, Probs_{EM} and Probs_{TM} with 1. Depending on the scenario the user selected, one or more of these matrices may be computed by its corresponding function. The probabilities from EM and tissue models (TM) will never be combined since no scenario proposes that.

After $\text{Probs}_{\text{final}}$ is computed, a hard assignment is performed using the getSegmentation() function.

7.1 Probabilities from Atlas

The registration of the atlas template to the test image and subsequent transformation of the label maps follows the same logic and methodology described in Part I when registering train data for building the atlas. Each transformed label map is stored as a column of the $\text{Probs}_{\text{Atlas}}$ matrix.

7.2 Probabilities from EM

The EM algorithm developed in the previous coursework was reused with some small but important changes. The covariances were replaced by variances and instead of the multivariave normal pdf, the univariate pdf is computed. Additionally, the k-means initialization has been removed and insights from the train data (when building the tissue models) was exploited to make better initialization of the clusters and their spread. This was also useful to obtain the correct labelling and no label reassignment was needed this time.

7.3 Probabilities from Tissue Models

The tissue models were already generated and saved as .mat file from Part I. The intensities from the image to be segmented were used as indices within each tissue model to get the probability that an intensity value belongs to a particular model. Then, if for some intensities all the 3 models give a probability of 0, the probabilities are turned to 1 to make a neutral contribution when combining with the probabilities of the Atlas, i.e. for those intensities only the Atlas probabilities are considered.

7.4 Intensities adjustment for the MNI template

The intensities of the MNI template ranges from 0 to 9999. We modified this range to match the range of the atlas we built (0-3351) using a simple linear normalization.

7.5 Execution Time

To segment a single image and generate the segmented volume, it takes about 230 seconds (less than 4 minutes). This time can be roughly divided into the following parts:

- Loading files (intensity, mask and ground truth volumes): 5 seconds
- Obtaining Probabilities from Atlas: 210 seconds
- Obtaining Probabilities from EM : 5-6 seconds
- Obtaining Probabilities from Tissue Models: 0.2 seconds
- Segmentation: 0.15 seconds
- DSC computation: 0.05 seconds
- Build and save segmented volume: 6 seconds

The most time consuming component is the atlas registration which is reasonable as a rigid plus a non-rigid registration is taking place in addition to the subsequent transformation of the 3 atlas label maps.

Loading and saving NifTi volumes are also expensive operations. The saving of the segmentation takes even higher time because the metadata is made to be automatically generated to allow correct orientation and display of the volume. The devised function for this is nifitiwriteWrapper() which was discussed in Part I.

8 Results and Discussion

8.1 Custom Atlas

	Custom Atlas			Custom Atlas + EM			Custom Atlas + TM		
Image	CSF	WM	GM	CSF	WM	GM	CSF	WM	GM
1003	0.615	0.794	0.863	0.742	0.936	0.959	0.691	0.865	0.895
1004	0.815	0.792	0.852	0.818	0.933	0.959	0.851	0.933	0.956
1005	0.772	0.754	0.837	0.908	0.919	0.953	0.903	0.914	0.947
1018	0.807	0.83	0.883	0.837	0.948	0.968	0.755	0.919	0.942
1019	0.711	0.777	0.854	0.771	0.932	0.962	0.776	0.923	0.955
1023	0.644	0.788	0.861	0.724	0.937	0.961	0.744	0.914	0.939
1024	0.801	0.789	0.864	0.819	0.939	0.965	0.835	0.922	0.95
1025	0.767	0.751	0.843	0.906	0.912	0.952	0.895	0.908	0.948
1038	0.784	0.792	0.858	0.844	0.936	0.96	0.747	0.925	0.948
1039	0.712	0.78	0.85	0.793	0.931	0.96	0.736	0.922	0.95
1101	0.862	0.824	0.894	0.869	0.933	0.961	0.848	0.932	0.956
1104	0.788	0.77	0.833	0.813	0.906	0.941	0.775	0.876	0.914
1107	0.767	0.836	0.898	0.788	0.901	0.931	0.631	0.749	0.814
1110	0.741	0.787	0.857	0.831	0.929	0.956	0.738	0.899	0.934
1113	0.75	0.818	0.876	0.807	0.921	0.941	0.693	0.816	0.84
1116	0.77	0.776	0.861	0.888	0.914	0.951	0.863	0.92	0.95
1119	0.78	0.775	0.849	0.9	0.906	0.934	0.862	0.879	0.908
1122	0.703	0.793	0.864	0.893	0.921	0.947	0.839	0.873	0.902
1125	0.762	0.771	0.858	0.903	0.906	0.944	0.891	0.906	0.937
1128	0.801	0.804	0.882	0.885	0.898	0.945	0.855	0.886	0.936

Figure 9: DSC results per tissue type and different technique combinations using Atlas built from train data.

The segmentation using our atlas gave fair results overall. Gray Matter is the tissue which consistently received higher score followed by White Matter. CSF has an average DSC of 0.77 but with larger spread than other tissue types. This is not strange about CSF as it is the smallest region and at delicate locations, and with the deformation of the atlas to fit the test image more pixels of from CSF are lost or misclassified.

This primary segmentation was combined with other methods, namely the EM algorithm and the information from tissue models. The integration of both techniques to the atlas showed great improvements on the scores.

The EM algorithm particularly excelled and pushed the medium DSC value of CSF to above 0.82. For some images, the value even reached 0.9 while for Gray Matter and White Matter it was the minimum Dice index for almost all images. The robustness of the EM algorithm is mainly due to a good initialization of the cluster centroids and variances. As mentioned in the implementation section, the initialization was based from processing the train data which has ground truth and therefore pixel intensities were classified by tissue types after which the mean intensity and variance of each tissue type could be drawn.

The combination of atlas with tissue models information brought much benefits for many images but for some other images and for different types of tissue, there was little improvement or sometimes the opposite effect. Yet, in general it showed more reliability than if using the atlas data alone.

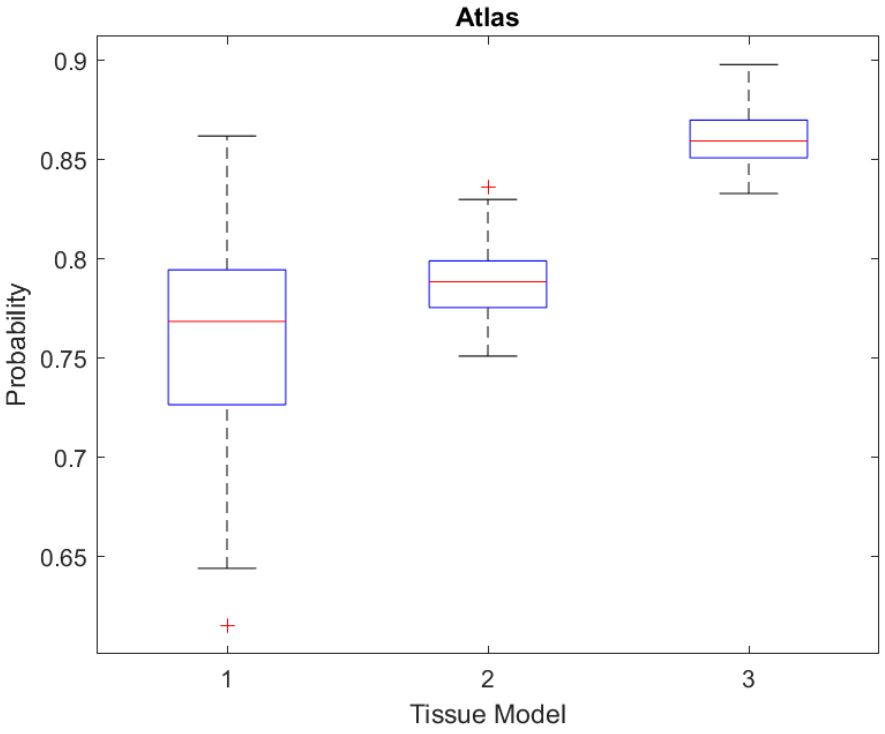


Figure 10: Boxplot for Dice coefficient for test images using Atlas segmentation

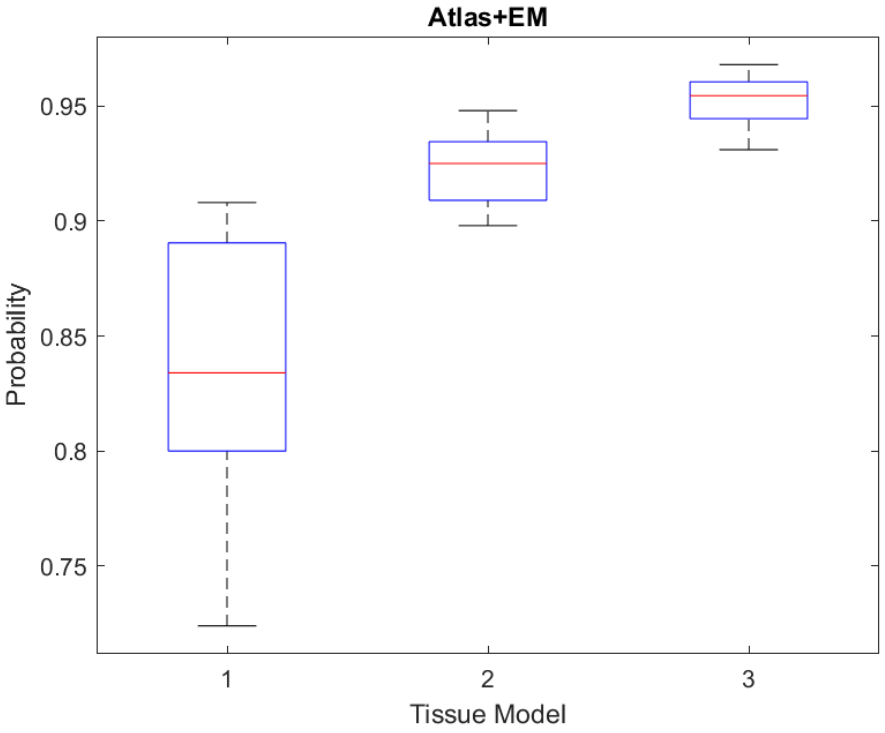


Figure 11: Boxplot for Dice coefficient for test images using Atlas + EM segmentation

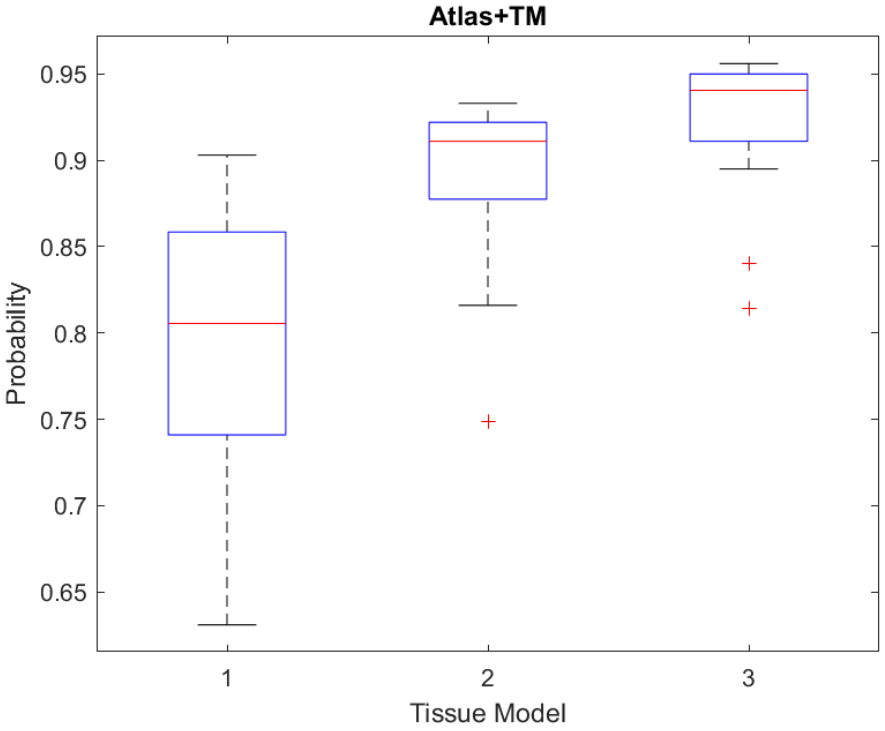


Figure 12: Boxplot for Dice coefficient for test images using Atlas + TM segmentation

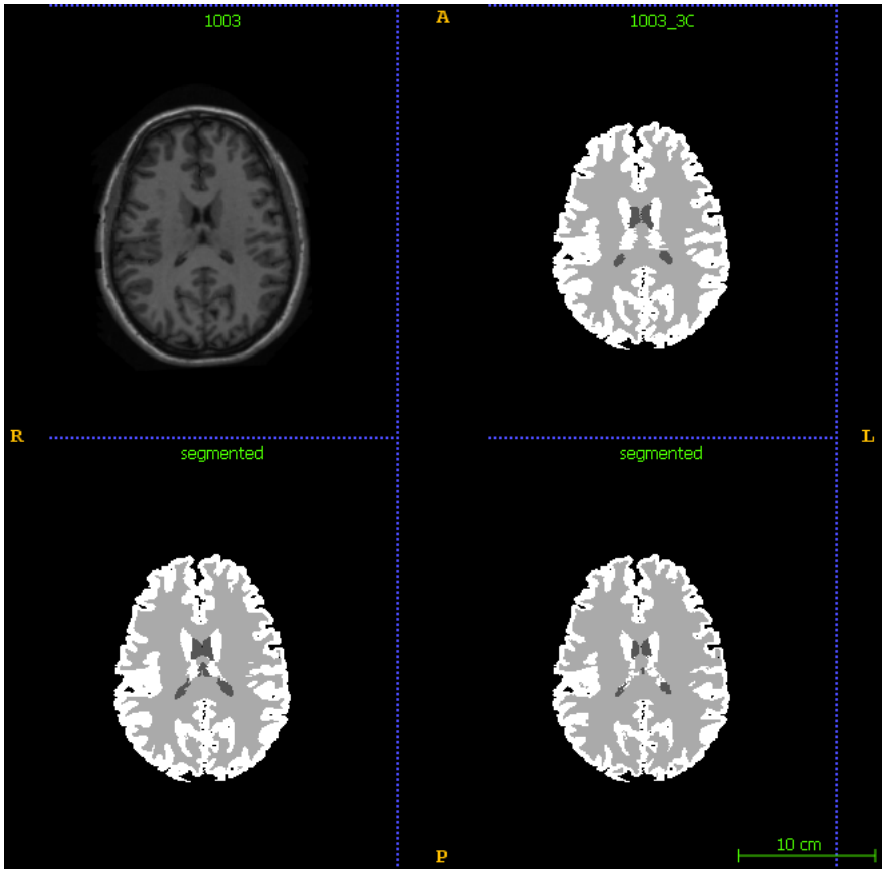


Figure 13: From left to right, top to bottom: intensity image, ground-truth segmentation, Atlas + EM segmentation, Atlas + TM segmentation

8.2 MNI Atlas

	MNI Atlas			MNI Atlas + EM		
Image	CSF	WM	GM	CSF	WM	GM
1003	0.566	0.714	0.861	0.632	0.89	0.934
1004	0.715	0.72	0.85	0.754	0.867	0.925
1005	0.853	0.69	0.86	0.911	0.858	0.928
1018	0.72	0.698	0.855	0.797	0.872	0.928
1019	0.384	0.618	0.805	0.58	0.856	0.926
1023	0.568	0.719	0.852	0.643	0.883	0.932
1024	0.68	0.722	0.862	0.746	0.882	0.936
1025	0.843	0.692	0.867	0.909	0.848	0.928
1038	0.731	0.677	0.821	0.821	0.871	0.927
1039	0.434	0.64	0.808	0.639	0.858	0.925
1101	0.791	0.723	0.862	0.822	0.869	0.928
1104	0.76	0.713	0.87	0.793	0.863	0.927
1107	0.743	0.636	0.853	0.732	0.859	0.919
1110	0.756	0.718	0.872	0.805	0.885	0.938
1113	0.748	0.714	0.858	0.757	0.886	0.924
1116	0.828	0.656	0.849	0.878	0.828	0.913
1119	0.85	0.663	0.846	0.89	0.858	0.916
1122	0.775	0.682	0.844	0.881	0.851	0.914
1125	0.864	0.728	0.849	0.907	0.851	0.917
1128	0.499	0.629	0.649	0.804	0.828	0.897

Figure 14: DSC results per tissue type and different technique combinations using MNI Atlas.

The table above shows the DSC results from the segmentation using MNI atlas alone and combining MNI atlas with EM algorithm. Except for some images, the scores for Grey Matter and CSF were close to the results obtaines from our atlas; however White Matter index only reached an average of 0.7 when using the atlas alone. Fortunately, the combination with the EM algorithm has helped to significantly raise the values for White Matter.

In comparison, the atlas built from the train data segments better the test images which is logical. In addition, the MNI brain is built from large samples that are not from the dataset of the test images and therefore it is more generic.

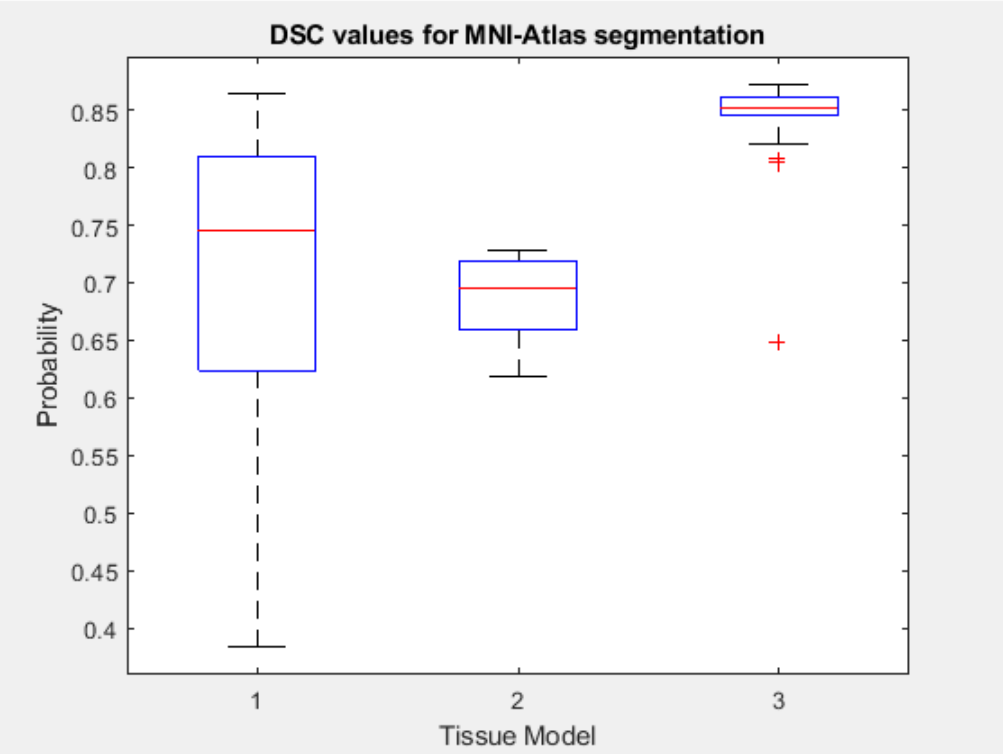


Figure 15: Boxplot for Dice coefficient for test images using MNI Atlas segmentation

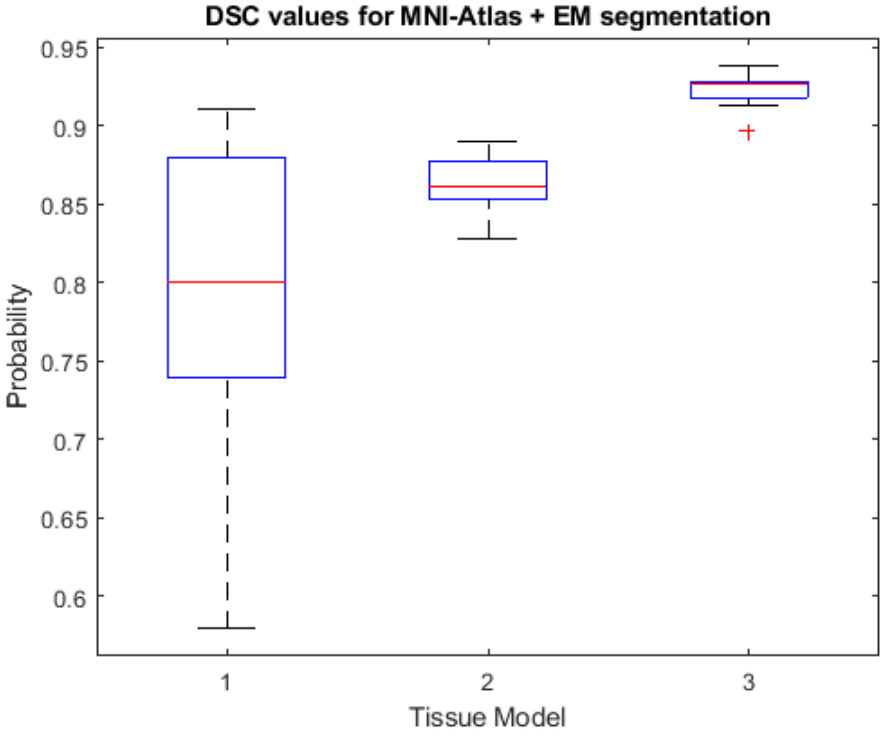


Figure 16: Boxplot for Dice coefficient for test images using MNI Atlas + EM segmentation

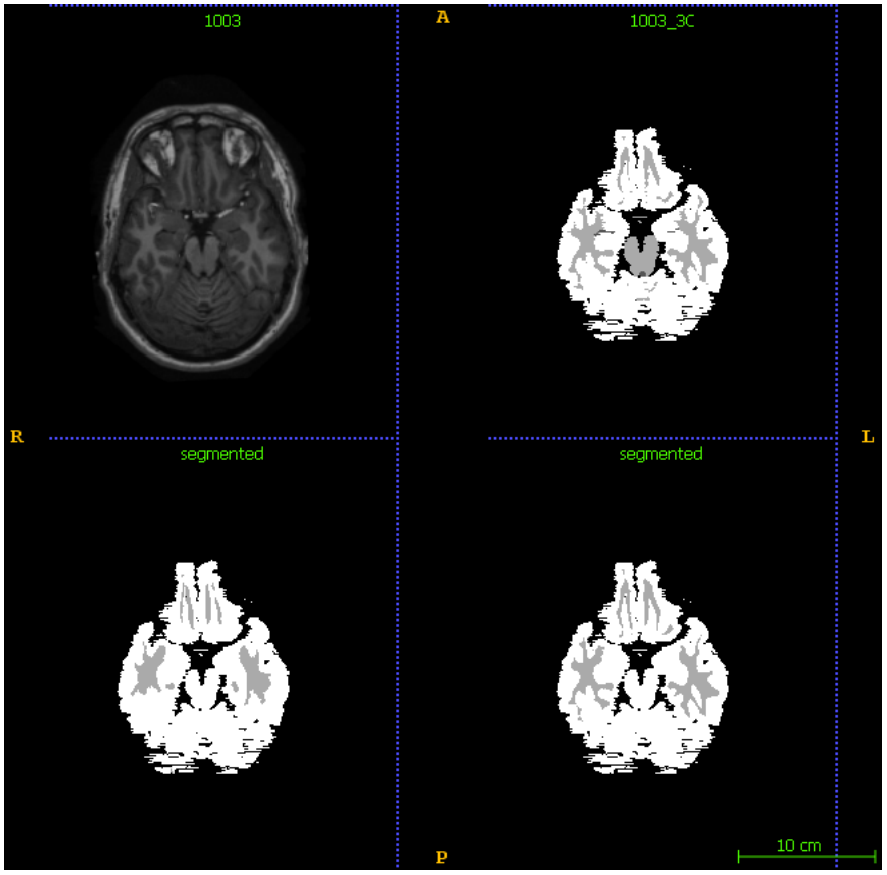


Figure 17: From left to right, top to bottom: intensity image, ground-truth segmentation, MNI segmentation, MNI + EM segmentation

9 Project Management

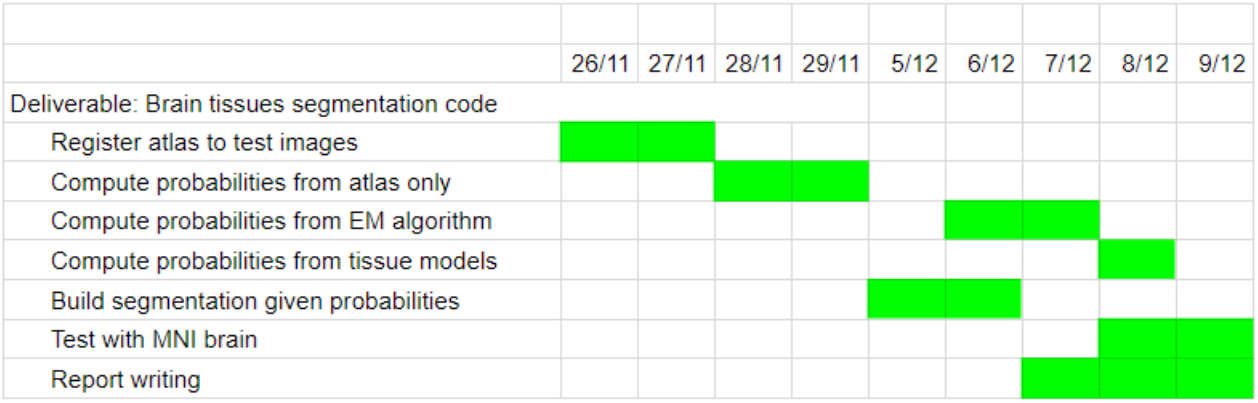


Figure 18: Gantt chart

10 Conclusion

Image segmentation is often the one of the initial steps for image analysis and is a key basis of many higher-level activities such as visualization, compression, medical diagnosis and other imaging applications. There exist several approaches to segment an image. Some of these techniques include edge based, region based, clustering based and many others. The goal of this coursework was to combine the use of a probabilistic atlas with other techniques, namely the EM algorithm and information from tissue models. Our experiments showed significant improvement on the segmentation when using a combined means of segmentation compared to atlas only. The integration with the EM algorithm in our case was more efficient than tissue models. Also the atlas we have built from the train data gave higher Dice index compared to the publicly available MNI atlas. Nevertheless the latter did well overall and its results could be improved with better prepossessing especially in terms of intensities adjustment.

References

[1] Rolf Heckemann. Segmenting brain images with maper. `https://soundray.org/maper/`. Accessed: 2018-11-02.

[2] William R Crum, Thomas Hartkens, and DLG Hill. Non-rigid image registration: theory and practice. *The British journal of radiology*, 77(suppl_2):S140–S153, 2004.

[3] Stefan Klein, Marius Staring, Keelin Murphy, Max A Viergever, and Josien PW Pluim. Elastix: a toolbox for intensity-based medical image registration. *IEEE transactions on medical imaging*, 29(1):196–205, 2010.

[4] Anita Khanna, Meenakshi Sood, and Swapna Devi. Us image segmentation based on expectation maximization and gabor filter. *International Journal of Modeling and Optimization*, 2(3):230, 2012.